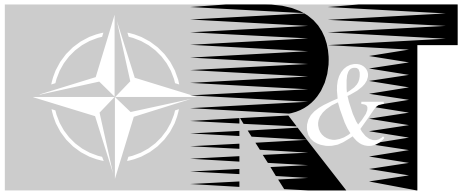
RTO-MP-064
AC/323(IST-022)TP/11

RTO-MP-064

NORTH ATLANTIC TREATY ORGANISATION



RESEARCH AND TECHNOLOGY ORGANISATION

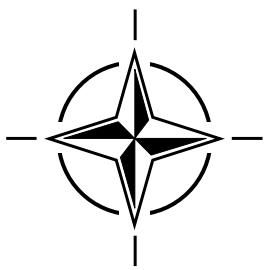
BP 25, 7 RUE ANCELLE, F-92201 NEUILLY-SUR-SEINE CEDEX, FRANCE

RTO MEETING PROCEEDINGS 64

Information Management Challenges in Achieving Coalition Interoperability

(les Défis de la gestion de l'information dans la mise en œuvre de l'interopérabilité au sein d'une coalition)

Papers presented at the RTO Information Systems Technology Panel (IST) Symposium held in Quebec, Canada, 28-30 May 2001.



Published December 2001

Distribution and Availability on Back Cover

This page has been deliberately left blank

Page intentionnellement blanche

RTO-MP-064
AC/323(IST-022)TP/11

NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION

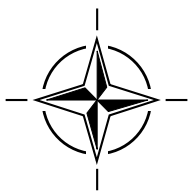
BP 25, 7 RUE ANCELLE, F-92201 NEUILLY-SUR-SEINE CEDEX, FRANCE

RTO MEETING PROCEEDINGS 64

Information Management Challenges in Achieving Coalition Interoperability

(les Défis de la gestion de l'information dans la mise en œuvre de l'interopérabilité au sein d'une coalition)

Papers presented at the RTO Information Systems Technology Panel (IST) Symposium held in Quebec, Canada, 28-30 May 2001.



The Research and Technology Organisation (RTO) of NATO

RTO is the single focus in NATO for Defence Research and Technology activities. Its mission is to conduct and promote cooperative research and information exchange. The objective is to support the development and effective use of national defence research and technology and to meet the military needs of the Alliance, to maintain a technological lead, and to provide advice to NATO and national decision makers. The RTO performs its mission with the support of an extensive network of national experts. It also ensures effective coordination with other NATO bodies involved in R&T activities.

RTO reports both to the Military Committee of NATO and to the Conference of National Armament Directors. It comprises a Research and Technology Board (RTB) as the highest level of national representation and the Research and Technology Agency (RTA), a dedicated staff with its headquarters in Neuilly, near Paris, France. In order to facilitate contacts with the military users and other NATO activities, a small part of the RTA staff is located in NATO Headquarters in Brussels. The Brussels staff also coordinates RTO's cooperation with nations in Middle and Eastern Europe, to which RTO attaches particular importance especially as working together in the field of research is one of the more promising areas of initial cooperation.

The total spectrum of R&T activities is covered by the following 7 bodies:

- AVT Applied Vehicle Technology Panel
- HFM Human Factors and Medicine Panel
- IST Information Systems Technology Panel
- NMSG NATO Modelling and Simulation Group
- SAS Studies, Analysis and Simulation Panel
- SCI Systems Concepts and Integration Panel
- SET Sensors and Electronics Technology Panel

These bodies are made up of national representatives as well as generally recognised 'world class' scientists. They also provide a communication link to military users and other NATO bodies. RTO's scientific and technological work is carried out by Technical Teams, created for specific activities and with a specific duration. Such Technical Teams can organise workshops, symposia, field trials, lecture series and training courses. An important function of these Technical Teams is to ensure the continuity of the expert networks.

RTO builds upon earlier cooperation in defence research and technology as set-up under the Advisory Group for Aerospace Research and Development (AGARD) and the Defence Research Group (DRG). AGARD and the DRG share common roots in that they were both established at the initiative of Dr Theodore von Kármán, a leading aerospace scientist, who early on recognised the importance of scientific support for the Allied Armed Forces. RTO is capitalising on these common roots in order to provide the Alliance and the NATO nations with a strong scientific and technological basis that will guarantee a solid base for the future.

Published December 2001

ISBN 92-837-1078-9

Printed by St. Joseph Ottawa/Hull
(A St. Joseph Corporation Company)
45 Sacré-Cœur Blvd., Hull (Québec), Canada J8X 1C6

Information Management Challenges in Achieving Coalition Interoperability

(RTO MP-064 / IST-022)

Executive Summary

Coalition interoperability is essential for coalition to have meaning. The information used by any coalition must be understood by all who need it, and the information must be effectively managed. With the expansion of NATO membership, and the peacekeeping roles now being undertaken by NATO it is evident that national defence systems must be capable of effectively inter-operating at many levels, and for a wide variety of purposes. This is not yet the case within NATO.

SYMPOSIUM

The three days of the symposium included 25 papers and an opening keynote presentation. The keynote speaker, Admiral Dyer of the US Navy, presented a challenging review of US approaches being taken to achieving information superiority and interoperability in his paper *Coalition Interoperability in an Information Centric Future*. A further paper from the US examining *Key Concepts for Information Superiority* also set the stage for many of the ideas and issues to be faced.

The symposium sessions were:

- Architectures And Standards: Fundamental Issues
  This session examined standards strategies, the use of meta-standards for information management, and codes of practice. It is clear that the status of standards for information management environments are still in need of substantial research.
- Information Management
  This is a major problem for coalition interoperability. Papers examined the use of ontologies and natural language for information management. The use of ontologies grounds the IM subject in the wider one of linguistics, and appears to offer much potential for alleviating NATO information interoperability problems.
- Mobile Software Technologies
  The use of agents for achieving adaptable coalition systems and the use of mobile interoperation standards such as JINI were described. Again this is a technical area that is likely to be exploited by the military for deployed operations.
- Interoperability Procedures And Practice
  The importance of interoperability testing, and the requirements to be imposed on COTS elements and standard were described and reviewed.
- Coalition Common Operating Picture
  Common pictures are a prerequisite to effective interoperability and three papers in this session addressed the issues of data fusion and systematic methods for COP construction and dissemination.

The conference provided some important insights into the problems of interoperability. The need for a strong relationship between the ambitions of information superiority and interoperability across similar systems and between communications and information systems was stressed. Some valuable emergent properties of the concepts, architecture and standards approaches for the next generation of systems were evident from the collection of papers presented, and the resultant discussions.

les Défis de la gestion de l'information dans la mise en œuvre de l'interopérabilité au sein d'une coalition

(RTO MP-064 / IST-022)

Synthèse

L'interopérabilité d'une coalition est une qualité essentielle car elle lui donne tout son sens. Les informations utilisées par toute coalition doivent être comprises par tous ceux qui en ont besoin, et doivent être gérées de manière efficace. Compte tenu de l'agrandissement de l'OTAN, et des missions de maintien de la paix qui lui sont actuellement confiées, il apparaît évident que les différents systèmes de défense nationaux doivent être capables d'interopérer efficacement à de nombreux niveaux, et pour une large variété d'objectifs. Or, ceci n'est pas encore le cas au sein de l'OTAN.

SYMPOSIUM

Les trois journées du symposium ont permis de présenter 25 communications et une présentation d'ouverture. Le conférencier principal, l'Amiral Dyer de l'US Navy, a effectué une présentation ambitieuse des méthodes US adoptées pour atteindre une certaine supériorité en matière d'information et d'interopérabilité, dans sa communication intitulée: *L'interopérabilité au sein d'une coalition dans un monde futur axé sur l'information*. Une communication ultérieure des Etats-Unis, sur *Les concepts clés de la supériorité de l'information*, a évoqué aussi de nombreux concepts et problèmes prévisibles.

Les sessions du symposium étaient les suivantes:

- Architectures et Normes: Questions fondamentales
  Cette session a examiné les stratégies en matière de normes, l'utilisation des meta-normes pour la gestion de l'information, et les codes de déontologie. Il est apparu clairement que des travaux de recherche importants restent nécessaires dans le domaine des normes pour les environnements de gestion de l'information.
- Gestion de l'information
  Ce sujet représente un problème majeur pour l'interopérabilité au sein d'une coalition. Les communications ont porté sur l'utilisation d'ontologies et du langage naturel pour la gestion de l'information. L'utilisation d'ontologies a placé le sujet de la gestion de l'information dans le contexte plus large de la linguistique et semble offrir un potentiel plus important pour résoudre les problèmes d'interopérabilité de l'information de l'OTAN.
- Technologies de logiciels mobiles
  L'utilisation d'agents pour la réalisation de systèmes de coalition adaptables et l'utilisation de normes d'interopération mobile comme JINI ont été décrites au cours de cette session. Il s'agit, encore une fois, d'un domaine technique susceptible d'être exploité par les militaires lors d'opérations de déploiement.
- Procédures et techniques d'interopérabilité
  L'importance des essais d'interopérabilité et les conditions exigées pour les normes et les éléments COTS ont été décrits et examinés lors de cette session.
- Modèle opérationnel commun pour une coalition
  Les modèles opérationnels communs (COP) sont une condition préalable à une interopérabilité efficace et trois communications de cette session ont abordé les questions de fusionnement de données et de méthodes systématiques pour la construction et la diffusion des COP.

La conférence a permis d'apporter des éclairages importants sur les problèmes d'interopérabilité. La nécessité d'une relation forte entre les ambitions d'interopérabilité et celles de supériorité de l'information sur des systèmes identiques et entre les systèmes de communication et d'information a été soulignée. Des approches intéressantes des concepts, des architectures et des normes de la prochaine génération de systèmes ont été mises en évidence dans l'ensemble des communications et des discussions qui en ont résulté.

Click inside the blue boxes or on the titles to view the corresponding section

(except for items marked in red, which were not available at the time of production)

Contents

† Paper not available at time of production.

Theme

Since the Gulf war, NATO nations have been involved in active coalition operations. In planning future C4I systems, coalition interoperability has always taken second place to national priorities. There is increasing recognition, especially amongst European NATO nations, that the majority of military operations must be conducted in coalition terms. This places new and more important requirements on the interoperability needed for such operations. In particular interoperability is a problem where coalition countries have different levels of technological advancement within their command and control systems. This raises the need to establish as widely as possible the 'lowest common denominator' which represents the interoperability baseline within that coalition.

Interoperability of C4I systems is fundamentally important for the conduct of coalition operations. The recent addition of new European partners within the alliance serves as a reminder of how difficult it is to predict who our partners will be within the next coalition. An overall coalition plan is needed that provides coalition interoperability plans for command, control, and communications systems, i.e. a secure coalition interoperability framework. This framework should cover NATO needs for interoperability plans effective in three broad time periods: present day, short-term, long-term. This symposium will address NATO interests and issues in developing such an information management framework – a crucial step toward achieving coalition C4I interoperability.

TOPICS TO BE COVERED:

1) Maintaining secure interoperability
2) Command system interfaces:
   - 2a) Information, data and service description languages
   - 2b) Information and service exchange mechanisms
   - 2c) Structures for information and system management
   - 2d) Management interoperability gateways
3) Coalition common operating picture
4) Communications interoperability
5) Command system adaptability/management with low performance communications

Thème

Depuis la guerre du Golfe, les pays de l'OTAN sont régulièrement engagés dans des opérations menées en coalition. L'interopérabilité au sein d'une coalition a toujours été considérée comme un élément secondaire de la planification des futurs systèmes C4I, qui a toujours privilégié les priorités nationales. Or, il est de plus en plus admis, et surtout par les pays européens de l'OTAN, que la plupart des opérations militaires seront désormais conduites au sein d'une coalition. Ceci augmente et crée de nouveaux besoins en matière d'interopérabilité pour ces opérations. L'interopérabilité pose un problème particulier si les systèmes de commandement et contrôle des différents pays d'une coalition accusent des niveaux de développement technologique différents. Il s'ensuit qu'il est nécessaire d'établir d'un commun accord le "plus petit dénominateur commun", qui représentera l'élément de base de l'interopérabilité au sein d'une telle coalition.

L'interopérabilité des systèmes C4I est d'une importance fondamentale pour la conduite d'opérations au sein d'une coalition. L'arrivée récente de nouveaux partenaires européens au sein de l'Alliance nous rappelle à quel point il est difficile de faire des prévisions sur l'identité de nos partenaires dans de futures coalitions. La nécessité se fait donc sentir d'un plan global pour les opérations en coalition comprenant des schémas d'interopérabilité pour les systèmes de commandement, contrôle et communications, c'est-à-dire d'une organisation sécurisée pour l'interopérabilité au sein de la coalition. Ce cadre doit couvrir les besoins de l'OTAN en matière de schémas d'interopérabilité sur trois grandes périodes temporelles, à savoir le présent, le court terme et le long terme. Ce symposium examinera les intérêts et les enjeux pour l'OTAN du développement d'un tel cadre de gestion de l'information, qui représente une étape critique vers l'obtention de l'interopérabilité C4I au sein d'une coalition.

SUJETS À EXAMINER :

1) Le maintien d'une interopérabilité sécurisée
2) Les interfaces des systèmes de commandement :
   - 2a) L'information, les données, et les langages de description des forces armées
   - 2b) Les mécanismes
   - 2c) Les structures pour la gestion de l'information et des systèmes
   - 2d) Les passerelles de gestion de l'interopérabilité
3) La description des opérations communes au sein d'une coalition
4) L'interopérabilité des communications
5) La gestion/l'adaptabilité des systèmes de commandement dotés de communications de qualité médiocre

Information Systems Technology Panel

ACKNOWLEDGEMENTS/REMERCIEMENTS

The IST Panel wishes to express its thanks to the RTB members from Canada for the invitation to hold this Symposium in Quebec and for the facilities and personnel which made the Symposium possible.

Les membres de la commission IST remercient les membres du RTB du Canada pour leur invitation à tenir cette réunion à Québec ainsi que pour les installations et le personnel mis à disposition.

This page has been deliberately left blank

Page intentionnellement blanche

Technical Evaluation Report

A J Alston
Defence Evaluation & Research Agency, Malvern
Concepts and Integration, C2 and Information Infrastructure
St Andrews Road, Malvern
Worcestershire WR14 3PS
United Kingdom

1 Introduction

1.1 Background

1.1.1 This paper constitutes the Technical Evaluation Report for the symposium on Information Management Challenges in Achieving Coalition Interoperability held in Quebec City from the 28th May to 31st May 2001.

1.1.2 This report is of four parts:

a. This Introduction.
b. Technical Content. A technical overview of the symposium consisting of the high and low points, an analysis of the papers against the objectives of the symposium and comments on specific papers.
c. Threads of Ideas. A discussion of four ‘threads of ideas’ that ran through the majority of the papers presented.
d. Conclusions.

1.2 Arrangements a the Symposium

1.2.1 The symposium was held in the Radisson in Quebec City; with most of the presenters and attendees also staying there. The conference facilities were excellent, and the close proximity of the accommodation and its location near to the centre of beautiful Quebec City ensured that all had a very easy and enjoyable stay.

1.2.2 The conference hall was excellent, with only a very minor detail of the position of the screen presenting some viewing problems to those at the back.

1.2.3 The administrative arrangements were excellent, with again only a minor comment on the lack of notice of those papers that were presented in French. This caused a delay in the proceedings whilst attendees retrieved listening devices.

2 Technical Content

2.1 General Comments

2.1.1 The keynote address from Vice Admiral Dyer was excellent and provided the context for the whole symposium. It was unfortunate that on such a broad topic as Information Management that there were not more Keynote Addresses; in particular the excellent and extremely relevant paper by Dick Hayes, “Key Concepts for Information Superiority”, presented on the Thursday morning. This paper suffered from the time restriction imposed and could have benefited from the extra time allocated to Keynote Addresses.

2.1.2 Generally the presentations were of a high standard. However, not all the papers were relevant to the topic of the symposium, and in particular few actually addressed the major topic of Coalition Operations. Others were let down by being too academic and making too many inappropriate assumptions in order to justify their thesis; but this will be discussed later.

2.1.3 Finally, there was not an even spread of papers across the symposium topics. Figure 1 details this.

2.2 Analysis of the Papers

2.2.1 The relevance of the papers was assessed by indicating which of the symposium key topics, as identified in the distribution pack, was covered by each paper. Two additional topics have been added to this list to cover areas that a number of papers concentrated on, namely “Concepts and Architectures for Coalition Operations” and “Specification, Design and Testing for Coalition Operations”.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Maintaining secure interoperability												x									x				
Information, data and service description languages	x	x		x	x			x	x	x												x			
Information and service exchange mechanisms									x		x	x										x			
Structures for information and system management					x				x	x	x	x										x			
Management interoperability gateways					x																				
Coalition common operating picture											x												x	x	x
Communications interoperability														x											
Command system adaptability/management with low performance communications														x											

Concepts and Architectures for Coalition Interoperability	x		x															x	x	x					
Specification, design and testing for Coaltion Interoperability						x	x						x		x	x	x			x					

Figure 1 Analysis of the Papers

2.2.2 Some key issues can be extracted from this analysis:

a. As has been mentioned there were a number of papers on the emergent topics of “Concepts for Coalition Operations” and “Specification, Design and Testing for Coalition Operations”. These reflect two issues high on researchers agenda: the lack of a well understood concept for interoperability in NATO and the importance of considering interoperability at all points in the engineering lifecycle, from definition of operational need through to testing. Both these issues are addressed later in this report.

b. There was only one paper on security, “Netcentric Warfare for dynamic Coalitions: Implications for Secure interoperability”, presented on the Wednesday. Judging from the large number of questions which followed this presentation it is obviously a topic of great interest and one perhaps deserving its own symposium in the future.

c. There were few papers covering the communication aspects of coalition interoperability. This perhaps reflects the research communities perception that the problems of information management are greater than those presented by networking disparate communications systems.

d. There seemed to be a poor understanding of the concept of a Common Operational Picture. Those papers which directly addressed this topic, and there were only three, all concerned themselves with the issues of real time air picture generation. Such a product is but one of the many information products that should be available to the operational user, and contributing to what has been termed the common operational picture.

2.3 Highlights

2.3.1 Two papers that attracted the highest praise in the post-symposium questionnaire were: “Information Interoperability and information Standards in NATO C2” by E Lasschuyt and M Van Hekken and “Ontologies for Coalition Operations” by A-C Boury-Brisset. These two papers lie at the heart of coalition interoperability and address the issue of how to ensure understanding of the data and information that is passed between systems. The two papers approach the problem from two different directions: the first the traditional approach of constructing large data models and the second the approach of defining a language through which aspects of the military operation can be discussed. These reflect the old and new approaches to solving the information exchange problem and are discussed later.

2.3.2 Another two papers that attracted praise were “Key Concepts for Information Superiority” by D S Alberts and R Hayes and “Network Centric Operations: Implications for Allied and Coalition Operations” by H E Keus. The first presented some of the latest thinking on Network Centric Warfare from the US and was one of the most thought provking of all the papers presented. The second called for, and presented the initial thoughts, on such a concept for NATO. It was obvious that without a concept for achieving coalition interoperability, and one that links to the US Network Centric Warfare concept, true interoperability cannot be achieved. This theme is discussed further later.

2.4 Lowlights

2.4.1 The paper “Standard or Standards? Some Issues to be considered in the use of Meta-data for Coalition Operations” by Braim, Miles, Furze and Peck suffered from being presented by a ‘third party’. It also suffered from discussing an ontological approach to interoperability that is not fully accepted by the research community. Also the use of the Dublin Core as the starting point for this work could be criticised as not being appropriate for MoD work.

2.4.2 The paper “Formal Approach to the interoperability of C4ISR Operating within a Coalition” by Bares suffered from too many inappropriate assumptions underpinning the main thesis; that it is possible to generate a formal mathematical approach to the quantification of a system’s interoperability. This is discussed further later.

2.4.3 “JINI in Military Systems Applications” by Wilkinson, Hains and Williams, attracted unexpected criticism. This was one of the few papers detailing a demonstrated technical solution and was perhaps too specific in its scope for most the audience, although it is a solution that has a very general applicability.

2.4.4 Finally, the paper “Coalition Requirements for Shared Situational Awareness” by Stewart, Pierre Collinson Evans and Harrison was criticised for two reasons: firstly the majority of the briefing was dedicated to explaining how to compose a workshop, which was of no relevance to the symposium, and secondly the only finding coming out of the workshops in connection with shared situation awareness was that the absolute minimum information should be shared. This was an unacceptable message.

2.5 Specific Comments

2.5.1 Interoperability should not be treated as just a technological issue. In the past, and the UK is as much to blame as anyone else, systems have been designed to meet specific military functional requirements with interoperability coming a poor second. Interoperability must be considered throughout the engineering lifecycle.

2.5.2 The solutions to interoperability presented at the symposium tended to rely on homogeneity in terms of a common approach or a single system or single data model to solve the problem of interoperability. These solutions ignore the fact that we live on a heterogeneous world. Even if a single nation could achieve a single technical solution that could be fielded, when operating in a dynamic coalition they would still have to interoperate with allies using different systems. It is important to design and build for heterogeneity and change from the start.

2.5.3 The Keynote address contained many messages, but one that stuck in my mind was the scale of operations. Whilst we talk about making our systems interoperable with allies we tend to think of NATO and perhaps even just two or three members within it. However, as Vice Admiral Dyer stated in the last four operations the number of coalition partners has ranged from 19 to 31. This brings a whole new meaning to the term Coalition Interoperability. In addition, as stated by McIntyre in his paper “Netcentric Warfare for Dynamic Coalitions: Implications for Secure Interoperability”, the coalitions likely in the future will be dynamic in nature where members do not have long standing working relationships. These are significant challenges to achieving coalition interoperability.

2.5.4 Finally, whilst the concept of Network Centric Warfare brings with it many operational benefits, it must not be seen as the panacea for all interoperability problems. There will be practical problems building it and opening it to coalition partners. As Dick Hayes stated when presenting his paper, “Key Concepts for Information Superiority”, ‘..sharing lies at the core of Information Superiority - and the entry fee is the Global Information Grid.’. If this entry fee is too high it will inhibit interoperability not facilitate it.

3 Threads of Ideas

3.1 Introduction

3.1.1 Throughout the papers there seemed to be four common threads that warrant a little more investigation:

a. Concepts and Architectures
b. Data Models and Ontologies
c. Designing for interoperability
d. Technologies for interoperability

3.2 Concepts and Architectures

3.2.1 As the US is rallying round the Network Centric Warfare concept, it is clear that NATO requires something similar. This was suggested by Keus in his paper “Network Centric Operations : Implications for Allied and Coalition Operations”. However, the model must be able to cope with heterogeneity (not all members of the coalition will have the same equipment), cost of membership must be low and it must be dynamic, both in terms of ‘minute to minute’ membership but also in terms of who can become a member.

3.2.2 To enable such a concept a clear definition of interoperability is required. At a general level, Gentlemen in his paper, “Planning for Interoperability”, stated that it was about ‘getting things to work together that weren’t planned to operate together.’ More specifically there was a general consensus that interoperability is not just about getting computers to talk together but also about information and knowledge sharing between operators and process sharing between organisations.

3.3 Data Models and Ontologies

3.3.1 It appears that we are at a cross-roads in our thinking on how to achieve data interoperability. Whilst all seem to agree that it is better for systems to use different data formats internally and externally (Gentlemen, “Planning for Interoperability”), there is a divergence between the ‘old school’ of very large heavily structured, battlespace-wide data models and the ‘new school’ where loosely structured, domain based ontologoies and meta languages rule. Whilst the latter may seem to offer a more pragmatic and flexible solution to data interoperability (a speaker from the floor stated that the vast majority of message traffic in the Kosovo Operation was unstructured) the thinking is still at an early stage and further research is required. However, whilst ontologies and meta models seem to offer a way forward, we must not throw away the vast amount of effort expended generating data models, like ATCCIS. Research is needed to investigate the relationship between data models and ontologies and how to migrate between the two.

3.4 Designing for Interoperability

3.4.1 As previously stated, Systems are still designed to meet national requirements, with coalition interoperability treated very much as a secondary, bolt on requirement at the design and implementation phase. Interoperability must be considered from the start of the system lifecycle, with an understanding of the operational processes and operational need associated with coalition operations. To support this there must be a method of sharing design products (for example, Operational Information and Technical Architectures) between nations, perhaps using the US C4ISR Architecture Framework or the NATO NOSE. Finally, there needs to be a means of testing whether interoperability has been achieved; as Amanowicz stated in his paper “The Role of NATO C3 Interoperability Testing Infrastructure to Establish the Polish Interoperability Architecture”, ‘..standards do not guarantee interoperablity - it is their implementations’.

3.4.2 However, one note of caution. Some papers suggested that there is a deterministic formalism to interoperability. For example Bares in his paper “Formal Approach of the Interoperability of C4ISR Operating within a Coalition”, suggested that levels of interoperability could be calculated and Dorion stated in his paper, “On the Performance of Military Information Systems Built in CORBA”, that the performance of Information Systems can be formulated directly in terms of Military Effectiveness. This is not supported work done in the UK on Emergent Properties and by Voas in his paper Challenges for Software Interoperability, where he suggested that it may never be able to predict the performance of component based architectures. Other work in the UK has suggested that traditional top-down, hard engineering approaches are not suitable for the engineering of large system of systems but ‘softer’ techniques are required, like Checkland and Wilson’s Soft System Methologies.

3.5 Technologies for Interopability

3.5.1 The technologies for coalition interoperability presented at the symposium broadly fell into two categories:

a. Those that enable sharing of information between all partners, most notably Network Centric Warfare, as presented by Hayes in his paper, “Key Concepts for Information Superiority” and Keus in “Network Centric Operations: Implications for Allied and Coalition Operations,” and the use of ontologies as presented by Boury-Brisset in her paper “Ontologies for Coalition Operations”, Lasschuyt in his paper “Information Interoperability and Information Standardisation for NATO C2 - A Practical Appraoch” and Farrington in the paper “Standard or Standards? - Some Issues to consider in the use of Meta-Data for Coalition Operations”.

b. Those that allow one participant to gain access to information of others. These were exemplified by technologies such as JINI in Haines’s paper “JINI in Military Applications” and Labbe’s paper “An Agent-Based Approach for achieving Interoperable and Adaptable Military Coalitions”.

3.5.2 Whilst the goal must be to have systems that share information between them, as much as possible must be a made of the legacy systems by extracting and reusing the information they contain, using the technologies discussed under the second category.

3.5.3 The use of COTS was only briefly touched on, Jarmakiewicz in his paper “The requirements for COTS IPv6 Network Application in Tactical Network Environment” and McIntyre in his paper “Netcentric Warfare for Dynamic Coalitions: Implications for Secure Interoperability”. The general consensus was that in the short to medium term COTS will be the foundation for future systems but will always require some degree of enhancing for military operations.

4 Conclusions

4.1 The symposium was very successful with many interesting and entertaining papers all administered in an efficient and effective manner.

4.2 Examining the main threads of thought through the symposium raises three potential topics for further investigation:

a. NATO/Coalition Concept for Interoperability, heavily based upon the US Network Centric Warfare concept and including the topic of how to design and test for interoperability.

b. Information Management Standards, in particular the future role of ontologies and meta-languages and their relationship to and migration from data models

c. Security for Coalition operations, in the context of the two topics above.

Planning for Interoperability

W. Morven Gentleman
Global Information Networking Institute
6050 University Avenue,
Halifax, NS, B3H 1W5
Canada
Morven.Gentleman@dal.ca

Summary

Interoperability is a major concern for NATO. In addition to the focus of this meeting, on facilitating coalition operations, interoperability is crucial to several other themes of research undertaken by the IST Panel, such as use of COTS products in larger systems or following an Evolutionary Software Development process. In coalitions, the C2 systems of one nation may want to take advantage of information and services available from the systems of other nations, correspondingly they may want to make available information and services to the systems of other nations. The challenge is that because of the independent development, these exchanges may not fit directly, but must accommodate the different perspectives that have been taken on the abstractions they represent in the different systems.

Introduction

Interoperability means that different systems can be used together, and in order to plan for interoperability, we need to think about the sense in which the systems will be used together. In this paper we will focus on software issues, however many hardware issues of interoperability can be viewed in the same framework. We will consider a *system* to refer to either one or more distinct application programs, or a collection of related objects in the sense of object oriented programming. The interaction when systems are used together can be as subtle as in the way activities are viewed by the user, or as the vocabulary with which entities are described, but the problems we will address here are at a more concrete nature. We will assume that some hardware is shared by the interoperating systems. Again, in general, when systems are used together, the hardware they share might not even be connected to computers, as when the different systems are assigning tasks to vehicles drawn from a common fleet. However we will restrict our attention to software systems that are running on the same computer, or on computers that communicate with each other over the same network, or at least on computers that read and write to the same storage media or displays. Systems that are used together must consider each other as unreliable, that is, each must allow for the failure of the other systems. Even if each system behaves exactly as it was designed to behave, the interactions among the systems may be unknown. Moreover, the behaviour of a system may not be understood correctly by other systems and their users, so the system may be misused.

The interoperability problem is that different systems need to be used together, not just each used in isolation. Sometimes these systems are complementary, each relying on others to

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

provide capabilities that the one system does not itself possess. Sometimes these systems are supposed to be comparable alternative components, providing “equivalent” capability where different users might pick a different alternative, yet expect the usage to be interchangeable. Sometimes just the ability to use different systems simultaneously in the same space is a challenge, even though there is no inherent reason for the systems to interact.

The essence of the interoperability problem is that systems typically are designed and developed independently, and that they evolve independently subject to different influences. Thus at any time, each system reflects decisions that were made in the context of that system in isolation, and these decisions may conflict with the corresponding decision made in some other system with which this one needs to interoperate. The complexity of planning for interoperability lies not just in the difficulty of recognizing these conflicts and resolving them, nor even in attempting to avoid conflicts arising in the first place (or in the future, for systems that currently interoperate without conflict) among the known interoperating systems. The intrinsic problem is that the designers of a system cannot foresee all the other systems with which their system may eventually be expected to interoperate. Without some overarching architecture, then, how can a single system be designed so as to reduce the risk of interoperability clashes with other systems, with which the need for interoperability is not yet recognized?

Three levels of interoperability

Interoperability exists at three levels: co-existence, interchange, and integration.

Co-existence

The least demanding level of interoperability is co-existence: systems that do not need to interact should be able to operate simultaneously in the same space without interfering with each other. The prevalent requirement is for co-existence with unknown other systems. Even at this level failures are commonplace. The primary cause for systems failing to coexist is that they make incompatible assumptions about some common resource in the environment that they share.

There are many types of such resources. Symbolic names are a particularly important example. A system uses symbolic names to bind it to resources which it needs from its environment, such as files, communications channels, devices, other parts of this system, other systems, environment variables, or environment services. To make the bindings more natural, the symbolic names chosen are often simple and obvious. Unfortunately, many environments, such as Unix or Windows, provide only a single flat name space shared by all systems, and hence clashes between what one system uses a name for and what another system uses the same name for are common. Clashes can be avoided by conventions as to name format, but conventions are not enforced and so are error prone. Some environments finesse the problem by using scoping mechanisms, so that names have a particular context, as is normal in modern programming languages. The Internet has effectively solved the

problem by multi-part names, such as domain names or Universal Resource Locators (URL), where there is regulatory control over the definition of the most significant parts of names.

Resource exhaustion is another typical conflict: Processor cycles, main memory, working backing store, persistent backing store, communications bandwidth, and real time are all resources normally shared by systems running in the same environment. Nevertheless, excessive consumption of any of these resources by some systems can starve other systems of enough of the resource to perform satisfactorily. (Some of this effect can be avoided, or at least alleviated, if the systems run on separate computers in a networked environment.) Of course the underlying problem may be that in some circumstances there simply is not a sufficient amount of the resource, but the effect impacts systems other than the one with exceptional resource demand. Ideally, what we would like to do is to bound the amount of resource a single system can claim, so there will be sufficient resource for other systems. Within its resource bound, the system should manage the resource, perhaps using alternative implementations with different resource usage. The bound should be represented as a parameter, so the system can adjust to whatever value is set for the bound. The value of the bound can be set manually, and tuned to reflect in-use conditions. Alternatively, in some environments the system can interrogate the environment for how much resource is available and what contention there is for it, then use some algorithm to choose an appropriate value for the bound. It is helpful to document the resource requirements of a system, preferably so it can be interrogated at run time. When it is not possible to bound the resource consumption, it is especially important that we still design the system to minimize its consumption of the resource.

Yet another typical conflict occurs with resources that cannot be effectively shared over brief time intervals and so must be allocated to one specific system at a time. For HCI reasons, input devices such as keyboards or pointing devices need to maintain a focus of attention to communicate with a single “current” system at a time, although over longer intervals that focus can shift to another system. Similarly, an audio output channel normally needs to be reserved for use by one system at a time. A display screen can be shared by several systems through the use of windowing techniques, but if a complex visualization is required, screen real estate quickly runs out, and the display may need to be treated as effectively reserved for use by a single system. In all of these cases, providing redundant resources to a single operator is not a satisfactory solution, because of HCI factors. Multiple keyboards, for instance, would be awkward and confusing. Thus contention for this kind of resource needs to be designed for by suitably delimiting the time during which focus is directed toward a specific system. Establishing clearly which other system focus has moved on to when it does shift is necessary but not sufficient. If that new context does not seem natural to the operators, the system will be error prone because they will by default assume what they think is the natural context, particularly under stress.

Mind space of operators is a resource which a greedy system can overconsume: Common conventions need to be established for any shared resource, for example we are asking for trouble if all applications use the joystick in velocity mode except one which uses it in acceleration mode. Whether it is screen conventions or error message handling, or simply the cognitive model of how services are thought about and directed, one system can preempt the stage and make it difficult for other systems to be used as well. A common characterization of such systems is that they suffer from “the center of the universe” syndrome. One such system may be tolerable if other systems can channel their needs through it. Two such systems are already unworkable. Today a widely used approach is to let a web browser take

on this central role. Although web browsers today providing a poorer user interface than windowing techniques can provide, it is usually good enough, and has the advantage of being familiar to more users.

Error recovery is one co-existence issue not directly attributable to conflicts over a shared resource. When there is a crash, it is often unclear what actually has failed, and frequently the users get lost. There is a big advantage to treating error handling and error recovery uniformly for all systems, so the action to be taken is the same whatever system actually failed.

All these interoperability failures are hard to anticipate because too often the assumptions, even the need by the different systems for the common resource, is not explicit, and the failures only manifest themselves in particular circumstances

Interchange

The next level of interoperability is interchange: not only must the interoperating systems co-exist, but they consume information produced by each other. Where this happens naturally it is wonderful. This typically only happens when the systems were designed together. More often, deliberate measures must be taken to achieve interchange. The most obvious problem is that the right information is available, but the representation of that information which a system produces differs from the representation required by the consuming system. The equally obvious remedy is to change either or both systems to use a common representation. This suggestion is all too often simplistic, for any of several reasons.

An atomic datum is usually little problem. The representation chosen can be precisely defined with ASN.1, and if the representations differ, mapping between them is straightforward. For compound data, however, because the systems use the information in different ways, and because the internal representation of the information may be determined by how it is used, each system may need a different internal representation. Data structures, for instance, may correspond to efficient processing algorithms; and sequencing of data items may reflect when data became available rather than when it is needed. Structure not needed by one system may be completely omitted in the internal representation used by that system, even though the structure could conceptually be present.

This, of course, only implies that a mapping may be required between the representation that the system uses internally and the external representation chosen for interchange. Care must be taken to ensure that performing this conceptually simple mapping does not imply prohibitive performance penalties. Even if a system cannot be changed, for instance where the system is a COTS product for which source code is not available, the mapping can be done in an external wrapper.

A more serious risk is that finding a single suitable external representation may not scale as more and more systems try to share the information. In particular, even when the core

information is the same, systems often produce or can exploit different ancillary information, and this ancillary information may be hard to fit into some forms of external representation.

It may be necessary to allow more than one representation for the same information. In particular, we may need to provide for representing incomplete data. While by definition incomplete data can not be directly usable in all situations, it may still have value, and in combination with other data the ambiguity may get resolved. As a simple example, indicating location of targets by giving Cartesian coordinates might be desirable, but often bearing or range is all that are available. Either of these alone cannot be represented if the standard representation is only Cartesian coordinates. Allowing both for Cartesian coordinates and for range and bearing representations accommodates incomplete data, moreover, it can simplify description of uncertainty. We shall return to this example later.

Another interesting example concerns interchanging Geographic Information System (GIS) information. GIS information is commonly represented in one of two different ways: as vectors, for example the coordinates of some point, or as rasters, for example the pixels in some photograph. The difference is not just superficial, but a deep difference in how the information is thought about. Vector representation is typically organized by features (a road, perhaps, a pier, or some important building) and pre-supposes those features exist and have been identified. Raster representation is typically just the sensor data, perhaps with annotations to suggest how pixels could possibly be interpreted. Multiple hypotheses might be considered, and a single pixel could participate in several pixel groups with different interpretations. The two representations are far from equivalent.

The producing system often has to flatten its internal data structures so that they can be represented externally as a single sequential file or as an unmultiplexed communications stream. At one time it was common for this file or stream to be made up of a sequence of fixed record (message) types, each with a fixed number and type of fields. The representation of the records and fields would be defined in a standard way, say by using ASN.1. The type of a particular record might be explicitly specified by one of its fields, or implicitly specified, perhaps by its position in the sequence. For ease of debugging and troubleshooting, an additional constraint was often introduced that the file had to be directly human readable, (as opposed to that that suitable display software could analyze, interpret, and provide a visualization of the data stream). The file or stream then could be published as a service to which other systems could subscribe.

In these older exchange formats, where message types were pre-defined, it was nevertheless often found valuable to have data type information redundantly represented in the data. This was called self-defining structure. Not only did this make possible interpretation of the data without reference to the pre-definitions, but it provided a validity check on the data as represented, and could assist in resynchronizing to the data stream if synchronization was somehow lost.

This style of information interchange could readily accommodate situations where different record types involved different amounts of data. However, since the set of record types was pre-defined, it was difficult to deal with unanticipated situations as they arose. This shortcoming ultimately led to the demise of the original EDI standard for electronic interchange of business information. More modern styles of information interchange, such as

XML, are based on list-processing languages. A list is a sequence of items, which may themselves be lists, so lists can be nested. A list type may again be specified by the value of some item in the list, or by its context. Lists can represent data. By means of a pre-defined list type, a list can be given a name, so it can be referred to symbolically thereafter, which enables representation of more complex data structures than simply sequences or trees. By defining processing operation semantics to be associated with other pre-defined list types, lists can specify computation to apply to other lists. Another pre-defined list type allows the definition of new list types. A great advantage of this style of representation is that it is extensible, providing the flexibility to cope with unanticipated situations by allowing new message types and new processing algorithms to be defined as needed. They also directly provide self-defining data structures, by requiring every normal message type to be defined. Appropriate choice of list type tags helps too.

The single synchronous stream of items, although simple, has disadvantages. Consuming systems may have to burn through much information of no interest to them in order to reach items they are looking for. It may be necessary for them to cache some of this apparently irrelevant information in case the information they want might make a back reference to something earlier. Moreover, reaching the information of interest may be delayed simply because of an arbitrary choice that the producing program has made in which way to walk its data structures when flattening them. These disadvantages can all be addressed by taking a different point of view: instead of direct transfer of information between the producer and its consumers, the producer can maintain a shared database that the consumers interrogate using random access. Note that although it sometimes is possible to maintain a single shared database of all information from all producers, it is not uncommon to find that maintaining the four ACID database properties (atomicity, consistency, isolation, and durability) across all of this is too restrictive. All that may be practical is something much weaker, that is, to maintain a federation of databases, each responsible only for its own ACID properties. This is also known as a data grid, and although the different databases often store different kinds of data, it is not unusual for them to be partially redundant. The schema adopted by the different databases may be incompatible with one another, and apparently similar data may have different semantics. Things are further complicated by implicit data which is not represented at all, but assumed and for which the assumptions of different systems may be incompatible. The database model puts the onus for querying on the consuming systems, but this does not necessarily imply polling for new information. A system can register with a database to be notified when the database has been updated by particular kinds of new information.

Beyond the advantage of random access, changing to the database model makes possible a dialog, whereby particular responses from the database can prompt interrogation from the consumer for more explanation, often described as “drilling down” into the data. It also makes possible dialog to ascertain exactly what representation the consumer wants. For example, for gridded GIS data, the consumer may want the data to be regridded to a particular grid other than the one used by the producer. The producer, which may have access to the raw information from which the grudded data was derived, can do this regridding with significantly less error can the consumer, interpolating from another grid.

One interesting aspect of information interchange, particularly in C2 systems, is that the information is not static but is time dependent. A producer can time-stamp his information as having been valid at a particular time, but it is even more useful if the information also provides an extrapolation forward in time, predicting what may be expected at some later

time. For this purpose the procedural capability of XML and other representations of that style can be invaluable.

Because information in C2 systems is time dependent, absence of data can itself be information too. Conversely, if the communications can be monitored by unfriendly forces, there is a risk of disclosing information simply by generating unusual traffic loads, even if the contents of that traffic are securely encrypted. A deliberately controlled dilution of the real traffic with noise to avoid this is essential to the representation.

The question of whether some system has information that could be useful to another system, or equivalently whether some system could on this occasion take advantage of information available from another system, is challenging, and leads to the requirement for meta information interchange, such as service descriptions.

Where large numbers of systems are to interchange information, something is needed to simplify and automate connecting them together. Most environments today facilitate configuring and binding I/O structures externally to the systems that use the I/O. We need to do this for our interchange connections. We also need automation to check the connections that have been set up. Plug-and-play wizards are used in other contexts to explore

Provision for error handling and error recovery is very desirable for interchange interoperability. Failures of the communications mechanism, flaws in the systems themselves, or misunderstanding of the interchange procedures can lead to error situations, and it is clearly better not to let these disable the systems. Particularly common forms of error are to have gaps in sequential data, or repetitions of the same data, or to have sequential data received out of order. If anticipated, these can be readily detected.

Integration

The most complete level of interoperability is integration: bigger systems constructed by combining other systems, which of course presupposes interchange. Gathering the information, especially when there are many sources is rarely done in a single interchange of information. Rather there are many interchanges of information, with some information required before other information can be accepted. Indeed some information may need to be processed before what additional information will be required (or even might be useful) will be known. We also need to specify what information persists across multiple interchanges, and what begins afresh with each interchange. We need to know the relationships, even just in time, between information from different sources. This all leads to the idea of protocols; that is, rules for sequential and concurrent transfer and processing of information. Protocols are not just the API of procedures that a subsystem uses. Typically procedure invocations must occur in a specific sequence. Threads must be identified for persistent data if multi-threading is possible. Thread abort must be arranged if that thread never completes. Gathering information from other systems about which little is known has many challenges. It is important to decide and clearly define just what has to be known about a source.

Obtaining information from other systems is not enough, however. The system that is to use the information needs to understand the information, even just to display it sensibly. The biggest challenge of interoperability at the integration level is that similar information from different sources is often not semantically equivalent to some degree. This is not a question of the representation of information, as addressed at the interchange level, but rather is an issue of the meaning of the information, indeed of what information there is. A C2 system offering a coalition commander the choice of two ostensibly equivalent units from different nations cannot ignore the tact that these units may be of different size, have different internal command structure, be equipped differently, trained differently, and may need to be supplied differently. If information on these units is drawn from separate sources, such ancillary information must come along. We observed earlier that even in as simple a situation as locating targets, different sensor systems provide different kinds of information. Integration interoperability intrinsically involves data fusion, combining somewhat related data to build a combined situation assessment. This can be considerably more complex than simply suggesting or taking action (even just displaying) in the situation, based on information from a single well-known source. It also can be more robust. It can accommodate incomplete and inconsistent data. It can allow for data uncertainties. It can maintain multiple hypotheses. It can take into account feedback and feedforward, as well as time-dependencies. Constraints, such as map features, can help resolve ambiguities. Sophisticated algorithms may be employed, but provision for human judgement is typically also required. Supporting human judgement may involve drilling down through an integrated situation display to appropriately visualize ancillary information from the specific source in order to assess its influence and to allowing manual annotation.

Integration of systems from coalition partners will be more effective if these systems have been internationalized so that they can be localized to the language and other cultural expectations of particular users. This is already common practice in commercial software.

It should be clear that the principal role of integration level software is to orchestrate the use of the available resources within an integration framework. This can entail such network computing aspects as concurrency, workflow, plug-ins, and multi-agent designs, with plug-and-play to discover what resources are available and how to connect them. Wrappers and glue are common. These things are not easily expressed in conventional programming languages. Even the paradigm of predicting outcomes with estimates that improve over time is unusual in conventional programming. For that reason, coordination languages, which include scripting and visual languages, may be more appropriate for the integration level". Debugging is also known to be hard, and requires tools such as history buffers and loopback tests, as well as tracking the effect of actuator directives. Security and authentication are particularly challenging to achieve in a framework that attempts to be open and "user-friendly.

The recognition that integration is possible, desirable, and maybe even necessary is already a key issue. Many technical details create integration awkwardness, especially between systems not design with interoperability as a consideration. As a simple example, software driven by an "event loop" is a standard way to implement state machines or graphical user interfaces. Multiple concurrent "event loop" and state diagram structures introduce technical difficulties and even ambiguities. As another example, systems expecting input from human operators today often attempt to be "user friendly" by allowing input in many different orders, and displaying output as it becomes available. Representing the necessary partial ordering on input is more difficult than even most protocol descriptions allow. Automatically

handling ambiguity or error situations is much harder than the usual single system solution of handing the situation over to a human to resolve. Fortunately, the increased tendency to build network based applications and rely on standard middleware such as CORBA has made the real practical problem more manageable. Security remains a technically intriguing issue, especially when the systems assume different security models.

Simulations may be the most effective way to perform “what if?” experiments to comprehend alternative strategies. Tolerating failure of requested actions is crucial to such simulations, as are missed operations or duplicate operations. An extensible user interface may be critical to using such simulations to review and annotate suggested alternatives.

Conclusions

Interoperability can offer a much richer choice of systems than is available in a closed scenario where all the systems that can interact are known in advance and designed together. To exploit this possibility, on the other hand, requires planning new systems in ways that have not traditionally been done.

URLs

ASN.1

http://asn1.elibel.tm.fr/

XML

http://www.w3.org/XML/

This page has been deliberately left blank

Page intentionnellement blanche

Standard or Standards? - Some issues to consider in the use of meta-data for coalition operations.

Mr. Jonathan Miles
Integrated Systems Sector
Defence Science & Technology Laboratory
Woodward Building, Room B105
St Andrews Rd
Malvern, Worcestershire, United Kingdom

Dr Stephen Braim
Integrated Systems Sector
DSTL, N103
St Andrews Rd
Malvern, Worcestershire, United Kingdom

Mr. Richard Furze, Dr Mathew Peck
New DERA
Woodward Building
St Andrews Rd
Malvern, Worcestershire, United Kingdom

Abstract

This paper discusses the issues that need to be addressed to allow the unambiguous exchange of information between coalition forces during multi-national operations. It analyses the problems caused by a lack of common meta-data vocabulary between nations.

The paper considers current initiatives to define meta-data standards within the UK, and describes a putative approach to yield inter-government interoperability. It then extends the work undertaken within UK and considers its potential to provide interoperability between nations operating within a coalition. Centralised, Decentralised and Federated approaches to meta-data interoperability are described and issues associated with each approach are highlighted. The Federated approach, consisting of a common core supplemented by local meta-data sets, appears to offer the best conceptual solution to coalition interoperability as it offers a common naming scheme tag set which provides additional extensibility and will allow for an interface to existing national naming scheme initiatives. This will provide a degree of centralised control with flexibility for individual organisational unit needs.

Keywords

Coalition, Meta-data, Dublin Core, Semantic, Semantics, Syntax, Information, Interoperability, Communication, XML, Schemas, Naming schemes, Namespaces, Initiatives.

Executive summary

The objective of this paper is to highlight issues associated with achieving meta-data interoperability between coalition partners during multi-national operations.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Meta-data interoperability offers the benefit of communicating the intended meaning of information (semantics) between distributed units belonging to multiple nations. This will help to reduce confusion and misinterpretation of command intent as it is distributed throughout the coalition units.

Heterogeneous systems that exist in each nation use meta-data terminology that has been developed using local initiatives. Attempts have been made to make the various nations systems interoperable e.g. Army Tactical Command and Control Information System (ATCCIS) with some success. However the ontology for such initiatives are hard to agree due to the complexity of organisational and system issues.

The development of a UK Defence Meta-data Standard, with an emphasis on representing information context and quality attributes is introduced. The initiative has demonstrated synergy between two different uses of the same meta-data set (context and quality); this will allow real benefits to be accrued across defence by the adoption and use of the same minimal set of meta-data. Further, the use of a simple set of guidelines has been suggested as a way of structuring the thinking behind the generation and the use of any (and every) information product.

Analysis has shown that there is a common dataset between differing implementations, suggesting that this “core” can be realistically used as a common set. It would seem appropriate to adopt this central set as a common core across UK Government, whilst at the same time allowing locally-defined, additional elements to be utilised. An important ingredient of this model is the small number of elements which are deemed “common”, but which still present a unified, and useful coherence to information provision and use.

The paper builds upon UK initiatives and considers three approaches to coalition information interoperability: -

A centralised approach will contain systems that can perform independently of one another if required to do so, but will normally operate as a subordinate part of the centralised system. The subordinate systems will be closely coupled together within the centralised system’s boundary. Complicated referential management processes will exist to ensure that interoperability is maintained. The references and associations between the individual systems will grow as nations join the coalition. This will make the management extremely difficult and hard to maintain. If a nation were to withdraw from a coalition the domino effect upon the whole centralised system’s references and associations may well become chaotic and unmanageable.

A decentralised approach containing many diverse systems with no overall control. This approach will allow nations to retain existing local naming schemes and tag sets and will satisfy local ownership issues as no common vocabulary will exist. However the systems will operate in a coercive management environment. Many independent management initiatives may occur on an ad hoc basis to satisfy local requirements of interfaces between systems. These will be difficult to co-ordinate and may result in duplication of effort and redundant processes.

A federated approach is a hybrid of the previous two approaches and will be managed by taking a pluralist management approach with a mixture of a central authority and voluntary collaborative initiatives working toward a common overall goal. Core common management processes will exist but the individual components will also retain local management processes for flexibility. A federated approach may reduce the complexity of the management and ontological problems that exist within both the centralised and decentralised paradigms.

It is shown that the initiatives within the UK can be extended to provide a possible means to surmounting coalition interoperability problems. The development of a common minimal tag set with the ability to interface with the local tag sets of coalition nations is suggested. This will allow for

common semantic meaning between all of the coalition participants and also allow for the retention of local semantic meaning within each nation. It will also allow for a minimal amount of centralised control within the coalition.

A possible technical solution based upon Extensible Markup Language (XML) with the use of Namespaces is also proposed.

Technical Team

Dr S.Braim, Defence Science and Technology Laboratory (DSTL), Malvern, UK

Mr R.Furze, Defence Evaluation & Research Agency (NewDERA), Malvern, UK

Mr J. Miles, DSTL, Malvern, UK

Dr. M.Peck, NewDERA, Malvern, UK

Glossary

Explicit: - Expressly stated, leaving nothing merely implied.

(Of knowledge, a notion etc.) Definite, clear.

Implicit: - Implied though not plainly expressed.

Ontology: - A vocabulary of basic terminology with precise definitions of what the terminology means and precise definitions of the relationships that exist between the defined terminology

Semantics: - Relating to the meaning in language.

Syntax: - The grammatical arrangement of words showing their connection and relation.

Naming Scheme: - A naming scheme is used to associate names to specific objects that exist within a particular domain for unambiguous identification purposes. The naming scheme is made more efficient by grouping objects that share common attributes into classes and sub-classes.

Namespace: - A namespace is used within Extensible Markup Language (XML) and allows an objects meaning to be defined and accessed by means of its uniform resource locator (URL). The XML file points to the definition of the object and allows an application to interpret the definition when conveying information about the object to a user. This prevents the need of having to repeatedly send the definition of the object with each communication and allows many files to share a common namespace definition with the individual element descriptions.

Contents

List of figures

List of Tables

1. Introduction

Information can only be of value if it is understood. It can only achieve this if it has a common meaning between the sender and the recipient of the information. The achievement of such common meaning between differing nations in the context of a coalition operation is a significant challenge.

The operational deployment of a joint-service UK force poses a number of issues concerning the interoperability of the communications and information transfer between organisational units that exist within it. Issues include the inconsistency of semantics and syntax during the passing and receiving of information between such units that impairs the ability to communicate the meaning of information richly and effectively.

Each organisational unit[1] will incorporate a subjective means of describing the information that they possess and will read their own subjective interpretation of the meaning of information that is passed to them.

It is likely that as a result of the formation of a coalition, systems that are in situ at the time will need to interact with previously unknown systems from other nation's domains. Without a common vocabulary, translation processes will need to be agreed and developed in order for communication to be possible. It is important therefore to take into account the implicit assumptions that the developers and users of the respective systems have made when defining the respective languages.

When two people communicate, they need to agree a set of rules for attaching meaning to the information content that passes between them. The less explicit the rules are, the more possibility exists that the recipient will misinterpret the communication as they make implicit assumptions. However Military operations are unique. One operation varies from another in the type of units they contain, deployed to meet the unique circumstances within the operation. Each unit works to its own training methods, procedures, processes, language and doctrine. Members of these units will find themselves working alongside others from various nations and service type. In order to achieve any unity of purpose, the differences that exist between them need to be co-ordinated and merged. However the individual units may not have the time to implicitly appreciate the working practices of how its coalition partners work. There is a need for explicit rules and guidance to enable a mutual understanding.

In a Military environment it is desirable to eliminate implicit assumptions at the very lowest levels of the command chain. This will eliminate misinterpretation of orders and reduce error. However the higher up the command chain the more potential exists for interpretation and flexibility for planning and decision making. This creates a compromise between accuracy (explicit) and flexibility (implicit). Each Military domain will need to examine its requirements. For example logistics and planning may benefit from a more rigid, explicit internal structure. Reconnaissance and intelligence units may benefit from a more flexible, implicit internal structure. All External communications should be as explicit as possible, leaving little potential for misinterpretation.

This suggests that any coalition interoperability system should contain explicit definitions to reduce ambiguity but also allow for extension to the explicit definitions to allow for flexibility as and when required.

Whilst the focus of a coalition effort into meta-data standards should be on the development of correct semantic definition, the issue of the syntax chosen to represent this definition also needs careful consideration.

Should all coalition nations adopt common syntactic terminology or should local syntax be mapped to a common semantic meaning? Even if a common definition of an object is agreed and has a common syntactic expression assigned to it, will a user from a local domain adopt the agreed terminology or insist upon using the vocabulary used within their environment? Such a decision may cause confusion and misrepresentation, leading to error. If under extreme duress a user has to focus thoughts upon correct syntactical usage instead of focussing on matters of greater priority, the consequences could be tragic.

Any agreement should take these issues into account when selecting syntax for description and representation and the development of appropriate terminology.

This paper describes current UK initiatives and the possibility of integration with initiatives from other coalition partners. It highlights issues that need to be addressed in order to achieve interoperability between the coalition participant's heterogeneous systems.

[1] The term 'organisational unit is to be read as representing an organisation or an element of that organisation such as a unit, sub-unit, etc.

2. UK Meta-data Development Initiatives

This section introduces current initiatives to define meta-data standards within the UK, and describes a putative approach to yield inter-government interoperability. A number of investigations into the definition of meta-data standards within the Defence and Government arenas has been undertaken in the UK. Amongst the more notable of these are the studies into the development of the Defence Command & Army Data Model (DCADM), the Joint Operations Command System (JOCS), the DERA Knowledge Store, the UK Intelligence CIS Architecture (UKINCA) and the meta-data initiatives supporting the development of the UK Government's Electronic Government Interoperability Framework (e-GIF). These have been the subject of other DERA studies [Ref. - IDM, TG10] and will not be investigated in any detail here.

Within the Web standards/commercial arena, huge amounts of effort has been expended into meta-data standards development studies, both at a standards level, and at a domain specific tag set level. Predominant at the standards level are the World Wide Web Consortium (W3C), the Organisation for the Advancement of Structured Information Standards (OASIS) the Internet Engineering Task Force (IETF), the International Standards Organisation Meta-data Working Group, the Object Management Group/Meta-data Coalition (OMG) and the Dublin Core Meta-data Initiative. At a domain specific level, initiatives such as the Rosettanet and the ebXML immediately come to mind. All of these initiatives are in the public domain and well publicised. To this end they are also not considered within this document.

The key aspects of the approaches to the development of these standards, those of public consultation and consensus of opinion, have been drawn upon in this document.

All of these initiatives have addressed the issue of 'mandate a new set of tags to be adopted by all' versus 'allow existing standards to be kept and mandate an additional set of tags to be used by all' from the same perspective. The issue of trying to enforce the usage of a totally new meta-data set of tags is in danger of being overlooked because of the rather obvious problems of introducing such a tag set. They have opted for letting the participating organisations maintain their own domain-specific tag sets, but have mandated that, as an absolute minimum for instances when information needs to be made available to users within the wider community, the adoption of an internationally recognised meta-data standard be mandated.

2.1 Meta-data in UK Defence

Research has been undertaken within the UK to determine an appropriate set of meta-data elements to describe information for defence purposes. Similar work (described above) is also underway to determine a framework for meta-data across UK government. There is clearly a need for these two initiatives to be aligned. This section will describe the two work areas, and then describe a framework within which the concepts can align.

The aim of defence research work was to determine a meta-data element set, adopting a minimalist approach (i.e. determining the minimum realistic number of necessary elements). This set was determined by consideration of open source material from a defence perspective. The work also adopted a "Purpose driven" approach, rather than an academic and semantic consideration of information characteristics. It considered 2 case studies, focussing on information context and information quality, to determine those essential attributes necessary to capture and convey the context in which information should be used, and the quality of information.

A similar start point was adopted by both pieces of work. A (dictionary) definition of context and quality was used to clarify their necessary meaning and characteristics. These characteristics were then analysed

and fully scoped by analysis based on a literature survey, viewed from a defence perspective, This approach was made viable by the large interest in these facets of information shown by organisations and materiel on the World Wide Web

2.1.1 CASE 1 – CONTEXT

Context is pervasive in our everyday lives. Daily actions are performed within a framework of external circumstances. Such circumstances (such as gravity) are accepted as being relevant to an action (such a dropping a glass), leading to a predictable result. In a similar way, the auditory cue of a phone ringing can result in a hand being outstretched (without looking) to pick up the handset from its "known" location, leading to a (relatively) predictable conversation. In both of these cases we have used a world-model, based on a lifetime of conditions and circumstances, to capture our (local) context, which we hold mentally. An example is a headline in a newspaper. It is designed to be eye-catching, but is often meaningless without the supporting story. Indeed several different stories could fit the same headline, leading to widely differing interpretation of the headline, depending on the context provided by the story. As an example, consider the following headline "Salisbury still in turmoil" – it could relate to flooding in the City of Salisbury, England, rioting in Salisbury, South Africa, or the problems faced by the England cricketer Ian Salisbury. Context is also crucial to distinguish fictional documents and text from non-fictional ones.

Context can also be related to the more dynamic flows normally associated with military information. The quantity of external or supporting information required to assist the decision of the Captain of a Royal Navy ship who is given the message "Sea skimming missile travelling at mach 2 directly towards us" is very small. However the amount of supporting information needed to assist the Army tactical HQ commander who receives the message "6 tanks travelling at high speed towards our position" is much higher (are they our 6 tanks which left 2 hours ago on a reconnaissance mission?; are their main armaments pointing towards us or away from us?). In both cases contextual information is crucial to making the correct decision – the quantity is the variable, driven, at least in part, by timeliness.

We can assert that: - Data + context → Information;

The corollary is that knowledge is information *within a context*.

Dictionaries provide the following definitions of context:

Context: (1) Conditions and circumstances that are relevant to an event or fact;

(2) Parts of a piece of writing / speech etc that precede and follow a word or passage and contribute to its full meaning

We can determine the attributes that are needed to capture context, based upon an analysis of these definitions. Three differing sets have been derived, each capturing a different flavour of context. The first two are derived from the first definition, listing key relevant "circumstances". The final set is derived from the second definition, capturing "related information" aspects.

Set 1	*Set 2*	*Set 3*
Title	Coverage	Related / linked items
Author	Description	Organisation
Creation Date	Subject and Keywords	Currency
Releasability / constraints over use	Format	
Purpose / why collected	Language	
	Reference / Identifier	

Table 1. Flavours of context

Some of these identified attributes actually refer to the item itself, rather than the context in which it should be utilised. They capture basic attributes of the information, and are useful for purposes other than context. Thus it is helpful to classify these three groups under the two headings of *information* attributes and *context* attributes:

Information attributes:	*Context attributes:*
Title	Releasability / constraints over use
Author	Purpose/ why collected;
Creation Date	Related/linked items
Coverage	Currency
Subject and Keywords	Organisation
Format	Description
Reference / Identifier	
Language	

Table 2. Information and context attributes

Thus we have derived a minimal set of relevant key features which can be preserved with each item of information to assist in its interpretation, and thereby, it is suggested, capture its context. Note that the *combination* of these two sets of attributes is proposed as the necessary minimum to capture context. The set of information attributes only captures the minimum basic set of features of the information item itself. These could be considered as the minimum data-set that should be mandated to be recorded with each information item. Ideally, the combined set of information and context attributes should be stored with each information item. Further detailed work is suggested to "trial" this meta-data set against realistic defence information, capturing both its minimal characteristics and its context.

Current Meta-data initiatives in the civil world

Although the concept of meta-data predates the Internet and the world wide web, universal interest in meta-data standards and practices[2] has exploded with the increase in electronic publishing and digital libraries, and the concomitant "information overload" resulting from the vast quantity of undifferentiated digital data available online. Anyone who has attempted to find information online using one of today's popular Web search services has likely experienced the frustration of retrieving hundreds, if not thousands, of "hits" with limited ability to refine or make a more precise search. The wide scale adoption of descriptive standards and practices for electronic resources is targeted at improving the retrieval of relevant resources from the Internet. Different communities of users meet such needs today with a wide variety of meta-data standards. The web community actively welcomes such variety, especially when sitting on top of a common framework.

Meta-data Schemas

Meta-data schemas are needed in order to be able to organise the content on the web. Meta-data schemas have many applications like for example searching and retrieving content, cataloguing content, making content machine understandable or machine-readable, transmitting content, etc. Implementers of meta-data can invent a meta-data schema themselves, but besides running the chance of reinventing the wheel, the danger is that it lowers the level of access when others do not know the schema. A broad landscape of individual meta-data schemas will maybe fulfil the need for localising content but will endanger the accessibility of the content by others. There seems to be a trade-off between localisation and interoperability.

A wide variety of standardisation initiatives have been or are being undertaken, both within formal standardisation organisations and within other organisations and both at a general level and at a more domain specific level. Standardisation of meta-data is a diverse area and the activities are spread over the different application areas and domains of meta-data schemas. Detailed differences in the meaning of meta-data definitions are in equal need of clarification. As a result, implementers who will need meta-data face the challenge of designing schemas that are compatible with both existing and emerging standards. They will need some core elements along with parts of more domain-specific element sets. In some cases they even might find themselves having to invent elements of their own. Since different schemas are needed in different situations, implementers cannot always use schemas "straight from the box". However, several initiatives are emerging that define standard mechanisms for the creation of schemas (e.g. IBM, Infoseek, and Microsoft).

The explosive growth in recent years of meta-data definitions and schema has not gone unnoticed. A number of key initiatives have standardised crucial, central attributes, and put into place frameworks and strategies to align and regulate meta-data definitions and application. The key initiatives are the Dublin Core (DC), the Warwick extension framework, and the Resource Description Framework (RDF). These initiatives are brigaded under the W3C banner, albeit that they were historically focussed on library catalogue needs. However their applicability to a broad range of meta-data schemes is now widely accepted. Use of DC-based meta-data schema is commonplace. It is also increasingly tied to XML and associated developments.

Dublin Core [3]

Dublin Core meta-data is specifically intended to support *resource discovery.* The elements represent a broad, interdisciplinary consensus about the core set of elements that are likely to be widely useful to support resource discovery. The Dublin Core meta-data standard is a simple yet effective element set for describing a wide range of networked resources. The Dublin Core standard comprises fifteen elements, the semantics of which have been established through consensus by an international, cross-

[2] Through bodies such as W3C, OASIS, OMG, MDC, ISO and ECMA.

[3] Named after the initial meeting at Dublin, Ohio, USA

disciplinary group of professionals from librarianship, computer science, text encoding, the museum community, and other related academic fields.

The Dublin Core element set is outlined in the Table (3) below. Each element is optional and may be repeated. Each element also has a limited set of qualifiers. These are attributes that may be used to further refine (not extend) the meaning of the element.

Dublin Core has as its goals the following characteristics:

- Simplicity of creation and maintenance
- Commonly understood semantics
- International scope
- Extensibility

Although the Dublin Core favours document-like objects (because traditional text resources are fairly well understood), it can be applied to other resources as well. Its suitability for use with particular non-document resources will depend to some extent on how closely their meta-data resembles typical document meta-data and also what purpose the meta-data is intended to serve.

The basic DC list is unordered, and is often quoted in the historical order in which the elements were defined and agreed.

Title	Creator	Subject and Keywords
Description	Publisher	Contributor
Date	Resource Type	Format
Resource Identifier	Source	Language
Relation	Coverage	Rights Management

Table 3. Basic DC list

A full description of the elements is given at Appendix A to this paper.

The ability of DC Meta-data set to represent Context

We are now able to compare the DC set of attributes with those proposed earlier in this paper which are suggested as sufficient to capture context. The DC elements have been mapped onto the information and context lists derived earlier, as shown in the following Table (4).

Information attributes:	*DC equivalent(s)*	*Context attributes:*	*DC equivalent(s)*
Title	Title	Releasability / Constraints over use	Rights
Author	Creator & Contributor	Purpose/ why collected;	
Creation Date	Date	Related/linked items	Source & Relation
Coverage	Coverage	Currency	
Subject and Keywords	Subject	Organisation	Publisher
Format	Format &Type	Description	Description
Reference / Identifier	Identifier		
Language	Language		

Table 4. DC element mapping

It can be seen that there is an excellent match between the DC element set and our requirements for meta-data to support information context. The DC set offers some degree of additional richness in three areas: the *contributor* is identified as well as the *creator* / *author*, both *source* and *relation* can be used to capture *related / linked items*, and *type* is used in addition to *format*. Only two areas are not reflected in the DC (*purpose/why collected*, and *currency*). In addition, the topic of *releasability / constraints over use* is only loosely represented by the *rights* field.

DC qualifiers

As its name implies, the Dublin Core is a starting place, not a final word. It was anticipated that local applications would need to add elements to meet local needs. In July of 2000, the Dublin Core Meta-data Initiative (DCMI) issued its list of recommended Dublin Core Qualifiers. The DCMI recognised two broad classes of qualifiers:

- Element Refinement. These qualifiers make the meaning of an element narrower or more specific. A refined element shares the meaning of the unqualified element, but with a more restricted scope. A client that does not understand a specific element refinement term should be able to ignore the qualifier and treat the meta-data value as if it were an unqualified (broader) element.
- Encoding Scheme. These qualifiers identify schemes that aid in the interpretation of an element value. These schemes include controlled vocabularies and formal notations or parsing rules. A value expressed using an encoding scheme will thus be a token selected from a controlled vocabulary (e.g., a term from a classification system or set of subject headings) or a string formatted in accordance with a formal notation (e.g., "01-01-2000" as the standard expression of a date). If a client or agent does not understand an encoding scheme, the value may still be useful to a human reader.

The element refinement thread is relevant to our needs; the encoding scheme is not, and will not be discussed further. The list of DC qualifiers (“refinements”) is presented in the following Table (5).

DC Element	*Element Refinement(s)*	*DC Element*	*Element Refinement(s)*
Date	Created Valid Available Issued	Relation	Is Version Of Has Version Is Replaced By Replaces Is Required By Requires Is Part Of Has Part Is Referenced By References Is Format Of Has Format
Description	Table Of Contents Abstract	Format	Extent Medium
Coverage	Spatial Temporal Modified	Title	Alternative

Table 5. DC qualifiers

As can be seen, this set of qualifiers adds significant richness to the relation field, allowing the linkages essential to capture context to be more fully defined and captured. It also adds a valid extension to the date field, thus aiding the capture of information currency.

Thus it is suggested that the DC elements, plus selected qualifiers from the agreed DCMI list, offer sufficient richness to capture the attributes of information essential to convey context. The only weaknesses are in the areas of *constraints* and *releasability*. The need to convey *constraints* could be embodied within the scope of the *description* element. However, consideration should be given to adding a defence-specific qualifier as part of the *description* element, to ensure visibility of the need for *constraints*. It is suggested that the topic of *releasability* be handled within defence either as a local qualifier under *rights*, or as a “defence interpretation” of the *rights* field. The difference between these two options is only likely to emerge after real use of these fields with realistic information sets and appropriate meta-data support tools. It depends on whether there is a perceived need for a pure DC *rights* field, when issues of information exchange and interoperability with Allies and coalition partners are taken into consideration. Table 6 below, summarises the final mapping of information and context attributes to their equivalents from the DC plus DCMI qualifiers set.

Information attributes:	*DC / DCMI equivalent(s)*	*Context attributes:*	*DC /DCMI equivalent(s)*
Title	Title	Releasability / Constraints over use	Rights or rights qualifier Description or description qualifier
Author	Creator & Contributor	Purpose/ why collected;	Subsume into description element, or add a new qualifier
Creation Date	Date; Date qualifiers (Created, Available, Issued Modified)	Related/linked items	Source & Relation; Relation qualifiers (Is Part Of; Has Part; Is Referenced By; References)
Coverage	Coverage; temporal & spatial qualifiers	Currency	Date valid qualifier
Subject and Keywords	Subject	Organisation	Publisher
Format	Format &Type	Description	Description
Reference / Identifier	Identifier		
Language	Language		

Table 6. Mapping of attributes to DC & DCMI

2.1.2 CASE STUDY 2 – QUALITY INFORMATION

Information and quality

There are increasing levels of interest in defining and/or eliciting the quality of information on the WWW. Representative initiatives are aimed at:-

- Formal research & analysis (university students, academia, legal, medical….)
- Widespread interest in site blocking (for pornography & adult material) – by parents, schools, public libraries etc
- Data & information quality on the Internet & Intranets for commerce. There is much commercial work on setting up and managing information and data warehouses, together with data cleansing tasks for industry “core” data.

There is thus a keen interest in defining and capturing the attributes (parameters) which best capture the quality aspects of information. The capture of suitable quality parameters can also assist in an

audit/review/improve cycle to yield higher quality information. Further, understanding the quality of information also allows a trade-off to be made between different quality attributes (e.g. timeliness vs. accuracy in the military context).

Definition(s)

The following are dictionary definitions of "quality": -

Quality (noun):	
(1) = Trait, characteristic	e.g. mahogany has the quality of high durability (quality, qualities)
(2) = Degree / grade of excellence	e.g. wood grain of low quality (a relative term)
(3) = Superiority of kind	e.g. silk of quality
Quality (adjective):	
Specifically means *high* quality	e.g. quality paper, a quality car - therefore similar to definition 2 above

Quality therefore has three differing meanings: a property, a degree of excellence or a statement of excellence. These differing meanings can also be contrasted as an attribute, a relative term, and an absolute term. The word thus exhibits a broad range of allowable uses; it is not surprising that its gratuitous use can cause a lack of consistency in interpretation.

We can thus construct a succinct definition for *quality information* (and also for *information quality*).

1. An attribute of information e.g. describing a characteristic of information – we will use information quality (information qualities) to capture this variant
2. Relatively good information – use as a description; information of high/low quality
3. Information of high intrinsic value or with high merits – this is generally taken as synonymous with quality information.

Information Qualities

We can determine the specific quality aspects the user/customer will look for in a "good" information product as a way of characterising the necessary information quality attributes (i.e. its qualities). Once captured, they can be used to determine the degree of satisfaction that is delivered by a specific information item, and thus determine whether or not we have "quality information".

The following table (7) list four examples of quality attributes, derived from open-source material: -

The most basic requirements of good information are:	
Objectivity:	That the information is presented in a manner free from propaganda or disinformation**.**
Pluralism:	That all aspects of the information are given and are not restricted to present a particular viewpoint, as in the case of censorship
Completeness:	That the information is a complete, not a partial picture of the subject

Table 7 - Example 1 (Ref. http://ils.unc.edu/~fents/310/)

Analysis of a wider range of these information quality attributes, drawn from a variety of sources, was then pursued. They were aggregated, grouped and ordered around those topics that occurred most regularly. In this way common themes, and thus "best practise", were allowed to emerge, but the breadth and richness of the wide range of material reviewed was maintained.

The analysis of these varied data sets has yielded the following "best of breed" set of quality attributes for defence information. The aim is to produce a set of attributes that are equally valid for pseudo-static documents / reference data and dynamic military reports. Four high level attributes are proposed – Accuracy, Credibility, Objectivity and Support. Table 8 lists additional details that they can each embrace or subsume.

Accuracy

Up to date; timely; currency; lifetime specified; coverage & scope identified; factual; information integrity (accuracy, confidence); stability; shortcomings identified.

Credibility

Author details; reputation of author & author organisation; credentials; peer or management review; quality controls.

Objectivity

Consistent; balanced; fair; reasoned; authentic; purposeful; Conflict of interests identified; context; purpose & audience identified.

Table 8– Quality attributes of information

Clearly the differing, detailed facets, which are listed under each major heading, need to be captured to varying degrees for different types of information. The list is best viewed as a "check list" for the information quality which the author (and/or owner) of the information should utilise to define the information quality, and the user of the information should use to assist in determining what confidence to place in the information. The use of an acronym / mnemonic based on the initials of Accuracy, Credibility, Objectivity and Support (ACOS) will perhaps assist users and producers alike in this task.

Meta-data to capture quality

Having derived a suitable (or least indicative) set of quality attributes that appear appropriate for capturing the quality of information, we must now consider how such attributes should be linked or stored with each item of information. Once again, we can look towards the DC + DCMI element set as a credible start point to determine whether beneficial mapping will occur. Initial mapping of the "ACOS" headings gave a good indication of success, thus the full set of necessary attributes, as cited in Table 8 can be mapped onto the DC/DCMI set of elements.

This is shown in Table 9; note that the DC/DCMI elements have been grouped under the ACOS headings, for simplicity.

ACOS – DETAILS	*DC & DCMI*
Accuracy	
Up to date; timely; stability	Creation Date, Date & Date qualifiers (Created, Available, Issued Modified)
coverage & scope identified	Coverage; temporal & spatial qualifiers
	Subject and Keywords
	Related/linked items Source & Relation; Relation qualifiers (Is Part Of; Has Part; Is Referenced By; References)
currency; lifetime specified;	Currency Date valid qualifier
shortcomings identified ; factual	Description
information integrity (accuracy, confidence)	**1**
Credibility	
Author details; Reputation of author & author organisation	Author Creator & Contributor
Credentials; meta-data.	Related/linked items Source & Relation; Relation qualifiers (Is Part Of; Has Part; Is Referenced By; References)
Reputation of author & author organisation	Organisation Publisher
peer or management review; quality controls	Organisation Publisher **2**
Objectivity	
Consistent; balanced; fair; reasoned; authentic;	Author Creator & Contributor
Conflict of interests identified	Releasability / constraints over use
purpose & audience identified; purposeful;	Purpose/ why collected

ACOS – DETAILS	*DC & DCMI*
context	Related/linked items Source & Relation; Relation qualifiers (Is Part Of; Has Part; Is Referenced By; References)
Conflict of interests identified	Description
Support	
Appropriate format; available standards	Format &Type
	Reference / Identifier
accessible	Language
accessible; links valid; ease of use & navigation; user support & documentation; site integrity; contact details of author	Organisation Publisher (quality….) **3**

Table 9 – Comparison between ACOS and DC/DCMI

Excellent linkages and relationships can be seen between the "standard" DC/DCMI meta-data element set, and the desired set of quality attributes. Three distinct areas are not well covered by the (existing) DC & DCMI fields. These are indicated by the Italics in Table 9, and are labelled 1, 2 and 3. A measure of the information integrity (label 1 - accuracy, confidence rating/percentage) is the most noteworthy omission, with no obvious area for inclusion in DC. The other two, more generic, areas (labelled 2 and 3) actually relate to the quality of the site (in a Total Quality sense), and thus can relate back to the site owner / owner organisation. They are essentially intangibles, which must be delivered as part of the information provision service.

2.1.3 DISCUSSION

The preceding case studies have clearly shown that a common set of meta-data elements, are of immediate application to capture both the context of information and also the quality of information. It has been shown that only a single additional defence specific field, to cover the accuracy and confidence of the information item, needs to be added to the DC/DCMI set derived by the W3C. With this addition, it has been demonstrated that a relatively simple, minimal set of meta-data elements can describe not only the basic parameters of the item of information, but can also be used to establish its context, and its quality. The power thereby offered by the use of this single set of meta-data elements is significant. Its use can greatly assist in realising the full potential of information across UK Defence.

2.2 Conclusions & summary of UK activities

Research case studies have demonstrated remarkable over all synergy between the needs of UK defence and those of "DC" community. Further it has been indicated by the Cabinet Office that the UK Government is likely to adopt the core DC elements as the basis for its meta-data standard.

The case studies have also demonstrated synergy between two different uses of the same meta-data set (context and quality); this will allow real benefits to be accrued across defence by the adoption and use of the same minimal set of meta-data. Further, the use of a simple set of guidelines ("ACOS") has been suggested as a way of structuring the thinking behind the generation and the use of any (and every) information product.

The analysis presented above has shown that the adoption of a core data set, utilising the central eight elements of the DC set (the information attributes of Table 6) can offer common elements found between differing implementations. It would seem appropriate to adopt this central set as a common core across UK Government, allowing local, additional elements to be utilised. For example the context attributes plus accuracy (as identified as lacking by the quality case study) could be considered as a UK defence-specific add-ons. Wider benefit would accrue by their adoption, but it is not essential. This simple model for meta-data standardisation is shown schematically in figure.1, below. Only three organisations are shown for simplicity. The approach is clearly extensible to cover many organisations with varying degrees of necessary commonality. An important ingredient of this model is the small number of elements which are deemed "common", but which still present a unified, and useful coherence to information provision and use.

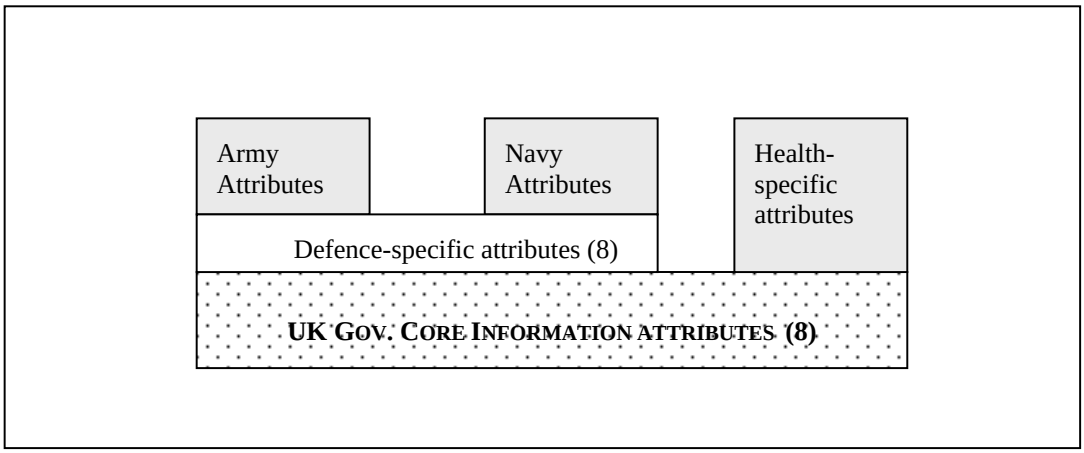


Fig. 1 Simple model for meta-data standardisation

The case studies, and this analysis, also strengthen the need for adoption of an appropriate set of meta-data across defence, and the need for tools to generate and search/visualise meta-data that are flexible and extensible.

3 Meta-data interoperability within a Coalition

3.1 Introduction

This section describes different approaches for achieving coalition information interoperability. It highlights issues concerned with each approach. It suggests that the putative UK solution to meta-data as illustrated in figure 1, can be extended to offer a potential solution to the problem of information integration within a joint multi-national defence coalition operation.

3.2 Alternative solutions

When considering the need for the interoperability of meta-data throughout a coalition, three alternative solutions present themselves: -

- A centralised approach involving the adoption of a international coalition meta-data tag set standard

- A decentralised approach involving interfaces between each nations existing meta-data tag set standards
- A federated approach based upon a hybrid of approaches 1 and 2

Centralised

A centralised approach would contain systems that can perform independently of one another if required to do so, but will normally operate as a subordinate part of the centralised system. The subordinate systems would be closely coupled together within the centralised system's boundary. Complicated **referential** management processes will exist to ensure that interoperability is maintained. The references and associations between the individual systems will grow as nations join the coalition. This will make the management extremely difficult and hard to maintain. If a nation were to withdraw from a coalition the domino effect upon the whole centralised system's references and associations may well become chaotic and unmanageable.

A global naming scheme consisting of a common explicit vocabulary would be shared between the participating nations. An international meta-data tag set standard would need to be developed and agreed by all nations within the coalition. The agreed tags would be retained and the remaining redundant meta-data tags would be discontinued. This agreement will not be easily achieved. Each nation may have legitimately developed local meta-data tag sets prior to any such agreement. They may well have bona-fide financial and logical reasons for retaining the ownership of any tag set that they have developed in its native format without wishing to alter, translate or migrate the tag set in any way. Faced with this fact it is unlikely that the development of an international tag set will have the full co-operation of all coalition participants. The process of achieving a compromise may even involve arbitration. It can be argued that any compromise will not suit all parties and that the development of a tag set in such an arbitrated way may not actually suit any party. This may cause a problem of ownership. The feeling of ownership is a very important point because it will help the participants to adopt the tag set standard. None of the nations may feel as though they fully own the compromised tag set and may consider it to be flawed. Consequently it may not be embraced. This will immediately place constraints over its potential use.

Consideration must also be given to the participation of an as yet unidentified body within a coalition at a future date. The assumption must not be made that a future participant will automatically comply with any existing international tag set standard. It may also have legitimate local issues that affect its adoption of the tag set. The problem may occur every time a nation joins the coalition.

Decentralised

The decentralised approach would contain many diverse systems with no overall control. It would allow nations to retain existing local naming schemes and tag sets and will satisfy local ownership issues, as no common vocabulary will exist. However further issues regarding the interoperability between the various domains emerge.

The systems will operate in a **coercive** management environment. Many independent management initiatives may occur on an ad hoc basis to satisfy local requirements of interfaces between systems. These will be difficult to co-ordinate and may result in duplication of effort and redundant processes.

As each nations naming schemes and tag sets have been developed in isolation of one another within their specific domain environment they will therefore have been defined by each domain's cultural influence. Consequently there may be substantial differences between the semantic meanings of the respective tag set definitions between domains. Each tag set will need to be compared with its counterpart from the other coalition nations. Any differences will need to be identified and interpreted by the other participants in such a way that the semantic meaning is mapped between each system. New mappings may need to occur every time a system is introduced, amended or discontinued. The scope of this task is difficult between

two domains resident in a single nation. When you consider the problem between all domains that exist within all coalition nations the problem increases significantly.

Federated

A federated approach will be a hybrid of the previous two approaches and will be managed by taking a **pluralist** management approach with a mixture of a central authority and voluntary collaborative initiatives working toward a common overall goal. Core common management processes will exist but the individual components will also retain local management processes for flexibility.

Conclusion

A federated approach may reduce the complexity of the management problems that exist within the centralised and decentralised paradigms. The development of a common naming scheme consisting of a minimal tag set with the ability to interface with the local naming schemes of coalition nations is suggested. This will allow for common semantic meaning between all of the coalition participants and also allow for the retention of local semantic meaning within each nation. It will also allow for a minimal amount of centralised control within the coalition

The issue remains of how to interface the local naming scheme to the common naming scheme and how to map and translate the local extensions of the common core standard. This may be achieved by importing a local naming scheme from one domain to another with associated context details stating where the naming scheme originated.

3.2.1 ONTOLOGIES

Ontological agreements are necessary for successful communication between parties as they provide a common point of reference that removes implicit assumptions and replacing them with explicit rules. This guarantees a common understanding between all participants. The agreement will also ensure consistency of information systems that are implemented according to the ontology. However ontological agreements are very hard to achieve: -

Centralised ontology

An example of an existing international initiative of an integrated information system is the Army Tactical Command and Control Information System (ATCCIS). This is a centralised system based upon a de-facto set of data definitions with additional definitions from participant nations. This has resulted in a highly complicated network of compound complex structures made up of atomic level relationships. These are extremely difficult to map consistently. ATCCIS Development has been evolutionary with many small incremental steps. Each step has had to be agreed by all participant nations due to the complexity of the system. Consider the impact to the existing coalition systems when a new member of a coalition is added or a member withdraws. ATCCIS is delivering successful results but has required a long development period. Any future similar interoperability process will require a similar commitment.

Decentralised ontologies

There will be little if any common ontology within a decentralised approach due to the coercive nature of the management processes. Ontological islands may appear and disappear as ad hoc communications links are created and broken. It is likely that it will be very difficult for the coalition as a whole to achieve interoperability between processes and systems.

Federated ontology

It is likely to be easier to achieve an agreement over the definitions of a minimal core meta-data tag set instead of a complex explicit vocabulary of every nation's terminology. The process may reduce the need for compromise and arbitration and is likely to be more expedient.

The flexibility offered by this approach is likely to allow local initiatives to continue with little or no adjustment. This will reduce development time and resource costs.

Conclusion

The Federated approach of adopting a minimal common core tag set as proposed in figure 1 with an interface to local naming schemes is more likely to be accepted by coalition participants with a reduction in the need to compromise local development initiatives. An ontology is likely to be agreed and future coalition members are more likely to accept the minimal core tag set. Should any nation leave the coalition, there is likely to be a minimal impact upon the remaining coalition nations systems.

3.3 Potential technical solution

The various coalition participants must give consideration to the existing naming schemes that are in use across the coalition and how they might be made to integrate.

One approach would be to allow each nation to post their respective definitions to central repository. Any subsequent information could include pointers to the local definition. The recipient could then access the definition and apply their equivalent syntax to the definition. If no equivalent exists the terminology used by the sender could possibly be adopted.

A potential technical solution to the importing of naming schemes may be achieved through the use of Extensible Markup language (XML) schemas and namespaces to define specific nation naming schemes and tag sets. The namespace will allow the definition of an object residing within one domain to be imported into another domain. It could then be mapped to a local naming scheme, for example: -

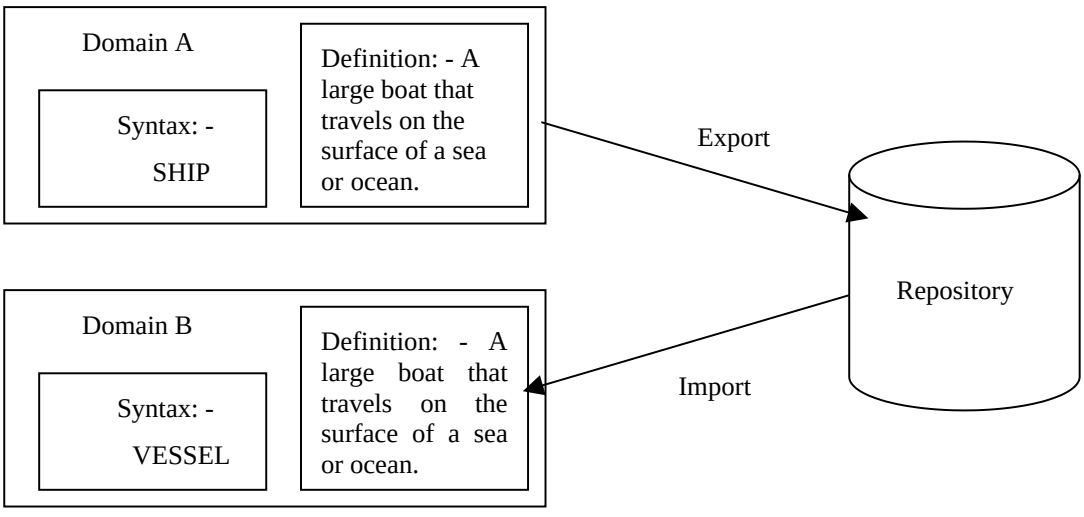


Fig. 2 The use of a local namespace

An object ‘ship’ is defined within a namespace in domain A. Domain B syntactically refers to an equivalent object to ship within its local domain as ‘Vessel’ and assigns Domain A’s definition of ‘ship’ to the local syntax. When domain A communicates with domain B and refers to a ‘ship’ Domain B understands the meaning, using its local syntax. This offers a potential mechanism to translate meaning between domains where necessary.

A problem with this approach is the management of the multiple linguistical values that will need to be stored to allow every permutation of one coalition nation being able to understand another. (Many: many relationships).

An alternative would be to allow an explicit set of permitted values to be stored within a coalition namespace and have every nation import the explicit definition into its local domain and assign its local syntax to it (Many: One relationship).

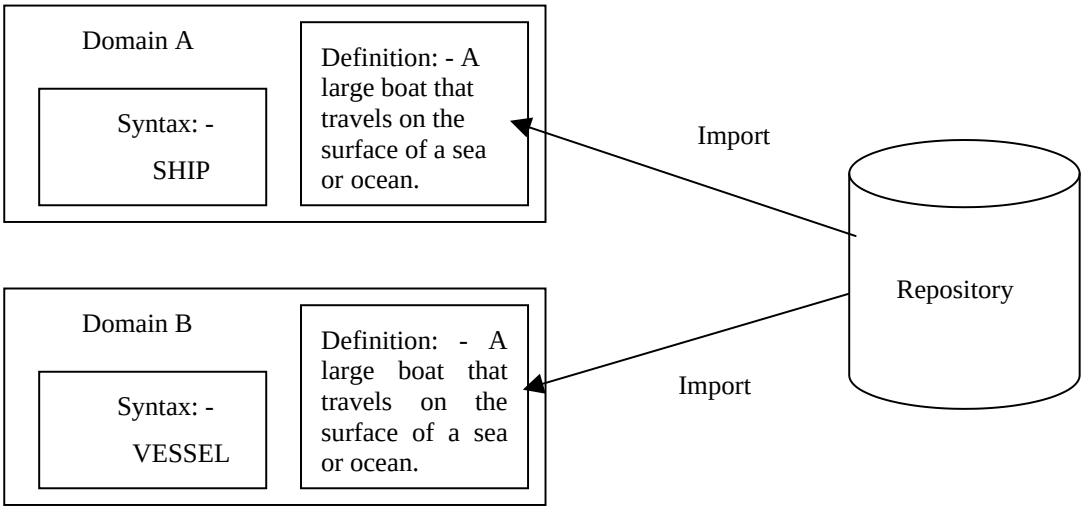


Fig. 2a The use of a coalition namespace

The coalition is again confronted with a complex problem of achieving mutual agreement of the permitted ontological values.

All XML schemas will require registration within a centralised repository for use by coalition participants. This will make users aware of the existence of the schemas and will allow user applications to access any schemas or namespace declarations. This will also enable the association of existing namespaces and the explicit definition of terminology within it from one domain to new XML schemas within another. It will also allow for the registration of domain specific semantics and syntax. The repository will require a location(s), ownership, stewardship and management processes, as it will contain many Schemas, Namespaces etc. from all coalition nations. Issues concerning the structure, methods of access replication, sighting and ownership of such repositories will need to be agreed.

Encoding languages are developed for specific purposes in mind and will do the job that they are designed to do. The problem arises when a language is used for a purpose beyond its scope or if many dialects of the same language exist that basically achieve the same overall goal but contain subtle differences. For example Extensible Markup Language (XML) can be written using a document type definition (DTD), XML schemas, Regular language description for XML (RELAX) etc. It is essential to agree the encoding language standard for use in any proposed system implementation as the absence of a

standard will increase the potential for problems due to the subtle differences or incompatibility of language communication between each domain within the coalition.

Conclusion

XML namespace technology appears to offer a solution that is compatible with the concept of an interface between core and local domain meta-data.

Two alternative approaches for an interface between domains incorporating the use of namespaces have been suggested. They can both achieve a solution but with a trade off of more local management processes against complicated ontological processes.

Any ontological agreement should include a definition of the encoding standard to be adopted to ensure a consistency of approach. The location, ownership, stewardship and management processes of any repository must also be agreed and explicitly defined.

4 Overall conclusions

- There is a lack of a common meta-data vocabulary between nations that leads to interoperability problems. This is due to the isolated development of heterogeneous systems to meet national requirements.

- A similar interoperability problem exists nationally with in the UK between UK defence and other Government departments – a minimal common core meta-data tag set, supplemented by locally defined elements, is proposed as a solution to this problem. It is suggested that this approach can be extended to assist in the coalition interoperability problem area.

- Research case studies have demonstrated considerable synergy between the needs of UK defence and those of “Dublin Core” community. UK Government is likely to adopt Dublin Core as a basis for its meta-data standard.

- Analysis has shown that the adoption of a core data set, utilising a sub set of the Dublin Core (The information attributes of table 6) can support the common elements found between differing system implementations.

- The case studies have demonstrated synergy between two different uses of the same meta-data set (context and quality); this will allow real benefits to be accrued across defence by the adoption and use of the same minimal set of meta-data. The use of a simple set of guidelines (“ACOS”) has been suggested as a way of structuring the thinking behind the generation and the use of any (and every) information product.

- The hybrid federated approach of a core meta-data tag set with an interface to national meta-data sets seems to offer the best solution to coalition information interoperability as it provides rigor and flexibility to adapt to individual nation needs.

- A federated approach is likely to reduce the complexity of management problems. This will allow for common semantic meaning between all of the coalition participants and also allow for the retention of local semantic meaning within each nation. It will also allow for a minimal amount of centralised control within the coalition.

- It is likely to be easier to achieve an agreement over the definitions of a minimal core meta-data tag set instead of a complex explicit vocabulary of every nation's terminology. The process may reduce the need for compromise and arbitration and is likely to be more expedient.

- A possible technical solution to the interface between core and local domain (Figure 1) is offered through the use of XML Namespaces.

Appendix A - Dublin Core Elements

Name:	Definition:	Comment:	Example:
Title	A name given to the resource.	A Title will be a name by which the resource is formally known.	Information, in Context
Creator	An entity primarily responsible for making the content of the resource.	Examples of a Creator include a person, an organisation, or a service.	Dr S Braim
Subject and Keywords	The topic of the content of the resource.	Typically, a Subject will be expressed as keywords, key phrases or classification codes that describe a topic of the resource.	Information, Information management, Knowledge management etc.
Description	An account of the content of the resource.	Description may include, but is not limited to: an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content.	
Publisher	An entity responsible for making the resource available	Examples of a Publisher include a person, an organisation, or a service.	DERA
Contributor	An entity responsible for making contributions to the content of the resource.	Examples of a Contributor include a person, an organisation, or a service.	
Date	A date associated with an event in the life cycle of the resource.	Typically, Date will be associated with the creation or availability of the resource.	01-12-2000
Resource Type	The nature or genre of the content of the resource.	Type includes terms describing general categories, functions, genres, or aggregation levels for content.	Text (Could be: Collection; data-set; event; image; interactive; resource; model; party; physical object; place; service; software; sound; text)

Format	The physical or digital manifestation of the resource.	Format may include the media-type or dimensions of the resource. Format may be used to determine the software, hardware or other equipment needed to display or operate the resource. Examples of dimensions include size and duration.	For example the list of Internet Media Types [MIME] defining computer media formats
Resource Identifier	An unambiguous reference to the resource within a given context.	Recommended best practice is to identify the resource by means of a string or number conforming to a formal identification system.	Example formal identification systems include the Uniform Resource Identifier (URI) (including the Uniform Resource Locator (URL)), the Digital Object Identifier (DOI) and the International Standard Book Number (ISBN).
Source	A Reference to a resource from which the present resource is derived.	The present resource may be derived from the Source resource in whole or in part. Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.	
Language	A language of the intellectual content of the resource.		For example, 'en' for English, 'fr' for French, or 'en-uk' for English used in the United Kingdom.
Relation	A reference to a related resource.	A string or number conforming to a formal identification system.	
Coverage	The extent or scope of the content of the resource.	Coverage will typically include spatial location (a place name or geographic co-ordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity).	

Rights Management	Information about rights held in and over the resource.	Typically, a Rights element will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights.	Crown Copyright

This page has been deliberately left blank

Page intentionnellement blanche

Commercial Off-the-Shelf Software Component Interoperability

Jeffrey Voas
Cigital
21351 Ridgetop Circle, Suite 400
Dulles, VA 20166 USA
voas@cigital.com
Phone: 703.404.9293, Fax: 703.404.9295

ABSTRACT

When a software system fails, a confusing and complex liability problem ensues for all parties that have contributed software functionality (whether COTS or custom) to the system. Potential contributors to the system failure include: (1) defective software components, (2) problems with interfaces between components, (3) problems with assumptions (contractual requirements) between components, and (4) hidden interfaces and non-functional component behaviors that cannot be detected at the component level. In this paper, our goal is to focus on the interoperability problems created by defective COTS software components, and in particular, hidden interfaces and non-functional component behaviors. And we will also briefly look into the problem of how to compose one particular type of non-functional behavior, the "ilities".

Tolerating Component Failures

More and more of the software is delivered to system integrators in a form described as a "black-box." The reason why particular software components are termed black-boxes is because they are packaged as executable objects (with licensing agreements that forbid de-compilation back to source code). A worthy goal, then, is to provide a methodology for determining how well a system can perform if particular COTS components are of such poor quality that interoperability problems arise.

The technique that we will discuss for assessing the level of interoperability between COTS software components and custom components is called "Interface Propagation Analysis" (IPA). IPA perturbs (i.e., corrupts) the states that propagate through the interfaces that connect COTS software components to other types of components. By corrupting data going from one component to a successor component, failure of the predecessor is approximated (simulated), and its impact on the successor can be assessed. Since many of the interoperability issues related to software problems are a result of one components intolerance to another, our approach allows for measuring the level of intolerance when one component fails and sends "junk" information (or even a lack of information) to its successor.

To modify the information (states) that components use for inter-communication, write access to those states is required (in order to modify the data in those states). This is done by creating a small software routine called **PRETURB**, that replaces the original output state with a different (corrupted) state. This is of course done as the system executes. By simulating the failure of various software components, we assess whether the remainder of the system can tolerate it.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Let's illustrate using AIX's **cos()** function (which is a fine-grained COTS utility for which we do not have access to the source code):

double cos(double x)

This declaration indicates that the **cos()** function receives a double integer (contained in variable x) and returns a double integer. Because of C's language constraints, the only output from **cos()** is the returned value, and hence that is all that fault injection can corrupt.

To see how this analysis works, consider an application that contains the following code:

```
if (cos(a) > THRESHOLD) {
   do something
  }
```

Our goal, then, is to determine how the application will behave if **cos()** returns incorrect information. To do so, we will modify the return value from the call:

```
if (PERTURB(cos(a)) > THRESHOLD) {
   do something
      }
```

We should mention that the IPA technique is more than an interesting research idea. It has been used successfully by the FAA's prime contractor on the new Wide Area Augmentation System (WAAS). This is a new air-traffic control technology that will be rolled out over the Pacific Ocean in 2007.

So in summary, IPA begins from the assumption that all software components will fail. Therefore simulating the failure of worrisome COTS components to determine the system's tolerance to their misbehavior is prudent. And note that this approach can also be used to determine how COTS components will react when custom components fail as well.

Composing "ilities"

Much of the work from the past 10 years into Component Based Software Engineering (CBSE) and Component Based Development (CBD) has dealt with Functional Composability (FC). FC is concerned with whether $F(A) \xi F(B) = F(A \xi B)$ is true (where ξ is some mathematical operator), i.e., whether a composite system results that has the desired functionality given that the system is created solely by joining A and B.

But increasingly, our community is discovering that FC, even if it were a solved problem (using formal methods, architectural design approaches, model checking, etc.), is still not mature enough for other serious concerns that arise in CBSE and CBD. These concerns stem from the problem of composing "ilities". "Ilities" are non-functional properties of software components and define characteristics such as security, reliability, fault-tolerance, performance, availability, safety, etc.

The problem stems from our inability to know *a priori*, for example, that the security of a system composed of two components, A and B, can be determined from knowledge about the security of A and the security of B. Why? Because the security of the composite is based on more than just the security of the individual components. There are numerous reasons for this, and here, we will just look at the factors of component performance and calendar time.

As an example, suppose that A is an operating system and B is an intrusion detection system. Operating systems have some level of authentication security built into them, and intrusion detection systems have some definition for the types of event patterns that likely warn of an attack. Thus the security of the composition clearly depends on the security models of the individual components. But even if B has a worthless security policy or flawed implementation, the composite can still be secure. How? By simply making the performance of B so poor that no one can log on, i.e., if the intrusion detection system is so inefficient at performing an authentication, then in a strange way, security is actually increased. And if the implementation of B's security mechanism is so unreliable that it disallows all users access, even legitimate ones, then strangely, security is again increased. While these last 2 examples are clearly not a desirable way to attain higher levels of system security, both do actually decrease the likelihood that a system will be successfully attacked.

And if we again use our same example of A as an operating system and B as an intrusion detection system, and this time we assume that A provides excellent security and B provides excellent security, we must accept the fact B's security is a function of calendar time. The reason for this is simply that new threats and ways to "break in" are always being discovered. So even if you could create a scheme such as Security(A) ξ Security(B) = Security(A ξ B), Security(B) is clearly a function of which version of B is being composed and what recent new threats have arisen.

So the question then comes down to: "which "ilities", if any, are easy to compose? The answer is that there are no "ilities" that are easy to compose and some are much harder than others. Further, there are no widely accepted algorithms for how to do so. We just demonstrated this problem for security. But note that the same holds true for others such as reliability. For reliability, consider a 2-component system in which component A feeds information in B and B produces the output of the composite. And assume that both components are reliable. So what can we assume about the composite's reliability? While it certainly suggests that the composite system will be reliable, it must be recognized that components (which were tested in isolation for their individual reliabilities) can suddenly behave unreliably when connected to other components, particularly if the isolated test distributions did not at all reflect the distribution of transferred information after composition. Further, there can be component behaviors that are termed as "non-functional", that cannot be observed nor manifest themselves until after composition occurs. Such behaviors can undermine the reliability of the composition. And finally, if one of the components is simply the wrong component although highly reliable, naturally the resulting system will be useless.

In addition to reliability and security, one "ility", that at least on the surface appears to have the best possibility of successful composability is performance. But even that is problematic from a practical sense. The reason stems from the fact that even if a Big(O) algorithmic analysis was performed on a component, the practical consequences on that component's performance after composition depends heavily on the hardware and other physical resources. That requires, then,

that many different hardware variables might have to be dragged along with a certificate making even minimal, worst-case claims about the performance of the component. Clearly, this appears to have serious pragmatic difficulties.

Note that non-functional behaviors are particularly worrisome in COTS software products. Non-functional behaviors can include malicious code (Trojan horses, logic bombs, etc.) and any other behavior or side effect that is not documented.

Another worrisome problem facing CBSE and CBD is "hidden interfaces". Hidden interfaces typically are channels through which application or component software is able to convince the operating system to execute undesirable tasks or processes. An example would be an application making a request to attain higher levels of permissions than the application should. Interestingly, IPA partially can address this issue by detecting hidden interfaces and non-functional behaviors by forcing software systems to reveal those behaviors/events after the input stream of a COTS component receives corrupted inputs.

Summary

This paper has briefly discussed interoperability issues for both software component failures and a lack of scalable composability theories for the “ilities”.

Component failures and how they propagate is a fascinating prediction problem in software engineering. IPA is a technique geared towards addressing it. And hidden interfaces and non-functional behaviors are problematic for CBSE and CBD.

In order for CBSE and CBD to flourish, technologies must exist that allow for the successful predictability as to how interoperable different software components are. Without predictability, interoperability cannot be known *a priori* until after a system is built. And it may be too late in the life cycle to financially recover if it is discovered that one or more of the components are not compatible.

Formal Approach of the Interoperability of C4IRS Operating within a Coalition

(*Approche formelle de l'interopérabilité de systèmes entrant dans une coalition*)

Michel Barès
DSA/SPOTI
18, rue du DR Zamenhof
92131 Issy les Moulineaux Cedex
France
Telephone : (33) 1 41 46 22 11
Fax : (33) 1 41 46 33 13
E-mail : michel.bares@dga.defense.gouv.fr

Abstract

Coalitions between nations are formed to face either a crisis or emerging minor conflicts. These coalitions are formed for the purpose of increasing efficiency, by the coordinated action of military means and the gathering of their related technical systems, for instance : networks, C4IRS. In merging these systems, we have to cope with a major problem, which is to have heterogeneous systems intercooperate The verb i**ntercooperate** is intentionally used to highlight the new needs differing completely from the simple exchange messages. The heterogeneity of these systems, inherent to national design and applications concepts, generates big deficiencies at the interoperability level. Since the solution of making gangways is not easily and reasonably generalized, the right thing to do is to provide all systems entering in a coalition with **interoperability mechanisms.** In this paper, we propose a formal approach which is relying on three main concepts : openness structure for a coalition, interoperability space with the definition of an interoperability matrix, intercooperability domain in which we are able to define parameters that allow us to assess interoperability from different points of view.

Résumé

Les nations sont de plus en plus souvent conduites aujourd'hui à former des coalitions, dès que se profilent de par le monde, soit des crises soit des conflits mineurs. Ceci, aux fins d'être plus efficace par la coordination de leurs moyens militaires respectifs et la réunion de leurs systèmes techniques afférents : réseaux, systèmes de commandement. La réunion de ces derniers, dès que l'on cherche à les faire coopérer, pose un difficile problème consécutif à leur hétérogénéité. La solution des passerelles n'est qu'une solution d'attente ne pouvant être raisonnablement généralisée ; aussi, convient-il, de doter ces systèmes de mécanismes d'interopérabilité. Dans cet article on propose une démarche formelle s'appuyant sur trois concepts principaux : structure d'ouverture pour une coalition, espace d'interopérabilité avec une définition de matrice d'interopérabilité, domaine d'intercoopérabilité qui permet d'évaluer l'interopérabilité sous différents angles.

Keywords : interoperability, cooperative systems, distributed systems, knowledge shareability.
Mots-clés : interopérabilité, systèmes coopératifs, systèmes distribués, connaissance partageable

1 INTRODUCTION

A coalition is put in place to face an unusual situation relative to a crisis or upcoming conflict. It is generally formed for the very purpose of increasing efficiency, by the coordinated action of military means and the gathering of their relating technical systems : networks, C4IRS.
In observing a coalition we can notice that :

- A coalition aims at a goal in order to make the situation evolve in a way favorable to the partnership's interests.
- A coalition is composed of different elements, in particular the ones which amalgamate computers , software, networks to sustain command aids.

- The systems put in the coalition are engaged to **(inter)cooperate** for executing a common mission, which has been established under particular conditions, with temporal constraints. The term (inter)cooperate is intentionally used (at that point) to highlight the new requirements of a coalition which are completely differing from the simple exchange messages, which can be illustrated by the following fact : a person who realizes, either by a lack of knowledge or insufficient know-how, that he alone cannot achieve an objective, is going to request the other people's assistance. Therefore when people are start to bring mutual assistance, they place themselves in a cooperation framework which cannot be successful if its members are not willing :
- To exchange more than information-data carried by simple messages but knowledge, this one must be enriched as long as the process is going on (validity of a knowledge may be depending on time).
- To exchange know-how in operating processes and methods application.
- To contribute to elaborating tasks belonging to dynamic processes.
- To share, in timely and appropriate conditions, useful knowledge for the evolution and the action of other agents of the cooperation.

What we can say at that point is that each participant of the coalition may be a great help to others if and only if they are able to put a bit "intelligence" in the different levels of the interoperability mechanisms.

1.1 Arising matters with a coalition

In the coalition's view, the first step we have to take is to make these systems (inter)cooperate in the coalition framework. One has a real and major problem to cope with, because most of the time, they are **heterogeneous**, as a result, they present big deficiencies at the interoperability level. One could object that it is always possible, to solve this question by making gangways, but one should be aware of what that represents a temporary solution. What is more, this solution cannot be easily and reasonably generalized as the number of systems in the coalition is getting more and more important. Therefore what seems reasonable is to design new concepts of **interoperability mechanisms** which can be implemented in each of them. Plus, we are going to consider that these systems placed in a coalition to be (inter)cooperative systems, they must have certain abilities like :

Openness ability :

Quality of a system, previously connected with others, to share a common understanding with them, relative to some matters of a coalition. For instance : ground evacuation, medical assistance. We can demonstrate that the openness of a system is a subset of the structure openness of the coalition.

Interoperability ability :

Capability of a system to (inter)operate with (interoperable) actions, relevant to the cooperation, more precisely orders and missions fixed within the coalition. Characteristics may be attached to it : possibilistic measure, interoperable competence, matrix of interoperability.

Intercooperability ability :

We will consider that a system is intercooperable when it is able to share its knowledge but also know-how with its neighbouring systems, in an optimal way, according to the comprehension it can get of the evolving situation.

Ability to conduct actions :

One admits that a system owns all the competence to do the required job in the coalition, and consequently, it can completely interoperate and furthermore intercooperate on all actions assigned to it. Of course this ability can fail if the conditions of temporal intervals are not strictly enforced; an action is only valid in a precise temporal interval.

1.2 Formal approach to a coalition (domains)

In our approach, we are about to define a space in which we will establish a distinction as in [Bares-1996], between three main domains that must be taken into account. These domains are obviously structured and in addition have semantic links. Their presence is all more justified that systems in order to be intercooperative require certain criteria and characteristics which are defined in these domains.

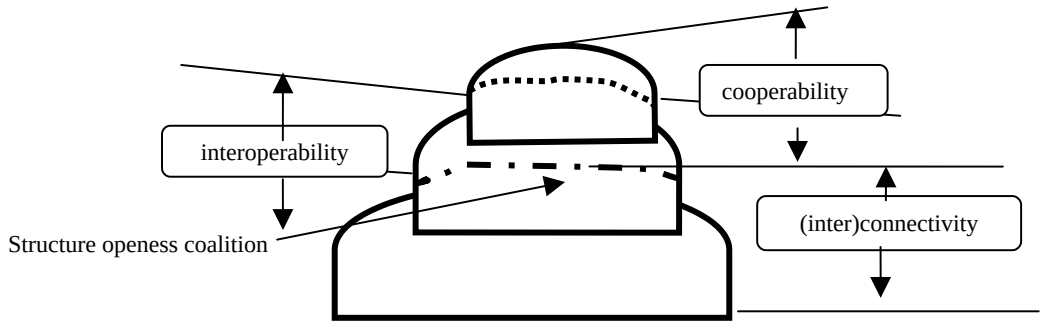


fig 1 Domains concerned by a coalition

Let us briefly give some details about the role of these three domains.

(Inter)connectivity:

This concerns essentially all necessary means to allow systems to communicate with each other, through a liaison and its relevant software mechanisms. We will consider interconnectivity in our approach as a prerequisite of interoperability.

Interoperability :

If we consider now that C4IRS systems must exchange more than simple messages, i.e., knowledge, we must go beyond interconnectivity framework, because the exchange of knowledge supposes that we have symbolic representations to carry this knowledge. Moreover, C3I systems in the future will be called upon, to bring each other a mutual assistance (a requisite in the NATO definition of C3IS) in their cooperative action to reach a common objective (called intercooperation later on). In such a perspective, C4IRS systems must be in position to have a mutual comprehension of what they are doing, of what processes they are running, and so on. At that point, we have to determine modalities that can obtain "intelligence'' and how to interpret it, in the exchange mechanisms.

To sum up, we can characterize the interoperability domain by the following points :

- A C4IRS becomes interoperable when it can organize itself and enrich its exchanges within an **openness structure** characterizing the coalition.
- The precedent point represents a necessary but not sufficient condition in an interoperable exchange; in addition, we need to have a common vision of the universe in which systems are going to cooperate with others.
- To start by taking into account semantics in the mechanism of exchange.

Intercooperability :

This represents the final objective to reach, through the definition of a world, in which all (cooperative) systems are able to share all elements constituting their common activity in the coalition, but also, to take systematically advantage of everything that is appealing to intelligent behavior.

1.3 Introduction to the openness concept

The role of what is called in this paragraph **openness domain**, is to specify, beyond interconnectivity, ways and limits of opening which are necessary to have a basic interoperability within the coalition framework.

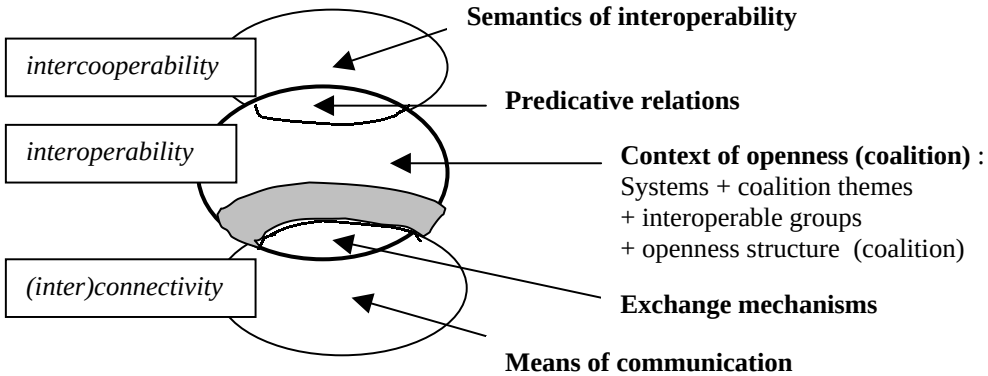


fig. 2 Openness context place in interoperability

If we consider that a coalition can be identified by :
- coalition-themes pertaining to the global mission assigned to the coalition for instance : medical assistance or civil rescue (depending on politics of the involved allied nations); they are also regarded as a set of knowledge required for it and describing a speciality, a feature, an ability,
- specific missions devoted to systems, a mission that can be regarded as a set of elementary (interoperable) actions may be devoted to the different systems participating to the coalition.
We will have to mention what systems are involved and how they are (relationships existing between systems and themes of the coalition).

- *Designation of a system :*

A **system i** will be designated by : $\mathbf{S^i}$ where $i \in [1, n]$, (n = number of systems placed in the coalition). They are supposed to be able to share a minimum common knowledge and to have common comprehension of fundamental orders.

- *Designation of coalition-theme* :

A **coalition-theme** will be designated by $\mathbf{T}_t$ with $t \in [1, q]$ (q is the maximal number of themes of the coalition). $\mathbf{T}_t$ encompasses a variable number of elementary actions (depending on the mission). An action **j** will be designated by $\mathbf{A}_j$. These themes can be stated by syntactic formulas obeying the syntactic rules of a formal language.

2 FORMALIZATION OF THE COALITION OPENNESS

After having designated first elements taking part to the coalition, we have to determine the network of existing relationships between systems and the coalition-themes in a formal way. In doing so we will define the concept of a context of openness the formalization of which is based on some mathematical notions : relation, Galois connections and lattices.

2.1 Concept of a context of openness

The concept of **openness context** enables us to emphasize the semantic point that will be attached to coalition-themes and systems operating in the coalition. We will formally define a context of openness by a triplet :

$$(S,\ T,\ R),$$

where : $S :: \{S^i\}_{i = 1,2,..,n}$ the set of the systems,

$T :: \{T_t\}_{t = 1,2,..,q}$ the set of the themes specified in the coalition,

R is a binary relation : $R \subset S \times T$.

The context may be given a priori when the coalition, put in place, is defining the mission of every system. It can be also defined a posteriori when the coalition is running and evolving.

example : 3 systems S^1, S^2, S^3 of three different nations are asked to interoperate within the framework coalition for rescuing civil people in an African state. Three particular coalition-themes are defined : ground evacuation operations (T_1), airborne transportation (T_2), logistical medical aid (T_3). This supposes that systems can (inter)operate on different actions relevant to the coalition-themes and secondly to exchange knowledge required to achieve their respective missions. Let us suppose, in defining the openness context, we obtain following couples :

$R(S^1, T_1), R(S^1, T_2), R(S^1, T_3), R(S^2, T_1), R(S^2, T_2)., R(S^2, T_3), R(S^3, T_1), R(S^3, T_2)., R(S^3, T_3) \subset S \times T$, we get a particular case which means :

the relation **R** on $\{ S^1, S^2, S^3 \} \times \{T_1, T_2, T_3\}$ is **total**

This openness context is summarized by the table :

Relation R	T1	T2	T3
S1	*	*	*
S2	*	*	*
S3	*	*	*

Tab. 1 openness context example

Considering strictly the semantic point of view, systems are totally open to the themes involved in this coalition. This example describes a situation which is ideal and will rarely take place in reality. From a

strict point of view, S^1, S^2, S^3, must be considered as **totally open** on coalition-themes. Consequently, we get a unique totally open couple in the sense of a Galois connection:

$$(\{ S^1, S^2, S^3 \} \times \{T_1, T_2, T_3\})$$

2.2 Interoperable Group (IG) notion

The table 1 describes an ideal case, because all systems of the set S are related to all themes of the set T.

We will define a condition of openness as follows:

$$\exists\ i, t \mid S^i \in S \text{ and } T_t \in T, \text{ we have : } (S^i, T_t) \subset R.$$

Now let : $S :: \{S^i\}_{i = 1,2,..,n}$, $T :: \{T_t\}_{t = 1,2,..,q}$,

$$R \subset S \times T.$$

We define an (totally) **interoperable group** as :

$$\text{interoperable-group} :: < \text{IG-}\#(< s > \rho < t >) >$$
$$\text{where : } s \in P(S) \text{ and } p \in P(T)$$

Remark:

ρ : to indicate that R is a total relation on s x t,

: all IG must be numbered to construct the lattice of the openness coalition later on,

in other words, there exists only one dependency between the subset s and the subset t.

2.3 Openness structure of the coalition

The IG represents an interesting notion because it enables us to represent formally a structure of the coalition openness. We are actually going to obtain a formal structure that will have interesting properties, the ones of a lattice. Let us illustrate that with an example, for that purpose, we consider a coalition C whose openness context is given by the following table :

	T_1	T_2	T_3	T_4	T_5	T_6
S^1	*	*			*	*
S^2		*	*	*	*	
S^3	*		*		*	

Tab. 2 openness structure of the cooperation C

First of all, we notice that the openness context of C is composed of 8 subsets. In view of what precedes we can “translate” this context through IG and we will obtain one after the other :

$$\text{IG-1 } (\{ S^1, S^2, S^3 \} \ \rho \ \{T_5\}),$$
$$\text{IG-2 } (\{S^1, S^2\} \ \rho \ \{T_2, T_5\}),$$
$$\text{IG-3 } (\{ S^1, S^3 \} \ \rho \ \{T_1, T_5\}),$$
$$\text{IG-4 } (\{ S^2, S^3 \} \ \rho \ \{T_3, T_5\}),$$
$$\text{IG-5 } (\{S^1\} \ \rho \ \{T_1, T_2, T_5, T_6\}),$$
$$\text{IG-6 } (\{S^2\} \ \rho \ \{T_2, T_3, T_4, T_5\}),$$
$$\text{IG- 7 } (\{S^3\} \ \rho \ \{T_1, T_3, T_5\}),$$
$$\text{IG-8 } (\{\varnothing\} \ \rho \ \{T_1, T_2, T_3, T_4, T_5, T_6\}).$$

We are able now to construct the diagram with the different IG we previously determined. As shown below, the nodes of the graph correspond to numbers IG.

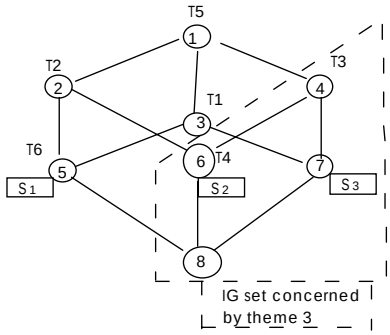


fig. 3 Open structure of a coalition C

Fig. 4, which represents the only possible openness structure of the coalition C. This one is obviously depending on the way of fixing the context. This openness structure presents a great deal of interest because from this diagram, we can interpret easily the openness structure when considering the following points :

- Every IG indexed by a number inherits all themes linked up to it in the diagram.
- Every node number is constituted of all the systems which are linked down to it.
- From this diagram we can envisage different consequences of C decision-makers'acts upon basic interoperability: elimination of a link, assignment of a system to a coalition-theme, suppression of themes or restriction of a node for security reason, et cetera.

3 CHARACTERISTICS FOR INTEROPERABILITY SPACE

As shown on the figure 3, the openness context represents the first step of the interoperability space, in the sense that we have determined different criteria concerning the coalition conditions, which must be taken into account in the interoperability mechanisms, particularly in predicative relations (definition of the interoperability competence). Space interoperability relies upon different notions whose certain of them are appearing on the figure below.

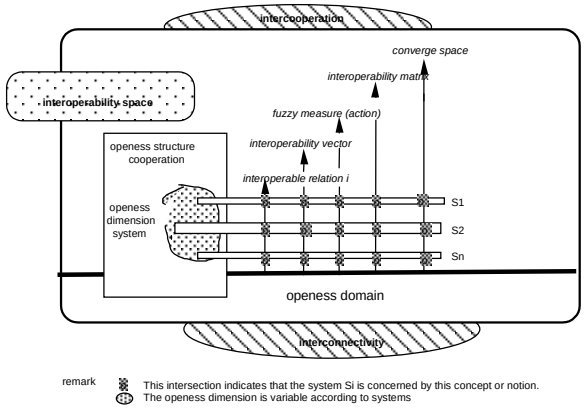


fig. 4 Interoperability space

3.1 Interoperable action

We will consider that an action is not interoperable in itself, but only with system(s) that are able to handle it. For that reason, we will always designate an interoperable action by a couple :

$$(S^k, A_j) \text{ where } S^k \in \{S^1, S^2, ..., S^n\}$$

Remark : This couple : (S^i, A_j) must encompass time variable (reification), because systems, and more actions, are likely to modify in run time. We will consider that its validity will depend on a **temporal window** or « window opportunity », which will be denoted as follows :

$$(S^i, A_j, \theta_M),$$

the system i acts (or (inter)operates) on the action j,
in the temporal interval θ, assigned to the mission M.

The time parameters will be fixed by those who are in charge of the coalition.

3.2 Definition of a predicative relation of interoperability

Presently we define a **relation** $\Re$, in a propositional calculus view, the arity of which is 3, and by which any system gauges its aptitude to operate an action of the coalition. This relation must be applied by every system to every action of the cooperation. It will be denoted as follows :

we form the proposition : $\Re(S^i, \{A_j\}, \theta_M)$,

$\forall\, i \in [1, n]$, (n : number of systems), $\forall\, j \in [1, p]$, (p : number of actions)

Remark :

S^i considers that it is competent to interoperate on $\{A_j\}$,

We suppose that all Aj can be described in a formal way (formal language words).

Each system is bound to determine a first condition, **necessary** but not **sufficient** of its interoperability. According to its own knowledge and truths about its neighboring world, a system is able to say if such an action is normally interoperable. In fact, the relation $\Re$ which allows to define an **effective interoperability** : a system S^i gauges its competence to operate on any action, in window time θ attached to the mission framework M, under normal and usual conditions.

As $\Re(S^i, A_j, \theta_M)$ is considered like a proposition,

so we can assign a truth value to it :

if Value(Val) $[\, \Re(S^i, A_j, \theta_M)\,]$ :: True (T/1)

that means :

S^i can interoperate on A_j, in time window θ, fixed by mission M. $\forall\, i \in [1, n]$, $\forall\, j \in [1, p]$

Val $[\, \Re(S^i, A_j, \theta_M)\,]$ :: False (F/0) $\Rightarrow$ **interoperable incapacity** of S^i on Aj.

Remark : In practice, those who are responsible for S^i are entitled to apply this relation, and thus, to decide about the (in)capacity of their interoperability in the light of the current context in which they are going to place their actions (in a formal way S^i is interpreting in its own possible world).

3.3 Fuzzy cube for an interoperable action

A fuzzy measure refers to a means of expressing uncertainty when, not disposing of complete information, it is impossible to use probability. We are going to determine numerical coefficients (in a subjective way), or **certainty degrees**, to indicate how it is **necessary** that such a system can interoperate on (or with) such an action beforehand declared as **possible**. In doing the (reasonable) hypothesis that a system only executes one interoperable action at a time, we can for instance, form a **universe W** from the following singletons :

$W = \{ (S^i, A_1), (S^i, A_2), (S^p, A_3), ...,(S^q, A_n)...\}$, with : $d(S^i, A_n)$ :: degree of possibility

$d(S^i, A_n) \in [0, 1]$, this value assesses the possibility which S^i executes the action A_n.

A fuzzy measure is completely defined as soon as a coefficient of possibility has been attached to every subset of a **universal set U**. If the cardinal number is n, to be rigorous, we must state 2^n coefficients, in order to specify the measure of possibility. Here, we will proceed more simply in observing that each subset of U may be regarded as an union of singletons it encompasses. So, the determination of the possibilistic measure can be done from only n elements. To define an interoperable action we here introduced :

(a) A **feasibility** measure comparable to a possibility,

(b) A **imperativity** comparable to a necessity which will be dual of (a),

(c) A **credibility** measure to assess trust put by systems in the fulfillment of an action by anyone of them.

(a), (b), will be defined thanks to distributions of possibility. Therefore we will represent an interoperable measure in a “fuzzy cube”.

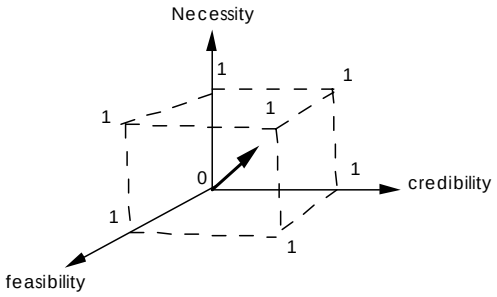


fig. 5 Fuzzy representation of an interoperable action

3.4 Matrices of interoperability

For a given system S^i, if we successively apply the relation $\Re$ to couples (S^i, A_j), j varying from 1 to p, we obtain for example:

$$\text{Val } [\Re (S^i, A_1)] :: T$$
$$\text{Val } [\Re (S^i, A_3)] ::: F$$
$$.....$$
$$\text{Val } [\Re (S^i, A_p)] :: T$$

Tab. 3 application of the relation of interoperability

We bring together these elements in order to get a binary vector. There are as many vectors as systems in the coalition.

Let a component of vector $V(S^i)_j$ (row j), if we have :

$$\text{val } [V (S^i)]_j :: F$$

$\Rightarrow \neg\exists\Re (S^i, Aj)$ for openness structure of the coalition, and therefore,
Si has no semantics to evaluate, [V (S^i)]j is not supposed to exist.

From the binary vector, or from the world resulting of the interpretation $\Re$, it becomes possible to affect fuzzy measures to each vector's components whenever the value is not false. These fuzzy vectors will be established in the following conditions :

We take: either couples of the world $\Re$, such as :
$((V (S^i)_{j = 1,2,..p}) :: 1$, or vector's elements V (S^i), such as :
$[[V (S^i)_{j = 1,2,..,p}]] (\Re) :: 1$

We assign a fuzzy measure to them, respectively corresponding to 3 dimensions, as described in fig. 5 :

$\Phi (S^i, Aj) \rightarrow$ measure of feasibility,
$N (S^i, Aj) \rightarrow$ measure of necessity,
$\lambda (S^i, Aj) \rightarrow$ measure of credibility.
$i \in [1, n], \ j \in [1, p]$

Every system is able to establish its own interoperability vectors.

whenever for $j \in [1, p]$, val $V (S^i)_j = T \rightarrow$ semantics evaluation to do.

This evaluation of the semantics has been made necessary because : either unpredictable facts arrived in the own system's world or an unexpected mission could have modified the world of S^i; which means that A_j has no longer the same meaning for the system S^i and possibly also for the coalition. In gathering all vectors of interoperability $V(S^i)$, we get this way, what call an **interoperability matrix**.

$$[I (S^i)_{i = 1,2,..,n}] = [V(S^1) V(S^2) .....V(S^n)]$$

This matrix represents only an apparent interoperability. It can be used in different ways :
- to indicate what is theoretically the most interoperable system, relatively to a determined action,
- to give the most adequate system to operate under special conditions : a mission which imposes a temporal constraint to operate an action. We will construct three kinds of interoperability matrices.

a) Matrix of feasible interoperability

This matrix gives a dimension of feasibility of the interoperability of $\{S^i\}$ will be denoted by :

$$[\ \text{I-}\Phi\ (S^i)_{i\ =\ 1,2,\ldots,n}\ ]$$

b) Matrix of imperative interoperability

The matrix of necessary interoperability is also constructed with fuzzy vectors of necessity as described above. It presents a great interest in informing us about necessary conditions which are imposed to some systems in their way of interoperating. This matrix will be denoted by :

$$[\ \text{I-N}\ (S^i)_{i\ =\ 1,2,3}]$$

Example with 3 systems and 4 actions :

$$[I - N(S^i)_{i=1,2,3}] = \begin{bmatrix} 0.8 & 0.8 & 0.0 \\ 0.0 & 0.6 & 0.8 \\ 0.8 & 0.0 & 0.6 \\ 0.8 & 0.6 & 0.8 \end{bmatrix}$$

Tab. 4 matrix of imperative interoperability

We observe that in the previous matrix, system 1 must have the strongest interoperability in spite of its component $I(S^i)_{2,1} = 0$, which can incidentally indicate an interdiction to interoperate on action A_2.

c) Matrix of credible interoperability

This matrix gives us a visibility on systems which are about in the best position to interoperate successfully. It will be denoted by :

$$[\ \text{I-}\lambda\ (S^i)_{i\ =\ 1,2,..,n}]$$

Example with 3 systems and 4 actions :

$$[I - \lambda(S^i)_{i=1,2,3}] = \begin{bmatrix} 0.3 & 0.0 & 0.3 \\ 0.0 & 0.0 & 0.3 \\ 0.8 & 0.3 & 0.3 \\ 0.3 & 0.6 & 0.3 \end{bmatrix}$$

Tab. 5 matrix of credible interoperability

We observe that in this example, system 2 presents small degrees of credibility; this means that all systems consider that it is likely to be the least successful to interoperate in the coalition.

4 COOPERABILITY SPACE CONCEPT

In this paragraph, we will try to go beyond the system's interpretation regarding actions and to see how any systems can interpret the other systems'ability for interoperating on actions. What one can summarize simplistically :

(1) interoperability $(S^i) \rightarrow$ system S^i interprets [S^i (interoperability) / {action(s)}], $\forall$ i $\in$ [1, n]

(2) intercooperability $(S^i) \rightarrow$ systems $\{S^k\}$ interprets [$(S^i$) interoperability /{action(s)}], $\forall$ i, k $\in$ [1, n]

We can still illustrate (1) and (2) in an explicit manner :

(1) for the domain of interoperability :

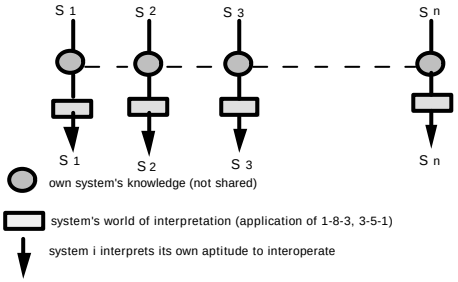


fig. 6 Interpretation in the interoperability domain

(2) for the domain of intercooperability :

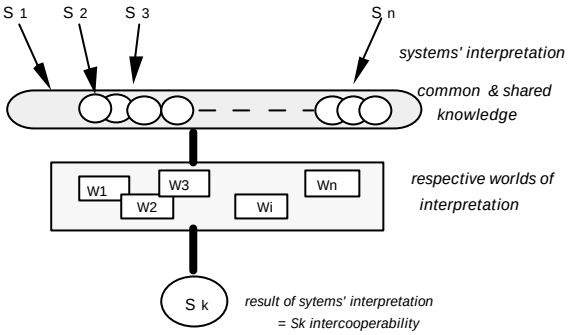


fig. 7 Interpretation in the intercooperability domain

4.1 Relation of intercooperability

So, what is going to get more important for systems in intercooperation it is the necessity to satisfy a permanent need of **mutual understanding**. In a practical way, that means they must :

- either share the same meaning relatively to the different objects they have to handle in their common universe's actions,
- or to take necessary steps to make **semantics** converge.

When defining the relation of the interoperability competence in 3.2, we have considered that systems, obviously placed in a symbolic context, are able to interpret their own ability to interoperate on actions as requested by the coalition. In this paragraph, we are now envisaging to go beyond, by seeking to extend the system's interpretation ability in defining a **relation of intercooperability competence**.

We will consider that a system S^i has a competence in intercooperability when, it will be able to "judge" the ability of adjoining (cooperative) systems to interoperate on a set of actions $\{A_j\}$, in a time window θ_M. This competence will be designated by the following quadruplet :

$$Si / Sk , \{ A_j \}, \theta_M \text{ (the symbol / indicates the way of interpretation)}$$
$$\forall i , k \in [1,n], j \in [1, p]$$

We will define a relation of intercooperable competence in the same way we do for the interoperability competence, this one will be designated by $\Re'$, in the following conditions :

$\Re'$ :: « is able to (inter)cooperate »

$\Re'$ :: « interpret the other systems' aptitude to interoperate on $\{A_j\}$ », we form the predicate relation :

$$\Re' [S^i / (S^k, \{A_j\}, \theta_M)] \forall i, k \in [1, n], \forall j \in [1, p].$$

This means that : S^i judges that (its confidence in the success of) S^k is able to interoperate on {Aj} in the time-window θ_M (this evaluation is made with a fuzzy measure of credibility).

In a predicate calculus view, the relation $\Re'$ defined in these conditions, is equivalent to a **propositional function**: S^i, S^k, A_j, representing the variable, θ_M may be considered here as a constant [1]. So, for a given S^i, we can evaluate the truth value of the predicate : S^k is interoperable on each $A_{j=1,2,..,p}$

(1) if *Val* $[\Re'[S^i / (S^k, \{A_j\}\theta_M)]$ :: True,

that means : S^i interprets that S^k is able to interoperate on the actions $\{Aj\}_{j=1,...,p}$,

(2) if *Val* $[\Re'[S^i / (S^k, \{A_j\})]$ :: False,

S^i considers that S^k is unable to interoperate on $\{Aj\}j=1,...,p$,

Nota bene : θ_M has been considered as a constant in (1) and (2), for previously mentioned reasons.

4.2 Fuzzy matrices pertain to Intercooperability

In gathering the vectors of intercooperability, we will get what we are now calling an **intercooperabilility matrix**. Although the interoperability matrix is unique, it is necessary to establish two categories of matrices in the domain of intercooperability.

[1] We make the hypothesis that the time-window's limits are well defined in the coalition.

1) The first category, called **intercooperability-system,** is going to indicate how the set of systems interpret their respective interoperability .
2) The second one, called **intercooperability-action**, is regarding actions, i.e. a matrix to comprehend the interoperability of the coalition from its elementary actions.

Matrix of intercooperability-systems

Now the question is to comprehend how the set of systems of the coalition, interprets the ability of interoperating one of them. Let us keep in mind that all systems are more or less interoperable according to the other systems' judgment. The intercooperability matrix of a system S^i will be denoted : **[C(S^i)]** and presents a great interest. In fact, we can make special computations about rows and columns of **[C(S^i)]**. Therefore, we obtain some interesting elements to characterize what we are going to call **intercooperable capacity** of the coalition, i.e. the visibility about the more or less easiness of system's interoperation.

Properties of a column

Let [C(S^k)] be the matrix of intercooperability-system of the system S^k, and consider the m^{th} column of this matrix. If we sum up all components of the **vector-column m** of the matrix [C(Sk)], we are going to get a certain scalar, designated by : $\alpha_c(S^k)$.

$$\alpha_c(S^k) = \sum_{j=1}^{p} \left[C(S^k)\right]_{j,m}$$

this scalar indicates how S^m assesses the interoperability « strength » of S^k regarding the actions of the cooperation. By way of an example it can be interesting to examine the following cases and consequences for the coalition :

- $\alpha_c(S^k) = 0$,

according to its evaluation, S^m considers that the system S^k is not interoperable : S^k must play no role in the coalition because it is unable to execute any actions. This does not mean that S^k has no interoperable capacity, as long as we do not know the other systems' assessment. In the cooperation framework, several issues are possible :

- Is there a misjudgment between S^m and S^k ?
- Is there nothing in common between S^m and S^k , like knowledge, processes, etc.?
- S^m and S^k cannot work together for reasons (the coalition would be better off if it tries to explain).

- $\alpha_c(S^k) \neq 0$,

That means S^m trusts more or less in the aptitude of S^k to interoperate with however a distinction :

if $\alpha_c(S^k)$ is in close proximity to 0, its trust is weak and S^m thinks that S^k is likely to fail in its interoperating processes,

if $\alpha_c(S^k)$ is in close proximity to 1, its trust is very strong, S^k must be strongly successful in its interoperations.

Matrix of intercooperability-action

We now define an other kind of matrix which is going to allow us to have a visibility of the intercooperability of all systems of the coalition. This special matrix is going to indicate what are the systems which are in the best conditions to interoperate on actions. These matrices will be called **matrix of intercooperability-action** for that reason**.** Let us go back to the matrices of intecooperability-system; if we take the j^{th} row in each of the previous matrices, we are forming a new matrix that reports about the systems' intercooperability capacity relatively to the action A_j. This matrix will be designated by [C(Aj)].

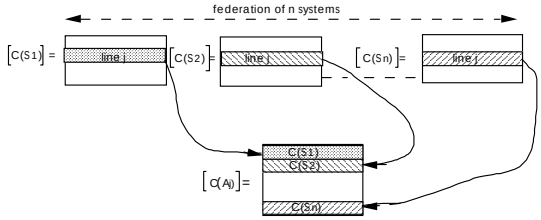


fig. 6 Matrix of intercooperability-action

The matrix of intercooperability-action presents interesting features :
- its shape is square,
- it allows to understand what are the systems of the coalition which are in the best position to interoperate on such an action A_j,
- it gives an idea about the lesser or greater systems' easiness to interoperate on particular actions,
- its columns and rows have interesting characteristics.

If we compute the p matrices corresponding to all actions of the coalition, we have a good visibility of the intercooperability in the coalition framework. That means we are able to say :
- what are the actions which are difficult to carry out,
- what are the ones which are likely either to get the coalition into trouble or to force the coalition to face difficult issues.

5 CONCLUSION

In this paper, we have introduced notions of openness context and interoperable group. We have afterwards demonstrated that it was possible to formalize the structure openness of a federation of systems representing the coalition. Then we have defined a notion of an interoperable action to which we have attached fuzzy measures **: feasibility, imperativity, credibility (**determined through distribution of possibility). By introducing a predicative relation for interoperability, we have shown later on that it was possible to construct a vector of **effective interoperability**, resulting of the system's interpretation of the facts in its own logic world. In this way, we got a quantitative evaluation of interoperability pertaining to a system of the coalition. The gathering of these vectors of interoperability enables us to define a **matrix of interoperability** which gives a right visibility about the global interoperability pertaining to the set of all systems of the coalition. We afterwards went beyond this ability of a system to interpret its own ability of interoperating and to see how it could interpret the other systems' ability for interoperating on actions. For that purpose, we have extended the relation of interoperability through a relation of intercooperability competence. This relation enlarges our comprehension field about the interoperability. We establish two kinds of matrices; the first one regarding the systems' interoperability, the other one concerning the actions. These matrices also present interesting properties, which have allowed us to establish a whole family of parameters, and doubtless represent a first significant step in our way of seeing the interoperability issue.

BIBLIOGRAPHICAL REFERENCES

[**Alberts-99**] Alberts D. S., Gartska J. J.,Stein F. P. Network centric warfare : Developing and leveraging Information Superiority. C4ISR Cooperartive Research Program (CCRP). Publications series.

[**Arkin & Al-90**] Arkin W. M. & Al. Encyclopedia of the US Military. Ed : Harper & Row. New York 1990.

[**Quine-82**] Quine W. V. Methods of Logic. Harvard University Press. Cambridge (MA), 1982.

[**Athans-80**] Athans M. System theoretic challenges and research opportunities in military C3 systems. In : IEEE Conference on Decision and Control. Albuqurque, NM. December 1980.

[**Barès-00**] M. Barès. Interoperability modeling of the C4ISR systems - *La modélisation de l'interopérabilité des systèmes de commandement.* In : Symposium on "System Concepts for Integrated Air Defense of Multinational Mobile Crisis Reaction Forces". Valencia – Spain May 2000

[**Barès-99**] M. Barès. Contribution to the formalization of the openness and interoperability of C3I systems. Research report. George Mason University, Center of Excellence in Command, Control, Communication, and Intelligence, System Architecture Laboratory. Fairfax 1999.

[**Barès-97**] Barès M. & Chaudron L. Interoperability of systems : from distributed information to cooperation. In IJCAI 97 « IA in Distributed Information Networking », Nagoya Japan, 22 Aug 1997

[**Chaudron-95**] L. Chaudron. Lattices for symbolic fusion. In OSDA'95, International Conference on Ordinaland symbolic Data Analysis. Paris une 1995.

[**Chaudron-96**] L. Chaudron. Firststeps towards first order logic in formal concept analysis. Conference on conceptual Knowledge Processing, Darmstadt, Germany, 28 Feb. 1 Marc.1996.

[**Chaudron-96**] L. Chaudron. SpaceFormAn : Empirical means for formal concept analysis of large data sets.
In Conference Ordinal and symbolic data, Darmstadt, Germany, 19-21 March 1997.
[**Clark- 99**] Clark T, Jones R. Organisationnal Maturity Model for C2. In : CCRT Symposium. Naval war college. June 29 - July 1, 1999.
[**Coakley-91**] Coakley T. C3I : Issues of Command and Control. National Defense University Press. Washington, DC.
[**CURTS-99**] Curts R. J. and Campbell D. E. Architecture : The road to Interoperability. In : Naval war college. June 29 - July 1, 1999.
[**Ganter-96**] B. Ganter and R. Wille. Formal Concept Analysis mathematical foundations. Ed : Springer. 1996.
[Goodman-99] Goodman I. R. A Decision-Aid Nodes in Command and Control Systems Based on Cognitive Probability Logic. In : Naval war college. June 29 - July 1, 1999.
[**LISI-98**] Levels of Information Systems Interoperability (LISI); C4ISR Architecture Working Group. Manch 1998.
[**Mitchell-99**] A. Mitchell. Allied C4I Interoperability with the Joint Internet controller. In : CCRT Symposium. Naval war college. June 29 - July 1, 1999.
[**Maurer-94**] Maurer M. Coalition Command and Control. Directorate of Advanced Concepts, Technologies, and Informations Strategies. National Defense University. Fort Mc Nair, Washington, DC.
[**Wheatley-99**] Wheatley G., Buck D. Multi-National Coalition Command and Control beyond NATO. In : CCRT Symposium. Naval war college. June 29 - July 1, 1999.

In French

[**Barès-96**] M. Barès. Pour une prospective des systèmes de commandement. In : chap. 5. Ed : Polytechnica. Paris, 1996.
[**Bouchon-Meunier-95**] Bouchon-Meunier B. La logique floue et ses applications. Ed : Addison-Weley. Paris 1995.
[**Chazal-96**] Chazal. G. Eléments de logique formelle. Ed : HERMES. Paris, 1996.
[**Guigues-86**] JL. Guigues and V. Duquenne. Familles minimales d'implications informatives résultantes d'un tableau de données binaires. Math. Sciences Humaines, 95 : 5-18, 1986.
[**Haton-91**] Haton J. P. & Al. Le raisonnement en Intelligence Artificielle. Ed : InterEditions. Paris, 1991.

This page has been deliberately left blank

Page intentionnellement blanche

Information Interoperability and Information Standardisation for NATO C2 – A Practical Approach

Eddie Lasschuyt, MSc
Marcel van Hekken, MSc
TNO Physics and Electronics Laboratory
P.O. Box 96864
2509 JG The Hague
The Netherlands
Lasschuyt@fel.tno.nl / VanHekken@fel.tno.nl

1. Introduction

1.1. Rationale

Interoperability between information systems is usually 'achieved' by enabling connection at network level. Making systems really interoperable, by letting them understand and manipulate the exchanged information, requires a lot more. Above all, *information standards* are needed in order to gain common understanding about what will be exchanged. Besides that, information standardisation should be considered from a *global* point of view, taking into account the whole range of systems that will potentially exchange information for a certain purpose. The importance and complexity of information standards are often underestimated. Gaining efficient and effective interoperability starts with thinking about information standardisation in its totality first.

1.2. Context

Coalition operations within NATO require extensive co-operation between military units and organisations of the participating nations. This means that interaction is needed at the Command and Control (C2) level in the first place. Information has to be disseminated in order to achieve optimal 'situational awareness' among all parties involved in an operation. For this purpose they require what is called a "common operational picture" (COP), being the same view on the battlefield.
The changed nature of military operations, producing increasing amounts of information, as well as recent developments in technology, have led to the widespread use of C2-supporting information systems. They are usually called "Consultation, Command, Control, Communications and Intelligence systems", in short "C4I" systems. Although many of the current systems are still under development and not fully integrated in the operational decision making process, nations are more and more dependent on these systems as backbone for their C2 information distribution and processing.
Combining these two facts, i.e. the importance of NATO-wide C2 information dissemination to obtain a COP and the increasing use of C4I systems, results in the need for C4I systems that are able to *co-operate*. This means that different systems, with different functionality, using different natural languages and made by different manufacturers, should be able to participate in an overall C4I network and seamlessly interchange operational information. We refer to this as *interoperable* C4I systems.

1.3. Overview

This article discusses a general and practical approach to reach interoperability among a *large* number of information systems of *different* nature. It is focussed on the subject of *information standardisation*, for the purpose of gaining interoperable *systems*. Based upon this approach, a number of considerations and recommendations are given for interoperability within the NATO C2 domain, i.e. between NATO C4I systems[1]. The article is primarily intended to give an overview of this problem area and to make the reader aware of its significance and difficulty. It could make him/her realise that information standards deserve more attention in his/her community (e.g. a policy division, Defence research lab or C4I-related working group) and it may trigger him/her to give more thoughts on the matter. We must emphasise, though, that some issues in

[1] When we say "NATO C4I system" this includes national systems that are potentially used in a NATO context.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

this article are not (yet) fully crystallised or haven't proven their value in practice (yet). Further discussion on these issues in international forums is strongly suggested.

In chapter 2 we start with a general introduction to interoperability, hereby setting the technical scope for this article. Chapter 3 defines the problem we want to solve. It does so in terms of possible interoperability architectures and factors that influence the choice. The theory of chapter 3 is applied in practice, on the NATO C2 domain, in chapter 4. Suggestions are given for improving the NATO standardisation efforts. Finally, chapter 5 summarises the conclusions made in the other chapters.

2. Information interoperability

2.1. Introduction

This chapter briefly explains what we mean by information interoperability, how an information standard fits in and why the latter is so important in establishing interoperability.

2.2. Types of interoperability

Interoperability is the degree to which entities are able to co-operate in achieving a common goal. There are many interpretations of the concept of interoperability between computer systems[2]. It varies from having a network connection and being able to transfer files or (a bit more sophisticated) send and receive e-mail, to using exactly the same applications at all systems and completely sharing the information they process.

In this paper we address *information interoperability*, achieved by the *automated exchange and interpretation of structured information* between/by systems. With minimum user intervention, systems must be able to *automatically* interchange certain information and utilise that for further processing. Especially important is the prerequisite that the information is *structured*, because this enables functionality such as distribution by subscription on certain topics, presentation of information on a map, fast search & retrieval facilities and filtering by specific selection criteria. The emphasis here lies on the exchange of *information* (rather than data), hereby preserving its meaning, integrity and context. Another precondition is that the exchange may not depend on proprietary products, such as database management systems and communication systems. In support of all these requirements, the information is often exchanged in a clustered manner, via some form of 'messages'. Summarised, this kind of interoperability offers an optimal connectivity between systems while preserving maximum independence of these systems.

Our definition of information interoperability is similar to what is called "level 4, 5 or 6 of interoperability" in NATO terminology [1]. This implies a physical connection between computer systems and (depending on the level) user-controlled accessibility restrictions. ATCCIS Replication [2] and automated ADatP-3 message exchange [3] are typical examples of this kind of interoperability.

2.3. Standardisation

The key notion for information interoperability is *standardisation*. By having common agreement on which information is exchanged, in what format, how this is done and under what conditions, it becomes easier to allow systems of different type to interoperate. Paragraph 3.2 explains about possible approaches of reaching interoperability, which depend on the degree of standardisation. We name the total set of agreements that make systems co-operate by exchanging information, an *interoperability standard*.

An important characteristic of an interoperability standard is its *scope*, which defines the objective of the standard, or, in other words, for what organisations, scenarios, systems, functions, etc. the standard is applicable. The scope must be clearly defined, so that it is absolutely unmistakable whether certain information under certain circumstances is part of the standard or not. An ambiguous scope will undoubtedly lead to confusion and possibly wrong use of the standard (for purposes it was not intended for).

[2] We deliberately do not consider interoperability in another context, for example between organisations. This article is focussed on system interoperability.

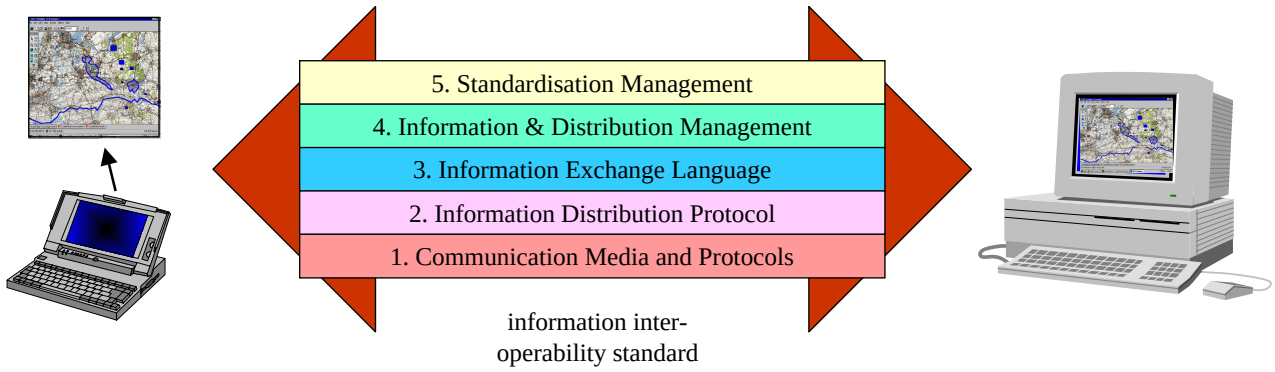


Figure 1 — Layered view on an interoperability standard

2.4. Interoperability layers

An information interoperability standard is composed of five layers (see figure 1), for each of which overall agreement is required:

1. Communication media and protocols.
   This layer provides basic data communication. Mostly existing (commercial or military) standards are used for this. Examples of such standards: TCP/IP, X.400, e-mail (SMTP), Combat Net Radio, CRONOS.

2. Information distribution protocol.
   Automated distribution of information (not: data) requires additional (higher-level) protocol standards in order to preserve meaning, integrity and context of the information. Concepts such as assured/confirmed delivery, transactional interaction, sequencing, queuing, forwarding, content-based routing, prioritisation, compression and encryption affect the way information is disseminated and must therefore be standardised. Examples of information distribution protocols: database replication, publish/subscribe.

3. Information exchange language.
   An unambiguous and structured description of the information to be exchanged within a certain scope is needed. Without a common understanding of the information, systems will never be able to interpret it in the same way. When a French C4I system reports a hostile unit to a German C4I system, both should have equal comprehension on the unit's size, location, etc. They must, so to say, 'speak the same language'. A *common exchange language*, or 'Esperanto', defines the semantics (what it means), the syntax (how it is structured) and the lexicography (how it is represented) of the information. Examples of exchange languages (see also par. 4.2): "Land C2 Information Exchange Data Model", AdatP-3.

4. Information & distribution management.
   In support of setting up and maintaining the exchange of information within an operational environment, certain additional rules and procedures must be agreed upon. They depend on requirements imposed by the environment. Examples relevant for the NATO C2 domain: security measures (e.g. an information classification scheme or a Public Key Infrastructure), rules on ownership of information, features that enable selectivity of information (e.g. filtering), a procedure to establish and manage information exchange contracts between organisations, rules that guarantee the use of world-wide unique information element identifiers.

5. Standardisation management.
   Finally, an interoperability standard cannot exist without proper agreement upon how to support the development and maintenance of the standard as a whole. This includes management organisations, procedures (e.g. for handling change proposals) and tools (e.g. for data modelling).

The lower layers have a more technical nature, while the upper layers consider more organisational matters. In our opinion, the *exchange language*, the middle (third) layer involving both technical and organisational issues, is the most challenging and important interoperability layer to be standardised. Therefore, this part of an interoperability standard, also called the *information standard*, will be the subject of the rest of this paper. (Many statements, however, also apply to the interoperability standard as a whole.)

2.5. Information standard

Standardising information as part of an interoperability strategy is an often-underestimated effort. A common exchange language is hard to obtain. We give three reasons. Firstly, unless the scope is very small, several organisations and/or nations will be involved. For that reason it will usually be very difficult to reach consensus on which information is relevant for exchange, how information elements relate to each other, what format is used for specific information items, etc. Every party has its own standards, habits, principles, technology and — above all — pride. Secondly, information analysis is a difficult process in which domain experts and information technologists need to co-operate closely. It takes a lot of patience and understanding to get a complete and unambiguous picture of a certain problem area. The information requirements may appear rather unclear and complex, which makes clarifying and structuring this an intensive and time-consuming effort. Thirdly, recording the results of an information analysis, for instance in a data model, requires special skills. The modeller must take into account many conditions in order to obtain an information structure that is broadly usable. It must be understandable for users, compact enough to implement, flexible so it can cope with future requirements (without radical changes), etc. In conclusion, making an information standard involves many players and many kinds of issues, varying from politics to technology, and is therefore a complex matter.

Besides being the most difficult one, the common exchange language (information standard) is also the most *important* interoperability layer in our view. Not that the other layers are unessential for achieving interoperability, but (in theory, in order to minimise the effort) they could be simplified very much by using commercial products and/or minimising the quality of distribution. The information standard, however, cannot be bought ‘off the shelf’ and cannot be reduced without narrowing the interoperability scope. More than the other interoperability aspects, the information standard affects the functionality that systems, by operating together, offer.

An information standard for exchange can be specified in several formats. Within the NATO context, the most widely used formats are relational data models, formatted messages and glossaries. We consider a *relational data model* as the best way to specify an information standard. Among other reasons this is because data models are commonly used and supported by methods and tools, represent a very unambiguous and structured definition of information (unlike glossaries), offer maximum flexibility in selecting information subsets to be exchanged (unlike formatted messages) and are both easy to be communicated with users and to be implemented in a database. Therefore this article assumes future exchange languages to be expressed in relational data models[3].

The next chapter outlines how information standards can be used to obtain interoperability between systems.

3. Interoperability architectures

3.1. Introduction

This chapter defines the problem of integral information interoperability at *large scale*. It does so by describing architectural aspects of interoperability and their effect on the efficiency of information distribution. We hereby concentrate on information exchange between *systems*, although in most cases we could also have spoken about organisations instead, because both represent a node in an information exchange network (technical versus operational view)[4]. Although the theory of this chapter is applicable in general, we mainly use examples out of the NATO C2 context.

---

[3] A rapidly rising specification technique is the eXtensible Mark-up Language (XML). It is in particular useful to define a standardised syntax for documents and messages. However, some kind of data model is still required to describe the meaning and context of the information. XML offers schema techniques for that, but these are less mature (regarding method, tools, etc.) than relational data models. Hence, we think that XML is a valuable instrument to pack information (in a message) for exchange between systems, but it should be applied in combination with relational data models that define the total set of available information.

[4] An (information) system is a local set of computers, databases, applications, etc. interconnected by a Local Area Network. An organisation utilises such a system. It supports the information supply that facilitates the business processes. When organisations are interoperable, in this article we mean their systems exchange information, usually over a Wide Area Network.

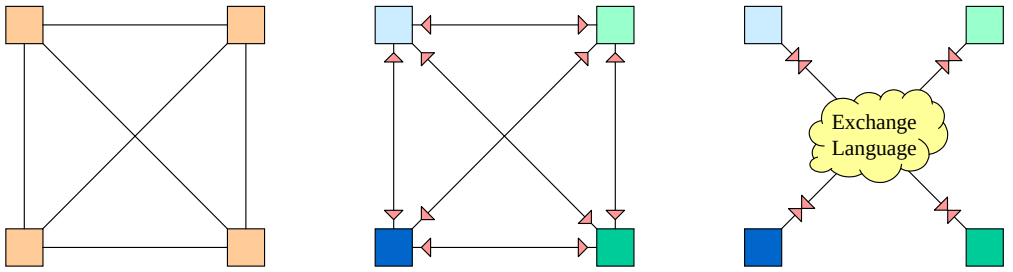
3.2. Basic architectures

As said before, information interoperability between computer systems is the ability to exchange and interpret information. In order to do so, systems must be able to ‘talk’ to each other and to ‘understand’ each other. The interaction is not necessarily directly. Instead, it may take place through one or more ‘interpreters’ or ‘translators’ (‘interfaces’ in computer terminology) that translate between information languages. There are three basic architectures for interoperable systems (see figure 2):

a. Standardisation of systems.
The internal architecture of each system is identical, including the information structure. One could say the interoperability standard is integrated within the systems. Information exchange is feasible without additional interfaces. This situation may occur when a distributed organisation is able to set up new systems for all its offices and base them on a single (standardised) architecture. A ‘corporate’ information standard is part of that. In most cases, however, this approach does not work, because the organisation will have to deal with different types of systems, often also legacy systems, with dissimilar internal architectures and information structures.

b. Bilateral exchange.
Every *type* of system[5] has its own internal architecture and uses a specific information structure. To exchange information, dedicated interfaces between each pair of interconnected systems are needed. The interfaces transform information from one format to another. For n systems connected to each other, this results in n(n-1) one-way interfaces in total. This solution is preferred when only two or three different types of systems are involved (and this will not change in the future). In case of more system types and more than a few interconnections, this architecture becomes highly undesirable, as it imposes numerous interfaces then.

c. Standardisation of the exchange language.
Every type of system has its own internal architecture and uses a specific information structure. Via an interface to a common exchange standard, information can be exchanged between all systems. In this case, n systems require only 2n one-way (or n two-way) interfaces in total. This third option is commonly seen as the most practical solution for integral information interoperability. The number of interfaces is minimal and proportionate to the number of system types. Besides that, the architecture offers flexibility in the sense that new systems can be added without having to adapt the other systems (by adding new interfaces).



a) systems of the same type, sharing an equal information structure, thus no interfaces

b) systems of different type, no information standardisation, resulting in 12 one-way interfaces

c) systems of different type, one common information standard, 4 two-way interfaces

Figure 2 — Basic architectures for information interoperability

Notice that an ‘arrow’ in figure 2 represents the translator on top of a system that transforms incoming (or outgoing) information from an external to an internal format (or vice versa).

Despite the advantages of the preferred basic architecture for interoperability (c), this solution in itself is not feasible for the whole ‘universe’ of systems, even if we restrict that to C4I systems within NATO. The ideal solution for NATO-wide interoperability would be a *single* standard for all information exchange between

[5] Be aware of the difference between systems and system *types*: systems of the same type have an equal architecture and use exactly the same type of information.

NATO C4I systems (see figure 3). However, a number of political, organisational, operational and technical issues (see further) make this solution unlikely to be ever achieved. Therefore, a subdivision in multiple exchange languages — each with a specific scope — will be necessary. Finding an optimal partition is the real challenge here. In support of this, the next paragraphs exploit the concept of “interoperability domains”.

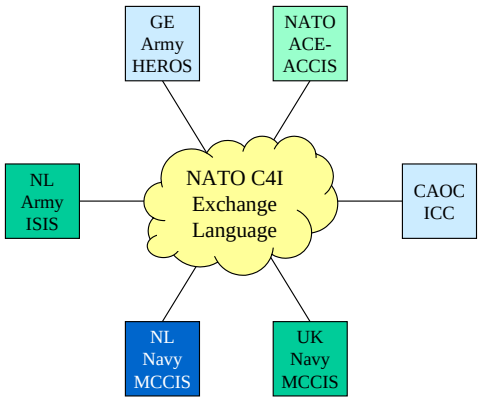
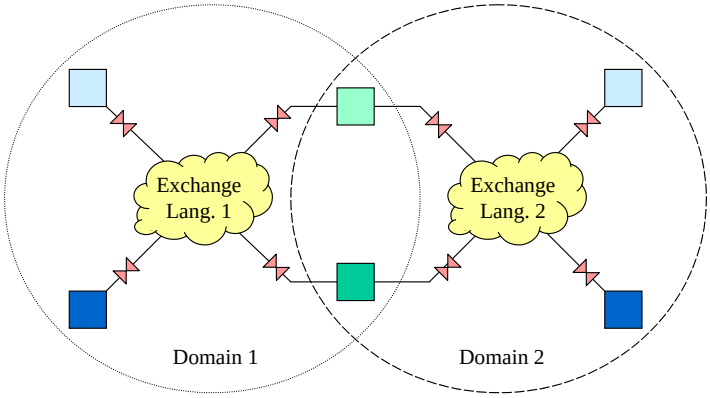


Figure 3 — The perfect (but impossible) solution

3.3. Interoperability domains

As we have seen, a number of systems can be made interoperable at information level by defining a common information standard (exchange language) which describes the information these systems want to share with each other. We define an *interoperability domain* as the total set of systems (or system types) that exchange information by means of the same exchange language. A system is said to be part of a domain when it interacts with other systems by making use of the domain’s exchange language. A system can belong to more than one domain in case it ‘talks’ *multiple* languages (see figure 4).



Example of two systems that are part of two interoperability domains, i.e. using two languages to exchange information with other systems.

Figure 4 — Multiple exchange languages

The size or *scope* of an interoperability domain determines *how many and what kinds of systems* belong to that domain. According to paragraph 2.3 the scope must be unambiguously defined. Because a domain represents an information standard, its scope can also be specified by means of the *kind of information* that is exchanged between systems within the domain. Two basic preconditions are valid when establishing the information scope:

1. Exchangeability.
   For the purpose of interoperability, only information which will (or can) be *exchanged* between systems is part of the standard. Thus, given a system, information that is not meant to be exchanged with other

systems, but just used internally, does *not* belong to a standardised exchange language. Notice that this also applies to (distributed) systems of the *same* type.

2. Commonality.
   The exchanged information is *shared* by the systems that require being interoperable, but not necessarily by *all* systems. Instead, the information part of the standard must be in common by *at least two* types of systems. This results in a range of possible scopes for the standard, limited by two 'extremes' (see figure 5a): the 'highest common denominator' (HCD, dark and shaded parts) versus the 'lowest common multiple' (LCM, only dark portion) of exchangeable information. Exactly which of the common information will be part of the standard depends on several factors (see further). Relevant in this is the ratio between HCD and LCM. For example, if the latter is only small compared to the former (see figure 5b), then a small part of the exchanged information is common for *all* systems.

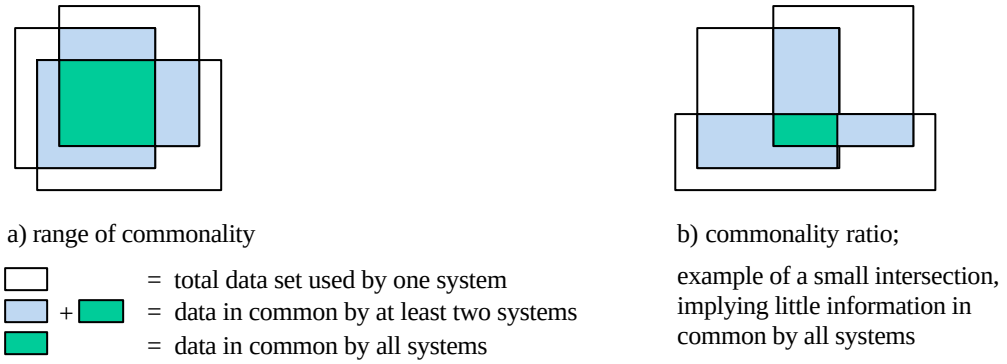


Figure 5 — Common information

Taking these two prerequisites as a starting point, the scope of an interoperability domain will primarily be based upon the operational requirements with respect to interoperability. Naturally, there will always be a drive for fitting all systems into a single domain (as in figure 3). But how large a domain eventually can become, depends on several things, such as:

- the number of different system types that must be included in the domain;
- the diversity in functionality of these system types;
- the amount of information to be exchanged between the systems;
- the degree of commonality of that information;
- the number, contents and change ability of 'legacy' information standards;
- the number of organisations using the systems within the domain;
- the number of additional parties involved in the standardisation process.

The more system types, functional diversity and exchangeable information, the more different types of information (subjects) have to be covered by the domain. This makes it harder to reach agreement on a common information standard. Also, which information is common for which systems affects the ease of agreement (e.g. because more bilateral than multilateral negotiation is necessary). Already existing standards may interfere as well, depending on their scope and whether their 'survival' or invariability is a precondition. And, of course, the number of actors in the standardisation effort, either representing one of the systems or involved for other reasons, highly influences the outcome of the standard. In general, when much information has to be harmonised between many players, the chances of success become smaller and the final result will cost more effort and time.

3.4. Domain structures

The previous analysis reveals that when a large number of systems needs to become interoperable, dealing with *multiple* domains is usually unavoidable. Of course, aiming at a single information standard is a good starting point, because it will result in the simplest (and cheapest and fastest) technical solution. But in practice this will often not be realistic. Therefore, making systems interoperable generally means *connecting different interoperability domains*.

We start with a relatively simple situation. We have three domains, for instance the C2 domain of the Army, Airforce and Navy of some nation. Due to reasons mentioned above, the national Ministry of Defence has chosen for this split-up. How can interoperability be achieved between C4I systems belonging to these different domains? There are two options (see figure 6): using direct links (a) or using an additional information standard (b). This corresponds to basic architectures ‘b’ and ‘c’ of paragraph 3.2, but one level higher. For figure 6b this results in a ‘second level’ exchange language. Regarding the usefulness of both options, the same reasoning (as in par. 3.2) applies over here: the second approach is better in case of more than two or three domains, because of the huge number of interfaces required otherwise. Only severe differences in information exchange requirements between pairs of domains, meaning little information in common for all three domains, makes the first option more suitable.

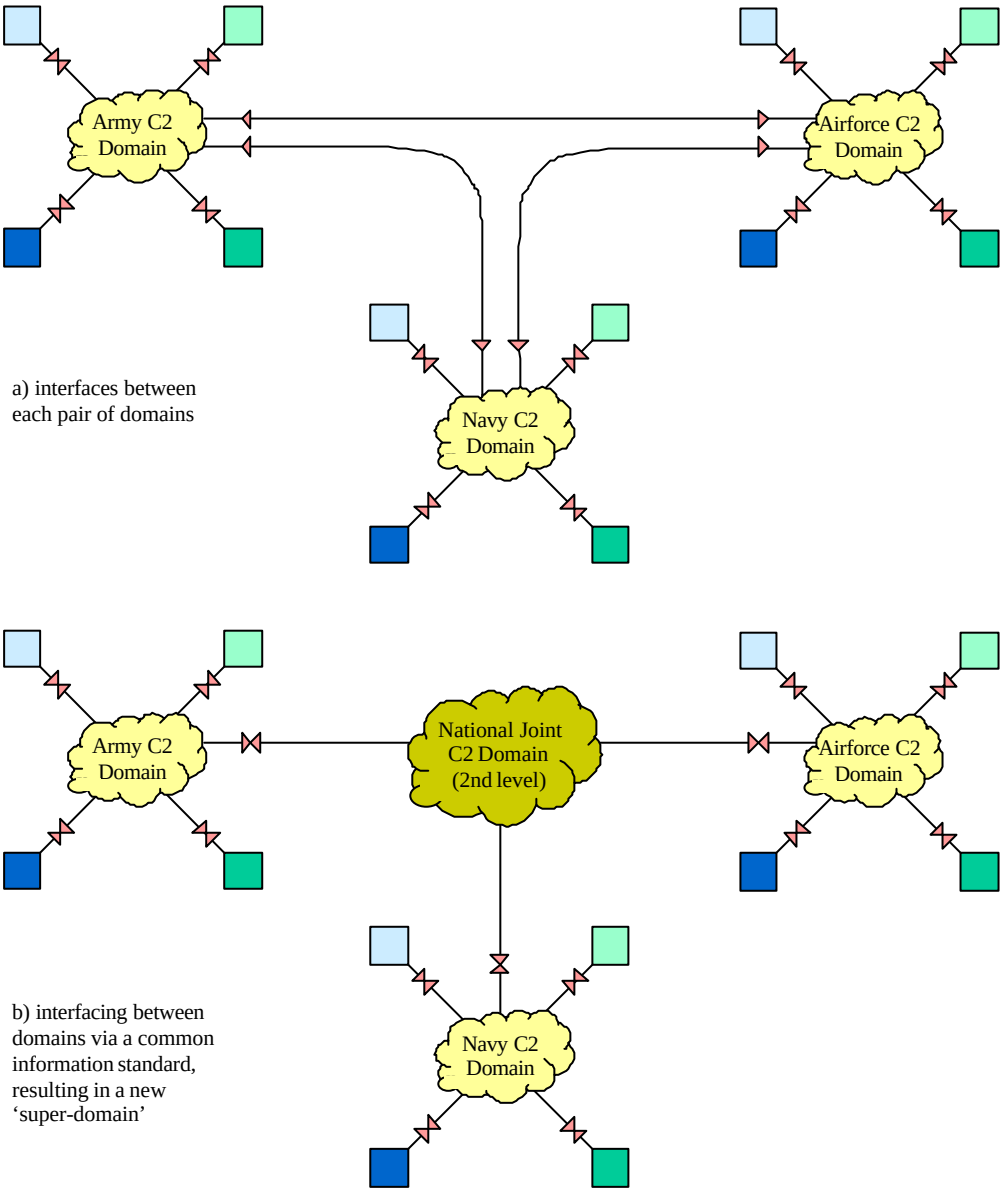


Figure 6 — Two options for interoperability between multiple domains

The connected domains in figure 6 are actually an abstraction of systems connected to multiple domains, as shown in figure 7. In option ‘a’ all systems are part of the three ‘force’ domains; in option ‘b’ each system belongs to one of the ‘force’ domains as well as to the ‘joint’ domain. Although both figures represent equal architectures, drawing systems as linked to only *one* domain (as in figure 6) may express the fact that this particular exchange language comes closest to (or is even equal to) the ‘natural’ language of these systems, i.e.

their internal data structure. It may also indicate that translation between languages takes place in sequence instead of directly. For instance, according to option ‘b’ the following transformations could take place when two systems exchange information:

- sequential: Army System X ↔ Army C2 Language ↔ Joint C2 Language ↔ Airforce C2 Language ↔ Airforce System Y
- directly: Army System X ↔ Joint C2 Language ↔ Airforce System Y

It is true the sequential translation seems less efficient, because it takes more steps. The advantage, however, is that less different types of translators are needed, due to reuse (e.g. the same “Army C2 / Joint C2” translator can be used by all Army systems!). This reduces costs, development time and system maintenance. Finally, hooking up systems to a single domain more or less imposes the environment in which the information standard of this domain is managed. Generally, the same organisations that are responsible for the directly linked systems are also involved in developing and managing the standard.
In conclusion, although it does not affect the architecture, drawing domains as linked to each other and systems as linked to a single domain, is often better. So a *hierarchy* of domains (with systems as ‘leafs’) is preferred over a ‘flat’ domain structure.

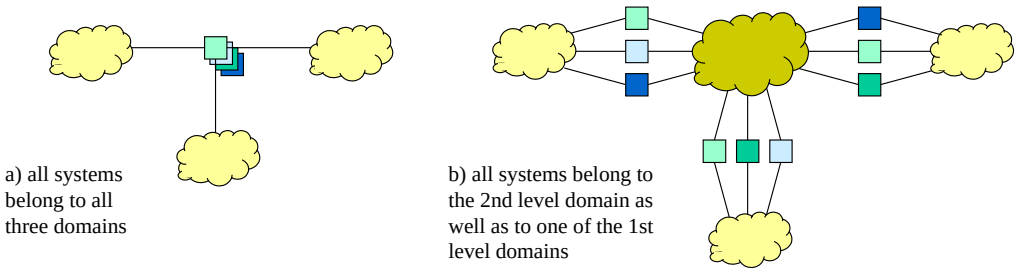


Figure 7 — Same two options as in figure 6, but drawn less abstractly

It is possible to connect systems to a *second* level interoperability domain. This indicates these systems use the exchange language underneath that domain to interact with each other. For them, this is the lowest-level domain to which they belong. In the example of figure 6b one could add a couple of joint C4I systems, as boxes directly linked to the National Joint C2 Domain. We call these systems ‘internal’ to that domain.

Here it may become rather difficult to fully understand the matter and realise the implications. We attempt to explain it as clearly as possible by exactly defining what it means when domains have different *levels* and when domains are *linked* to each other. Figure 8 could help to envisage things.

Domain levels. For $n \geq 2$, an n-th level interoperability domain connects two or more (n-1)-th level domains (as well as zero or more internal systems). This means that systems belonging to different (n-1)-th level domains (as well as internal systems) are able to exchange information by means of the common exchange language inherent to the *n-th* level domain. That these systems also use the (n-1)-th level language to *mutually* exchange information is irrelevant at this abstraction level.
With respect to the information *scope* of an n-th level domain, the same two preconditions apply as for first level domains (exchangeability and commonality, see par. 3.3). Hence, the ‘super-domain’ contains (1) only information that is also encompassed by the connected domains (exchanged by systems over there) and (2) only information in common by at least two of the connected domains. However, since an n-th level domain can also have ‘internal’ systems, these rules also apply to these systems (as if the domain was first level). This makes that, on top of the subset of information out of the ‘subdomains’, the n-th level domain may contain additional information types as well.

Domain linkage. If we look again at figure 6, we see two kinds of connections. Firstly, a link between two domains of the same level (6a). This implies (1) that a system of one domain can exchange information with a system of the other domain and (2) that this is done by ‘talking’ the language of either his own domain or the other domain. Secondly, a link between two domains of different levels (6b). This implies again (1) that a

system of one domain can exchange information with a system of the other domain, but in this case (2) that it must take place by using the language of the *highest-level* domain. For instance, an Army system and a Joint C4I system will interact via the National Joint C2 Language.
Connecting more than two domains in a *row* only makes sense when there are higher-level domains in between. Suppose we have a 'chain' of three domains of the same level. The systems at both ends *cannot* exchange information with each other, because their 'own' domains are not directly linked. (They would only be able to interact indirectly, if a system of the middle domain would forward the information.) Now imagine a chain of five domains with levels 1 - 2 - 3 - 2 - 1. Between systems of the first and of the third, fourth or fifth domain the 3rd level Esperanto is used, while between systems of the first and of the second domain the 2nd level language is spoken. Summarised, an interoperability domain can only act as 'intermediate' domain when it is defined at a higher level than the domains it must connect. Being of a higher level implies mandatory usage of its information standard for exchange.

To conclude this paragraph about domain structures, one should notice that an information environment that is large enough might contain numerous interoperability domains at several levels. Figure 8 illustrates how a large number of systems (or organisations) may be interconnected by means of several information standards at three different levels. Yet, to obtain such a structure is far from easy. The next paragraph explains the factors that play a role in finding an optimal domain structure and shows what real-life factors disturb the theory.

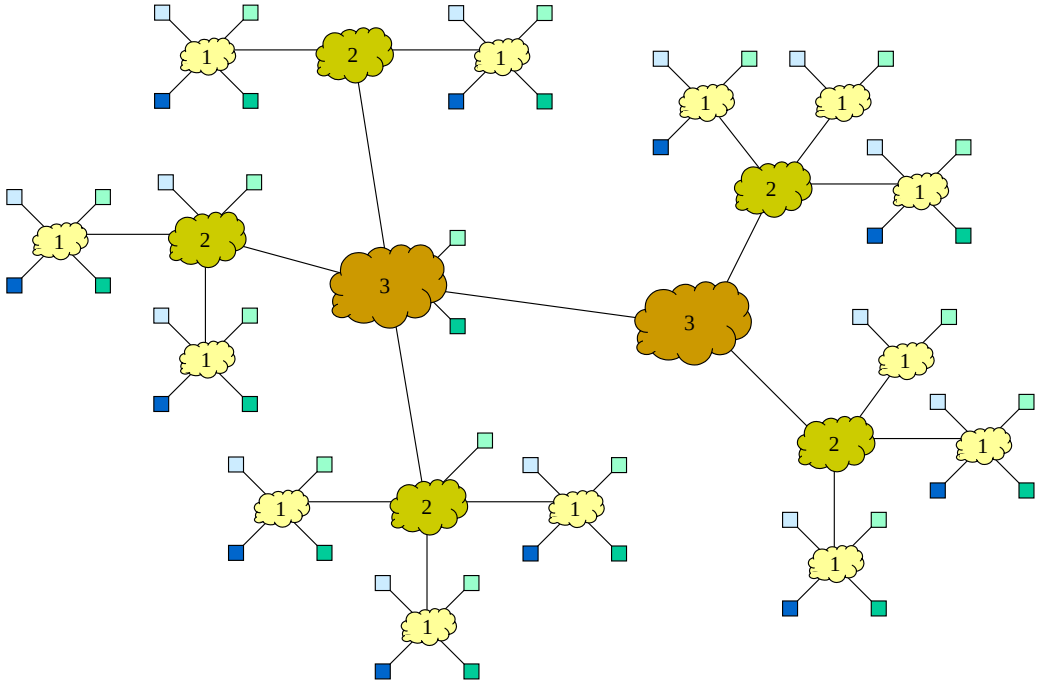


Figure 8 — Interoperability domains at different levels

3.5. Domain factors

Given a context for interoperability, what domains at what levels should be defined in order to obtain an optimal solution? By optimal we mean that the domain structure is such that a minimum standardisation effort will result in an efficiently working information exchange. This paragraph investigates several relevant factors and offers some guidelines.

We firstly consider the optimal size of a single domain. Scoping an interoperability domain affects a number of factors in positive or negative manner. The *larger* the scope of a domain:

- \- the more difficult it will be to reach common agreement on that domain;
- \- the more difficult it will be to maintain the information standard;
- \+ the more systems can interact via the same standard;
- \+ the fewer domains are needed to cover the whole problem area;

+ the less information transition between domains is required;
- the more overlap there will be with other domains;
- the higher the possibility that there is overlap with legacy domains;
- the more complex the information standard will be (to understand and implement);
+/- the more generic the information standard will be[6].

For *smaller* domains these factors are affected in opposite direction.

Rules for finding the optimal scope are hard to give. Fact is that the first factor, the exertion to agree on a standard, has the most impact on the final result. And, as we have seen in paragraph 3.3, this depends on several aspects such as the diversity of the information and the number of organisations involved. In general, an interoperability domain should have a scope of *maximum size*, under the condition that the information standard is still *manageable* with regard to overall approval, maintenance and implementation.

Another guideline is that the information inside a domain should always be related to a particular *subject* (that is, information should be of a specific *type*). This subject is often associated with certain functions and/or organisations as well. An information standard should not cover a variety of hardly related subjects; instead, it should have a *strong internal correlation* [10].
Similar to the subject, the information *owners*[7] should be correlated. The total set of information enclosed by a standard must be owned and exploited by a *limited* number of *related* organisations. This keeps the standard manageable. This is not the case when there are too many potential owners for a certain type of information and these owners are not organised such that only a few representatives are involved in the standardisation process.
So, subject and ownership usually determine the contents and scope of an information standard. A subject/ownership area can be very wide, but also rather specific. Example areas in our context are NATO C2, NATO Air C2, NATO Intelligence, NL Army C2 and NL Airforce Ground Operations.

We now consider the possible divisions of a complete context area into interoperability domains. Similar to the previous statements about separate domains, a domain *division* should also be based upon the *subject (type)* and *owner* of the information. This makes that each domain is an information standard for a *particular* subject and/or owner. Diversity in information types and owners exists along many dimensions, for example:

- topic or function of the information (e.g. personnel, materiel);
- purposes or activities for which specific information is used (e.g. viewing the current situation vs. supporting the planning process);
- the required quality of the information (e.g. real-time vs. non-real-time data);
- organisations that are formally responsible for specific information (e.g. nations, NATO).

Paragraph 4.3 contains a more extensive list of dimensions, aimed at the NATO C2 area.

This approach forms the basis for obtaining relatively autonomous (loosely coupled) domains. The functional relation (subject) and organisational influence (ownership) between domains should be weak and well defined [10]. In other words, domains should have *minimum overlap*. In the first place, this minimises the co-ordination between the developers of different interoperability domains. More synchronisation increases the complexity and the implementation costs of standards. Furthermore, minimum overlap can reduce the total number of domains and/or information translations. Also, it prevents multiple ‘paths’ — that is a choice between information standards — for exchanging information between two arbitrary systems. For instance

[6] Large information standards (data models) tend to have a generic nature, which means that specific information elements are represented in a non-specific manner. For instance, geographical demarcations such as no-fly zones, air corridors and unit sectors may be expressed as ‘control features’. This increases the flexibility of the standard, because similar new information elements can be included without (or with little) changes to the model. A serious drawback is the loss of semantics, which makes the standard more difficult to comprehend and less clearly scoped. Also, implementing applications on a generic data model is more complex.

[7] An information owner (in our context) is a (military) organisation responsible for certain information. The owner creates and manages that information. For example, a Spanish regiment that has observed some incident and reports it via a C4I system, will be the owner of the information representing this event. Although information may be copied and sent to other organisations, only the owner is allowed to modify or delete the ‘master copy’. Each piece of information has exactly *one* owner. These basic rules of ownership are generally accepted as foundation for distribution of information.

(see figure 9), enemy data between national Army C4I systems could be exchanged via a Land C2 standard or an Intelligence standard. Finally, with non-overlapping domains unambiguous information is avoided. Two or more standards that (partly) cover the same type of information rarely use exactly the same structure and format for that information. This causes conflicts — systems may understand certain information differently — and the “degree of standardisation” diminishes.

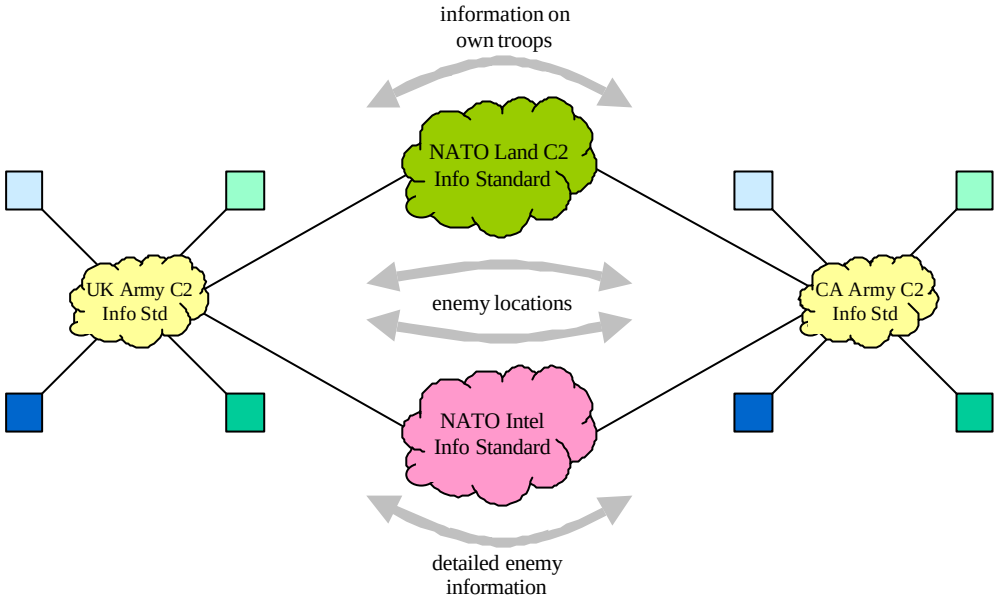


Figure 9 — Example of multiple paths due to overlapping information standards

The *level*, at which domains should be defined, depends on their role as common language for certain systems within the interoperability context. When an information area is subdivided into separate domains, systems that must be interoperable are usually grouped accordingly. A higher-level domain often origins from the requirement to enable systems of different domains to be interoperable as well. Such a domain will contain a subset of information in common by the lower-level domains. In case there are also systems that make use of the super-domain as their first level domain, it may contain additional information. This does imply, however, that the same criteria for scoping and subdividing domains (as described above) must be taken into account again.

As stated before, the domain structure should be founded on the type of information and/or ownership, hereby aiming for a minimum overlap between domains. However, this reflects an ideal situation. In reality, there are several factors that distress the ultimate domain structure, such as:

- sizeable subjects with many details and many complex relationships between data elements;
- a large number of separate information owners for a particular subject;
- already existing information standards that are not permitted to be substituted;
- involved organisations not able to agree on a logical subdivision;
- unfinished information standards due to long-lasting development.

How these (and possibly other) factors exactly will have an effect on the eventual domain structure, cannot be predicted; this depends on the situation. One should accept as a fact that the perfect solution is difficult to obtain and influenced by many factors. Chapter 4 will continue with a practical insight on this matter, aiming at the best reachable solution for NATO C2 interoperability.
Before that, the next paragraph briefly describes some technical consequences of having a domain structure.

3.6. Technical implications

Up to this point the discussion about interoperability has been concentrated on the aspects ‘information’ and ‘management’. But making systems part of *multiple* interoperability domains, one of our recommendations above, has some significant *technical* implications as well. Without going into detail, we give the most

important requirements to be fulfilled in order to make systems interoperable according to the approach explained before:

- Multiple translators are part of a system. Along the lines of figures 6, 7 and 8, each system needs to be able to 'talk' multiple exchange languages. The system has to transform its internal information structure into the exchange languages and vice versa. When information is being exchanged, one or more translators are run, depending on the domain to which the source or destination system belongs. If the concept of 'sequential' translation (see par. 3.4) is applied, there are also translators in use that convert between different exchange languages.
- The translation process between two information structures may be quite complex, especially if one is more generic than the other is. This is caused by, among other things, different sets of valid information element values ("attribute domains") without a one-to-one mapping. Loss of information may even occur due to severe differences. A translator is surely not a trivial component of a system.
- The difficulty of translation between two information standards highly depends on their *compatibility*. Standards that consist of similar information elements and structures are easier to convert into each other than standards that differ completely. In terms of data models this means the models should at least have an (almost) equal *core* structure, i.e. a common '*framework*'. This contains the fundamental and central parts of the data models. Thus, in order to 'ease' interoperability, exchange languages should be as compatible as possible, by basing them on the same framework.
  This requirement is not always feasible. A system's internal information structure is usually primarily influenced by system requirements, such as performance and application functionality, often resulting in structures unlike the framework. And for legacy systems it will be virtually impossible to change the internal database towards that framework structure.
- Besides compatible, exchange languages should also be *extendable*. Information standards tend to be never complete and finished; regular upgrades are inevitable. This is due to a long development time, the quickly changing military environment, evolving information requirements, new systems, etc. The information standard must therefore be *flexible*, meaning that new information types can be added without (or with minimally) changing the existing structure. This also ensures *backward compatibility*, the ability of systems to keep working with old versions of the standard. Flexibility can be achieved by specifying the basic information elements in the underlying data model (i.e. the framework) in a *generic* way.

4. NATO C2 interoperability

4.1. Introduction

The previous two chapters have revealed some theoretic aspects of information interoperability and information standards. In this chapter we will apply the theory to the NATO C2 environment and come up with a possible practical approach for achieving interoperability at information level. Guidelines are given for obtaining an optimal domain structure with usable information standards, enabling interoperable C4I systems.

We must stress the fact that the presented approach is an *example* of how it could work. Our view on the current NATO interoperability and standardisation developments and on NATO long-term policy is not complete and accurate enough, simply because it is very difficult to oversee this whole field. So, there may be relevant factors we have not taken into account. Nevertheless, being an example the method may still very well serve as a first step in the right direction.

4.2. Current NATO developments

Table 1 gives an overview of some of the most important C4I developments within NATO [4,8,9], varying from specifications to real systems. All developments are concerned with interoperability and standardisation in some way, if only to facilitate the (system) internal distribution of information. Per development we cite the scope of the information to be exchanged and how the corresponding "information standard" is called (and/or looks like). Notice that we only consider the information *contents* here; quality aspects like security and actuality of information are not included in the scope.

System or Specification	Meaning	Information Scope	Information Standard — Name and/or Appearance
ADatP-3	Allied Data Publication 3	C2	ADatP-3 formatted messages
NCDM	NATO Corporate Data Model	C2	NATO Reference Model + functional views
Bi-SC AIS	Bi- Strategic Command Automated Information System	C2	(none yet, to be integration of ACE-ACCIS and MCCIS)
ACE-ACCIS[8]	Allied Command Europe - Automated Command and Control Information System	C2 (SC Europe)	(none yet, to be integration of i.a. BICES, JOIS, LOGFAS[9])
AIntP-3	Allied Intelligence Publication 3	C2 Intel	AIntP-3 data model
BICES/BICC	Battlefield Info. Collection and Exploitation System / BICES Initial Core Capability	C2 Intel (Land)	ACE Intelligence Data Model
PAIS	Prototype ACE Intelligence System	C2 Intel	PAIS database
JOIIS	Joint Operational Intelligence Info. System	C2 Ops/Intel	JOIIS database
ADAMS[9]	Allied Deployment and Movement System	C2 Logistics	Logistic Database
ACROSS[9]	ACE Resource Optimisation Software Sys.	C2 Logistics	Logistic Database
ATCCIS	Army Tactical C2 Interoperability Spec.	C2 Land	Land C2 Info. Exchange DM
MIP	Multilateral Interoperability Programme	C2 Land	Land C2 Info. Exchange DM
ACCS	Air Command and Control System	C2 Air	ACCS Conceptual DM
ICC	Initial CAOC Capability	C2 Air	ICC database
MCCIS	Maritime Command and Control Information System	C2 Sea (SC Atlantic)	MCCIS databases + ADatP-3 and “OTH-Gold” form. msg.’s
Link 11/16/22	Tactical Data Links 11/16/22	C2 Air/Sea	Link 11/16/22 form. messages

Table 1 — Some relevant C4I developments within NATO context

Most information standards mentioned here cover both the operational and tactical levels (and sometimes more), although the exact scope with regard to the operation level is unclear for the majority. Some standards seem to support particularly higher-level information. For example, ACE-ACCIS supports the consultative process between senior commanders and agencies. Other standards mainly comprise low-level information, such as the Tactical Data Links that incorporate things like tracks and engagement orders. Nevertheless, in all cases at least some (but often much) ‘generic’ information at operational/tactical level is included in the standard. The status of an Army Battalion inside the operation area, for instance, is surely of interest for many users and will be supported by many C4I systems. This fact, together with the similarities in information scope between much of the systems and specifications (as shown the table’s third column), causes considerable overlap among the existing (or emerging) information standards. As explained in paragraph 3.5, this situation is undesirable. The question now is: how can we improve this? In the next paragraph we first outline a potential ideal collection of C2 information standards for NATO, followed by a paragraph that suggests a practical way to apply this, while taking into account the current developments.

4.3. Optimal NATO domain structure

Taking NATO C2 as our context for interoperability and assuming no standards exist yet (or all existing ones can be replaced), what domains at what levels should be defined in order to obtain an optimal solution? Using the guidelines of paragraph 3.5 we attempt to find the best possible subdivision of the NATO C2 area into separate interoperability domains (information standards). Various kinds of partitions could be employed; each determines how domains are created based upon a *specific* categorisation of information. Eleven possible dimensions for subdivision have been identified:

a. functional area (data about C2, politics, administration, law, sensor & weapon systems, etc.);
b. operation level (strategic, operational, tactical or technical data);
c. command level (e.g. data for Division/Brigade vs. Battalion and lower);
d. operation type (e.g. data required for Article-V or Crisis Response Operations);
e. operational context (data for an operation, exercise, test, simulation, etc.);

[8] Due to the Bi-SC AIS developments, ACE-ACCIS might never become a sole system, because it may have been integrated before it is finished. In this article we still assume it is a separate system under development.
[9] ADAMS and ACROSS are part of the Logistics Functional Area Services (LOGFAS), a part of ACE-ACCIS. Both systems use the same database, the Logistic Database (LOGBASE).

f. region (e.g. data used in the regions Europe and Atlantic);
g. responsibility, i.e. nations and multinational organisations (NL, AFNORTH, etc.);
h. operational theatre, matching the military forces (Land, Air or Sea data);
i. subfunctional area (for C2: data about Intelligence, Operations, Logistics, NBC, etc.);
j. time and dynamics (current situation, plans, historical events, encyclopaedic data, etc.);
k. function regarding interoperability (operational info, security info, distribution info, etc.).

Of course, not all of these subdivisions are suitable. We now explain which are feasible as a basis for a NATO C2 interoperability domain structure.

(a) Due to our scope, i.e. NATO C2, the functional area will be pure C2. Though C2 information is related with other kinds of information, for instance of administrative nature (personnel, economics, finances, etc.), we still think the boundary is quite sharp. Looking at the global information requirements for primary operational/military tasks (e.g. obtaining situational awareness of the battle space), there is *no significant* overlap with other functional areas. In addition, including other functional areas will result in a C2 domain too big to handle, above all because of the huge amount of organisations and systems that will be involved. So, C2 interoperability domains should exist apart from other functional domains[10]; this article does not address the latter.

(b) The operation level will be limited to operational and tactical, because the levels above and below are not the primary context for Command and Control. As already said, most C4I systems are intended for the operational and tactical levels. Making distinction between the two levels is not relevant, because there is severe overlap with respect to the type of information involved and because several C4I systems cover both levels anyway[11].

(c) Although the command level is linked to the operation level, in principle our scope reaches from Corps down to the smallest unit sizes. Modern operations with multinational and flexible forces require potential C2 information exchange with any unit regardless of its size. Dividing information according to command level is therefore not useful.

(d) Considering the present diversity in operation types, we may presume that more new types will probably emerge in future and that most kinds of operational information are applicable for most types. Therefore, it is of no use to define a separate interoperability domain per operation type.

(e) Our operational context includes operations and exercises. The information needed in both cases is equal ("train as you fight"), so domain partitioning is undesirable. Other settings, for instance a military simulation environment, fall outside our scope.

(f) Regions are no base for domains, because the type of information required for C2 hardly depends on a region. Operations should be conducted in the same manner everywhere.

(g) Mapping interoperability domains on NATO nations and NATO organisations, possible owners of specific information, may be useful from the point of view of minimising the number of parties concerned with the standardisation process. Also, the type of information used within those often dissimilar 'worlds' could differ considerably. In the case of *nations* a domain separation is valid: national rules and feelings (politics) make all-enclosing information standards on C2 virtually impossible. Hence, nations could act as boundaries for interoperability domains. For *NATO organisations*, on the other hand, it is different. The organisational structure of NATO is big and complex; many associations exist and many organisations have common responsibilities. Thus splitting up information according to the NATO structure is not obvious. Besides that, NATO itself endeavours enterprise-wide systems and standards.

(h) Historically, military forces have different doctrines and thus different information requirements derived from that. Subdivision of interoperability domains founded on Army, Navy and Airforce seems convenient.

(i) Subfunctional areas of C2, such as Intelligence and Logistics, cover diverse subjects. This implies much information of different type, especially regarding the details. A division in domains based upon subfunctional areas seems logical. However, there are two problems. Firstly, the areas also *share* much information, namely

[10] Nevertheless, there are tendencies which strive for integration of all functional areas around C2. Examples are: the inclusion of 'in-depth' sensor and weapon data according to the philosophy of Network Centric Warfare; the (Dutch) effort to integrate the national Defence 'green' and 'white' domains (C2 and administrative/management data respectively).

[11] What *is* relevant is this context is the 'quality' of information, such as its timeliness and accuracy. Tactical information usually requires to be more real-time and more precise than operational information. However, this article only concerns about information *types*, not qualities. On the other hand, quality aspects could have been used as dimensions to divide information. For example, one could distinguish between real-time, near real-time and non-real-time data. We did not consider such categorisations, because they seem to be not very useful.

'generic' C2 information (about units, materiel, locations, plans, etc.). So, domains based upon them would have quite some overlap. Even worse, "Generic C2" is a subfunctional area in itself, because there are quite some C4I systems that mainly provide a global situational picture, leaving out details about logistics, etc. Secondly, the subfunctional area partitioning and the Land-Sea-Air partitioning (see item 'h') are diametrically opposed. Each force makes use of subfunctional areas, which could imply severe overlap between force-related and subfunctional information. Despite of these two problems, we still think it is useful to have interoperability domains projected on subfunctional areas of C2 (in parallell with the force-based domains). For certain subfunctional areas much of the information is very specific and independent of the other areas; a relatively small part is generic. This argues in favour of separation. Another reason is that such information tends to be applicable for all forces. This is proven by the fact that some systems supporting a certain subfunctional area are indeed being built in a joint effort. If the force domains are limited to generic information, then the overlap with the subfunctional area domains is minimised. Summarised, distinct domains for C2 subfunctional areas are feasible and the mutual overlap is taken for granted. Logistics, Intelligence and Generic C2 appear to be the most obvious areas to create separate domains for. For now, we limit ourselves to these three areas, but some other areas (e.g. Personnel) may be a candidate as well. The remaining subfunctional areas should be included in the selected ones (e.g. Operations, NBC and CIMIC could be part of Generic C2, provided they encompass only little dedicated information).

(j) The subdivision of information in accordance with its timely and dynamical nature is particularly interesting for end users (and may result in a corresponding application set). In line with the C2 decision process they usually work in a way information about (for instance) current situation and planning is employed *separatly* (e.g. in different overlays). However, these kinds of information sets typically have a substantial part in common, so they are not applicable as domains.

(k) Finally, in the perspective of interoperability not only operational data is exchanged, but also all kinds of supporting data. This could be security information describing who owns a C2 data item and whether it is classified, or distribution information about who is subscribed to what C2 data. Such supporting information inherently belongs to the interoperability domain — it must be standardised as well — even though most of it is normally not contained in the operational information standard, but defined separately. Although essential for interoperability, we do not consider such information any further in this article.

In conclusion, NATO C2 information at operational and tactical level should be subdivided along the following dimensions:

1. owners: nations + NATO as a whole;
2. themes: Land, Air and Sea;
3. subfunctional areas: Generic C2, Intelligence and Logistics (for now).

This results in a possible interoperability domain structure as displayed in figure 10. Some explanation:

(i) The national domains are of national concern. Here we have taken the (probable future) Dutch situation as an example. Keep in mind that the top of the picture should have been 'multiplied', because (in principle) all NATO nations will be connected to the NATO domains; to keep the overview, only one nation is drawn.

(ii) The subfunctional area (NATO) Generic C2 has been integrated with the three themes (forces). This means that the NATO Land, Air and Sea domains contain *generic* Land/Air/Sea C2 information. (Notice that the NL domains are *not* generic, but include subfunctional areas as well.)

(iii) An additional "NATO Joint C2" domain covers what the three thematic domains have in common and require to exchange. This domain is limited to *generic* C2 too.

(iv) Besides the Joint C2 domain, the NATO Intelligence and Logistics domains are *joint*[12] as well. The three domains are thus supposed to cover the greater part of the whole area of joint NATO C2 information; possible additional subfunctional areas (see item 'i') may complete this.

(v) The systems (little boxes), some of which virtual (dotted), are examples that serve to explain the usage of the interoperability domains. Figure 10 shows that systems of any kind can potentially exchange information with each other by using one or more exchange languages[13].

(vi) Notice that most systems are directly linked to just one domain, in line with the guidelines about domain hierarchy in paragraph 3.4. There is one example system, ACE ACCIS, for which this is not possible, because that system is part of domains not connected to each other. This may illustrate the (unmanageable?, undesirable?) diversity in functionality of this (future) system.

[12] Evidently, all NATO interoperability domains contain information of *combined* nature.

[13] Provided that at least a network connection is available as well (interoperability layers 1 and 2, see par. 2.4).

(vii) The bar on the right side of the picture denotes the level at which the domains are defined, relative to the national force domains that got level 1. The five NATO force and subfunctional area domains are second level, because they are meant to connect the corresponding national information areas (which are *virtual* domains in this context, because it is of national concern how to organise the own information). These domains do *not* have different levels with regard to each other, since they cover (more or less) separate information areas. The NATO Joint C2 domain, on the other hand, is of the third level, because this domain acts as the common language for three other (second level) domains.

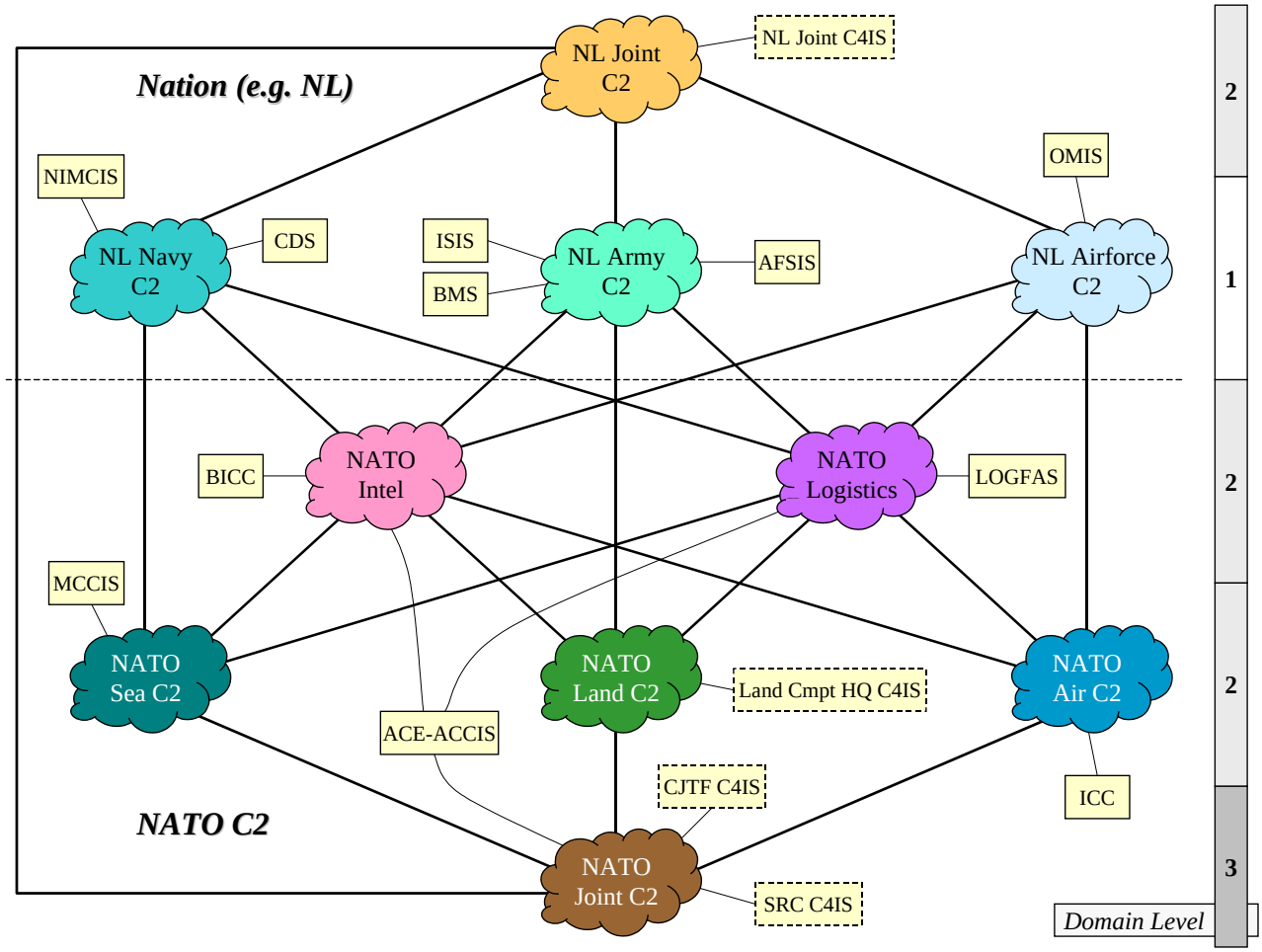


Figure 10 — Possible optimal interoperability domain structure for NATO C2

4.4. The optimal solution in practice

In our opinion, NATO should aim for a formation of information standards in line with what is shown in the previous paragraph. The final optimal NATO C2 interoperability domain structure may appear to look somewhat different, for instance due to additional subfunctional areas, but we think the general idea is feasible in reality.

How can this optimal long-term objective be achieved? The current developments (see table 1) are a ‘fact of life’ and can probably not be altered too much. Instead, the policy should be to ‘divert’ these programmes such that they will grow towards the intended structure. This implies certain developments will have to integrate (see further).

Recent developments within NATO reveal that NATO is indeed working in a similar direction. This is especially valid for the “Bi-SC AIS” programme [4,5,6], that aims to converge ACE and ACLANT systems. This must result in a single system consisting of a “Core Capability” (common services) and several “Functional Area Services” (specific applications). It illustrates that NATO has the intention to integrate

things. However, it appears the emphasis currently lies on the integration of *systems*, not information in particular. Moreover, NATO aims at a *single* all-enclosing system, a 'Utopia' which we think is unreachable. It is technically complex, requires consensus among many players and is unmanageable (while operating) due to its magnitude.
Instead on systems, it might be better and easier for NATO to focus on *interoperability standards*, including information standards. Enabling many different systems to exchange and understand the same information is already hard enough. Integrating these systems is even more difficult and, in fact, unnecessary. The main goal behind this integration into the Bi-SC AIS is to gain *interoperable* NATO C4I systems[14], for which an interoperability standard is sufficient. (NATO comes closer to this approach in another of its integration efforts, namely the planned migration of the Tactical Data Links [7]).

Suppose the approach described above appears to be feasible — also politically — in reality. Then the information standards mentioned in table 1 might be directed towards figure 10 in the following way:

- The MCCIS databases and the ADatP-3/OTH-Gold formatted messages for MCCIS are the basis for a Maritime Data Model, which covers the NATO Sea C2 domain.
- Given its maturity, the Land C2 Information Exchange Data Model (ATCCIS) is the most likely candidate for the NATO Land C2 domain.
- The ACCS Conceptual Data Model should develop into the NATO Air C2 information standard (possibly by also integrating the ICC database).
- The Tactical Data Link messages should be integrated with the ACCS data model and the MCCIS formatted messages.
- The AIntP-3 and ACE Intelligence data models (possibly together with the PAIS/JOIIS databases) must result in a single NATO Intel information standard.
- The Logistic Database forms the obvious basis for the NATO Logistics domain.
- Finally, the NATO Reference Model (part of the NCDM) can be the basic framework (see par. 3.6) for all NATO information standards. It induces a core structure in order to ensure compatible and flexible data models.

Important is that these developments are closely monitored and co-ordinated from a central point of view. Their scope should be clearly specified and communicated, so that no overlap or blind spots can occur and the aimed interoperability domains are indeed obtained.

In general, if NATO should decide to follow the above-mentioned approach, it not only needs to define a policy that defines the global objectives (interoperability by means of information standards, etc.), but also a frame of reference on how these objectives should be realised. This includes, among other things, the intended overall interoperability domain structure (which can be considered as a NATO information exchange 'meta standard'), preconditions for developing the independent information standards (such as the data model framework), preconditions for other aspects of an interoperability standard (e.g. communications and security, see par. 2.4) and a migration path for 'redirection' of the current related C4I developments. This is what we normally call an "information architecture" [10], being a vision on how the NATO information requirements should be accomplished, hereby fulfilling prerequisites with respect to structure, components, flexibility, etc. Essential in this is the role of a central high-level NATO body, for instance the NATO Data Administration Organisation (NDAO), which must co-ordinate the different domain developments and make sure the NATO information architecture is indeed adopted.

One of the major problems here is, and will always be, the existence of (legacy) standards and systems already in use. But this is not different from the Bi-SC AIS approach, where this problem occurs just the same. Prescribing NATO-wide interoperability standards may even be less problematic in our approach, because existing systems can remain and would 'only' need an interface on top.

[14] Some other general objectives of system integration are re-use of software and common user interfaces. It seems in NATO context these goals are less important than achieving interoperability (or not valid at all).

Another problem has to do with the scope of the proposed information standards. They contain much information and involve many players and systems. This implies laborious consensus and a long development time. But again, a similar issue must be solved for Bi-SC AIS as well.

If people can be convinced of the importance of information standardisation for the purpose of interoperability and if the quality of the approach presented in this paper is proven, then the problems will be overcome and an interoperability domain structure such as introduced here, may become reality.

5. Conclusions

Below, the conclusions drawn in this article are summarised.

- Current and future military operations within NATO require extensive co-operation (information exchange) between participating military units, organisations and nations and, as a consequence, interoperability between their supporting C4I systems.
- The interoperability concept has many interpretations, but a commonly used form is information interoperability, because it offers optimal connectivity between systems, while preserving maximum independence. Information interoperability is defined as the ability of systems to automatically exchange and interpret information that is common to those systems.
- In the (mostly occurring) case that more than a few systems have to exchange information, standardisation of the ‘interface’ is a key factor to achieve information interoperability. Otherwise, dedicated interfaces are needed between every pair of interconnected systems, leading to an exponential grow of the number of interfaces required.
- Information interoperability requires the standardisation of several aspects. One is the exchange language or information standard, in our view the most challenging and important, but also most difficult, aspect. The difficulties in defining such an information standard are a consequence of technical, operational, organisational and political matters.
- One single information standard for all information exchange between systems is a solution that is unlikely to be ever achieved, even if we restrict the ‘universe’ to NATO C4I systems. Therefore, a subdivision in multiple information standards — each with a specific scope — will be necessary. The set of systems that exchange information by means of the same ‘scoped’ information standard is called an interoperability domain.
- We assume the following preconditions for information standards that support interoperability. Firstly, only information which will (or can) be exchanged is part of the standard. Secondly, the information within the standard is used (and exchanged) by more than one system within the domain, but not necessarily by all systems.
- For interoperability domains the same argument with respect to standardisation is valid as for individual systems. When the number of domains increases, it is better to standardise the exchange between them by defining an information standard (domain) at a higher level. This line of reasoning may continue for a number of hierarchic levels.
- The scope of an interoperability domain (information standard) can depend on several factors. In general, a domain should be of maximum size and should have minimum overlap with other domains, under the condition that the information standard is still manageable with regard to overall approval, maintenance and implementation. This requires a co-ordinated interoperability domain division strategy, as part of an overall information architecture.
- The division into interoperability domains should be based upon the subject or type of the information. Diversity in information types may exist along many dimensions. For NATO C2, the following dimensions seem the most preferable ones: owners (nations, NATO), themes (Land, Air, Sea) and subfunctional areas (Generic C2, Intelligence, Logistics). Based on these dimensions, we have defined a possible optimal interoperability domain structure for NATO C2.

- In reality, several factors distress this optimal domain structure. Already existing information standards, lack of (political) agreement on a logical structure, and unfinished information standards (due to long-lasting development) are the most obvious ones. Looking at the various C4I (information standardisation) developments within NATO, compared to the optimal interoperability domain structure, leads to the following conclusions. The objective to gain a single integrated system ("Bi-SC AIS") may be too ambitious; in order to obtain NATO-wide interoperability an interoperability standard will be sufficient (and hard enough to achieve). The C4I developments could be 'directed' towards this optimal domain structure.
- Concerning interoperability within the NATO C2 area in general, we conclude that information standards and their scoping and subdivision are still very much underestimated aspects of interoperability and that more attention for these aspects is needed in the future.

References

1. NATO Interoperability Planning Document (NIPD)
2. Army Tactical Command and Control Interoperability Specifications (ATCCIS), Working Papers 14-X series (2000)
3. Allied Data Publication 3 (ADatP-3), STANAG 5500
4. "NATO signals an all change", Jane's International Defense Review (February 2000)
5. Bi-SC AIS Implementation Strategy (March 2000)
6. Bi-SC NATO Common Operational Picture - Operational Requirements (July 2000)
7. Bi-SC Data Link Migration Strategy (December 2000)
8. (Mobile) C2 in Crisis Management Operations, NL Ministry of Defence (October 1998, in Dutch)
9. Policy for Operational Information Supply, NL Ministry of Defence (April 2001, concept, in Dutch)
10. Information Architecture, Van der Sanden / Sturm (2000, in Dutch)

UML Modeling Rules for Interoperable Architecture Artifacts

Michel Lizotte
Defence Research Establishment Valcartier (DREV)
2459 Blvd. Pie XI (North), Val-Bélair, Québec, Canada G3J 1X5
e-mail: Michel.Lizotte@drev.dnd.ca

1. Summary

In recent years, time required to develop software interoperability solutions has become a key factor in the success of coalition operations. This paper introduces a set of Modeling Rules refining and restricting the Unified Modeling Language (UML) usage to a minimal set of models. Such an approach reduces precious time of software architects to get the right information and to understand the real issues of the problem. It enables navigation into software models from high-level artifacts down to the required level of detail using an object and behaviour perspective.

2. Introduction

Poor software interoperability is a deficiency well recognized by nations involved in military coalition operations. In order to prevent problems and improve software interoperability, the elaboration of coalition interoperability solutions should ideally begin by exchanging software models between architects, from high-level artifacts eventually down to the required level of detail. The Unified Modeling Language[1] (UML) is an appropriate candidate to achieve this requirement. In addition to its intrinsic qualities, UML is a great achievement since it provides object software developers a common basis for exchanging software models. But UML still requires refinement and tailoring to reach this goal.

This paper introduces a set of Modeling Rules refining and restricting UML usage to a minimal set of models. These rules were developed throughout practical experience in developing object-oriented software. Their goal is to produce effective but maintainable models facilitating system evolution and maintenance, while minimising the impact of development personnel changes. In other words, one objective is to reduce useless effort on the part of developers to maintain huge volumes of documentation that are rarely used and often not up to date. Often, such documentation is not kept up to date because of its cost. The second objective is to force developers to provide a useful minimal set of models and keep them up to date.

The next section of this paper, Section 3, outlines the problem being addressed. It is followed by a presentation of the context within which the approach has been developed. Section 5 provides an overview of the approach. Some top-level and diagram-level Modeling Rules are then presented and summarised. In this paper, the expression “Modeling Rules” does not mean ‘how to model’ but rather ‘how to link modeling elements and manage systems complexity’.

3. Problematic

The rationale behind the approach is illustrated through a cartoon. The story presented below is about a Canadian architect who has the mission to make interoperability happen between a Canadian, a French and a US system.

[1] UML is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system ([UML 00] foreword).

Paper presented at the RTO IST Symposium on “Information Management Challenges in Achieving Coalition Interoperability”, held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

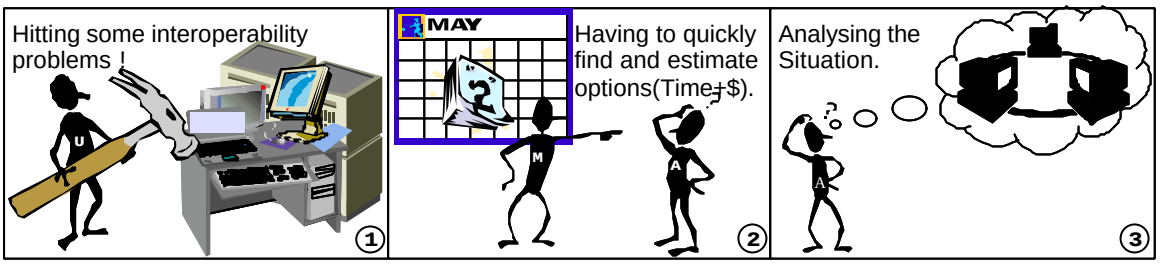


Figure 1 An interoperability problem

1. A Canadian user is experiencing interoperability problems. He cannot achieve his critical mission within the expected time frame because of those problems. He reports the problem to higher instance managers who agree to make it a priority requirement to be met before the next similar mission, in 6 months. The U on the bean character stands for User.
2. In order to address this deficiency, a manager tasks a Canadian architect to propose options (e.g. time, effort and money estimates) for solving the problem. The M on the bean characters stands for Manager while the A stands for Architect.
3. The Canadian architect develops an overall course of action to reach the objective: (1) study the current situation; (2) diagnosis the problem; (3) analyse options; and (4) implement an option.

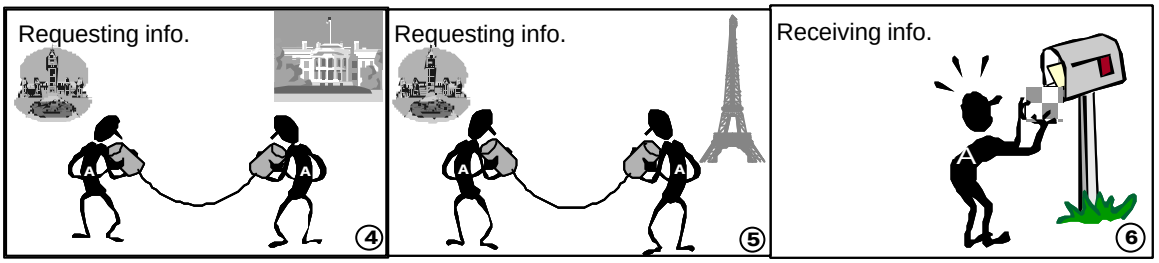


Figure 2 Getting information about the systems

4. His first action is to communicate with a US architect in order to know more about the French system. He expresses his need for more information to fill information gaps about their system.
5. He has the same kind of communication with his French colleague.
6. The Canadian architect receives answers that are different from those he was expecting. His request, which was clear from his point of view and also clear from the point of view of his coalition colleagues, had a different meaning for the three architects.

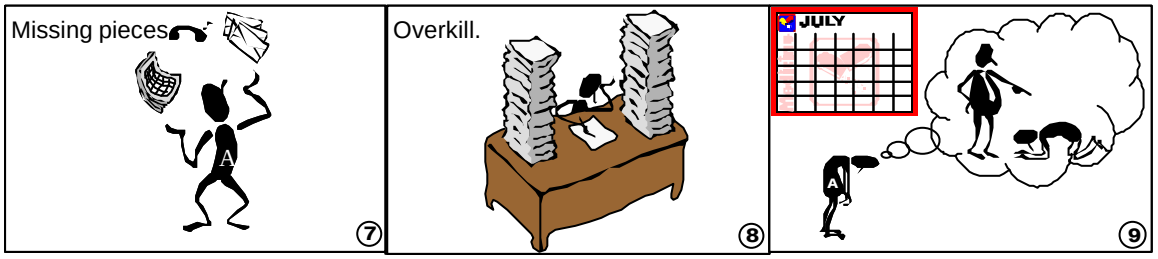


Figure 3 Loosing time

7. The US reply provides some information but does not include information sought by the Canadian architect.
8. The French architect provides the answers but hidden within a huge pile of documents.
9. Three months have passed and the Canadian architect is far from meeting his manager's request. An almost useless communication cycle has taken place. Even with good intentions from all parties, each a specialist in software architecture having different semantics for the same words (e.g. wrong French to English translation, different usage of the same word, etc.) have led to a loss of a very precious resource for everybody: time. The problematic of understanding foreign systems has

similarities with personnel changes on software development projects. The objective is to reduce time required to introduce new people to the relevant part of the software.

4. Historical Background

The rules being presented here are elaborated in an R&D context where the application domain is military intelligence. The project is called All-Source Intelligence Producer (ASIP). Its main objective is to provide validated software models bringing new technologies and innovative functionality to intelligence users. The approach is to elaborate innovative software models, to select emerging technologies, to build a software testbed and to conduct "realistic" military exercises in order to assess the models and measure the impact of the innovation they bring to the user. The ASIP prototype is currently being funded through major Canadian R&D efforts such as The Land Intelligence and Electronic Warfare Automation (LIEWA) project. The first LIEWA major experiment was conducted in August 2000 involving 20 military users [LIEWA 01].

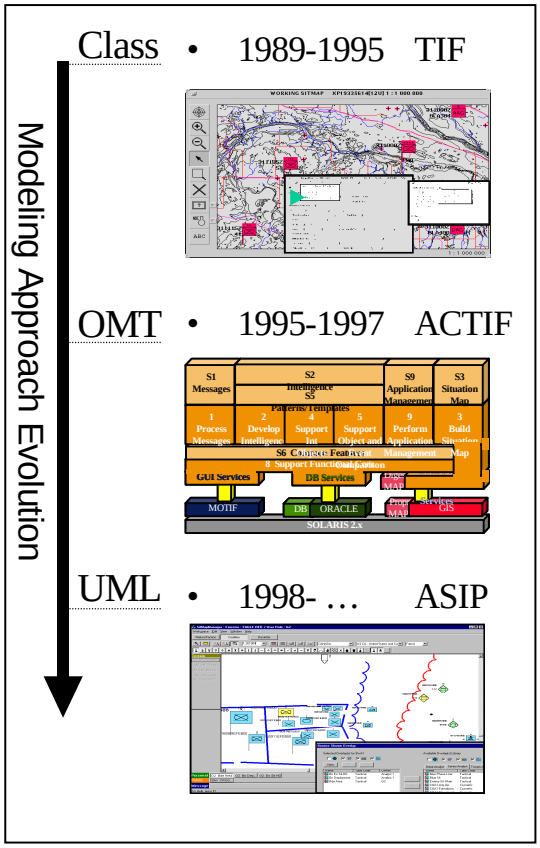


Figure 4 Experience of object-oriented modeling

The ASIP project has a historical background that has significantly influenced the above-stated goal of the Modeling Rules being introduced in this presentation. From 1989 to 1995 an important object-oriented prototyping and trial effort called the Tactical Information Fusion (TIF) project took place at DREV [TIF 92] [NL 95]. Following this, a two-year software architecture effort using Object Modeling Technique (OMT) [Rumbaugh 91] was conducted [ACTIF 97]. The aim was to elaborate a software architecture based on the lessons learned during the TIF project. ASIP started at a later date, using this target architecture, which includes around 350 classes. ASIP 1.2 was used for the first LIEWA major experiment. It implements a subset (around 15 %) of the target architecture.

ASIP 1.2 is an evolutionary prototype offering advanced features to draw military situations on an electronic situation map. Intrasystem communication between the ASIP server objects running on Solaris and the ASIP clients running on Windows NT are achieved through CORBA technology. Among the advanced features, ASIP 1.2 provides a collaborative work capability, implemented through the CORBA event service, allowing users to share live overlays. With its intersystem CORBA IDL interface, ASIP 1.2 also opens a door to interoperability with other systems.

These developments have emphasized the importance of a good balance between a very modest and a full-blown modeling effort:

- The TIF project used basic class models with a very informal modeling approach. This modelling effort was insufficient and led to difficult integration of new developers on the project and little advance in know-how for future projects.
- The ACTIF project had a first set of Modeling Rules based on OMT, MIL-STD-498 [MIL-498 96] and DMR Productivité Plus ^{MC}[DMR 90]. The approach was followed for the Architecture phase but not for the following Design and Construction phases since it was too complex to have a good cost-benefit ratio.
- The ASIP project is now using a second version of Modeling Rules taking into consideration these lessons i.e. trying to achieve a good balance. The approach is now based on UML and looks promising.

5. Overview

The approach consists in regulating architecture artifacts using UML as the baseline. The approach is based on the following key principles:

- UML, an open standard well accepted by the industry, is used. In particular, UML includes the XML - Extended Mark-up Language - Model Interchange (XMI) specification aimed at supporting software model exchange between the tools of different vendors.
- Three views of UML are regulated through the proposed Modeling Rules:
  - The Static View presents the object perspective of the software (classes and their packaging) at different levels of detail.
  - The Use Case[2] View presents the high-level behaviour perspective. It targets a software developer audience. However, most Use Cases remain readable by a non-specialist audience. Only very technical ones are more laborious e.g. items related to the application framework.
  - The Interaction View presents the detailed level of the behaviour perspective. It identifies class operations participating in the realisation of Use Cases.

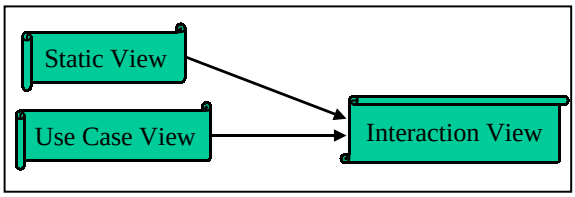


Figure 5 Views relationships

- The Modeling Rules link model elements of the different views forcing coherence and enabling interactive examination of the software architecture. Although, they do not ensure that architects will design better systems.

The approach, illustrated in Figure 6, defines two perspectives for examining a software system: object and behaviour[3]. Our practical experience has emphasized the importance to distinguish and keep both, contrary to an object-centric approach. The proposed set of rules is organised along these two axes. The object perspective comprises architecture artifacts focusing on the static structure of the system while the behaviour perspective includes artifacts emphasising the dynamics of the system. The object modeling is presented using the UML static view while the behaviour modeling is presented using the UML use case, interaction and (occasionally) state machine views.

Object and behaviour modeling are parallel activities with their own deliverables. Both activities involve a top-down refinement of the models through a hierarchy of deliverables. Iterations over the same deliverables are also taking place. They produce different versions during the different development phases. They are both continuous activities beginning from the start of the first development phase until the end of the last development phase. The Modeling Rules proposed here do not prescribe any specific development method but rather specific modeling deliverables. In order to position those artifacts on a timeline, they are presented with a three-phase development pattern that can be mapped to many methods.

The elements of the Object perspective are presented with seven (7) diagram types. They are all UML Class diagrams with different purpose and evolution. The first four types (o0, o1, o2 and o3A) package and introduce the system's objects. At least one version of these diagrams is delivered during the Architecture activity. The first three (o0, o1 and o2) are usually stable at the beginning of the Design activity while the fourth one (o3A) will be gradually stabilised during the Design activity. No class operations are shown in these diagrams but Use Cases are associated to object packages and classes. The last three types (o3B, o3C, o3D) provide details about the object software and database implementation. At least one version of these is delivered during the Design activity but they continue to evolve until the end of the Construction activity.

---

[2] Service Case, a non UML term, would be a better expression since the purpose is to describe an object software specification as opposed to a software usage specification aiming at end user validation.

[3] Behaviour [UML 00]: … valid sequences … resulting from external and internal behavioral effects … includes methods, interactions, collaborations, and state histories …

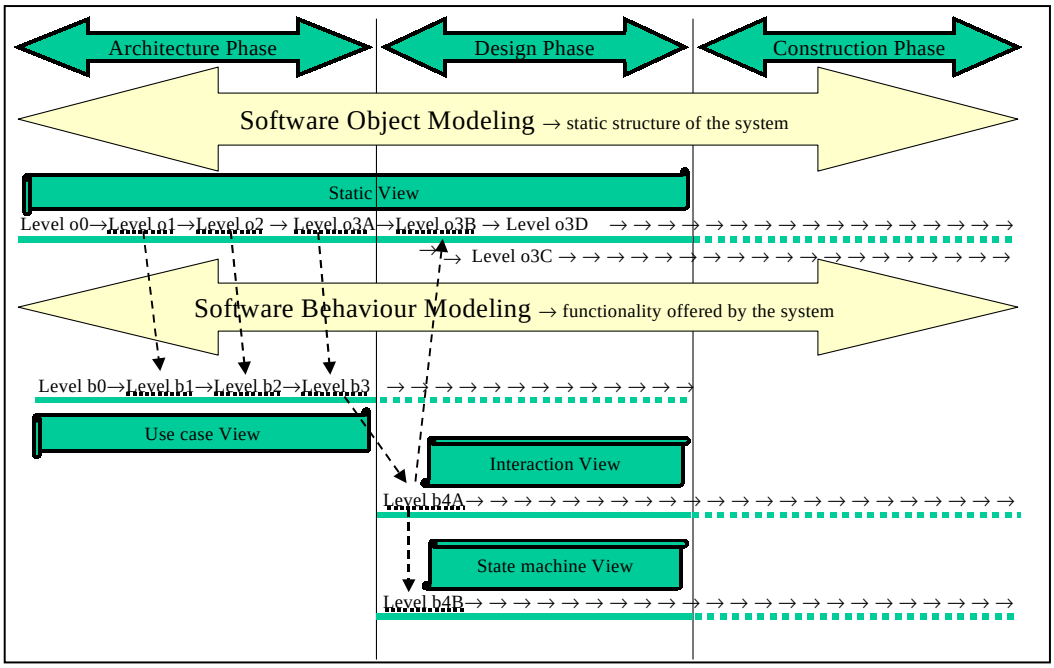


Figure 6 Modeling Rules overview

The elements of the Behaviour perspective are presented using six (6) diagram types. The first four (b0, b1, b2 and b3) describe the system's behaviour with UML Use Case diagrams. At least one version of these diagrams is delivered during the Architecture activity. The first three (b0, b1 and b2) are usually stable at the beginning of the Design activity while the fourth one (b3) will be gradually stabilised during the Design activity. The last two types (b4A and b4B) provide details about the dynamics of specific services. The fifth one (b4A) is a Sequence, Collaboration or Activity Diagram expressing the logic used to implement the service. The last one (b4B) is used to indicate the condition under which some operations are available. At least one version of both types (b4A and b4B) is delivered during the Design activity but they continue to evolve until the end of the Construction activity.

6. Top-level Rules

The Modeling Rules top-level subset (MR-T) introduces the main concepts. It organizes and regulates the foundation for the overall approach.

UML Models[4] (MR-T-1)

The Object perspective uses three UML models: Object Architecture Model (OAM), Object Design Model (ODM) and Database Design Model (DBDM).

- The OAM articulates conceptual level classes and relationships. Class attributes may appear but only to give a flavour of the class semantics. The OAM packages objects into a hierarchy that is also used by the ODM and the DBDM. The last level of this architecture model does not have to be synchronized with the ODM and the DBDM. For instance, these detailed design models may have additional classes and relationships not present in the OAM.

[4] Model [UML 00]… abstraction of a physical system, with a certain purpose… organized into a … hierarchy, where the top-most element represents the boundary …

- The ODM details implementation level classes with attributes and operations. Usually, this detailed design model adds classes and relationships to the OAM and uses the same hierarchy. The ODM targets a good object execution design. Attributes and operations shown in the ODM are those of the source code.
- The DBDM details the persistence implementation structure addressing integrity and performance issues. As for the ODM this detailed design model usually uses the same hierarchy established by the OAM. While the ODM aims at a good object execution detailed design, the DBDM targets a good storage design. Our practical experience has emphasized that they need to be separate because of their different purposes.

The Behaviour perspective uses two UML models: Behaviour Architecture Models (BAM) and Behaviour Design Models (BDM).

- The BAM articulates the high-level services that the system provides. It partitions the system functionality into software-oriented Use Cases i.e. targeting a developer audience.
- The BDM details the implementation logic for services. This detailed design model usually uses the same hierarchy established by the BAM and goes beyond. It describes the logic behind the services identified in the BAM. The BDM expands the BAM hierarchy and establishes a strong linkage with the ODM.

Object Levels (MR-T-2)

The three object models (OAM, OAD and DBDM) use the same four object levels: System Object, Object Division, Object Section and Class. The UML concepts of package[5] and stereotype[6] are used to define these levels. Stereotypes are extension mechanisms permitting customisation and extension of UML model elements with new semantics.

- The System Object is the top-level object package of the system. It is partitioned into Object Divisions.
- An Object Division is a second level object package. Each one is partitioned into Object Sections.
- An Object Section is a third level object package. Each one owns a set of classes.
- A class is the third and last object level. It corresponds to the well-known concept of class[7].

The UML subsystem[8] and components[9] concepts are not used as level names (true in the behaviour perspective as well). A level item does not necessarily provide interfaces and operations and is not necessarily a distributable piece of software. However, any level item (e.g. Object Section) can be a subsystem or a component. In ASIP, all Object Divisions and some Object Sections are subsystems. In addition, some level items are also components.

Behaviour Levels (MR-T-3)

The Behaviour models uses seven (7) behaviour levels: System Service, Service Division, Service Section, Service Group, Service Unit, Master Operation and Subordinate Operation. The terminology for the first three levels is synchronised with the object models e.g. a behaviour item associated to an Object Division is a Service Division. As for the Object levels, the UML concept of stereotype is used to define the behaviour levels. The first five (5) levels are Use Case stereotypes while the last two (2) are

---

[5] Package [UML 00]: …grouping of model elements. … may be nested within other packages …

[6] Stereotype [UML 00]: … generally represents a usage distinction.

[7] Class [UML 00]: …the descriptor for a set of objects with similar structure, behavior, and relationships.

[8] Subsystem [UML 00]: … offers interfaces and has operations, and its contents may be partitioned into specification and realization elements.

[9] Component [UML 00]: … represents a distributable piece of implementation of a system …

Operation stereotypes. The System Service is the first behaviour level of the system. It is a Use Case attached to the System Object and partitioned into Service Divisions.

- A Service Division is a second behaviour level element. Each one is a Use Case attached to an Object Division and partitioned into Service Sections.
- A Service Section is a third behaviour level element. Each one is a Use Case attached to an Object Section and partitioned into Service Groups.
- A Service Group is a fourth behaviour level element. Each one is a Use Case attached to a conceptual class and partitioned into Service Units.
- A Service Unit is a fifth behaviour level element. Each one is a Use Case attached to a conceptual class and partitioned into Service Groups.
- A Master Operation is a fifth behaviour level element. Each one is attached to a design class and uses Subordinate Operations. It is a significant operation of interest for understanding the class. It is usually part of the public interface of a class.
- A Subordinate Operation is a sixth behaviour level element. Each one is attached to a design class. It is a significant operation contributing to understand the concept of execution of its Master Operation.

7. Object Perpective Rules

The Modeling Rules object subset (MR-O) organizes and regulates the object behavior modeling artifacts. The elements of the Object perspective are presented with seven (7) diagram types.

System Object Context (MR-O-1)

This diagram aims at making the reader:

- comprehend the system scope in terms of data and discover external data used by the system;
- look at the top-level partitioning of the object perspective; and
- develop some insight into what the system does with data.

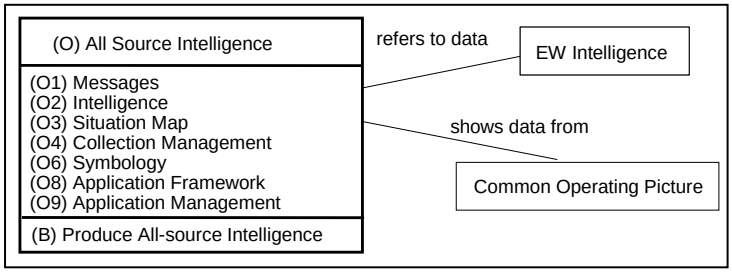


Figure 7 System Object Context diagram

The System Object Context diagram (o0) shows: the targeted System Object package and its links with foreign System Object packages. It is used to present the three object models i.e. OAM, ODM and DBDM. It has usually the same content for the three models. The System Objects are shown using the UML Class notation with an adapted significance of the three compartments. The top compartment holds the System Object name (a "what-it-is" flavour); the middle list compartment holds a list of Object Division packages partitioning the system using an object perspective; the bottom list compartment holds the name of the single System Object Use Case (a "what-it-does flavour"). The middle and bottom list compartments are not shown for foreign System Objects i.e. only the name compartment is used.

System Object (MR-O-2)

This diagram aims at making the reader:

- comprehend the kind of objects handled by the system;
- discover sections belonging to a division; and
- develop some insight into what the system does with specific Object Division data.

The System Object (o1) diagram shows: the Object Divisions packages under the targeted system; their interrelationships and; their links with foreign System Object packages (introduced at the higher-level o0). It is used to present the three object models i.e. OAM, ODM and DBDM. It has usually the same content for the three models. As for o0, Object Divisions are shown using the UML Class notation with an adapted significance for the three compartments. The top compartment holds the Object Division name; the middle list compartment holds a list of Object Section packages partitioning the Object Division; the bottom list compartment holds a list of Service Division Use Cases attached to the Object Division. The middle and bottom list compartments are not shown for foreign System Objects i.e. only the name compartment is used.

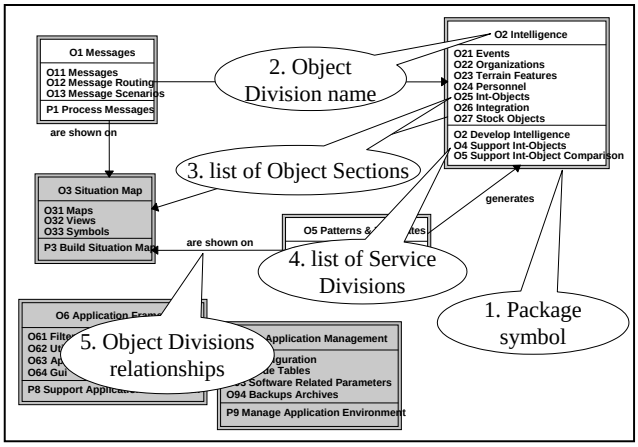


Figure 8 System Object diagram

Object Division (MR-O-3)

This diagram aims at making the reader:

- determine the scope of the division in terms of objects;
- discover classes belonging to a section; and
- develop some insight into what the system does with specific Object Section data.

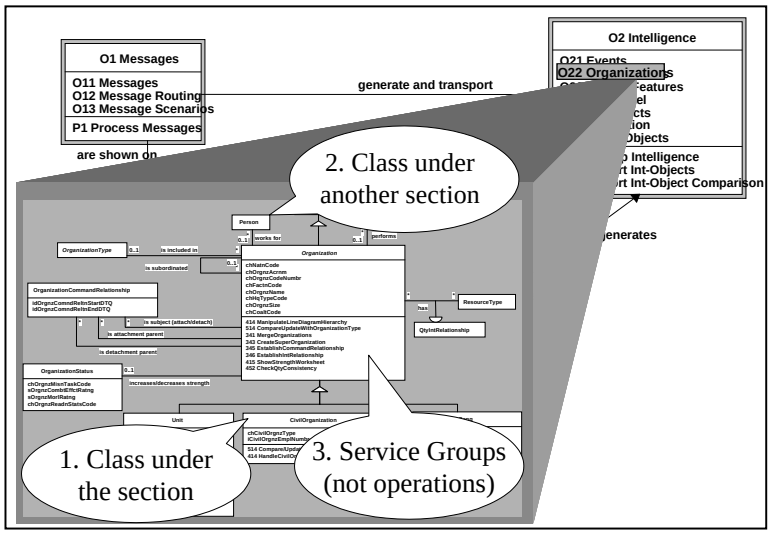


Figure 9 Object Division and Section Architecture diagrams

An Object Division (o2) diagram shows: Object Sections under the targeted division; their interrelationships and; their links with foreign System Objects presented at the higher-level o1. It is used to present the three object perspective models i.e. OAM, ODM and DBDM. It has usually the same content for the three models. As for o0 and o1, Object Sections are shown using the UML Class notation with an adapted significance of the three compartments. The top compartment holds the Object Section name; the middle list compartment holds a list of classes under the Object Section; the bottom list compartment holds a list of Service Sections attached to the Object Section. The middle and bottom list compartments are not shown for foreign System Objects i.e. only the name compartment is used.

Object Section Architecture (MR-O-4)

This diagram aims at making the reader:

- comprehend the semantic of the section's classes;
- discover relationships of the section's classes; and
- develop some insight into what the system does with specific Class data.

An Object Section Architecture (o3A) diagram shows conceptual classes[10] under the targeted section, their interrelationships and their links with foreign conceptual classes of other sections. It is used to

[10] A conceptual class does not necessarily exist in the source code while a design class does.

present the OAM. Conceptual classes are shown using the UML Class notation with a different usage of the third compartment. The bottom list compartment does not hold a list of operations but a list of Use Cases. For a class under the targeted section, it shows semantically significant attributes and all Service Groups attached to the class. For foreign classes, it shows only class names.

Object Section Design (MR-O-5)

This diagram aims at making the reader:

- acquainted with the physical implementation of the section's classes in the software;
- discover physical class attributes and relationships of the section; and
- well-informed as to what the system does with specific Class data.

An Object Section Design (o3B) diagram shows: design classes[10] under the targeted section, their interrelationships, and their links with foreign design classes of other sections. It is used to present the ODM. Design classes are shown using the UML Class notation with a different usage of the third compartment. The bottom list compartment does not hold a list of operations but a list of Use Cases. For a class under the targeted section, it shows attributes and all Service Units attached to the class. For foreign classes, only class names are shown. Our practical experience has emphasized that once construction has begun, this diagram should be maintained through reverse engineering from the source code.

The Object Section Design (o3B) diagram usually results from an evolution of the Object Section Architecture (o3A). In order to optimize data structure and algorithms, this evolution may involve design decisions such as creation of new classes (e.g. containers), merging of classes with super-class, association redundancies, addition of physical attributes (e.g. modification flag) and, increasing inheritance. Changes to an o3B diagram do not involve changes to the corresponding o3A diagram except if it involves a significant modification in a class or relationship meaning.

Object Section Database (MR-O-6)

This diagram aims at making the reader:

- acquainted with the physical implementation of the section database storage; and
- find out attributes and relationships that are implemented in the database.

An Object Section Database (o3C) diagram shows: tables under the targeted section, their interrelationships, and their links with foreign tables under other sections. It is used to present the DBDM. Tables are shown using the UML Class notation with only the first two compartments. The second compartment, that shows columns, is not used for foreign tables i.e. only table names are shown. Our practical experience has emphasized that once construction has begun, this diagram should be maintained through reverse engineering from the database definition language (DDL).

Class Design (MR-O-7)

This diagram aims at making the reader:

- acquainted with the physical implementation of a specific class;
- discover physical attributes and relationships of the class; and
- discover the physical operations of the class.

A Class Design (o3D) diagram focuses on a single class and its relationships with other classes. It is used to present the ODM. In opposition to the o3B diagram, the third compartment shows real operations of the class. This diagram should be created and maintained through reverse engineering from the source code.

8. Behaviour Perpective Rules

The Modeling Rules behaviour subset (MR-B) organizes and regulates the behavior modeling artifacts. The elements of the Behaviour perspective are presented with six (6) diagram types. Contrary to the object perspective, the interpretation of relationships in a Use Case diagram is often difficult in terms of expressiveness and accuracy. The textual description is essential [Cockburn 00] to understand the relationships and some Use Case diagrams can be skipped without major drawbacks. Analysis of relationships between Use Cases is important since it can bring the need for new classes at design time in order to capture reusability of a common behaviour between different objects. It can have a retroaction on the behaviour perspective architectural partitioning.

System Service Context (MR-B-1)

This diagram aims at making the reader:

- discover entities influencing or being influenced by the system.

The System Service Context diagram (b0) shows: the targeted System Service Use Case and its links with actors. It is used to present the two behaviour models i.e. BAM and BDM. It has usually the same content for the two models.

System Service (MR-B-2)

This diagram aims at making the reader:

- aware of the kind of services offered by the system; and
- aware of the relationships between the service divisions.

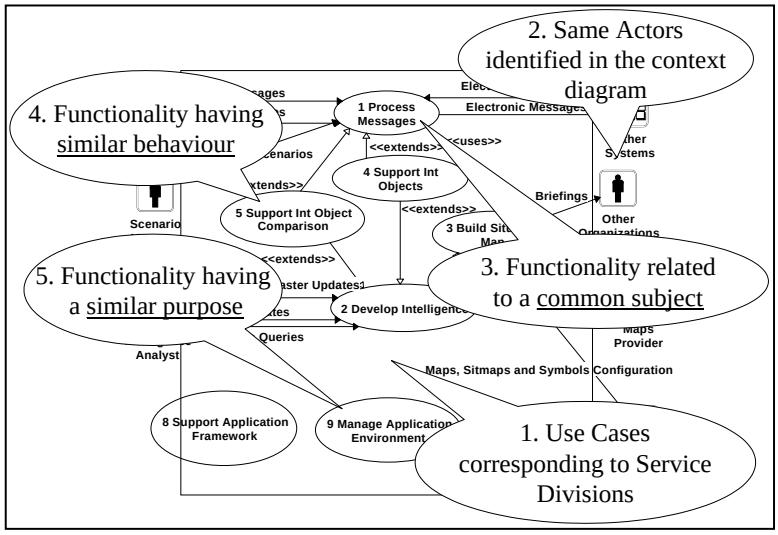


Figure 10 System Service diagram

The System Service (b1) diagram shows: the Service Division Use Cases of the system, their interrelationships, and their associations with actors (introduced at the higher-level b0). It is used to present the two object models i.e. BAM, and BDM. It has usually the same content for the two models.

Following are some practices that were observed *a posteriori* about the behaviour partitioning in ASIP. A Service Division or Service Section is either a set of functionality elements:

- related to a common subject e.g. a Message object;
- having similar behaviour e.g. comparison of objects; or
- having a similar purpose e.g. managing the application environment.

Service Division (MR-B-3)

This diagram aims at making the reader:

- determine the scope of the division in terms of functionality;
- aware of the relationships between the service sections.

A Service Division (b2) diagram shows: Service Section Use Cases under the targeted division, their interrelationships, and their links with actors introduced at the higher-level b1. It is used to present the

two behaviour perspective models i.e. BAM and BDM. It has usually the same content for the two models.

This diagram can be replaced by an enumeration of the Service Section Use Cases of the Service Division.

Service Section (MR-B-4)

This diagram aims at making the reader:

- comprehend the detailed functionality offered by the section;
- discover relationships involving the section's service units.

A Service Section (b3) diagram shows: Service Group and Service Unit Use Cases under the targeted section, their interrelationships, and their links with actors. It is used to present the two behaviour perspective models i.e. BAM and BDM. It has usually the same content for the two models.

The Service Group level did not exist in the first version of the modeling rules but experience has demonstrated that it is required to group service units that should be managed and documented together.

This diagram can be replaced by an enumeration of the Service Section Use Cases of the Service Division.

Service Unit Design (MR-B-5)

This diagram aims at making the reader:

- comprehend how the service units provide the service; and
- find out the main operations and classes involved in the service unit.

A Service Unit Design (b4A) is either a UML Collaboration, Sequence, or Activity Diagram showing: Master Operations and Subordinate Operations, and their interactions in the context of the targeted Service Unit. It is used to present the detailed design portion of the BDM. In a Sequence diagram, the Master Operations appear as messages from an object external to the service being described by the diagram.

This diagram is a key for software architect reviewers in the prevention of performance problems and logic complexity that are often not seen before performance testing phase, particularly for distributed object-oriented software.

Service Unit Availability (MR-B-6)

This diagram aims at making the reader:

- discover when master operations of the sevice unit are available.

A Service Unit Availability (b4B) diagram is a UML statechart diagram showing: significant states involved in the Service Unit, available internal transition operations of the states, and other available transition operations involving a state change. The name of the state is shown in the first compartment of the state symbol. Internal transition operations are shown in the second compartment of a state symbol. Other transition operations are shown with relationships between state symbols. This diagram type is used to present the detailed design portion of the BDM.

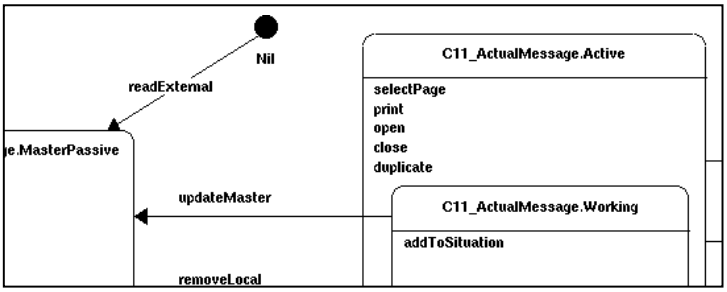


Figure 11 Service Unit Availability diagram

This diagram is not required for Service Units where the availability of operations is not an issue.

9. Summary of the approach

The following elements summarise the approach:

- The Modeling Rules are split into Object and Behaviour Modeling.
- The Object perspective uses the UML Static view.
- The Behaviour perspective uses the UML Use Case view, Interaction view and the State machine view.
- The Object perspective uses four (4) Object levels: System, Division, Section and Class.
- The Behaviour perspective uses five (5) architecture Service levels: System, Division, Section, Group and Unit.
- The Behaviour perspective uses two (2) design Operation levels: Master and Subordinate.
- Object perspective items are referring to Behaviour perspective items but class operations are not identified before the object design artefacts is done.
- Behaviour perspective design items are referring to Object perspective items.
- The object perspective uses four (4) types of architecture diagrams (o0, o1, o2 and o3A) and three types of design diagrams (o4B, o4C and o4D).
- The behaviour perspective uses four types of architecture diagrams (b0, b1, b2 and b3) and two types of design diagrams (b4A and b4B)

The Modeling Rules allow software developers to understand a set of linked UML diagrams describing a complex system without having first to understand how those diagrams are organised. Some benefits for software architects are:

- Handling the system complexity with UML stereotypes and packages.
- Enabling navigation paths through the models.
- Facilitating impact analysis of system changes (e.g. using b3A diagrams, that links service units with class operations, one can find out service units using an operation that must be modified).
- Enforcing a complete synchronisation between the diagrams that facilitates the understanding, evolution and maintenance of the models.

10. Conclusion

UML and related Object Management Group (OMG) standards such as XMI already bring a first level of modeling interoperability. In addition, a set of modeling rules such as the ones presented here facilitates appropriate information exchanges about complex systems without having to sip through huge documentation packages to understand. For instance, it facilitates communications allowing a nation:

- to request a specific modeling artifact;
- to provide the relevant information; and
- to understand quickly modeling artifacts obtained from others.

Such communication improvements would reduce the decision cycle time related to software interoperability and the development of interoperability solutions, for instance by determining:

- the opportunity or relevance of interoperability between systems;
- the feasibility of such interoperability;
- the specific software items that should be involved in the interoperability solution;
- if a system includes software items that can be used to build a new system (i.e., componentware); or
- if a system can be implemented with different technologies using the same software models.

An approach being tested at DREV has been presented. The Modeling Rules presented here are being used to build the software models of ASIP. The approach is still under development and needs adjustments e.g. specific rules indicating the level of detail to show in Sequence diagrams.

The main message to retain from this paper is that nations should agree on a set of common modeling artefacts in order to ease the development of interoperability solutions. There is a need for a common modeling language. By analogy with speaking languages, UML 1.3 provides a very good dictionary but grammatical rules needs to be further developed. One could consider the approach presented as an attempt to provide some of these additional grammatical rules but there is still much effort required to investigate, extend, implement and experiment with a set of Modeling Rules. Until now, our experience has emphasized usefulness of:

- an independent behaviour perspective contrary to an object-centric approach;
- a set of coherent links between the different views;
- a database model (object or not) aiming at a good storage design separate from the object design model; and
- reverse engineering to maintain design diagrams once construction has begun.

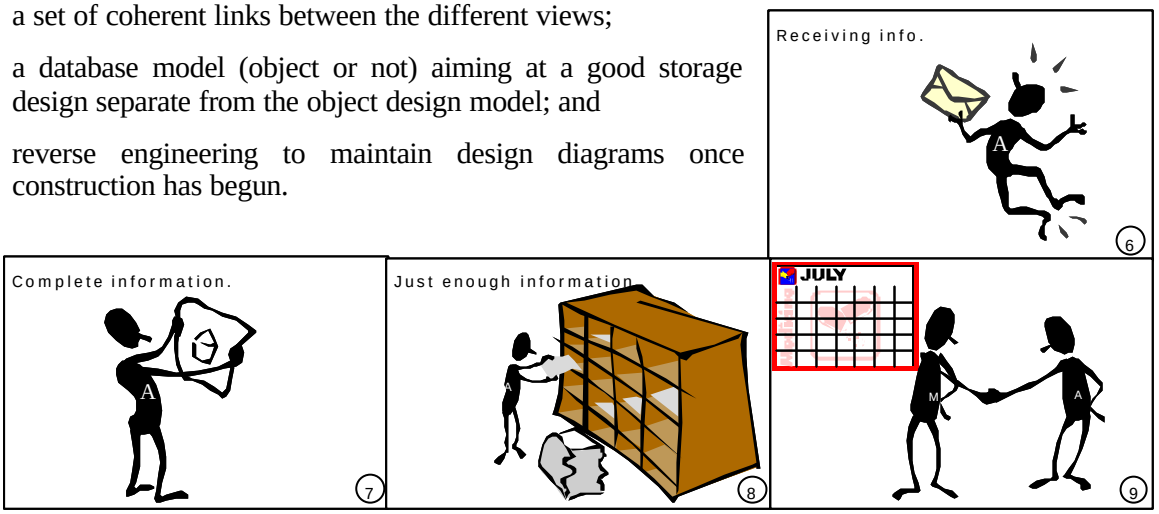


Figure 12 Getting just the right and needed information

This conclusion is illustrated (Figure 12) by revisiting the last few frames of the cartoon used previously to introduce the problematic. The first three steps are identical. Step 4 and 5 are similar but the requests are made more precise using the common grammatical rules.

6. The Canadian Architect receives answers that are coherent with what he was expecting.
7. Using a common language, the US reply provides just the right needed information.
8. The French Architect sends just enough information without overloading the Canadian Architect.
9. Finally, the Canadian Architect is able to meet his manager's request.

11.References

[TIF 92] Gauvin, M., Lapointe, S. and Lizotte, M., "Tactical Information Fusion Prototype: Evaluation Through an Abbreviated Command Post Exercise", DREV R-9503, September 1995, UNCLASSIFIED.

[NL 95] Gauvin, M. and Thibault, G., "Tactical Information Fusion Prototype: Participation in the ABCA Northern Lights Command Post Exercise", DREV R-9531, January 1996, UNCLASSIFIED.

[ACTIF 97] "ACTIF Architecture, Functional and Internal Standards", DMR Consulting Group Inc., Contract W7701-6-2782, November 1997, UNCLASSIFIED.

[LIEWA 01] Blain, Dominique (2001), "Mesurer et expliquer l'impact du partage en direct des situations militaires du système ASIP sur la qualité du renseignement",' Université du Québec à Montréal, (Doctorate thesis to be published).

[UML 00] "Unified Modeling Language Specification", Object Management Group, Framingham, Mass., 2000.

[MIL-498 96] "MIL-STD 498, Software Development and Documentation", Defense Information Systems Agency, 19 March 1996, UNCLASSIFIED.

[DMR 90] "Guide de développement d'un système", Recherche et développement Groupe DMR inc., Montréal, Québec, 1990.

[IEEE 99] "IEEE/EIA Std 12207.1-1997, Software life cycle processes - Life cycle data", The Institute of Electrical and Electronics Engineers, Inc., New York, NY, 1998.

[Cockburn 00] Cockburn, A., "Writing Effective Use Cases", Addison-Wesley, Mass., 2000.

[Cockburn 98] Cockburn, A., "Surviving Object–Oriented Projects", Addison-Wesley, Mass., 1998.

[Rumbaugh 91] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W., "Object-Oriented Modeling and Design", Prentice Hall, Englewood Cliffs, N.J., 1991.

[Rumbaugh 99] Rumbaugh, J., Jacobson, I., Booch, G., "The Unified Modeling Language Reference Manual", Addison-Wesley, Mass., 1999.

Modelling Command and Control Information Systems by UML

Heinz Faßbender & Gerhard Bühler
FGAN/FKIE
Neuenahrer Straße 20
53343 Wachtberg-Werthoven
Germany
fass@fgan.de & buehler@fgan.de

1. SUMMARY

The complexity of command and control information systems is increasing continuously. The result of the augmentation of their interoperability to systems of own troops or foreign nations raises the difficulty for administration and maintenance. A possible approach to manage this problem is the use of a visual modelling annotation. It helps to manage the complexive structures of command and control information systems.

In the context of information modelling, the data model which is represented as an entity relationship diagram [Chen, 1976], has become a standard. This is a first step of the abstraction from the source code. But to give a complete survey, the design of the complete system has to be modelled.

In former times when the structured style of programming was used, the architectures of programs were modelled by structured design techniques [Yourdon & Constantine, 1989]. In the last decade structured programming has been more and more replaced by object oriented programming. This leads to the definition of many object oriented analysis and design methods which are unified in the **unified modelling language (UML)** [OMG].

UML now is the standard for modelling object oriented information systems. In particular, UML offers an annotation for modelling interfaces. Hence, it seems to be the most promising candidate for modelling command and control information systems which should be interoperable.

In this paper we illustrate how we model the existing experimental integration platform for command and control information systems **INFIS** which has been developed in our institute and which serves as the German testbed for interoperability tests in the *ATCCIS study* [ATCCIS] and the *Multilateral Interoperability Programme (MIP)* [MIP], by UML. That means, this paper will not include a description of modelling the different phases in the development of a new command and control information system. It describes how to model an existing command and control information system.

The paper is structured as follows. In Section 2, the global structure of INFIS is modelled by a self-defined annotation and the usage of UML will be motivated. We present our UML-modelling process for INFIS in Section 3. This process determines the structure of the following sections. Modelling INFIS' high-level structure, its low-level structure, and their combination will be described in Sections 4, 5, and 6, respectively. We finish the UML-modelling process of INFIS by presenting a dynamic model that illustrates the interactions between INFIS' objects for computing an application.

2. INFIS

INFIS is structured as a three level architecture (cf. Figure 1) of at least one **Domain**. Since the system is a multiuser system, it consists of a finite amount of graphical user interfaces **GUIs**. They are implemented as a Java Applet. The number of GUIs that are active, maybe arbitrary. Since the GUIs are implemented as a Java Applet, the system can be used totally platform independent with the following two minimal requirements to the resources of the clients:

1.) Only an internet browser and

2.) a connection to the internet or an intranet is required.

The information of a Domain is stored in a database system **DBS** corresponding to the ATCCIS data model which is the main requirement for realising the interoperability to other ATCCIS-conformant command and control information systems. The third tier is called the **Kernel**. Together with its connected GUIs it is called a

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Subsystem. The Kernel implements the application logic of the Subsystem and the connection to other Subsystems, as well as to other Domains. In particular, it implements the ATCCIS replication mechanism. Furthermore, the interfaces between the GUIs and the Kernels are implemented by the CORBA standard [OMG, 2001] which also supports the platform independent access to the system. The interface between a Kernel and the DBS is realised by a proprietary implementation that supports the concurrent access to the DBS.

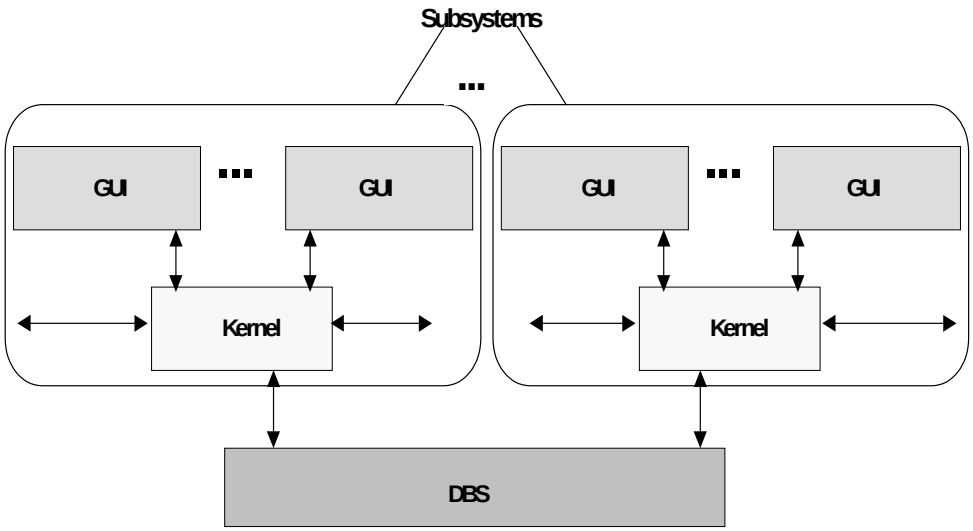


Figure 1: A Domain of INFIS

The most interesting part in modelling the system is the **Kernel**. Its structure is illustrated in Figure 2. Every component of a Kernel is implemented as an Ada95 task [Barnes, 1995]. The tasks communicate by messages. A task may communicate to every task which is in the illustration in Figure 2 close to it. I.e. every task except of the Applications can communicate with each other. **Applications** are positioned in the upper tier. They can only communicate with the **Application Control**, i.e. it is the border between the Applications and the rest of the Kernel. As an example of an Application we will use the initiation of the actual situation's display. A Session of a user is controlled by the **Session Control** and the access to the data base system is controlled by the **Data Base Control**. The **Basic Operating System** offers functionality of a multiuser and multiprocess operating system. It is realised by a proprietary implementation to support the platform independent implementation of the system.

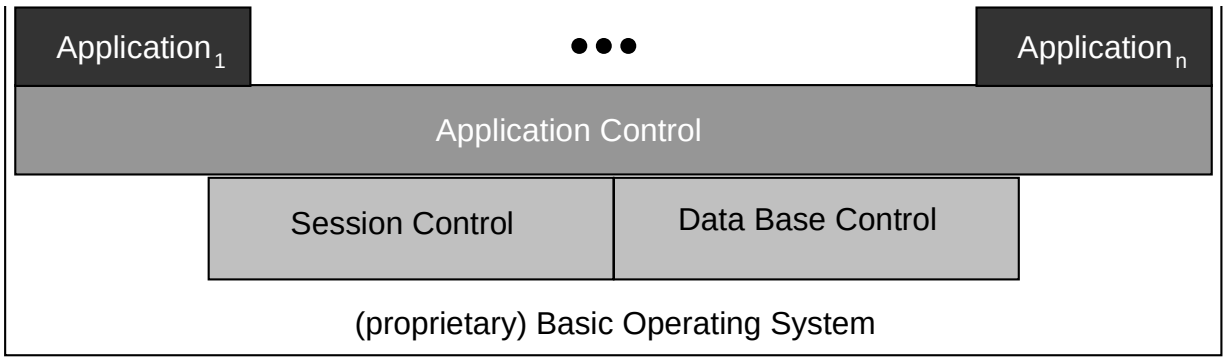


Figure 2: Structure of a Kernel

INFIS' architecture and the structure of a Kernel are illustrated in a self defined annotation. It has only an illustrative and informal meaning, i.e. the models may be misunderstood by their readers and a lot of additional explanations are needed to give them a nearly formal meaning. Furthermore, there does not exist any software development tool which would support this annotation and there is no possibility to automatically produce code from these models with a case tool.

These are two reasons, why we have decided to develop formal models of INFIS by an annotation that fulfils the following requirements:

1. The annotation has to be understood by interesting people, since it has a nearly formal meaning.
2. The annotation must be standardised.
3. There have to exist software development tools that produce code from the models automatically.

The only existing annotation which fulfils all of those three requirements is the **Unified Modelling Language** that has been defined by Grady Booch, Ivar Jacobson, and James Rumbaugh from Rational Software Corporation [Rational], and that has been standardised by the Object Management Group OMG [OMG] which is a non profit consortium of nearly 400 companies. That is why we have decided to model INFIS by UML.

In the rest of the paper we will define the UML-modelling process for INFIS and we will illustrate the different phases of this process by presenting some models as examples.

3. UML-MODELLING PROCESS FOR INFIS

The UML-modelling process for INFIS is illustrated as an UML-model itself in Figure 3. Notice that by this process only the existing system is considered. In Section 8 we will describe how to extend this process for integrating new applications to the system.

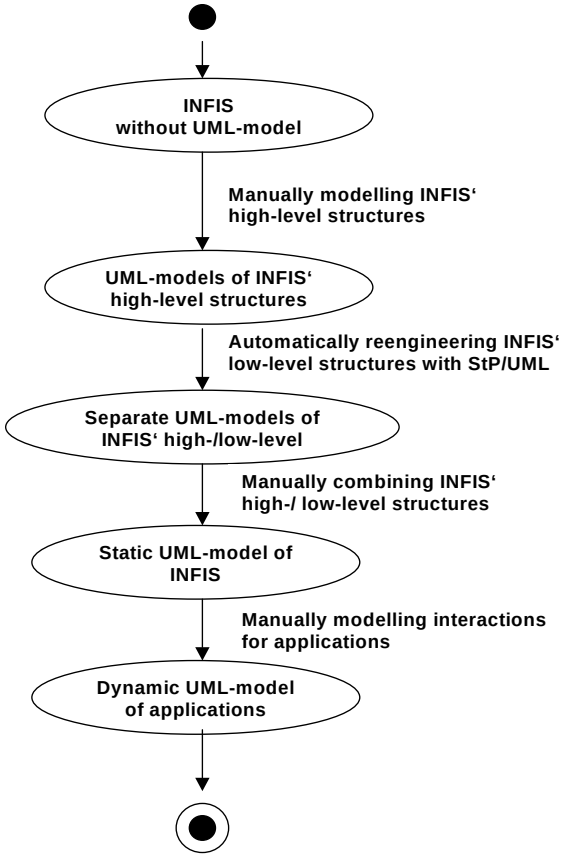


Figure 3: UML-Modelling Process for INFIS

The process starts with the existing version of INFIS which is only modelled by the self-defined annotation that is illustrated by Figures 1 and 2. First of all, the high-level structures of INFIS are modelled manually. After that, or in parallel to the modelling of the high-level structures, we use the reengineering component of the software engineering tool *Software through Pictures* from Aonix [Aonix] for modelling INFIS' low-level structures that include the defined classes, their relationships, and the hierarchy of inheritance. After we have finished the modelling of INFIS' high-level structures, as well as its low-level structures, we combine these structures manually. Finally, we develop a dynamic model for applications which describes the interactions between the objects of the static model by computing the applications.

4. UML-MODELS OF INFIS' HIGH-LEVEL STRUCTURES

We start the modelling process by modelling INFIS' high-level structures. In opposite to the low-level structures which will automatically be illustrated by the reverse engineering component of the software development tool, the high-level structures have to be modelled manually.

In this section we present two UML-models of the high-level structures which correspond to the two models in Figures 1 and 2. The global architecture of INFIS is illustrated by the UML-model in Figure 4.

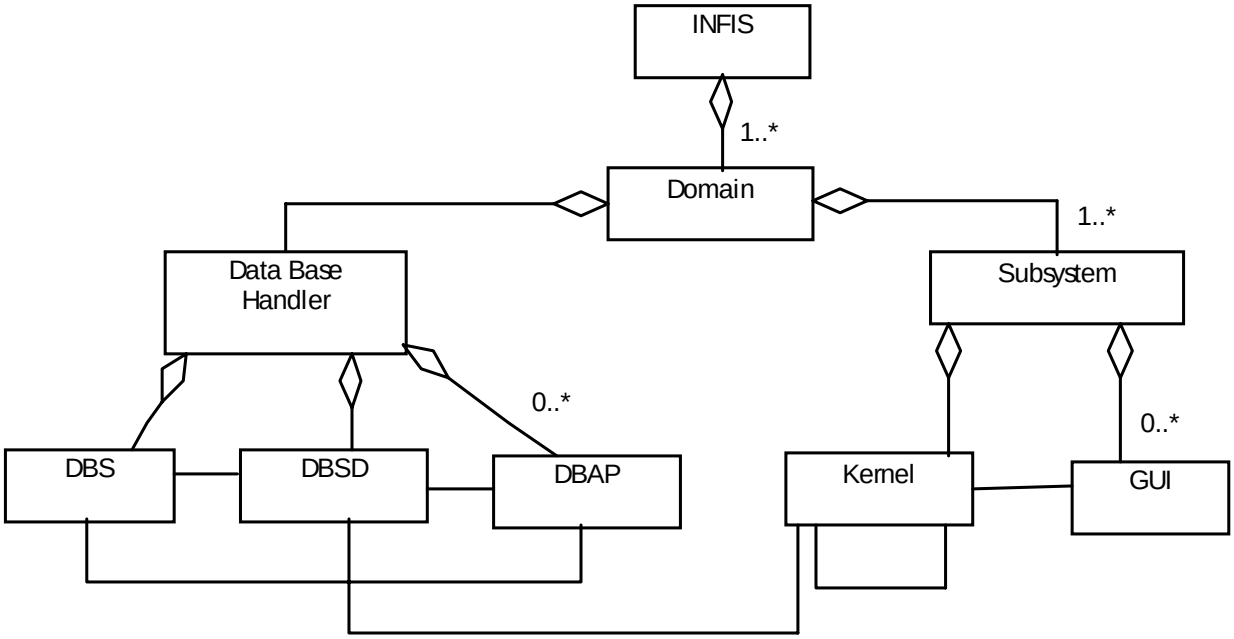


Figure 4: UML-Model of INFIS' Global Architecture

This model gives a more detailed description of INFIS' structure than the model in Figure 1. It contains the components of the architecture which are illustrated as *classes* in rectangles. The lines terminated by a rhomb, are called *aggregations*, i.e. it expressed a *part of*-relation. For example INFIS consists of at least one (Notation: 1..*) Domain which itself consists of at least one Subsystem and exactly one (This is the default value, if the end of an association or aggregation is not qualified.) Data-Base Handler, etc. The pure lines are called *associations.* An association between two classes indicates that there may be a message exchange between those classes.

Notice that the Data Base Handler, the DBSD, and the DBAP are components which implement the concurrent access to the Data Base System DBS.

The structure of a Kernel is illustrated as UML-model in Figure 5.

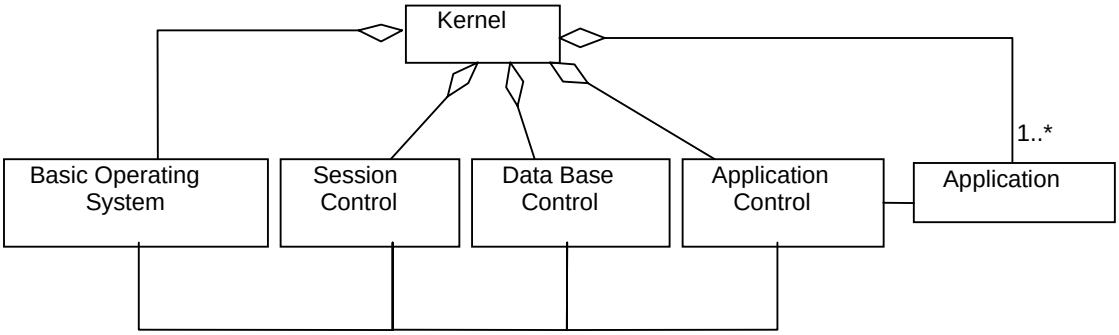


Figure 5: UML-Model for a Kernel

As described in Section 2 and illustrated in Figure 2, a Kernel consists of (*aggregations*) a Basic Operating System, a Session Control, a Data Base Control, and an Application Control which are associated to each other (*associations*). Furthermore, at least one Application belongs to the Kernel. The Applications are associated only to the Application Control.

If the UML-model in Figure 5 is compared to the model of the Kernel in Figure 2, then the difference of the expressive power between the two modelling annotations becomes obvious.

Beside manually modelling INFIS' high-level structures, its low-level structures can be modelled automatically. The result of this phase of the UML-modelling process for INFIS will be illustrated in the following section.

5. UML-MODELS OF INFIS' LOW-LEVEL STRUCTURES

For modelling INFIS' low-level structures, i.e. the classes in the code, their associations, and the inheritance hierarchies of the system we use the reengineering component of the software development tool *Software through Pictures* from Aonix. We have chosen this tool, because we also use the Ada95 development tool *Object Ada* from the same company. Software through Pictures produces a lot of class diagrams and class tables that include additional information about the classes of the system. Because of the syntactical structure of Ada95 programs which is not pure object oriented (compared to Java programs), the classes in the produced models only consist of attributes. Thus, the methods have to be manually added to the classes of the models. Nevertheless, the automatic generation of the models saves a lot of time for modelling and it reduces the number of mistakes in the models.

To give an impression of the result of the reengineering component of Software through Pictures, a part of the inheritance tree is shown in Figure 6. The classes of INFIS are represented by the nodes of the tree. The topmost node represents the class called **node**. It is the fundamental class of INFIS. There exist upto 1000 derivations from this node.

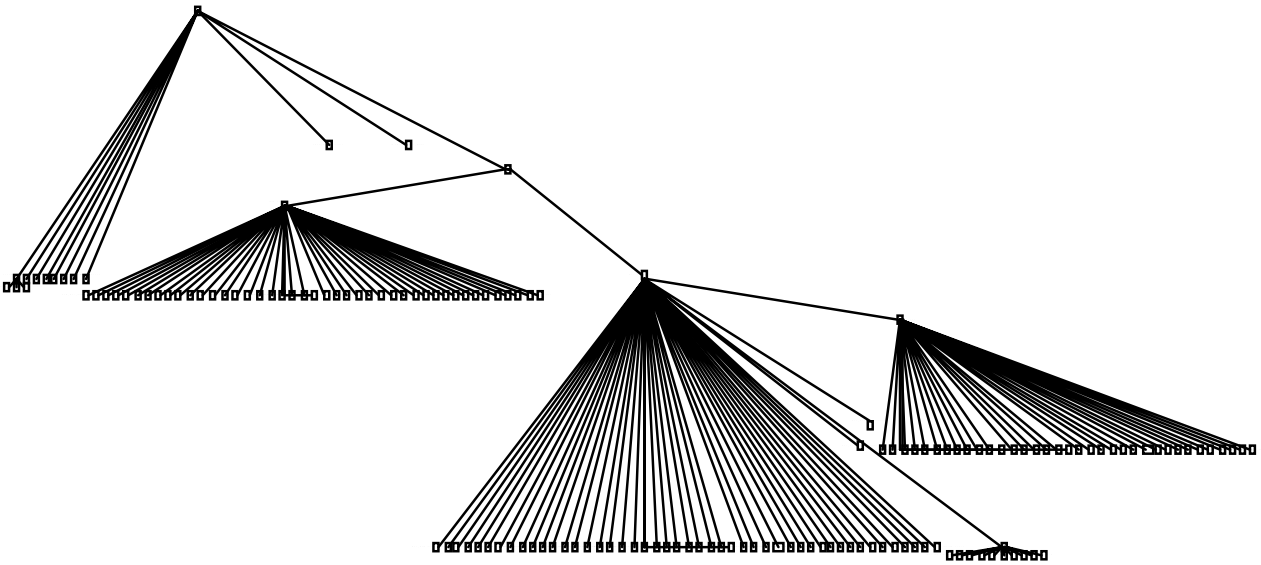


Figure 6: Inheritance Tree for the Application Control Components

In Figure 7 we present a more detailed result of the reengineering tool. Notice that the diagram in Figure 7 is not a part of the diagram in Figure 6 which include the inheritance dependencies whereas Figure 7 only includes associations between classes and packages.

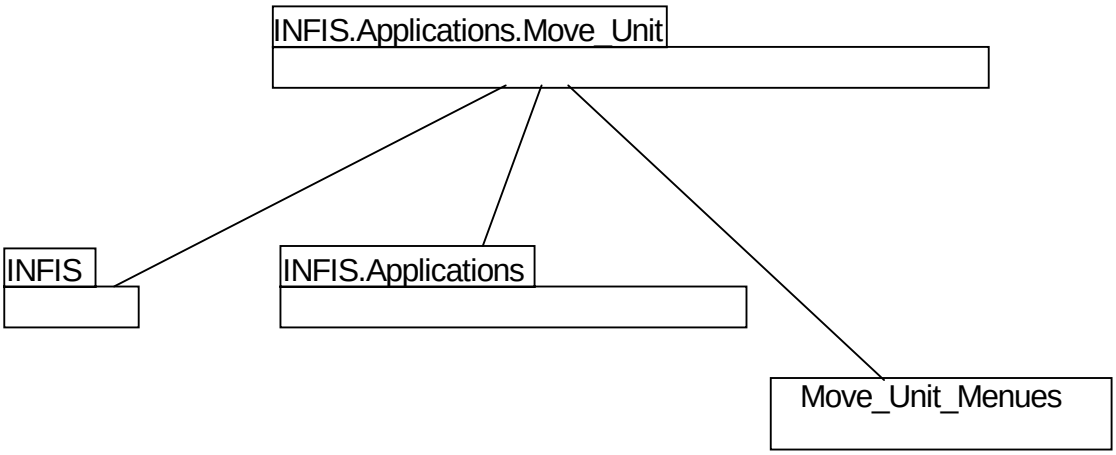


Figure 7: Detailed Result of an Automatic Generated UML-Model of the Low-Level Structure

This UML-model illustrates the associations of the INFIS application **Move_Unit** that allows to give a unit a new position on the situation display. The application **Move_Unit** is modelled as the package *INFIS.Applications.Move_Unit*. This results from the mapping of Ada95 programs to UML. It is associated to the parent packages *INFIS* and *INFIS.Applications*. Furthermore, it is associated to the class *Move_Unit_Menues* which stores the name and the coordinates of the unit that should be moved.

After the the UML-models of INFIS' high-level structures in the previous section and the illustration of the results of the reengineering process we will explain how to integrate the UML-models of the high- and low-level structures.

6. COMBINATION OF UML-MODELS OF INFIS' HIGH- AND LOW-LEVEL STRUCTURES

During reengineering of the low-level structures we found out that the classes in the high-level structures are implemented by a huge amount of classes and packages in the code. That is why we have to connect the abstract classes in the high-level models to the implementations of their representations in the low-level models. Since the specification of UML 1.3 only allows that a package implements an interface (and not a class), the connections are realised via interfaces. For example, if we consider the high-level model in Figure 5 and the low-level model in Figure 7, we identify the package *INFIS.Applications* the name of which immediately indicates that it represents the implementation of the abstract class *Application* in Figure 5.

Then we simply combine the two models by associating the abstract class *Application* in Figure 5 with the package *INFIS.Applications* in Figure 7 via the Interface *Application Interface*. The result of this combination is represented in Figure 8 where also another low-level model that represents the implementation of the application *INFIS.Applications.Lokalis* is added.

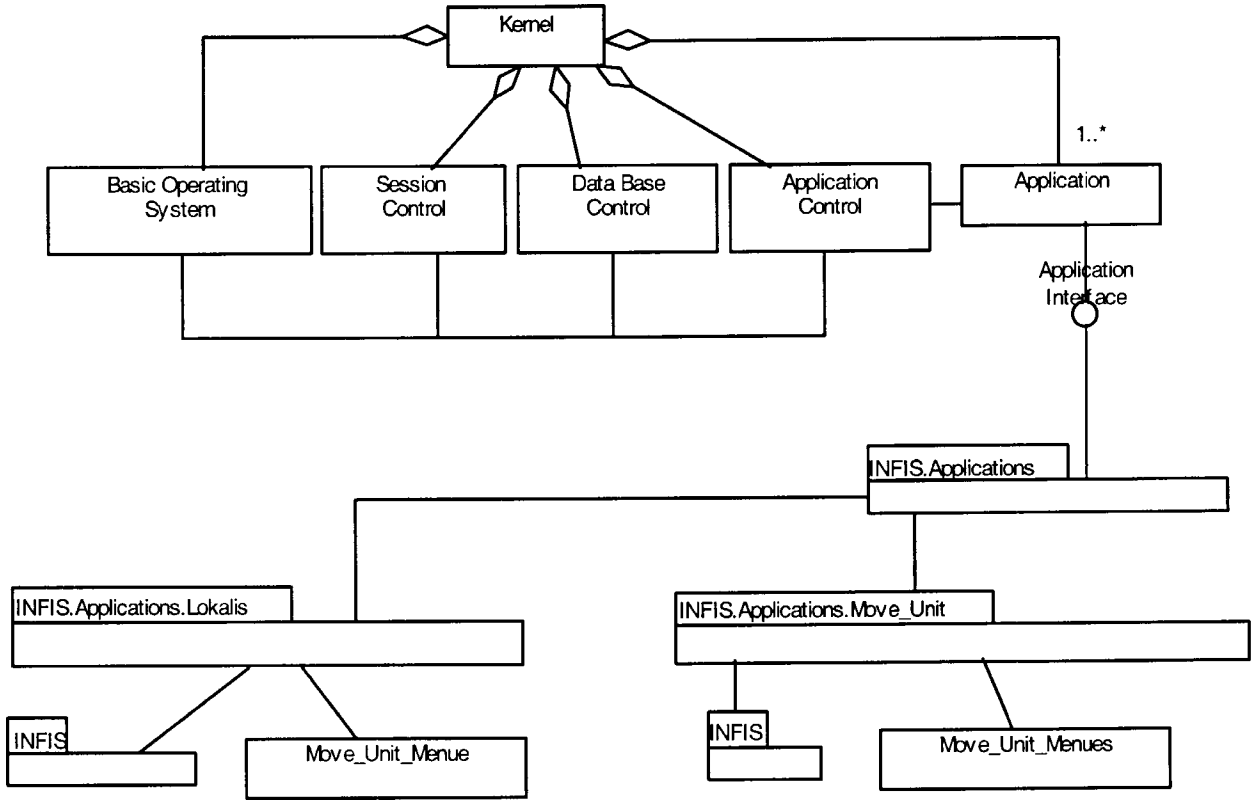


Fig. 8: Combination of UML-Models of INFIS' High- and Low-Level Structures

The immediate integration of all UML-models of INFIS' low-level structures into packages of INFIS' high-level structures would obviously lead to unreadable illustrations. To solve this problem, we use the functionality of Software through Pictures which allows to switch into the substructures of packages as well as of classes.

7. DYNAMIC MODEL OF INFIS' APPLICATIONS

Up to now only the process of modelling the static structure of INFIS has been described. But, UML also offers some annotations as **sequence diagrams, collaboration diagrams, state-transition diagrams,** and **activity diagrams** to model the dynamic behaviour of the system. We will explain the process of modelling INFIS' dynamic behaviour by modelling the *computation of an application* through the different components of INFIS' static structure.

For this purpose, we use the modelling technique of *sequence diagrams* that is also used for modelling business processes. Sequence diagrams are the best choice, because

1. **collaboration diagrams** subsume the interactions between objects by abstracting from the order of the interactions. Since the order is an important fact in describing the dynamic behaviour, collaboration diagrams are not as appropriate as sequence diagrams.
2. **state-transition diagrams** illustrate the reactions of one single object on events. They are not appropriate for modelling the interactions between objects.
3. **activity diagrams** are very close to sequence diagrams. But they are designed for modelling parallel processes.

In Figure 9 we present the UML-model for the computation of a general application.

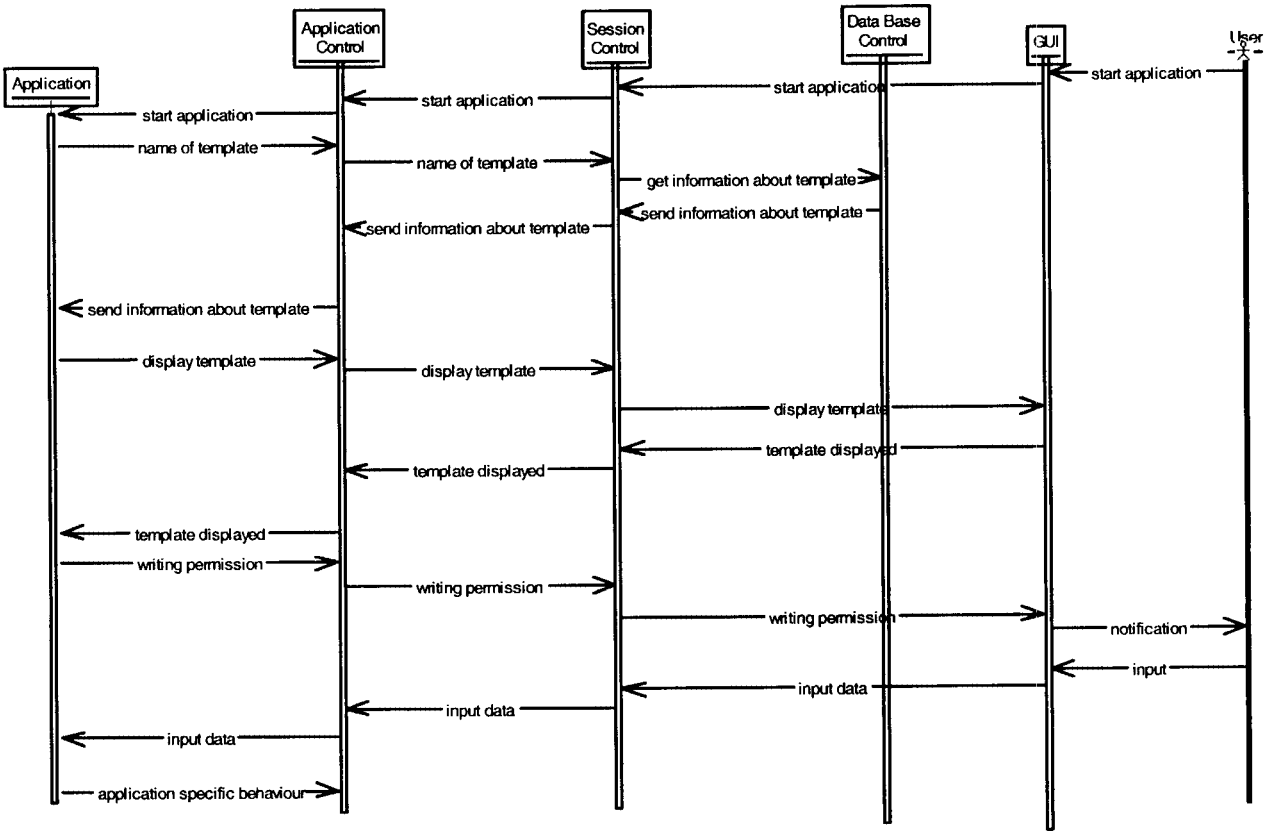


Fig. 9: UML-Sequence Diagram for the Computation of an Application

In the sequence diagram there is one bar for every object and one bar for the user, if the user is involved in the process. The time axis is vertically from top to bottom.
When the user starts an Application, the GUI, the Data Base Control, the Session Control, and the Application Control are still running, but the Application object does not exist. Then the Application's initiation is sent from the GUI via the Session Control to the Application Control which starts the Application by creating an Application object. The information of the associated template is recalled by the Data Base Control from the Data Base and the template is displayed by the GUI which sends a confirmation to the Application. After that, the user's writing permission is sent to the GUI and the user is notified about this permission. Then the user may write data to the template which is transferred to the application. The rest of the Application's behaviour depends on the Application.

8. CONCLUSION

We have presented a mechanism for modelling an existing experimental command and control information system by UML. For that purpose, we have used the software development tool Software trough Pictures from Aonix. The defined modelling process is a combination of automatic model generation by the reengineering component of Software through Pictures and manually constructed models. Furthermore, it consist of an integration of the two types of models and of modelling the dynamic behaviour of Applications by sequence diagrams.

Beside these diagrams the specific behaviour of the objects are illustrated by **state-transition diagrams** like the one in Figure 3. We omit the presentation of such a diagram, since it would be too detailed in the context of this paper. Nevertheless, by the detailed description in state-transition diagrams we are able to use a specific feature of Software through Pictures that automatically transforms the information in those diagrams beside the information of the class diagrams into code. Thus, beside its illustrative character, the models serve as a starting point in reengineering and extending INFIS by new Applications.

This will be another task, namely **forward engineering**. During the modelling process we also intend to produce a **use case model** that models the user requirements of the system. In the forward engineering process we start by extending this model for further use cases which describe the additional functionality of the system. Then a new Application will be modelled by defining its class diagram and its specific behaviour in a sequence diagram and a state-transition diagram. Then Software through Pictures produces a code framework from those diagrams which has to be extended manually.

The shortly described forward engineering mechanism shows the importance of the UML-modelling process, since the produced UML-models serve as a well-suited starting point for extending and redesigning the system as well as a documentation of the system for new software developers.

REFERENCES

[Aonix] http://www.aonix.com

[ATCCIS] *Army Tactical Command and Control Information System* (permanent), SHAPE Policy & Requirements Division, Mons (Belgium)

[Barnes, 1995] Barnes, J.: *Programming in Ada95.* Addison-Wesley, 1995.

[Chen, 1976] Chen, P.: *The Entity-Relationship Model – Toward a unified view of data*. ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976.

[MIP] http://www.dnd.ca/dlcspm/mip/index.html

[OMG] http://www.omg.org

[Rational] http://www.rational.com

[Yourdon & Constantine, 1989] Yourdon, E. & Constantine, L.L.: *Structured Design: Fundamentals of a Discipline of Computer Program and System Design*. Prentice Hall, 1989.

This page has been deliberately left blank

Page intentionnellement blanche

Natural Language Access for C4I Systems

Dr. Matthias Hecking
FGAN/FKIE
Neuenahrer Straße 20
53343 Wachtberg-Werthoven
Germany
hecking@fgan.de

1. Summary

There is no precise meaning of the term ‘interoperability’. In the proceedings of the conference *Multi-Lingual Interoperability in Speech Technology* (see [1], p. 133) the term is further specified. Interoperability includes aspects between systems, between people, between people and systems, and for different tasks. In this paper we deal with the aspect of interoperability between people and systems. The basic message is that the use of human language can improve the cooperation between people and systems. But there is another aspect of using human language in man computer interaction. New military NATO operations and growing coalitions demand for more complex C4I (Command, Control, Communications, Computers and Intelligence) systems. The development cycles of these systems are becoming shorter. This also means, that the users will have less time to learn how to use these systems. A natural way to communicate is to use natural language. If the C4I systems would have the ability to process spoken language, this would reduce the training curve for new systems and would simplify the usage of the systems.

Today, the usability of human language technology (HLT) is restricted to narrow and well defined application areas (domains). Another requirement is that the language must be restricted as well. This means, that the vocabulary and the grammatical structures must be limited enough such that processing time becomes acceptable. The military domain and the stereotyped military command language seem to be appropriated for using HLT.

In our project NATLAC (natural language access) we try to show that the available methods, techniques, and tools of computational linguistics are mature enough to look whether they are applicable to C4I systems for different purposes. Especially, the scientific progress in the field of speech recognition is promising.

In this paper, we will present the project NATLAC in more detail in section four and we will report about our experiences gained in the use of the speech recognizer. Prior in section two we give a short overview of the available information concerning military use of HLT and we mention those projects that use human language in C4I related areas. In section three possible applications of HLT in C4I systems are described.

2. The Military Use of Human Language Technology

In 1996 the NATO technical report *Potentials of Speech and Language Technology Systems for Military Use: an Application and Technology Oriented Survey* (see [3]) efficient speech communication was recognized as a critical capability in many military applications. The report classifies the military applications in six categories among other things ‘command and control’ and ‘computers and information access’. Most of the report deals with application and techniques of *speech recognition* (the transformation of the spoken words into a string of written words). Only a very small part of the report deals with *language processing*. This term comprises those technologies to understand what was spoken (given by the speech recognizer). Language processing includes syntactic, semantic, and dialogue analysis.

Other reports and proceedings of the NATO deal with speech processing in more detail. In *Multi-Lingual Interoperability in Speech Technology* (see [1]) the multi-lingual aspect of speech recognition is discussed.

Paper presented at the RTO IST Symposium on “Information Management Challenges in Achieving Coalition Interoperability”, held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

The report of *Databases for Assessment of Military Speech Technology Equipment* (see [4]) lists speech recordings made in military environments for research purposes and the report *The Impact of Speech Under "Stress" on Military Speech Technology* (see [5]) investigates in more detail the usability of speech technology under a specific aspect.

There are military systems in use, which contain HLT components. In [2] a cross-language automatic interpreting system is described which the NATO forces use. The system is able to recognize 4000 phrases in English (e.g. "I am a member of the NATO peacekeeping forces.") and to play the corresponding spoken phrase in another language through a loudspeaker. The system was developed with a COTS speech recognizer and is based on a standard portable PC. In this approach only a speech recognizer is used. The relation between the pairs of sentences in English and the other language is realized through simple mapping. No natural language processing technique is used. This is different in the *CommandTalk* system.

CommandTalk is a spoken-language interface to the ModSAF battlefield simulator (see [7], [8], [9]). The system was developed by SRI International. The simulation user can use ordinary spoken English to create forces (e.g. "Create an M1 platoon designated Charlie 4 5."), assign missions to forces, change missions during execution, and control all the functionality of the simulator program (e.g. "Center on M1 platoon"). The principal design goal of the CommandTalk system was to let the commanders interact by voice with simulated forces as if they would command actual forces. The development of the simulator is based on various components developed at SRI. As the speech recognizer component the Nuance system (see [11]) was used. The natural-language parsing and semantic interpretation is done by the Gemini subsystem. For the CommandTalk system an application-specific grammar and wordset was developed. Beside the information from the utterance the system uses linguistic context, situational context, and defaults to produce a complete interpretation of the utterance. The system is also able to handle dialogues and combinations of language and mouse input.

Operational C4I systems with HLT components are not in use. This opens a wide field of research.

3. Application of Human Language Technology to C4I Systems

There are different possible uses of HLT in C4I systems. In this section we try to classify these uses. We distinguish:

1. Recognition of speech to *control* a command and control system. This means, that another input modality in addition to the keyboard and mouse is available to initiate functions of the system. To realize this, the emphasis lies on speech oriented development. After identifying those functions that should be activated by voice, a natural-language command language must be designed and integrated into the speech recognizer module. Language processing techniques are not necessary.
2. Natural language *access* in spoken or written language to C4I databases. This would mean, that e.g. facts about military situations or availability of military forces could be inquired without using complicated menus and keyboard inputs. In this case after processing the speech the natural language processing techniques have to be used to identify the meaning of the inquiry.
3. Recognition of speech and subsequent language processing as a possibility to *input* data into the C4I system, e.g. the automatic processing of the audit message of an observation post.
4. Processing of spoken or written natural language input (e.g. radio messages or transmission, web pages) for keyword spotting or *information extraction*. This would deliver information relevant for use in the C4I system.

Note also, that for all uses of HLT the systems can use different language, so that multilingual approaches are possible.

4. The Project NATLAC

4.1 The Architecture

From the different possible uses of HLT in C4I systems we have chosen for further research in our research project NATLAC (**Na**tural **L**anguage **Ac**cess) the spoken access to C4I databases. The ATCCIS (see [10]) database delivers the domain model. As a scenario for our prototype system the planning of a multinational operation is used. In this scenario an action, action tasks, units, objectives, geographical points, reporting data, and contexts are used to model the situation. In the first step, we are realizing the natural language front-end that will be able to answer simple spoken questions concerning this situation, e.g., "Gehört das 9. Deutsche Batallion zu den verfügbaren Einheiten?" ("Does the 9th GE battalion belongs to the available units?"). The scenario is elaborated enough, so that more complicated language and domain problems can be modeled too in the future, e.g., complex questions or dialogues. The long-term objective of the project NATLAC is the construction of a dialogue system for a subset of spoken German referring to the scenario in the ATCCIS database.

The planned prototype is capable of recognizing spoken German. The architecture is shown in Fig. 1. For converting the acoustic input into a stream of words we will use a COTS product (the *speech recognizer*). At the moment we experiment with the Nuance (see [11]) system. But we also consider other products. The speech recognizer uses a *lexicon* and a *grammar* of the subset of the German language. In addition, the recognizer needs an acoustic model. In this model each written word is associated with its pronunciation given as a sequence of phonemes. This acoustic model can be extended to those words not included in the scope of supply.

The recognized words are then analyzed through linguistic means. First the input is *syntactically analyzed* with the help of a parser, a lexicon, and a grammar. The grammar and the lexicon determine the ability of the natural language component. The result of the syntax analysis is a *parse tree*. This parse tree contains all available syntax information that can be determined only from the input sentence. No other sources are used.

Then the parse tree is *semantically analyzed* to build a semantic representation of the stream of words. To realize this the semantic analysis component uses the *context* of the dialogue to resolve ambiguities e.g. to decide which platoon is meant in the sentence “The platoon attacks hill 234”. The *ontology* component and the *semantic* component provide semantic information e.g. of how concepts of the domain are interrelated (e.g. a tank is an armored vehicle).

Finally, the semantic representation is used as the foundation for the access to the ATCCIS database. The DB access component constructs from the semantic representation an SQL statement that is dispatched to the underlying database system. The database delivers the result, and it is presented to the user as spoken output.

To give NATLAC dialogue capabilities a *dialogue analysis* component is necessary. This component is capable of identifying individual dialogues and dialogue steps. This is the foundation for the capability of the system to become active by itself and to control the flow of the dialogue (e.g. to insert a clarification dialogue).

All components of the system are coordinated through the system kernel. Most of the components will be realized through Prolog (see [12]).

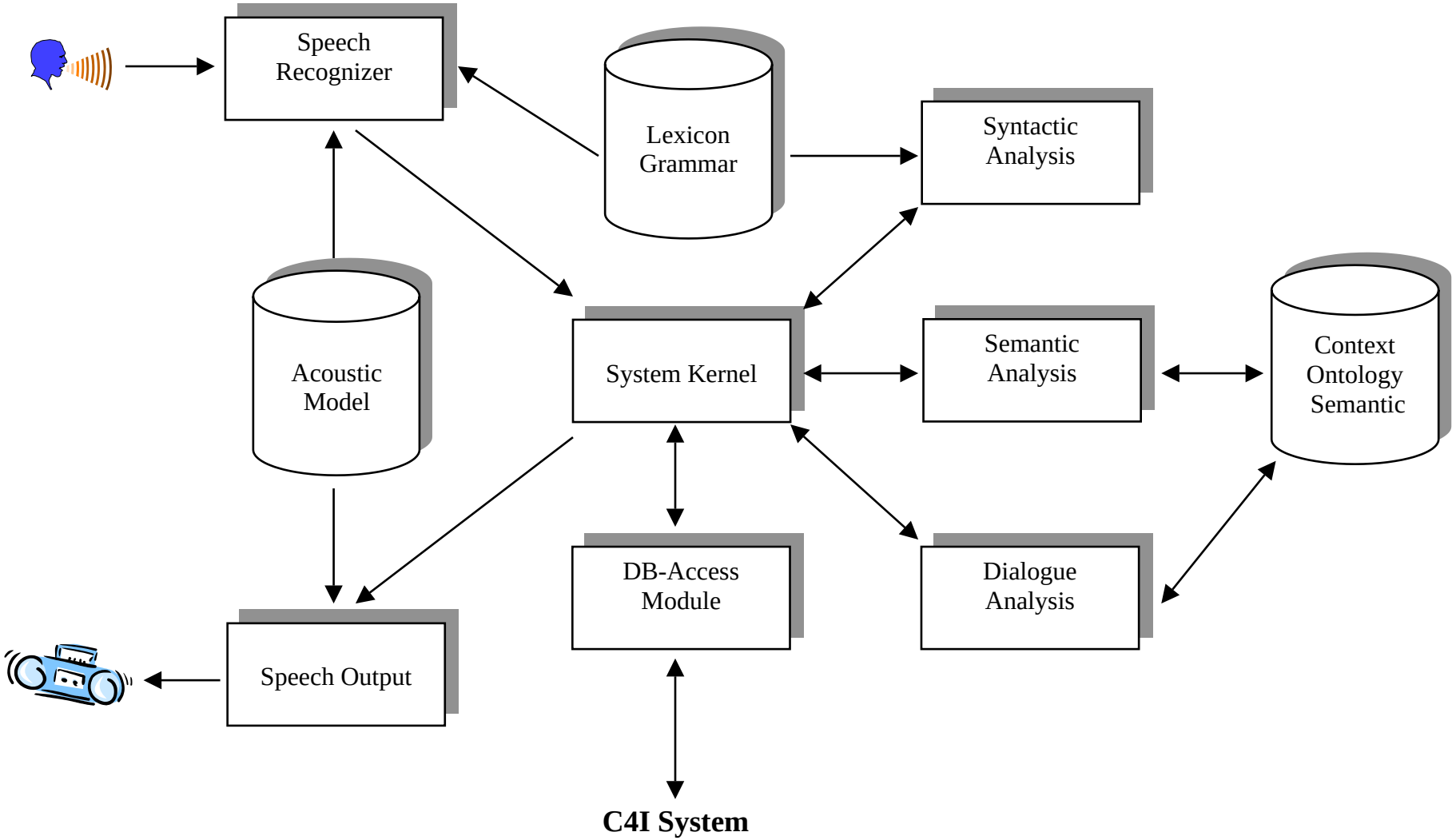


Fig. 1: Architecture of the NATLAC System

4.2 Experiences with the Speech Recognizer

Speech recognizers convert the spoken words into a sequence of written words. To do this, they need a lexicon and for each word in the lexicon the pronunciation. After recording the spoken words the recognizer tries to map the recorded sounds to the stored pronunciation. We use the Nuance system (see [11]) as the speech recognizer. For the mapping the Nuance system uses probabilities of sequences of phonemes and probabilities of sequences of words. The latter one is computed from a grammar that must be specified by the developer. The developer also has to supplement the vocabulary for those words and respective pronunciation that are not part of the delivered lexicon. We developed a small grammar that describes the syntactical correct simple statements and questions in German. Fig. 2 shows the top-level part of the grammar.

```
.SATZ
        [
        (
            ?ADV
            [
                (   ; yes/no-questions
                    V ?ADV NP
                )
                (   ; statements
                    NP ?ADV V
                )
            ]
            ?ADV
            ?PRO_REFL
            ?ADV
            [
                (   ?[NP PP AP] ?ADV )
                (   NP ?ADV ?[NP PP AP] ?ADV )
                (   AP ?ADV PP ?ADV )
                (   PP ?ADV PP ?ADV )
            ]
            ?[
                ( ?zu INF)
                PA2
                VERB_PRE
            ]
            ?ADV
        )
        ]
```

Fig. 2: Top-Level Grammar

The depicted rule describes yes/no-questions and simple statements. .SATZ is the top-level non-terminal symbol. It represents the whole sentence. The other abbreviations (non-terminal symbols) represent parts of sentences (NP = nominal phrase, PP = prepositional phrase, AP = adjective phrase) or word classes (ADV = adverbs, V = verbs, PRO_REFL = reflexive pronouns, PA2 = perfect participle, VERB_PRE = verb prefixes). '?' means the part is optional, '[ ]' means choose one, and '( )' means sequence in this order. The non-terminals are further defined through other non-terminals and/or terminal symbols, i.e. the single words. Fig. 3 shows several rules. The first simply divides the adverbs (ADV) into sub-classes (ADV_LOK, ADV_DIR,...). The other rules are lexical rules. They associate the non-terminals with the appropriate German adverbs (e.g. 'da', 'dort', 'hier' are adverbs of location).

```
ADV [ ADV_LOK ADV_DIR ADV_TMP ADV_MOD ADV_KONJ ADV_STELL
      ADV_INT_DIR ADV_INT_TMP ADV_INT_MOD ADV_INT_KONJ
      NEG_PART ]

ADV_LOK        ADV_TMP                          ADV_INT_DIR
    [              [                                [
    da             bald          neulich            wo
    dort           bisher        nun                woher
    hier           bislang       soeben             wohin
    ]              danach        sofort             ]
                   eben          unlängst
ADV_DIR            einst         vorhin
    [              gerade        zeitlebens
    fort           gestern       zugleich
    her            heute         zuletzt
    herum          immer         ]
    hin            jetzt
    weg            morgen
    ]
```

Fig. 3: Lexical Rules

To ensure that the recognizer achieves a high recognition rate, the developer has to extend the lexicon with those words not part of the delivered vocabulary. This concerns primarily compound nouns. Fig. 4 shows several examples. In the left column the German words are shown. The right column contains the pronunciation of the words. The pronunciation is given through the CPA (Computer Phonetic Alphabet).

panzergrenadiere	p A n t s * r g r e n A d i r *
panzerhaubitze	p A n t s * r h aw b I t s *
panzerhaubitzen	p A n t s * r h aw b I t s * n
panzerlehrbrigade	p A n t s * r l e r b r i g A: d *
panzerlehrbrigaden	p A n t s * r l e r b r i g A: d * n
panzerpionierkompanie	p A n t s * r p i o n i r k O m p A n i
panzerpionierkompanien	p A n t s * r p i o n i r k O m p A n i * n
panzerschnellbrücke	p A n t s * r S n E l b r Y k *
panzerschnellbrücken	p A n t s * r S n E l b r Y k * n
pionierbrigade	p i o n i r b r i g A: d *
pionierbrigaden	p i o n i r b r i g A: d * n
pionierlehrbrigade	p i o n i r l e r b r i g A: d *
pionierlehrbrigaden	p i o n i r l e r b r i g A: d * n
raketenartillerie	r A k e t * n A r t I l * r i
raketenartilleriebataillon	r A k e t * n A r t I l * r i b A t A l j O n
raketenartilleriebataillone	r A k e t * n A r t I l * r i b A t A l j O n *
raketenwerfer	r A k e t * n v E r f * r

Fig. 4: Pronunciation of Unknown Words

The specified grammar rules together with the extended lexicon is used by the speech recognition engine to transform the spoken words into written form. Fig. 5 shows an example output of the recognizer. The spoken sentence which should be recognized is “der panzer überrollt die stellung” (“the tank overruns the position”). The recognizer produces three hypotheses and associated confidence scores. Hypothesis no. 1 is the correct one. Both other hypotheses are incorrect.

Recognition errors can have various reasons:

- The spoken sound deviates from the stored sequence of phonemes. This can be due to, e.g. a dialect, a cold, background noise, or simply improper pronunciation.
- There are words that don't have a proper stored pronunciation. This affects those words missing in the lexicon.
- The specified grammar overgenerates, i.e. the grammar accepts more sentence as syntactically correct as should be.
- The grammar is too complicated so that too many phonetic alternatives are possible.

```
Grammar: .SATZ
Transcription: der panzer überrollt die stellung
Result #0:    der panzer überrollt tisch stellung (conf: 76, NL conf:  0)
Result #1:    der panzer überrollt die stellung (conf: 76, NL conf:  0)
Result #2:    schwer panzer überrollt tisch stellung (conf: 74, NL conf:  0)
Total Audio: 2.55 sec
...
```

Fig. 5: Example Output of the Speech Recognizer

To cope with the recognition errors the NATLAC system must be able to identify the correct hypothesis and to rule out the incorrect ones. This can only be done by the linguistic means. In a first step the syntax analysis must identify for each recognized word the word class (i.e. whether it is an adverb, noun, verb, ...) and all morphological information (e.g. tense, gender, number, case, ...). Then, these information are used by the parser which tries to build up the parse tree. This can only be done, if the morphological information for each word gives a consistent parse tree. If the constructed parse tree contains an inconsistency, the hypothesis is rejected. Also, semantic information can be used to rule out improper recognition hypotheses.

The needed parser must also be able to process ill-formed speech input, i.e. phenomena like false starts, repetitions, hesitations, fragmentary utterances, and corrections.

5. Conclusion

Interoperability also pertains to interaction between people and systems. Spoken human language can improve the interaction between people and systems. And that might be true for C4I systems. These systems offer restricted and well defined military domains and the military command language which is used together with these systems is restricted enough so that human language technology might be usable. In our project NATLAC (natural language access) we try to show that the available means of computational linguistics are mature enough to look whether they are applicable to C4I systems. In the project, we have chosen the spoken access to C4I databases as the research topic. The long-term objective of the project NATLAC is the construction of a dialogue system for a subset of spoken German referring to a scenario in the ATCCIS database. For this purpose, we have designed a general architecture we are about to realize the first component, the speech recognizer. In a first step, we have developed a grammar for simple German statements and questions and we have extended the vocabulary. The speech recognizer produces several hypotheses. To identify the correct hypothesis other components are necessary. Especially, linguistic means are needed.

REFERENCES

[1] NATO IST Panel. *Multi-Lingual Interoperability in Speech Technology*. Cedex: RTO/NATO, RTO Meeting Proceedings 28, RTO-MP-28, AC/323(IST)TP/4, 2000.

[2] M. Hunt et al. *A Military Operational Automatic Interpreting System*. In: [1], pp.87-90.

[3] Steeneken, H. J. M. *Potentials of Speech and Language Technology Systems for Military Use: an Application and Technology Oriented Survey*. NATO, Technical Report, AC/243(Panel 3)TP/21, 1996.

[4] NATO IST Panel. *Databases for Assessment of Military Speech Technology Equipment.* Cedex: RTO/NATO, RTO Technical Report 25, RTO-TR-25, AC/323(IST)TP/6, 2000.

[5] NATO IST Panel. *The Impact of Speech Under "Stress" on Military Speech Technology*. Cedex: RTO/NATO, RTO Technical Report 10, RTO-TR-10, AC/323(IST)TP/5, 2000.

[6] Shuttic, G. and George, B. *Robust Speech Recognition in Tactical Communications Environments*. The MITRE Corporation, http://www.mitre.org/technology/mtp00/human_language_2000.shtml#robust_speech.

[7] *CommandTalk*. SRI International, http://www.ai.sri.com/natural-language/projects/arpa-sls/commandtalk.html.

[8] Moore, R. et al. *CommandTalk: A Spoken-Language Interface for Battlefield Simulations*. In: Proc. of the 5th Conf. on Applied Natural Language Processing, Washington, DC, pp. 1-7, ACL, 1997.

[9] Stent, A. et al. *The CommandTalk Spoken Dialogue System*. In: Proc. of the 37th Annual Meeting of the ACL, pp. 183-190, University of Maryland, College Park, MD, ACL, 1999.

[10] NATO. *The Land C2 Information Exchange Data Model*. AdatP-32 Edition 2.0, 31 March 2000.

[11] http://www.nuance.com

[12] Intelligent Systems Laboratory. *SICStus Prolog User's Manual*. Swedish Institute of Computer Science, Kista, Sweden, 2000, http://www.sics.se/sicstus.

Ontologies for Coalition Interoperability

Anne-Claire Boury-Brisset
Decision Support Technology Section
Defence Research Establishment Valcartier
2459 Pie-XI North, Val-Belair, QC
G3J 1X5, Canada
Email: Anne-Claire.Boury@drev.dnd.ca

Abstract
Future command and control information systems will have to take into account interoperability issues so that information can be effectively shared and exploited within coalition operations. In this context, interactions between participants require mechanisms to facilitate the exchange of information and provide a shared understanding of the situation based at least on a commonly agreed terminology. One solution to facilitate the communication between agents is to build a common ontology that represents a shared model of a domain. In this paper, we show the role of ontologies in coalition environments, we present methods and tools for collaborative ontology construction and describe how ontologies can facilitate interoperability between heterogeneous information sources.

1 Introduction

Coalition operations are going to become an increasingly important feature in future years. Therefore, it will be necessary to provide commanders with access to timely and relevant information from disperse and heterogeneous information sources for conducting their operations. In coalition contexts, there is a mix of equipments, operational procedures, and computer systems involved. To deal with this heterogeneity, future command and control information systems will have to address interoperability issues so that coalition information can be effectively shared and exploited. The problem of interoperability between systems that exists within an organization is more challenging across organizations with different national doctrines. For example, one problem is that organizations often refer to the same concepts using different names. Another problem is the use of different data units between countries that can be confusing when exchanging data. To this end, interactions between coalition participants require mechanisms to facilitate the exchange of information and provide a shared understanding of the domain based at least on a commonly agreed terminology.

A modern command and control system must provide commanders at all levels of command with the capability to share a common view of the battlespace. The Common Operational Picture (COP) is the capability that will provide the Commander with the degree of situational awareness required to direct military operations. Development of improved situation awareness or common operational picture (COP) capabilities remains a priority within nations. Consequently there are several COP initiatives underway. The objective of the CINC21 (Commander in Chief 21st Century) Coalition effort is to define and to conduct a set of multinational coalition command and control related experiments to advance the state of knowledge and contribute to the interoperability of future coalition operations. A four-nation collaboration (C-CINC21) between Australia, United-Kingdom, United States and Canada, has been established to address key interoperability issues that will have an impact on technologies supporting Canada's future COP. In this context, the four nations have defined a number of experiments with the objective of providing interoperability across nations involved within the collaboration efforts and enabling shared situational awareness. Some of the objectives of this initiative are to experiment with innovative knowledge management concepts and tools as well as to develop ontologies for coalition interoperability.

In the context of coalition operations, we consider ontologies as a key component to provide a shared understanding of a domain and facilitate knowledge level interoperability among heterogeneous information

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

sources. In this paper, we present concepts and tools related to the representation, building and exploitation of ontologies for coalition operations. The remainder of the paper is organized as follows. In section 2, we introduce the research field of ontologies and their role for coalition operations. Then, we present the main ontology description languages and emerging representation standards that could be suited for ontologies. An approach for collaborative ontology construction in a distributed context is described. Finally, we succinctly present information exchange services that could be provided to communicate information between nations.

2 Ontology engineering

2.1 Definition and role

During the last few years, the concept of ontology has attracted much attention within the artificial intelligence (AI) community [Uschold and Gruninger, 1996] because of its ability to formalize domain concepts and to facilitate information exchange among application programs.
According to AI researchers, an ontology is defined as a formal and explicit representation of a conceptualization [Gruber 1993]. It allows one to represent, in a more or less formal way, concepts of a domain of concern and their relations. An ontology is described using attributes, properties, relations between concepts, and eventually constraints and axioms. An ontology can be used to provide a formal and shared understanding of some domain, facilitating exploitation by both human agents or computer programs.

Ontologies can be classified into different categories, ranging from general domain-independent knowledge to domain-specific knowledge.

- *Representation or meta-ontologies* conceptualize knowledge representation formalisms.
- *Upper-level ontologies* define general-level descriptive terms that form the foundation for knowledge representation. For example, *space*, *time* or *object* are domain independent terms that apply to all domains.
- *Task ontologies* help specify activities or business processes. Concepts such as *activity*, *resource*, *role* are general terms that can be used to describe any activity.
- *Domain ontologies* represent specific knowledge concepts. For example, *weapon* or *missile* are specific terms of the military domain.

Different types of ontologies have to be specified and agreed among coalition participants to facilitate information exchange. Domain ontologies formalize the relevant concepts of the military domain (ex. Logistics). At a meta level, ontologies of time and space are particularly important in military domains because reasoning about time and space requires a formal means to describe spatial and temporal entities. Furthermore, ontological standards have to be established to provide a meta-description of information sources, for example it is necessary to agree on a metadata set that should be used to describe geospatial information.

Ontologies can be used for different purposes.

- Ontologies facilitate communication between knowledge workers and systems.
- Ontologies enable to improve the engineering of knowledge-based systems by providing the basis for domain knowledge representation.
- Ontologies facilitate information integration and interoperability between heterogeneous knowledge sources. Information agents make use of ontologies to enable access to heterogeneous knowledge sources.
- Ontologies can be exploited to index and access semi-structured information sources. They facilitate information retrieval over collections of heterogeneous and distributed information sources. Especially, Internet search engines need domain ontologies to organize information and guide search processes.
- In the natural language understanding domain, ontologies provide the basis for domain knowledge representation and help identify the semantic categories that are involved in understanding discourse in that domain.

2.2 Examples of ontologies

Many concrete ontologies have been developed for several years, for example the large ontology CYC for human knowledge modeling, the linguistic ontology WordNet, the Enterprise and TOVE ontologies in the

domain of enterprise modeling, UMLS for the medical domain and engineering ontologies such as EngMath, PhysSys.

More recently, the concept of ontologies has become a popular subject in fields such as knowledge management and information retrieval on the Internet, due to the overload of unstructured information and the difficulty for keyword-based search engines to perform semantics search. In particular, an increasingly important area for ontologies is electronic commerce [McGuinness, 2001] where there is a need of standards for the exchange of information. Communicating applications must have a shared understanding of a domain and intelligent agents must agree on a common ontology to be able to communicate effectively. For example, e-commerce agents must adopt common ontologies, they must agree on terms such that product, transaction, if they are to interact without misunderstanding.

In the military domain, the importance of ontologies has been recognized for years. Common representation of plans has been a subject of interest for a long time. The ARPA/Rome Laboratory Planning Initiative (ARPI) led to the creation of the KRSL plan language. Later, as part of the O-Plan project, A. Tate [Tate, 96] proposed a structure for a plan ontology using new insights gained in the knowledge-sharing community in the US and Europe.
As part of the DARPA Joint Forces Air Component Commander (JFACC) program, an ontology for air campaign planning has been built to represent a wide variety of knowledge content in the air campaign domain [Valente *et al*, 1999]. The objectives of this work were to integrate knowledge acquisition and modeling efforts from developed knowledge-based applications, to create a repository for general knowledge about air campaign to use in several applications and to facilitate interoperation and communication between systems with a shared terminology. The JFACC ontology is represented in the knowledge-representation framework LOOM, based on description logics and is currently being applied in three applications that make use of LOOM's representation and reasoning facilities.

The DARPA HPKB (High Performance Knowledge Base) project [Cohen et al, 98] promotes technologies for developing very large, flexible and reusable ontologies and knowledge bases. In the context of the *crisis management* challenge problem, two knowledge-based systems were developed to answer questions about international crises such as "what will the US response be if IRAN closes the Strait of Hormuz ?"

In our work in the domain of search and rescue (SAR) at Defence Research Establishment Valcartier (DREV), we have built an ontology to support SAR activities. The role of the ontology is to provide a better structuring of SAR knowledge and help index and retrieve information in domain-related documents [Boury-Brisset, 2000]. The SAR ontology is composed of several sub-ontologies related for example to pilot qualifications and experiences, SAR resources and equipment, environmental and weather factors.

3 Ontology specification languages

3.1 Traditional specification languages

The formality of the language chosen for representing an ontology is dependent on the degree of automation in the various tasks that the ontology is supporting. « *If the ontology is a framework for communication among people, the representation can be informal … If the ontology is to be used by software tools or intelligent agents, then the semantics of the ontology must be made much more precise* » [Uschold and Gruninger, 96].

Different formalisms and knowledge representation languages have been proposed to describe ontologies. Some are limited to describing concepts, attributes, and relations and resemble conceptual models in databases or object-oriented models (ex. UML class models). They describe models of concepts and their instances in taxonomies, for example, military entities such as vehicles, ships, aircraft or missiles. Others use knowledge representation paradigms such as first-order predicate logic, frame-based or description logic. They are, therefore, more sophisticated languages that support inferences. In this category, the main formalisms are the following:

- Ontolingua is a language proposed to construct portable ontologies using KIF (Knowledge Interchange Format), a formalism based on first-order predicate calculus designed to facilitate the exchange of

knowledge between heterogeneous languages. Ontolingua is also based on the Frame Ontology that allows the specification of ontologies following the paradigm of frames (using class, instances, subclass-of terms).

- LOOM, a descendent of the KL-ONE language, is a knowledge description framework based on description logics that integrates an efficient automatic classifier.
- Flogic (Frame Logic) is a language that integrates frame-based features and first-order predicate calculus.

3.2 XML-based languages

The exchange of information across the Web and the cooperation among heterogeneous agents have led to the development of a set of specification languages based on Web standards such as XML and RDF [Harmelen and Fensel, 1999]. Several XML-based formalisms could be used for describing ontologies. We present some of these formalisms, their knowledge representation capabilities and limitations.

3.2.1 XML / XML Schemas

XML (Extensible Markup Language) [Bray et al 1998], the metalanguage developed by the World Wide Web Consortium (W3C) for information structuring and exchange on the Web, seems *a priori* to be a good candidate to describe ontologies in a distributed environment. XML allows the definition of customized markup languages with application-specific tags, e.g., <QUANTITY> or <SPEED>, for representing information in particular application domains and defining data structures. XML Document Type Definitions (DTDs) provide a way to explicitly declare the tag sets and their structure, to be used in particular units of data. The advantages of XML are that it has a human-readable and well-defined syntax. Furthermore, there exist software tools for parsing and manipulating XML. However, even if XML allows the specification of user-defined tags, it does not provide the semantics required for an ontology. What does the <RESOURCE> tag mean? In which unit is a <SPEED> concept represented? Furthermore, even if DTDs define the legal nestings of tags in a document, it does not represent the notion of an ontological class hierarchy. So, the inheritance mechanism is missing.
The description of ontologies requires ways to explicitly specify relations between concepts, hierarchies of concepts, in order to offer more expressiveness. So, new proposals are emerging to address this aspect on top of the XML language, for example XML and RDF schemas. These meta-models should be considered for the construction of coalition ontologies.

Due to several DTDs limitations, in particular DTDs are not adequate for describing data contents, XML Schemas is a new W3C proposal aimed to replace DTDs. XML schemas are built in XML, provide data types as well as relationships between elements, and support namespaces. However, XML shcemas present limitations for object-based knowledge representation, in particular the lack of inheritance that is necessary for ontology representation.

3.2.2 RDF / RDFS

RDF (Resource Description Framework) [Lassila and Swick 1999] is a W3C proposed recommendation that provides a means for adding semantics to a document without making any assumptions about the structure of the document. It provides a model for representing metadata (including ontology-like information) for Web resources in XML in order to efficiently access relevant information. The RDF data model that consists of triples (resource, property, statement) does not provide mechanisms for defining relationships between properties and resources. RDFS (RDF Schema) [Brickley and Guha, 2000] is a declarative language designed for specifying attributes and their corresponding semantics. Using RDFS, objects, classes and properties can be described. Predefined properties can be used to model *instance-of* and *subclass-of* relationships as well as domain restrictions and range restrictions of attributes.
In regard to ontologies, even if RDFS is less expressive than frame-based or predicate calculus languages (ex. the definition of axioms is not possible), RDF provides two important contributions: a standardized syntax for writing ontologies and a standard set of modeling primitives (like instance-of and subclass-of relationships).

The choice of one particular specification language for modeling ontologies depends on the task they are designed for. For representing taxonomies of concepts or for the interchange of ontologies on the Web, XML-based languages are well suited. In contexts requiring higher expressiveness, logic-based ontology specification languages are more appropriate. In the coalition context, interoperability issues between command and control information systems favour the use of XML-based standards for the final encoding of our ontologies.

4 Ontology construction tools

Ontology development tools relying on the knowledge representation languages presented above (section 3.1) have been built to facilitate the construction of ontologies, and new ones are continuing to emerge. These environments provide functions to browse, create, edit, modify, and use ontologies. Among these tools, some are standalone applications (ex. Protégé) whereas others enable the collaborative construction of ontologies (ex. Ontolingua Server, WebOnto). An evaluation of the most relevant tools implemented is described in [Duineveld *et al*, 99]. The tools considered in this study are:

- Ontolingua Server [Farquhar *et al*, 96] developed at Stanford University (KSL);
- WebOnto developed at Knowledge Media Institute of the Open University;
- Protégé 2000 [Roy *et al*, 2000] developed by Stanford's Medical Informatics Section;
- OntoSaurus, developed by ISX Corporation from the LOOM project at University of Southern California's information Sciences Institute;
- ODE (Ontological Design Environment) developed at LAI of the Technical University of Madrid.

We summarize in the table below the characteristics of these tools based on Duineveld's study and some more recent aspects of the tools described in publications.

Evaluation criteria / tools	**Ontolingua Server**	**WebOnto**	**Protégé 2000**	**OntoSaurus**	**ODE**
High-level primitives	Primitives from the Frame Ontology	Language OCML. Many primitives available from a base-ontology	OKBC knowledge model: concepts, attributes, properties, sub-class relation.	Representation power of the LOOM language	Conceptual level: concepts, relations, axioms, subclass-of, disjointness, exhaustivity partition.
Reuse ontologies from libraries	Important repository of ontologies	Many ontologies. Possibility of reuse the base-ontology	Not available	Yes	Examples only
Interface (clarity, consistency)	Good graphical interface (browsing) Difficult to use	Very good Clear	Good interface: clear and easy to use. Customizable layout	Ontology browsing mainly. Poor editing function	Not available for evaluation
Collaborative functions	Best collaborative tool. Notion of users, groups. Read/Write access Notification mechanism	Synchronous cooperation: lock function. Mode broadcast / receive	Not a collaborative tool	Lock function for collaborative editing	Asynchronous cooperation only
Web Server or local installation	Web Server	Web Server	Local installation	Web browser or local installation	Local installation
Import / Export	Many formalisms: CLIPS, LOOM, Epikit, CORBA's IDL, KIF	No export	JDBC Database RDF	Export in Ontolingua, KIF, IDL.	Export: Ontolingua, F-Logic

Table 1: Comparison of Ontological Engineering Tools

5 Ontology development methodologies

The development of ontologies is a modeling activity that is usually carried out by ontological engineers (also called ontologists) that have sufficient understanding of the domain of concern and are facile with knowledge representation languages. As for the design of knowledge-based systems, the help of experts of the domain is required to build and validate ontologies.
Ontology construction is a complex and time-consuming activity. Therefore, methodologies have emerged based on the experiences gained in the construction of large ontologies. These ones aim at making the development of ontologies more an engineering process rather than an art. A recent survey of ontology development methodologies is presented in [Jones et al, 98]. The more comprehensive methodologies for ontology development are summarized in Table 2. Except the IDEF5 ontology capture method [KBSI, 94] that is part of the complete and mature enterprise modeling methodology IDEF (Integrated Computer Aided Manufacturing DEFinition), the other methodologies result from important projects aiming at constructing large ontologies:

- The TOVE (Toronto Virtual Enterprise) project [Uschold andGruninger, 96],
- The Enterprise ontology [Uschold and Gruninger, 96],
- The Methotonlogy method [Fernandez et al, 97].

The main stages that can be derived from these methodologies consist of the following:

- Identification of the task for which the ontology is being developed;
- Definition of the requirements for the ontology: purpose and scope;
- Informal specification: Build informal specification of concepts;
- Encoding: Formally represent the concepts and axioms in a language;
- Evaluation of the ontology.

The extensibility and maintenance of ontologies are considered by most of the authors but is not part of the methodologies.

TOVE	Enterprise	Methontology	IDEF5
Motivating scenarios (set of problems)	Purpose: level of formality	Purpose and scope	Purpose: objectives and requirements
Requirements (informal competency questions)	Scope: range of information i.e. list of relevant concepts	Knowledge acquisition through analysis of texts and expert interviews	Scope: boundaries of the ontology Data collection using knowledge acquisition techniques.
Formal specification of objects, attributes and relations	Formal definition of terms	Conceptualization: specification of concepts, instances, relations using an informal representation	Data analysis : produce a list of objects.
Formal requirements (formal competency questions)		Integration from other ontologies Formal implementation in a representation language	Initial ontology development: definition of “proto-concepts” in a schematic language
Axiom specification	Formal specification of axioms		Ontology refinement into a more structured language based on KIF.
Evaluation	Evaluation	Evaluation + documentation	Test and validation using actual data.

Table 2: Comparison of methodologies for ontology construction

In [McGuinness, 2000], D. McGuinness proposes a set of general guidelines for the construction of ontologies in distributed environments. We provide some of these guidelines that are particularly relevant in the context of coalition environments due to heterogeneity.

- Articulate the expected uses of the ontology and user profiles;
- Use a controlled vocabulary that is familiar to users;
- Specify mappings between multiple standard controlled vocabularies;
- Allow for users extensibility of the mappings and support users in adding new synonyms into a thesaurus;
- Specify semantics of terms and provide semantic retrieval;
- Provide partition extensibility;
- Given that objects can have many properties and only a few are useful for a particular use, provide mechanisms to help users focus their attention on objects that is relevant to them.

6 Collaborative ontology construction for coalition

As mentioned by D.L. McGuinness, it is becoming increasingly common for ontologies to be developed in distributed environments by authors with disparate backgrounds. In this context, protocols for distributed ontology generation and maintenance are required.

6.1 High-level protocol for collaborative ontology construction

In our context, researchers participating in the ontology construction for the C-CINC21 initiative are distributed geographically in different countries and have different backgrounds and military doctrines in mind. Furthermore, they have consolidated various experiences in military domains through different projects.
To maximize the exploitation of these disparate experiences and backgrounds while following a rigorous methodology, we are adopting a high-level protocol for the collaborative building of ontologies. This includes stages where participants are involved as a team and stages where participants work individually.
The first stage consists of specifying a set of areas that are relevant to be captured in an ontology to support coalition operations. The selected areas include in particular the description of country profiles, aspects of logistics, orders of battle. An interesting source of information for building the country profiles ontology is the CIA World Fact Book [CIA WFB]. For each area, a leader is responsible for the construction of the ontology based on his experience related to the topic (e.g. modeling activities, system design) in collaboration with participants of other countries. This aims to maximize knowledge reuse in order not to build ontologies from scratch when there exists some work already done in another context. This is a recommendation from most of ontology construction methodologies, given that ontology building is a time-demanding activity. In addition to military domain ontologies, support ontologies are also identified in order to represent temporal and spatial information as well as heterogeneous resources. In a next stage, participants should agree on the formalism required to build and exchange the ontologies and eventually on a tool that would facilitate ontology construction. The formalism should be kept at a conceptual level to facilitate communication between people (validation of models by military people).

To summarize, the guidelines for the collaborative tasks are the following:

- Identify sub-domains to be captured in the ontology. This includes:
  - application ontology,
  - support ontology (representation of time/space, resource, etc);
- Identify a leader/nation responsible for each identified domain
- Determine the formalism and the tool that will be used to capture and exchange the ontology.

The development of the ontologies is the next activity. Even if this is a collaborative process, we can describe the tasks conducted by ontologist leaders as individual ones. The process follows the main stages of most ontology development methodologies. The guidelines for these tasks are the following:

- Define the basic terms in the ontology and provide definitions in natural language to remove ambiguity. Definitions of terms have to be agreed among participating nations and validated with military people.
- Build a preliminary ontology: formally specify the semantics of the concepts by describing their properties and relationships with other concepts.
- Publish the ontology to get comments from other nations;
- Integrate comments from other nations;
- Validate the final model with other nations;
- Encode the ontology in the chosen language.

Ontology construction in this context should be a collaborative and iterative process supported by tools. In the next section, we propose an approach and a tool to address the collaborative aspect of ontology building.

6.2 Collaborative ontology building and critiquing

Ontology construction tools described in section 4 provide the functions required for editing, browsing and visualize ontologies through user-friendly interfaces. Most of them provide synchronous cooperation by implementing lock functions and define groups of users with different Read-Write access. However, in a distributed environment, it becomes important for different ontologists to have a workplace to put their comments about parts of ontologies that are built by other people. In this context, we have introduced the concept of *ontology critiquing* as part of this work in order to enable discussions about modeling decisions. For example, one person shall question whether if a new concept (class) should be defined as a subclass of an existing one or as an instance of a class with its own properties. In another case, one has to decide whether if a proposed class should be divided into two distinct classes in order to better reflect the world being modeled.
Therefore, in the perspective of providing a powerful collaborative ontology building tool, concepts such as *issues* or *decisions* related to the concepts proposed should be associated to the ontology meta-model. The tool should also enable to indicate (or automatically identify) the person who suggested or made a change in the ontology. Furthermore, during ontology construction, it would be interesting to associate with the concepts being defined some examples of the *related questions* that the ontology should help answer. This should also be added to the ontology description and should help validate the purpose and scope of the ontology.

To satisfy these requirements, we are currently experimenting with the building of ontologies using the Teximus tool [Teximus] that offers much flexibility. Teximus is a Web-based environment for the modeling of any domain content that provides the automatic generation of the Web pages presenting this domain. One of the strengths of Teximus is that it allows users to define the meta-model needed for their application. In our context, we can create our own meta-model for ontology specification (using concepts, attributes, relations) and for ontology critiquing (by adding issues, decision and related-questions properties to concepts).
Using such a tool, each ontologist leader contributing to the ontology development process could build the ontology he is responsible for and get comments from other people in an efficient manner. Any participants in the ontology construction process could browse the ontologies being built by people from other nations and give their comments and input about the ontologies (critique, suggest modifications, propose new concepts, etc.).

7 Information exchange services

As mentioned in the introduction, one of the objectives of the project is to provide an infrastructure and various services for information management within coalition environments built on top of the ontologies, in order to help participants in the coalition to get improved situational awareness. The information services that could be provided include: semantic search and information retrieval based on ontologies among heterogeneous information sources, publishing of information, advertising of new publications, etc.
The information services provided by each nation should communicate with one another using a predefined communication language. For that purpose, the format of the exchanged messages between the information services should be specified in order that the coalition information service attached to each nation is able to

interpret them. Each national coalition information service should implement the services by exploiting the ontology and link it to its various information sources and systems. So, the domain models provided by ontologies can be exploited to provide intelligent information retrieval among heterogeneous sources.

8 Related work

Besides this project, similar efforts are ongoing. CoAx (Coalition Agent Experiment) is a DARPA-AFRL-DERA programme [CoAx] conducted by several partners from universities and military organizations. The project aims to demonstrate how advanced software agent technology can provide an infrastructure able to support the demanding information and command and control requirements of a coalition force. The project addresses a number of problems including dealing with the need to share systems and information, providing mechanisms to translate information between systems, dealing with different levels of trust, etc.
In a more general context, the DARPA CoABS project (Control of Agent based Systems) [CoABS] aims at building a framework for the run-time integration of heterogeneous multi-agent and legacy systems. It is designed to meet the challenges of the military environment, as well as address the heterogeneity among the participating agent research communities.

DAML (DARPA Agent Markup Language) [DAML][Hendler and McGuinness 2000] is a recent DARPA project that aims at exploiting emerging technologies to enable agent to dynamically identify and understand information sources and provide interoperability between agents in a semantic manner. The identified stages in this project are to create an Agent Markup Language built upon XML, create tools that embed DAML markup on Web and other information sources, build agent-based programs that use it, and experiment them with military-specific problems.

9 Conclusion

Even if this project is at a preliminary stage and the development of ontologies is just starting, we have already learned some lessons or anticipated some difficulties. The challenge of building shared ontologies resides in the differences in culture, doctrine and backgrounds of the ontology builders. The proposed protocol should help participants agree on most important concepts. Whenever it is not possible, individual nations should provide mechanisms to solve the differences that cannot be resolved collaboratively. Moreover, solutions must be provided for the integration of different ontologies in a coalition context, possibly exploiting web standards. In particular, D. McGuinness provides some guidelines related to ontology merging. If merging small ontologies may not be too difficult, it becomes more difficult to provide tool support for larger ontologies. In any case, the problem is to determine terms that should be merged (objects with the same name and same semantics), and terms that may not be merged (terms that are disjoint by their definition). Furthermore, relationships between concepts should be analyzed when merging ontologies.
Another problem encountered up to now is the difficulty to define the boundaries of the ontologies and the level of details required in the modeling activity in order to exploit them effectively. We do not want to focus on a scenario for the building of coalition ontologies, which could constitute a bias in the modeling activity. However, the knowledge of related questions and information needs in different coalition operations contexts could help orient and refine our modeling efforts.
Another important issue to be addressed will be the validation of ontologies. In the near future, the implementation of this proposal and its demonstration through the use of a realistic coalition scenario and integration with command and control information systems will help validate our approach.

10 References

[Boury-Brisset, 2000] A.-C. Boury-Brisset, Knowledge Modeling and Management for Command and Control Environments, in Proceedings of the 2000 Command and Control Research and Technology Symposium, June 26-28, 2000, Monterey, CA.

[Bray et al, 1998] T. Bray, J. Paoli, C.M. Sperberg-McQueen, Extensible Markup Language (XML) 1.0, W3C Recommendation, 10 February 1998, http://www.w3.org/TR/REC-xml.

[Brickley and Guha, 2000] D. Brickley, R. Guha, Resource Description Framework (RDF) Schema Specification, W3C candidate recommendation, 27 march 2000, http://www.w3.org/TR/2000/CR-rdf-schema-20000327

[CIA WFB] The CIA World Fact Book, at http://www.odci.gov/cia/publications/factbook/

[Cohen et al, 1998] P. Cohen, R. Schrag, E. Jones, A. Pease, A. Lin, B. Starr, D. Gunning and Murray Burke, The DARPA High-Performance Knowledge Base Project, in AI Magazine, Winter 1998, pp. 25-49.

[CoAX] CoAX project, http://www.aiai.ed.ac.uk/project/coax/

[CoABS] CoABS project, http://coabs.globalinfotek.com/

[DAML], The DARPA Agent Markup Language, http://www.daml.org.

[Duineveld et al., 1999] A. J. Duineveld, R. Stoter, M. R. Weiden, B. Kenepa, V. R. Benjamins, Wondertools? A comparative study of ontological engineering tools, in Proceedings of the Workshop on Knowledge Acquisition, Modeling, and Management, KAW'99, Banff, Canada, 1999.

[Farquhar et al, 1996] A. Farquhar, R.Fikes, and J. Rice, The Onlingua Server: A Tool for Collaborative Ontology Construction, in Proceedings of Knowledge Acquisition Workshop, Banff, Canada, 1996.

[Fernandez et al, 1997] M. Fernandez, A. Gomez-Perez, N. Juristo, METHONTOLOGY : From Ontological Art towards Ontological Engineering, AAAI-97 Spring Symposium on Ontological Engineering, Stanford University, March 24-26th 1997.

[Gruber, 1993] T. Gruber, A translation Approach to Portable Ontology Specifications, in Knowledge Acquisition, 5(2), pp. 199-220, 1993.

[Jones et al, 1998] Jones D., Bench-Capon T., Visser P., Methodologies for Ontology Development, in Proceedings of IT&KNOWS (Information Technology and Knowledge Systems) of the 15th IFIP World Computer Congress, Budapest, Hungary, 1998.

[van Harmelen and Fensel, 1999] F. Van Harmelen, D. Fensel, Practical Knowledge Representation for the Web, in Proceedings of the IJCAI Workshop on intelligent information integration, 1999.

[Hendler and McGuinness, 2000] J. Hendler, D. McGuinness, The DARPA Agent Markup Language, in IEEE Intelligent Systems, Vol. 15, No. 6, Nov/Dec. 2000.

[KBSI 1994], IDEF5 Ontology Description Capture Method Overview, KBSI Report, 1994, http://www.idef.com/idef5.html.

[Lassila and Swick, 1999] O. Lassila, R. Swick, Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation, 22 February 1999, http://www.w3.org/TR/PR-rdf-syntax

[McGuinness, 2000] D. L. McGuinness, Conceptual Modeling for Distributed Ontology Environments, In Proceedings of the Eighth International Conference on Conceptual Structures Logical, Linguistic, and Computational Issues (ICCS 2000), Darmstadt, Germany. August 14-18, 2000.

[McGuinness, 2001] D. L. McGuinness, Ontologies for electronic commerce, in IEEE Intelligent Systems, January/February 2001.

[Roy et al, 2000] N.F. Roy, R. W. Fergerson, & M. A. Musen. The knowledge model of Protege-2000: Combining interoperability and flexibility, in Proceedings of the 2nd International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000), Juan-les-Pins, France, 2000.

[Tate, 1996] A. Tate, Towards a Plan Ontology, in Newsletter of the Association for Italian Artificial Intelligence (AIIA), 1996.

[Teximus] at http://www.Teximus.com.

[Uschold and Gruninger, 1996] M. Uschold, M. Gruninger, Ontologies : principles, methods and applications, Knowledge Engineering Review, 11(2), 1996, pp. 93-155.

[Valente et al, 1999] A. Valente, T. Russ, R. MacGregor, W. Swartout, Building and Reusing an Ontology of Air Campaign Planning, in IEEE Intelligent Systems, January/February 1999, pp. 27-36.

Data Management for Coalition Interoperability

Bernhard Kües
Competence Center Informatik GmbH
Lohberg 10
49716 Meppen
Germany

1. SUMMARY

Internationally agreed standard data definitions are crucial for coalition interoperability between C4I systems. A new standard data model for NATO consultation, command and control (C3) is the *NATO Corporate Data Model*. It comprises a collection of various data models including one generic reference data model used as its joint conceptual core. The *NATO Corporate Data Model* and all related administrative information are available in the *NATO C3 Repository*, a data dictionary and administration tool to support data management within NATO. In addition to this data dictionary support, a hierarchical structure with organisational interfaces between national and international data management authorities will be necessary in order to achieve a well-performing data management in the multinational community of NATO.

2. INTRODUCTION

Data management is an essential basis for interoperability of C4I systems, particularly in a multinational community with various national views on information and different procedures for information exchange. Common data definitions and data exchange standards must be achieved and maintained, which often requires time-consuming efforts in multinational working groups and standardisation committees to agree on compromises between those different views. In particular, common requirements for the types of information to be exchanged must be identified and then represented in well-defined and semantically correct data models, which specify the correct meaning and the structure and relationships of those data elements that are relevant for the specific business field.

3. GENERIC DATA MODELS

The large variety of information requirements has already led to numerous data models, which have been developed in NATO for different subject fields and purposes – in most cases one unique data model for each individual C4I system, usually focussed on the specific system requirements.

A special kind of data model is the *ATCCIS Data Model* (the full name is *"ATCCIS Battlefield Generic Hub Data Model"*; see Figure 1), which has been developed in a multinational interoperability study during the past nine years (Ref. 1). This data model forms a basic part of the system architecture specification of the *Army Tactical Command and Control Information System* (or ATCCIS), which has been developed with the objective to achieve highly advanced interoperability (based on "NATO Level 5 of System Interconnection") between command and control information systems of national land forces for multinational operations. This *ATCCIS Data Model* has been proposed to NATO to become an internationally agreed interoperability standard with the NATO name *Land C2 Information Exchange Data Model* (Ref. 3). It will be a prerequisite for fully automated database-to-database data replication between heterogeneous national and multinational command and control information systems, e.g. using the *ATCCIS Replication Mechanism*, which allows user-controlled, selective data replication between various database management systems based on pre-defined, but dynamically changeable so-called "data exchange contracts".

The outstanding advantage of the Model fully attributed relational ATCCIS (or Land C2 Information Exchange) Data is its generic structure, which provides a stable set of commonly usable data definitions. Information requirements that are specific for the concerned subject area only and that thus are not yet included in the generic *ATCCIS Data Model* can be captured in national or system-specific extensions to this model. Such extensions can be supplemented in an easy and straight-forward manner without any modifications to the generic core of the data model.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Due to its generic structure, which mostly consists of generic entities with generic relationships, the model is very flexible in its usage (see Figure 1 for the key entities, all of which are associated with nearly 150 further entities to specify detail information, e.g. through sub-type entities). Although originally developed to support information exchange requirements for conventional warfare, it can also be used to handle types of information from other subject fields. This has been demonstrated, for example, in implementations for operations other than war without any modification of the data model. Also, the generic structure of the model is independent from the level of command.

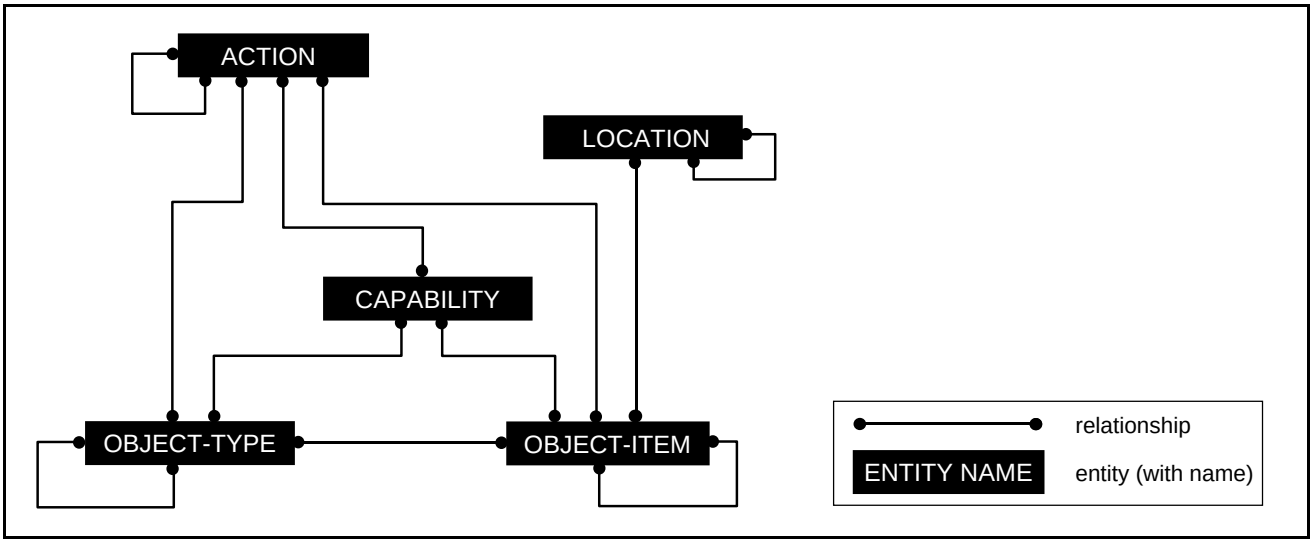


Figure 1: Generic key entities of the "ATCCIS Battlefield Generic Hub Data Model" (or "Land C2 Information Exchange Data Model") and of the "NATO Reference Data Model" (IDEF1X notation)

4. THE NATO CORPORATE DATA MODEL

Because this *ATCCIS Data Model* has found a wide acceptance in many NATO nations, it has also been chosen by the NATO Data Administration Group (NDAG) as starting point for the new *NATO Reference Data Model*, which will become the generic core of the *NATO Corporate Data Model* as a **joint** conceptual reference model. However, while the *ATCCIS Data Model* is focussed on automatic data exchange, the *NATO Reference Data Model* will be used for NATO data administration purposes. Version 1.0 of the *NATO Reference Data Model* has been completed in January 2001. Currently the model is being reviewed, extended and generalised from the limitations of a land view to the more general joint view (see Figure 2). The *NATO Corporate Data Model* has been proposed as NATO standard under the new standardisation agreement STANAG 5523, also called ADatP-32 (Ref. 2).

However, in order to become a successful standard, the *NATO Corporate Data Model* needs to be embedded in a well-functioning and NATO-wide data management. This will be controlled by the NDAG, a working group with representatives from the NATO nations as well as from Partnership-for-Peace nations. Due to its multinational environment, the NATO data management – like any other international, national or enterprise data management – must be organised in a strictly hierarchical structure. To be successful in the entire NATO community, the *NATO Corporate Data Model* needs to be accepted by the NATO nations as common basis for their own data definitions, either directly as national starting point for a nation's reference model or as common reference, to which other national data definitions and models can be mapped, so that national data can be translated into data as defined by the *NATO Reference Data Model* without semantic losses. This way, data exchange among national systems or between national and NATO systems can be facilitated, if necessary even in a fully automated manner.

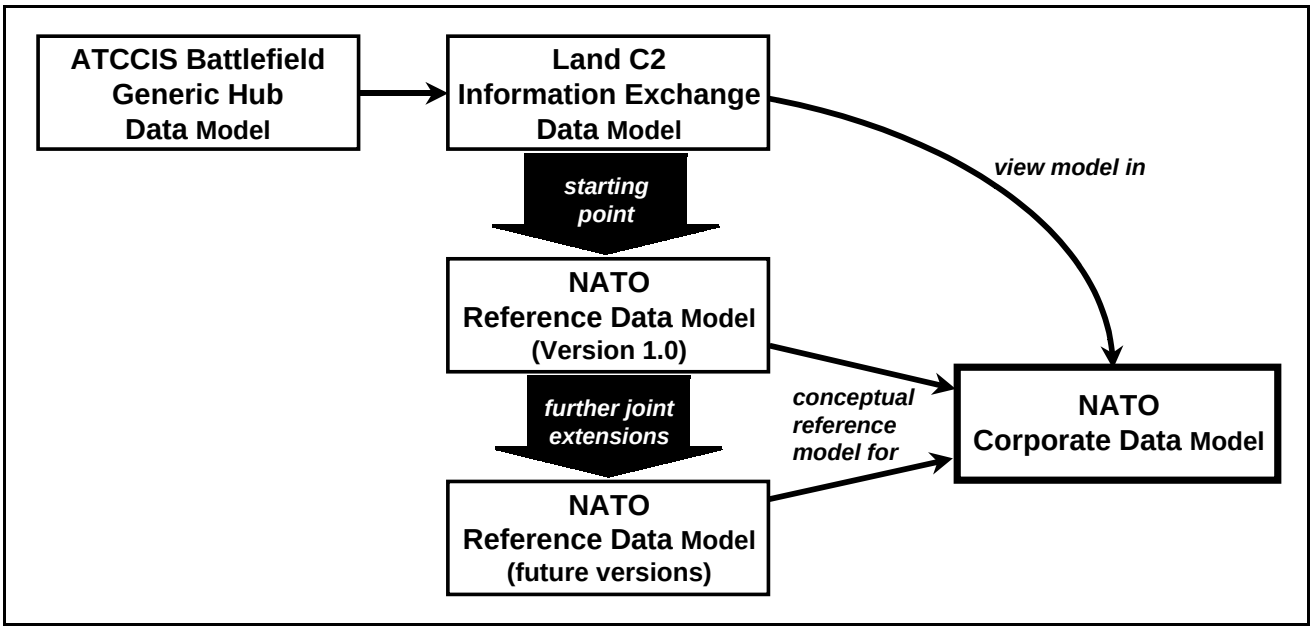


Figure 2: The NATO Corporate Data Model and its related data models

5. MULTINATIONAL DATA MANAGEMENT STRUCTURES

Data management is a complex business. Some of the data management functions are planning and utilisation of rules, procedures, organisational structures, methods and tools to identify, define and represent meaning and structure of data inclusive associated business rules, and also to ensure the quality of the resulting data models.

If data management is to be conducted in an international environment such as NATO and, in particular, if interoperability between C4I systems of different nations or between national and NATO systems shall be achieved, interfaces need to be implemented not just between C4I systems, but also between the nations' data management organisations and the NDAG as NATO's data administration body (Ref. 4). As a minimum, organisational interfaces with strict data management procedures in a hierarchical data management structure are required. Data management must always be organised in a top-down manner. Therefore, the data management procedures of NATO should also influence national procedures in order to achieve interoperability among data management systems of national and international data management instances. This means, the data management products, e.g. data definitions, rules and procedures that have been internationally agreed as being common for all nations, should be considered to be mandatory constraints for national data management organisations as well. A national data management organisation should be responsible in every NATO nation for the national implementation of the internationally agreed data management constraints.

On the other hand, nations are free to decide how they organise their own internal data management structures. Likewise, the nations may add any national extensions in terms of data definitions, procedures, etc. to the international data management products, as long as they implement the multinational core as part of their own national data standard. This principle can be applied through all levels of the data management hierarchy (see Figure 3). For example, the nations may decide to organise their national data management separately for the three military services army, navy and air force, and then leave it to the services again how to organise their own data management internally, taking account of any service-specific requirements and special features. In principle however, it is advisable that the data management structures should follow the hierarchy of the respective organisations involved, because fields of responsibility are clearly defined then already.

Between the data management instances at the various organisational levels within the data management hierarchy, exchange of management information will need to take place. This will be both ways: Information about standardised data elements need to be provided top-down, whereas feedback or change proposals may be submitted bottom-up.

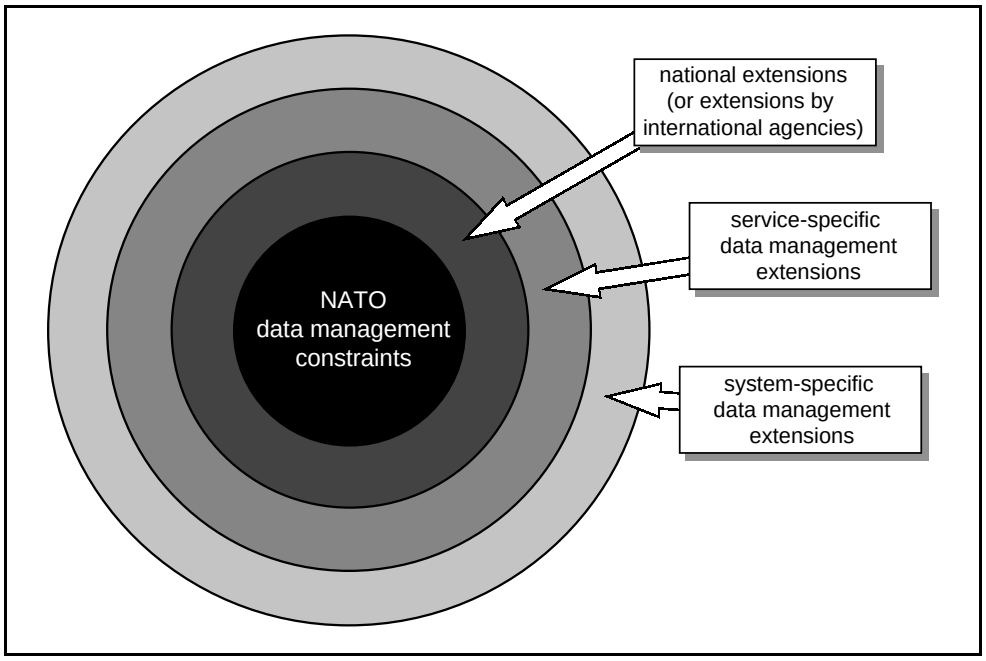


Figure 3: Hierarchy of data management constraints in a multinational community

6. THE NATO C3 REPOSITORY

To support this necessary exchange of data management information across organisational interfaces in an efficient way, suitable electronic interfaces should also be available. On the top level within NATO this is realised by means of an Information Resource Dictionary System called the *NATO C3 Repository*. This is a data dictionary and a data administration support tool, which is centrally controlled by the NDAG. The *NATO C3 Repository* provides all necessary details about the data elements of the *NATO Corporate Data Model* to national and international data management bodies. In other words, it contains all so-called "meta data" that is considered necessary to define and to administrate the data elements of the *NATO Corporate Data Model*. This meta data consists of definitions for entities, attributes and relationships. This also includes domain specifications with lists of allowable data values as well as business rules to be applied. In addition, XML schemata are provided.

Each nation is invited to utilise the *NATO C3 Repository* as much as possible. This will include its application for national data administration purposes as well. Access to this repository is possible for data management organisations through the Internet.

The *NATO Corporate Data Model* is not a data model in the common sense. Rather, it can be regarded as a "container" of data models, because it includes the aforementioned generic *Reference Data Model* as the joint conceptual core reference as well as a collection of view data models, which usually define specific views on the data of a particular subject field or C4I system (see Figure 4). In addition, view models may also represent standardised data specifications for information exchange between C4I systems, which also includes standard message structures.

In addition, mapping information between the *Reference Data Model* and the various view models will be held in the *NATO C3 Repository*. This will allow to link the individual data elements of a specific view model to the standardised definitions of the corresponding standard data elements within the Reference Data Model.

One special view model is the *Land C2 Information Exchange Data Model* (or *ATCCIS Data Model*) which, as mentioned above, has also been used in its other role as the yet "land-oriented" starting point of the modelling activities for the joint *Reference Data Model*.

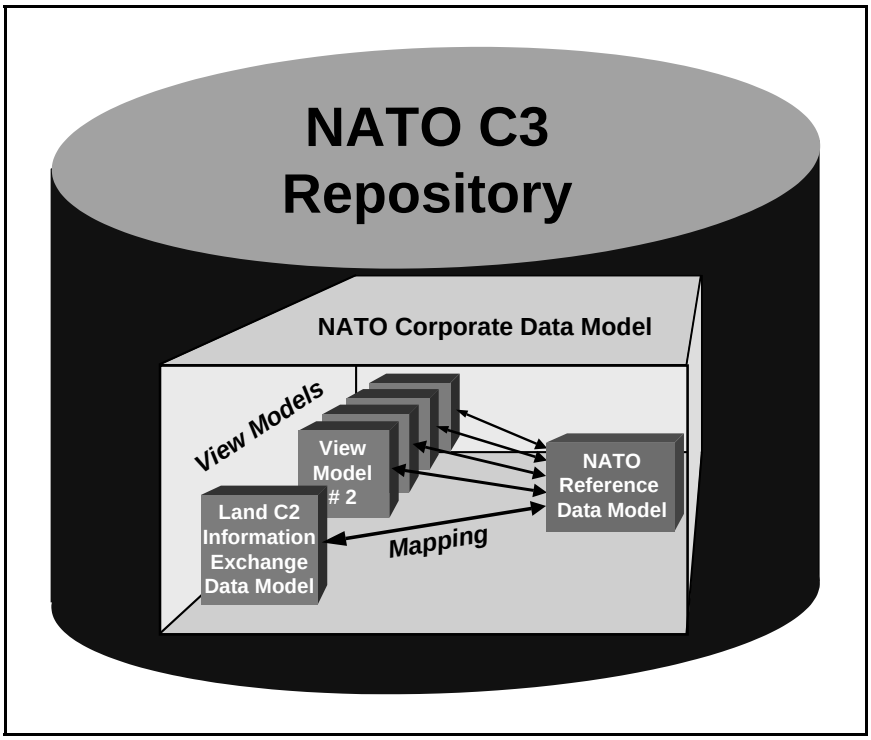


Figure 4: NATO C3 Repository containing the NATO Corporate Data Model with NATO Reference Data Model, view models and mapping information

For legacy systems, either the *Reference Data Model* or, for instance, the *Land C2 Information Exchange Data Model* can also be used as a kind of "common language" for data exchange between C4I systems based on different data definitions and structures. Interfaces can be developed, through which data can be exchanged after having been "translated" from the system-specific data structure to the common data exchange language as defined by the standard data model used (or vice versa). This way, data may be exchanged even in a fully automated manner, for example by means of a data replication technique such as the ATCCIS Replication Mechanism. The number of necessary information exchange interfaces could be reduced this way to one per system. Another advantage is that the development of such data exchange interfaces could be entirely independent from the knowledge about the data structures of other systems (see Figure 5).

7. CONCLUSION

The *NATO C3 Repository* as the container for the *NATO Corporate Data Model* shall become NATO's most important data management tool to support all nations and services striving for interoperability. It shall particularly support national and international data management organisations in taking account of common constraints when defining data structures during development or maintenance of C4I systems.

No matter whether the data model for a C4I system is directly derived from the *NATO Corporate Data Model* or whether its data structures are mapped to the *Reference Data Model* resulting in a data translation mechanism: It will certainly benefit from the *NATO Corporate Data Model*, of which the purpose is to form the semantic basis for coalition interoperability.

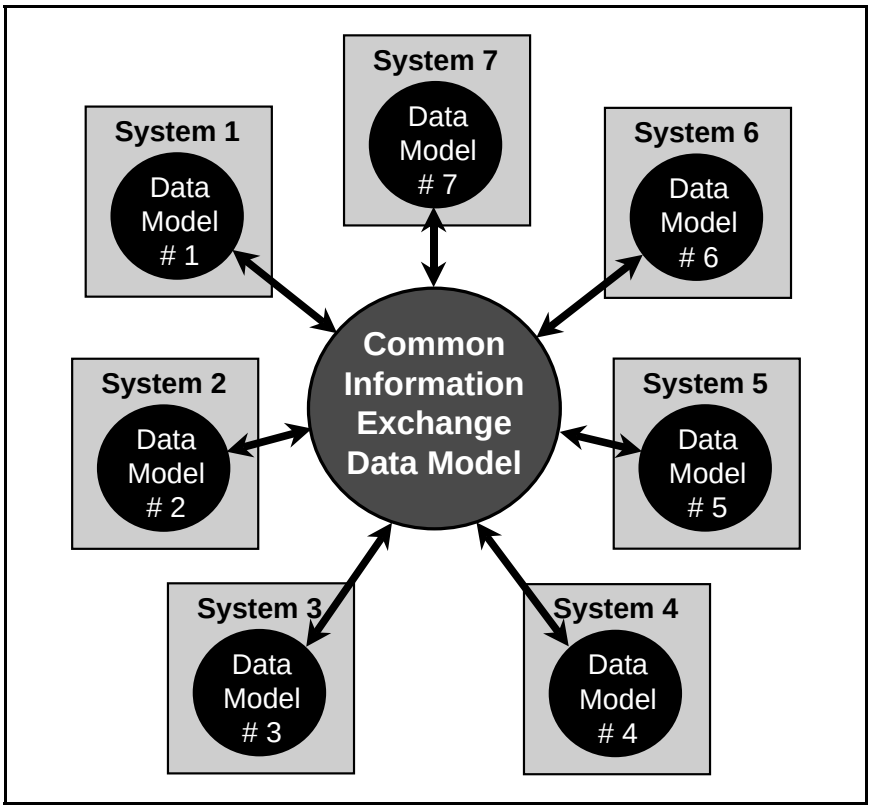


Figure 5: Data exchange between heterogeneous C4I systems through data translation between system data models and the common information exchange data model

8. REFERENCES

[1] ATCCIS Permanent Working Group: ATCCIS Working Paper 5-5, Edition 3.0, *ATCCIS Battlefield Generic Hub 3 Data Model Specification*; SHAPE, Belgium, 10 July 1998 (NATO Unclassified)

[2] STANAG 5523, ADatP-32, Part I: *The NATO Corporate Data Model – Concept and Description*; January 2001 (NATO Unclassified)

[3] STANAG 5523, ADatP-32, Part IV: *The Land C2 Information Exchange Data Model*, Edition 2.0; 31 March 2000 (NATO Unclassified)

[4] NATO Consultation, Command and Control Board (NC3B)/Information Systems Sub-Committee (ISSC)/NDAG: *NATO C3 Data Management Architecture (NC3DMA)*; Working Paper 5, AC/322(SC/5-WG/3)WP5, Version 2.0, 26 February 1999 (NATO/EAPC/PfP Unclassified)

An Agent-Based Approach to Achieve Interoperable and Adaptable Military Coalitions

Zakaria Maamar
College of Information Systems
Zayed University
PO Box 19282, Dubai
United Arab Emirates
zakaria.maamar@zu.ac.ae

Nabil Sahli
Computer Sciences Department and Research Center in Geomatics
Laval University
Ste-Foy, QC G1K 7P4 Canada
nabil.sahli@ift.ulaval.ca

Bernard Moulin
Computer Sciences Department and Research Center in Geomatics
Laval University
Ste-Foy, QC G1K 7P4 Canada
bernard.moulin@ift.ulaval.ca

Paul Labbé
Defence Research Establishment Valcartier
Val Bélair, QC G3J 1X5, Canada
paul.labbe@drev.dnd.ca

David Demers
Defence Research Establishment Valcartier
Val Bélair, QC G3J 1X5, Canada
david.demers@drev.dnd.ca

Summary

Military coalitions not only exploit complex information technologies but also must be able to adapt to changing international and departmental agreements, operational procedures, and new technology insertion. A software agent is an autonomous entity that is able to carry out complex operations on behalf of users. In addition, a mobile agent is able to move from machine to machine, performing its operations locally. This paper discusses how software agents could be used to aid the interoperability and adaptability of military coalitions. Defining, managing, and adapting these coalitions' processes by way of agents is also discussed.

Keywords: coalition, interoperability, adaptability, and software agents.

1. Introduction

The purpose of this paper is to discuss interoperability and adaptability issues in the context of military coalitions. Software Agents (SAs), originally developed within the field of Distributed Artificial Intelligence (DAI), seem to be a promising approach to address challenges imposed by coalition operations. Interoperability requires dealing with distribution and heterogeneity constraints of systems. Adaptability requires dealing with the dynamic characteristics of the application domain, such as handling unforeseen events. In this paper, the medical-evacuation domain illustrates these characteristics.

It seems that SAs, either static or mobile code (soft-mobility), which can run on fixed or deployable computer assets (asset-mobility), are well suited to environments that require adaptation and evolution [1]. Military coalitions are examples of such situations. A coalition could be set up for multiple purposes, e.g. humanitarian

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

assistance, peacekeeping, peace enforcement, etc. Moreover, it may occur that the mandate of a coalition changes. In fact, unexpected requirements could arise and need to be met, usually in a short period of time. For example, peacekeeping could turn to peace enforcement. As a result, coalition processes should be reviewed and adapted to fulfil the requirements of the new situation. Therefore, it is important to use appropriate technologies that could enable coalition adaptability. In this paper, we suggest using SAs as a means to achieve this flexibility. Adaptability means that:

- New participants, e.g. military forces, could join the coalition.
- Certain participants could be withdrawn from the coalition.
- Certain participants, e.g. Non-Governmental Organizations NGOs, could be involved intermittently in the coalition.
- Certain participants' features, such as role and security level, could change during operations.

Coalitions are virtually never identical. We view a coalition as a set of interconnected military entities that are organized to achieve a specific mission. Their systems, called Command & Control Information Systems (CCISs), are usually heterogeneous and distributed. Ensuring the interoperability of these systems is crucial to the coalition success. Indeed, designing a coalition means dealing with the following issues (adapted from [2]):

- Heterogeneity: many CCISs have differences with respect to hardware platforms, operating systems, and programming languages.
- Semantic unification: when different CCISs have to share information, they should understand and use a common vocabulary.
- Security: since a coalition consists of several CCISs from different origins, security must be present in different forms and at different levels. For instance, users must be authenticated according to their privileges and data must be encrypted.
- Standards compliance: by complying with existing standards, it makes it easier to design adaptable coalitions. Among these standards, the Common Object Request Broker Architecture (CORBA) from the Object Management Group (OMG) can be considered.

Additional issues should be considered, such as: How to provide mechanisms whereby CCISs can advertise their capabilities? How to execute processes spanning several CCISs? How to keep track of such processes for adaptability purposes?

In a coalition, a-priori agreements often dictate information that could be disclosed to another country according to the current situation or nature of conflict to be resolved. These agreements require a certain level of adaptiveness from the CCISs involved in the coalition. However, it is difficult to predict all possible interactions and situations that a coalition might encounter. Adaptiveness could be supported by an appropriate agent-based approach. In our approach, we suggest enhancing agents with the following abilities: interleaving planning and execution; creating additional agents when needed; and finally delegating planning to agents created for this purpose. Based on a priori agreements, an agent creates initial plans for its operations (in the coalition CCIS domain). If it lacks information locally to continue planning, this agent proceeds as follows (this presumes that the initial step of a feasible plan can be determined with relative certitude - to avoid backtracking inefficiencies):

- The agent creates a delegate-agent. This delegate-agent is mandated to look for the information that is required to pursue planning.
- The agent carries out the partial plan it has created (i.e., "plan a little, execute a little" in the absence of all desired information).

When the agent has finished its execution, it interacts with its delegate-agent regarding the information it requested. In what follows, we illustrate how planning, execution, and delegation ensure the adaptiveness of coalitions. For illustration purposes, we use an example of a medical evacuation application.

The paper is organized as follows. Section 1 proposes an overview of our study, i.e. interoperable and adaptable CCISs. Section 2 presents the basic concepts that are used in this study. Section 3 describes the architecture we suggest to achieve the interoperability of the coalition's systems. Section 4 deals with the medical-evacuation domain, as an illustration of the adaptability of the coalition's processes. Finally, Section 5 presents concluding remarks.

2. Background

This section is divided into three parts. The first part introduces types of interoperability in coalitions. The second part explains the notion of software agents. Finally, the third part presents the characteristics of CCISs.

2.1 Interoperability types in coalitions

Countries set up coalitions for different purposes: humanitarian assistance, peace support operations, etc. However, several constraints could hinder coalitions from achieving success. Among these constraints, we cite:

- People from different countries, at different locations, and at different moments contribute to the definition of the same operations. They may have cultural and organizational differences and possess informatics resources based on a variety of technologies.
- At different hierarchical levels, people make decisions during the performance of operations. Decisions may be based on information that is not well understood (or understood differently) by all participants. Furthermore, efficient operations require co-ordination, synchronization and the understanding and sharing of a common intent. Also, decisions may require the interaction of CCISs that are distributed and heterogeneous.
- In the theatre of operations, it is complex to provide and maintain a high level of informatics assistance to military users. For example, it is likely not possible to allocate to each combat unit a technician having a high level of expertise regarding PC software, Unix software, etc. Moreover, it is not possible for a military user to be aware of the characteristics of the multiple CCISs that are involved in the coalition.

On the basis of the constraints cited above, coalitions are appropriate candidates for stressing interoperability issues and concepts. Coalition CCIS interoperability aims at achieving efficient collaborative interactions between multiple systems, groups of people and individuals to aid coalition co-operation and co-ordination.

In coalitions, we identified three types of interoperability (adapted from [5]): interconnectivity, integration, and collaboration. Interconnectivity, defined as basic interoperability, allows simple data transfer with no semantic, whereas application-level integration enables applications, e.g. CCISs, running in any environment to exchange services, even if different organizations designed these applications at different times. In coalitions, working at the application-level is not necessarily enough, particularly if the military forces aim at merging their operational processes. Therefore, collaboration at the command level is required. In what follows, the three types of interoperability are summarized (cf. Figure 1):

- Physical interconnectivity: to guarantee basic communication, computing resources are first interconnected to exchange messages.
- Application integration: its main purpose is to carry out operations among different computing resources. Generally, these resources are distributed across networks and heterogeneous at hardware, software, and terminological levels.
- Command collaboration: it goes beyond application integration by expanding military operational processes to other structures. To this end, a collection of components, such as software agents gathered into Multi-Agent Systems (MASs), could be set up. These components collaborate (more than just interoperate).

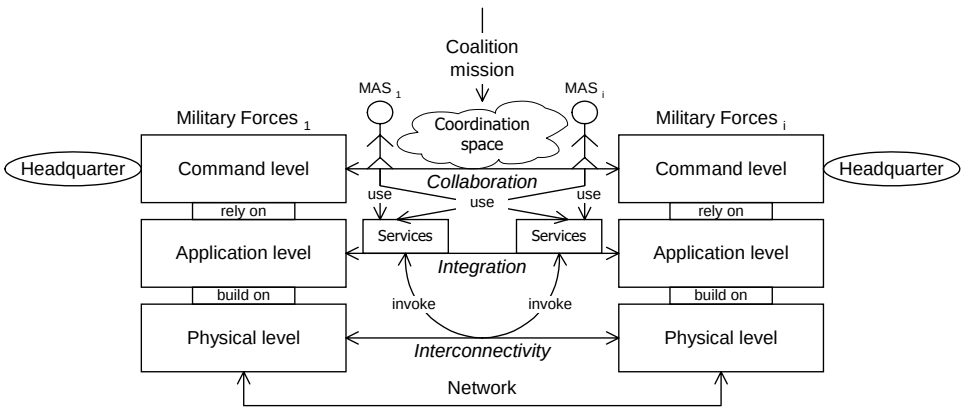


Figure 1 From interconnectivity to collaboration, through integration

The types of interoperability as discussed above present similarities with the work done in [13]. The author suggests three layers, namely business architecture, application architecture, and technological architecture. Achieving the interoperability between these layers requires integrating: 1) the processes for the business layer, i.e. collaboration in Figure 1; 2) the enterprise applications for the application layer, i.e. integration in Figure 1; and 3) the middlewares for the technological layer, i.e. interconnectivity in Figure 1.

2.2 Software agents

In DAI, researchers have studied several issues related to the distribution and co-ordination of knowledge and actions in environments involving multiple entities, called agents. Agents can take different forms depending on the nature of the environment in which they evolve. A particular type of agents, SAs, has recently attracted much attention [1]. A SA is an autonomous entity that acts on a user's behalf, takes initiatives, makes suggestions and can make decisions (depending on the permissions etc. afforded to it). A SA could be enhanced with mobility mechanisms; thus it could migrate from machine to machine in a heterogeneous network [3]. In fact, agent execution can be suspended at an arbitrary point and resumed when arriving on another machine (assuming compatibility, permissions etc. issues are addressed). For more details on agents, see [6].

2.3 Command & control information system

When information technologies are an inherent part of the commanders' decision-making process, CCISs help commanders to obtain a view of the situation in which they are involved (cf. Figure 2). CCISs emerge from their organization's structure, tasks, and functions [4]. A CCIS structure represents an assembly of facilities, arranged to meet the CCIS's objectives. To reach these objectives, the CCIS's functions are triggered in order to carry out the needed tasks. Different types of functions exist; they vary from planning and weather forecast to data fusion. Certain functions may receive messages from the external environment, e.g. sensors, through communication modules. For the purpose of this paper (mainly for Section 4), we assume that the CCISs include a specific function that gives information on the Medical Treatment Facilities (MTFs) that are available in a theatre of operations.

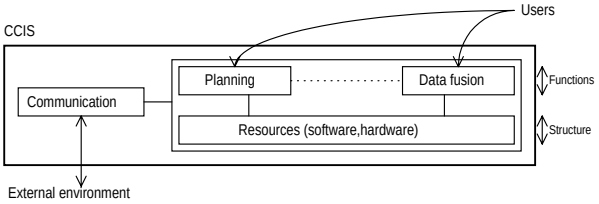


Figure 2 CCIS simplified architecture

3. Coalition interoperability – design perspective

In the literature, different agent-based approaches for system interoperability can be found [7,8,9]. All these approaches assume that the network infrastructure is fully reliable and the information throughput is unlimited. Unfortunately, these assumptions do not often hold in coalitions.

Assuming the previously mentioned approaches and types of interoperability in coalitions, we propose a MAS-based architecture that aims at achieving interoperable and adaptable coalitions (cf. Figure 3). These MASs interact remotely as well as locally. In the latter case (locally executed but co-ordinated and synchronized with other agents), agents interact through a facility called the advertisement infrastructure that consists of a bulletin board and a repository of active-agents.

3.1 Types of agents

In Figure 3, the MASs integrate different types of SAs: interface-agents assisting users; CCIS-agents invoking CCIS functions and satisfying user needs; resolution-agents co-ordinating the satisfaction of user needs; control-agents managing MASs; and finally, a supervisor-agent monitoring the advertisement infrastructure. In these MASs, resolution-agents are able to create delegate-agents and transfer them either to the advertisement infrastructure or to other distant MASs (and their advertisement infrastructures). Delegate-agents carry out operations on behalf of resolution-agents. In what follows, the different agents as well as the advertisement infrastructure are presented. This presentation includes examples (in *italic sentences*) from the medical-evacuation field. It is used as an introduction to the next section.

- Interface-agent: it assists users in formulating their needs, *e.g. transferring a patient from a combat zone to a military treatment facility MTF*, maps these needs onto requests, forwards these requests to the CCIS-Agent in order to be processed, and provides users with answers the CCIS-agent gives.

- CCIS-agent: it receives user requests from interface-agents. The CCIS-agent only processes these requests if they require the involvement of the CCIS that delegated this CCIS-agent. Furthermore, the CCIS-agent acts on its behalf in order to maintain the CCIS' autonomy. To this end, the CCIS-agent advertises its services. In fact, each service, *e.g. locating MTFs within a CCIS' area of responsibility*, corresponds to a CCIS's function. Advertising services means posting notes on the bulletin board of the advertisement infrastructure. The CCIS-agent sends remote requests to the supervisor-agent.

- Resolution-agent: it processes user requests, only if these requests are transmitted by the CCIS-Agent and need the involvement of several CCISs to be completed, *e.g. transferring a patient may require visiting several MTFs and transportation centres (TCs) in different areas of operations*.

  Initially, the resolution-agent creates a delegate-agent and transfers it to the advertisement infrastructure. As soon as the delegate-agent arrives, the supervisor-agent authenticates it. Then, the delegate-agent waits for the resolution-agent's queries about the services to look for on the Bulletin Board (the information publishing mechanism used by the advertisement infrastructure).

  In order to identify the CCISs that are required to satisfy user requests, the resolution-agent sends remote queries to its delegate-agent. This agent browses the bulletin board, identifies appropriate CCISs through the services offered by their CCIS-agents, and finally informs its parent resolution-agent remotely. Then, the resolution-agent creates the procedure needed to fulfil the user's request. Such a procedure is called an itinerary, *e.g. the patient will visit MTF_1 and MTF_2*.

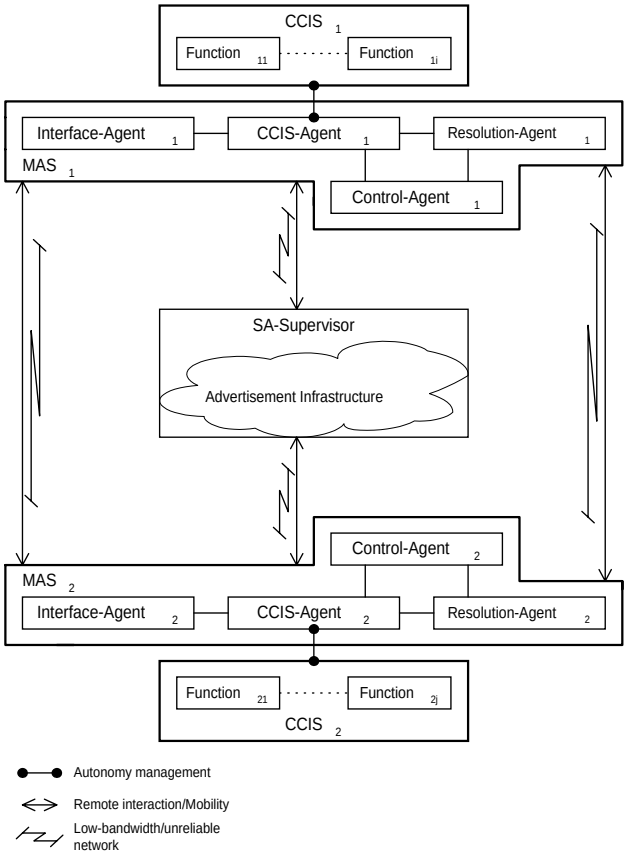


Figure 3 Architecture for interoperable and adaptable coalitions

- Control-agent: in an environment consisting of mobile agents, mobility operations consist of transferring the agents through the net to other distant systems, authenticating these agents as soon as they arrive, and finally installing these agents to resume their operations. The control-agent is in

charge of all these operations. For instance, when a delegate-agent moves, it first interacts with the control-agent in order to be transferred to the advertisement infrastructure.

- Supervisor-agent: it is in charge of several operations. It manages the advertisement infrastructure by receiving CCIS-agents' advertisements, sets up a security policy in order to monitor the delegate-agents accessing this infrastructure, and finally, allows delegate-agents to operate within this infrastructure. The supervisor-agent uses the repository of active-agents to register all the delegate-agents and CCIS-agents that were respectively permitted to enter the advertisement infrastructure and advertise their services. The repository of active-agents is also updated when resolution-agents decide to withdraw their delegate-agents from the advertisement infrastructure.

- Advertisement infrastructure: in coalitions, CCISs are spread across networks and may communicate through low-bandwidth and/or unreliable channels. Moreover, a military user may use a VHF Combat Net Radio (or an alternative, possibly geo-referenced, wireless information exchange technology) to send and request information. This military user usually relies on mobile devices, such as portable computers, that are only intermittently connected to networks. Instead of overloading the network, we suggest migrating certain agents to the advertisement infrastructure. Within this infrastructure, these agents could locally browse the bulletin board, looking for appropriate CCIS services and information. For the mobile user case, the delegate agent can negotiate to obtain services and information from CCISs and communicate results to the mobile user during the next connect time.

3.2 Discussions

When distributed agents need to communicate, for example to exchange information regarding their capabilities, two approaches are suggested: federation and autonomy.

In the federation approach, three types of scenarios are possible: facilitator, broker, and mediator. In the facilitator scenario, several agents are integrated into groups. Communication between these groups of agents takes place through interface agents called facilitators. To carry out its operations, a facilitator uses two types of services: message routing and message translation. Brokers are similar to facilitators. However, they provide additional services such as notifying agents that indicated their interest regarding certain messages. Finally, a mediator agent is more complex than a broker. For instance, it can use co-ordination and learning mechanisms.

The autonomy approach is completely different from the federation approach. An autonomous agent should have the following characteristics: it is neither controlled nor managed by other agents; it can communicate directly with other agents; it has knowledge about other agents and its environment; it possesses its own goals and motivations. According to these characteristics, this means that all the services of the federation approach are associated with the agents' capabilities.

In Figure 3, communication between agents is a combination of federation and autonomy approaches. In fact, the advertisement infrastructure plays the role of a support facility for the agent interactions. Meanwhile, this facility is just a platform where agents can execute. Therefore, agents need to be autonomous in order to manage their (inter)actions.

4. Coalition adaptability - application perspective

In this section, we discuss medical evacuation within a coalition. Medical evacuation aims at transferring patients to an appropriate medical treatment facility [10,11]. Patients' requirements and facilities' capabilities must be taken into account to achieve effective resource utilization.

4.1 Situation description[1]: medical evacuation

In medical-evacuation situations, the intervention area may be decomposed into several regions, with each region having a commander in charge of conducting operations. We assume that each country contributing to the coalition is in charge of a specific region and has a corresponding CCIS. Each region could have its own medical service specialities (e.g., acid burns, broken bones, life-threatening trauma, eye injuries, dehydration). Therefore, we associate regions and medical specializations with MASs as presented in Figure 3. In these

[1] The situation described here is a simplification of reality in order to not overload this paper's content.

MASs, CCIS-agents with their CCISs are aware of the incoming and outgoing flows occurring within their regions. In the rest of this paper, we focus on the flow involving patients. Each region has several Transportation Centres (TCs). Among them, imagine two are respectively used for incoming flows and outgoing flows. Both could be either co-located or in different places. Each TC is associated with a TC-agent. Usually, each region has several Medical Treatment Facilities (MTFs). MTFs have associated MTF-agents to manage the allocation of their resources within the MAS. Each MTF-agent knows how many unoccupied beds are available, what kinds of equipment and treatment expertise are in its MTF, which TCs are located near this MTF, etc. MTF-agents regularly send updates to CCIS-agents. Within Figure 3, in addition to interface-agents, CCIS-agents, resolution-agents, and control-agents, MTF-agents and TC-Agents are integrated into MASs for the medical evacuation application area (cf. Figure 4).

It may occur that a patient requires specific treatments. As a first step, the patient is sent to a local MTF. There, the patient receives initial treatments, waiting for transfer to an appropriate MTF (when necessary), either in the same region (intra-region move) or in another region (inter-region move). Transferring patients between MTFs is subject to the following constraints:

- The authority of each region, i.e. the commander, should be informed when patients intend to use the MTFs and TCs under his authority.
- Conflicts on resources, e.g. beds, are more likely in inter-region moves.
- Each patient attempts to create an itinerary that specifies the initial MTF, the final MTF and the intervening MTFs and TCs. MTFs are limited by their number of beds, facilities and personnel (expertise). Their number of seats and their already assigned itineraries limit transport vehicles for other patients. Patients with specific care needs during transport introduce further constraints.

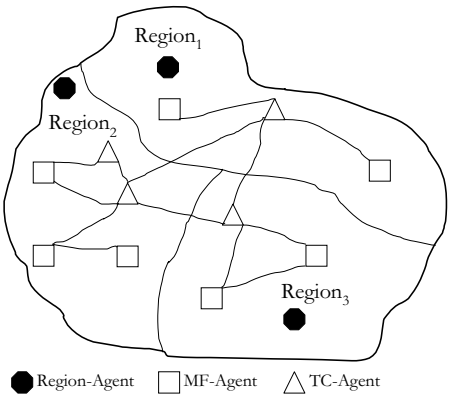


Figure 4 Agentification of intervention areas

In what follows, we explain how to plan (cf. Section 4.2) and then execute (cf. Section 4.3) a patient transfer.

4.2 Planning phase

In this phase, a MTF that meets the patient's requirements has to be identified. Each patient has a medical record that is managed by a patient-agent. First, the patient-agent sends messages to CCIS-agents, looking for a MTF that would be able to care for its patient. CCIS-agents forward messages to their MTF-agents. As soon as it gets replies, the patient-agent's first choice is one of the MTFs, which is within the patient's region. The second choice is a MTF in another region. In the second case, CCIS-agents entrust their operations to resolution-agents. Such operations mainly deal with looking for the MTFs using the advertisement infrastructure.

- First choice - intra-region move: after identifying the MTF that will host its patient, the patient-agent asks this MTF-agent to book a bed for its patient. The MTF-agent keeps this booking active for a specific period of time. This period depends on the patient's state as well as on the booking percentage in this MTF (when operating near full capacity, empty beds cannot be kept unoccupied for long periods). The MTF-agent answers the patient-agent, confirming its booking with an expiration date and time. The next step for the patient-agent consists of searching for the TCs that reach this MTF. According to TC and transport availability, a possible itinerary is designed and a confirmation for the patient's arrival-date is sent to the MTF-agent. If this itinerary planning fails, the patient-agent carries on with the same strategy but with another possible MTF of the same region (if such an MTF exists).

- Second choice - inter-region move: it may occur that the MTFs of a given region do not offer the treatments that a patient requires, don't have available beds or cannot be reached via operational TCs. Thus, the patient-agent of this patient looks for MTFs in other regions. An inter-region move requires transferring the patient:
    - From the initial MTF to an outgoing TC of the original region.
    - From this outgoing TC to an incoming TC of the destination region.
    - From this incoming TC to the selected MTF.

As soon as an appropriate MTF in another region is identified, the patient-agent proceeds as follows: it creates a delegate-agent and sends this delegate-agent to the outgoing TC of the patient's original region. The delegate-agent is in charge of planning the patient's sub-itinerary from this TC to the final destination, which is the selected MTF. Meanwhile the patient-agent is in charge of planning the sub-itinerary that leads the patient from the original MTF to the outgoing TC. In fact, the patient-agent entrusts a part of its responsibilities to the delegate-agent (cf. Figure 5).

As soon as it arrives at the outgoing TC, the delegate-agent interrogates MTF-agents, through CCIS-agents, regarding the expertise of their MTFs. To this end, the delegate-agent specifies the patient's state and requirements (for our example, we assume that at least one MTF will answer positively). When it gets a reply from a specific MTF, consisting of a booking with an expiration date, the delegate-agent interacts with an incoming TC of this MTF's region, regarding the itinerary to follow within that region. Finally, the delegate-agent confirms to the destination MTF the patient's estimated arrival date and time. At the same time, the patient-agent works on establishing a sub-itinerary for its patient from the original MTF to the outgoing TC. As soon as the patient-agent finishes, the patient is transferred to this TC via the planned itinerary (In certain cases, the patient's evacuation can commence even all parts of the itinerary are not finalized [11]. The patient-agent must customize its planning according to the current patient's position, "plan a little, execute a little"). The patient-agent is also part of this transfer. When the patient-agent arrives at the outgoing TC, the delegate-agent provides this patient-agent with the sub-itinerary that needs to be followed. Finally, the delegate-agent is deleted.

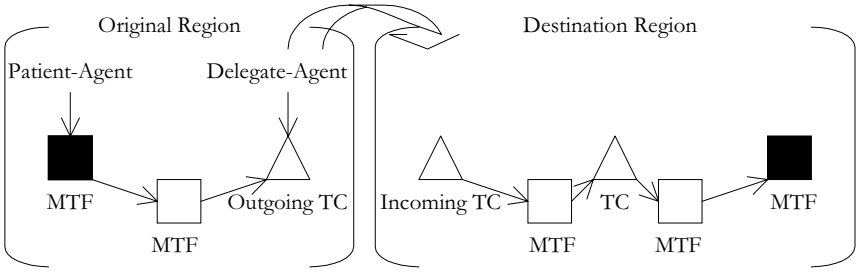


Figure 5 Patient-agent delegating its responsibilities

In this description of the planning phase, Delegate-agents are only sent to the outgoing TC for inter-region patient transfers. In reality, this idea of dispatching a Delegate-agent to plan ahead can be applied to any multi-step patient itinerary. This helps deal with any required modifications as TC, MTF, and transport availabilities evolve, as is presented in Section 4.3.

4. 3 Execution phase

Once the patient's itinerary is established, he is evacuated to the destination MTF. This evacuation may consist of visiting several TCs as well as several MTFs. However, each step of the itinerary could be subject to modifications. Therefore, the patient-agent follows its patient's itinerary. Dealing with any modification requires that the patient-agent create another delegate-agent. This agent will precede the patient-agent, by one step. The delegate-agent's role is to visit the destination site of a step and notify its patient-agent in case unforeseen events occur. Examples of these events could be a departure from a TC is delayed, a departure is cancelled, etc. In order to deal with these events, the delegate-agent should have information on the expected state of a site in order to compare it with the current state of this site. If any difference is noticed, the patient-agent is informed in order to perform corrective actions. To this end, the patient-agent needs to re-plan a part of the initial plan in order to deal with this difference. For instance, if a departure to a TC has been cancelled,

it would be advantageous to be aware of this situation before the patient arrives at this TC (in order to re-plan future itinerary "legs").

4.4 Discussions

Coalition adaptability requires interleaving planning and execution. Keeping the classical distinction between planning and execution is no longer valid, especially in dynamic environments, such as medical evacuation. Agents need to adapt their plans based on the actions that other agents carry out as well as events that were unforeseen. In [12], the author claims that if there are several agents that may help or prevent an agent from achieving its goal, computing a full plan may take so long that after completing the computation, the plan becomes out-of-date, because the world has changed in the meantime. This reinforces the choice of using Delegate-agents and the "plan a little, execute a little" modus operandi to achieve effective resource utilization.

In other research projects, interleaving planning and execution means interrupting planning, executing the portion of the plan that has been designed, and finally, resuming planning. This is time-consuming, particularly in critical situations. In our work, planning is not suspended; a delegate agent is created and mandated to pursue planning while the initial part of the plan is carried out. As soon as the agent completes the execution of the present part of the plan, it obtains from the delegate agent the next portion of the plan and can resume execution at this point. This process can be repeated until the completion of the entire plan. Therefore, interleaving planning and execution is enhanced with delegation.

This ability to interleave planning and execution seems to fit naturally with applications such as logistics (move materiel toward active troops) and medical evacuation (move patients out of harm's way). In these cases, the desired general direction for movement is evident, even though the exact itinerary (and even the final destination) may be unknown. It is similar to the commercial airline passenger re-routing problem during, for example, a major winter storm in northeastern North America. For a long distance trip, it is usually better to go to a large "hub" airport than stay at a small regional one – the options for re-planning one's itinerary from there are much larger. Analogously, a patient may be evacuated from a battle area to a TC, since that represents the first step in a successful itinerary for treatment, irrespective of the patient's final destination.

5. Conclusion

In this paper, we presented an agent-based approach that aims at enhancing interoperability and adaptability of military coalitions for various operations including humanitarian aid. Interoperability means being able to exchange accurate information and provide timely services, despite several constraints such as distribution and heterogeneity. Adaptability means being able to cope with changes that could occur in an environment.

In a coalition context, MASs and their SAs are able to fulfil different operations, from user needs specification to the initiation of CCIS functions. We proposed different types of SAs to support coalitions, such as Interface-agent, CCIS-agent and Resolution-agent. Whereas MASs appear to offer benefits to coalition support, we must be aware of their limitations. MASs must allow a large degree of human interaction. Therefore, it is unrealistic to expect to be able to provide a "fully" automated coalition support. MASs can be thought of as a way of facilitating software and human agent collaboration. Hence, the human dimension of an application should be taken into account when designing MASs for coalition support.

Acknowledgements

This project has been supported by a contract from the Defense Research Establishment Valcartier, Quebec, Canada.

References

[1] N. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1) pp. 7-38, 1998.

[2] Z. Maamar, B. Moulin, and Y. Bédard. Software agent-oriented frameworks for the interoperability of georeferenced digital libraries on the world wide web: The SIGAL project. In R. Fegeas M.F. Goodchild, M.J. Egenhofer and C.A. Kottman, editors, *Interoperating Geographic Information Systems*, pages 335--354. Boston: Kluwer Academic Publishers, 1999.

[3] D. Lange and M. Oshima. Dispatch your agents; shut off your machine. *Communication of the ACM*, 42(3) pp. 88--89, March 1999.

[4] S. Malerud, Feet E.H., and U. Thorsen. A method for analysing command and control systems. Norwegian Defence Research Establishment (FFI) N-220 Kjeller, Norway.

[5] T.A. Au. A primer of CORBA: A framework for distribution applications in defence. Technical report, DSTO-GD-0192, DSTO Electronics and Surveillance Research Lab, Salisbury, Australia, 1999.

[6] S. Franklin and A. Gaesser. Is it an agent, or just a program? A taxonomy for autonomous agents. in *Proc. of 3rd International Workshop on Agent Theories, Architecture and Language (ATAL'96)*. Springer-Verlag, LNAI, 1996.

[7] R. Bayardo, W. Bohrer, R. Brice, A. Cichocki, G. Fowler, A. Helai, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, M. Rashid, M. Rusinkiewicz, R. Shea, C. Unnikrishnan, A. Unruh, and D. Woelk. InfoSleuth: Semantic integration of information in open and dynamic environments. In *Proc. of the 1997 ACM International Conference on the Management of Data (SIGMOD)*, Tucson, Arizona, 1997.

[8] S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom. The TSIMMIS project: Integration of heterogeneous information sources. In *Proc. of the 10th IPSJ*, Tokyo, Japan, 1994.

[9] C.A. Knoblock, Y. Arens, and C.N. Hsu. Cooperating agents for information retrieval. In *Second International Conference on Cooperative Information Systems*, Toronto, Canada, 1994.

[10] O. Lassila, M. Becker, and S. Smith. An exploratory prototype for reactive management of aero-medical evacuation plans. Technical report, CMU-RI-TR-96-03 The Robotics Institute, Carnegie Mellon University, Pittsburgh, U.S.A, 1996.

[11] V. Saks, G. Braidic, A. Kott, and C. Kirschner. Distributed medical evacuation planning: What problem should each agent solve? in *Proc. of the Workshop on Constraints and Agents, in The Fourteenth National Conference on Artificial Intelligence (AAAI'97)*, Providence, Rhode Island, 1997.

[12] S. Ambroszkiewicz. Agent virtual organizations within the framework of network computing: a case study. in *Proc of The First International Workshop of Central and Eastern Europe on Multi-agent Systems (CEEMAS'99)*, St. Petersburg, Russia, 1999.

[13] W. Hasselbring. Information System Integration. *Communication of the ACM*, 43(6), pp. 33-38. June 2000.

Jini in Military System Applications

Tim Wilkinson, Sue Haines, Craig Williams
Defence Evaluation and Research Agency
St Andrews Road
Malvern
Worcestershire
UNITED KINGDOM
WR14 3PS

Abstract

Jini is a distributed computing environment that extends the concept of 'plug-and-play' to networking. A Jini-enabled hardware or software component, or service, can be connected to a Jini network and announce its presence. Existing services on the network can be notified of the arrival of the new service and reconfigure themselves accordingly. Jini provides support for a client that wishes to use such a service, allowing it to locate the service and call on it to perform tasks.

This paper is the output of a study to investigate the possible role that the Jini technology could play in the Joint battlespace. In particular, the paper explores Jini's ability to provide a robust, reliable, secure and scalable solution to network service provision, and details the benefits and limitations of Jini with recommendations of how those limitations may be overcome.

Keywords

Jini, Java, Distributed Systems, Security, Robustness, Scalability, Legacy Systems

Executive Summary

The emergence of low-cost mass-produced computing and information devices, and the ability to rapidly and dynamically link them together over a wide geographical area, presents a major opportunity and challenge to the conduct of future military operations.

Jini offers the potential to provide a 'design free' environment to achieve the flexibility and dynamic characteristics required in the Joint arena. With Jini it is possible to have extensively networked, auto-configuring systems developed from mass produced components using commercial technologies tailored to meet specific military applications.

Jini is a distributed computing environment that extends the concept of 'plug-and-play' to networking. A Jini-enabled hardware or software component, or service, can be connected to a Jini network and announce its presence. Existing services on the network can be notified of the arrival of the new service and reconfigure themselves accordingly. Jini provides support for a client that wishes to use such a service, allowing it to locate the service and call on it to perform tasks.

Using the Jini/Java framework, it is possible to build potentially robust and flexible networks of clients and services. The clients and services themselves can be written in pure Java, and implement the Jini communication protocols directly, or they can consist of legacy code that participates in the Jini federation by proxy.

The degree to which a legacy application can exploit the features of Jini depends strongly on the interface that the application exposes to programmers. For example, if the legacy application comes under the white-box category (the source code is accessible), then, with respect to licensing constraints, the code can be modified to accommodate Jini to any depth. In such a case, Jini could possibly be used to create a

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

fully distributed version of the application. For applications under the grey- and black-box categories (details of internal operation are not accessible), it is likely that Jini could only be applied at an application level.

This study has shown that with only a small amount of effort, a legacy application can be extended to use the most basic Jini functionality. However, to extend the application to take advantage of some of Jini's most advanced features, a potentially time consuming re-write of the legacy application may be required. It is important to note that trying to shoehorn an existent software system into a role that it was not designed for, is a non-trivial task. If significant changes are required, it can often take more effort to convert existing code than it would to develop a new system. This is not just the case when Jini-enabling a legacy application, but occurs whenever the internal design of a legacy application has to be significantly altered in order to integrate it into a new software architecture.

1. Introduction

The emergence of low-cost mass-produced computing and information devices, and the ability to rapidly and dynamically link them together over a wide geographical area, presents a major opportunity and challenge to the conduct of future military operations.

Jini offers the potential to provide a 'design free' environment to achieve the flexibility and dynamic characteristics required in the Joint arena. With Jini it is possible to have extensively networked, auto-configuring systems developed from mass produced components using commercial technologies tailored to meet specific military applications.

This paper provides a overview of Jini, and how it can be integrated with legacy applications. It addresses issues of Jini security, robustness, ability to operate work over Wide-Area-Networks and scalability issues. It concludes by describing military issues associated with using Jini and provides a set of conclusions.

2. Overview of Jini

2.1 A Network of Services

Jini is a distributed computing environment that extends the concept of 'plug-and-play' to networking. A Jini-enabled hardware or software component, or *service*, can be connected to a Jini network and announce its presence. Existing services on the network can be notified of the arrival of the new service and reconfigure themselves accordingly. Jini provides support for a *client* that wishes to use such a service, to allow it to locate the service and call on it to perform tasks. There are a number of scenarios that could exploit Jini's features:

- A new printer connected to a network could announce its presence and capabilities. A client could then use this printer without having to be specially configured to do so.
- A digital camera could be connected to a network. It could present a user interface that will not only allow pictures to be taken, but is also aware of any printers capable of printing them.
- A configuration file that is copied and modified on individual machines could be made into a network service from a single machine, reducing maintenance costs.
- New capabilities extending existing ones could be added to a running system without disrupting existing services, or any need to reconfigure possible clients.
- Services could announce changes of state, such as a printer running out of paper. Listeners, typically of an administrative nature, could watch for these and flag them for attention.

A Jini system, or *federa*tion, is a collection of clients and services all communicating by the Jini protocols. In its simplest form, this will consist of applications written in Java, communicating using the Java Remote Method Invocation (RMI) mechanism over a network running TCP/IP. However, it should

be noted that the Jini specification is independent of network protocol, but the only current implementation employs RMI over TCP/IP.

2.2 ADVERTISING A SERVICE

All Jini federations have one type of service in common: the *lookup service*. The lookup service acts as a broker/trader/locator between other services and clients. It is essentially a well-known point of contact that other services can use to advertise their availability, and which clients looking for a service can approach to obtain access to a service. More than one lookup service can participate within a Jini federation.

A service consists of a Java object or collection of Java objects residing on a *server*. The server performs various tasks on behalf of the service. The first task it performs is to register the service with a lookup service. In order to perform this registration, the server must first find a lookup service. This can be done in two ways: if the location of the lookup service is known, then the server can use unicast TCP to connect directly to it; if the location is not known, the server will make UDP multicast requests, and lookup services will respond to these requests.

When the lookup service gets a request, it sends an object back to the server. This object, known as a registrar, acts as a proxy to the lookup service, and runs in the service's Java Virtual Machine (JVM). Any requests that the server needs to make of the lookup service are made through this proxy registrar. Once the registrar has been obtained, the server can upload a copy of the service to the lookup service.

2.3 OBTAINING A SERVICE

A client wishing to make use of a service goes through the same initial steps to obtain a registrar as a service wishing to advertise its availability. When a registrar has been obtained, the client queries the lookup service for the existence of the service it is hoping to make use of. If such a service exists, the client downloads a copy of the service into its own JVM and makes requests of it locally.

2.4 SERVICE PROXIES

In many circumstances, such as when a service is actually controlling some piece of hardware, or when network bandwidth is low, it may be inappropriate to download the entire service to a client. In such a case, rather than registering a copy of the service itself with the lookup service, the service will instead register a proxy to itself. The proxy will communicate with the original service probably using RMI, although it is by no means constrained to do so. A client wishing to make use of this service would download a copy of the proxy from the lookup service and make requests of the service via this proxy.

2.5 PARTITIONING OF CODE AND DATA

Java objects representing services or proxies to services are transported over the Jini network in a passive form, to reside first on a lookup service, and then on a client wishing to use the service. On arrival at the client, the Java objects are activated within the local JVM. The passive form is essentially a 'snapshot' of the object's state, taken using Java's serialisation mechanism, and it is this *serialised* data that is moved around the network. However, an object actually consists of code (a class definition) and data, and it cannot be reconstituted from the data alone – the code is also required. This is where a Jini distributed application differs from an ordinary one: the code is not likely to be on the client-side. If this code were to reside on the client-side then Jini would loose most of its flexibility since it wouldn't be possible to add new hardware and software components to a network.

The class definitions are most likely to be located on the server-side. So class definitions for services and proxies to services must also be downloaded, usually from where the service came from. This could be done using a variety of methods, but most commonly an HTTP protocol is used, and classes are downloaded from a Web server. Service objects downloaded from the lookup service by a client contain a reference to this Web server, which is used by the client to access the class files it needs to reconstitute the service.

2.6 LEASING OF SERVICES

In addition to providing a flexible approach to networking, the Jini concept supports robustness and fault-tolerance over a collection of disparate services through leasing. In the scope of this document, robustness is defined as the ability of the system to cope with error during the execution of a process. This applies to all parts of the Jini system, from the ability to connect to a lookup server through to leasing and the use of Jini services.

In Jini, the main use of leasing is for a server to request that a copy of a service be kept on the lookup service for a certain length of time, for delivery to clients on request. The lookup service acts as a *grantor* of the lease and decides how long it will actually create the lease for. Once it has done that, it will attempt to ensure that the request is honoured for that period of time.

The lease grantor passes a proxy to the lease back to the server (the *lease holder*), which allows the lease holder to cancel the lease, renew the lease, or query the time remaining until the lease expires. When a lease expires, it does so silently. That is, the lease grantor will not inform the lease holder that it has expired. It is up to the lease holder to renew the lease before it expires, if it wishes the lease to continue.

Essentially, leasing performs a ‘garbage collection’ role, modifying the Jini federation when services become unavailable. If services that have registered with the lookup service fail to renew their lease, because the service has crashed or has been withdrawn, and the lease expires, then references to the service are removed from the lookup service, and will no longer be available to clients.

Leasing is not restricted to the service-lookup service side. More generally, a service may employ leasing to ensure fair allocation of resources amongst a number of its clients. A client wishing to make use of the service would be required to obtain a lease on that service, which may or may not be renewed after a given period of time. In this instance, leasing forms a bond between the client and the service such that the service is made aware that a client may not exist, if lease renewal does not occur. Conversely, leasing ensures that a client can be made aware of the unavailability of a service via the denial of a lease renewal request.

Later sections of this document cover the ability to use Jini leasing to provide robust systems in more detail.

2.7 DISTRIBUTED EVENTS

Jini provides the ability for clients and services to notify each other of a change in their state by exchanging event objects. The networked nature of Jini has led to a particular event model that differs slightly from the other models already in Java. The differences are caused by factors such as:

- Network delivery is unreliable: messages may be lost.
- Network delivery is time-dependent: messages may arrive at different times to different event listeners. So the state of an object as perceived by a listener at any time may be inconsistent with the state as perceived by others.
- A remote listener may have disappeared by the time the event occurs. Listeners have to be allowed to ‘time-out’, like services do.

The event model can provide Jini federations with an extra degree of flexibility, as it represents an active approach to maintaining federation state, rather than the passive approach undertaken by leasing. For example, a client can register as an event listener with the lookup service, and can be made aware of when a particular service they are interested in becomes available. When the service starts, the lookup service generates a Jini event that is received by the client, and the client can respond accordingly.

2.8 TRANSACTION SERVICE

Transactions are a necessary part of some distributed operations. Frequently two or more objects may need to synchronise changes of state so that they all occur, or none occur. This happens in situations such as control of ownership, where one party has to give up ownership at the ‘same’ time as another asserts

ownership. What has to be avoided is only one party performing the action, which could result in either no owners or two owners. The transaction mechanism ensures:

- *Atomicity*. All the operations of a transaction must take place, or none of them do
- *Consistency*. The completion of a transaction must leave the participants in a 'consistent' state. For example, the number of owners of a resource must remain at one.
- *Isolation*. The activities of one transaction must not affect any other transactions.
- *Durability*. The results of a transaction must be persistent.

In practice, transactions use the two-phase commit protocol. This requires that participants in a transaction be asked to 'vote' on a transaction. If all agree to go ahead, then the transaction 'commits', which is binding on all the participants. If any 'abort' during this voting stage then it forces abortion of the transaction on all participants.

Jini has adopted the syntax of the two-phase commit method. It is up to the clients and services within a transaction to observe the transaction mechanism. An implementation of the two-phase commit protocol is provided by the transaction manager service, supplied by Sun as part of the Jini distribution.

2.9 JAVASPACES SERVICE

The JavaSpaces service, again supplied as part of the Jini distribution, provides a storage facility for Java objects. Using JavaSpaces, Jini clients and services can create and manipulate collections of Java objects. They can acquire leased storage for objects, search for stored objects, and remove stored objects from the space.

In addition to simply providing storage for entities in a Jini federation, JavaSpaces can be applied to other uses. For example, collections of applications can use a JavaSpace as a 'shared whiteboard' for depositing and retrieving objects to support collaborative working. The objects stored in the space can represent work to be done, and thus be used to build producer/consumer-style distributed applications.

2.10 JAVABEANS AND JINI

Jini can be though of as an extension of Sun's JavaBeans component architecture into the distributed computing domain. In JavaBeans, the emphasis is shifted from programming software to building a system by assembling and integrating existing components. Each component, or *JavaBean*, consists of a Java object, or set of objects, which is capable of being manipulated and customised using builder tools. When combined with intuitive visual metaphors, these builder tools have the potential to allow non-programmers to construct software systems from ready-made JavaBeans.

The Jini federation, and its notion of a heterogeneous collection of clients and services communicating via RMI and remote events, is analogous to the interaction of a set of JavaBeans, where the Beans communicate by making direct method calls on each other, or by exchanging events. Moreover, the JavaBeans architecture also accommodates the concept of a service, defined through an interface, that JavaBeans can expose, discover and use.

The future may see the convergence of the Jini and JavaBeans technologies, and the inevitable creation of *JiniBeans* – visually customisable, reusable distributed software components.

3. Jini and Legacy Applications

While Jini is written in pure Java, neither clients nor services are constrained to be in pure Java. They may include legacy code written in C or C++, act as wrappers around non-Java objects, or even be written in some other language altogether. They just need to talk the Jini protocols. This gives a Jini federation flexibility beyond Java, in which the framework is supplied using simple Java mechanisms, but is still capable of supporting the interoperability of disparate services and clients from a variety of sources.

In order to investigate the issues that surround the interoperation of legacy code (non-Java) clients and services within Jini, a Jini federation consisting of Commercial-Off-The-Shelf (COTS) software

products, bespoke applications and pure Java components was constructed. The applications chosen to feature within the federation represent a cross-section of the types of information and office productivity tools that are increasingly utilised within the command-level environment. The aim of the federation was to demonstrate the two main features of Jini, namely its robustness and its flexibility, while at the same time drawing out the benefits and problems that arise when third-party software products participate in a Jini federation.

The degree to which a legacy application's internal structure is accessible suggests different approaches to take when integrating with Java:

- *white box*, where access to the source code allows a legacy application to be significantly rewritten to operate within the Jini/Java framework.
- *grey box*, where the source code of a legacy application is not available, but the application provides its own extension language or Application Programming Interface (API).
- *black box*, where only a binary executable form of the legacy application is available, and their is no extension language or API.

The study showed that with only a small amount of effort, a legacy application can be extended to use the most basic Jini functionality. However, to extend the application to take advantage of some of Jini's most advanced features, a potentially time consuming re-write of the legacy application may be required. It is important to note that trying to shoehorn an existent software system into a role that it was not designed for, is a non-trivial task. If significant changes are required, it can often take more effort to convert existing code than it would to develop a new system. This is not just the case when Jini-enabling a legacy application, but occurs whenever the internal design of a legacy application has to be significantly altered in order to integrate it into a new software architecture.

4. Jini Security and Firewall Issues

4.1 JINI SECURITY

Jini, in its current form, does not address any of the following security issues.

- Service to client authentication.
- Client to service authentication.
- Delegated authorisation.
- Integrity of downloaded service proxies.
- Protecting the lookup service.

While Jini itself does not address these issues, all of the above with the exception of the last one, can be addressed to a degree through complementary mechanisms. Each of these security issues is discussed below, together with possible solutions.

Service to Client Authentication. A Jini client cannot determine where a service is executing nor can the client determine who is operating the service. The client can only make security decisions based upon where the service classes may be downloaded from and any cryptographic signatures associated with these classes.

Java Authentication and Authorization Service (JAAS) is a Java extension that provides the tools to support authentication and authorisation. JAAS currently only works with Java 2 SDK, Standard Edition, v1.3 and later versions. JAAS *may* provide the building blocks necessary to provide service to client authentication. In practice, extensions to RMI security maybe a more effective way of achieving this.

Client to Service Authentication. A Jini service cannot determine *who* is attempting to access the service. The service can make security decisions based upon where the client classes are loaded from, but this does not determine *who* is using it or where they are using it from.

Using JAAS, clients and services can be developed that base their security decisions upon *who* is accessing a service. However, for this to be effective, communication between the client and service must also to secure. Also both client and service must co-operate in the authentication process, implying that Jini clients would need to be developed that were authentication aware.

Delegated Authorisation. Services may subcontract other services to service requests from their clients. There is no simple way for a service to indicate who it is working for, nor is there a simple way for services to grant access permissions to other services based upon who the end user is.

Delegated authorisation is discussed in the latest draft of 'Java Remote Method Invocation Security Extension'. However, there are no indications of when delegated authorisation will be incorporated in RMI (and therefore impact upon Jini).

Integrity of Downloaded Service Proxies. Java has provision for the cryptographic signing of Java Archive files (.jar files), which define the behaviour of a downloaded service. Thus it is possible to detect if a jar file has been intercepted and tampered with. However, there is no provision signing the actual data objects, known as a Service Items, which are stored by the lookup servers. Thus it is not possible to detect if a service item has been tampered with. This could lead to a corrupted service, whose trust is based upon the cryptographic signatures associated with its jar file, being allowed unwarranted access to a hosting machine.

Interception can be handled by using protocols such as Secure Sockets Layer (SSL). Java does not provide facilities for SSL as standard. But implementations of SSL can be developed using the Java Secure Socket Extension (JSSE). This only prevents tampering with items whilst in transit. It does not protect the service items while they are stored in the lookup server. To address this issue, some customisation of the generation of the service items would be needed, so that the actual data was also signed in some way. For example, by adding an attribute to the service item that was based upon a hash value calculated from the service items data. This would allow for the receiving client to perform a check to see whether the service item had been tampered with.

Protecting the Lookup Service. Service items held within the lookup service can be over written by service items generated by malicious services. A malicious service can download all the service items from a lookup server that conform to a particular service template. The downloaded service items include the service ID, an attribute that uniquely identifies that service. The malicious service can then use this ID to register an alternative, malicious service. This has the effect of overwriting the previously registered service item, thus preventing clients from finding the previously registered services. This does not effect clients that have already established connections to the legitimate services.

This is by far the hardest issue to address. It could be addressed, at least in part, by enhancing lookup server to include authentication and authorisation components. In this case, the overwriting of a service item would only be allowed if it was an authorised action from an authenticated source. Precisely how this can be achieved, without undermining some of Jini's flexibility is not clear.

4.2 JINI AND FIREWALLS

For Jini to be fully exploited in a Joint environment, thought needs to be given to the problem of accessing Jini services from different security domains. In particular, a deployed unit may need to connect to a service, where that connection involves one or more firewalls and maybe also a satellite link. This is illustrated in Figure 4-1.

The issue here is not only can we access Jini services across a domain boundary, but can we achieve this without weakening the security of the domain.

Possible approaches to operating Jini through boundary security devices such as firewalls have been explored, and Figure 4-2 details an architecture designed to have minimal impact on the firewall. Here the assumptions are that RMI is tunnelling through the firewall – over http or https. No other additional assumptions are made about the firewall.

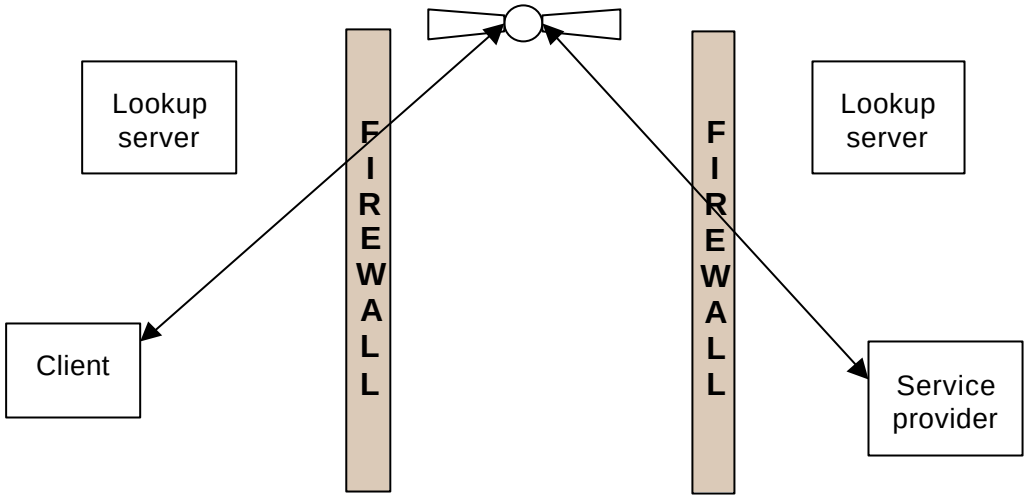


Figure 4-1: Deployed Jini

The steps in getting service proxies through a firewall, numbered as shown in Figure 4-2 are:

1. Service provider registers service with Lookup server, specifying that the service is exportable.
2. Tunnel service registers with the lookup server, downloads all service items that are marked as exportable, and registers an interest in future changes to these service items and the arrival of new exportable service items.
3. A tunnel listener, on the outside side of the firewall, connects to the tunnel service and the tunnel service passes it the exportable service items from the lookup server;
4. Tunnel listener registers the exportable service items with the lookup server outside the firewall, specifying that the service has been imported.
5. Client contacts its local lookup server, and downloads the service items that it requires.
6. Client and service provider interact via RMI (this can be achieved by tunnelling RMI requests using the http protocol).

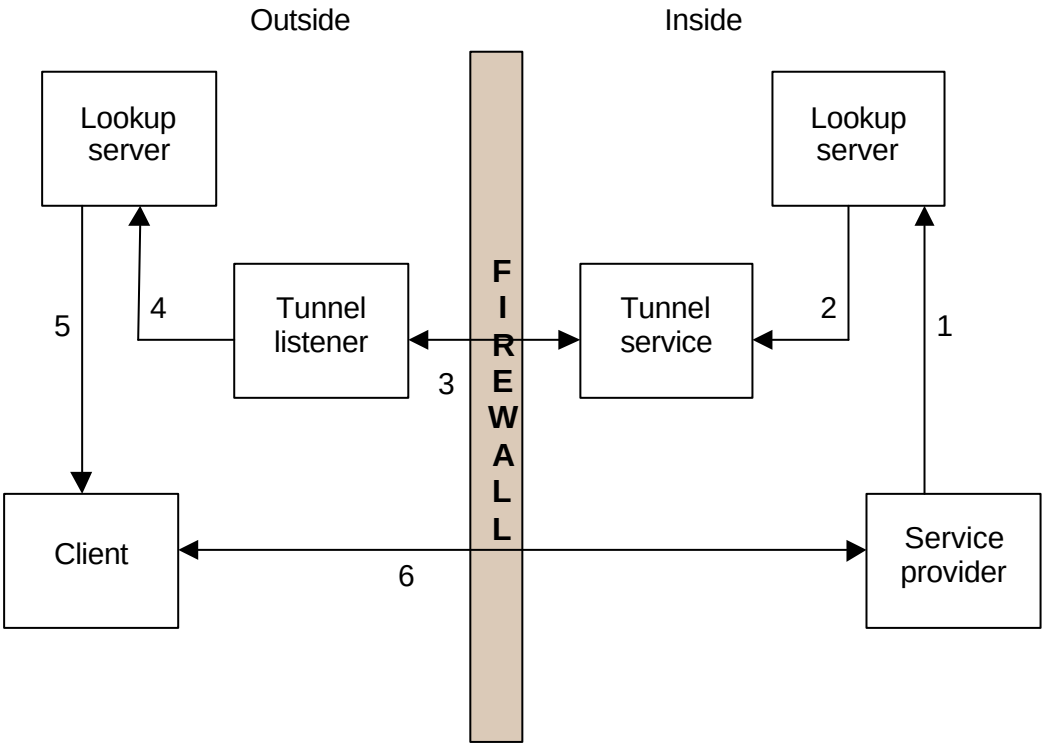


Figure 4-2: Jini Tunnelling

This type of approach is suggested because:

- It protects the lookup server, inside the firewall, from external connections. This is necessary because the lookup server is vulnerable to overwriting attacks, which could deny service or provide malicious services to clients inside the firewall;
- The client should see no semantic difference between using a service that is on the same LAN to using a service on a different LAN. However, if http tunnelling is required, communication between client and service provider will be slower than that experienced between the client and local services.
- The service provider can decide whether it wishes to allow its services to be accessed beyond the firewall. This is indicated by explicitly adding an exportable attribute to the service item published at the lookup server. The tunnel service in this architecture only exports service items that are marked as exportable.

In addition, the client can choose to use services that are within the same domain in preference to those that have been imported (for performance reasons). This choice can be made due to the presence of an imported attribute, associated with imported service items.

5. Robustness

5.1 NETWORK ISSUES

Jini provides no specific mechanisms for dealing with an unreliable underlying network, and as a result, the underlying network and network protocols can greatly affect the ability of Jini to function. For the purposes of this report, TCP/IP and UDP are assumed to be the only protocols in use.

By far the biggest single factor on the robustness of the network, and thus Jini, is the physical embodiment of the network. In reality, this could be a physical network cable connecting computer to computer, or an airborne transmission over, for example, a radio modem.

TCP/IP provides a network layer on which Jini can function. As part of its functionality, it automatically ensures that packets are received in the correct order, packets arrive in their entirety, and packets arrive uncorrupted through error checking. Any lengthy gap between packet arrivals can be assumed to be a network error or timeout problem.

Jini is at the mercy of both the physical network and the layers of TCP/IP. Any inability of either of these to function in any capacity will result in the inability of Jini to function.

5.2 PROTECTING LOOKUP SERVERS

A Jini network relying on a single lookup server has a single point of failure and this has a large bearing on robustness.

Multiple lookup servers can greatly improve the chances of the Jini network surviving network errors. The most typical lookup server error that might be encountered is the inability to physically reach a particular lookup server through the network. The easiest, and probably most effective at combating network problems, is a solution involving a degree of redundancy.

The use of more than one lookup server, can be easily implemented. When the network is working at optimum performance, the services registered are mirrored on each server. When a client encounters a network situation where communication with one server is limited or non-existent, the other server(s) can be used to access services.

In the event of a complete ‘net split’, where a sizeable network falls into two smaller networks due to a communication failure between the two, services registered on the unavailable network eventually lose their leases on the lookup server and only valid services are left.

In this situation, there is no 'main' lookup server. Servers either side of the net-split hold references to the same services that were available before the split. These services should still be available for lookup by

clients after the split. Only when their leases expire do they become unavailable. By using leases, a simplistic 'self-healing' Jini network is created.

Alternately, an implementation using inactive redundancy could be used. In this instance, a single 'main' lookup server exists, but with inactive servers ready to take over should the main server fail. The exact logic behind the events prompting a take-over by an inactive server is left to the designer.

Due to the use of a ‘Service ID’, which should be unique to every service available, lookup servers do not accidentally re-register a service as network links go down and come back up.

5.3 LEASING

When a service is used un-leased, no attempt is made to ensure that access is allowed to the service, it is assumed that all access is allowed.

An un-leased situation has serious robustness issues. Without leasing, during a period of inactivity by the client (or service) for any length of time an error can occur and no parties involved would be aware of this fact. An attempt to contact a service after a long period could lead to the supply of old information or the attempted use of a service that is no longer available.

It is suggested that an un-leased situation only be used when a client is only likely to lookup a service, use it immediately and never (intentionally) use it again.

When leasing is in place, the fact that a lease to a service can be issued could be construed as a signal that Jini, the service and the network are all currently working correctly.

A lease can exist for any length of time and, in general, the shorter the lease is the sooner an error with the service or network can be detected. It should not be forgotten that a short lease time would result in increased requests for leases over the network and, thus, more network traffic.

Leasing also allows for control over resource management. A resource can theoretically allow an unlimited number of leases but in reality, most services will have the capability to deliver to only a finite number of clients. Using leasing, a service can allow leases to the point where it considers itself at maximum capacity and can refuse leases until some have been freed up. The actual implementation of the logic behind granting leases is left to the developer.

5.4 TRANSACTIONS

A transaction is a method whereby all data resources required by an operation are asked to ‘lock’ themselves, i.e. not allow access by other operations, until the locking operation has performed the necessary manipulation on the data resource and has released its locks.

Transactions can help robustness by ensuring that concurrent access to resources occurs in a controlled manner. When an execution requires more than one resource, all are locked before any transaction takes place to ensure that data is current and cannot be corrupted.

Whilst gaining information from the various network resources, it may be necessary for an application to obtain information from multiple sources at the same time. Network limitations may mean that some resources can be contacted quicker than others. Without transactions, data for an easily contacted source may have been updated or invalidated before the more difficult sources have been queried.

With the use of transactions, all resources are ‘locked’ from performing operations that might invalidate data before the application tries to access the data. None of the resources will allow data manipulation by other clients until the application has completely finished thereby ensuring data integrity.

It should be noted that a transaction that tries to lock a large number of resources is more likely to fail, if a single resource is unable to comply, than a transaction with a small number of locks. In some situations, transactions can fail indefinitely.

5.5 EVENTS

Remote events are very similar to ordinary events in Java, except they are designed to be sent on a network. They allow for asynchronous communication between a Jini service and a client.

The use of RMI requires synchronous communication, which may not be desirable over a noisy network. Responses may be slow or never arrive causing RMI calls to fail.

When using Remote Events, the update takes substantially less time to perform and a response is not required. A client that is behind a noisy network or has even dropped from the available network altogether will not cause the service to fail, but an event cannot be guaranteed to reach its target.

6. WANS

The issues faced by the use of WANs (Wide Area Networks) are not completely dissimilar to those on a LAN (Local Area Network). Jini itself provides no specific features to cope with issues presented by a WAN, but gives the developer all the tools necessary to easily build in robustness to help with any adverse situations that may arise.

6.1 BRIDGES, ROUTERS AND HUBS

Since the TCP/IP layer should effectively mask the existence of any of these kinds of physical networking devices, Jini itself should have no dealings with them. Any problems with these networking areas cannot be overcome by Jini and are within the realm of the networking, not Jini.

6.2 LEASES

It should be taken into consideration that a packet/request sent over a WAN would implicitly take longer to reach its destination than on a LAN because of the physical distance and the systems through which it might pass. Thus, lease times should be increased accordingly for the difference in travelling time. Lease times that are too short could result in the lease having expired before it can be used.

A suggested technique to cope with lease times could be to apply for a lease of a specified time and if the arriving lease has already expired, apply for a new lease of a longer period and keep doing this until a valid lease is received. This assumes, of course, that the lease grantor is amenable to granting a lease of the specified duration.

Lease renewal times also need to be considered, the time taken for a renewal request to reach its intended target and then for the resulting lease to return to the client should be no longer than the time to the end of the current lease, otherwise a client can be left without a service until the new lease arrives.

6.3 TTL (TIME TO LIVE) AND JINI

Although Jini has no control over the TTL, which is part of the network layer, it should be considered when designing a Jini network. The TTL is the number of network ‘hops’ a packet should cover before is it considered out of range and continues no further on the network. For example, a packet with a TTL of one will only travel to the next logical point on the network and will not be relayed any further.

A network with machines using high TTL figures could be prone to flooding the network if a large number of services and clients try to make communication at the same time. Packets may migrate to a point on the network where they have no real relevance and services appear on lookup server where they may be, conceptually, of no use. For example, a printer appearing on a lookup server the other side of the planet may not be desirable.

Conversely, a network using low TTL figures may find that services are not available because packets are not migrating far enough to reach the lookup server.

7. Scalability Issues

There are three main bandwidth issues that need to be addressed with regard to Jini and scalability:

- The overhead due directly to Jini functionality (i.e. look-ups, registrations, leasing, etc)
- The bandwidth required to dynamically move clients, services and proxies around the network

- The extra overheads incurred when the network consists of a variety of domains, LANs, WANs and firewalls.

Each of these is now considered in turn.

7.1 JINI OVERHEAD

The Jini look-up requests and registrations are:

- In general, relatively small packets of data.
- Only required once (unless the network fails or reconfigures in some way).

Of some concern is the situation where a service registers itself in its entirety (i.e. provides a copy of itself that can be downloaded by clients, in order to be executed locally). Depending on the application these classes may be of significant size. This can be alleviated by designing the system in such a way as to limit the use of services which have to be run locally and/or limiting their use to local domains to avoid having multiple copies of these services propagated throughout the system (i.e. registered with multiple look-ups). Sensible use of Jini 'groups' can be used to achieve this, since services can be restricted to only register with look-up services belonging to particular groups.

Leasing does pose more of a problem, as despite the fact that lease requests and renewals contain little data, they do (by definition) need to be resent on a regular basis. Careful consideration will need to be given as to the length of lease supplied by different services. The requirement for robustness within the network needs to be balanced against the increase in bandwidth that this will incur. Where possible services which are non-critical (particularly those that expect to have many clients) should offer long leases to reduce the number of renewals.

One further option is to include leasing times within the scope of a Network Management service. This would enable the network to ask services to increase the length of leases (system wide) whenever congestion occurred, or to reduce lease times (thereby increasing the robustness of the system) if spare resources were available.

7.2 DOWNLOADING CLIENTS, SERVICES AND PROXIES

Clients and services can be extremely large applications. Where possible, thin proxies should be used to access the applications (i.e. small interfaces allowing RMI access to the services across the network). There are situations however, where downloading an entire service may be advantageous:

- The service may require access to local files.
- If the service will be accessed a large number of times, the bandwidth saved by removing the need for repeated calls over the network may outweigh the one-off cost of transferring the application.
- In situations where the network connection cannot be guaranteed, running the service locally helps to make the system more robust, since the network link only needs to be available for the time taken to download the application.

Similarly, having dynamic, downloadable client applications could offer significant benefits to the system (e.g. offering true 'plug and play', and thereby reducing the logistics problem by being able to carry 'generic' spare PC's, without the need for a local software/hardware expert). This needs to be assessed against the increase in network traffic that this would entail.

It should be borne in mind that in many cases, the clients / services may only need to be downloaded once, and therefore the effect on the network loading can be controlled (possibly by scheduling these downloads to take place during 'quiet' periods) and minimised.

Again, a sensible approach to the design of the system can help to alleviate this problem, such as providing alternative 'lightweight' services / clients, which can be offered as a substitute if the network is congested or if the requestor and data are separated by several subnetworks / domains / firewalls.

7.3 HETEROGENEOUS NETWORKS AND SECURITY

Running a system over a series of networks separated by firewalls will inevitably increase the loading on the system.

- There may be a need for gateway services to be provided or additional proxies to allow cross – network access to the Jini applications.
- Services may need to register with multiple registries in order to be visible to all clients.

If the networks are included that are heavily bandwidth limited, this will increase the reliance on a Network Configuration/Management service to allocate and police bandwidth usage within these areas. This will incur a management overhead, albeit a relatively small one.

In most cases these problems can be avoided by ensuring that the services required by clients are located within the same local domain. This will require multiple copies of applications over the system as a whole, but this does have the beneficial side effect of increasing robustness (i.e. in emergencies the network could be reconfigured to temporarily route clients to a copy of the service elsewhere).

7.4 OTHER ISSUES

Lookup Service. MarshaledObject's, (or service proxies) are stored in a flat file. This will hamper performance if the lookup service is required to store huge amounts of object proxies. This is also the case where proxies are stored in memory. To improve upon this it is recommended that lookup servers be designed to store particular types of services or to support particular groups of clients – which again highlights the need for sensible use of Jini 'groups'.

RMI is the current RPC method implemented in Jini. RMI's reference layer keeps connections open for each client. Therefore, every service that is running (and activated for the case of Activatable objects) will require a connection. This will affect operating system (OS) resources by increasing the number of file descriptors, or handles in use. Maximum limits depend on the OS used. This reinforces the recommendation stated above, i.e. that the loading of clients and services needs to be carefully balanced and spread over the Jini network. To avoid this, for very large federations of services, RMI could feasibly be replaced with another protocol for interaction between proxies and services.

Java Spaces provide a powerful tool for distribution of state and a common data storage area, while providing developers with synchronisation support. This simplifies development but can degrade performance if a Space is made available globally, since a Java Space can only reside on one machine – therefore forming an access bottleneck. Limiting access to the Java Space is one option to avoid this problem (i.e. only allowing 'local' applications to share data in this way). Another is to attempt to distribute the Java Space, by building services which mirror data from Space to another – this is not directly supported within Java / Jini at this time.

Access control. Since the discovery of services is handled via the lookup service, this can be modified to provide basic access control functions. For example, if the network is becoming overloaded:

- clients can be forced to use local and / or 'lightweight' services – simply by not providing the locations of any services that are less 'network friendly';
- certain low priority clients may be temporarily denied access to any services within the network;

- access to ‘resource hungry’ services, such as Dynamic Clients may be blocked (since these will tend to require relatively large amounts of bandwidth when setting up the client to use the requested service).

It should even be possible to recognise that a particular service is being accessed by a large number of clients, and the lookup may be able to redirect future clients to an alternative or even start up a new copy of that service, to balance the network loading.

8. Military Impact

An architecture based around Jini like technology could offer the following benefits to a military CIS:

- Increased robustness.
- Ease of configuration / maintenance.
- Simpler logistics (through using generic components).
- Extensibility / flexibility.
- Support for network management services.

In addition, this type of architecture provides a route for increasing the integration of legacy tools and at the same time, gives a firm basis for developing future systems from highly modular collaborative components.

This section will outline how this might be achieved, and what implications this might have on future CIS.

8.1 Increased Robustness

As the pace of modern warfare increases, military CISs are under pressure to provide more information, with greater accuracy, in shorter and shorter time-scales, in order to allow commanders to operate within the decision cycles of their opponents.

This in turn increases the reliance on automated and semi-autonomous, high technology CIS infrastructures and services. The reliability and robustness of future systems is therefore of paramount importance.

As discussed elsewhere in this paper, Jini provides mechanisms for detecting the loss of services / clients, and facilities to ‘self-repair’ the Jini federation in the event of failures.

The leasing mechanism can be tailored to provide rapid detection of the loss of critical components within the network, and replacements can be rapidly brought on-line to take their place.

The Jini lookup service allows clients / services to rediscover each other (or discover replacements), and reconnect after a failure, without any external intervention.

There are single-points of failure within the system – such as the lookup services themselves, but even these can be rapidly replaced and immediately used by any surviving clients / services.

8.2 Ease of Configuration / Maintenance

Since the architecture can be robust, reconfiguring the system, or temporarily removing sections of the network can be handled by the existent Jini functionality. Services can easily be added, updated, moved across the network or removed entirely, and clients will automatically find them (or replacements) and use them (if appropriate). This greatly reduces the need to update clients if changes are made to services – this is especially true if Dynamic Clients are provided.

This would be of particular benefit for entities such as mobile headquarters, which will need to take the network apart and rebuild it again, whenever the HQ moves to a new location.

8.3 SIMPLER LOGISTICS

The Dynamic Client concept provides a true 'plug and play' environment. This provides a mechanism for clients to access services that they had no prior knowledge of. This is advantageous in itself as it allows new or updated services to be easily added into the network, without having to upgrade all of the existing clients.

It also means that generic 'spare' workstations can be held in stores, and used as necessary, without the need for a specialist to set up that machine to match the client's particular requirements.

8.4 EXTENSIBILITY / FLEXIBILITY

The Joint environment assumes a large number of information sources are available, many of which will be located in positions vulnerable to hostile action. It is likely that at least some of these sources will be transient in nature (i.e. they will periodically transmit data and / or be available for interrogation).

A Jini infrastructure provides support for exactly this kind of situation, allowing clients to detect, locate and access services as and when they appear. It is also a relatively simple matter to direct clients to secondary sources (i.e. data sources which may be less accurate, up to date and / or comprehensive), if a primary source is temporarily unavailable.

This also applies to clients as well as services. As military personnel take over different posts, they may require access to different data and applications. The sort of architecture described here (especially if Dynamic Clients are provided) allows staff to easily utilise any of the networks capabilities, without requiring extensive updates to any workstation they may have. Obviously a stringent security architecture needs to be in place to prevent this freedom / flexibility from being abused.

8.5 SUPPORT FOR NETWORK MANAGEMENT SERVICES

A Joint environment is likely to consist of a variety of networks, separated by different types of firewall. As such, any infrastructure must be able to cope with restricted bandwidth and access. Also, since the network is likely to be in a hostile environment, bandwidth and access constraints may well become more restrictive at various points during the course of the operation.

This is covered in more detail in the section on 'scalability'. However, it is worth noting at this point that Jini may be able to provide support for Network Management services that can assist with these areas.

For example, all Jini services are given a unique service ID when they register with a lookup service. This is guaranteed to be unique, since it contains the IP address of the host for the service that is registering and the lookup service.

This implies that by tracking the availability of services and taking note of their service id's, some information about the state of the network can be inferred. If all of the services from a particular machine or local domain become unavailable, then this should lead to the conclusion that a hardware failure has occurred (either a workstation or some part of the network connectivity). This allows a rapid detection of network failures, and may trigger automated self-repair responses.

Since the discovery of services is handled via the lookup service, this can be modified to provide basic access control functions. This can be used as a security mechanism (i.e. restrict requests for certain services to particular trusted clients), or as a Network Management function.

As was stated within the 'scalability' section, Network Management functions may be provided which increase / decrease leasing times as the network load fluctuates, or restrict access to certain services or areas of the network (or even block low-priority clients) if resources become too limited.

It is even possible to provide 'dummy' services within the lookup that can be used to help protect the network from intrusion. The lookup services are in an ideal position to monitor clients access patterns and attempt to identify suspicious behaviour. In such cases it can be advantageous to allow the intruder to continue to access the system unaware that they have been detected, until their location (and possibly

purpose) can be determined and / or countermeasures put in place. The lookup service could connect the intruder into 'safe' false applications which mimic real services on the network, to limit the threat of damage / theft whilst this process takes place.

Note that none of this functionality is directly provided by Jini, but there are building blocks available which could form the basis for implementing such capabilities.

8.6 SYSTEM INTEGRATION

As has been discussed previously, it is relatively straightforward to 'wrap' any legacy tool and connect it into a Jini federation. Depending on the openness of the application (and whether access to the source code is available), different levels of integration are possible – but almost all tools would seem to benefit to some extent from being connected into the network in this way.

If this type of architecture is selected as a basis for future military CIS however, the major benefit will occur in the longer term when applications have been developed that have been designed to take full advantage of the enhanced capabilities this type of integration can offer. Future systems may consist entirely of modular components that are connected (or collaborate automatically) as required to form 'virtual applications'. With a Jini-like infrastructure, any Joint system would be ideally placed to take advantage of this type of application.

8.7 CONCEPT OF OPERATION

A new member of a HQ's staff should be able to take any generic personal workstation and be able to connect into the CIS network at any location. Setting up the machine could consist of installing a single 'lightweight' Java application - which provides access to the Dynamic Clients available within the Jini federation.

Once this is complete, the officer should be able to select any service types that they require, and have their machine automatically updated to access them. They should now be able to locate and utilise any service that they need to perform their role – and be automatically reconnected to an equivalent service (or secondary service if none is available), if network failures disrupt their original connection.

If a particular service does not exist, it may even be possible to build a 'virtual application' from a set of modular building blocks, simply by specifying the required capability of the tool (using an agreed specification language).

The network itself should be able to reconfigure in times of stress (i.e. failure and / or overloading), in order to protect as many services and clients as possible (particularly those with higher priority). This should include self-repairing, and automatic restoration of clients / services as the system recovers.

Clients should be able to register an interest in specific events (such as certain data sources becoming available) and be automatically informed if these occur.

This will all be completely transparent to the users of the system – who will simply request data and services, and will be given access to these (or offered alternatives) as soon as they are available.

Maintenance could be fully automated, with clients receiving updates across the network the first time that they try to access a service that has been upgraded.

In a highly dynamic situation where:

- the HQ may be on the move, with the network being disrupted on a regular basis.
- data sources may be highly transient and / or degraded or destroyed by enemy action.
- the CIS may be degraded due to services / domains being unavailable.

The officer should always be able to rapidly connect, and get as much support as can be feasibly provided at any moment in time.

9. Conclusions

Jini is a distributed computing technology that supports the interoperation of hardware and software components in a potentially resilient fashion. A Jini federation consists of three key elements: a service, a client (that wishes to make use of the service), and a lookup service (which allows the client to find the service). By using leasing and the Jini event model, clients and services can adapt in response to the appearance and disappearance of other clients and services, allowing for the creation of flexible and fault-tolerant distributed applications.

Using the Jini/Java framework, it is possible to build potentially robust and flexible networks of clients and services. The clients and services themselves can be written in pure Java, and implement the Jini communication protocols directly, or they can consist of legacy code that participates in the Jini federation by proxy.

The degree to which a legacy application can exploit the features of Jini depends strongly on the interface that the application exposes to programmers. For example, if the legacy application comes under the white-box category (the source code is accessible), then, with respect to licensing constraints, the code can be modified to accommodate Jini to any depth. In such a case, Jini could possibly be used to create a fully distributed version of the application. For applications under the grey- and black-box categories (details of internal operation are not accessible), it is likely that Jini could only be applied at an application level.

This study has shown that with only a small amount of effort, a legacy application can be extended to use the most basic Jini functionality. However, to extend the application to take advantage of some of Jini's most advanced features, a potentially time consuming re-write of the legacy application may be required. It is important to note that trying to shoehorn an existent software system into a role that it was not designed for, is a non-trivial task. If significant changes are required, it can often take more effort to convert existing code than it would to develop a new system from scratch. This is not just the case when Jini-enabling a legacy application, but occurs whenever the internal design of a legacy application has to be significantly altered in order to integrate it into a new software architecture.

Jini, in its current form, does not address any of the following security issues.

- Service to client authentication.
- Client to service authentication.
- Delegated authorisation.
- Integrity of downloaded service proxies.
- Protecting the lookup service.

However, while Jini itself does not address these issues, all of the above, with the exception of the last one, can be addressed to a degree through complementary mechanisms.

For Jini to be fully exploited in a Joint environment, thought needs to be given to the problem of accessing Jini services from different security domains. The issue here is not whether Jini services can be accessed across a domain boundary, but whether they can be accessed without weakening the security of the domain. DERA has explored possible approaches to operating Jini through firewalls, and has proposed an architecture to allow this with minimal impact upon the firewall.

The core of the Jini architecture is the lookup and auto discovery process; additional Jini features are designed to sit on top of these core functions and are optional. Therefore, most of the Jini features that can be used to design a robust system are optional, and Jini does not automatically provide a robust solution. What it does provide to the system developer is a set of tools that can be used to construct a robust system. A Jini network, therefore, will only be as strong as the developer wants it to be.

Any amount of network and service availability checking can be implemented using the Jini helper libraries, but with stronger reliability comes network overheads. For each service, the actual need for

robustness should be considered. For example, a service that only needs information once a day may not need to even hold a lease, the client could simply search for a service each time it requires it. On the other hand, if a service is critical and requires continual updating, and the network can support it, leasing times should be kept short so that any errors can be caught and dealt with.

The Jini interfaces used should also be considered when taking into account robustness. A small ‘proxy’ interface is small in network packets and could be considerably more likely to reach its intended target on an inherently error-prone network than a larger interface or an entire serialised service.

Jini’s ability to control adverse situations in a WAN environment is limited to reporting errors, and relying upon the software developer to handle the error in an appropriate manner.

Any routing or packet control problems are the concern of the network layer.

Delay in packet delivery due to distances travelled should be considered when developing a Jini network. Problems such as lease expiration before use and lookup servers falling out of synchronisation can result from badly developed Jini code.

The actual overhead imposed by Jini is small. If the architecture is designed sensibly, extremely large Jini confederations should be possible, even within a bandwidth limited environment.

At all stages of the development of any system, the benefits of using any Jini functionality needs to be weighed against the bandwidth costs.

The ‘ideal’ situation, of all clients being able to access all services and share data with any entity within the federation, is infeasible for large systems. This is exacerbated in an environment that includes multiple firewalls and network segments with very limited bandwidth.

Towards a Comparison Approach of Architectures for Interoperable Environments

Abdelhamid Elkadhi
Computer Sciences Department and Research Center in Geomatics
Laval University, Ste-Foy, QC G1K 7P4 Canada
abdelhamid.elkadhi@ift.ulaval.ca

Bernard Moulin
Computer Sciences Department and Research Center in Geomatics
Laval University, Ste-Foy, QC G1K 7P4 Canada
bernard.moulin@ift.ulaval.ca

Zakaria Maamar
College of Information Systems, Zayed University
PO Box 19282, Dubai, United Arab Emirates
zakaria.maamar@zu.ac.ae

Summary

In this paper we propose an approach to compare different architectures for interoperable environments. We present four different architectures: three of them use stationary agents and well-known negotiation organizations (use of a broker agent, use of the contract net protocol), the fourth one uses mobile agents and a meeting infrastructure. We propose a comparison function based on three main factors (message type, message size and risk). We also present the approach that is used to compare the architectures considering various scenarios which influence the interactions between the different agents involved in the negotiation process supported by the various architectures.

Keywords: architecture comparison, interoperability, mobility, software agents.

1. Introduction

Technical and procedural interoperability is a critical issue for Command and Control (C2), especially at the level of information and order exchanges (Reddy 1997). This issue is even more critical in a coalition context where several heterogeneous and distributed C2 information systems should interact efficiently and effectively. There are several ways to achieve interoperability between heterogeneous and distributed Information Systems (IS) according to the available technologies (Lange & Oshima1999). A first approach consists of associating each information system with a front-end component which translates the incoming and outgoing messages of this IS; the translation dealing with disparities of message protocols, used languages and ontologies (semantic differences between used vocabularies). An approach to insert front-end components requires that the different interconnected systems send messages through a communication network in order to exchange information and provide services to fulfill requests from other ISs. It is foreseeable that in such an approach, interacting C2 systems will need to exchange high volumes of messages, which might put considerable stress on communication networks, especially in critical circumstances.

Another approach to interoperability is based on recent technological advances in the domain of software agents (Bayardo et al. 1997) (Maamar et al. 1999), and especially mobile agents (McGrath et al. 2000). In such an approach, mobile agents might move across communication networks from one node to another in order to carry out specific tasks in the appropriate locations. An advantage of such an approach is a lighter load on communication networks and a smaller number of exchanges of critical messages between C2 systems. However, every C2 system might not accept to host mobile agents on its site for several reasons, among them security. In such a situation, it might be useful to implement the concept of meeting infrastructure (Maamar & Charpentier 2000) that is able to host mobile agents in a common, neutral and secure environment, each agent

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

representing the interests of a C2 system. In such a meeting infrastructure, mobile agents interact locally in order to carry out various collaborative tasks. Intuitively, we might think that message exchanges in a local environment such as a meeting infrastructure are more secure and less costly than message exchanges across communication networks. However, we have not found in the open literature any study on the systematic comparison of different interaction architectures between C2 systems.

In this paper, we present an approach, which aims at comparing different architectures for interoperable environments. Our application domain consists of several C2 centers which require resources offered by different resource-providing sites in order to carry out different tasks. Conflicts might arise in the reservation of resources. In order to compare various ways of making these C2 centers and sites interoperate, we examine three interoperability environments which are based on different architectures, namely *Broker-Agent* (Genesereth & Ketchpel 1994), *Contract-Net protocol* (Davis & Smith 1983) and *Meeting Infrastructure* (Maamar & Charpentier 2000). The broker-agent architecture and the contract-net architecture involve exchanges of messages between the C2 centers and the different resource-providing sites, while the meeting infrastructure architecture involves the use of mobile agents. We developed a system that enables us to simulate the activities of these different architectures and to track the number and types of the exchanged messages. The simulation can be carried out for different scenarios (we can change the number of C2 centers, resource-providing sites, available resources on each sites, for different periods).

We devised an evaluation function in order to evaluate the proposed architectures according to different scenarios. These evaluations are used to compare the different architectures, considering several factors such as the message type, the message size, and a risk factor associated with the possible loss of a message.

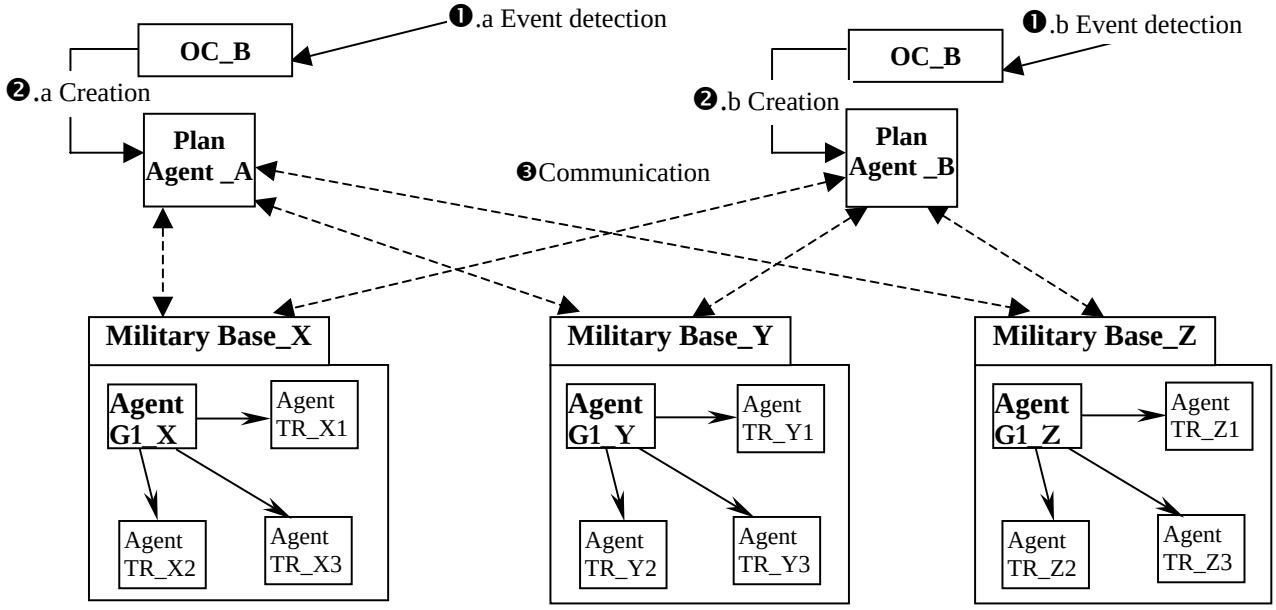


Figure 1: Two OCs and three military bases

2. Useful definitions and application domain

In this section we present some useful definitions as well as the application domain that has been chosen to illustrate our comparison approach.

A *software agent* is a computer program that is able to act autonomously and to adapt to its environment changes in order to reach its objectives (Jennings et al. 1998). The agent receives stimuli from its environment and can act in order to change it. Being autonomous, the agent is able to control its actions and its internal state and to act without the intervention of other human or software agents. An agent can react to changes occurring in its environment. It can be proactive, taking the initiative when deemed appropriate. It interacts with other software agents using communication protocols and with users using interfaces appropriate to each particular communication task.

A *mobile agent* is a particular kind of software agents whose main characteristic is the ability to move from one node to another across communication networks in order to perform its activities. Agent mobility is implemented using a migration function, which enables the agent to suspend its execution on the initial node, to transfer its code to the destination node, and to resume its activities when accepted on the destination node.

For illustration purposes we chose the following application. We consider two systems, each being associated with an *Operations Center* (OC). An OC monitors airspace and must react rapidly and efficiently when unexpected events occur such as the detection of an unidentified plane. In order to react to such events, an OC uses various resources among which interception planes that can be provided by various military air bases distributed over the territory. We can think of the military air bases as service providers to OCs. When several OCs need the same resources for different missions to be accomplished at the same time, there is a conflict that must be solved. To this end OCs must interact with the air bases managing the conflicting resources. In our illustrative application we consider two OCs and three military air bases. We assume that the OCs have a similar organizational structure, similar functionalities, and execute tasks in a similar way. Figure 1 shows the two OCs being able to use the resources offered by three military air bases. In this figure we display the main agents that support the operations of the OCs and the air bases. We distinguish three kinds of agents:

- *PlanAgent*, which represents an OC and plays the role of a service consumer;
- *Agent G1*, which represents an air base and plays the role of a service supplier;
- *AgentTR*: which represents a type of resource supplied by an air base.

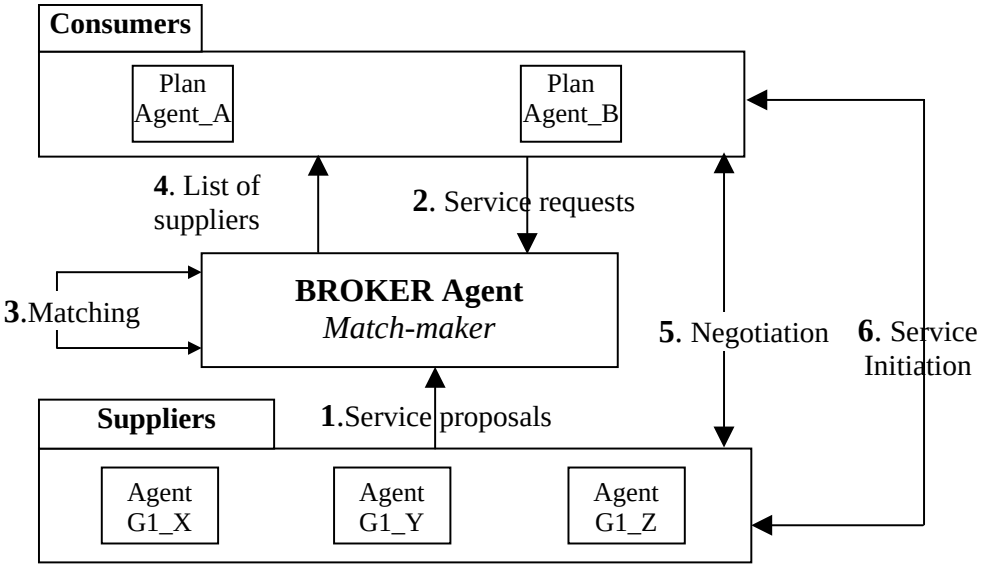


Figure 2: Environment based on a *broker* agent *match-maker*

We consider the typical scenario, which starts with the detection of the intrusion of an unidentified plane in the airspace monitored by an OC (operations 1.a and 1.b in Figure 1). The OC creates a PlanAgent able to deal with this kind of event (operations 2.a et 2.b). The PlanAgent will have to make reservations of resources needed to support the OC's tasks. To this end, it must communicate with the agents representing the military air bases (operation 3). The following agents represent each military base. In each base an Agent G1 (agent G1_X, G1_Y, G1_Z in Figure 1) supervises the other agents of the base. Each TRAgent (denoted AgentTR in Figure 1) manages a kind of resources (such as F18 or DOUGLAS DC3 planes) in a military air base.

The military bases and their agents supply services and the PlanAgents of the OCs consume these services . Hence PlanAgents need mechanisms to select the best suppliers on the basis of different criteria such as minimizing costs for using the resources (for example, minimizing the distance between the selected base and the operation theater).
In the following section we examine four interoperability environments that can be used to implement the interactions between the suppliers and consumers of services. Three of these environments are based on stationary agents and the fourth one uses mobile agents.

3. Presentation of the interoperability environments

In this section we present the four interoperability environments that we will compare. To this end, we analyze the interactions that occur between the agents in each environment.

3.1- Environments using a Broker-Agent

A *Broker Agent* is a system that mediates interactions between suppliers and consumers of services. It receives proposals from suppliers and requests from customers. Then, it matches proposals with requests. Finally, it provides customers with a list of suppliers offering the requested services. We call such an agent a *Matchmaker*. After receiving the list of suppliers the consumer negotiates with them and choose one that provides the most appropriate services. Figure 2 shows how this approach applies to our application domain. We have two consumers (OCs) and three suppliers (military bases). In Figure 2 we represent the various types of interactions and the chronology of operations taking place in this environment. Let us notice that in this environment all messages are exchanged remotely.

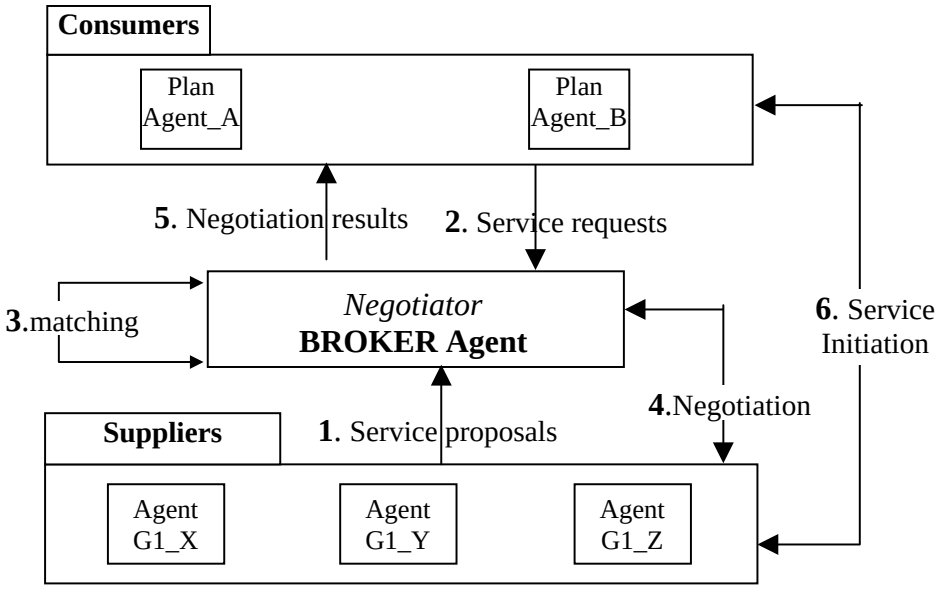


Figure 3: Environment based on a negotiator *broker* agent

In order to ease the PlanAgents' tasks, we can imagine another type of broker agent that would return to each consumer the best offer, instead of the list of potential suppliers. This type of broker agent known as a negotiator broker-agent carries out all the negotiations on behalf of consumers with the suppliers. Although such a broker agent can become a communication bottleneck, it may be suitable for environments where designers wish to limit the number of remote interactions. Figure 3 shows how this architecture can be adapted to our application domain. We use the same conventions as in Figure 2.

Let us now consider another environment in which every consumer can interact with every supplier. We do not have a broker agent anymore and the consumer and supplier agents have the capabilities to negotiate directly. We chose to use the *Contract-Net* protocol as a mechanism that regulates the interactions in this environment (Davis and Smith 1981).

3.2- Environment using the contract-net protocol

Contracting processes in human organizations inspired the Contract-Net protocol. Agents coordinate their activities through contracts to accomplish specific goals. An agent, acting as a manager, decomposes its contract (a task or a problem) into sub-contracts to be accomplished by other potential contractor agents. For each sub-contract the manager announces a task to the network of agents. Agents receive and evaluate the announcement. Agents with appropriate resources, expertise, and information reply to the manager with bids that indicate their ability to achieve the announced task. The manager evaluates the bids it has received and awards the task (sub-contract) to the most suitable agent, called the contractor. Finally, manager and contractor exchange information during the accomplishment of the task.

Figure 4 shows how this approach applies to our application domain. PlanAgents (representing the OCs) play the role of managers and AgentG1s (representing the military air bases) are potential contractors. In our case it is not necessary to decompose tasks since each contracts corresponds to a request for resources. In Figure 4 the dashed arrows correspond to manager's announcements and plain arrows represent contractors' bids. All the messages are exchanged remotely. The managers announce potential contracts (step 1). In Figure 4 the rounded arrow (step 2) corresponds to an "internal negotiation" which takes place in each military base in order to choose which sub-contract to bid for. The contractors send their bids (step 3).

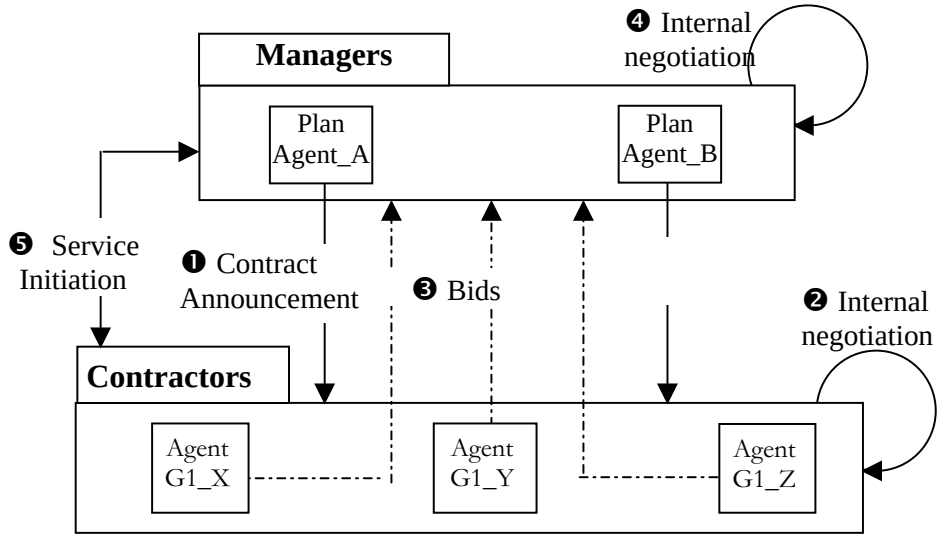


Figure 4: Environment based on the *Contract-Net* protocol

The rounded arrow (step 4) corresponds to an "internal negotiation" which takes place in each OC in order to decide which bid to select. Finally, a request for initiating the requested service is sent by each PlanAgent to the selected AgentG1 (step 5). Using the *Contract-Net* protocol permits to solve the problem of bottleneck that arises when using broker agents. Agents are free to negotiate as they wish, i.e. no third party is involved.

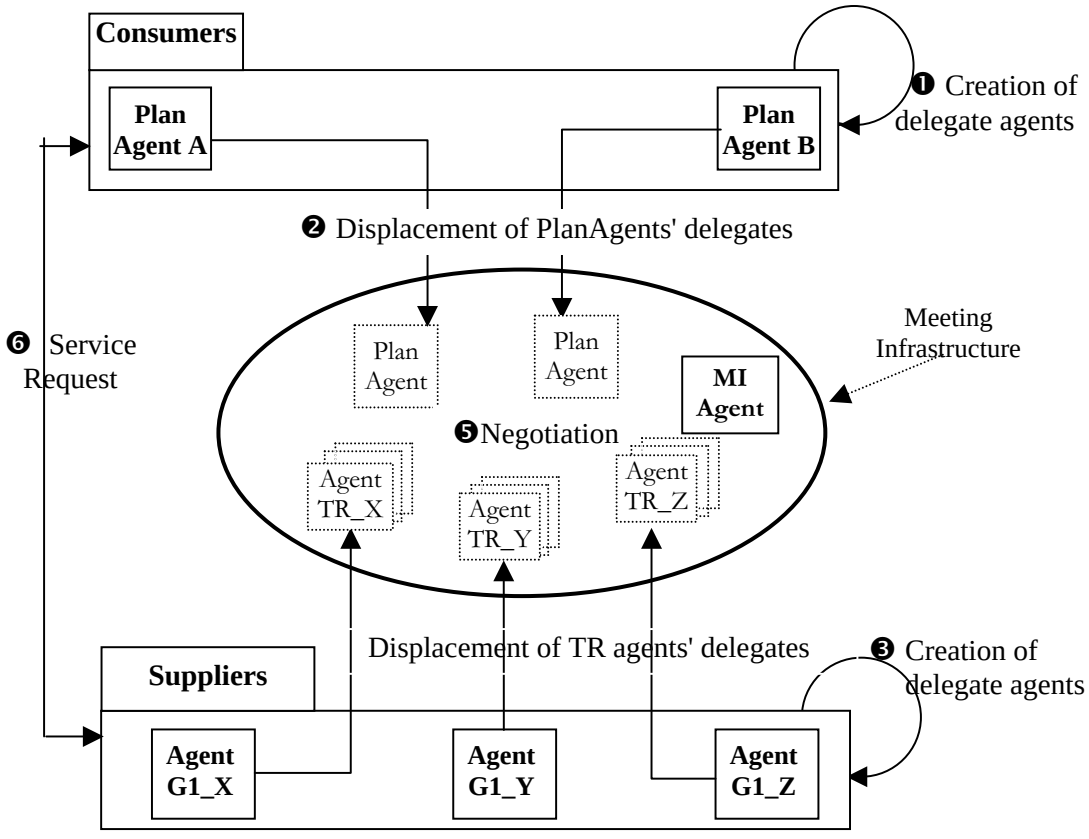


Figure 5: Architecture based on a meeting infrastructure

However, such an approach requires that a large number of messages are remotely exchanged. This might be a problem in environments with a low bandwidth and in which security is an issue. In order to deal with this problem, we consider in the next section a fourth environment based on a meeting infrastructure in which agents will be able to move and to negotiate locally using the *Contract-Net* protocol.

3.3- Environment using a meeting infrastructure

The *meeting infrastructure* is a workspace in which suppliers and consumers can meet in order to negotiate. All messages are locally exchanged. It is clear that a meeting infrastructure can be used if that negotiating agents have mobility capabilities. In such an approach most interactions are local, while a new service must be supplied which enables the suppliers and consumers to send delegate agents to the meeting infrastructure.
Figure 5 represents how such an approach applies to our application domain. An agent called MIAgent manages the meeting infrastructure. This agent controls the access to the workspace and monitors the agents' behavior, which are located in it. Figure 5 illustrates the chronology of operations that enables a consumer to select the best supplier for a given resource this consumer requires. A PlanAgent located in an OC which needs some resources creates a delegate PlanAgent (rounded arrow of step 1). Then, this delegate agent migrates to the meeting infrastructure (step 2). In the same way the suppliers create delegate agents that migrate to the meeting infrastructure (step 4). More precisely, each TRAgent of each military base creates a delegate agent (step 3). Delegate agents of Plan Agents and of TRAgents can then negotiate using the contract net protocol (step 5). Consequently, all the negotiation messages are exchanged locally. When a negotiation between a PlanAgent and a TRAgent is completed successfully, the agents must send messages to their respective parents in order to inform them (this step is not displayed on Figure 5). When such an agreement is reached, the PlanAgent (on the OC) interacts with the Agent G1 of the contracted military base in order to get the proposed service (the corresponding messages are exchanged remotely). Introducing a meeting infrastructure is a means to reduce the number of messages exchanged remotely, while providing the flexibility offered by the use of a negotiation protocol such as the contract net. Moreover, in such an environment it is easier to ensure a good level of security for the exchanges.

4 – Comparison Method

In this section we present the method that we propose to compare the four architectures.

4.1 Evaluation function

We identified the different kinds of messages that the agents could exchange during their interactions in our application domain:

- **Proposal** to use a resource
- **Counter-proposal** relative to a proposal.
- **Acceptation** of either a proposal or a counter-proposal.
- **Definitive refusal** of either a proposal or a counter-proposal.
- **Weak refusal** of a proposal.
- **Modification** of a proposal.
- **Announcement** of a service by a supplier.
- Sending the list of **suppliers** (for a *broker agent match-maker*).
- Sending the **results of a negotiation** (for a *broker agent negotiator*).
- **Inform / update** (message exchanged between an AgentG1 and a TRAgent).
- **Identification** of an agent by the IR Agent.

The four proposed architectures are compared on the basis of the number of exchanged messages and the types of messages that are exchanged. Indeed, on the providers' side resources allocation (scenarios, algorithms) is dealt with in the same way by the four architectures and does not influence the comparison. However, the way agents negotiate is typical of each architecture. Hence, it is relevant to compare the number and types of messages exchanged during the negotiation. Formula 1 expresses the evaluation function.
In function f, the number of sent messages is associated with a weight a_i. Each message type has a corresponding weight that depends on the following factors:

- Message transfer: local or remote.
- Message size.
- Risk associated to sending a message (possible interception, confidentiality of message content)

$$f(arch) = \sum_i [\, a_i \times nbmsg_i \,]$$

f : Evaluation function.
arch : architecture to be evaluated.
i : Message type .
a_i : Fixed coefficient characteristic of message type i.
$nbmsg_i$: number of exchanged messages of type i.

Formula 1: evaluation Function of an architecture

Let us mention that we will not take into account in the comparison of the architectures the phase which aims at initiating the requested service after the negotiation phase. Because the service initiation phase is carried out in the same way for all the architectures (the same number of remote messages is exchanged between consumers and suppliers), it has no influence on the comparison.
Obviously, local messages are preferred to remote messages because a large number of remote messages might reduce an architecture's efficiency. However, other factors should be taken into account. Message size must be considered in the comparison because large messages might have an effect of an agent's processing and might induce delays for information transfer. The message size is related to the number of parameters of the message. The risk factor is related to the confidentiality of the information contained in the message and is of interest especially in military applications. For example, a proposal to use certain planes has a greater importance than a simple acceptation message. In other words, the risk of having a proposal intercepted is higher than the risk of having an acceptation intercepted.
We assume that each of the three factors (message type, message size and risk) can be computed independently of the other factors. We also consider that the message type is more important than the other two factors because it greatly influences the time required to deliver the message. The importance of the two other factors depends on certain weighting coefficients used to compute the weight a_i that will be presented later. Let us denote the three factors in the following way (*i* represents the message type):

Local/remote message : ***Li*** ; Message size: ***Ti*** ; Risk: ***Ri***.

We choose the following values :

Li = 1 if message i is local; *Li* = 4 if message i is remote.

The number of parameters[1] of a message of type i est denoted *Ni*.

$$Ti = \frac{Ni}{4}$$

Ri takes its value in [1,2,3,4] depending on the importance of message content

We chose to keep the value of the three factors between 1 and 4. *Ni* can take a maximum value of 16 (in the case of a proposal there are 16 parameters). Hence, *Ti* can take a maximum value of 4. *Ri* takes a value among [1,2,3,4]. For each message, the value of Ri depends on the importance and confidentiality of the message content (risk of being intercepted). For example, Ri=4 for proposal and identification messages, Ri=1 for acceptation and weak refusal messages. Formula 2 is the function used to compute the weight a_i.

$$a_i = C1 \cdot Li\,(C2 \cdot Ti + C3 \cdot Ri) \quad \text{with } \mathbf{C}i \text{ a positive constant.}$$

Formula 2: computation of weight a_i

As we mentioned earlier, *Li* has a greater importance than the other two factors, namely T_i and R_i. This is the reason why it is multiplied by the weighted sum of the two other factors. Each of the three factors is weighted by a positive constant Ci. This will enable the user to adjust the relative importance of the factors as he wishes. Finally, let us mention that there are different ways of comparing the four architectures. We propose to use Formula 1 in order to get a global evaluation of each architecture and to compare them accordingly. However,

[1] At the beginning of this section we listed the different message types that our agents can exchange. Due to space limitation, we cannot present the detailed structure of each message type in this paper. However, we can mention that the message types have different numbers of parameters.

a user might be interested in a more detailed comparison such as comparing the number of exchanged messages for each message type. Our system allows such a detailed comparison.

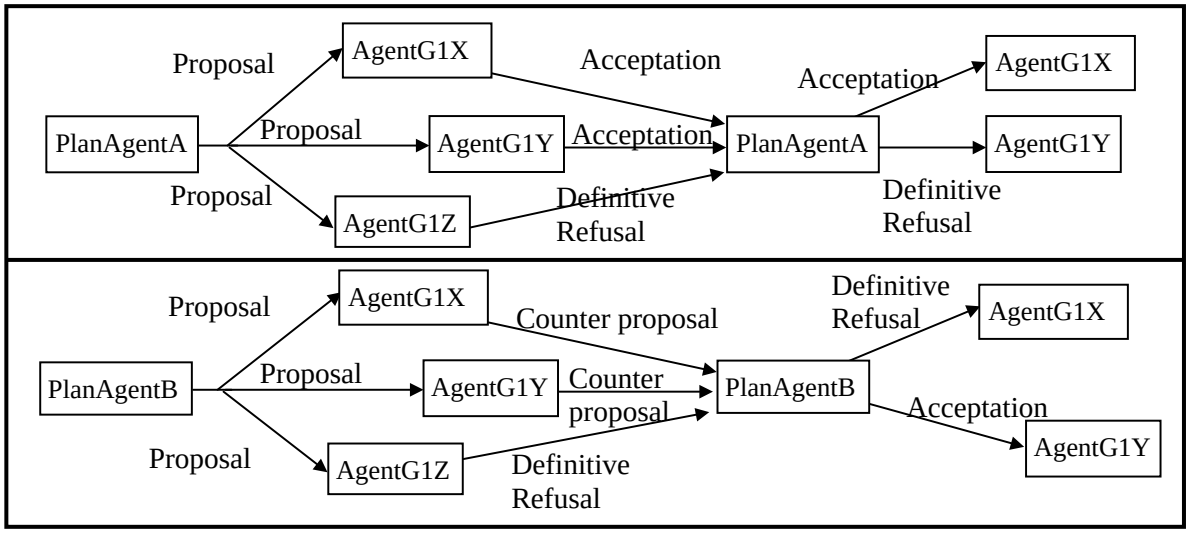


Figure 6: A scenario for message exchange

4.2 Comparison approach

In order to compare the architectures we considered several scenarios corresponding to different possibilities of message exchanges between the providers and consumers of services with regard to the resource allocation problem. Figure 6 presents such a scenario. In this figure rectangles represent agents and arrows represent the exchanged messages. The upper part of Figure 6 shows the messages resulting from a proposal generated by PlanAgent A and the lower part presents the messages resulting from a proposal generated by PlanAgent B.

In fact, a given architecture may be more appropriate for certain scenarios and less efficient for others. For this reason the comparison approach is composed of two steps. In the first step, a number of resource allocation scenarios are randomly chosen. Each scenario is executed with each architecture and evaluated according to Formula 1. These results provide a global measure of the performance of each architecture. In the second step the four architectures are evaluated for each scenario and the comparison provides the relative performance of the architectures for this scenario. For example, the scenario of Figure 6 is applied to the four architectures which are evaluated using Formula 1. Then, the architectures are compared. In this paper we present the evaluation of only two architectures: one which is based on the contract-net and the other which is based on the meeting infrastructure. The results are presented in Table 1 and Table 2 where all the exchanged messages are accounted for. Each table provides several elements for the corresponding architecture: the types of exchanged messages, the number of occurrences and the weight a_i for each message type and the value of function f. In order to compute the weights we set the weighting coefficients to 1 (C1=C2=C3=1).

For the evaluation of the architecture based on the meeting infrastructure, we assume that one of the three AgentTRs is already in the meeting infrastructure and that AgentPlanA has already its delegate in the infrastructure. Consequently, only three agents will have to move to the infrastructure.

Message type	**MsgNb**	**a_i**	**MsgNb * ai**
Proposal (remote)	6	32	192
Counter-Proposal (remote)	2	23	46
Acceptation (remote)	4	13	52
Definitive refusal (remote)	4	10	40
Information / update (local)	3	4	12
Consultation of resource availability	3	4.25	12.75
Resource availability	3	4	12
$\sum$ MsgNb :	**25**	**f (arch)**	**366.75**

Table 1 : Evaluation of the architecture based on the contract net

This assumption reflects the fact that a delegate agent which has moved to the meeting infrastructure for a negotiation, stays there in order to participate in other negotiations. Hence, the migration of a delegate of an agent is done only when the agent has no delegate in the meeting infrastructure.

Message type	**MsgNb**	**a_i**	**MsgNb * ai**
Sending the results of the negotiation (remote)	4	19	76
Migration of a TR Agent	3	32	96
Proposal (local)	6	8	48
Counter-proposal (local)	2	5.75	11.5
Acceptation (local)	4	3.25	13
Definitive refusal (local)	4	2.5	10
Identification of an agent by the IR agent	3	5.5	16.5
Information / update (local)	3	4	12
∑ MsgNb :	**29**	**f (arch)**	**283**

Table 2 : Evaluation of the architecture based on the meeting infrastructure

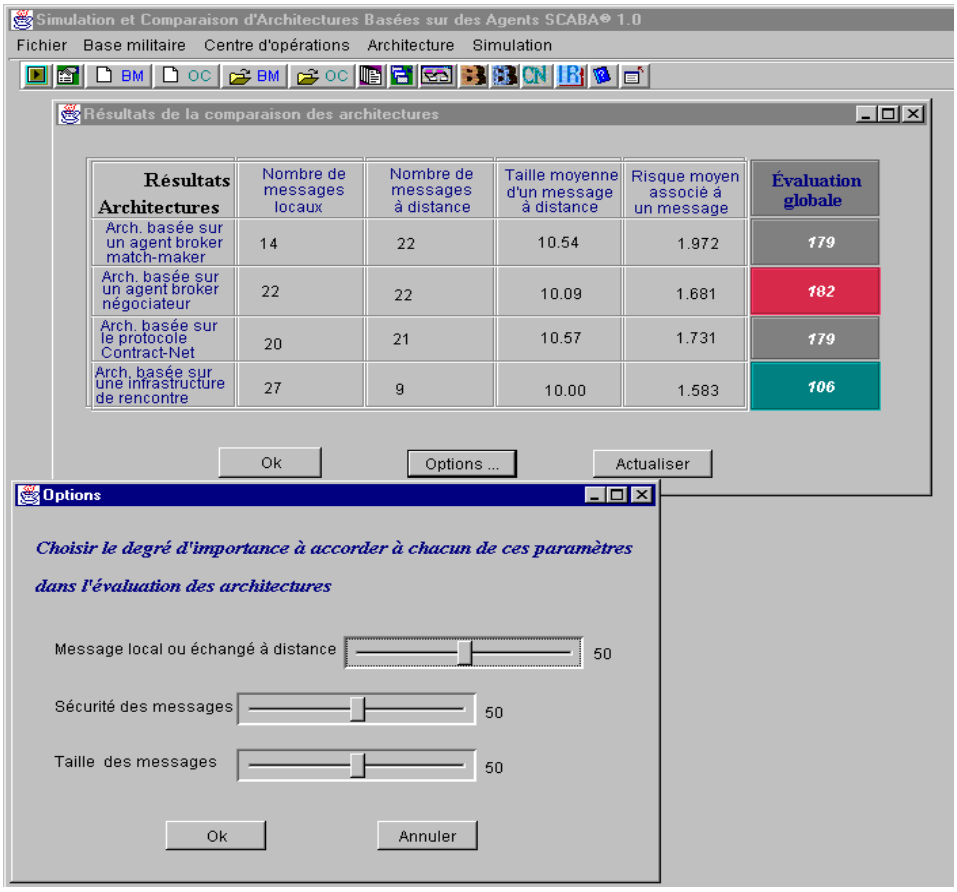


Figure 7: An example of a screen used to display comparison results

4.3 Interpretation of results

We observe that the evaluation of function *f* (Formula 1) is significantly better for the meeting infrastructure (283 in Table 2) than for the contract net (366.75 in Table 1). Consequently, the meeting infrastructure is certainly better with respect to the chosen scenario and to the assumptions that we have set.

Considering the number of exchanged messages, the architectures are quite similar (25 in Table 1 et 29 and in Table 2). However, there is a difference between the types of exchanged messages. In the architecture based on the contract-net, most messages are sent remotely whereas in the meeting infrastructure, most messages are

local. We see that the weights a_i are significantly different, mainly because of the security factor: messages exchanged within the meeting infrastructure are more secure than messages exchanged remotely.
The system that we developed compares the four architectures with respect to various scenarios. It also allows the user to change the values of the weighting coefficients used to compute the weights a_i. In addition to the comparison based on Formula 1, the system enables the user to compare the architectures considering different factors separately: Number of local messages exchanged in each architecture; Number of remote messages exchanged in each architecture; Average size of a message for each architecture; Average risk associated with a message type exchanged in each architecture.
An architecture score depends on the considered factors. Hence, the user has the flexibility to identify the factors that are most important from his point of view and to compare the architectures accordingly (Figure 7).

5. Conclusion

In this paper we proposed an approach to compare different architectures for interoperable environments. We presented four different architectures: three of them use stationary agents and well-known negotiation organizations (use of broker agent, use of the contract net protocol), the fourth one uses mobile agents and a meeting infrastructure. We proposed a comparison function based on three main factors (message type, message size and risk). We also presented the approach that is used to compare the architectures considering various scenarios which influence the interactions between the different agents involved in the negotiation process supported by the various architectures. This project showed that comparing different types of architecture is feasible. It would be interesting to apply this approach on different applications in the military domain.

Acknowledgements

This project is supported by a contract from the Defense Research Establishment Valcartier, Quebec, Canada.

References

(Bayardo et al. 1997) Bayardo, R., Bohrer, W., Brice, R., Cichocki, A., Fowler, G., Helai, A., Kashyap, V., Ksiezyk, T., Martin, G., Nodine, M., Rashid, M., Rusinkiewicz, M., Shea, R., Unnikrishnan, C., Unruh, A., Woelk., D., "Semantic integration of information in open and dynamic environments", In Proceedings of the 1997 ACM International Conference on the Management of Data (SIGMOD), Tucson, Arizona, 1997.

(Davis & Smith 1983) Davis, R , Smith., R. G., "Negotiation as a metaphor for distributed problem solving", *Artificial intelligence*, vol. 20, 63-100, 1983.

(Genesereth & Ketchpel 1994) Genesereth, M.R. , Ketchpel, A.P., Software agents, *Communication of the ACM*, 37(7):48--53, July 1994.

(Knoblock et al. 1997) Knoblock, C.A., Arens, Y. et Hsu. C.N., "Cooperating agents for information retrieval", In Second Internl. Conference on Cooperative Information Systems, Toronto, Canada, 1994.

(Lange & Oshima1999) Lange, D. et Oshima, M., "Dispatch your agents; shut off your machine", *Communication of the ACM*, 42(3):88-89, March 1999.

(Maamar 1999) Maamar, Z., "Vers une approche conceptuelle fondée sur des Business Objects pour des CCISs interopérables", *L'objet*, Vol 5, No. 3-4, 367-389, 1999.

(Maamar et al. 1999) Maamar, Z., Moulin, B.,Bédard, Y., "Software agent-oriented frameworks for the interoperability of georeferenced digital libraries on the world wide web: The SIGAL project", In R. Fegeas M.F. Goodchild, M.J. Egenhofer and C.A. Kottman (Edts.), *Interoperating Geographic Information Systems*, 335—354, Boston: Kluwer Academic Publishers, 1999.

(Maamar and Charpentier, 2000) Maamar, Z. and Charpentier, R., "A Business Object-Oriented Environment for CCISs Interoperability", *Journal of Information \& Software Technology*, Elsevier Science Editor. Vol. 42, Issue 3, 211-221, 2000.

(McGrath et al. 2000) McGrath, S., Chacòn D., Whitebread K., "Intelligent mobile agents in military command and control", in proceedings of the Workshop on Agents in Industry, in Autonomous Agents 2000 Conference, Barcelona, Spain, June 2000.

(Reddy 1997) Reddy, P.C., *Joint Interoperability: Fog or Lens for Joint Vision 2010*, Air Command Staff College, Report AU/ACSC/0137C/97-03.

The Requirements for COTS IPv6 Network Applications in Tactical Network Environment

Piotr Gajewski, Artur Bajda, Jarosław Krygier, Jacek Jarmakiewicz
Military University of Technology
ul. Kaliskiego 2
00-908 Warsaw, Poland
E-mail: pgajewski@wel.wat.waw.pl, abajda@wel.wat.waw.pl
jkrygier@wel.wat.waw.pl, jjarmakiewicz@wel.wat.waw.pl

SUMMARY

This paper deals with requirements related to the possibilities and limitations of using the commercial-off-the-shelf IPv6 components in tactical communications and information systems.

It describes the requirements based on military communications network needs analyses. "NATO C3 Technical architecture" by NATO C3 Board Information Systems Sub-committee is the main document used for COTS technology evaluation with respect to their application in military environment. Unfortunately, the COTS evaluation method has not been defined yet. They describe only required standard features such as: maturity, availability and stability.

The quantitative and qualitative assessment criteria become essential from the following COTS exploitation point of view: quality of service, users mobility, survivability, security, interoperability and management possibility. This paper describes all mentioned above assessment criteria and the IPv6 usage by COTS elements.

1. INTRODUCTION

The battle digitalization determines detailed requirements for data transmission between different users as well as command and control systems, which fundamentally support unit mobility.

Therefore, wider COTS products exploitation and their adaptation for military system purposes are needed. Additionally, fast Internet technology development stimulates the usage of such products in military telecommunication networks. Especially it concerns to the IPv6 based tactical communications systems. In such a case the requirement definition on the COTS products for military purposes is needed.

"NATO C3 Technical Architecture" is an important document, which covers guidance for the possibilities of COTS technologies usage [1]. The reflection of the Information Technology evolution in terms of the Off-The-Shelf-Technologies architectural concepts and their availability is the main purpose of the paper. Taking into consideration various requirements, e.g. NATO policies, operational requirements and future technology situation, the document gives, among others, fallowing guidance for military networks:

- awareness of the architectural concepts that potentially meet the requirements;
- vision on the off-the-shelf technology using from a short to a mid-long term perspective;
- provision of recommendations for the design of NATO Information Systems architecture and for the development of the overall NATO Information Systems architectural framework.

The document mentioned above points out the directions in the communication systems evaluation.

2. BASIC TACTICAL COMMUNICATION SYSTEMS REQUIREMENTS

This is commonly known that military communication systems significantly differ from commercial ones. It is associated with special command systems requirements for military communication system. An important susceptibility to realize hierarchical command process in distributed mobile mesh network environment is visible in modern communication systems. Communication systems have to be instantly reconfigured and decentralized, survivable, manageable and redundant. Effective way to improve efficiency of the military applications is achieved using COTS products. Within the military environment there is great interest in using IP technology within communication networks [2].

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Following important requirements and characteristics dedicated for communication systems are taken into account:

- mobility;
- throughput;
- security;
- interoperability;
- manageability;
- survivability;
- quality of service.

Mobility. A communication infrastructure is characterized by dynamic (often rapidly changing) topology. Each network elements can take different place in communication system structure but they have to be put into hierarchical command system. Working time on fixed positions for command posts changes from 15 minutes to a few hours. This time depends on command structure level. At the battalion echelon the working time can be just 15 minutes, but at the brigade level about 4 hours. More time subscribers spend moving. They can move in geographical area but they keep location in hierarchical command structure.

Throughput. Wireless links still possess low capacity. At present, many mobile military systems support data rates ranging from 2.4 to 64 kbps. Systems planed for future usage will support data rate ranging from 2.4 kbps – 2.048 Mbps (and even up to 50 Mbps in the radius of 0.5 km). The radio communications throughput may be even lower while fading, noise and interference condition are considered.

Security. It refers to the ability of protection against unauthorized access to the stored, processed and exchanged information. The above-mentioned requirements apply to information as well as to voice protection and date flow control. The role of security is to protect the data being held by CIS by addressing the following [1]: *confidentiality* (restriction of information to those authorized to see it), *integrity* (preservation the information in its original form unless amended or deleted by authorized people) and *availability* (having access to the information as and when required). Security services are split into four groups: data interchange, communication, operating systems and security management.

Interoperability. The ability of systems, units or forces to provide services and accept services from other systems, units or forces and to use the services as exchanged to enable them to operate effectively together. From the perspective of achieving interoperability requirements, C3 system development is consider based on the three views: operational, system and technical.

Manageability. The ability of the information and communication systems to be properly managed. Manageability can be considered from two perspectives: functional management area and layered reference model. Functional management area includes fault, configuration, performance and security management, while layered reference model applies the communication elements layer, network layer and service management.

Survivability. Service survivability is a primary factor in military operations. The loss of the communication system services availability and its facilities during conflict can cause unimaginable consequences. Communications systems have to be built with such redundancy that allows services provision in intentionally disaster conditions.

Quality of service. This requirement concerns all mentioned above requirements and moreover, the communication services availability, which have to be permanently available for subscribers.

In the future, when military communication systems are built, using commercial elements, the criteria defining standard selection and adoption will be important. Following formal criteria are defined in [1]: maturity, availability and stability.

Maturity of standards depends on the feasibility of standard and technology implementing it. A standard is considered to be mature if it is technically implemented and if the underlying technology is well understood, robust and tested. At the same time a standard is considered less mature when it is implemented with relatively new technology.

Availability of standard depends on the level of adoption of the standard by vendors and on the implementation of the standard within different products. A standard is considered available if 2 or more products exist that implement the full standard and if it is available from different vendors. A standard is considered less available when only one product implements it or if it is available from a single vendor.

Stability depends on the advancement or changes expected or planed for a standard. A standard is considered stable if no significant changes are expected or planned within the next two years. A standard is considered less stables if significant changes or many changes are expected within two years or when incompatibility exists between current and expected or planned releases of standard.

Taking into consideration evolving commercial application process, the standards can be split into two categories: *mandatory* and *emerging*. The mandatory status shall provide formal criteria and required level of interoperability, whereas in the set of emerging standards we can found these standards that probably will be adopted in near three years.

While the formal criteria, for the military domain, define the standards adoption possibility, the standard status definition would be useful in the evaluation strategy development for military communication systems.

Unfortunately, guidance for standards adoption pointed out in [1] is permanently delayed. Especially it can be found in IPv6 protocol set, where commercial standards development leaves behind the standards profiles as well as the commonly presented trends.

3. IPv6 STANDARD FEATURES

Research concerning the next generation protocol for Internet started in 1995, when the RFC's were published [3], [4]. Currently major problems concerning IPv6 implementations are solved.

IPv6 offers following significant features:

- a sufficiently large address space;
- globally unique and hierarchical addressing, based on prefixes (like in CSPN) rather than address classes, to keep routing tables small and backbone routing efficient;
- a mechanism for the network and subscribes interfaces autoconfiguration ;
- support of encapsulation of other protocols as well as itself;
- class of service to distinguish types of data;
- improved multicast routing support (in preference to broadcasting);
- built-in authentication and encryption;
- transition methods to migrate from IPv4;
- compatibility methods to coexist and communicate with IPv4.

Security and mobility efficiency play essential role from military point of view. The security in the IPv6 protocol is characterized by robust procedures based on IPSec (Triple DES). Besides, Ipv6 supports real time services during subscriber's mobility on quite high QoS level.

4. REQUIREMENTS FOR COTS IPv6 APPLICATIONS

With reference to [1] the functional network profile specified for IPv6 has covered 24 standards so far. They are presented in figure 1.

Only this Requests For Comments are included that are promoted to Standards Track category in this profile. Two years earlier, in 1998, in [1] apart from presented RFC's, the RFC 2132 and 2236 were included, which now are not valid. Besides the network layer, the lower layers are shown in figure 1: data link and physical. In physical layer only one wireless technology (IEEE 802.11 standard) is presented. But currently described profile for IPv6 does not support wireless technology. The profile does not define the IPv6 and WLAN interworking possibility (arrow).

Dynamic process of IPv6 standardization can be observed recently. Security and mobility management issue, which are important for military domain, are being solved. Mobility management will be considered in this paper.

IPv6 perform special function for mobility management, which must be adapted to military environment and must allow command process flexible.

We can take into account several requirements for IPv6 protocol applications that are necessary to exploit it in the military domain e.g.:

- compatibility with Mobile IPv6 standard protocol;
- possibility of ad-hoc network building;
- protocol independence with reference to changeable topology;

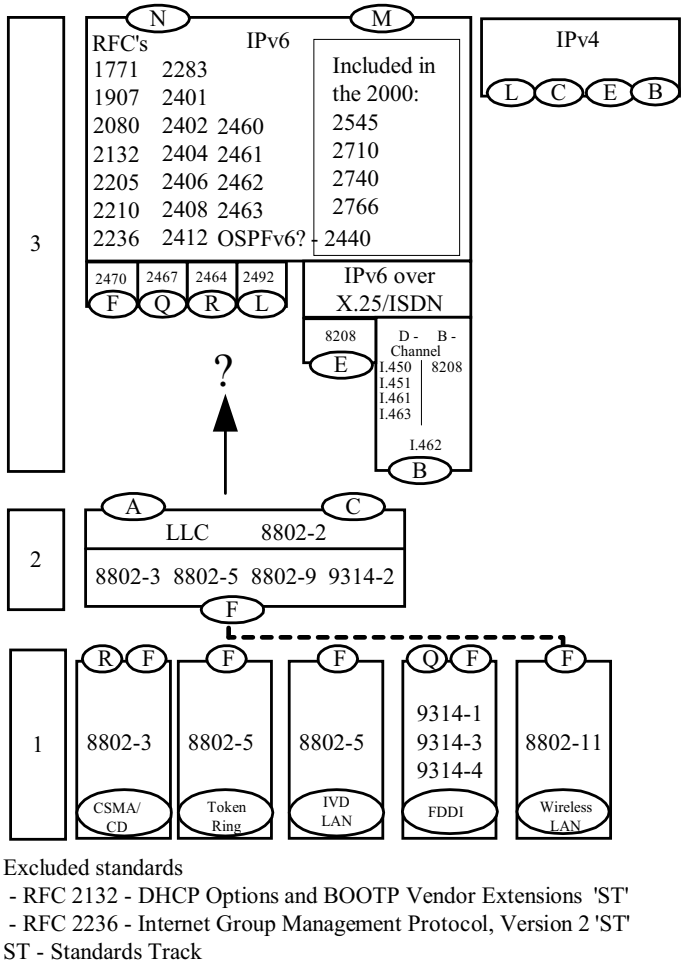


Figure 1. Functional IPv6 network profile

- manageability movements in distributed environment;
- support robust security procedures;
- QoS (Quality of Services) providing and real time services support;
- minimization of the information delivery delay in new location update;
- packet lose elimination during handovers;
- optimal usage of radio resources ensuring a stand-by mode in low power equipment support;
- cooperation with changeable bit rate.

At the current status of Mobility IPv6 development it is significant to adopt guidance defined in [5]. This paper deals with requirements related to the IPv6 protocol and network elements providing mobile functionality for tactical domain.

5. IPv6 IMPLEMENTATION STRATEGY

Strategies of IPv6 components implementation have been developed in many countries. RDEC CECOM presented an attractive strategy for COTS product implementation [6], [7]. This strategy may be summarized as adopt, adapt and developed. It follows, that commercial technology (if it is necessary) must be suited to the environment where it is adopted. Especially it applies to the military domain. Possibility of usage COTS IPv6 or Wireless LAN technology first of all depends on secure susceptibility. Currently offered COTS products that support IPv6 technology is adapted in order to use of Triple DES encryption algorithm, while WLAN products use 128 bit WEP encryption. Besides, Data Terminal Equipment, utilized by the users, can exploit its own encryption methods that are situated at the upper-layer of the reference model.

IPv6 and WLAN technology usage, especially concerning mobility requirement, can have great importance for military domain. IPv6 protocol also supports terminal mobility.

Mobile IP supports terminals that move from one subnetwork to another as packets are being sent, without interrupting this process. Compared with IPv4, IPv6 can provide more mobility support. The mobile node (MN) is a host or a router that changes its attachment point from one subnetwork to another without changing its IP address. The MN access the subnet via a home agent or foreign agent. The home agent (in IPv6 protocol) is situated at the router operating in home network for the mobile nodes, while the foreign agent is situated in the router operating in the visited network. The remote node that internetworks with MN is called the correspondent node. The mobile IP architecture is illustrated in figure 2.

In this example, correspondent node A sends packets to the MN via the home agent and foreign agent. MN replies to node A using standard routing.

Present research is lead to achieve:

- minimal packet delays and handover latency, allowing real time services support;
- elimination packet losses during handovers between mobile node;
- optimized routing;
- support of a large number mobile nodes;
- resistance to failures link or node;
- large amount of mobile node traffic support;
- support of high security level for services;
- ability to support QoS protocols;
- ability to adapt to changes in the network topology.

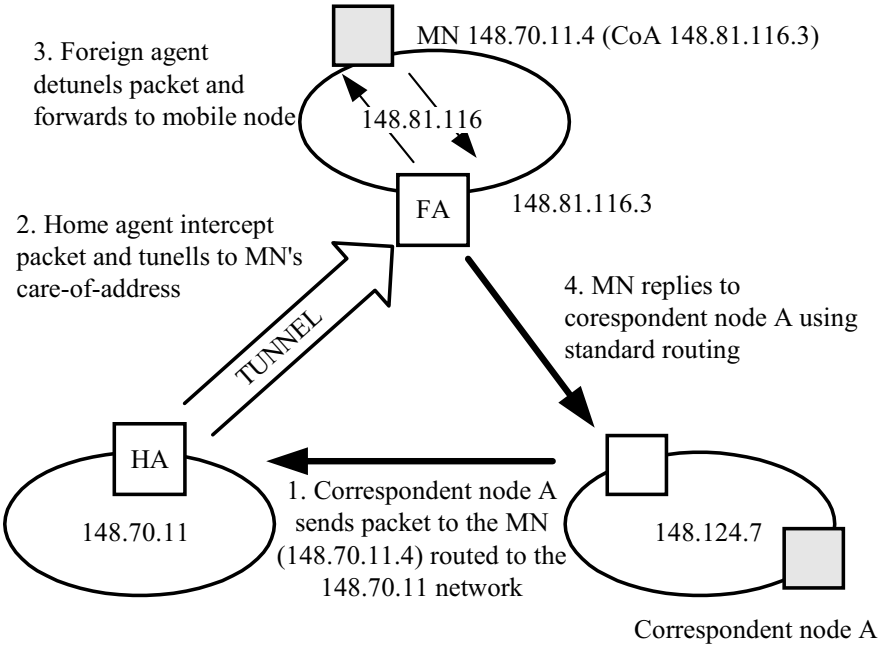


Figure 2. Mobile IP operations

Both mobile and stationary network infrastructure used in military domain (hosts and routers) have to meet following requirements [5]:

- each IPv6 node must be able to process a Home Address option received in any IPv6 packet;
- each IPv6 node should be able to process a Binding Update option;
- each IPv6 node should be able to maintain a Binding Cache of the bindings received in accepted Binding Updates.

In order to make mobile node be able to operate correctly, while away from home, at least one IPv6 router, on the mobile node's home link, must work as a home agent for the mobile node. The following additional requirements apply to all IPv6 routers capable of serving as a home agent:

- each home agent must be able to maintain an entry in its Binding Cache for each mobile node for which it is serving as the home agent. Each such Binding Cache entry records the mobile node's binding with its primary care-of address and is marked as a "home registration".

- each home agent must be able to intercept packets (using proxy Neighbor Discovery) addressed to a mobile node for which it is currently serving as the home agent, on that mobile node's home link, while the mobile node is away from home.
- each home agent must be able to encapsulate such intercepted packets in order to tunnel them to the primary care-of address for the mobile node indicated in its binding in the home agent's Binding Cache.
- each home agent must be able to return a Binding Acknowledgement option in response to a Binding Update.
- each home agent must maintain a separate Home Agents List for each link on which it is serving as a home agent;
- each home agent must be able to accept packets addressed to the "Mobile IPv6 Home-Agents" anycast address for the subnet on which it is serving as a home agent, and must be able to participate in dynamic home agent address discovery;
- each home agent should support a configuration mechanism to allow a system administrator to manually set the value to be sent by this home agent in the Home Agent Preference field of the Home Agent Information Option in Router Advertisements that it sends.

Finally, the following requirements apply to all IPv6 nodes capable of operate as mobile nodes:

- each IPv6 mobile node must be able to perform IPv6 decapsulation.
- each IPv6 mobile node must support sending Binding Update options; and must be able to receive and process Binding Acknowledgement options;
- each IPv6 mobile node must support use of the dynamic home agent address discovery mechanism;
- each IPv6 mobile node must maintain a Binding Update List in which it records the IP address of each other node to which it has sent a Binding Update;
- each IPv6 mobile node must support receiving a Binding Request option, by responding with a Binding Update option.
- each IPv6 mobile node must support sending packets containing a Home Address option; this option must be included in all packets sent while away from home, if the packet would otherwise have been sent with the mobile node's home address as the IP Source Address.
- each IPv6 mobile node must maintain a Home Agents List.

The requirements implementation in military domain allows subscribers mobility support.

6. EXAMPLE OF IPv6 COTS COMPONENTS USED IN MILITARY DOMAIN

Recently developed military communications and information systems are more and more often based on the COTS products or its implementation. Aforementioned requirements for tactical communication systems can be fulfilled using IPv6 protocol supporting WLAN technology. One should agree that direct usage of commercial equipment (without modification and adaptation) might be limited but not impossible. As an example, the application of COTS IPv6 network proposed for tactical military communication system is presented.

Figure 3 depicts a general architecture of IPv6 mobile military communication system. Local area subsystem (LAS) is performed using WLAN technology. HIPERLAN 2 standard was proposed. HIPERLAN 2 is based on the TDMA (Time Division Multiplexing Access) method, which is adapted to VoIP (packetized voice transmission). This feature is not supported by the IEEE 802.11 standards, where CSMA/CA is utilized. Support of subscriber’s mobility is implemented in HIPERLAN 2. Information that mobile terminal can move up to the 36 km/h are presented in [8]; it is enough to ad-hoc communication network organizing on the Command Points (CP). IPv6 mechanisms support subscriber’s mobility both near CP and while they are distant from CP, where SCRA, UHF, HF radio systems are utilized. Furthermore, communications services can be supported in LAS areas by Mobile IP extension, e.g. HAWAII, which are define in IETF RFC’s.

Information (from testbed) presented by Lucent Technologies [9] shows possibility of using HAWAII for voice transmission. Protocol efficiency concerns: handoff latency 5ms, propagation information delay from correspondent node to mobile node 25 ms, total packet losses for voice transmission, less than 1 packet per handoff.

All interchanged information in communication system can be supported using IPv6 protocol (excluded voice services realized in HF and UHF tactical radio network, but including the data transmission supported by WEB technology).

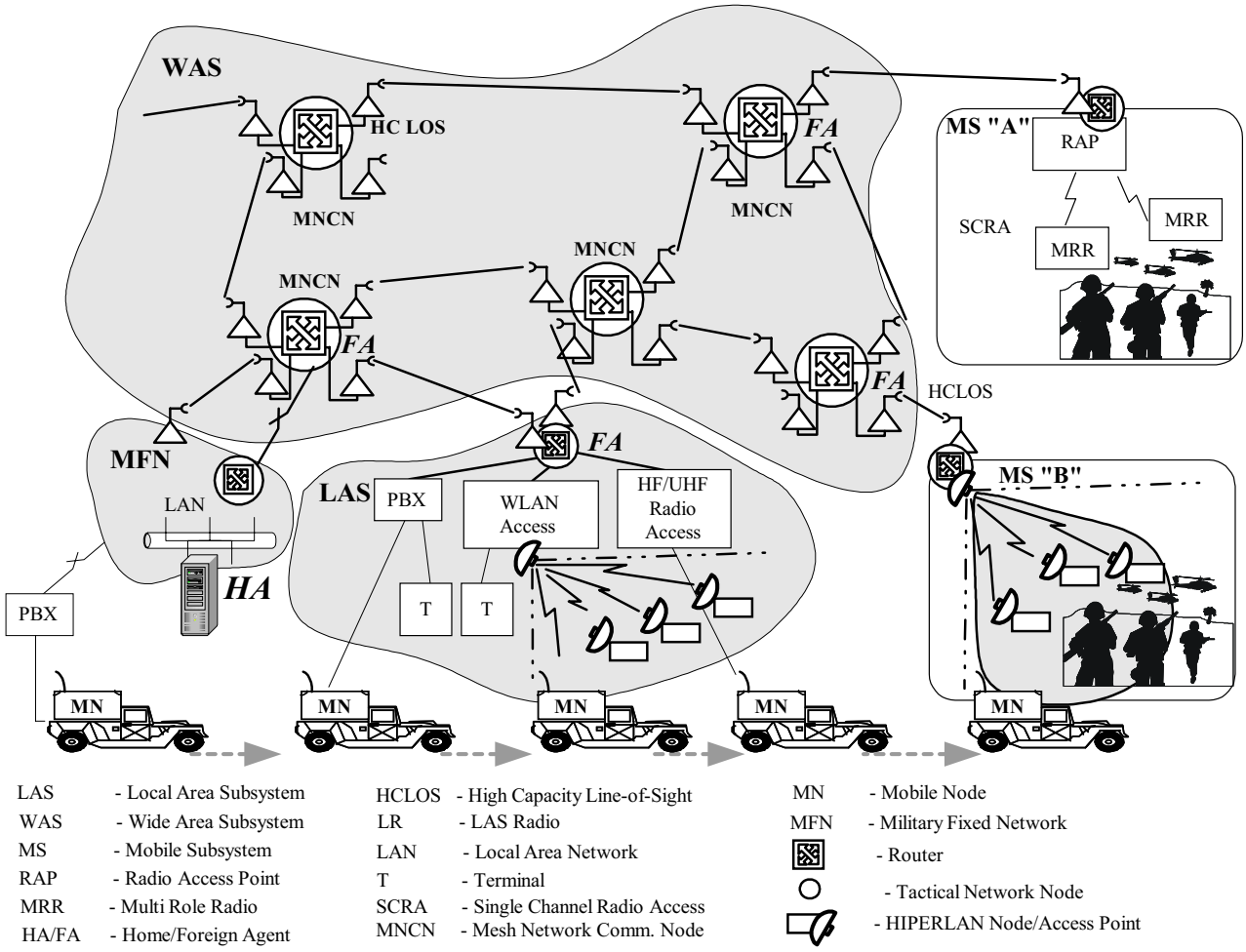


Figure 3. General architecture of mobile IPv6 military communications system

Mobile subscriber's tasks require the communication services interchange in following links:

- between mobile user and fixed or mobile Command Posts, own or other Command Point;
- between mobile user and the mobile Command Posts in the neighbourhood of Command Point in the range of WLAN operation;
- between mobile user and Command Posts distant from Command Point e.g. in combat net radio, using SCRA, UHF or HF network.

In order to achieve this functionality Command Posts and network infrastructure should be supported by the following components:

- mobile nodes (MN) (command and control posts) based on commercial IPv6 application:
  - LAPTOP computers;
  - standardised operating systems, i.e.: MS Windows 2000;
  - WLAN interfaces (HIPERLAN 2);
  - standardised TCP/IPv6 software;
- routers supporting Mobile IP functionality with the HAWAII extension;
- home and foreign agents, placed in IPv6 routers (according to requirements for COTS IPv6 applications);
- combat Net Radio infrastructure (SCRA, UHF, HF network) supported by gateway IPv6 functionality;
- delta/VoIP Gateways at the IP network input.

This network infrastructure is able to support following services:

- subscriber authentication and authorisation;
- access to message handling system support by X.400 protocol and e-mail intranet extension;
- access to combat data bases;
- file transfer protocol;

- HTTP based applications (i.e. HTML-WWW, XML);
- X-Windows;
- voice transmission;
- video connections and conferences.

CONCLUSIONS

The COTS components applied in military domain support the military communications systems cost reduction. From the process of COTS components adopting point of view, guidance presented by NATO NC3 Agency is very important. Unfortunately, commercial technology development is very fast while current military standardisation process keeps continuously behind. For this reason, the military network development process is significantly complicated.

It seems to be useful to expand standardization process from two stages (*mandatory* and *emerging*) to three stages (*mandatory*, *emerging* and *under development*) in order to point directions of COTS technology in development. This could be useful to plan research concerning adoption in longer time perspective.

REFERENCES

1. NATO C3 BOARD Information Systems Sub-Committee “NATO C3 Technical Architecture”, AC/322(SC/5) WP/31, 1998/2000
2. B. Adamson, Tactical Radio Frequency Communication Requirements for IPng, RFC 1677, 1994
3. S. Bradner, A. Mankin, The Recommendation for the IP Next Generation Protocol, RFC 1752, 1995
4. S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification RFC 1883, 1995
5. D. Johnson, Ch. Perkins, Mobility Support in IPv6, Internet draft, 2000
6. D. Oliva, C4ISR Development Philosophy, RDEC CECOM Meeting, Fort Monmouth 2000;
7. J. Barbarello, W.Kasian, US Army Commercial Off-The-Shelf Experience, The Promises and Realities, Proceedings of NATO RTO IST, Brussels 2000;
8. TR 1001 031, Broadband Radio Access Networks (BRAN), HIgh PErformance Radio Local Area Network (HIPERLAN) Type 2, ETSI, 1999;
9. Ramachnadran Ramjee, Thomas La Porta, Luca Salgarelli, Sandra Thuel, Kannan Varadhan, LiLi, IP-Based Access Network Infrastructure for Next-Generation Wireless Data Networks, IEEE Personal Communications, August 2000;

Click here to view PowerPoint presentation; Press Esc to exit

German Air Force Procedures for Implementing Interoperable Information Systems in C², Weapon, and Support Systems to Support NATO Led Combined Joint Task Force Operations

Klaus Kulke
kk Consulting
Hubertusstr.7
D-85521 Ottobrunn

Summary

This paper summarises the German Armed Forces conclusions and activities to establish operational interoperability on future tactical data link networks with forces of other nations. It describes the steps taken by national authorities to meet this challenging goal and it describes the currently achieved status.

The key conclusion is a strict application of interoperability management procedures to develop appropriate platform documents, first on conceptual level and then, on specification level with the subsequent implementation of interoperable platform Information Systems based on a Task Force – centric and Link Network - centric approach.

The approach does consider the NATO Combined Joint Task Force (CJTF) Concept and the 'New ways to do Business' as outlined in the new joint command and control philosophy within the NATO military command structure.

Background

In 1996 the German Air Force initiated a study to investigate the operational and procedural impact of introducing MIDS Link 16 into national command & control systems and weapons system. The study recommended the implementation of a single joint authority to meet the challenge of operational and procedural interoperability of all services and the development of detailed management concepts and solutions with a description how to implement and how to use best the capabilities of MIDS/Link 16.

One area of great concern at that time was a clear description of operational requirements and operational concepts and a clear guidance from NATO how to approach the operational interoperability issue.

A second Link 16 interoperability study from national industry investigated a solution for the development and presentation of user requirements. The 'Lessons Learned' from 'Operation Allied Force' were included to consider NATO's plans for Peace Support Operations (PSO) and recent experiences. The emerging Combined Joint Task Force (CJTF) concept played a major role during that study.

The conclusions and results of the study recommended the development of a 'National Handbook for Tactical Data Links', which would cover an overall interoperability concept addressing all relevant issues with unambiguous and mandatory management procedures and clear guidance.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Solution

A 1st draft of the proposed 'Management Handbook TDL' was developed by industry and provided to national authorities for consideration. It concentrates on the MIDS/Link 16 issue and describes how national generic and platform-oriented concepts are derived from the NATO Conceptual Framework by making clear national statements in terms of national commitments to operations and tasks of the CJTF-concept.

Operational tasks where identified and described as laid down in the Allied Joint Publications (AJPs) for 'Joint Amphibious & Maritime Operations', 'Joint Air and Space Operations' and 'Joint Land Operations'. This was put in context with the provisions for the supply of adequate and timely data by means of new modern tactical data links, which follow a network principle.

The 'Management Handbook' contains Data Item Descriptions (DIDs) to provide guidance for the development of platform Concept of Operations (CONOPS) and Business Models (BusMo) in-line with the Combined Joint Task Force principles. The CONOPS provides a good understanding of the relevant roles and tasks for all members of a specific Task Force. In terms of "interoperable Information Systems", it also provides a good understanding of potential interoperability requirements amongst all participants, based on the doctrine, concepts and procedures of the respective task force concept as outlined in the relevant Allied Joint Publications (AJPs).

The key to the success to operational interoperability between platforms from different nations and different services with different operational roles on a common link network is the development of 'high quality validated Information Exchange Requirements (IERs)'. This is achieved by developing appropriate Business Models for each operator/crew member of an operation centre/platform. Based on the CONOPS and the assigned responsibilities and tasks of each operator, the Business Model would describe the required information for each operator activity and strives to identify data sources and data sinks within that Task Force.

The development of 'high quality validated IERs' is achieved when comparing Business Models of all participant of the Task Force. This helps to determine whether the established IERs for each participant on the net match the IERs of the other participants and will show whether there are 'loose ends' within the information flow. It is expected that additional discrepancies will surface during this process, which would require rectification. This process establishes early operational interoperability of Information Systems for the various platforms. Interoperability in this sense has been defined as "…..providing the correct information to the correct authority at the correct time".

The 'Management Handbook' also contains a Data Item Description (DID) for the development of a Concept of Link Employment (COLE). It determines, which 'high quality validated IERs' are going to be exchanged over what kind of link network and with whom. It defines the applicable network roles and features as well as optional capabilities of the communication terminal, which require support of the platform Information System. As such it tackles the development of Information System Specifications based on a network-centric approach at the very first phase of the development and implementation of systems.

This proposed approach has been named R.O.S.E. and stands for "Requirements-oriented Operational Structured Evolution". It covers the NATO C3 Interoperability Design Domain of the NATO Consultation, Command & Control Interoperability Environment (NIE). It is complementary to TULIP (Through Live Interoperability Planning), which addresses the planned and implemented platform link capabilities.

Concept Implementation Status

German authorities have acknowledged the need for strict management procedures, a joint interoperability organisation and adequate interoperability test networks and test capabilities. This is documented in the ‘Conceptual Framework for Tactical Data Links (TDL) Bundeswehr’ however, the implementation of this overall concept is very demanding and presents a number of risks.

The new German procurement process allows for a demonstration phase to reduce possible risks. There are plans to set up a ‘National Joint Link Interoperability and Management Demonstrator (LIMD)’ to demonstrate and validate the overall concept based on integrated building blocks of Combined Joint Task Force (CJTF) operations.

R.O.S.E. is part of this demonstration. As a matter of fact, it has been applied formally for the first time during the development of the German Air Force (GAF) MIDS Link 16 Combat Search and Rescue (CSAR) helicopter program with excellent results. CSAR Task Force Operations supported by Rescue Augmenting Forces have proved to be a demanding operation. R.O.S.E. allowed the generation of CONOPS, Business Model and Concept of MIDS/Link 16 Employment (COLE) for the GAF CSAR Helicopter in a relative short time.

The application of R.O.S.E. to other national platforms and the subsequent comparison of results recorded in these conceptual documents with same structures and similar contents will further prove the capability and efficiency of R.O.S.E. to produce “high quality validated Information Exchange Requirements (IERs)”.

In accordance with the NATO C3 Interoperability Environment (NIE) Testing Concept developed by the NATO C3 Interoperability Environment Testing Working Group (NIETWG) ….“requirements for testing are primarily derived from operational requirements for interoperable information and communications system in support of C3”. Furthermore, “prerequisites for acquiring optimum NATO C3 Interoperability Environment (NIE) Standards are: high quality validated Information Exchange Requirements (IERs)”.

Platform CONOPS, Business Model and COLE provide the baseline documents for interoperability assessment and testing.

This page has been deliberately left blank

Page intentionnellement blanche

Performance management of C2ISs through QoS

Eric Dorion
Eric.Dorion@drev.dnd.ca
Information and Knowledge Management Section
Defence Research Establishment Valcartier (DREV)
2459, Pie XI North
Val-Bélair, Québec, CANADA
G3J 1X5
http://www.drev.dnd.ca

Summary

This paper emphasizes the importance of performance management of Command and Control Information Systems (C2ISs) in the context of a coalition. It describes some of the concepts that are being developed or used at DREV for ensuring that performance and efficiency in systems can be reproduced and improved instead of applying ad hoc solutions. The concept of Quality of Service (QoS) is adapted to our needs and key technology aspects are considered.

Introduction

Information systems have been used in the military community for decades. While it is generally recognized that these systems help the military commander in his tasks, very little is known about their true contribution or added value. The need of knowing this true contribution is not as important as the one of reproducing it in all systems or improving it. For this to arrive, we need to have a generic understanding of the human decision-making process. This process is the one that military information systems, or more precisely Command and Control Information Systems (C2ISs), are meant to support or improve. While this understanding is important, we also have to consider systems aspects. Although C2ISs differ greatly from one another, it is possible to draw commonalities for all systems and these will be exploited. This article is not about performance of C2ISs: It is about performance *management* of C2ISs. There are numerous solutions that improve systems performances, but strategies to use these solutions in a concerted way are scarce. Strategies and frameworks can only be developed if solid basic concepts are defined. In this regard, Quality of Service (QoS) will be introduced. We argue that QoS is the corner stone concept of performance management. Finally, since C2ISs technology aspects are so important, we will consider some technologies and state-of-the-art tools that will support C2ISs from a performance management viewpoint.

The Human Decision-Making Process

Military information systems are commonly referred to as Command and Control Information Systems (C2ISs). C2ISs support the commander in his decision-making process. One generic theoretical background for this process is the OODA loop (Observe, Orient, Decide, Act) and is represented in Figure 1. C2ISs aim at improving the efficiency of one or more portions of the OODA loop. The OODA loop represents the decision-making process of a commander in a C2 situation. The top half of the loop constitutes the situation analysis part and leads to what is called “situation awareness”. It is composed of 2 phases:

Paper presented at the RTO IST Symposium on “Information Management Challenges in Achieving Coalition Interoperability”, held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

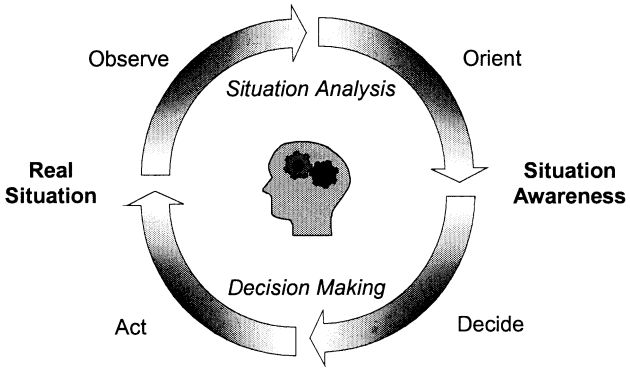


Figure 1: The OODA loop

Observe: Obviously, for the commander to gain situation awareness, he will have to have the current situation presented to him. This is necessary as it will enable him to use resources at his disposal effectively to eventually gain advantage over the opponent forces. In this phase, an efficient C2IS will be one that gives a representation that is perfectly correlated with the ground truth. Of course, this is rarely the case as information is often incorrect or out-dated. However, such information is not necessarily useless and still may prove to be of value to the commander.

Orient: The commander has to look beyond the simple representation of the current situation and ask himself if there are patterns from which he may infer knowledge. A merchant ship may constitute a meaningless piece of information, unless politics, sociology or even history issues lead the commander to think that this ship might smuggle drugs or refugees. His understanding of the current situation will only be complete if he reflects on what he sees and what he cannot see. As some patterns are well defined and recognized (computerizable), and as an overloaded picture may overwhelm the commander, a C2IS may prove its efficiency by easing the commander's task in this phase.

The lower half of the loop is the one in which the commander will actively assign and deploy resources to influence the likely future of what he perceives as being the real situation. Again, it is composed of 2 phases:

Decide: This is the part in which the commander will decide to act on a course of action. The C2IS helps him by suggesting a set of them depending on available resources and other parameters. The performance and efficiency of a C2IS at this stage is related to the accuracy of the suggested choices or courses of action.

Act: In this final part of the loop, the commander issues orders, monitors the progress of ensuing activities according to the plan and corrects minor deficiencies. He even steps forward in the loop and begins a new OODA cycle. The C2IS is the best tool for disseminating timely information, might it be synchronous or asynchronous. Its performance can be easily assessed with simple measures of performance.

C2ISs can support the decision-making process in every step of the OODA loop. At a macroscopic level, C2ISs that support the ease and speed of execution of the OODA loop for as many cycles as necessary are considered to be performant and efficient. This is highlighted in [1].

> *"The military community typically states that the dominant requirement to counter the threat and ensure the survivability of the ship is the ability to perform the C2 activities (i.e., the OODA loop) quicker and better than the adversary.*

> *Therefore, the speed of execution of the OODA loop and the degree of efficiency of its execution are the keys to success for shipboard tactical operations."*

This is the ultimate objective of C2IS performance management. It also constitutes the link between the system and the user. In wanting to perform his C2 duties, the commander, whether he is aware of it or not, specifies performance requirements on the C2IS. In other words, a set of performance guarantees is imposed on the underlying information system. The challenge is to express these C2 requirements in terms of digestible parameters for the C2IS. C2ISs must in turn meet these system requirements.

C2ISs in a Coalition

C2ISs have these particularities that have to be considered if we want to manage their performance adequately. First, heterogeneity is an unavoidable characteristic of these systems, especially in the context of a coalition. Second, the components of these systems (hardware and software) are highly dispersed over network topologies. Performance management strategies have to be developed with these 2 characteristics in mind. Everything else will simply fail with respect to our goal of reproducing performance or improving it.

Introducing Quality of Service

Bearing in mind the knowledge of the decision-making process modelized by the OODA loop and the generic characteristics of C2ISs as systems, we have to give ourselves some tools that will help us devise solid strategies to maintain performance management. We introduce the concept of Quality of Service (QoS) that is a first step in this direction. Of course, QoS is an old concept and applied in different contexts. However, we argue that it should be extended to C2ISs as the best way to deliver performance and efficiency from C2ISs to the military users. We extracted fundamental notions from different definitions of QoS and formed our own that we think might be best suited to support our task.

The Notion of Predictability and Consistency

> *"In the simplest sense, Quality of Service (QoS) means providing consistent, predictable data delivery service. In other words, satisfying customer application requirements."* [2]

The keywords in this definition are "consistent" and "predictable". A service response is said to be predictable when the conditions for its delivery are known over a fixed period of time. These conditions are specified by QoS measures like latency, jitter, throughput, accuracy, etc. Perfect predictability of a service delivery means that the conditions under which the service is rendered are known for an infinite period of time and null predictability occurs when the conditions are never known in advance. Predictability is further discussed in [3].

Consistency is the difference between the nature of an expected response and the actual one. That is to say, the closer the nature of a response is to the expected one, the higher the consistency is. For example, if a service requester expects to receive an image and gets one, then consistency is said to be high. If a string of ten characters was expected and a fifty pages document is actually received instead, then consistency is low. One might think that the strong typing used in object-oriented (OO) programming languages like Java or C++ ensures responses consistency. It does to a certain extent, but the nature of certain responses like streaming video can still affect consistency without violating strong typing. Indeed, if a 30 seconds video stream is expected when a 5 minutes video stream is actually received, then consistency is affected.

A distributed application could take advantage of consistency and predictability by adapting its behavior accordingly to the variations of these properties. Of course, establishing service predictability and consistency is difficult and subject to hot debates, but the key notion here is that distributed applications have to adapt to their environment, much like living beings have to adapt to their environment to survive. Predictability and consistency of service rendering are useful only if the distributed application make use of them by adapting.

The Notion of Guarantee

> *"Quality of Service (QoS) is to the ability of a network element (e.g. an application, host or router) to have some level of assurance that its traffic and service requirements can be satisfied."* [4]

Ensuring QoS in a system (from a performance perspective) means that certain levels of service delivery have to be guaranteed. From thereon, tools must be built, measures must be taken, strategies must be followed and architectures must be devised to support and maintain these guarantees.

The Notion of Management

"Managing the service response of an Internet network is often referred to as Quality of Service..." [3]

"Quality of Service (QoS) is the term that is being used to loosely organize the collection of activities and technology initiatives that have emerged to improve and control network oriented resource management based on mounting experience with distributed, Internet applications." [5]

"For service providers and network engineers, QoS means engineering and managing network resources to deliver performance levels that satisfy their users' expectations." [6]

"Quality of Service is a network "term of art" for describing technologies that allow service providers to manage network congestion rather than simply to add capacity." [7]

Management is a central concept to providing QoS. [8] defines QoS Management Function (QMF) as the collection of smaller QoS mechanisms (e.g. negotiation, admission control, monitoring, etc.) that helps to meet QoS users' expectations and requirements. The QMF is necessary because there is no single solution to ensure QoS in a distributed application. This comes from the fact that:

Information systems are dissimilar between one another. A distributed application may be used in different scenarios. (For example, a C2IS may be used in a search and rescue operation and later in military operations) QoS mechanisms evolve. (e.g. new negotiation process, new monitoring tools, etc.)

QMF is necessary in that it helps achieving predictability and consistency. The methodology developed in this thesis to ensure QoS in military distributed C2ISs from the performance viewpoint is a QMF.

The Notion of End-to-end

> *"To enable QoS requires the cooperation of all network layers from top-to-bottom, as well as every network element from end-to-end. Any QoS assurances are only as good as the weakest link in the "chain" between sender and receiver."* [4]

QoS is really achieved in a distributed application only if it has been addressed at every level of the OSI 7-layer model . Unfortunately, existing QoS solutions do not address all levels in the stack. IntServ/RSVP and DiffServ, the 2 main QoS architectures operate solely at the IP level. From network designers point of view, end-to-end QoS is achieved when packets are delivered to the application level with respect to a certain QoS context. The fact is that all the tools and mechanisms offered at the IP level within these QoS architectures are only good if the application itself specify, tune and shape its own message traffic. This is to say that similar QoS strategies have to be developed at the application level (between the distributed objects and/or within the middleware) to achieve end-to end QoS. This exercise should also be pursued further by encompassing business logic. Applied to the military realm, Quality of Information (QoI) should be delivered to the commander. Indeed, the military does not know how to specify low-level QoS requirements, but he very well knows his needs, and can specify them in terms of MoEs. Common sense shows us that different levels of QoS should be decoupled and renamed according the layer is addresses in the OSI 7-layer stack as suggested in Figure 2. Notice that the QoI level stands out of the stack. At this level, business processes requirements have to be specified. In the military domain, it is addressed with MoEs/MoPs specifications, a language that is spoken by commanders.

Working Definition of QoS

The important notions that we have extracted from the few definitions above should now make it possible to forge our own that is hopefully more complete:

> *"Quality of Service is an encompassing term for the collection of activities, management functions and strategies that aim at guaranteeing the end-to-end, predictable and consistent behavior of network-dependent applications"*

A working definition of QoS is like an enterprise mission statement. Though is does not help in anyway in improving system performance, it gives us the focus necessary to achieve fundamental changes in our approach to performance management.

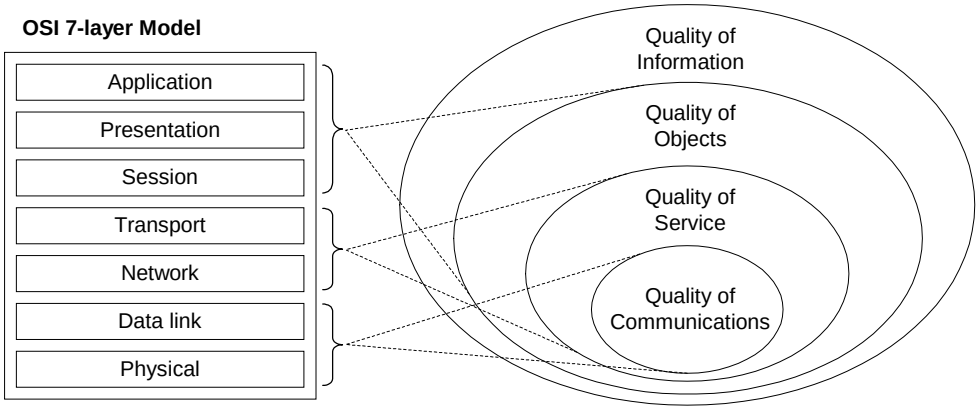


Figure 2

Key Technology Aspects

We gave ourselves both a model for the decision-making process of the commander and a definition of QoS that addresses the important issues embedded in the term “performance management”. We also considered some of the systems aspects that are common for C2ISs deployed in coalitions. It is now essential that we identify key technologies that will support the performance management framework. Work is being done at Defence Research Establishment Valcartier (DREV) to implement a performance management framework for existing in-house systems that use the technologies described in this text.

Measuring Military and Systems Aspects

For a system to meet certain performance criteria, whether they are stated by the user or by the environment, a means of measuring both military aspects and systems aspects is capital. Furthermore, it is necessary to have a measuring system that enables the linkage between military measures and systems measures. The Measures of Merit (MoM) system has provisions to address these issues (see Figure 3). It is a hierarchy of measures ranging from the low-level system measures of performance (MoPs) up to the high-level military measures of effectiveness (MoEs) and measures of force effectiveness (MoFEs). This model described in [9], though not complete, serves as a good basis for our framework.

QoS Supporting Architecture

We gave ourselves a definition of QoS. As we already know, QoS is not a new idea. However, QoS has been historically applied at the network level. We argue that QoS should be a more encompassing paradigm and extend to all levels of the OSI 7-layer model (Figure 2). QoS has nonetheless been given 2 supporting architectures that prevails today, namely, IntServ/RSVP [10,11] and DiffServ [12]. It is beyond the scope of this paper to describe these architectures. However, Figure 4 shows how these architectures are combined in our experiment in order to get the best of both worlds. This approach is also described in [3].

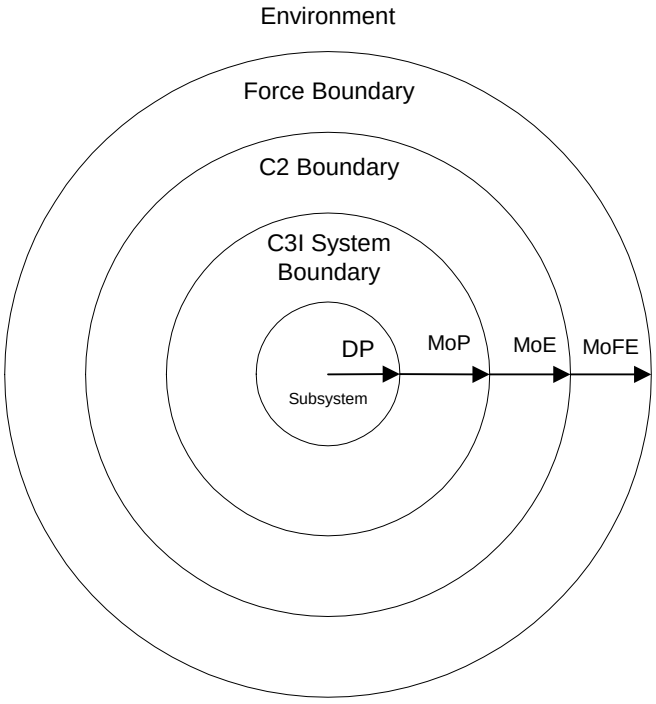


Figure 3

Self-Introspection

Measuring military and systems properties do as much good as these quantities are exploited in order to better the own system's performance. If not, then there is no point in setting up flashy measures. In this regard, self-introspection, self-management of systems components and resources is essential to ensure adequate performance management. Technologies that support this include CORBA, JavaBeans, etc. The idea behind this is to address the notion of management embedded in the QoS definition.

QoS Propagation Mechanisms

QoS is only as good as QoS information is passed along to all components composing the C2IS. It is therefore essential that an adequate QoS propagation mechanism be used. This addresses the notion of end-to-end and predictability in the QoS definition. The CORBA Messaging specification [13] addresses this issue.

Real-Time Systems

Performance requirements almost always fall under two categories: Accuracy and timeliness. Even accuracy requirements eventually influence timeliness. QoS's notion of predictability demands that careful attention should be given to timeliness of requirements. Not that late answers are less useful, but that their lateness should be known in advance. RTCORBA [13] combined with the real-time operating system QNX along with real-time sensible hardware is our choice candidates at DREV.

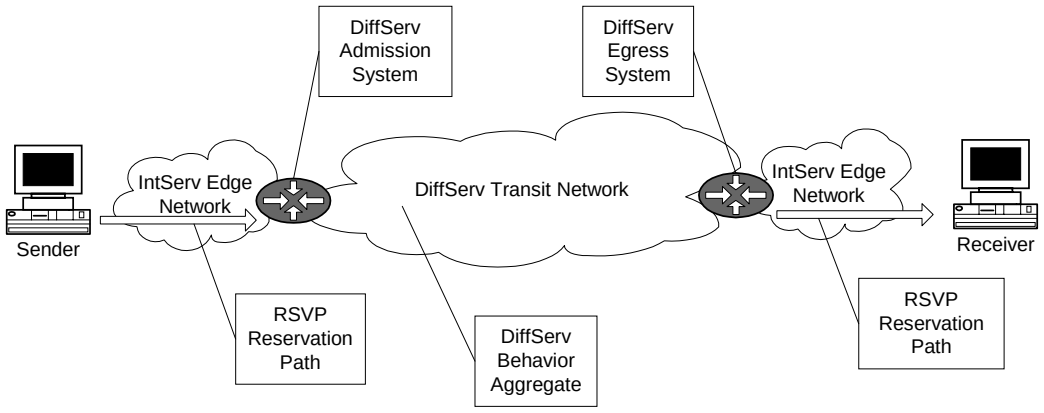


Figure 4

Conclusion

In the information management context, performance management is not a choice: It is a necessity. Information superiority as a fact is also dependant on how the commander perceives his C2IS as a supporting system to his decisions. This is cleary a function of utility and in the end a function of performance. Reproducing performance in different systems and in different contexts depends on how performance management is handled. Ad hoc solutions deny it while solid strategies based on QoS approaches and architectures promote it. The most important however lies in the fact that performance has to be taken into account early in the development process of the system, or if using legacy systems, considered as a major part to maintain it.

References

[1] Stéphane Paradis, Richard Breton, and Jean Roy. *Data fusion in support of dynamic human decision making.* Defence Research Establishment Valcartier, 1999

[2] Vicki Johnson, *The IP QoS faq,* 1999. `http://www.qosforum.com/docs/faq/`.

[3] Geoff Huston. *Internet Performance Survival Guide: QoS Strategies for Multiservice Networks.* John Wiley and Sons, inc., 2000.

[4] Stardust.com, *White Paper – The need for QoS*, July 1999. `http://www.qosforum.com/white-papers/Need_for_QoS-v4.pdf`.

[5] Richard E. Schantz. *Quality of service*. Kluwer Academic Publishers, 1998. `http://www.dist-systems.bbn.com/papers/1998/QoSArticle/`.

[6] Arnold W. Bragg. *Quality of service: Old idea, new options.* IT Professional: Technology solutions for the enterprise, 1 (5):37-44, September 1999.

[7] Editor in chief, *QoS: New term in the IP lexicon*. IEEE Internet computing journal, July 2000.

[8] ISO/IEC. *Information Technology – quality of service: Framework*. Technical Report ISO/IEC 13236, ISO/IEC, Case postale 56, CH-1211, Geneva 20, Suisse, December 1998.

[9] Vernon M. Bettencourt. Command, control, communications, intelligence, electronic warfare measures of effectiveness (C^3IEW MOE) workshop. MORS Report MORS FR 9210, Military Operations Research Society, Fort Leavenworth, Kansas, October 1992.

[10] R. Braden, D. Clark, and S. Shenker. *Integrated services in the Internet architecture: An overview.* Request for Comments 1633, Network Working Group, June 1994. `http://www.ietf.org/rfc/rfc1633.txt`

[11] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. *Resource ReSerVation Protocol.* Request for Comments 2205, Network Working Group, September 1997. `http://www.ietf.org/rfc/rfc2205.txt`

[12] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. *An architecture for differentiated services.* Request for Comments 2475, Network Working Group, December 1998. `http://www.ietf.org/rfc/rfc2475.txt`

[13] OMG Team. *The common object request broker: Architecture and specification.* Object Management Group. October 2000. `http://www.omg.org`.

The Role of Nato C3 Interoperability Testing Infrastructure to Establish the Polish Interoperability Architecture

M. Amanowicz, P. Gajewski, P. Łubkowski, K. Łysek
Military University of Technology
Kaliskiego 2, 00-908 Warszawa
Poland

SUMMARY

The need to improve interoperability at all levels of Consultation, Command and Control (C3) support, including areas from political consultation to tactical battlefield operations, remains the most important question for NATO. This is a main statement of NATO Policy for C3 Interoperability (NC3IP) that provides nations, NATO Military Authorities and other NATO bodies that will support their efforts to enhance C3 interoperability and achieve standardisation objectives according to the NATO Interoperability Framework (NIF).

In order to achieve and to maintain C3 interoperability the concept for NATO Interoperability Environment Testing Infrastructure (NIETI) was developed. The scope of NIETI is to support the life-cycle testing of NATO Interoperability Environment (NIE) elements and operational interoperability within NATO and with Allied systems, including Partnership for Peace (PfP) nations by collecting, analysing and managing interoperability data and making tests and demonstrations to provide specific interoperability information. According to this all nations are working out their interoperability requirements and making tests, which are able to confirm this requirements.

The Polish interoperability requirements derive from NC3IP and NIETI. In the paper authors describe the NIETI process that includes gathering, managing and analysing data. They also present its impact on the developing of the Polish Communications and Information Systems (CIS) interoperability testing infrastructure which was established in the Communications Systems Institute (CSI) of the Military University of Technology (MUT). In the paper authors briefly describe some experiments, which were performed in collaboration with NATO C3 Agency.

1. INTRODUCTION

The new NATO Policy for C3 Interoperability calls for a new and more pragmatic method for improving interoperability. This new approach is required because of the increasing importance of multi-national force deployments, new NATO roles, and the need to cope with rapid advances in communication and information system technologies.

In a recent publication the NATO C3 Committee has attempted a rigorous definition of the methods and the facilities used during the life cycle of CISs. But the new NATO members and especially PfP partners expect some support in solving of CIS interoperability. That problem is very complex and that needs a wide international cooperation.

The necessary requirement for NATO and NATO partners' CIS ability to cooperate effectively depends on the implementation of common standards. So, interoperability is considered as a higher level of CIS standardisation that includes the applications, procedures and interconnections. But as it turned out standards alone cannot solve any real life CIS interoperability problems. The work on the standards remains theoretical as long as someone implements them. This is typically done at choice and it is difficult to state what has been implemented precisely. It means that standards should be coupled with coordination implementation in all systems involved and must be accompanied by appropriate concepts of operation, procedures as well as by empirical verification of interoperability. It is necessary to be aware of the systems possibilities and the differences between them in order to maximise their interoperability capabilities for planning purposes. It is also important to decide which standards are needed or require changes and those, which should not be used at all while developing new systems and modifying the old ones.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

The NATO Policy for C3 Interoperability provides nations, NATO Military Authorities and other NATO bodies with the C3 elements of the NATO Interoperability Framework (NIF) that will support their efforts to enhance C3 interoperability and achieve standardisation objectives, within a coherent, manageable programme.

The concept for NIETI derives from the NATO Policy for C3 Interoperability [3]. The Nations recognised theirs requirements for systems' interoperability testing, also for standards and prototypes. They expressed the need for a testing infrastructure that would maximise the use of existing NATO and national testbedding facilities.

Doctrinal, structural as well as technical changes within the Polish Armed Forces (PAF) create good conditions for CIS development with compliance to NATO CIS and with strict respect of international, especially NATO, standards. The general concept of the Polish CIS Interoperability Environment Testing Infrastructure is based on the NATO Interoperability Framework Technical Infrastructure. What is more, a close collaboration between CSI and NC3A gave an opportunity to create national testbedding architecture for CIS interoperability evaluation with respect to the NC3A experiences.

2. METHODOLOGY OF CIS INTEROPERABILITY EVALUATION

Evaluation of CIS interoperability in each stage of its development requires appropriate software tools as well as testbedding facilities for examination of communications and information systems and their equipment [2]. It is also necessary to stress that CIS interoperability problem should be investigated over all of system life, as it is shown in Figure 1. Typical methods of CIS interoperability evaluation cover a wide range of techniques from system analysis, computer modeling and simulation, prototyping, testbedding demonstrations, acceptance testing to operational testing (lessons). Some specific facilities have to be used to implement these methods like software models, testbeds, test and integration facilities, reference systems and operational systems.

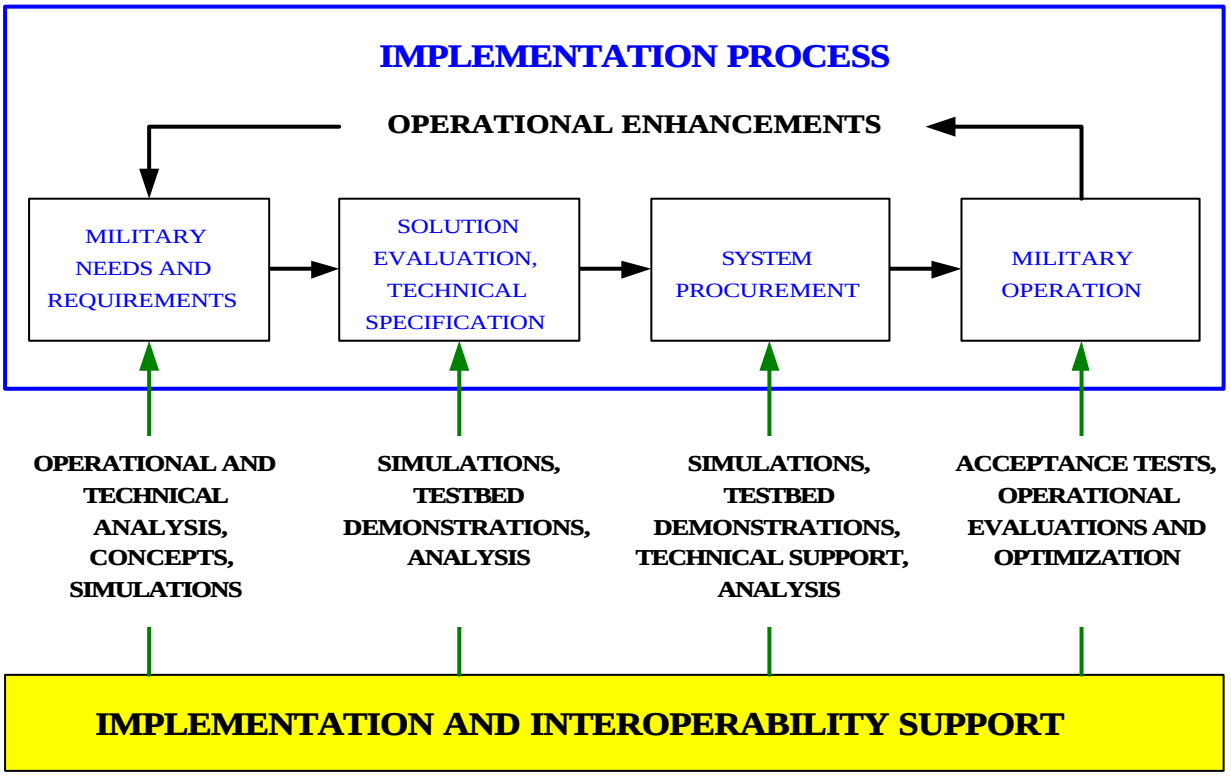


Figure 1. CIS interoperability evaluation during system life

General scheme of CIS interoperability evaluation scenario is presented in Figure 2. First of all we should recognise what does interoperability mean in particular situations, what is its scope and which is a way to achieve it. The structure and tasks of military forces, their commands and control

procedures, the services needed for them and the dimensions of common action, interconnection levels and needs for interoperability are the question for the first step of such analysis.

In general, CIS interoperability modeling is a multidimensional task. It requires very large multilevel databases about systems and their elements as well as the details of implemented protocols and procedural and technical standards. Calculation procedures should enable both effective manipulations of these data as well as obtaining reliable information about the achieved level of interoperability. It is desirable that methodology of CIS interoperability evaluation should enable identification of critical areas (elements and parameters) that are essential in fulfilling interoperability goals and in identifying corrective actions if the equipment is not available or if it cannot meet the requirements.

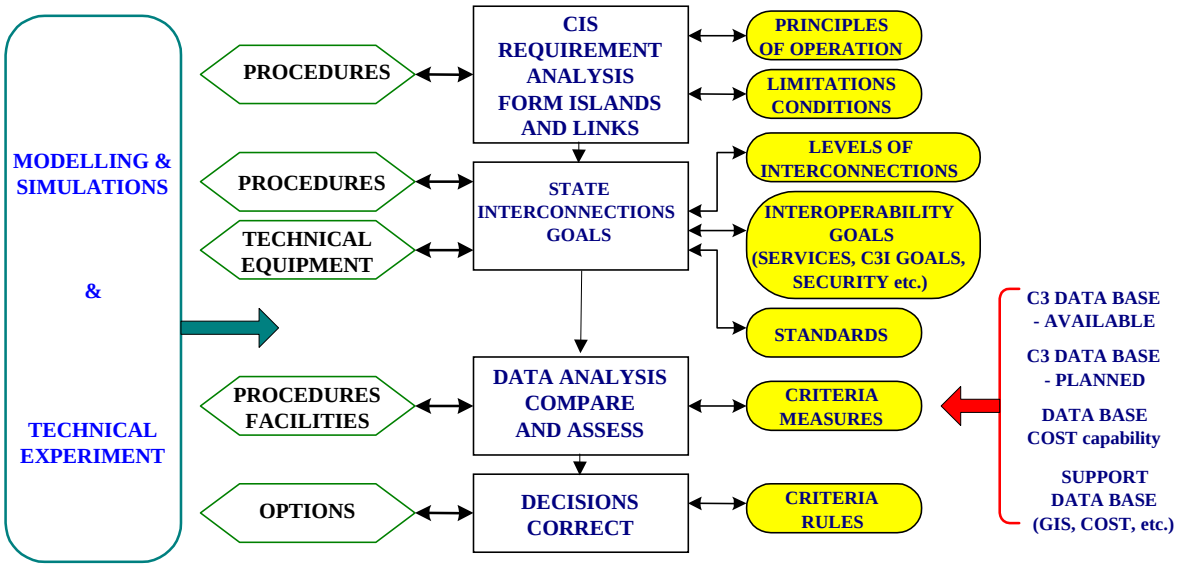


Figure 2. General scheme of CIS interoperability evaluation scenario

The general idea of military CIS interoperability modeling environment is presented in Figure 3. Methodology of interoperability modeling is in fact a multistage process that combines operational requirements, CIS data, standards, interfaces and modeling facilities in order to obtain the results necessary for making correct decisions. The marked areas presented in Figure 3 can be considered as “interfaces” of this methodology with its environment.

In many cases CIS interoperability identification can solve only limited range of interoperability questions. In particular, the fact that systems are described by the same technical specification (standard) does not mean that they can exchange services without any difficulties. In consequence, it is necessary to draw a special attention on experimental works that should be performed with application of different types of testbeds. It allows carrying out wide range of technical experiments both in real and simulated environment taking into account time and costs limits. Sensible fusion of theoretical methods with practical verification should give complete and certain solutions of interoperability problem [1].

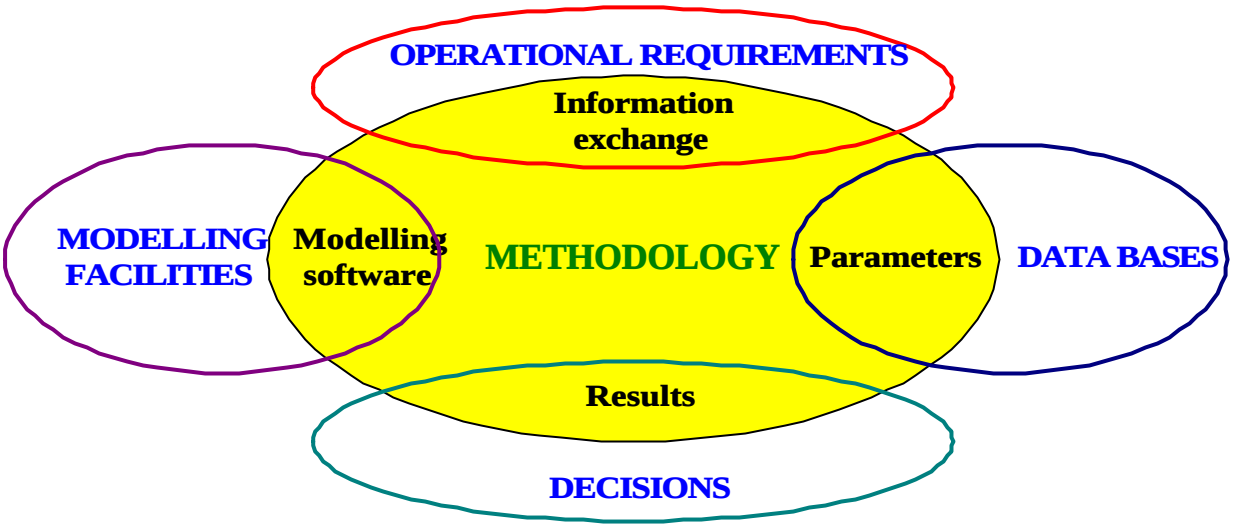


Figure 3. Interoperability modeling environment

3. NATO INTEROPERABILITY ENVIRONMENT TESTING INFRASTRUCTURE

The scope of NIETI is to support life-cycle testing of NIE elements and the operational interoperability within and between NATO to NATO, NATO to NATO nations, and NATO to Partnership for Peace (PfP) nations [3]. The NIETI comprises the following:

a) NATO and national interoperability
   - ♦ test facilities and test beds.
   - ♦ test tools.
   - ♦ laboratory facilities to host equipment under test.
   - ♦ wide area networks for distributed tests.
   - ♦ exercises, tests, demonstrations, trials and live operations, that are focused to provide specific items of interoperability information.

b) Commercial test facilities or scientific institutes, as appropriate.

The aim of NIETI is to provide an integrated multi-national testing capability making maximum use of existing NATO and national testing facilities and field activities, with a management structure to employ the capability.

The objectives of NIETI refer to:

a) promote the effective use of the NATO and national test facilities and opportunities for testing of NIE elements and operational interoperability,
b) improve quality of specification compliance of delivered products,
c) be the focal point for interoperability for NATO or NATO led operations, identifying where interoperability is achievable, may be improved or tolerated at a reduced level, and where significant shortfalls exists,
d) provide support to NATO and the nations to promote interoperability across nations and systems,
e) provide guidance on, and awareness of, interoperability capabilities for system users and procurers,
f) provide advice on priorities of testing needs,
g) co-ordinate the use of interoperability testing capabilities to satisfy the Interoperability Requirements identified in the Rolling Interoperability Programme (RIP).

Because of the scope and volume of information to be collected, the Interoperability Sub-Committee (ISC) - SC/2 authorised the formation of a NIETI Project Team, under the authority of the NIETWG (SC/2-WG/3), which is responsible for collecting information and realising tasks that have been agreed.

4. NIETI PROCESSES

Planning for NATO C3 interoperability is fundamental to the entire NATO C3 Planning Process from the initial identification of C3 requirements through the testing, validation and fielding of NATO C3 systems. Interoperability planning is an integral part of NATO C3 Planning. Interoperability Requirements (IORs) may be identified in the Bi-SC[1] C2 Plan, through NATO Standardisation Objectives or via Lessons Learnt from operations or exercises.

Where IORs cannot be met by use of standards or products identified in NIE, then an Interoperability Deficiency (IOD) exists which is also documented in RIP. IODs may also arise because NIE standards and products are not implemented in relevant NATO or national systems, or because the required interoperability has not been proven by test. Rectification of IOD is conducted via Interoperability Task (IOT), which may form part of NATO Capability Package or National Equipment Programme.

Completion of IT will normally require testing and/or validation of standards, products or fielded systems. This activity will be identified as Interoperability Testing Requirements (IOTR) and conducted through NIETI. IOTRs may also arise directly as a result of operations and exercises, field tests and demonstrations and in support of NATO and national developmental activity. IORs, IODs and IOTs are documented and managed via RIP. The IOTRs are documented in RIP and managed through NIETI management structure.

The NIETI process (Fig.4) takes input data from a number of key sources [3]. These are then stored and analysed before derived information is used to provide products and services (NIETI Outputs) to NATO and nations. NIETI output can then be fed back into the interoperability test process and impact the next iteration of NIETI inputs. NIETI management structure underpins the process.

Test system information will cover complementary requirements of what systems exist, how they may be secured and what they are capable of doing. Information will also be gathered on test events such as exercises, demonstration, trials and tests that can be used for interoperability testing. This will also be a major area of effort for NIETI team. NIETI seeks to identify and keep current knowledge of the following:

a) NATO and national test facilities that performs testing of standards or interoperability evaluations between C3 systems. These facilities must allow for the testing of other NATO nations' and potentially PfP nations' equipment.

b) NATO and national exercises, tests, demonstrations, and trials in which NATO and national C3 systems testing may be involved. An eclectic range of events is desirable: national single-service events will be gathered and made available to authorised NIETI users in the same way as those acquired for NATO combined and joint events.

Information will be obtained on interoperability requirements identified within the most likely operational scenarios. This will enable focus to be gained according to greatest interoperability need. Although this type of information will be sourced from NATO Strategic Commands (SHAPE, SACLANT), much will also be gathered directly from NATO nations' appropriate military commands. There are four distinct information source categories:

a) lessons learned from live operations
b) risk scenario (contingency) planning
c) wargame modelling
d) new IOTRs arising from interoperability testing of C3 systems in procurement or in service.

As indicated IOTRs arising from a variety of sources will be documented in RIP and managed through NIETI management structure. NIETI team will be responsible for assessing presented IOTRs and making recommendations as to priority, test solution, location and method. IOTRs will be tracked and supported to completion by NIETI team.

One of NIETI prime objectives is to assist NATO standards and products testing. The NIE C3 Testing Concept document states that, high quality validated standards and products are prerequisite for interoperability. Testing at this level will help remove ambiguities in standards and/or make them easier to implement in a practical manner. This will ease the acceptance and in-service operational testing tasks.

---

[1] The coordinated position of the two Strategic Commands: Allied Command Europe (ACE) and Allied Command Atlantic (ACLANT).

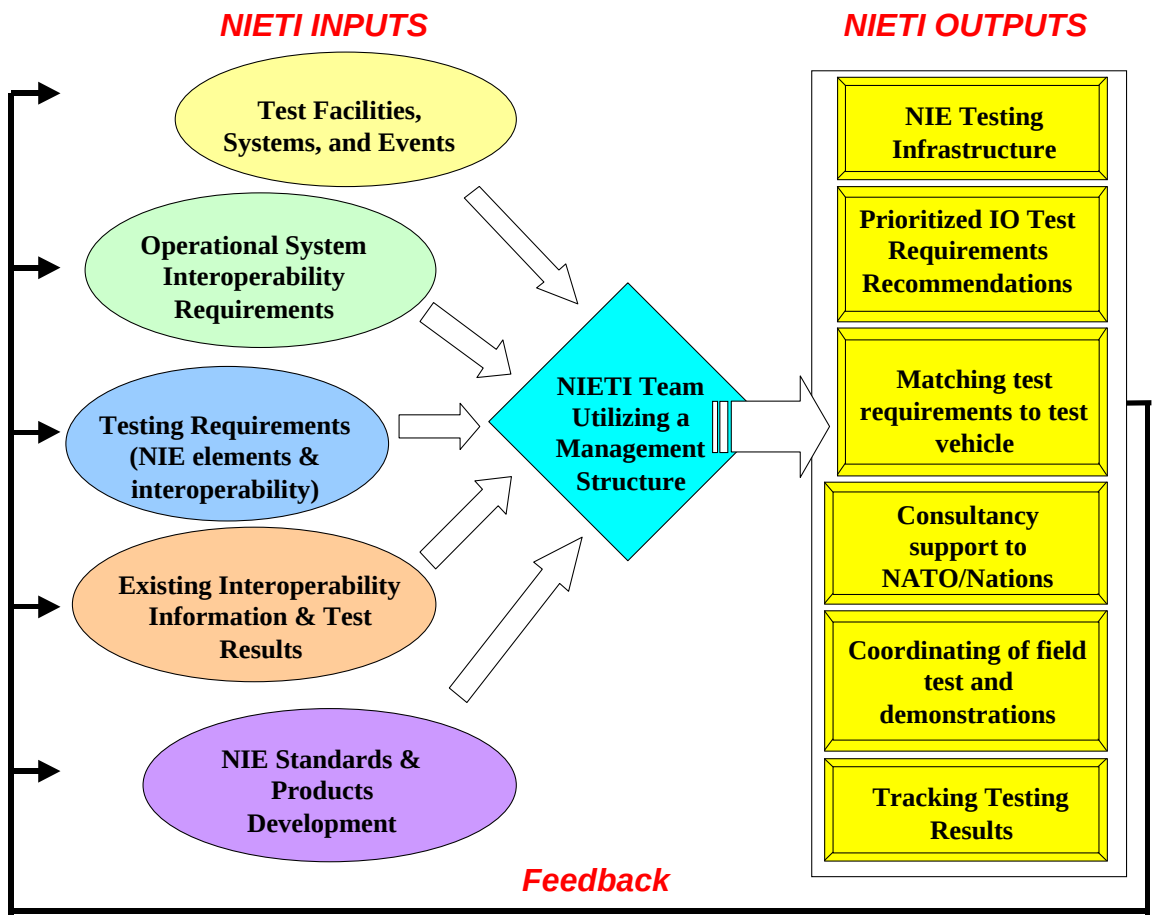


Figure 4. NIETI Input – Output Process

NIETI must gather sufficient information on NATO standards and products to know which standards require testing. Information relating to testing the NIE standards and products will be obtained primarily from relevant NATO Sub-Committee Working Groups. However, as the policy matures and is implemented for systems procurement, much of this information will be sourced directly from knowledge of NATO and national IOTRs.

Data gathering will be realised by using templates – which will be used to gather essential NIETI details. The following categories of templates have been developed:

a) Testing Facilities
b) Test Events
c) Operational Interoperability Requirements.

Information will be stored in electronic format in commercially available requirements management software application. This will enable tight configuration management of templates - and as importantly, the information sourced through them.

NATO and national interoperability databases that are made available NIETI will be used to help determine recommendations for testing priorities, locations and times for testing. Currently there are three NATO tools that appear to contain useful information to NIETI. This list will be expanded as other tools are identified.

a) Interim Joint Operational Tactical Interoperability Database (JOTID)
b) DAKIS (German acronym for database for CIS)
c) Rolling Interoperability Programme.

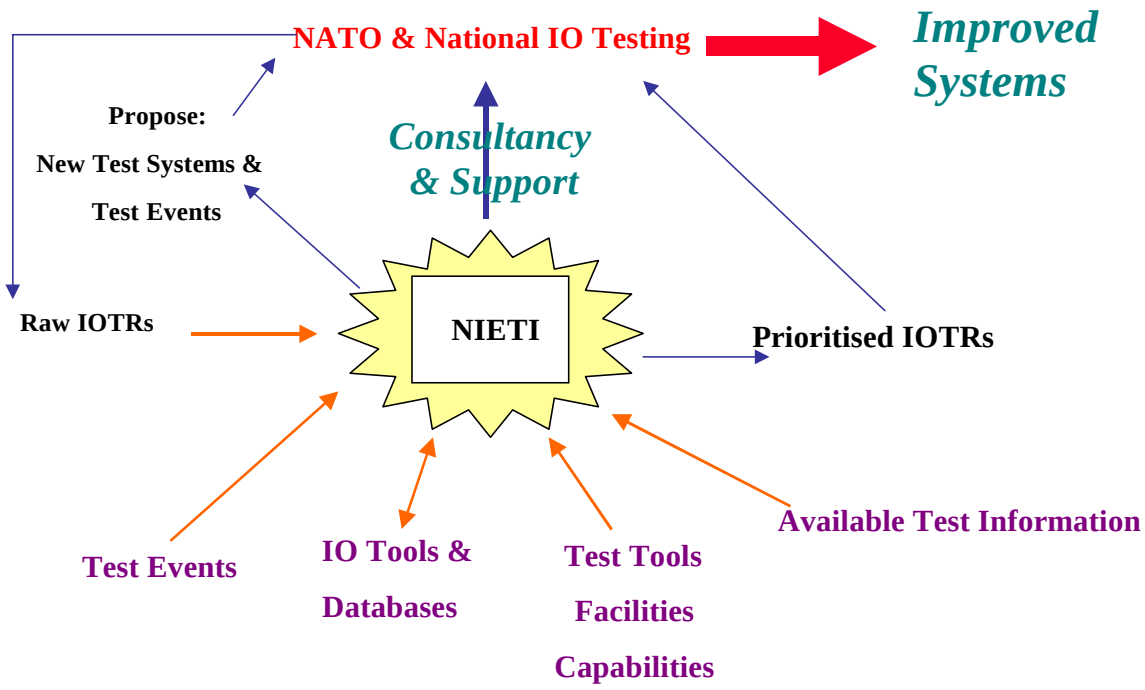


Figure 5. NIETI Team Services

Based upon the data gathered and the analysis performed by NIETI team, several major outputs and services will be generated and maintained by NIETI team (Fig.5). These include:

a) database of testing facilities, demos, exercises, tests, and trials that will comprise the infrastructure and can be used for testing. The database will include information such as location, Point of Contacts (POCs), available time;

b) recommended list of prioritised interoperability and standards testing will be developed based upon a prioritisation scheme developed by NIETI team and forwarded for approval and implementation;

c) limited storage of testing plans and testing results. This will be the base when no adequate distribution capability exists at the testing facility to make testing results available to NATO and the nations;

d) consultancy to NATO and the nations on interoperability issues, to include advice on the types of testing to conduct, the appropriate testing facilities to use, and planning and conducting testing;

e) co-ordinating field tests and demonstration between NATO and national tests to ensure the maximum amount of synergy and benefit;

f) provide recommendations on the appropriate match between testing facilities, system, and/or events to the identified IOTRs.

5. THE POLISH INTEROPERABILITY ENVIRONMENT TESTING INFRASTRUCTURE

In order to achieve the required degrees of interoperability, the elements of the NATO Interoperability Environment should be subject to a rigorous process of test, verification and validation throughout their design and implementation.

Testing of NIE standards and products will require the availability and selective use of NATO and national test facilities within NIE Testing Infrastructure. NIETI supports the achievement of interoperability by validating standards adopted or developed by NATO and by verifying products. This concept provides for co-ordinated NATO activity with national facilities and focuses on interoperability within NATO organic systems and between them and national systems.

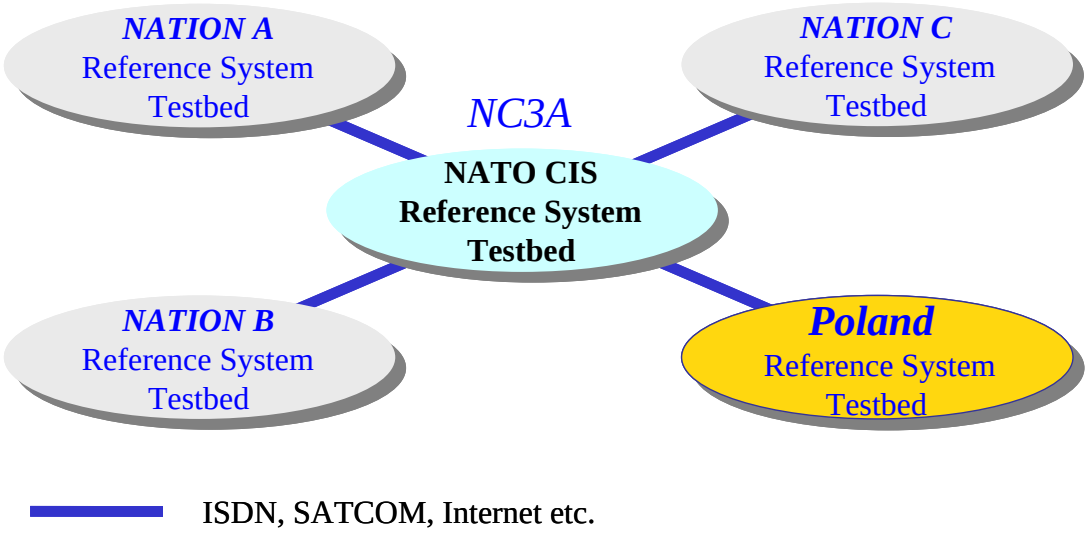


Figure 6. NIETI Concept

The NATO organic capability (Fig.6) is represented by the NATO C3 Reference System Testbed located in NC3A in the Hague (The Netherlands). National systems (or elements) and the Reference System should be interconnected by communications links with the aim to form a distributed testbed and to examine interoperability issues using dial-up ISDN lines, SATCOM, Internet or other communications media.

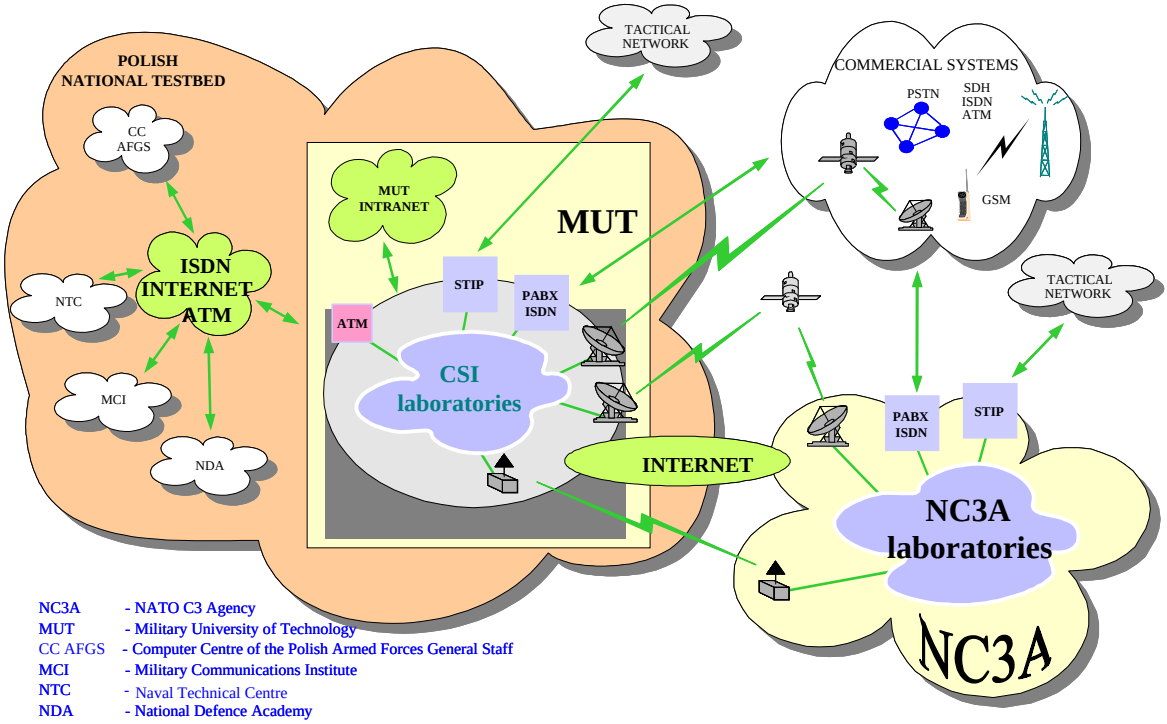


Figure 7. The architecture of PIETI Project

The Polish Interoperability Environment Testing Infrastructure (PIETI), as a part of NIETI, should provide testing for C3 standards and products and for operational interoperability, with the aim to improve C3 systems throughout their all life cycle. PIETI (Fig.7) will enhance interoperability by assisting the evaluation of rapidly evolving systems, and by evaluating standards, through enabling more effective use of testbeds, demonstrations, exercises, tests, etc. It consists of five military centres working in CIS area: Computer Centre of the Polish General Staff, Naval Technical Centre, Military

Communications Institute, Military University of Technology and National Defence Academy. ATM technology will be used to create the backbone transmission network of the system.

This distributed testbed will be connected to security military networks and to many commercial systems by various transmission media like wire, fibre optics, HF radio, satellites, etc. Several communications links established between MUT in Warsaw and NC3A in The Hague give a unique opportunity to provide common experiments for the NATO and the Polish CIS interoperability.

PIETI allows carrying out wide range of technical experiments both in real and simulated environment taking into account time and cost limits. A special attention will be put on performing tests of C3 systems and equipment planned for deployment within the Polish Armed Forces [2]. PIETI allows delivering to the commanders a great amount of different types of information necessary for effective performing command and control functions.

For effective use of resources, PIETI Management Structure must be established. It will be used to manage the PIETI and assist in determining priorities of tests. The PIETI Management Structure will also be responsible for ensuring that test plans, analysis, and results related to PIETI are documented and, if appropriate, distributed within NATO and the nations.

6. MUT EXPERIENCES IN CIS INTEROPERABILITY TESTING

The Military University of Technology, which is the main Polish military educational and research centre, plays leading role in developing PIETI. The process of establishing the University part of PIETI is split into several phases. During initial phase internal communications infrastructure (Intranet) was established which connects several LANs and research laboratories. The main elements of this system were installed in Communication Systems Institute (CSI). CSI is equipped with communications installations like PABX, ISDN, ISLAN and with software packages appropriate for CIS interoperability modelling and testing (Fig.8).

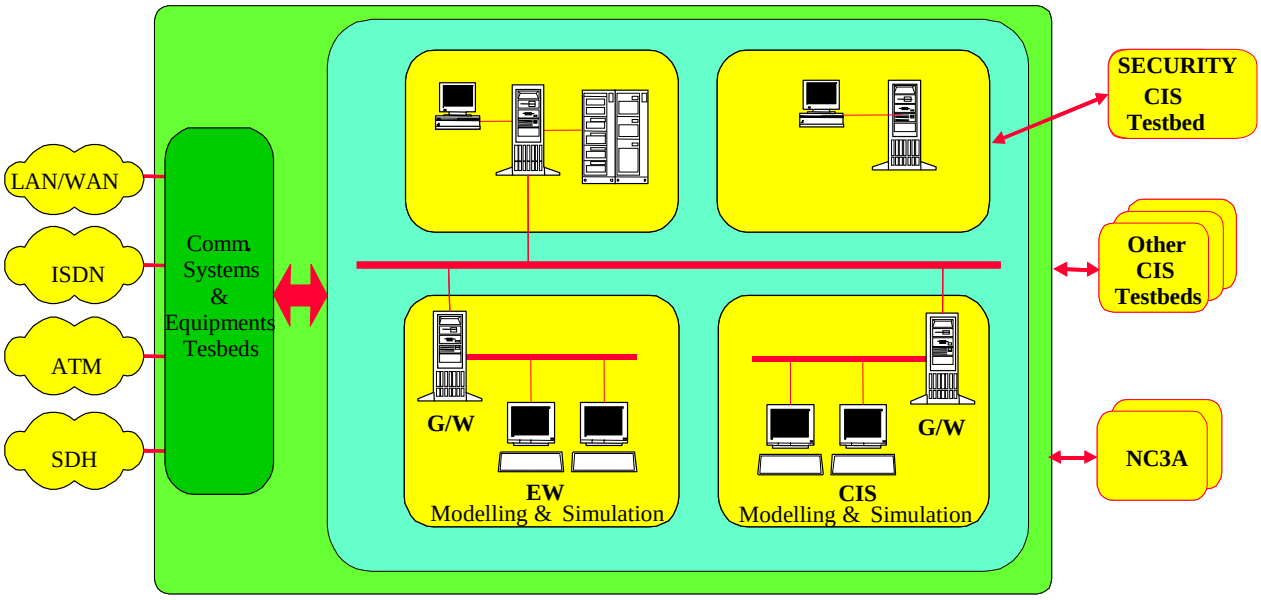


Figure 8. MUT CSI laboratory equipment

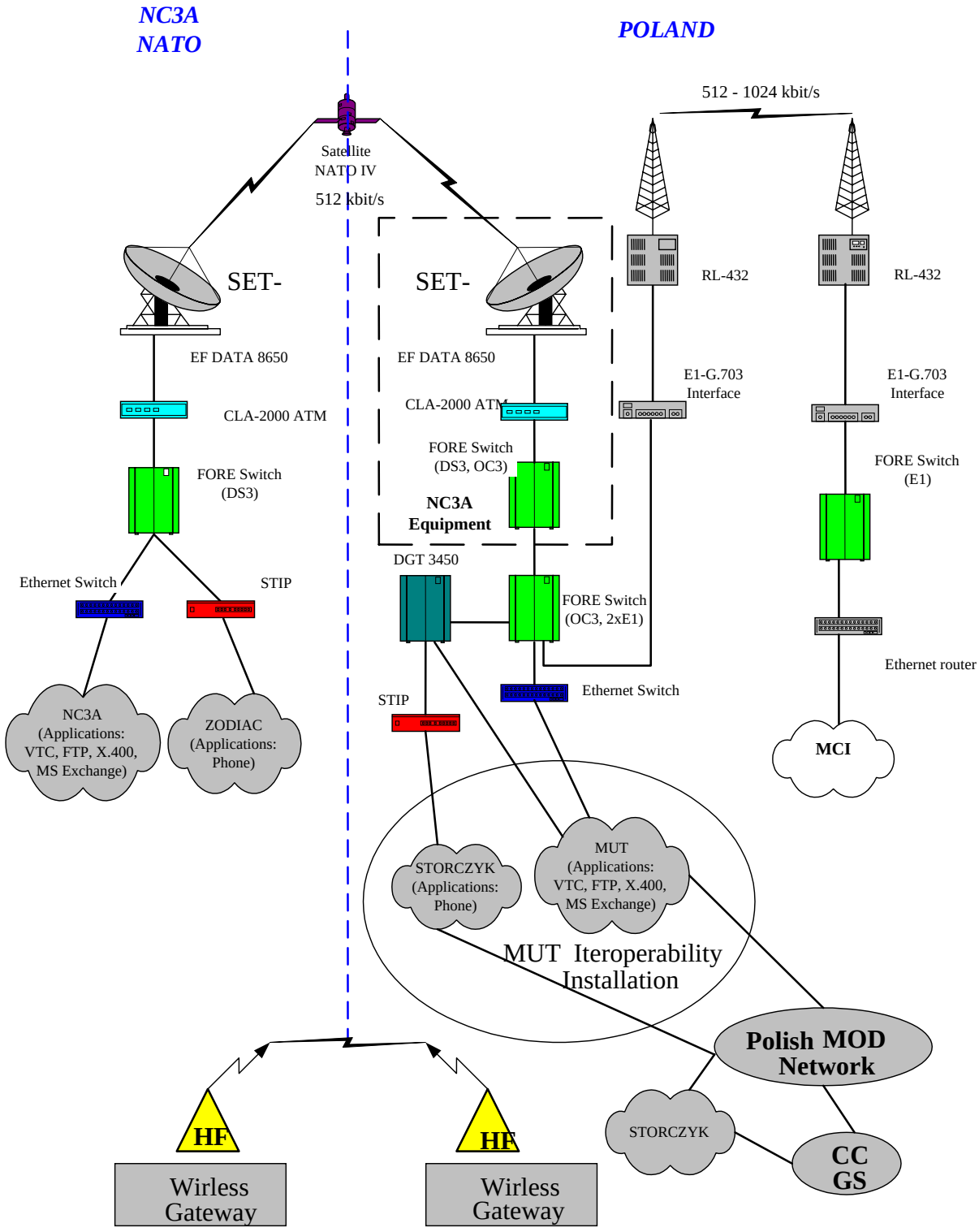


Figure 9. Configuration of MUT CSI demonstration

During this phase the possibility of using of different transmission media (wire, fibre optics, HF radio, VHF radio, etc.) to interconnect military networks like for instance non-secure and secure networks, strategic and tactical networks as well as the ability of these networks interworking with several commercial systems were examined. Besides, some external communications links were also established. In particular CSI Communications Laboratory was connected to NC3A testbedding infrastructure.

Second phase, launched in 1999, includes common experiments on NATO and the Polish CIS interworking. Trials performed in 1999 were focused on providing videoconference and LAN to LAN

connections over ISDN Basic Rate Interface (2x64kb/s) and HF radio. Special attention was put to the ability of application sharing and file transferring during videoconference as well as to SMTP and HTTP services. Client-server architecture was used with server located in The Hague and clients located both in NC3A and MUT. It was possible to load WWW pages, send and receive e-mails using Netscape 4.0, MS Internet Explorer 4.0 and Microsoft Outlook.

The year 2000 demonstration was jointly organized by NC3A (supported by Marconi Fore and Dutch TNO), Computer Center of the Polish Armed Forces General Staff, Military Communication Institute and Military University of Technology as a coordinator. The basic elements of Wide Area Network established for demonstration purposes are presented in Figure 9. The network was based on four ATM nodes and managed by host located in NC3A (The Hague). Nodes in MUT and MCI were connected by 1024 kb/s radio link (RL 432). For NC3A-MUT connection, satellite terminals SET4 (in The Hague) and SET10 (in Warsaw) were used. The data was exchanged at 512 kb/s transmission rate. Good propagation conditions (no clouds, no rain, and no interference) allowed achieving high quality and continuity of transmission. Only screening the SET10 antenna beam with trees caused short disruptions during NATO IVA satellite tracing.

This telecommunication infrastructure was used for testing interoperability of selected C2 applications based on NC3A Virtual Command Center (VCC) and the Polish Common Operational Graphics. In addition several trials were performed with the aim to evaluate the following standards and/or products on C2 systems interoperability:

- STANAG 4486 (SHF SATCOM link)
- STANAG 5066 (HF link)
- RS 449, SONET OC3/SDH STM-1, DS3, E1/G.703, Ethernet 802.3
- IP over ATM (ATM Forum)
- FTP, TCP (Internet Engineering Task Force standards)
- X.400, MS Exchange, NetMeeting
- GIF, JPEG, MDPv2.

The Polish elements of the system as well as the results of trials were presented to the members of the NATO Interoperability Environment Testing Working Group (NIETWG) during the meeting in Warsaw.

Next phases of PIETI development will focus on extending the scope of interoperability testing and on connecting MUT testbedding infrastructure to the others centres.

CONCLUSION

As it was shown in the paper adoption of the NIETI concept will give a great benefit to NATO and the nations through more efficient and cost-effective use of NATO and national resources. This also allows supporting standardisation effort in effective way. The already performed experiments confirmed the real progress in integration process and the ability to achieve real coalition interoperability, which could support C3 functions.

The Military University of Technology testbedding environment creates additional benefits, which could extend the number of its potential applications. This refers especially to potential role of such infrastructure in education and training of C4I personnel as well as for demonstration of emerging communications and information technologies.

REFERENCES

[1] M.Amanowicz, P.Gajewski, M.Barszczewski, *Procedural and technical aspects aiming at achieving the Polish Armed Forces and NATO CIS interoperability*. 2nd NATO Symposium with Partners “Cooperation in CIS – Narrowing the Gap”, Brussels, October, 1997

[2] M.Amanowicz, P.Gajewski, *Polish experiences in achieving interoperability goals*. AFCEA Technet Europe Symposium, Brno, October 1999

[3] *NIETI CONOPS*, AC/322(SC2-WG/3)(WP/2) version 1.3, July 2000

This page has been deliberately left blank

Page intentionnellement blanche

Key Concepts for Information Superiority

Dr. David S. Alberts
Special Assistant to the ASD(C3I)
Director, Research & Strategic Planning
Office of the Assistant Secretary of Defense (C3I)
United States Department of Defense
Room 3E172
6000 Defense
The Pentagon
Washington, DC 20301-6000, USA

SUMMARY

Information Superiority is on the critical path of our journey to the future. This paper explains why this is so and identifies a number of specific actions that are necessary to facilitate our journey and to expedite progress toward our goal.

BACKGROUND

The purpose of this paper is to acquaint you with some of the key challenges faced by the US Department of Defense in realizing our goal of transforming the force for the 21st Century. The paper covers four interrelated topics: information superiority, network centric warfare, interoperability, and the challenges ahead. Its arguments are also contained in expanded form in a new book, still in production, entitled *Understanding Information Age Warfare*.

I want to thank the members of the Information Superiority Metrics Working Group, an informal collaboration forum that includes senior people from the US Government, our Federally Funded Research and Development Centers, academia, and industry. This group, which meets at least once per month, has provided valuable comments and suggestions on the Information Superiority concepts and metrics discussed in this paper.

A Revolution In The Making

Much has been made, both in the US and among its allies and coalition partners, of the Revolution in Military Affairs (RMA). The applicability of new, Information Age technologies to warfare has been widely recognized. Improved sensors; enhanced capability to process, fuse, and share information; precision munitions with long stand-off ranges; network-centric forces; and effects-based operations employing offensive information weapons are all discussed as factors helping to transform the battlespace.

However, these same developments are also associated with new vulnerabilities and sources of uncertainty. Perhaps the greatest uncertainty is whether we will be able to master these technologies, understand their implications, and harness them in the near future. Alternatively, we may be driven to them as others learn to apply them in order to frustrate our more traditional approaches and weapons systems, and to exploit the opportunities (the "seams") we create for adversaries by our own partial and unsynchronized efforts to embrace these new technologies.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

While the future cannot be known, it is clear that the US Department of Defense has begun a transformation process. Doctrinal change began with the publication of *Joint Vision 2010* and related efforts to implement those ideas, and continues today as *Joint Vision 2020* is promulgated and efforts are made to translate it into action. Each of the US Services has also undertaken efforts to fulfill its Title 10 responsibilities to train and equip fighting forces in keeping with the RMA. The Army's Advanced Warfighting Experiments (AWE), the creation of its digitized brigade, and the concept of a lighter and more readily deployed forces are part of this effort. The US Marine Corps is conducting its Hunter series of exercises (Hunter Warrior, Urban Hunter, etc.) and has organized its own laboratory on war fighting issues in order to both develop new concepts and exploit cost effective technologies to meet modern mission needs. The US Navy's series of Fleet Battle Experiments and its exploration of concepts such as "Ring of Fire" to improve sensor-to-shooter relationships reflect its commitment to the RMA. The USAF Expeditionary Force Experiment (EFX), converted after one year to the Joint Expeditionary Force Experiment (JEFX) series has allowed it to explore split-based operations and new organizational forms enabled by changes in information technologies.

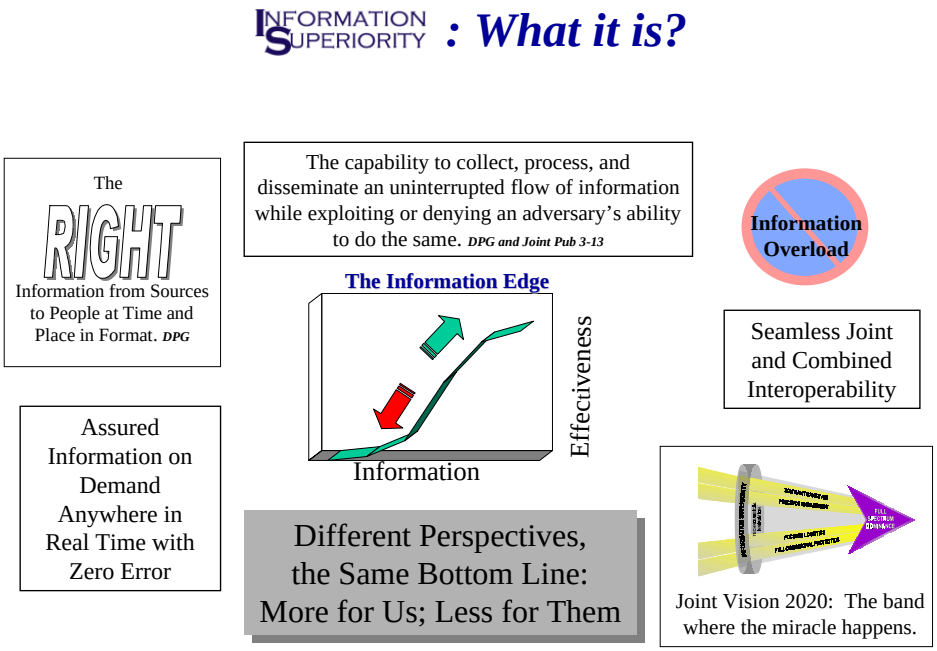
With encouragement from Congress, Joint Forces Command (JFCOM) has undertaken the mission of research, development, and training of Joint Task Forces (JTF) and the processes to make them effective across a wide range of mission types and situations, including coalition operations. The J-9 of JFCOM has been explicitly charged with developing a program of Joint Experimentation. These now include major multi-service events such a Millennium Challenge 02, focused command post experiments such as Unified Vision 01 (conducted in May of 2001 to try out new organizational concepts and ways of managing information within a JTF), and Limited Objective Experiments (LOE) on selected topics. These LOEs, which provide an opportunity for greater control and more in-depth analysis, are likely to be increasingly important as new concepts emerge and begin to mature. This year, for example, J-9 JFCOM has scheduled or completed LOEs on Open Source Information, Presentation and Collaboration Technologies, and Coalition Operations.

While the RMA and the research and experimental efforts to explore it have dealt with a wide variety of topics, three common themes have emerged: information superiority, network-centric concepts, and the crucial role of coalition operations. Each of these topics is covered in some detail below.

Information Superiority: What Is It?

As is illustrated in **Figure 1**, everyone writing on the topic of Information Superiority has felt free to offer their own definitions. While some are authoritative, others are better expressions for the purposes of their authors. My personal favorite happens to be the one associated with *Joint Vision 2020*, but not actually found there: "the band where the miracle happens." This choice reflects the fact that many of us are still trying to work out exactly what information superiority really is and how we can go about creating and exploiting it. While I will offer a useful definition of my own, I want to stress that we must all remain open to definitions and interpretations that will enable us to better understand and exploit the RMA.

Figure 1



Information Superiority has, as a concept, been emerging for a number of years. It currently means slightly differently things to different people who look at it through the lens of their position in the organization and their job responsibilities. It is important that we all understand what Information Superiority is, how we get it, and how we use it to gain our National Security objectives. Information Superiority is the enabler of the RMA and the foundation upon which Joint Vision 2010 (JV 2010) is to be built.

The writings of Carl von Clausewitz **(Figure 2)** are famous for his articulation of the fog and friction of war. As a result of this enduring characteristic of war, military organizations have for centuries been designed to accommodate the lack of available information; that is, how to deal with the fog of war. Fog is all about uncertainty: uncertainty about where everyone is, what their capabilities are, and the nature of their intentions. Until recently a commander could not even have a timely and accurate picture of his own forces let alone be comfortable about where the enemy was and what they were up to.

Figure 2

Friction is about the glitches that occur in carrying out plans to synchronize forces or even to accomplish the most simple tasks. Some of this friction can be attributed to fog, some to poor communications, and some to a lack of shared knowledge.

To compound the problem, decision making in war carries with it an extremely high cost of error. Therefore, it is not surprising that military concepts of operation, organizations, doctrine, and training have always been preoccupied with reducing the effects and risks associated with fog and friction.

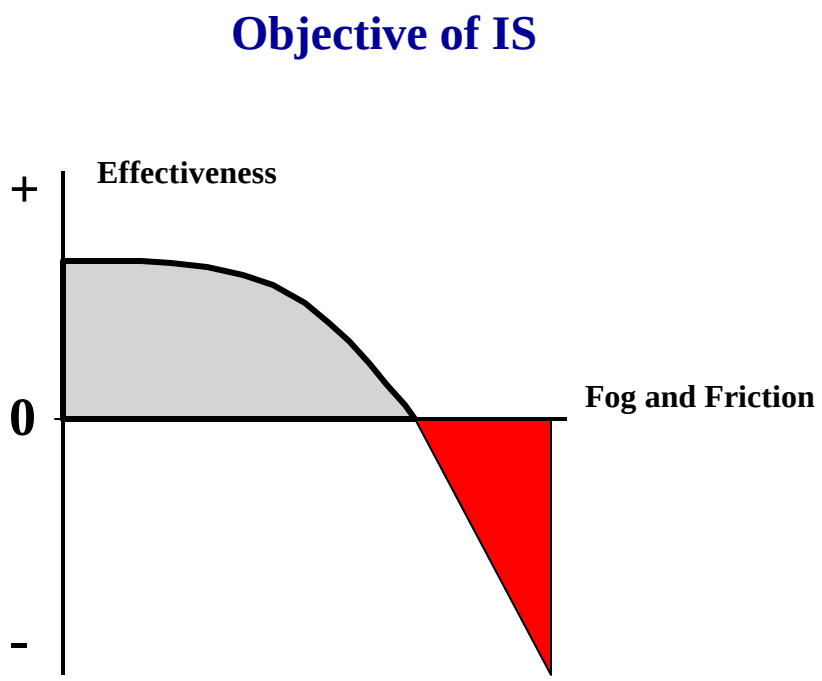
Taken together, these enduring characteristics of war have shaped our traditions, our military culture, and our thinking. Departure from these norms will be difficult and will require a high degree of proof that the new way is not only better, but is also robust.

Recent advances in technology offer us an opportunity to reduce fog and friction. However, despite all of the advances that have and will likely be made, significant residual fog and friction will persist. The nature of this residual uncertainly is, as yet, unclear and its implications are not fully understood. Nevertheless, there is an historic opportunity for us to reconsider how best to deal with the fog and friction that will persist, and this is likely to have profound implications for military operations and organizations.

Objective of Information Superiority

Figure 3 illustrates the relationship between the amount of fog and friction and the level of synchronization that is likely to be achieved in military operations, which is directly related to effectiveness. For almost all of recorded history, we have operated in various parts of the shaded (black) area depicted, trying to avoid the worst parts of this space (the lower right).

Figure 3



The Information Age gives us an opportunity to move into the slashed area. We must recognize that there is a limit to our ability to reduce the fog and friction of war and that in many cases it may not even be possible. We have witnessed the complexity of 21st century missions in Somalia and Bosnia as well as our limitations in being able to collect, process, and distribute needed information for air attacks during *Operation Allied Force*.

Hence, our goal in examining the role of information in warfare is to better understand not only how to create and leverage Information Superiority, but also how to better deal with the residual uncertainty that will surely exist.

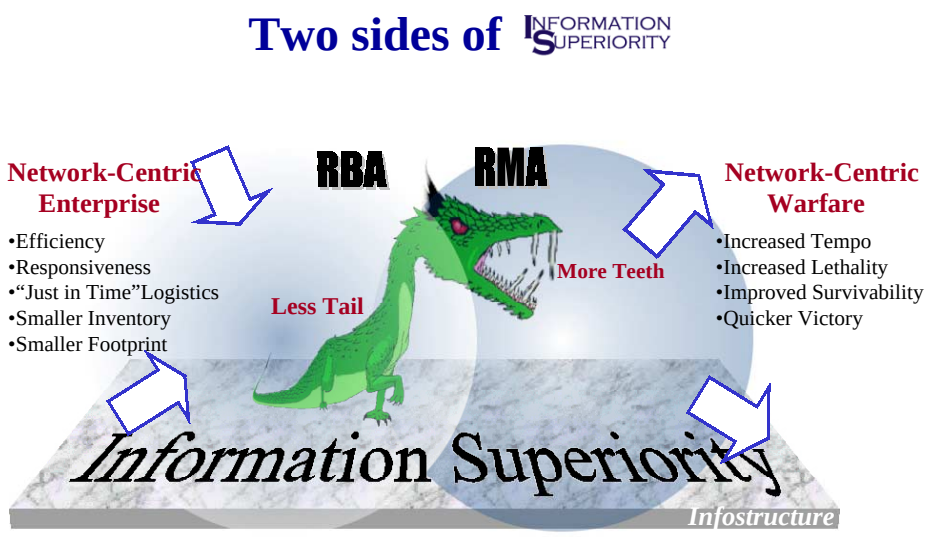
Visionaries who have proclaimed that we will have total awareness or that we will eliminate the fog of war are indeed false prophets—and dangerous ones at that. This is not only for the obvious reason that they could lead some down an unproductive road, but perhaps more importantly, they are poisoning the well for ideas to capitalize on emerging information and networking capabilities that will provide real opportunities to improve our military effectiveness.

The Two Sides of Information Superiority

The concept of Information Superiority is equally applicable to both "sides" of the DoD—the business side of the house and the warfighting side of the house.

As illustrated in **Figure 4**, the net result will be less tail and more teeth.

Figure 4



A Relative State that is Achieved when a Competitive Advantage is Derived from the Ability to Exploit an Information Advantage

The Revolution in Business Affairs (RBA) thus complements, contributes to, and supports the Revolution in Military Affairs (RMA). The RBA will make DoD a network-centric enterprise increasing efficiencies and responsiveness. "Just-in-time" concepts will be adopted when they make sense to drive down inventories and reduce footprints in theater. This will make deployments more rapid and allow us to engage more quickly.

The Revolution in Military Affairs (RMA) will build upon the lessons learned in the RBA adapting the notion of Information Superiority to the military domain. This will transform warfare into Network Centric Warfare, increasing the tempo of operations, the speed of command, and as a result, achieving greater lethality and survivability. The net result will be an opportunity for quicker and more decisive victories.

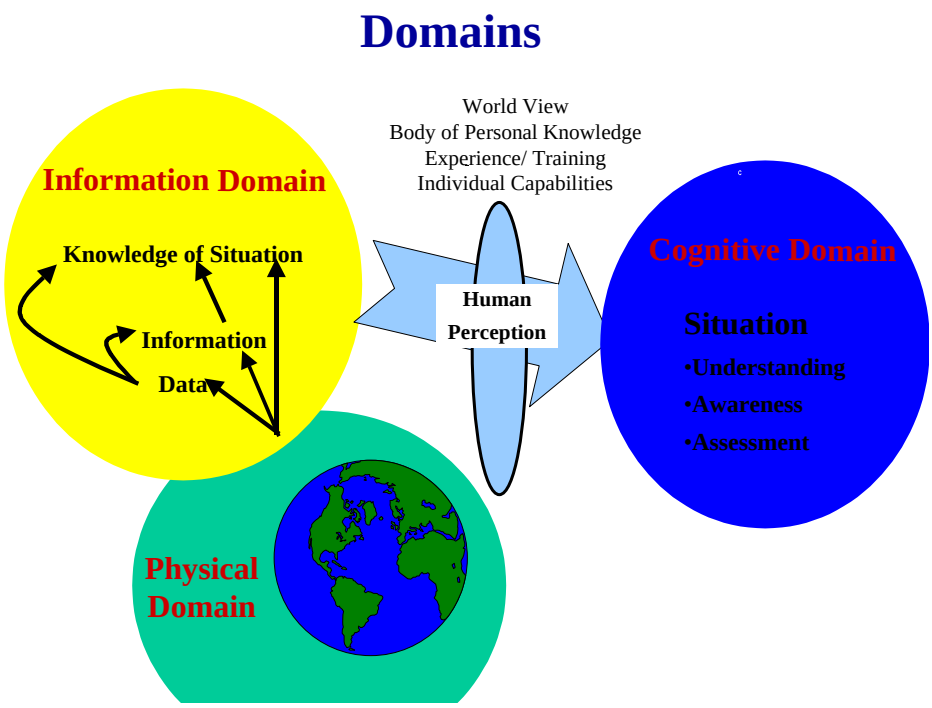
Both these revolutions are dependent upon the achievement of Information Superiority, which is a relative state (vis-à-vis an adversary) where a competitive advantage is derived from the ability to exploit a superior information position—that is, the competitive edge that comes from having better information and knowing what to do with it.

Having established this background, let me offer a working definition. *Information Superiority is a* ***state*** *achieved by establishing a* ***relative information advantage*** *from which a* ***competitive advantage*** *can be gained.* All of this will be explored in depth below. At this point, focus on the key terms. Information Superiority (IS) is a state, a condition that can be achieved, or lost, over time. That state is understood to be a relative information advantage—it is comparative as well as transitory. Finally, having an information advantage does not result in information superiority unless it can be converted into a meaningful competitive advantage. Hence, IS is not a goal in itself, but a means to an end. Its existence and importance depend on its contribution to that end and to the value placed in the end, which will normally be military mission accomplishment.

Domains

To understand how information affects our ability to perform military operations it is necessary to think about three domains—the physical domain, the information domain, and the cognitive domain, which are illustrated in **Figure 5**.

Figure 5



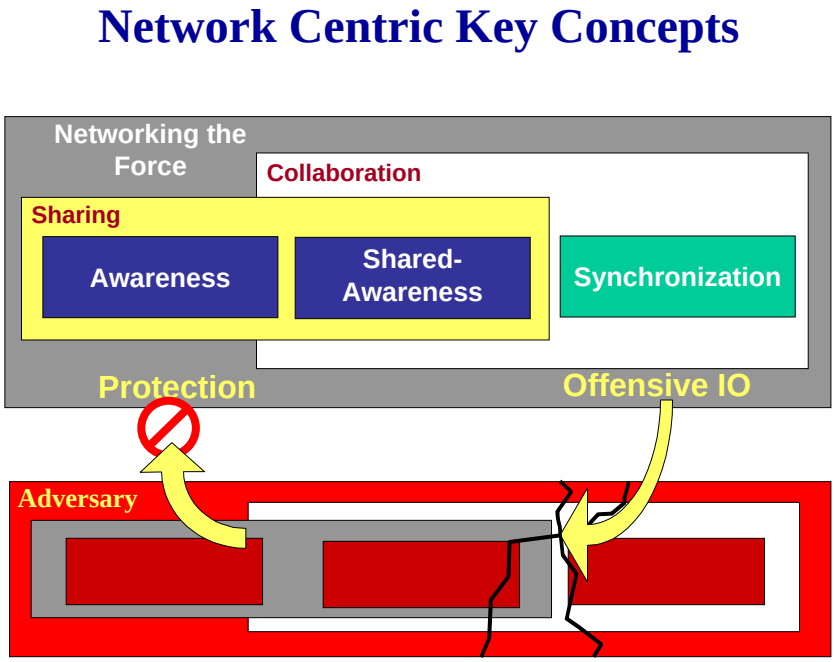
The physical domain is characterized as reality or ground truth in our analyses and models. The situations that the military seeks to influence and the actions it takes exist in this domain. The information domain is where information and information systems exist. This information may or may not truly reflect ground truth. For example, a sensor observes the real world and produces an output (data). With the exception of direct sensory observation, all of our information about the world comes through and is affected by the capabilities that comprise the information domain, and it is through the information domain that we communicate with others (telepathy would be an exception). The cognitive domain is what happens in our brains. This is the place in which perceptions, beliefs, awareness, and understanding reside. Decisions take place in this domain. Note that *all* of the contents of the cognitive domain pass through a filter or lens we have labeled human perception. This filter consists of the individual's worldview, the body of personal knowledge the person brings to the situation, their experience, training, and individual capabilities (intelligence, personal style, perceptual capabilities, etc.). Since these human perceptual lenses are unique to each individual, we know that individual cognition (understandings, etc.) are also unique. There is one reality, or physical domain. This is converted into selected data, information, and knowledge by the systems in the information domain. By training and shared experience we try to make the cognitive activities of military decision makers similar, but they nevertheless remain unique to each individual, with differences being more significant among individuals from different Services, generations, and countries than they are among individuals from the same unit or Service.

NETWORK CENTRIC KEY CONCEPTS

A network-centric model of warfighting is not simply an improvement or extension of a platform-centric model; it involves a new way of thinking about military operations—a new mental model of Key Concepts—

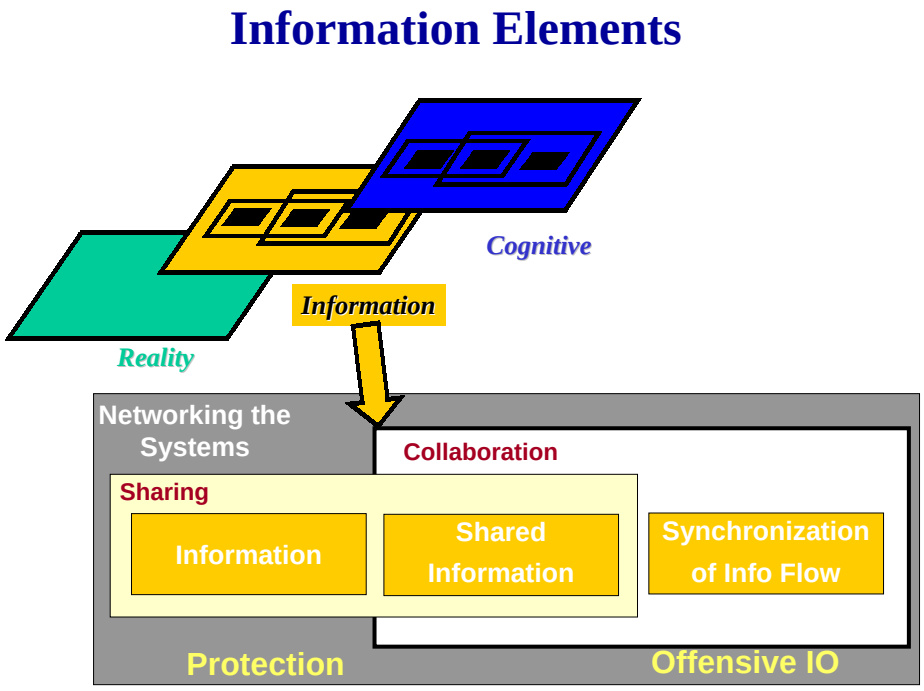
as depicted in **Figure 6**. This new mental model is focused upon sharing and collaboration to create increased awareness, shared awareness, and as a result, improved synchronization. This replaces the old linear and sequential model in which information is collected, processed, and provided to a decision maker, followed by the passing of the decision along for execution. The new mental model serves to integrate military operations and provides an opportunity to employ new, more responsive approaches to command and control.

Figure 6



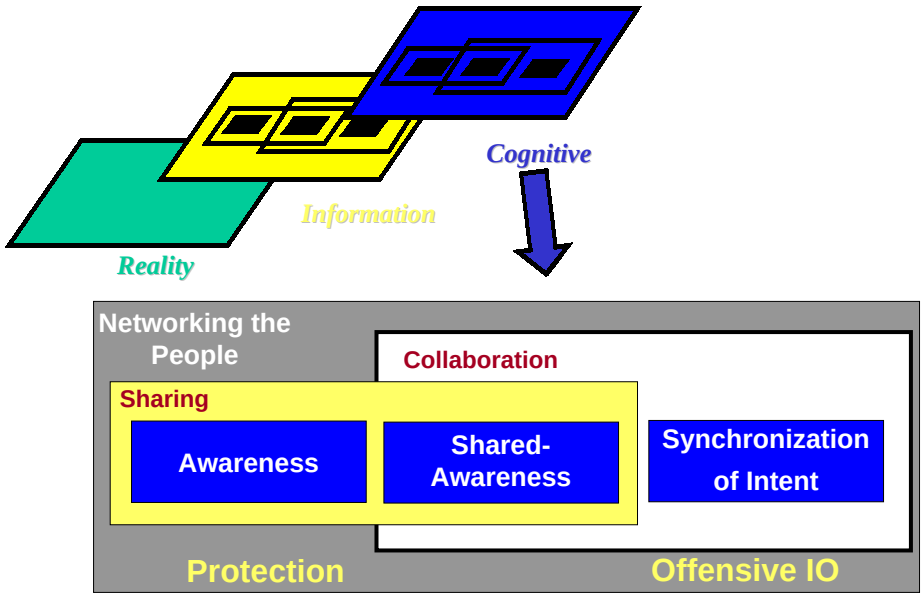
In the final analysis, we are trying to achieve synchronization in the physical domain (effects in the battlespace), and in order to achieve this we must first achieve effects in the cognitive domain. The new mental model actually exists in each of the domains—the information, the cognitive, and the physical. **Figures 7** and **8** depict the Information and Cognitive views. These views differ in regard to the nature of what is being shared, the nature of collaboration, and the object of synchronization.

Figure 7



To understand the new mental model one needs to start with the view from the information domain (**Figure 7**), with the sharing of information and with collaboration designed to help ensure quality information (e.g., identify and resolve conflicting information). The result is what we would call a common operational picture—that is a synchronized set of information across the battlespace.

Figure 8



The next step in understanding the new mental model is to move to the view from the cognitive domain (**Figure 8**). It is here that the distinction between information and awareness is made. From this perspective, it is awareness and shared awareness that are increased by sharing, and collaboration with decisions (across the battlespace) being the object of efforts to synchronize.

Information Superiority Value Chain

Thus, the new mental model is really a synthesis of what needs to occur in each of the domains. **Figure 9** depicts the relation between the results of sharing and collaboration integrated across the domains and our goal of achieving a competitive advantage. Working back from this desired result, a competitive advantage derives from achieving both decision superiority and the ability to execute.

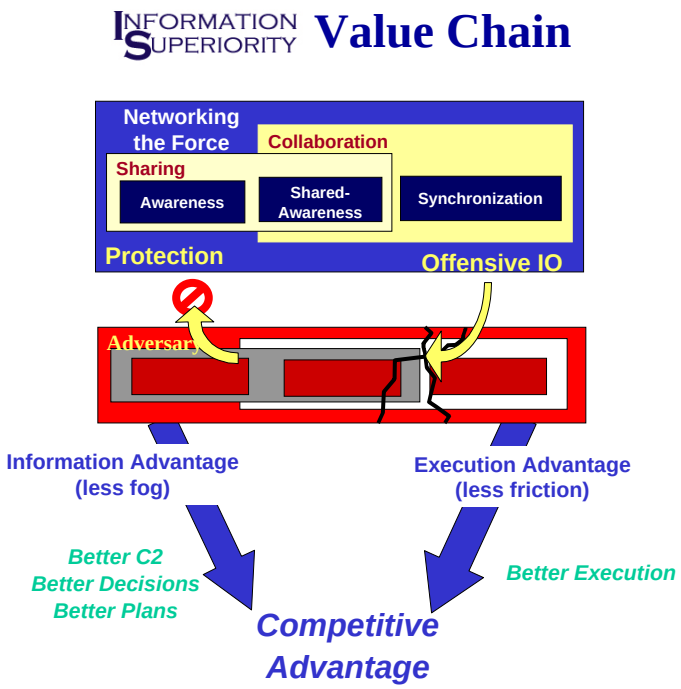
Decision superiority is enabled by 1) an information advantage which can be thought of as having less fog compared to an adversary, and 2) execution superiority enabled by less friction.

Battlespace Information, Awareness, Knowledge

As used in the vernacular, the term "information" may include data, information (data in context), or even knowledge, which organizes information into cause and effect and temporal statements about how some part of the world works. As used in this paper, the term *information* relates to all three, provided they are representations and not unperceived (physical reality that has not been sensed) or in the brains of individuals (the cognitive domain). Information thus may exist as sensed, as perceived through direct or indirect observation, or as shared by individuals through some means of communication. Note that having more information means just that—greater information rather than information superiority.

Battlespace information is related to the totality of what is known by fusion of the relevant key elements of information that characterize the battlespace. By and large this is explicit information (an aircraft carrier is located here, moving in this direction at that speed; the adversaries submarines are still in port at 0430, etc.) that requires little interpretation and can be communicated relatively quickly and easily. It is factual, often focusing on forces, terrain, weather, refugee movements, and so forth.

Figure 9



By contrast, battlespace awareness involves identification of patterns in the battlespace information that are understood because of a priori knowledge that may exist because of training, experience, or study. For example, Soviet-trained ground forces had to move their artillery ammunition well forward prior to launching an offensive because they lacked the transport to move it fast enough once an attack was underway. Hence, information reports of ammunition convoys moving up close to the forward edge of the battle area (FEBA) and ammunition supply points being established close behind that line could be read as a warning that an offensive was imminent. However, only an intelligence specialist or commander with prior knowledge of Soviet doctrine would be able to quickly and authoritatively interpret the pattern. Battlespace awareness therefore exists only in the cognitive domain.

Battlespace knowledge also exists in the cognitive domain. However, this concept implies that the knowledge includes an understanding of the cause and effect patterns and the temporal dynamics related to the observed patterns. Hence, battlespace knowledge allows the commander and key staff members to extrapolate from the existing patterns to the emergent or future battlespace situations. Battlespace knowledge is an essential ingredient for information superiority, but neither defines it nor is a sufficient ingredient.

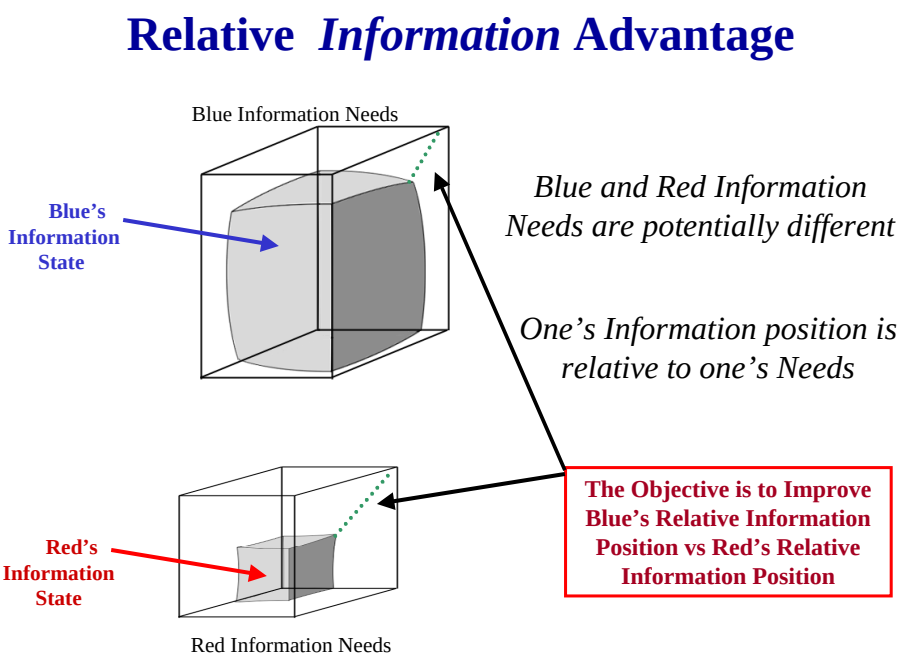
Relative Information Advantage

Information Superiority is of course a relative concept. It describes a state of imbalance that exists given two or more parties in the information domain. Some have mistakenly thought of Information Superiority simply in terms of the information and communications capabilities that one party has in comparison to others. This idea leads to an emphasis on information processes—collection, analysis, dissemination, and so forth. But this is not what Information Superiority is all about. It is important to assess a party's information capabilities relative to its needs. Concepts of operation, command approaches, organizational forms, doctrine, TTP, ROEs, level of education and training, and the characteristics of weapons systems (a mission capability package) determine a party's information-related needs. The ability of a party to successfully carry out an operation depends in large part on the degree to which these information needs are met. Parties' information needs vary considerably. Throughout history mission capability packages were designed to minimize the amount of information and communications required because capabilities in these areas were very limited. The capabilities we currently have allow us to develop mission capability packages that take advantage of these capabilities but do not force our adversaries to mirror us in this regard. Consequently we will face adversaries whose information-related needs will be asymmetrical to ours. What will matter is which party will do a better job satisfying their respective information needs, not which side has better information-related capabilities.

Minimizing one's information-related needs is, however, *not* a winning strategy. Matching concepts of operations to information-related capabilities *is*. Many advantages accrue to organizations that successfully master the art of creating and leveraging information. Using Information Age technologies, organizations can put Information Age concepts to work moving information, not people, conducting distributed operations, and substituting information for mass. The key is to find the right balance in which information-related capabilities are matched with a concept of operations, organization, approach to command and control and the capabilities of the people and the weapons systems.

Note that the metrics used to identify and quantify information superiority must be applicable across all levels of war and apply to any military missions or tasks. Just as concepts such as "combat loss ratios" can provide measures of military force effectiveness across air, land, and sea combat and can be applied to combating insurgencies as well as to conventional war, so the metrics for information superiority need to be developed at a level of abstraction that will make them robust across a wide range of cases.

Figure 10

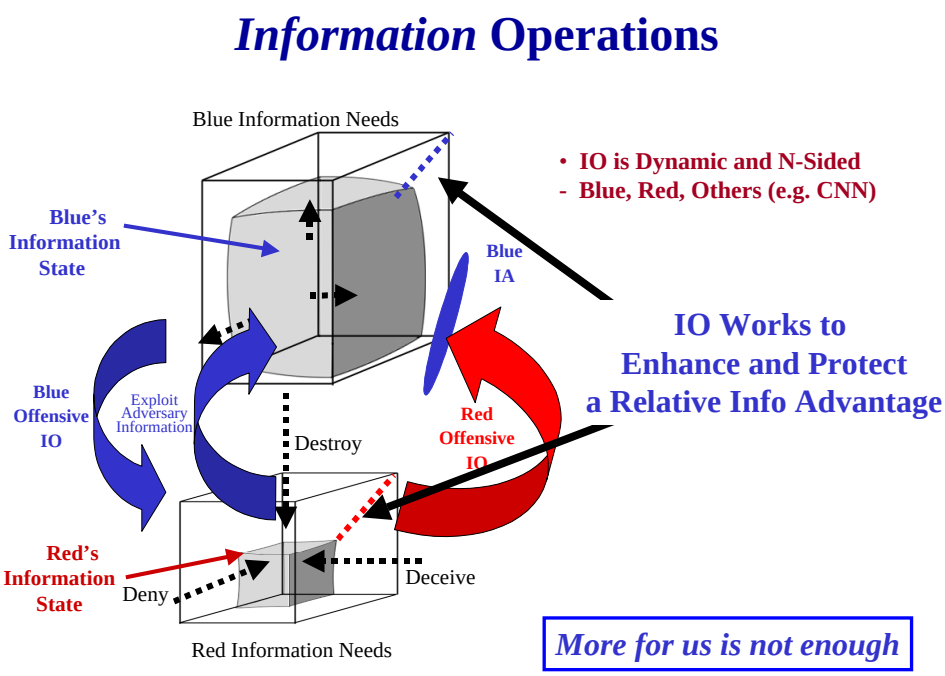


As illustrated in **Figure 10**, *relative information advantage* must be conceptualized and measured in terms of the different forces' information positions, not their information situations. In the case shown, Blue has a much greater information need, but has nevertheless come closer to fulfilling that need than his adversary. While both forces are in a negative information position, the Red force has been much less successful relative to its own requirements. Hence, Blue would be seen as having a relative information advantage.

Information Operations

Information Age Warfare will place a premium on information operations. As illustrated in **Figure 11**, both sides will seek to employ a range of tools to ensure achieving and maintaining an information advantage. These will include classic military techniques, such as destruction of assets or information denial and deception; technical approaches, such as jamming and interception; computer techniques, such as viruses and Trojan horses; as well as the use of public communications media. Information operations also include exploitation of the information systems of the adversary or items taken from it. The goal, however, remains the same—creation of a decisive information advantage.

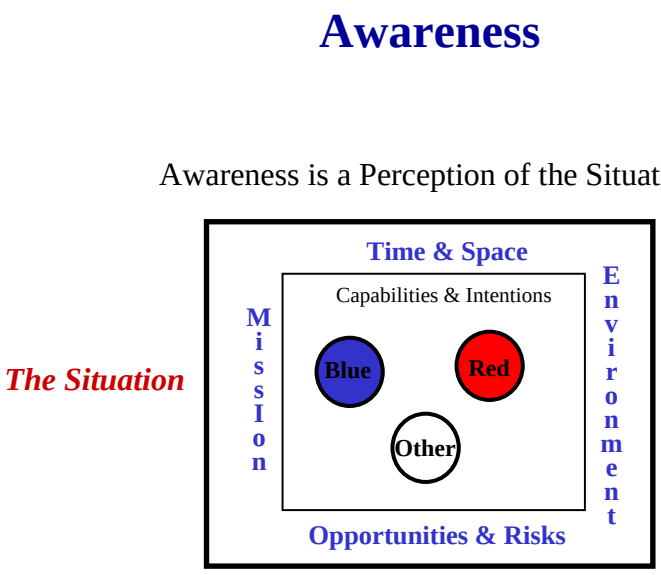
Figure 11



Awareness

When the term *situational awareness* is used, it describes the awareness of a situation that exists in part or all of the battlespace at a particular point in time. In some instances, information on the trajectory of events that preceded the current situation may be of interest, as well as insight into how the situation is likely to unfold.

Figure 12



The components of a situation are highlighted in **Figure 12** and include missions and constraints on missions (e.g., ROE), capabilities and intentions of relevant forces, and key attributes of the environment. Relevant elements of the environment include: terrain, weather, social, political, and economic elements. For most military situations, time and space relationships (e.g., weapon ranges, rates of advance across different terrain) and the opportunities and risks relevant to the forces are also crucial elements.

Battlespace Awareness

Battlespace Awareness is the result of the activities we undertake to enhance our information and protect it. Awareness *always* exists in the cognitive domain. It covers not what the information systems "know," but what the people (commanders, key staff, etc.) know and are aware they know.

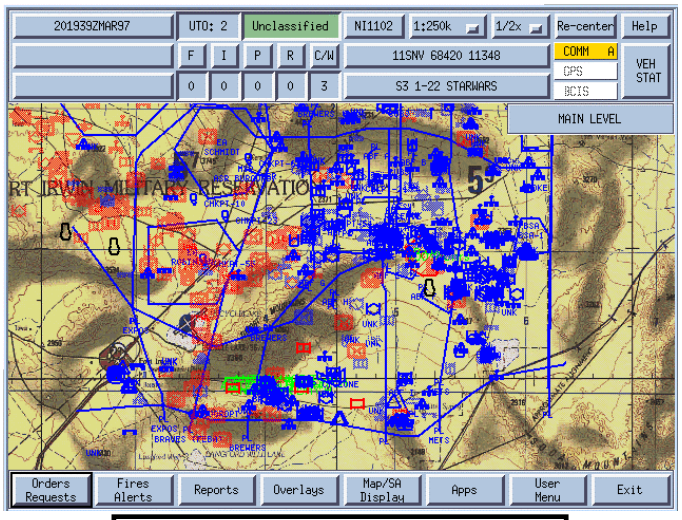
Figure 13 depicts the view to be provided for a U.S. Army Force XXI Brigade commander. Information consistent with the picture seen by this commander will also be available to others in the battlespace. This is the concept we call a common operational picture.

Note that the forces are not lined up, and that a FEBA is a concept that has limited meaning in this environment. In fact, the forces are intertwined with enemy forces, much as civilians and neutrals currently are intertwined in Kosovo. Hence, the potential for the commander's awareness to be incomplete or failure to recognize important differences is a significant element distinguishing the information system from awareness.

Figure 13

Battlespace Awareness

Force XXI Common Operational Picture for Brigade Commander



Note that this is a Non-Linear Battlespace

21

Sharing

There are two basic steps to achieving a competitive advantage. The first is generating a superior information position; the second is translating that superior information position into an advantage in one's competitive domain. In traditional combat situations, this would be combat power. In Operations Other Than War (OOTW), this would be the ability to gain the objectives at hand.

Generating a superiority information position involves both being able to improve our understanding of the situation and degrading an adversary's ability to collect, process, understand, and utilize information. In other words, it is a two-sided (actually n-sided) "game."

Having information *somewhere* is not enough. Network Centric Warfare is predicated upon the ability to not only develop good information but to share it to create shared awareness. Thus, translating a superior information position into a competitive advantage involves turning information into awareness and sharing it, creating and managing (actionable) knowledge, and then applying this understanding to the situation by effective Command and Control and execution.

The concept of sharing lies at the core of both information superiority and network-centric warfare. Sharing data, information, and knowledge creates increased awareness because different actors in the battlespace have different elements of the situation, abilities to fuse them, and experience within which to interpret what is known. At the same time, sharing is an essential process for creating shared awareness. Obviously, shared awareness is essential if actions are ultimately to be synchronized in the battlespace.

The network itself (in the case of the US, the Global Information Grid or GIG) is the entry fee for sharing data, information, and knowledge. Unless some type of network is created, sharing cannot occur.

Given that the instruments for sharing have been created, the sharing process needs to be recognized as simultaneously having organizational, behavioral, and technical components. The degree of sharing (and indeed the effort expended to share) can be understood both to have a technical feature (interoperability or the ability to link different systems in the information domain) and an organizational feature, which we have termed "cooperability", or the desire of the organizations (national systems, coalition military commanders, etc.) to genuinely share. The technical component enables sharing, the organizational and behavioral components determine the value that is generated because they control the richness of the sharing and the effort made to understand the perspectives of the partners.

Collaboration

Collaboration involves actors within the C4ISR system *actively sharing* data, information, knowledge, perceptions (awareness of facts or factors, understandings of situations, etc.), or concepts when they are *working together toward a common purpose*, and how they might achieve it *efficiently or effectively*.

Note, first, what collaboration is *not*. When information systems passively (without current, conscious human intent) share data, information, or knowledge (for example doctrinal publications) or make it available across a variety of users, no collaboration has taken place. These are simple cases of information sharing. Moreover, exchanges that are not related to a common purpose should be excluded. For example, routine reports on unit status or spot reports on enemy activity are normally only loosely coupled to the tasks, missions, or objectives of the organization and are passively, not actively shared. Similarly, routine briefings, such as the "five o'clock follies," in many command centers are not collaborative events unless the participants take advantage of them to interact to resolve particular issues.

Collaboration, then, requires active communication as part of working together. The classic example is collaborative planning, where actors with different functional and geographic areas of responsibility focus their attention on achieving assigned missions. Their goals are to share a common understanding of the situation; take advantage of their differential knowledge, expertise, information, and capabilities; and organize the activities they control in time and space such that they will (a) avoid mutual interference and (b) have a synergistic effect. In other words, they want to plan so their actions will be synchronized. Of course, collaboration may well extend into an integrated process of execution/re-planning as the mission is pursued.

Integrated product teams are essentially organizational forms designed to encourage (or ensure) collaboration takes place. They were created under the theory that complex problems often require functionally different

expertise. They assume that the costs of the collaboration (whether time or resources) will more than be recovered by the higher quality of the results, which blend the knowledge available from different sources. In some cases they are also seen as more rapid, particularly when they replace serial processes where each group waits for another to finish before they begin, or processes that shut down while people rest. In that sense, military command centers, which have employed shifts over time and used overlapping duty hours to "hand off" their knowledge and situational awareness for decades were pioneers in collaboration. However, most of that collaboration historically took place within functional areas.

Inherent in the idea of collaboration in a military context is the notion that a mission will be accomplished. When the collaboration is used to ensure more efficient mission accomplishment, the appropriate metrics are Measures of Performance that show how the same level of effectiveness can be accomplished with fewer resources or higher levels achieved with the same level of resources. These metrics often focus on residual force levels or capacity after mission accomplishment, but can also look at levels or rates of force expenditure during mission accomplishment. When the focus is on mission accomplishment itself, the appropriate metrics are Measures of Effectiveness and may also extend to Measures of Force Effectiveness.

When collaboration in the information domain is enriched, considerable improvement can be expected. First, by sharing data, the C4ISR system greatly improves the likelihood of sharing a common picture of the battlespace. When data is pooled from sensors, the quality of the underlying database can be expected to improve. That same database will also be more up to date—the delay inherent in one node in the C4ISR system sitting on a data item, placing it into a product, and disseminating that product all but disappears. In essence, the most basic fusion of information (Level 1) is greatly improved by sharing data.

Second, by sharing information more rapidly a similar effect occurs—more command centers are aware of more information sooner. This has potential synergistic impacts. The information item is seen from multiple perspectives—for example, its intelligence, operations, and logistics implications can be recognized sooner. Similarly, multiple perspectives reviewing the data increases the likelihood of anomaly detection. Working in a highly uncertain environment with adversaries who are attempting to conceal their activities and deceive the friendly C4ISR system, anomaly detection is a crucial tool in situation awareness. Finally, the more rapid dissemination of information through preset automated data sharing also allows for more rapid integration into battlespace awareness. In an information age mission, speed of C4ISR processes will often be crucial.

Finally, information age systems also allow for better availability of "pre-real time knowledge." Military forces, particularly technology heavy ones, depend on the doctrine, training, and skills of their personnel. However, not all forces are fully "up to speed" in all areas all the time. Forces train for a set of operating environments, with an expected set of coalition partners, and classes of adversaries, with particular types of equipment. The global responsibilities of the US military, however, virtually guarantee that some unfamiliar locations, adversaries, equipment, or coalition partners will be encountered. The information age military will, however, have enormous reach back capability to access knowledge and examine it both individually and collectively. Access to databases (plans and detailed maps of urban environments, order of battle data for unanticipated coalition partners and adversaries), manuals (field repairs for specialized equipment or foreign equipment used in the theater), information sets (symptoms for local diseases or biological weapons), and knowledge (local customs, adversary doctrine, profiles of enemy leaders) will all enable improved operational effectiveness.

However, this richer sharing of information does not come without costs. These will primarily be in the form of greater demands for bandwidth to deliver the shared information and computational power (either in the rear or forward) to organize and present it. The human factors problem of accepting what comes from a computer as "real" and failing to understand the uncertainty inherent in these shared items will also be an issue and must be addressed both in the training of users and in the design of the information representations.

Perhaps most important, sharing information in the battlespace will make demands for time and attention on commanders and key staff members who are already heavily burdened (physically and cognitively) and will be

tired and under stress. Early work with computers inside armored vehicles has shown that displays can distract key personnel from their immediate warfighting tasks. Hence, human factors will be a crucial element in designing successful information-sharing systems.

All collaboration passes through the information domain. Even when face-to-face, collaborators send information (voice, facial expressions) to their partners. However, true collaboration—sharing in order to work together toward a common purpose—occurs in the cognitive domain as the partners interact and develop awareness, understanding, and concepts that would not have emerged without their exchange.

Clearly, collaboration requires communication. While this is often direct, it can certainly be asynchronous. For example, academic authors have long collaborated by exchanging written drafts, with episodic meetings and discussions. More recently, E-mail has made this process faster and simpler. However, the quality of the interaction can vary greatly, depending on whether the collaborators share a common language, background and culture (national and organizational), the level of engagement of the participants (are they serious, do they accept the goal), and their confidence in the collaboration medium (including their ability to use it when technical capability is required).

The potential benefits of "cognitive" collaboration are enormous. A better understanding of the military situation and the factors that are driving it are the most obvious benefits and correspond to a common understanding of the problem in the civilian arena. The opportunity to improve planning through collaboration is also enormous—having both those responsible for conducting an operation and those responsible for supporting it involved enables a much richer plan as well as greater insight into the contingencies that can be expected. Collaborative decision making can also be expected to generate better choices, particularly when complex problems are being addressed. Finally, collaboration will improve the linkage between planning and execution. As these two functions merge, effective collaboration will provide greater organizational agility—the capacity to react more effectively in a rapidly changing operating environment.

There is one drawback that should be anticipated when collaboration is used—a loss of speed in C4ISR processes. Research into small group dynamics, decision making under stress, crisis decision making, and coalition C4ISR all indicates that collaboration slows decision making. Hence, collaboration tools need to be designed with this pitfall in mind, training in their use will be essential if the problem is to be minimized, and contingency planning (which takes advantage of the richer interaction and deeper understanding of the problem) must be used to reduce the need for new decisions in complex situations.

Synchronization

The dictionary provides an elegant and succinct definition of "synchronization" as "purposeful arrangement of things in time and space." This has direct application to the military arena and to network-centric warfare. For us, synchronization is an output characteristic of a command and control (C2) process. C2 arranges and continually adapts the relationships between and among military actions in time and space. When done well, it creates an adaptive control system that achieves the assigned missions or objectives.

Because it is physical actions that are synchronized, synchronization always takes place in the physical domain, but is the result of fusing or bringing together elements of the information, cognitive, and physical domains. The activity of synchronization allows integration across a number of dimensions—time (sequencing), space (simultaneity), warfare level (strategic, operational, and tactical), warfare arenas (air, land, sea, space, cyberspace), and organizations (echelons of command, functional organization).

Synchronization can take the form of preplanned operations, very much like American football (**Figure 14**), or flowing adaptation, more like soccer (**Figure 15**). Massive invasion operations from World War II, such as OVERLORD or the US island invasions in the Pacific were typical and generally successful examples of preplanned plays. Soviet doctrine used similar "plays" to control its forces and provide well understood patterns of events for missions such as punching through an enemy's defenses. Their plans literally took the

form of templates, beginning with the ability to recognize what types of operation were appropriate (their "correlation of forces" algorithms) to the general allocation of space and roads to attacking forces by echelon, fires, and logistic support. The strength of this approach was demonstrated in the initial success of the Egyptian penetration of Israeli positions in the Sinai in 1973. However, its limit was also demonstrated when the plan had been fully executed and the more adaptive Israeli force was able to regain the initiative and defeat them.

Figure 14

By contrast, the German blitzkrieg doctrine was an early type of the more adaptive style that characterizes soccer. Initial German attacks were carefully pre-coordinated, like American football plays. However, once a force broke through the enemy defensive lines the local commander (typically a division or corps commander) had great discretion and was expected to seize the initiative and exploit the situation. The German advantages arising from their more capable units (and excellent leadership at both the commissioned and non-commissioned officer levels) and their greater use of tactical radios early in World War II gave them the capability to adapt their tactics and operational level decision making. In Network Centric Warfare, a similar, but greatly expanded capacity for adaptation becomes available. In an ideal case, the force is literally able to "self-synchronize," with each element making constant adjustments to its actions because of its rich understanding of the battlespace as a whole, the friendly commander's intent, and the actions of other friendly forces. This is very similar to the constant adjustments needed in serious soccer, where shifts from offense to defense and adjustments in the point of attack are rapid, continuous, and decisive.

Figure 15

Figure 16

Network Centric Warfare

NETWORK CENTRIC WARFARE **Translates IS into Combat Power**

A Warfighting Concept
that Enables a Network Centric Force
(Robustly Networked Sensors, Decision Makers, and Shooters)
to Significantly Increase Combat Power by Achieving

- *Increased Shared Awareness*
- *Increased Speed of Command*
- *Higher Tempo of Operations*
- *Greater Lethality*
- *Increased Survivability*
- *Streamlined Combat Support*
- *Effective Self-Synchronization*

IS Will Enable the Realization of Emerging Concepts such as Network Centric Warfare **(DPG)**

28

Network Centric Warfare

Network Centric Warfare (NCW) (**Figure 16**) takes Information Superiority and translates it into effective combat power or in the case of OOTW into effective responses and actions.

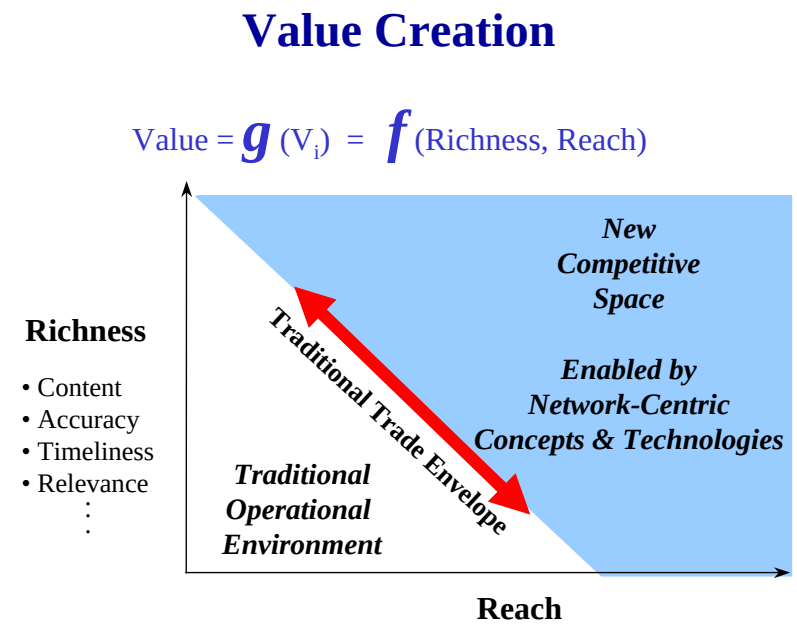
NCW is a warfighting concept enabled by Information Superiority. It is predicated upon a network-centric force, a force where sensors, decision makers, and shooters (or actors) are robustly networked in order to share awareness, develop knowledge, and be able to collaborate with one another to achieve synchronization of effort.

Significantly increased combat power is in the offing as a result of the achievement of improved awareness and the ability to share it. This, in turn, provides the opportunity to increase speed of command, improve synchronization, increase tempo of operations, and streamline combat support. The result will be greater lethality and effectiveness, increased survivability, and reduced collateral damage and risk.

Value Creation

Information Superiority and Network Centric Warfare are about creating value from information. This is nothing new. However, the information environment in which today's organizations operate is markedly different than it was just a few years ago. Evans and Wurster have developed a simple way to understand the nature of the information environment and its relationship to the ability to create value (see **Figure 17**). They describe the information environment as a two-dimensional space with one axis representing information *richness* (what we would call the quality of information) and the other axis representing information *reach* (part of what we mean by information sharing). They argue that value is a function of both richness and reach.

Figure 17



Source: Phillip B. Evans and Thomas S. Wurster, "Strategy and the New Economics of Information," ***Harvard Business Review***, September-October 1997, p. 74.

They observe that, in the past, information environments required tradeoffs between richness and reach (the traditional trade envelope) and that only recently have we been able to simultaneously get more of both—and by doing so are able to move to a new part of the information environment space (the part of the space in **Figure 17** that is called the new competitive space).

Organizations that have learned to operate in this new portion of the information environment have, in fact, been able to create an information advantage and turn it into a competitive advantage.

Value and Networks

The advent of networking has enabled us to break out of the traditional trade envelope. Networks contribute to value creation by changing the economics of information through simultaneously providing three services at affordable costs. First, by bringing together information from multiple sources to be correlated and fused, networks increase the richness of the information available. Second, networks create value by providing access to and facilitating the sharing of information, which enhances reach and creates shared awareness. Third, networks enable collaboration, which transforms shared awareness into collaborative planning and synchronized actions that create a competitive advantage.

Military Value Chain

Figure 18 uses these ideas to represent the military value chain, dividing the circle in which richness is equated to an information advantage—C2 is substituted for reach (including quality of interaction) and combat power replaces value. We believe this is a useful way to represent the information—C2-related differences in investment, system, or, indeed, in mission capability package options.

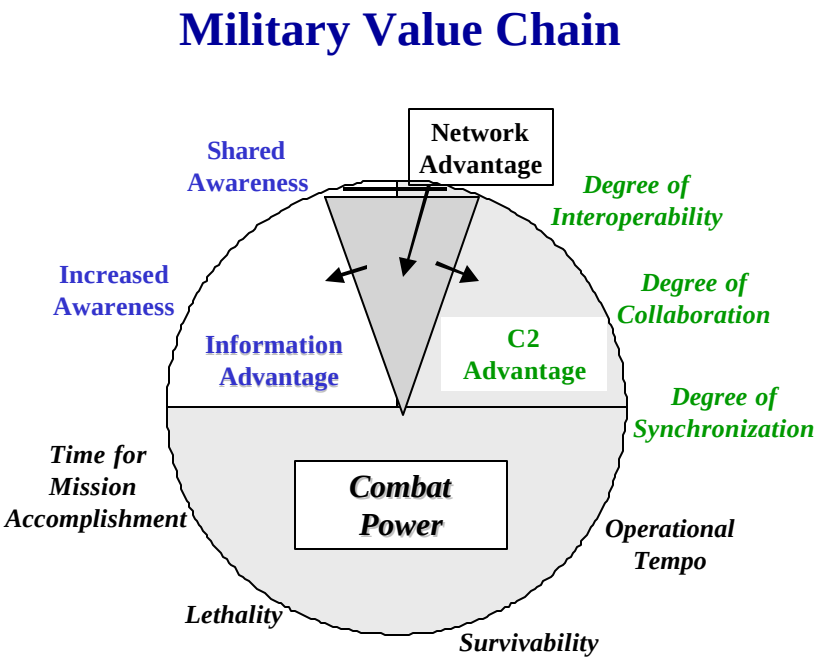
The attributes associated with an information advantage include both increased awareness and shared awareness. Both of these attributes are important because a particular innovation may only increase the quality of awareness or only share a previously achieved level of awareness. Some innovations may, in fact, affect both either positively or negatively. C2 is, among other things, concerned with communicating about the nature of the mission and circumstances with others. The degree to which members of the force can share information is related to the degree of interoperability that exists, while the manner in which they operate is related to the degree of collaboration. **Figure 18** provides a number of attributes associated with combat power. These attributes are logically arranged from left to right as the degree of synchronization relates to the operational tempo that can be achieved, which in turn may affect lethality and survivability which may be related to the time required to achieve the mission. The attributes for combat power that are selected will, of course, depend upon the situation.

Interoperability

As discussed earlier, the simple term “interoperability” has a variety of meanings or connotations. Most authors have used it to mean technical interoperability—whether two elements of the information domain can “talk” to one another. Levels of technical interoperability have also been defined, since some systems can be made to talk with one another through work-arounds and laborious processes (often involving human linkages) and others can be made fully interoperable in the sense that they literally share data and information and work together seamlessly. More profoundly, information systems also need to be “semantically interoperable.” This is a very difficult achievement, particularly when the systems belong to or were developed by military organizations from different cultures. However, semantic interoperability is an essential ingredient if true shared awareness is to be achieved, as well as for meaningful collaboration.

Cooperability, which implies organizational and behavioral efforts to share data, information, and knowledge, serious efforts at collaboration and collaborative planning, and meaningful attempts at force synchronization, is the most elusive type of interoperability, even among close allies and coalition partners. However, without cooperability, genuine synchronization is unlikely.

Figure 18



Key Challenges

Information superiority and network-centric warfare represent challenging arenas. They will not, and cannot, be achieved easily. They will not be fully achieved for some time. However, there are places to start. We have begun to understand these concepts and continue to explore them and their practical implications. For now, I see three major areas where we, as alliance partners, should focus our efforts.

In the *technical* arena, we need to work on the fundamental barriers. We must be building systems that can work together. This means engineering and testing what are increasingly called "systems of systems." In truth, since our alliance includes a number of nations with advanced technical capabilities and unique systems for sensing, fusing, storing, retrieving, searching, disseminating, and sharing information, we must work at engineering successful "federations of systems." However, development is not enough. We need robust efforts to actually test these networks and systems under real world conditions whenever possible. At the same time, to minimize the risks inherent in our information systems, we must also work together to generate protection, genuine information assurance. This will also require serious effort by all the nations, including some meaningful testing and investments, to reach an acceptable level of security risk in our coalition systems.

More challenging, but no less important, is the ability to achieve semantic interoperability across the alliance. This can start with an effort to generate common relevant operational pictures (CROP). These efforts can build on the work done within NATO to support effective operations in Bosnia and Kosovo. However, they must also extend to integration across warfare arenas (combined land, maritime, and air pictures) and to accepting inputs from the sensors and reporting systems from all members of the coalition. Of course, coalition experimentation and exercises are the forum by which this level of semantic interoperability can be established and tested.

Finally, we will need to work together to ensure an effective spirit of cooperability at both the organizational and behavioral levels. Information superiority and network-centric warfare imply new ways of doing command and control. We must explore these together, come to understand how they relate to our coalition operations, and learn to perform them effectively. Ideas such as "self-synchronized" forces hold great promise,

but could also generate increased fog and friction within an alliance unless they are well understood and embraced by all those with forces in the field. Moreover, gaining information superiority and the competitive advantage it implies assumes we are ready to engage in genuinely collaborative processes. These will only become reality if they are properly developed across the alliance and supported by genuine efforts to share data, information, and knowledge.

The only approach that will work is that of co-evolution of mission capability packages. These are integrated sets of doctrine, organization, personnel, education, training, material, and leadership. If, instead, we develop information systems in isolation, or fail to develop all the elements needed to generate new capabilities, we will not achieve genuine information superiority. If we do, the potential payoffs in national interests, lives, and treasure are virtually incalculable.

Conclusion

In conclusion, Information Superiority is not just a fashionable Information Age bumper sticker—it translates into combat power.

To enable DoD to achieve Information Superiority, collaborative efforts are required to field the Infostructure necessary to provide us with a protected dominant information position and the ability to leverage it.

This page has been deliberately left blank

Page intentionnellement blanche

Network Centric Operations: Implications for Allied and Coalition Operations

Dr. Hans E. Keus
Program Manager Maritime C4I
TNO Physics and Electronics Laboratory
P.O. Box 96864
2509 JG The Hague
The Netherlands

1. Abstract

Network Centric Warfare (NCW) or perhaps a better term would be Network Centric Operations (NCO) is rapidly becoming one of the areas where the most likely progress in efficiency and effectiveness of military operations will take place. The benefits of information technology and specifically network technology in the civil business area is starting to make itself feel in the military domain too.
In the US a lot of attention is given to NCW to start adapting US forces to the ideas of NCO. However, little is still known about how to achieve coalition-based NCO. This paper will go into some detail in discussing issues involved in Coalition-based NCO or CNCO as we will call it. After a short summary of the main issues of NCO we will try to identify some of the most important key factors involved in CNCO and discuss some of these items. Special attention will be given to concepts of interoperability. A migration path based upon the here proposed methodological approach is suggested as a means to achieve CNCO.

2. Introduction

The three basic elements of Network Centric Operations (NCO) are Information, Communications and Operations. NCO is the optimal use of information, made available by sufficient secure communications to plan and execute fast, timely and decisive operations against opposing forces or in non warfare situations (like humanitarian actions). For a discussion on some of the terminology on NCW see [Alberts et al, 1999].
NCO is the synthesised combination of these three basic elements and not just an addition of them. The combination of these factors leads to some specific areas where network centric operations have their main impact:

A. Shared situation awareness and understanding.
B. Enhanced capability for co-operative and co-ordinated planning and engagement.
C. Vertical and horizontal consultation and information capabilities.
D. Enhanced means of rapid intelligence gathering.

List 1 Main Areas of NCO improvements

The developments we expect to take place in the military domain are the same we see already occurring for some time in the civil and commercial domain. There we encounter a rapid adoption of open standards, systems and components, information becomes vastly available by means of the internet, complete company processes become net-oriented, and the rapid availability of information is speeding up business processes significantly. Completely new services come into being because of the existence of network or web technology.

In the military domain equivalent developments are likely to take place. Much effort is already put into the realisation of concepts like advanced sensor netting above, at and under water, in the optimisation of the sensor-to-shooter pipeline, in VTC facilities (video teleconferencing), and in speeding up the C2 cycle with unfortunately still poorly defined concepts like self-synchronisation. The evolution towards NCO is not a simple and evident process. There exist some real problem areas which need to be studied carefully before being able to solve some of the major difficulties involved with NCO.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Some of these problem areas are:
- the very dependence on systems integration levels and levels of interoperability;
- the consequences of the secure aspects of information;
- the dangers of information overload situations;
- backseat driving (military as well as political) because of the ease to skip several levels of command;
- the need to develop appropriate concepts of operations and doctrines;
- the fact that many of the frequently used concepts are still merely words and still poorly understood.

Another major issue is that NCO is currently a predominant US matter. Other nations lag behind or do not (yet) have the means to invest as heavily in the network centric concept. Because of this threatening unbalance the implications of making NCO sufficiently achievable for a coalition are big and challenging to say the least. Typical characteristics of a coalition are that they can be coincidental of nature, especially when we look further then only NATO. A coalition of one day may exist of adversaries of a another. The question of trust and information sharing is a major issue here.
Also when dealing with non-US forces and with some of the non European forces as well the unbalance in technology will pose serious problems for achieving a sufficiently high level of integration needed for network centric operations. NCO just takes more than adding the individual forces of participating nations together. Much will depend on the ability to achieve or having achieved already a sufficient level of interoperability. Starting from a level too low may result in an increased time required to bring a coalition-based network centric operation up to speed or even that we never achieve it.

3. Elements of Coalition-based NCO

Many developments and advanced concept and technology demonstrations on NCO are taking place in the US forces, in single service as well as in joint operations. Through these experiments a lot of knowledge and understanding is currently obtained about single nation NCO. To investigate some of the major implications involved with CNCO we will first try to identify the key areas of attention for NCO and then focus on those topics in these areas which have the most impact on the coalition aspect of NCO.

In Figure 1 the most important elements of NCO are shown. We can distinguish 4 levels which should be considered when dealing with NCO:

1. The Intentional Level, where the goals to be achieved are formulated.
2. The Human Level, with the command structure working to achieve the mission goals.
3. The C4I Level, which is the supporting layer for the human level to carry out command.
4. The Material Level, which consist of the assets needed for the mission.

The pyramid of NCO elements gives a good overview of all involved areas on which the impact, the consequences and the actual implementation of NCO should be considered. It gives a framework of what should be studied and where issues must be resolved with respect to NCO.

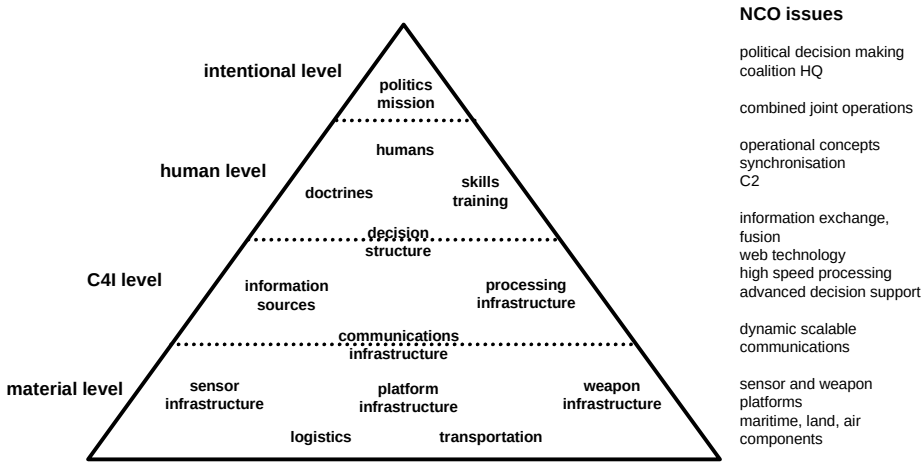


Figure 1 Elements of Network Centric Operations

The framework of Figure 1 is valid for both the one nation type of NCO as well as for coalition-based NCO. In the following sections we will look at some of the problem areas which exist when dealing with coalition-based NCO. The coalitions we will focus on are not only NATO-based, where we have US forces working together with European and PfP nations, but also coalitions which may come into being without the US or NATO.

In this paper we do not discuss every single element of the NCO pyramid. Instead we will take an approach in which we focus on three different perspectives of the pyramid:

- Political.
- Operational and Conceptual.
- Technological.

Many of the issues which needs to be considered with respect to CNCO belong to one of these three perspectives.

3.1 Political Perspective

On the level of politics a coalition can be created and it can be terminated. The political level can decide on the level of information exchange, which is so essential to NC operations, on the use of national information, given to the coalition to be used. We already have problems of information availability in our current multi-nation operations but in NCO it is even more pronounced since information sharing is a key enabler of NCO. Past *and* current operations still show us the difficulties for allies to obtain and get access to US-only information, of which SIPRNET is only an example. In other words, the principle and the amount of 'US-only information' can become a real showstopper for coalition-based NCO where the US is included. But not only the sharing of information is an issue, also the authentication of it is: how much is national information trusted in coalition networks when nothing is known about the origin?

Another key conceptual area of NCO (see *List 1 Main Areas of NCO improvements*) is co-operative planning and co-ordinated engagement. Especially in an advanced state of network centric operations and warfare we need to be able to deal with transfer of command even more and faster then we are already doing now in our current coalition operations. Especially in lethal engagement situations we have to be sure that fast and decisive decision making using multi-nation resources is ensured by the political level. The possibilities for backseat driving on the political level can be a big impediment for this.

3.2 Technological Perspective

Of a complete different nature than politics are the technological problems involved in achieving a true NCO for a coalition. The key factor here is the possible technological unbalance which can exist between different partners of a coalition. For the US the implications and implementation of advanced information communication technology and especially the focus on NCO poses already an enormous challenge. It is even more so in a coalition context. The technological gap which already exists between US and non US forces, even high ranking European ones threatens only to get wider.
One of the major improvement areas of NCO is shared situation awareness and understanding. This requires a high level of sharing of information, compatibility of information processing capabilities, both with respect to data and information fusion as well as using the same IDCRIT for instance. In the past we have seen deadly examples when this requirement is not fulfilled. For high quality common operational and tactical pictures advanced sensor netting and an interoperable information distribution and processing infrastructure is required. When we wish to assure that identical pictures and interpretations on different places exists *and* will be used to base decisions upon, the ability to fuse different sources and types of information must be sufficiently available throughout a coalition force (it should be noted that in the US this technology is currently still subject to strict export regulations).
The key technology here is Information Management (IM) of which fusion is only a part. IM involves the ability to gather, analyse, process, distribute and interpret data and information to the level of acquiring sufficient knowledge and understanding.
From a technological and also an organisational perspective solving the problems associated with the unbalance in technology will become a big challenge. One of the solutions might well be not to try to establish a balance but to solve it through other means, like using NCO capability levels as discussed in sections 3.3 and 4.

In this context it is important that we realise that not only the availability of NCO technology is sufficient but also the ability to use it, to interpret the fast growing amount of data and to make the right decisions based upon the mutual understanding. For CNCO Compatible concepts of operations (CONOPS) and especially personnel skills and training play an equal important role as the technology itself. It will simply not be enough to make some of the technology available, because the human skills to handle it will still be lacking.

Even when coalition partners have sufficiently high technology levels the underlying systems still need to be interoperable with each other. This poses another problem at the technology level. We will look into that in more detail in *section 4 Interoperability*.

3.3 Conceptual and Operational Perspective

As we have stated before it does simple not suffice to bring forces together to achieve a network centric operation. We need to carefully create the synergy between the elements listed in *Figure 1 Elements of Network Centric Operations*. This synergy can only be obtained when the principle concepts of NCO are sufficiently understood. Many information and knowledge about NCO is still on the terminology level and not on the methodological level, supported by a well defined infrastructure, information exchange requirements, information and data schemes, interfaces, etc. (see *section 4 Interoperability)*.

In the coalition context of NCO we will frequently deal with unbalanced technological levels as discussed in the previous section. To be able to handle this may call for an approach in which we distinguish levels of NCO capability which will be dependent on the state of advancement the contributing nations. The coalition NCO capability level which may be achieved will be a result of the technological C4I infrastructures, but also on the human skills and training, on the different doctrines and operational concepts of the participating nations. Interoperability is the keyword here.

In all the areas and the elements of NCO the concept of interoperability plays a central role. It is sufficiently important to be discussed separately.

4. Interoperability

Interoperability is a key factor to achieve integration with respect to systems, procedures, doctrines and even organisations and humans. Many interoperability considerations in the past were concentrated on a system level but other levels are at least as important in NCO. In Figure 2 we have given a full overview of all the areas for which interoperability must be considered and defined in the context of NCO and CNCO.

The areas which are distinguished are:
1. The action level.
2. The procedural level.
3. The hybrid system level of both human and machine.
4. The services level.
5. The world level.

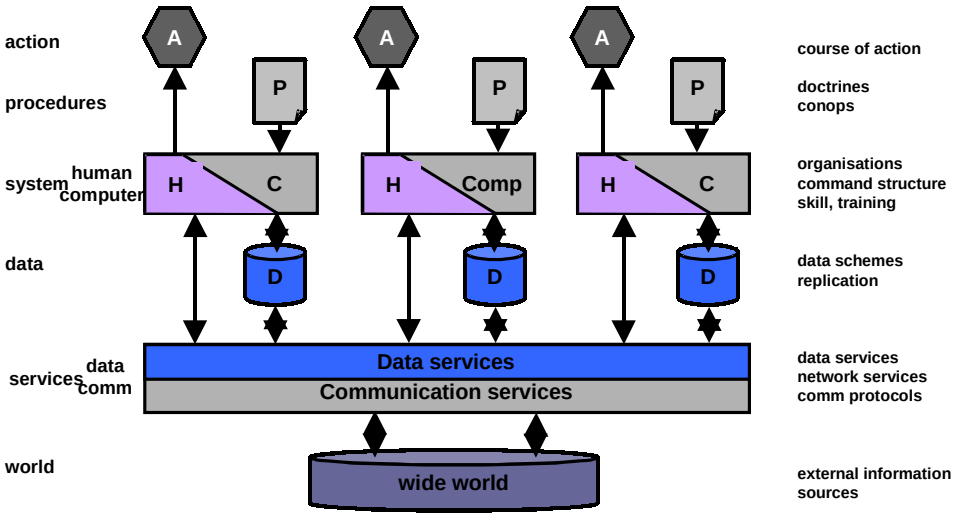


Figure 2 The Interoperability Areas

Because of the central role in NCO of information interoperability on the services and data layer is important. Because of the goals we wish to achieve with network centric operations (see *List 1 Main Areas of NCO improvements*) we need to have synchronisation and compatibility on the action and procedural levels, and on human decision making as well. This includes static aspects like interface descriptions between system components and data schemes, but it also involves dynamic aspects of the command & control process.
A structure of levels of interoperability and degrees of compatibility is needed with respect to 'Understanding the Situation' and 'Deciding on Action', the two main C2 processes. This is interoperability on the procedural and action levels of *Figure 2 The Interoperability Areas*.

In a coalition-based network centric operation the networked force infrastructure needs to be composed from its individual components into a synergetic whole. Because of the earlier mentioned unbalances which may exist not always the highest NC level of operation can be achieved. We need to be able to decide from the individual force characteristics the maximum attainable degree of NCO on the strategic, operational and tactical level. This leads to an interoperability matrix as shown in Figure 3. Here we see the various interoperability areas combined with the three operational levels. For each cell a series of standards, interfaces, compatibility levels or degrees should be defined. The most important ones are the Measures of Merit, with which we can start to develop a more methodological approach to NCO and CNCO than we have at the moment.

According to [COBP, 1999] four hierarchical levels of Measures of Merit (MoMe) can be distinguished:
1. Measures of Force Effectiveness (MoFE).
2. Measures of C2 Effectiveness (MoE).

3. Measures of C2 System Performance (MoP).
4. Dimensional Parameters (DP).

List 2 Four Hierarchical Levels of Measures of Merit

The Measures of Force Effectiveness (MoFE) focus on how a force performs its mission or the degree to which it meets its objectives. Examples include territory gained or lost, rate of advance, combat loss ratios, and casualty ratios.
The Measures of C2 Effectiveness (MoE) focus on the impact of C2 systems within the operational context. Examples include the ability to formulate plans that work to achieve objectives, the capability to create a common operating picture of the battlespace, and reaction time.
The Measures of C2 System Performance (MoP) focus on internal system structure, characteristics, and behaviour. Performance measures of a system's behaviour may be reduced to measures based on time, accuracy, capacity or a combination that may be interdependent.
The Dimensional Parameters (DP) the properties or characteristics inherent in the physical C2 systems. Examples include bandwidth of communication linkages, signal to noise ratios, component size, number and variety of wavebands, and luminosity of display screens in command centres.

Op levels / Areas	Strategic	Operational	Tactical
action			
procedures			
humans			
systems			
data			
data services			
communications			
world			

Each cell addressing issues like:

- standards

- level of :
interoperability
compatibility
connectivity

Needed are MOP, MOE and/or MOM for each cell

Defined in joint, coalition and OOTW context

Figure 3 Interoperability Matrix

The current MoMe hierarchy as described in [COBP, 1999] is not yet tailored to NCO and the specific coalition aspects of NCO. The concept of the MoMe hierarchy should be further studied and developed in order to achieve a better understanding of CNCO, to be able to identify capability levels and to reach a state in which we can start to design the different elements of NCO as given in Figure 1.

5. Recommendations

As can be derived from the previous sections it is not likely that CNCO will spontaneously come into being. In the first place the concepts underlying NCO are still poorly understood and not yet based on sound scientific methodologies. In the second place it will take a lot of effort to achieve the required amount of integration as discussed in section 4.

It is an illusion to think that from completely separately defined and developed national systems and system components a C4I architecture can be put together with NCO capabilities. The interoperability requirements to achieve the synergy required for NCO are too severe.

Therefore we shall always require a certain core capability, both in the one nation situation (read the US) as well as in the coalition situation (NATO, Europe or otherwise). Because of the requirements to be fulfilled and the technology required it is questionable whether a coalition-based NCO can ever be achieved without US or NATO elements and system components involved. To investigate the requirements of a core CNCO system

and the interfaces and interface levels we might consider using a series of JWID (Joint Warrior Interoperability Demonstrations) exercises which should be designed to contain an increasing amount of NC character.

It may very well be that for instance the NATO BI SC AIS can become the core component for achieving CNCO. The path followed in the development of this NATO BI SC AIS is given in Figure 4, where the ACE and the ACLANT specific functionality is meant to be merged into one single BI SC AIS.

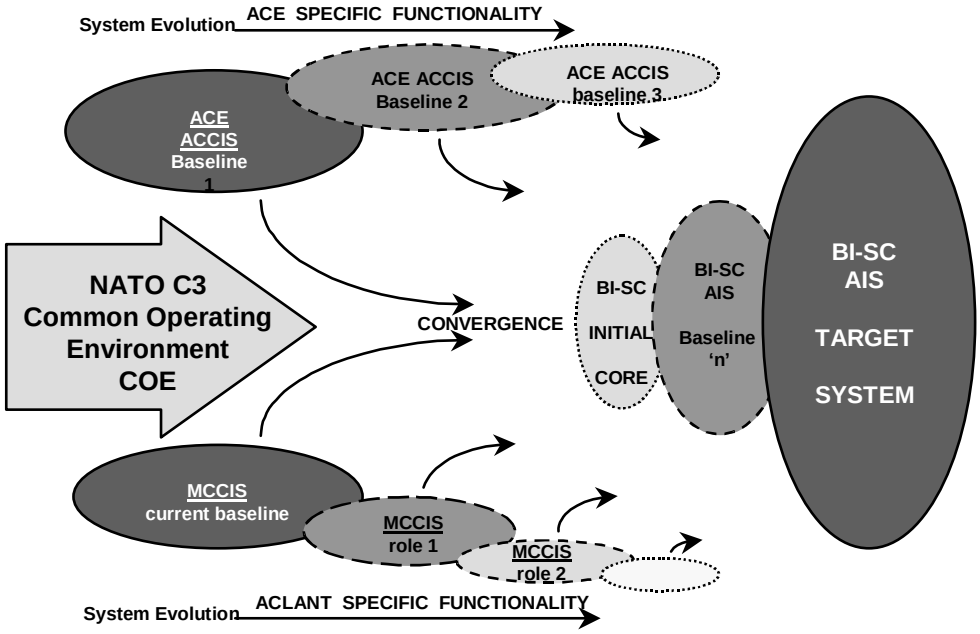


Figure 4 The Migration Path towards a NATO Bi SC AIS

Consequently we recommend a stepwise (but not strict sequential) approach to achieve desired capability levels of NCO and CNCO.

Step 1. The potential and the unavoidable evolution towards NCO must be recognised within coalition nations. Without this the initiation and synchronisation needed to develop NCO capability will not take place.

Step 2. We need to have a sound scientific fundament underlying NCO. A model and methodology with which NCO functionality can be described and with which parameters, metrics, the different hierarchical levels of the MoMe and the various capability levels of NCO can be defined. Based on this methodological approach concepts like speed of command and self-synchronisation can be better defined and be made quantifiable.

Step 3. Based on the NCO model and on the interoperability concepts discussed in section 4 we need to establish performance or capability levels of NCO. These levels will serve as the guidelines for national and coalition-based R&D plans to achieve the chosen level of NCO capability. This needs also to include strategies to cope with unbalanced technological levels of coalition partners and with restrictions to information exchange requirements.

Step 4. The core system and system components of NCO need to be identified and developed. Like we said the NATO BI SC AIS may serve as such a core system or component.

6. Conclusions

We have discussed in this paper some of the implications of coalition-based network centric operations. We are still in the process of slowly discovering what NCO is all about, what the critical factors are and how to

tackle and solve the many political, conceptual and technological issues. The differences between a one nation NCO (read the US) and coalition-based NCO are significant and will require dedicated attention to be solved.

Badly needed is a good methodological approach to NCO and CNCO. This will take a lot of effort. But we need this in order to guide the various investments of nations in the path to achieve CNCO capability. As one of the most promising and revolutionary concepts CNCO must be given a dominant place in the C4I Vision and Policy documents of many nations. By its very nature NCO in a multi-nation context can be nothing else than a synchronised and joint effort. Co-ordination of R&D efforts on NCO and CNCO within NATO and within the European Defence Force is therefore of the utmost importance.

7. References

[COBP, 1999] Code of Best Practice (COBP) on the Assessment of C2,
RTO-TR-9, AC/323(SAS)TP/4, March 1999

[Alberts et al, 1999] D.S Alberts, J.J. Garstka, F.S. Stein, Network Centric Warfare: Developing and Leveraging Information Superiority, 2nd Edition (Revised), August 1999

A Road Map to the NATO Virtual Enterprise

Y.A.J.R. van de Vijver and J.G. Stil
Information and Communication Technology Division
National Aerospace Laboratory NLR
P.O. Box 90502
1006 BM Amsterdam
The Netherlands
(vyver@nlr.nl, stil@nlr.nl)

1 SUMMARY

In this paper information management challenges are described, and ways to achieve coalition interoperability, by defining a road map towards a NATO virtual enterprise. Such an enterprise strongly supports the "interoperable communications"-target of the Defense Capabilities Initiative (DCI), launched at the NATO summit in Washington, April 1999. The building blocks of virtual enterprises will be discussed. These blocks are increasingly becoming standards, therefore allowing higher and higher levels of abstraction in interoperability. Starting from a historical example, and continuing with a Joint Warrior Interoperability Demonstration and results from a recent research program, this paper will describe the journey on the road to the NATO Virtual Enterprise. The paper will be concluded by looking forward to the goal and discuss the road towards it.

2 INTRODUCTION

In the future, more and more military operations will be conducted by a coalition of NATO nations. This places new and more important requirements on the interoperability needed for such operations.

In Section 3, the building blocks, or stepping stones, of the NATO Virtual Enterprise will be described. Although the term "Virtual Enterprise" emerged as one of Information Technology's hot buzzwords during the 1990s, the first steps on the road toward the Virtual Enterprise have already been set in the Apollo space program by increasing standardization of hardware and software components. A more down to earth example from NLR's own history of how standardization of systems has evolved over the last twenty years will be given in Section 4.
The implementation of virtual enterprises has become increasingly more feasible by recent developments in Information and Communication Technology. Combined with fast data communication, these developments make it possible for geographically distributed teams to work together as if they were co-located. NATO must embrace these developments as stepping stones toward the NATO Virtual Enterprise.

The present state-of-the-practice of the NATO Virtual Enterprise is the level of interoperability demonstrated in present day interoperability trials, for example the 2000/2001 Joint Warrior Interoperability Demonstration (JWID). During this event, a lot of military computer systems originating from various NATO nations are interconnected and operated against the background of an operational war scenario. The Netherlands' JWID 2000 interoperability demonstration (developed for the Royal Netherlands Air Force by NLR) will be given in Section 5 as an example of current NATO interoperability achievements.

In the long term, NATO Interoperability Frameworks should be aiming at aligning with commercial efforts. A possible road map towards the installation of a NATO Virtual Enterprise should consist of the stepwise adoption of the building blocks of such an enterprise, for instance, a NATO Command and Control Working Environment. Initiatives towards this goal are taken (e.g., NATO C3 Agency's Virtual Command Center). Results from research programs may provide additional capabilities to support and improve these initiatives. One example of such a research program is EUCLID RTP 6.1, entitled Advanced Workstation for C3I, which finished end of 1998. This program resulted in a common business model for C2, and in a demonstrator based on a multi-agent system architecture and an ATCCIS-compliant ontology. These were used to develop a dozen agent-based decision support tools from seven European countries, communicating via a CORBA-compliant

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

communication layer. The results of the EUCLID RTP 6.1 project will be described in Section 6. The road map to the NATO Virtual Enterprise will then be further elaborated in Section 7.

3 THE STEPPING STONES TOWARDS A VIRTUAL ENTERPRISE

A Virtual Enterprise can be defined as ”A temporary alliance of parties, come together to share core competencies and resources in order to better respond to opportunities and threats, and whose co-operation is supported by computer software and networks”. It presents an option to exploit opportunities and to provide products/services that no single party may be able or willing to provide alone. Alliances of parties are not new: already in the 1960s a number of aerospace projects, such as the Apollo space-project, satisfied this definition. NATO itself is a prime example. New is the intensity of the use of ICT means, connecting the parties in real-time and enabling real-time situation assessment and decision making.

The ICT means can be joined together into groups of capabilities according to their functionality. These capabilities are considered the building blocks or stepping stones[1], necessary to enable the Virtual Enterprise. The road to the Virtual Enterprise is constructed from these stepping stones starting at the lower level of Communications, and progressing to the level of end-user Applications (see Figure 1). Security and Management Services should be active through all building blocks and require extra attention within international collaboration.

A key enabling technology and catalyst in the set-up and maintenance of virtual enterprises has been the technology of heterogeneous distributed networked environments. These environments, which are part of the Computer Network stepping stones in Figure 1, support instantaneous collaboration across organizational and geographical boundaries, while protecting information and other assets against unauthorized access. Integration of network and information infrastructures can only efficiently be carried out if these rest on open standards which continuously comply with the speed of change in technology and which are supported by state-of-the-art tools.

Early on, NLR’s Information and Communication Technology Division has recognized the need for an organization-wide solution and piloted basic building blocks for what became part of the SPINEware middleware [5], which shields users from lower level complexities. Commercial vendors now also provide these building blocks, for instance, Samba-server, which allows Windows-based PCs to access files on Unix workstations, and VMware, which enables windows-based applications to run on Linux workstations. Within the military community, standardization of this layer of stepping stones is encouraged by, for instance, NATO OSE and the DII-COE Common Operating Environment specification.

[1] Stepping-stone (Collins):
One of a series of stones acting as footsteps for crossing streams, marshes, etc.
A circumstance that assists in progress towards a goal

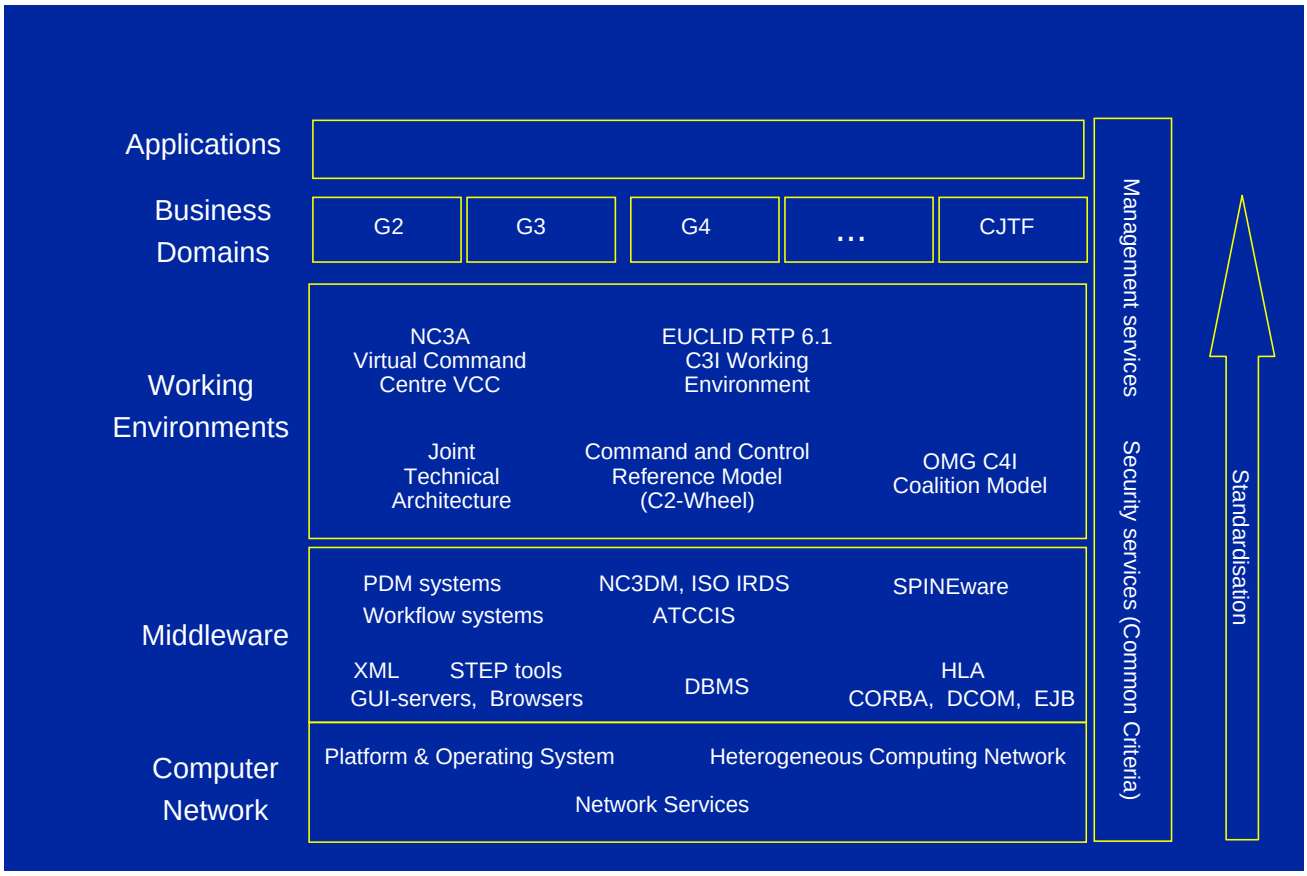


Figure 1 Stepping stones of the Virtual Enterprise

The Middleware stepping stones de-couple application-specific capabilities from any dependencies on the "plumbing" layer that consists of heterogeneous operating systems, hardware platforms and communication protocols:

- Database Management Systems (DBMS) take care of the storage and handling of data.
- Component integrators such as CORBA and OLE/COM/DCOM separate monolithic applications into components, which can be located where it is most cost efficient to execute them (e.g. close to a database engine).
- Web-based user interfaces using a web browser are platform independent and provides the same look-and-feel on any platform.
- Stimulated by the US Department of Defense, the High Level Architecture (HLA) encourages simulation re-use and interoperability.
- Information exchange languages, such as the Standard Generalized Mark-up Language, the HyperText Mark-up Language HTML, and the Extendable Mark-up Language XML standardize information exchange.
- STEP, the Standard for the Exchange of Product Model Data, is a comprehensive ISO standard (ISO 10303) that prescribes how to represent and exchange digital product information. In order to do this, STEP covers geometry, topology, tolerances, relationships, attributes, assemblies, configuration and more.
- Product Data Management (PDM) and Workflow tools can be applied to support standardization of products and processes. Examples of commercial workflow tools for the latter types of application are Windchill and Enovia. These elaborate packages, that combine product data management with workflow capabilities, are being used by main aerospace industries.
- A common Data Model based on ATCCIS, ISO IRDS (Information Resource Dictionary System Framework [ISO 10027 1990]) and the integrated NC3DM (NATO C3 Data Model) allows sharing of information on a higher level.

The Working Environments stepping stones provide the tools for creating a user-oriented, single, virtual computer that hides the details of the underlying heterogeneous network, and that may be tailored to support particular business domains, such as G2, G3, G4, and also the Combined Joint Task Force (CJTF) Center.

Working environments may cross organizational boundaries and therefore provide the environment for the virtual enterprise. Based on these lower layer stepping stones, Business Domains can be constructed to fully exploit the high level of interoperability created by these stepping stones, without already becoming application-specific. The stepping stones in these layers will be discussed in more detail in sections 6 and 7.

In addition to these functional layers of stepping stones, some general services have to be standardized as well. The security service stepping stones are of utmost importance in a military environment. In a complex and multi-company environment the security policy could apply at different levels: The internal network and systems of each partner, the communication links between partners, the access "doors" to each company network, the communication software between partners (e-mail, ftp etc.), the data, the responsibilities, different national laws, etc.
Each company could have different a security policy and a fundamental issue is the level of trusted relationship that is introduced between the partners companies.
For Virtual Enterprises a simple way of proceeding follows these rules:

Each party guarantees a basic level of security on its internal systems following policies and procedures;
Each party applies security mechanisms on the access "doors" to internal systems, complying with internal policies and procedures;
Common security mechanisms are applied on the communication links and software harmonizing security policies and national laws.

To this end stepping stones such as User identification, Perimetrical Security, Data Security, Access Control, Cryptographic Mechanisms, Anti-virus tools and Firewalls should be addressed.

The Common Criteria, which have formed the basis for standardization of security services, have now been merged with commercial standards, resulting in the ISOIEC 15408 IT-Security Standard for dual-use.

The Operation and Management Services includes the framework for managing the assets of the Virtual Enterprise and/or the projects via which the goals of the collaboration are established. Such a framework usually includes process management with PDM and ERP tools, configuration management tools, quality assurance (as ISO 9000 and CMM), information storage management, performance monitoring and disaster recovery.

4 HISTORY SHOWS THE WAY

An example from NLR’s history that shows the increasing need for and use of standardized, Common-Of-The-Shelf (COTS) tools is the Operations Management Information System OMIS [4]. OMIS is a command and control system to support the Royal Netherlands Air Force in its task to prepare aircraft for missions to be flown. OMIS has been in use at Volkel Air Force Base in the Netherlands since 1983.

OMIS assists in the communication of relevant information between different control centers and units at an Air Force Base. OMIS provides all users with consistent and up-to-date information, needed to perform their task, for instance, allocation of aircraft, fuel, pilots, and weapons. Air Task Orders and Air Task Messages are processed and communicated as well as reports to higher command levels. Air Traffic Control information on planned and actual times of departure and landing of aircraft are registered. Changes in Alert Status are distributed to all connected units upon arrival. A schematic overview of the OMIS functionality is shown in Figure 2.

The 1983 OMIS consisted of tailor-made application software running on COTS hardware, which consisted of DEC PDP-11/84 minicomputers, interconnected with each other via DECNET (including crypto-devices), and DEC VT-420 terminals. In addition to the application software, functionality that is less application specific, like database management or data replication, was also tailor-made.

The modernization of OMIS (called OMIS-2) was triggered by the lack of interoperability capabilities and by the technological advances in commercially available hardware and software. The lack of interoperability capabilities led to a complete redesign of the data model at application level. The ATCCIS (Army Tactical Command & Control Information System) standard data model was used as a basis for the new application data model. All entities in the OMIS-2 functional environment were re-analyzed, normalized and placed in a

so-called *ATCCIS-able* data model. Adoption of the ATCCIS concept facilitates future coupling with other national and possibly international Command and Control systems that are based on the ATCCIS model.

Figure 2 Overview of OMIS functionality

The advances in hardware and software led to the adoption of, for instance, the Oracle Relational Database Management System for implementing the new data model. At network level the interoperability requirement was met via the application of standard network hardware and software (PC's operating with Microsoft Windows NT4). The OMIS-2 user interface is based on the Microsoft Window Multiple Document Interface to display the various windows (in OMIS-2 called totes). OMIS-2 has been installed at Volkel Air Force Base in the Netherlands in the middle of 1999 and has been successfully in operation since.

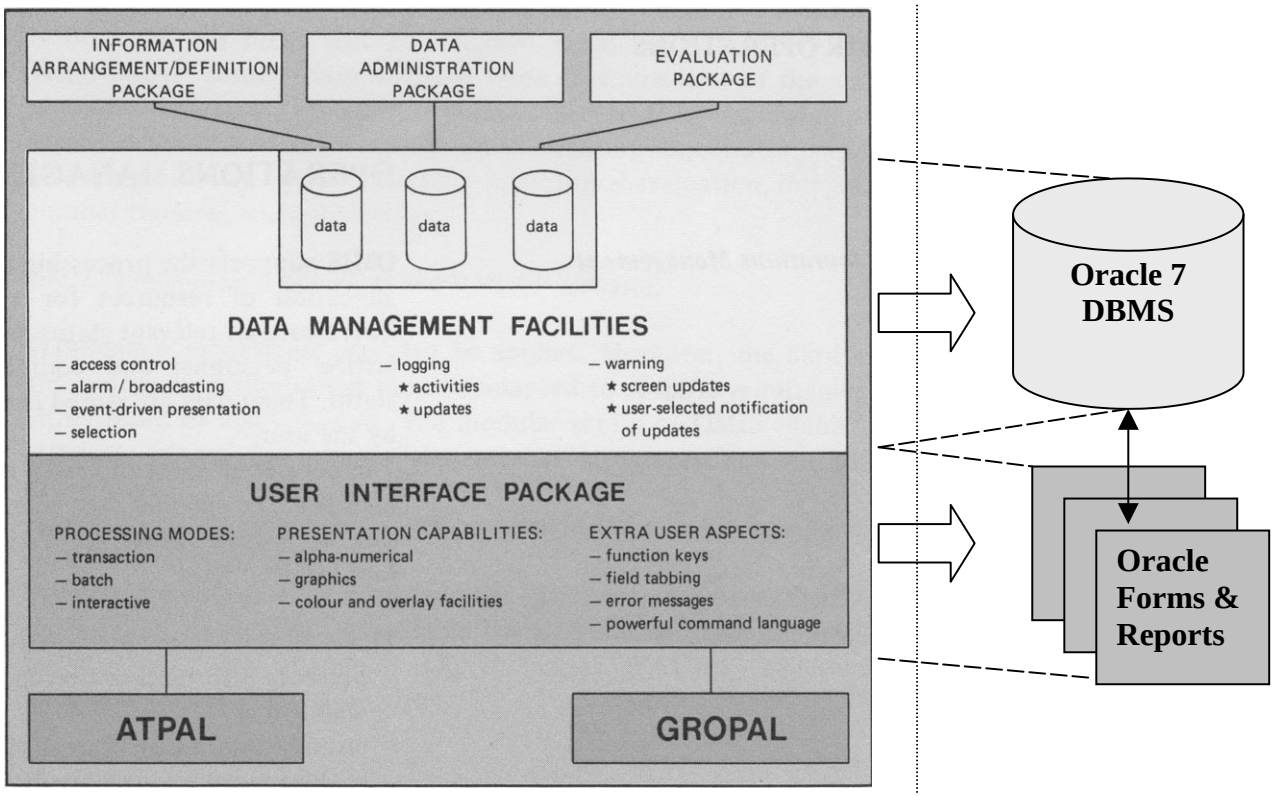


Figure 3 Comparison of standardisation between OMIS (left) and OMIS-2 (right)

Figure 3 shows the difference between OMIS and OMIS-2 with respect to commercially bought and tailor-made hardware and software, and is an example of the progress in standardization between the early 1980s and the late 1990s. Referenced against Figure 1 (see following section), OMIS used standard platform and

operating systems on top of network services, and OMIS-2 added the Database Management System and Graphical User Interface Builders to the standard, and ATCCIS as a first attempt towards a Command and Control business domain definition.

5 THE PRESENT

As the example in the previous section shows, the current status of standardization is at the level of middleware (see Figure 1 in Section 3). Database management systems are no longer developed for an application, but simply bought from commercial vendors and tailored to the need of the application. A Graphical User Interface (GUI) is built with the help of tools (called GUI Builders) that produce standard layouts and handlers which, again, may be tailored to the need of the application. Similar stories may be told for communication middleware such as CORBA and DCOM, exchange languages such as HTML, SGML, XML and STEP, the High Level Architecture (HLA) for simulation re-use and interoperability, and increasingly for product and process management tools (PDM, Workflow).

NATO also has a lot of work already in progress to achieve coalition interoperability. An example of an environment where already a lot of interoperability trials are carried out, is the Joint Warrior Interoperability Demonstration (JWID).

JWID interoperability activities concentrate both on the exchange of messages that are formatted according to messages formatting standards and on the information storage structure within military systems. With regard to message exchange standards, both military and civilian standards are considered. Examples of military message text formatting standards are the Allied Data Publication no. 3 (ADatP-3), the US Message Text Formatting standard (USMTF) and the *Over The Horizon-Gold* (OTH-GOLD) standard. Examples of non-military standards are the afore-mentioned SGML and XML. In the context of application-internal information storage structures, a typical example of standardization is ATCCIS (Army Tactical Command and Control Information System). The Army Tactical Command and Control Information System project is developing specifications to share data automatically between different command and control systems of participating nations. The ATCCIS Replication Mechanism (ARM) enables selective data replication between Command and Control systems that adopted the ATCCIS standard for their internal data structure.

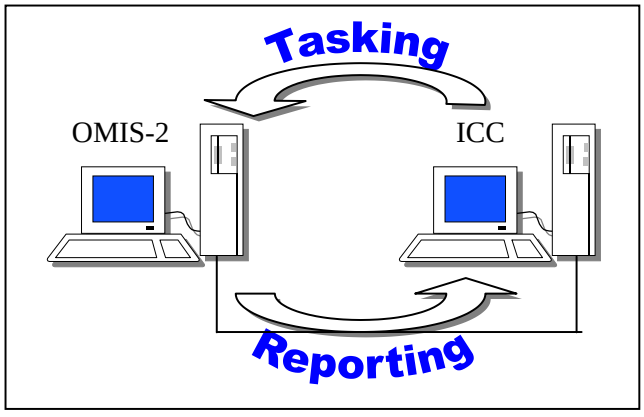


Figure 4 Database Link Principle

The Royal Netherlands Air Force and the National Aerospace Laboratory (NLR) of the Netherlands participated in JWID'00 and demonstrated a prototype of an interface between ICC and OMIS-2 (see Figure 4). ICC (Initial CAOC (Combined Air Operations Center) Capability) is a NATO system developed by NC3A and operational at CAOC Kalkar. OMIS-2 (Operations Management Information System) is a national Command and Control (C2) system, of which the software has been developed by NLR. It is operational at Volkel Air Force Base. The implemented prototype interface is meant as a replacement for the swivel chair interface that has been operational so far.

Both ICC and OMIS-2 are client / server systems using Oracle databases. The client applications connect to their database via SQL*Net, a standard Oracle networking product on top of the TCP/IP protocol. The interface between OMIS-2 and ICC utilizes the same SQL*Net protocol.

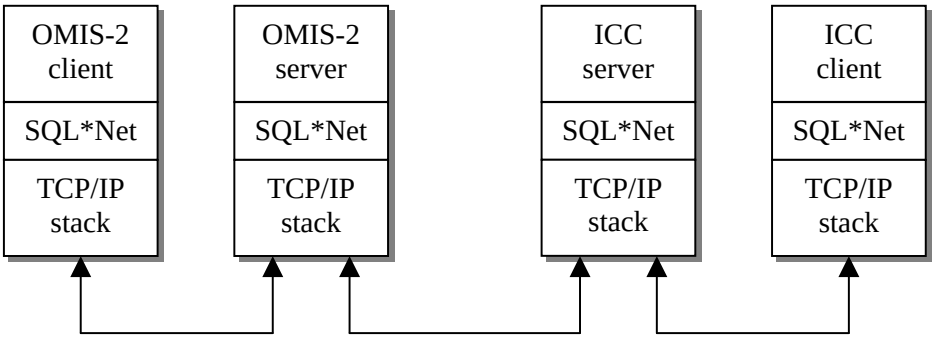


Figure 5 Interface Network Concept

The connection between the two database servers is actually a client / server connection where the OMIS-2 database server acts as client of the ICC database server. The initiative to exchange information comes from the client (OMIS-2). The connection between the two databases is realized using an Oracle database link. This mechanism is part of the distributed database option of Oracle. Database links provide the user access to data stored in a remote database. Remotely stored data can be manipulated in a similarly way locally stored data is manipulated (see Figure 6). Synonyms in the database are used to make the actual location of the data completely transparent. This technique is also used to access data stored in the OMIS-2 REAL database from the OMIS-2 CPX (exercise) database.

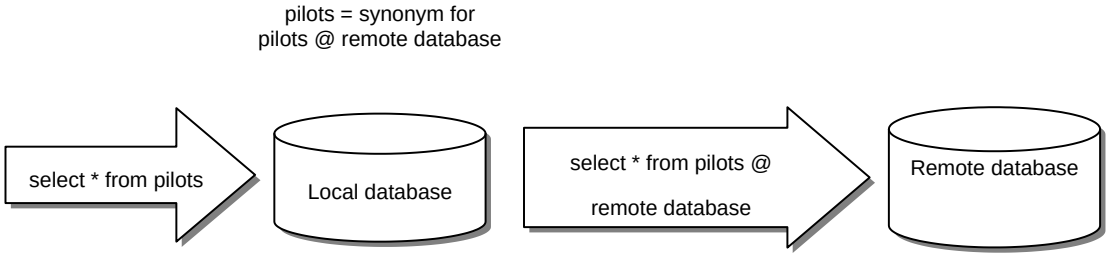


Figure 6 Database Link Principle

6 THE NEXT STEP

As shown in the previous section, the current state-of-the-practice of interoperability and standardization is at the level of the Middleware stepping stones in Figure 1. These stepping stones may be considered to be fairly standardized and integrated in operational systems to enable interoperability at this level. In the short term, a similar level of standardization must be sought for the layer of Working Environments. Necessary components for interoperability at this level of abstraction are a common process definition and a common data model definition.

An example of a command and control working environment is the demonstration environment created within the EUCLID (European Co-operation for the Long-term In Defense) Research and Technology Program 6.1, called Advanced C3I Workstation [2]. This workstation consisted of the following key innovative technologies:

- An agent-based information/software architecture to integrate diverse artificial intelligence based applications, and
- An integrated suite of command decision support tools applying AI technologies.

The integration architecture comprises a multi-agent system architecture [3] for developing and running software structured as multiple co-operating, intelligent agents, and a user interface framework that provides

the user interface between one or more users and multiple agents, including a map-and-overlay display called the DOHP (Digital Overhead Projector). Both components embrace CORBA object request brokering, enabling multi-agent software to be distributed at run-time across a mixed network of workstations and PCs.

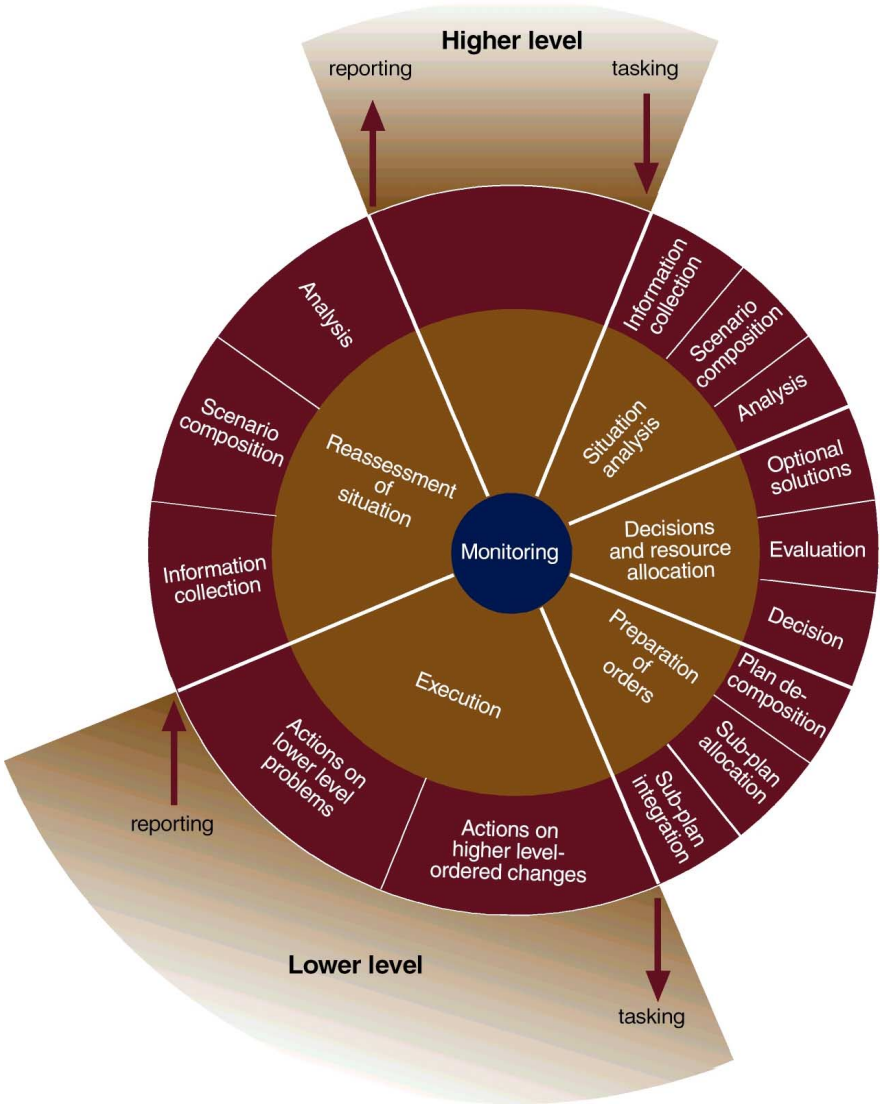


Figure 7 General Command and Control Process (C2-wheel)

One of the major achievements of the EUCLID RTP 6.1 program has been the development of a general command and control reference model, called the “C2 wheel” (see Figure 7). This reference model is a consolidated process model for army, navy and air command and control, accepted by 7 European countries. This model has been submitted to the Object Management Group OMG during the Coalition Day Event in Manchester, United Kingdom, 18-19 April 1998, to serve as a basis for the OMG efforts to develop a C4I domain and process model (the Coalition Model).

The integrated suite of decision support tools in the EUCLID RTP 6.1 Working Environment contains some 14 tools to support different aspects of joint army-air and naval-air situation assessment and planning, grouped according to the C2 wheel:

- For report analysis and situation assessment, decision support tools have been developed for automated message processing, wide area picture compilation, using fuzzy logic and clustering algorithms to identify significant enemy behaviors and groupings, and a publish-and-subscribe mechanism to notify other decision support tools of changes to the wide area picture;
- For army-air decision support, planning and tasking, decision support tools have been developed for storing, displaying and manipulating vector feature data, for automated terrain analysis & mobility

corridor construction, for Course of Action comparison based on Weapon Effectiveness Indices and Weighted Unit Values, for maneuver planning, for ORBAT browsing, and for air and fire support resource allocation [1];

- For naval-air decision support, planning and tasking, decision support tools have been developed for engagement co-ordination, for terrain analysis, for terrain exploitation and display, for maneuver plan browsing and situation prediction, and for maneuver co-ordination and formation planning.

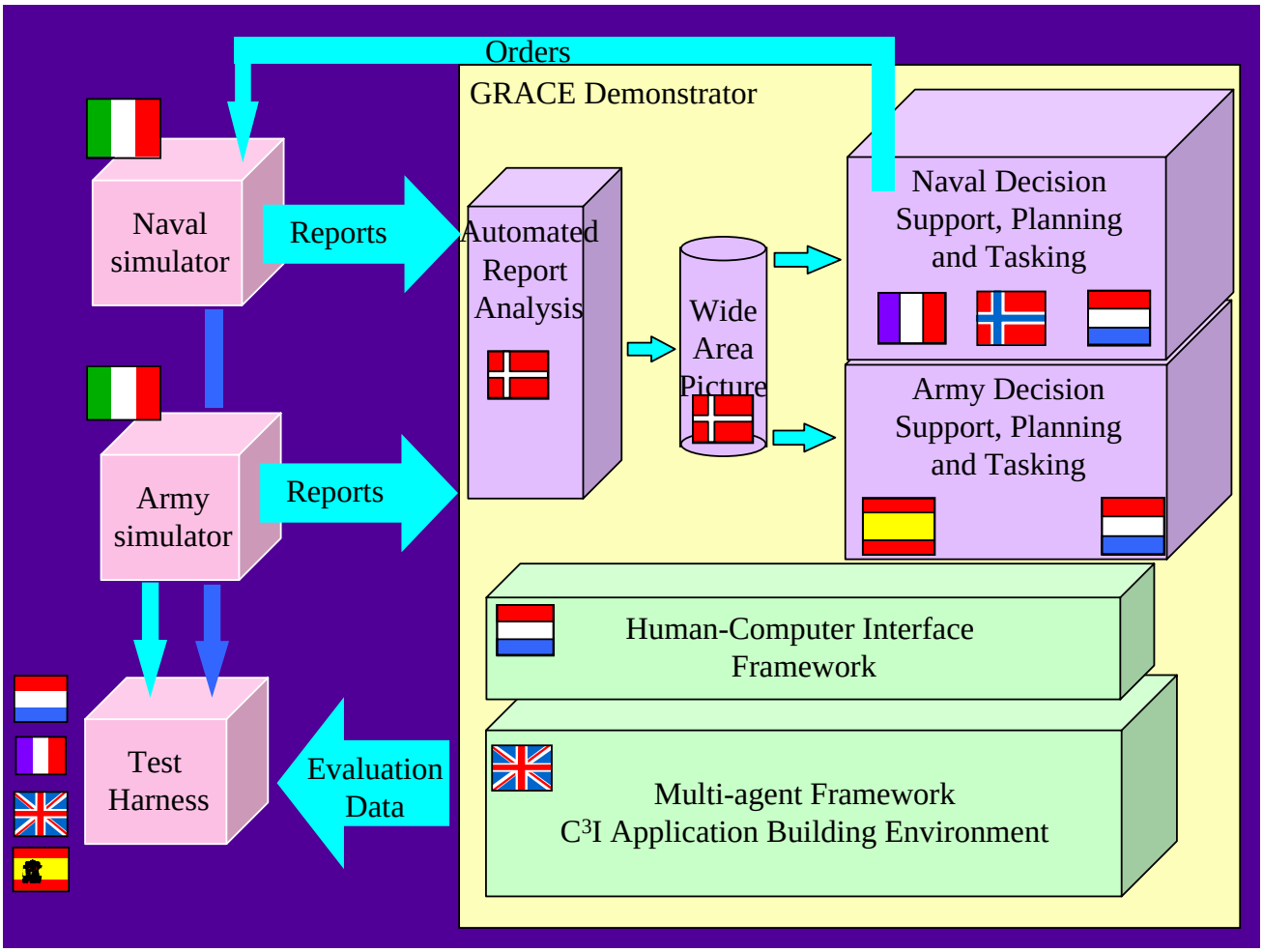


Figure 8 Overview of components of the EUCLID RTP 6.1 Workstation

These tools were generally intended to support the human command team, by automating only those aspects of a task that are better suited to the machine. Each of the tools was evaluated and demonstrated using simulated data from a naval landing force scenario or an army peace enforcement scenario. The benefits from the use of artificial intelligence techniques compared with manual planning are:

- Automatic alerting to significant events or changes in the situation;
- Quicker planning;
- Consideration of more alternative plans;
- Improved consistency and accuracy of plans, e.g. through plan critiquing;
- The ability to take more constraints (time, space, terrain, resources) into account.

One of the other major accomplishments of the EUCLID RTP 6.1 program was the development of a common object oriented data model, based on the ATCCIS model. This data model handles information like:

- Units and Organizational structures, including status, organizational dependencies, position, perception information and encyclopedic keys
- Weather information
- Facilities such as bridges, oilrigs and ports
- Naval and army unit information, including air assets.

The model is used by some components as an internal data storage format, and by all components as an external exchange format. Most of the decision support tools used the common model to structure information associated with the services they provide and request. This is particularly the case for all situation and encyclopedic information, which is held by the situation analysis component and provided to requesting decision support tools.

Some of the decision support tools do not use the common model as their internal software implementation, because in the program the common model was derived a-posteriori from the information requirements of the different decision support tools. In future developments, the common object model can be used a-priori of the development of interoperable decision support tools. It is worth noting that at the more detailed levels of the model interesting discussions emerged and were resolved about how to represent both army and naval concepts in the same model. Such issues are of increasing importance with the increased need for in joint operations.

A common object model in its own is not enough to ensure valid interactions between distributed components. It is also necessary to have agreed services and service protocols. For the future it would be beneficial to relate these aspects to agent communication mechanisms such as KIF, KQML and the FIPA standards, although none of these are yet sufficiently well established that adopting them yields real interoperability benefits.

7 THE ROAD TO THE NATO VIRTUAL ENTERPRISE

In the previous sections, the Virtual Enterprise concept has been introduced and the effort of ICT on the road to the virtual enterprise has been demonstrated. Common working environments enable crossing of organizational and geographical boundaries. Standards for information exchange are applied successfully, both in civil and military environments.

There are still a large number of open issues requiring further developments. In spite of the fact that most projects are currently focussed on the development of a Virtual Enterprise infrastructure, various aspects remain without proper solution. Much more work is necessary to support the dynamical creation and reconfiguration of virtual enterprises. NATO operations usually are of shorter time-spans and out-of-area. Therefore, the NATO virtual enterprise must be able to be created and reconfigured fast. In addition, the NATO virtual enterprise for such operations cannot always rely on large-bandwidth connections. The NATO virtual enterprise must also be able to cope with temporarily losses of connections and must be safe against information warfare attacks (information assurance). With respect to this last item, the dual-use ISO/IEC 15408 IT-Security Standard is particularly worth mentioning.

These special NATO requirements on virtual enterprises are not the focus, or at least not to the required extent, of commercial developments. Therefore, NATO research must concentrate on these specific problems and on technology to integrate solutions with commercially, and for dual-use, developed virtual enterprise technology. In this way, NATO may benefit largely from the commercial research and move quickly ahead on the road towards the NATO Virtual Enterprise. In doing so, special emphasis must be placed on business modeling. A further elaboration of the presented C2 wheel, for instance in the C4I model adopted by the Object Management Group OMG, could be an excellent starting point.

8 CONCLUSIONS AND FURTHER WORK

In the future, more and more military operations have to be conducted by a coalition of NATO nations. This places new and more important requirements on the interoperability needed for such operations.

In this paper has been described how information management challenges may be met in achieving coalition interoperability by defining a road map towards a NATO Virtual Enterprise. Such an enterprise strongly supports the “interoperable communications”-target of the Defense Capabilities Initiative (DCI), launched at the NATO summit in Washington, April 1999.

An example from NLR’s own history of how standardization of systems has evolved over the last twenty years has been given in order to show that the first steps on the road towards virtual enterprises has already been taken years ago. The implementation of virtual enterprises has become increasingly more feasible by recent developments in Information and Communication Technology. The stepping stones toward a virtual

enterprise have been described. NATO must embrace these developments as stepping stones toward the NATO Virtual Enterprise. Further work within NLR will concentrate on a further elaboration of the C2 wheel, on ontology, and on joint business models. Working environments will be based more and more on COTS, thereby pushing the level of standardization upwards.

An example from the 2000/2001 Joint Warrior Interoperability Demonstration (JWID) has been given to indicate the current NATO interoperability achievements. An example of a research program has been given to indicate possible ways forwards in the short-term future. In the long term, NATO Interoperability Frameworks should be aiming at aligning with commercial efforts. A possible road map towards the installation of a NATO Virtual Enterprise has been described.

9 REFERENCES

1. Y.A.J.R. van de Vijver, *Time-critical allocation of Tactical Air Resources to Targets*, In: Proceedings of the NATO/RTO Symposium on Advanced Mission Management and System Integration Technologies for Improved Tactical Operations, Florence, Italy, 27-29 September 1999
2. http://public.logica.com/~grace
3. Chris Dee, Paul Millington, Ben Walls, and Tim Ward, *CABLE: A multi-agent architecture to support military command and control*, In: Proceedings of the Practical Application of Intelligent Agents and Multi-agent Systems (PAAM'98), London, 1998.
4. J.G. Stil, *Modernizing OMIS, an operational Air Force C2 system, using COTS hardware and software products*, In: Proceedings of the NATO/RTO Symposium on Commercial-Off-The-Shelf Products in Defense Applications "The Ruthless Pursuit of COTS", Brussels, Belgium, 3-5 April 2000.
5. B.C. Schultheiss, E.H. Baalbergen, *Utilizing supercomputer power from your desktop*, HPCN Europe 2001 conference, 25-27 June 2001, Amsterdam, the Netherlands.

This page has been deliberately left blank

Page intentionnellement blanche

Netcentric Warfare for Dynamic Coalitions: Implications for Secure Interoperability

Mark McIntyre and Sherri Flemming
Defence Research Establishment Ottawa
3701 Carling Avenue
Ottawa, Ontario, K1A-0Z4
Canada

1. ABSTRACT

The term network centric, or netcentric, warfare is commonly used in the military literature to connote the future of military operations where timely and ubiquitous sharing of information among operational forces will lead to dramatic improvements in mission effectiveness. If this mode of operation is to be successfully employed in coalition operations, then particular attention must be paid to secure interchange of information among the coalition partners. In this paper, various modes of netcentric warfare are discussed and the implications of introducing information security at various levels are presented. In particular, the network quality-of-service requirements for the different netcentric modes are discussed with emphasis on the trade-offs that must be made when information security is an important consideration.

2. INTRODUCTION

The complexity of the environments in which modern conflicts will be waged has lead to an increasing interest in, so-called, netcentric warfare. Littoral and urban operations where land, sea and air forces may be required to work together in unpredictable and high-activity areas are prime examples of a complex theatre. The goal of netcentric warfare is to improve the efficiency and effectiveness of military operations in these difficult environments through synergistic employment of sensors, decision makers and effectors within an information network.

A netcentric capability of any degree is built upon reliable and secure data links coupled with networking and procedural standards shared by participating units. This insures that required information can be exchanged, be it sensor data, environmental awareness information or command and control directives. Examples of netcentric operation at both the command and sensor level are discussed in references [3-6]. Achieving such a capability is a challenge for national forces but it is a far greater challenge for coalition forces where the units participating in a netcentric operation may have different equipment, policies, procedures, and information security needs. The purpose of this paper is to examine the secure interoperability capabilities that are required by dynamic coalition partners in order to insure that they can engage in netcentric operations.

For the purposes of this paper we will define a coalition as a military body composed of forces from more than one nation and/or service that changes over time and is brought together and structured for a particular operational purpose. NATO's Combined Joint Task Force (CJTF) concept is an example of a coalition. A Task Force is a temporary grouping of units, under one commander, formed for the purpose of carrying out a specific operation or mission [2]. 'Combined' denotes operations or organizations in which elements of more than one nation participate and 'joint' entails the involvement of elements of at least two services [2]. A CJTF, therefore, is a deployable multinational, multi-service formation generated and tailored for

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

specific contingency operations. It could cover a wide range of potential missions including conventional deterrence, humanitarian relief, peacekeeping or peace enforcement, crisis response, and conflict.

Unlike the NATO example above, we would also like to consider coalition arrangements where the partners will not have a well-established trust relationship upon which secure international information exchange usually rests. These coalitions will be called dynamic coalitions. We will assume, though, that there is some established policy for exchanging information securely with dynamic coalition partners. A goal of this work is to develop guidelines for a reasonable security policy for netcentric operations among coalition partners where there will likely be strong asymmetry in secure networking capability.

We begin by discussing the various decision cycles that must be supported in typical coalition operations. We then go on to define various modes of netcentric operation termed, wide-area, metropolitan-area and local-area netcentric warfare and show how they relate to these decision cycles. Examples of each mode of netcentric operation are presented. Continuing with modern network terminology, we consider the costs and benefits of introducing coalition information security at the physical and data link layer of a coalition network, at the network and transport layer, at the application layer and finally at the human layer. Based on this discussion we argue that the need for secure interoperability in a dynamic coalition environment favours the use of network and application layer security infrastructures over traditional link layer solutions that are commonly employed by military forces. We also consider the practical implications of such infrastructures at the various levels of netcentric warfare.

3. WARFARE DECISION CYCLES

In a discussion of netcentric warfare, it is important to focus on the warfare missions to be accomplished and the various decisions required to support those missions rather than on the enabling network technologies [3]. To this end, consider Figure 1 that illustrates the stages in military decision making overlaid on the traditional layering of the process into strategic, operational, tactical and engagement phases.

In each layer in Figure1, there are five stages: detection, localization, classification, situation assessment and resource allocation. They correspond to the basic questions: Is there anything out there? If there is, where is it and what is it? And finally, is it a threat and what should be done about it? The decision loops in this diagram are a modified version of the so-called “Observe, Orient, Decide and Act” or OODA loop that has been popularized in the military literature [3]. Here, the “observe and orient” portions of the loop have been expanded to include detection, localization and classification. We feel that this expansion of the cycle phases to include detection is necessary because of the fundamental importance of detection decisions in any military decision cycle. Moreover, the difficulty of the detection process when dealing with stealthy platforms in complex, cluttered environments is increasing.

The outer layer in Figure 1 illustrates the decision cycle for strategic decision-making. Typically, these decisions are based on information from a wide variety of data sources spread over a large geographic area fed back to a centralized strategic command decision center. Working inwards, the next layer shows the operational decision cycles that usually occur over a smaller geographic area such as a theatre of battle. At the operational level, force commanders must effectively share timely and accurate information in order to complete whatever mission has been assigned. The next layer inward is the tactical decision cycle that is often performed at the platform or brigade level in order to engage an adversary in either a hard-kill or soft-kill scenario. Finally, the inner-most cycle, referred to as fire control and weapons defence, is that which must be performed in order to accurately launch and guide a hard-kill weapon or effectively employ a

soft-kill measure in order to complete the defined mission. At each layer, each of the five stages of the decision cycle must be performed although the quality of decisions required at each stage will vary between the outer strategic layer and the innermost layer. Here we use the term "Quality of Decision" in a general way to refer to factors such as the spatial coverage, accuracy and timeliness of decisions. The impact of netcentric warfare on quality of decisions will be discussed in the next section.

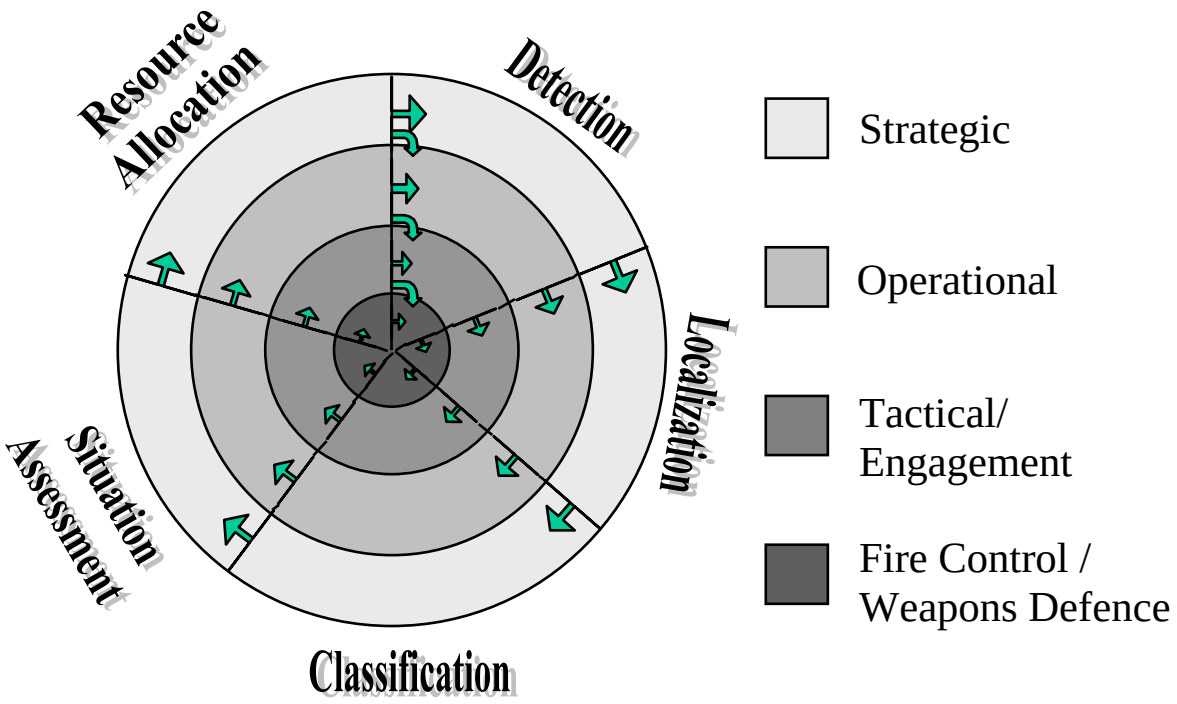


Figure1: Warfare Decision Cycles

4. NETCENTRIC WARFARE

The term network centric, or netcentric, warfare is commonly used in the military literature to connote the future of military operations where timely and ubiquitous sharing of information among operational forces will lead to dramatic improvements in mission effectiveness. The idea that information must be shared among participating forces to effectively complete a mission is certainly not new. However, the dramatic advances in networking technology over the past two decades are leading to a situation where planners will only be constrained by the question of what information should be shared among force members as opposed to what information can be shared.

There does not appear to be a well-accepted definition of network centric warfare so we will adopt the following definition for the purposes of this paper: *Netcentric warfare is the synergistic employment of sensors, decision makers and effectors (potentially weapons) within an information network to improve the capability of a force, or force element, to complete a defined mission*. This definition is quite generic but further refinements can be accomplished by specifying the area over which an information network is deployed. In analogy to terms used to describe the spatial extent of networks, we introduce the terms wide-area, metropolitan-area and local-area netcentric warfare in the following subsections.

In order to be more concrete, we also need to be more specific about the meaning of the phrase "improvements in capability" in the above definition. In this paper we limit our

consideration to improvements in the spatial area or temporal period over which mission decisions can be made and improvements in the timeliness and accuracy of decisions within the various decision cycles discussed in the previous section. The quality-of-service offered by the supporting networks will determine to what degree these capability improvements can be met. However, it is important to ensure that any improvements in mission capability offered by netcentric warfare do not come at the expense of an information advantage enjoyed by coalition partners over any potential adversary. Secure interoperability helps insure this information advantage so we consider information security as an important network Quality-of-Service (QoS) that must be provided in addition to the more traditional QoS factors. The challenges of providing networks with the appropriate QoS for the various levels of netcentric warfare will be discussed further in Section 5.

4.1 WIDE-AREA NETCENTRIC WARFARE

Information and network technology and procedures applied to enable information flow between entities and assets at, and immediately subordinate to, the strategic level provide a netcentric, wide-area capability to achieve multi-national and/or multi-service security objectives. Netcentric operation at this level is important to improve the strategic decision making process. It enables strategic battle space awareness leading to the generation of operational support requirements and intelligence. Information received at this level is used to constitute a recognized strategic picture for each theatre or operation. However the process of correlation, fusion and/or association of strategic sensor information and strategic intelligence preparation of the battle space data would be done at the theatre level which is discussed in the next section.

Figure 2 illustrates a national, wide-area netcentric scenario in which strategic decision centers, shown as Wide-Area Surveillance (WAS) sites, gather and assimilate data and information from widely dispersed strategic sensors. The goal of such systems is to provide continuous temporal coverage of all areas of concern to the sovereignty of a nation. Also shown in the diagram are mobile assets such as ships and aircraft that provide a capability to extend or complement the spatial coverage and accuracy of a strategic surveillance system. Alternatively, they are the platforms capable of carrying out the decision cycles in the inner three layers of Figure 1 based on tasking from the WAS sites. Figure 2 illustrates that the communication links in wide-area netcentric warfare may be either fixed or wireless but that the transmission distances involved are large in either case.

While the scenario shown in Figure 2 is a national one, information from coalition partners can play an important role through provision of alternate sensor information that may not be available at the national level. This is especially true when national forces are deployed to international locations. In this case, national decision makers must be aware of strategic situational information at the deployed location and will likely have to rely heavily on coalition sources. This information is generally highly processed and sanitized for coalition use because of the sensitivities associated with connecting national strategic systems. For example, it is highly unlikely that raw, real time data from strategic sensors would be made available to coalition partners. Rather, only particular track information would be supplied. It should also be noted that the time scales for the observation and decision-making cycles at this level will often be relatively slow. This is determined by the distances involved as well as the speeds of the sensing, decision making and effecting assets involved in a strategic mission. These considerations impact the security considerations that need to be given to coalition netcentric warfare at the wide-area level.

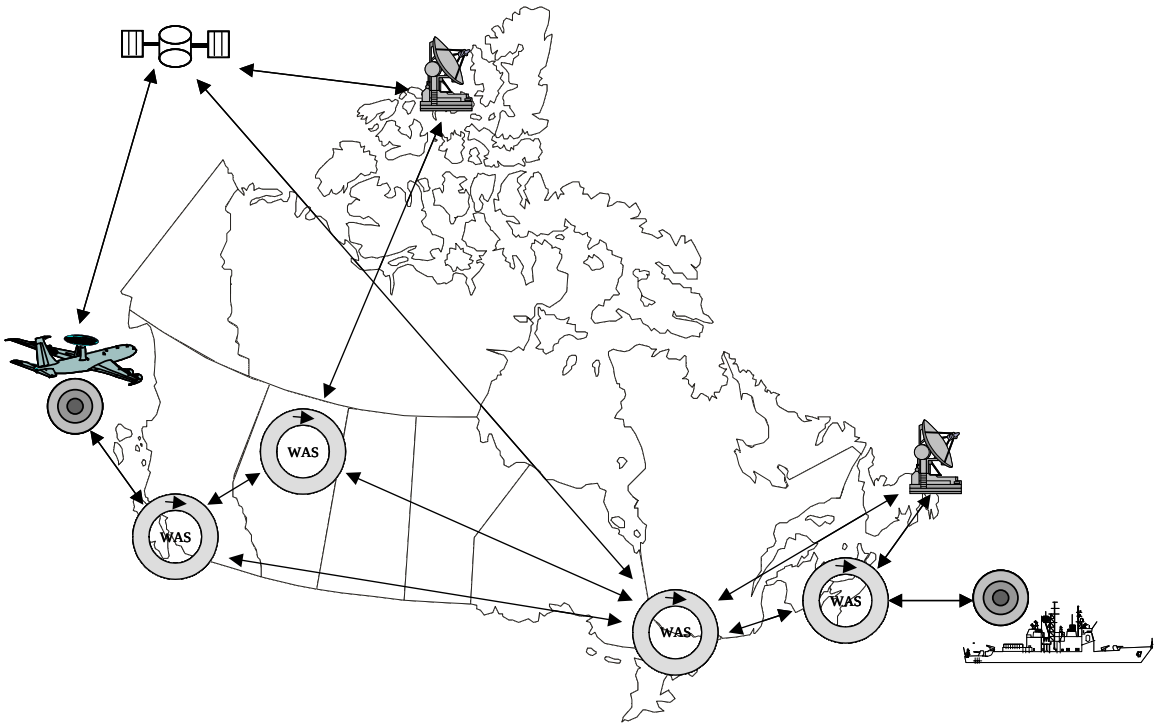


Figure 2: Wide-Area Netcentric Warfare

4.2 METROPOLITAN AREA NETCENTRIC WARFARE

Metropolitan area netcentric warfare concerns the information flow amongst friendly participants in a theatre or operational area. At this level, a major concern is the facilitation of joint operations with data, information and intelligence exchanges between air, land and maritime entities possibly from multiple nations. Coordination, consensus, economy/efficiency of resources, battle space coverage as well as accuracy and speed of command are objectives to be improved by netcentric warfare at this layer. A recognized operational picture for the theatre is produced at this level with the correlation, fusion and/or association of sensors information, order of battle (ORBAT) information and intelligence preparation of the battle space data. This picture is sufficiently detailed to assist with tasking decisions and following, to some degree, the task execution.

As shown in the diagram, the spatial area over which operations occur at this level is much smaller than in wide-area operations. With shorter distances, the necessary reaction times are shortened. This is because required speed of decisions is proportional to the speed of the assets relative to the size of the operational area. This implies that any supporting network must provide appropriate qualities of service to support the need for improved decision making speed. Figure 3 also illustrates the fact that the majority of assets involved in metropolitan area netcentric warfare will be mobile which implies that the supporting networks will have to be based on wireless communications links. It appears that it is this level that most often is referred to when people talk about netcentric warfare in the literature. The challenges of netcentric warfare at this level as well as at other levels will be discussed in the next section.

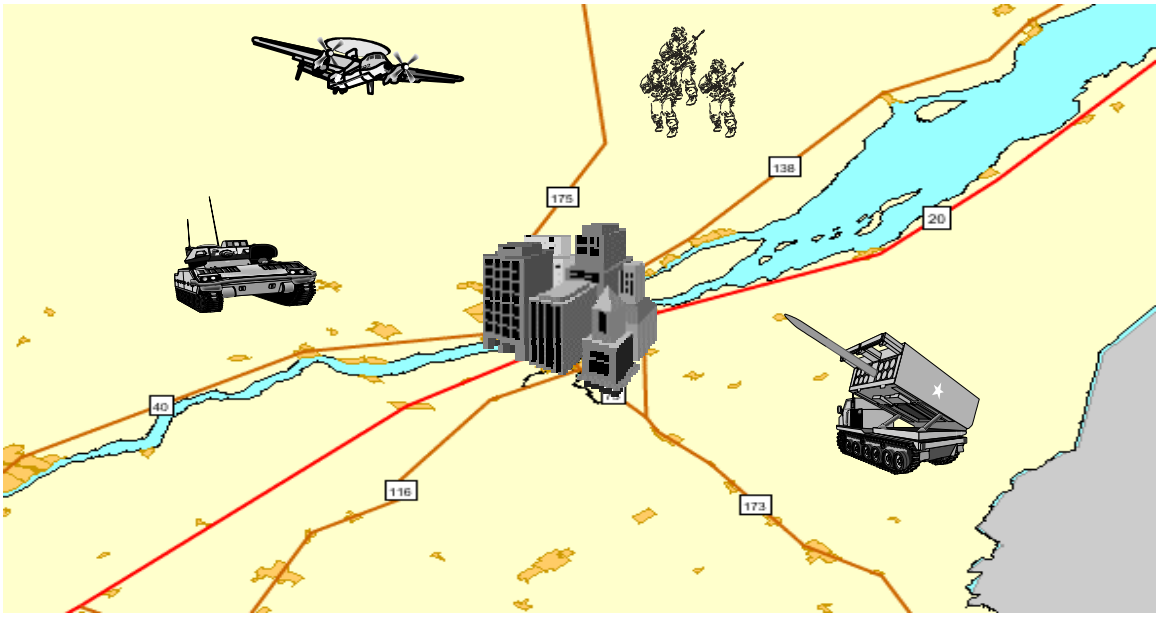


Figure 3: Metropolitan-Area Netcentric Warfare

4.3 LOCAL AREA NETCENTRIC WARFARE

We will use the term local-area netcentric warfare to refer to the networking of sensors, decision makers and effectors in an area that one would normally associate with a local-area network. As shown in Figure 4, an example would be the effective integration of sensors on board a ship or aircraft to facilitate detection decisions by sensor operators, fusion, localization and classification decisions by sensor supervisors and command decisions by tactical commanders. Alternatively, the effective networking of decision makers of various types within a coalition operations center, in Kosovo for example, would be another example of local-area netcentric warfare. In both of these examples, there is less need for mobility within the fairly small area of concern so wired communication links would certainly be more viable than in the metropolitan area case. However, at the local-area level there is often a need for high-speed connectivity because of the need to process large amounts of sensor data. The focus here is more on the synergistic and timely integration of sensors, information systems and weapons at the local-area level. Multiple nations may contribute sensors, information systems, or weapons that can be linked without restrictions on information exchange for exploitation at this level. These local area entities are concerned with force control decisions and aim to achieve greater precision and shortened response cycles by improving the accuracy and timeliness of information sharing, processing and utilization as well as the accuracy of the information shared.

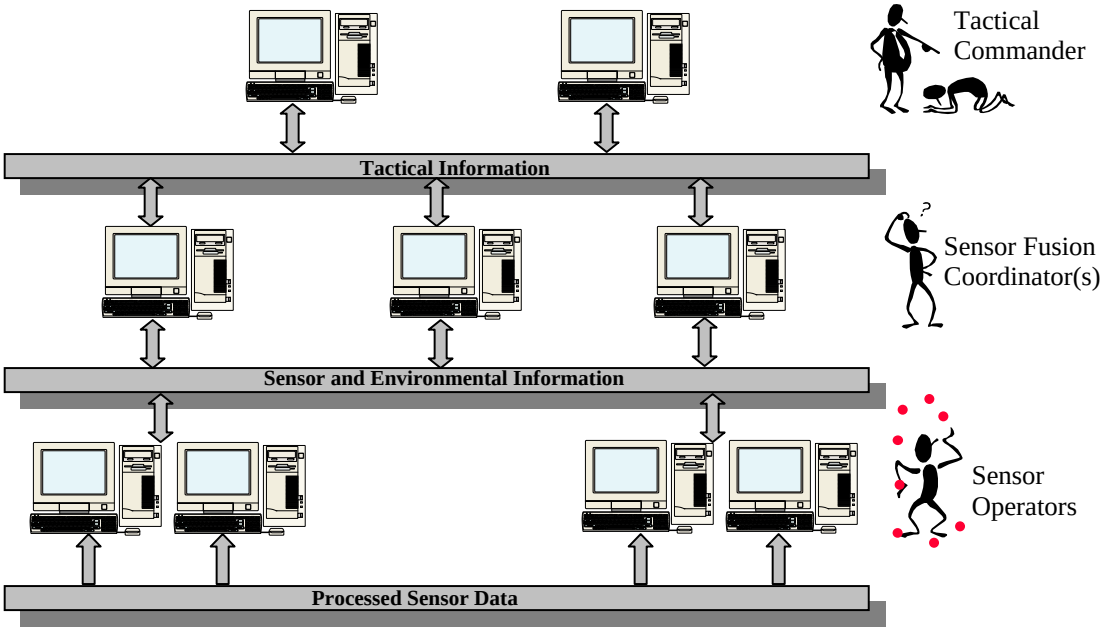


Figure 4: Local-Area Netcentric Warfare

5. CHALLENGES OF NETCENTRIC WARFARE

Recall that the aim of netcentric warfare is to improve the accuracy, timeliness as well as both the spatial and temporal coverage of mission decisions through synergistic employment of sensors, decision makers and effectors within an information network. Note that in common networking terminology, temporal coverage is usually referred to as availability. In order to realize these improvements an underlying network must provide certain qualities of service that vary depending on the mission needs.

First the network must provide the spatial/temporal coverage required by the mission. For a wide-area surveillance mission, the spatial coverage should be as wide as possible and the temporal coverage should be as continuous as possible. These coverage requirements decrease as one moves to the tactical surveillance mission down to the engagement and weapons control layers shown in Figure 1.

The second quality of service that networks supporting netcentric warfare must provide is information capacity (in bits/sec) between participating units. As well, there must be some measure of reliability associated with this capacity (usually equated to bit error rate). A third quality of service for networks that is an important consideration is latency, which is a measure of the delay between sending information and receiving it at the other end. Low latency is required for applications where real-time response is important.

The final quality of service we will consider, and the focus of this paper, is security. Information security is essential to maintain any information advantage one has over a rival. Security services for network information are usually broken down into authentication services, confidentiality services and integrity services. Authentication entails ensuring that entities involved in information transmission are the entities that they claim to be. Confidentiality services insure that information transmitted between decision makers across a network is not available to others who may not be authorized to see it. And finally, integrity services ensure that information is not tampered with in transit across a network. Integrity services are similar to

reliability services but reliability deals with modifications to the transmitted information due to natural errors while integrity deals with intentional modifications to data by some third party.

The provision of high-capacity, low-latency networks to support netcentric decision making is a challenge of varying complexity depending on the coverage of the network. This challenge is compounded as requirements for security services are added. In wide-area applications such as those discussed in Section 4.1, the decisions supported by the network are often strategic in nature. In strategic decision making the supporting data is usually highly-processed with a high information content (although there are some cases where relatively unprocessed sensor data may be required). This implies relatively low network capacity requirements. Also, latency of information transmission is often not critical. However, the high information content of the data carries fairly strong requirements for the security services, authentication, confidentiality and integrity. So, the QoS requirements for wide-area netcentric warfare are relatively loose for capacity and latency but quite stringent for security. It is expected that the number and affiliation of participants on the wide-area net do not change much over time and that they are mostly fixed in geography. These characteristics of wide-area netcentric operation allow for the exploitation of commercial infrastructure to provide some of the required capability for primary or redundant purposes.

Metropolitan-area netcentric warfare supports theatre and force-level decision making often in operational surveillance missions as shown in Figure 3. It is these types of scenarios that have been most often equated with the term netcentric warfare. A distinguishing feature of metropolitan-area netcentric warfare is the fact that the participating platforms are usually quite mobile within the fairly large network area of coverage. This implies that the network links must be wireless. Another feature is the fact that the participating platforms may change quite often as national contribution to the mission changes over time because of other commitments, inability to sustain, or escalation of conflict, etc. This is probably the most dynamic mode of netcentric operation and this most definitely influences the capability to provide networks of required capacity and reliability.

While much of the information that is shared during tactical surveillance missions is highly-sensitive in nature, its sensitivity is often short lived (platform position, course and speed information for example). This, taken together with the smaller area of coverage, compared with wide-area networks, implies that the requirements for security services are somewhat lower in this case than in the wide-area NetCentric Warfare (NCW) case. For example, the requirements for platform authentication may be less stringent in an operational scenario where the number of participating platform is relatively small and well known. However, because of the tactical nature of the mission, the capacity and latency requirements are usually more stringent than for strategic decision making. The limited capacity of wireless links for mobile platforms makes the networking requirements of netcentric warfare particularly difficult to meet.

In local-area, platform-level applications, the capacity and latency requirements for sensor data and processed information are usually quite stringent. This stems from the need to distribute and process large amounts of data and information in real-time to support offensive and defensive tactical decisions. However, the security requirements at the local-area level are often less demanding because it is easier to physically secure a small area. For example, the operations area of a military platform such as a tank, a plane or a ship or the operations room at a command headquarters can be physically secured thereby controlling the need for network security services such as authentication, confidentiality and integrity. As more emphasis is placed on coalition operations, this assumption may change since a wider variety of personnel may need to have access to operations area settings in coalition scenarios.

The table below summarizes the QoS requirements for the three types of netcentric warfare that have been discussed.

	Coverage	Capacity	Latency	Security
Wide-Area NCW	Large	Low-med	Med-high	High
Metro-Area NCW	Medium	High	Low	Med-High
Local-Area NCW	Small	High	Low	Med

Table 1: QoS requirements for Netcentric Warfare

6. IMPLICATIONS FOR SECURE COALITION INTEROPERABILITY

Secure interoperability in dynamic coalitions can be especially challenging since, by definition, the partners in such a coalition will not have a well established trust relationship upon which secure international information exchange usually rests. In this paper we will assume, though, that there is some established policy for exchanging information securely with dynamic coalition partners. A goal of this work is to develop guidelines for a reasonable security policy for netcentric operations among coalition partners where there will likely be strong asymmetry in secure networking capability. Following the OSI hierarchy, we discuss the implications of attempting to provide information security at the link level, the network level, the transport level, the application level and the human or content level. At each of these levels, we examine the issues associated with sharing information securely among partners in a dynamic coalition.

6.1 PHYSICAL AND LINK LAYER

Fundamental to any network are the point-to-point links that provide the capability to communicate digital information between the nodes of a network. Just as the capacity of these links limits the overall capacity of any network constructed from them, the security of the links limits the overall security of the network if link-level security is fundamental in a coalition security policy. This includes both physical security and cryptographic security. However, it would not be reasonable to rely on physical security of links in a dynamic coalition environment. For this reason we will not consider physical security alone in our discussions.

Link-level standards will be required among coalition partners just to provide basic communication capability. However, if link-level cryptographic security is also required, common standards will be needed for bulk encryption algorithms and for key exchange protocols. Moreover, in order to support the data rates required for most modern networking needs, hardware acceleration of cryptographic algorithms would be needed. Depending on the strength of the crypto material, this need to share high-performance encryption products for local-area, metropolitan-area or wide-area links among coalition partners varies between difficult and impossible from a policy point of view.

6.2 NETWORK AND TRANSPORT LAYER

Open network and transport layer standards for Transport Control Protocol/Internet Protocol (TCP/IP) networks offer the potential of using these networks as a backbone for information exchange among coalition partners. As well, open standards being developed by the Internet Engineering Task Force (IETF) for secure exchange of packets at the IP layer will likely

provide the foundation upon which secure coalition TCP/IP networks can be based. The growing world-wide acceptance of TCP/IP standards and availability of IPsec based virtual private network systems is making this an increasingly attractive option. This is especially true in wide-area applications where the Internet may be the only communication channel that is easily accessed by coalition partners.

If a high-capacity, high-availability secure channel is required to provide a secure link between different coalition local-area networks, both transport-mode VPNs and tunnel-mode VPNs will likely require hardware acceleration. This is similar to the link-level requirements discussed in the previous subsection and one would expect the key management issues to be similar. A big advantage of the network-layer option over the link-level option is that there are many more commercially available VPN products. Furthermore, the IPsec security standards are international and they are open for scrutiny by all. Taken together, these factors mean that sharing of VPN hardware among coalition members and the sharing of necessary security information to support the hardware is a viable option from a policy point of view. However, the degree of trust that should be attributed to these commercial platforms for nationally sensitive material is an important policy question.

An important software standard for secure interaction with a web server through a web browser is Secure Socket Layer (SSL). Because of its wide acceptance, SSL will likely be relied upon to provide transport-level security in web-based coalition applications. In a web browser running on most modern workstations, SSL software adds little noticeable overhead in normal web interaction without any need for hardware acceleration.

6.3 APPLICATION LAYER

There are a growing number of commercial software standards and products that are being designed to support the security needs of both individual users and groups of users who wish to communicate securely on public networks. These products are usually implemented at the application layer of the OSI hierarchy and offer the potential of providing some security services in coalition operations without the need to share hardware. Examples of such software standards and products include Secure Multipurpose Internet Mail Exchange (S/MIME) that provides standards for secure e-mail and security infrastructure software packages such as PGP, Entrust and Baltimore. These packages provide a number of security services bundled together with the infrastructure needed to support strong authentication as well as confidentiality and integrity of data files both in transit on a network and in storage on network hosts.

Application-layer security software offers the potential to meet some of the secure information sharing requirements of netcentric warfare. In particular they offer secure messaging capability, digital signatures, secure storage and transmission of a variety of documents and secure access to centralized servers from potentially remote clients. There are several limitations to current software packages though. First they are generally designed and developed to operate in either a high-performance LAN or WAN where communication bandwidth, processing power are information storage are in good supply. Second, the performance of these packages on wireless networks needed to connect mobile platforms in metropolitan area netcentric warfare is not well understood. And lastly, it is not clear whether it would be desirable and possible to augment these packages with encryption algorithms specifically tailored for coalition security applications in the case that the commercial algorithms were not acceptable under coalition security policy.

6.4 HUMAN LAYER (CONTENT LAYER)

The application layer is often considered the top of the OSI hierarchy but the real top of the stack is the human layer, especially when one is considering information security. It is at this layer where information content is generated and understood and where the requirement for security of that content arises. It is also at this level where security policy is generated and interpreted including guidelines for both physical and cryptographic protection of sensitive information.

In a coalition environment there will usually be discrepancies in security policy among the participating countries. There may also be several degrees of participation with some members not contributing as much as others or at the same level as others which may introduce several established levels of trust. In static coalitions where the members have developed a level of trust over an extended period of time, a coalition security policy can be established. However, in dynamic coalitions, it is unlikely that the coalition will last long enough for an appropriate level of trust to develop in order to allow development of a unique security policy. It is most likely that security policy standards, enabled through use of commercial hardware and software products will have to be relied on.

7. SUMMARY

The concept of netcentric warfare promises to link forces together to carry out missions in ways that were not previously possible through the application of modern information and network technology. However, the way and means to this end is quite complicated and is more than a technical solution; there are political and procedural considerations as well. This paper has suggested that an approach to break down the complexity, identify the requirements and work toward solutions for netcentric warfare within a dynamic coalition is to divide the structure/problem into several modes: wide-area, metropolitan-area and local-area netcentric warfare analogous to network terminology. These layers roughly correspond to the spatial coverage, information exchange and decision processes at the strategic, operational and tactical levels.

Supporting secure interoperability in netcentric operations that involve dynamic coalition partners can be a very challenging problem. In dynamic coalitions where the partners are unlikely to have a well-established trust relationship, national policy constraints will impede the sharing of government-developed information security hardware and/or software. This implies that secure interoperability for these scenarios will have to rest on commercially available or open-source information security hardware and software. Depending on the network qualities of service required for particular netcentric operations, the need to rely on current commercial security products may prove to be a limiting factor in determining the viability of netcentric operation. This is especially true in metropolitan area netcentric warfare where high platform mobility may preclude the use of commercial products.

Secure interoperability among coalition partners at the link layer will be difficult to accomplish because there are very few commercially available link layer products that are known to the authors. At the network layer, virtual private network technology will likely play a role in securely connecting national networks to allow exchange of sensitive information. At the application layer, security packages such as public key infrastructures and SSL or S/MIME will also play a role in providing security services in wide-area and local-area netcentric applications. At both these levels however, important security policy questions will have to be answered. Specifically, to what extent will national policy allow nationally sensitive data to be released on coalition networks that are protected using commercial products and services?

In this paper, the use of network analogies applied to netcentric warfare has allowed us to consider the problem from a new viewpoint. It has also allowed us to examine several qualities of service that supporting networks will be required to provide in netcentric operations. By focusing on the tradeoffs that must be made when security is an important QoS, we were able to draw some conclusions about security services for coalition netcentric operation. In particular, our approach uncovers and highlights a specific concern or requirement for commercial security solutions at the network and applications levels for mobile forces. It is hoped that the preliminary discussions included here will lead to more structured considerations of netcentric warfare and its implications for future military forces.

REFERENCES

[1] Cragg, Anthony, "The Combined Joint Task Force concept: a key component of the Alliance's adaptation", NATO Review Nr 4, July 1996, pp. 7-10, http://www.nato.int/docu/review/articles/9604-2.html

[2] "NATO Glossary of Terms and Definitions", AAP-6 (V) Modified version 02, 7 August 2000, http://www.nato.int/docu/stanag/aap006/aap6.html

[3] Smith, Dr. Edward A., "Netcentric Warfare: Where's the Beef", C4ISR Cooperative Research Program (CCRP), February 2000, Http://www.dodccrp.org/IS/eSmith/NCW_eSmith.htm

[4] Alberts, David, Dr. and John J. Garstka, "Information Superiority and Netcentric Warfare", CCRP Information Superiority / Command and Control Seminar, December 13, 1999. http://www.dodccrp.org/ncw.htm

[5] Berni, Alesandro and Lorenzo Monzone, "Wireless Tactical Networks in Support of Undersea Research", Proc. NATO/RTO Symposium New Information Processing Techniques, Istanbul, 9-11 Oct 2000

[6] Rice, J.A., "Telesonar Signaling and Seaweb Underwater Wireless Networks", Proc. NATO/RTO Symposium New Information Processing Techniques, Istanbul, 9-11 Oct 2000

Data-Translation: Leveraging Legacy Data for NATO

Martin R. Krick
NATO C3 Agency
Postbox 174
2501 CD The Hague
The Netherlands
krick@nc3a.nato.int/wilkes@nc3a.nato.int

Abstract

This paper describes an ongoing effort at NC3A to provide one integrated database which contains data from a number of different sources. Initially, these sources are legacy NATO systems. Later, other systems, including messaging interfaces of a wide variety, and national systems, will be added. A common data model is used as the lingua franca between systems. A COTS product has been identified that creates translator boxes to provide interfaces to and from the legacy systems.

1 INTRODUCTION

The NATO C3 Agency has responded to customer requirements with the Integrated Data Environment project, which has been evolving over the past two years. The intention of the effort is to provide one integrated database which contains data from a number of different sources; in the first place these will be legacy internal NATO systems. Later, other systems, including messaging interfaces of a wide variety, and national systems, will be added as requirements and political concurrence allow. It is foreseen that IDE will play a significant role in the Core Capability package for the Bi-SC AIS.

2 THE PROBLEM

Many of the data exchange problems that have confronted and bedevilled NATO for the past few decades have arisen from the fact that early systems were conceived, developed and implemented as stand-alone, or "stovepipe", systems by groups of users and technicians whose requirements horizon extended no further than the immediate needs of the system on which they were engaged. In the early days, interoperability of data models was not even considered relevant.

As time progressed, and the initial desirability of being able to pass information from one system to another became a more firm requirement, many mechanisms were devised to address these issues, but always with the caveat that the software within the in-service systems, seen to be of such acquisition cost as to be untouchable for interoperability needs, could not be modified to assist in the process of bringing systems together to provide for any meaningful direct exchange of data. In addition, because early systems were so expensive, and therefore made available only to the smallest possible community of users, and because many of the more senior users had no ADP facilities at all, or at most a simple teletype, these early mechanisms were specified to be able to be used in manual environments, leading to the definition of a range of messages. Once again, these message definitions were aimed at encapsulating the specific needs of the group of users responsible for the definition of each message; correlation between messages was not a driving force in the definitions.

Many studies were carried out when the nature of the problem became so large that it could no longer be ignored; these studies stressed the need for common standards for data definition, but could not provide low-cost solutions and their conclusions were therefore ignored. In essence, they proposed a "data fusion" approach, which is nowadays seen to be both impractical and unnecessary.

3 PREVIOUS STUDIES

Many studies were carried out when the nature of the problem became so large that it could no longer be ignored; these studies stressed the need for common standards for data definition, but could not provide low-cost solutions and their conclusions were therefore ignored. In essence, they proposed a "data fusion" approach, which is nowadays seen to be both impractical and unnecessary.

4 THE DATA FUSION APPROACH

The principle behind a data fusion approach is to define a single data model, and implement a single database, which will encompass the entire set of data currently held in all existing systems. This approach has some advantages, but also has many more major drawbacks which make it an impractical proposition. If we take two or three existing systems, and create new database which holds all the data previously held in the three individual databases in accordance with a new all-encompassing data model, then the new database will not be the same as any of the old ones. Each application suite in the original systems must therefore be re-written.

It might be possible to create a database interface package for each system to make the new database appear as the old database, but that too would be substantial effort (and there would be no *ab initio* guarantee of feasibility) and would represent an

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

additional load for the original system which it might well be ill-suited to handle.

A further major, and potential even more serious, disadvantage is that if another legacy system were to be added to the fusion set, it may impose changes on the data model which would have a knock-on effect on all current systems within the fusion set. This would lead to potentially exponentially soaring costs, and to management problems of equally soaring complexity. Little wonder that the NATO committees of the time were not persuaded to follow down this route!

The perceived advantages and disadvantages of the data fusion approach can be summarized as:

- single view of all data
- single physical database from which all applications can draw data

whereas the disadvantages are:

- need to agree the (large) data model between 19 nations and all NATO HQs and Agencies
- immediate impact on all legacy systems which are required to conform to the new global data model
- fine for a small number of systems (three or maybe four)
- ongoing management overhead for the fusion schema
- complexity increases dramatically with the number of systems
- process becomes unmanageable with large numbers of systems

It should be noted that the advantages are not matched by any known requirement for all data to be perceived in a single view, nor that there should be a capability of providing a single database implementation which would hold all data; these advantages represent theoretical technical possibilities only. By contrast, the listed disadvantages are very real, not least the political problems associated with the first of those listed. Corresponding agreements in related areas are not famous for the speed with which such agreements have been reached nor for the technical clarity of the final agreements.

5 OTHER MORE RECENT STUDIES

In the last ten years, other initiatives have been taking place on a lower profile basis, and the fruits of their endeavours are now beginning to become visible in a number of places; national implementations based on these initiatives have been put in place and have become sufficiently mature for reasonable projections to be made. Principal of these initiatives is the multi-nation ATCCIS[1] study, sponsored and led by NATO, with active participation at varying levels by eleven nations. The major outputs of the ATCCIS study to date have been:

- a wealth of well-documented analysis
- a fully specified data model for information exchange
- an ATCCIS Replication Mechanism (ARM) for selective transfers of data between two or more ATCCIS-conformant databases

The primary achievement of the data modellers is that they recognised that they were endeavouring to specify a data model to facilitate the exchange of information rather than for the design or development of systems; thus the level of detail of the model is appropriate to information exchange, and much low-level data, which would typically be found only in specialist systems, was not included. This separation of "local" data and "global" data has been one of the foundation links of the NC3A work on the Integrated Data Environment.

6 THE INTEGRATION APPROACH

The separation of local and global data leads immediately to the concept of an IDE which addresses only some of the totality of data held in all existing (and future) systems. It also leads directly to the recognition that the IDE can be established (either as a virtual database or as a real one) for new purposes, and that the existing systems can be left with their current databases and database management systems – be they rudimentary or advanced – with the immediate benefit that no changes to those systems are required. Indeed, it became one of the design objectives of the IDE work that the IDE concept should be seen to be non-intrusive from the perspective of any legacy system.

In the integration approach, data are translated from the native (legacy) environment to the common data model of the IDE, so that the translated data subsets reside in a single database or transmission mechanism with one common data model describing all data. We may think of this common data model as a "lingua franca".The integration approach offers as advantages:

- single view of all global data
- no impact on legacy systems
- no requirement to have a single database
- all future applications can draw global data from existing databases
- process remains manageable with large numbers of systems
- ongoing management overhead for the integrated database is much smaller than for the fusion approach
- technology is mature and in use in large commercial organizations

and as disadvantage:

- as of mid 2001, the technology has not been proven within a NATO operational system (but a demonstrator has been produced, and is clearly scalable to full operational use).

It may be seen that almost all of the disadvantages of the fusion approach have been stood on their heads for the IDE approach. The single view

[1] The common ATCCIS Generic Hub 4 data model was forwarded to NATO in 1999. NATO initiated a standardisation process for this data model, now called the Land C2 Information Exchange Data Model (LC2IEDM). The respective STANAG 5532 (ADatP-32) has been submitted as draft and is expected to be agreed in 2001

of all data, which was never supported as an operational requirement, has been scaled down to become a single view of all global data, for which operational requirements most certainly exist. The previous high impact, in terms of both cost and operational implications, of the fusion approach, has become a zero impact on those systems. And, the management problems remain tractable.

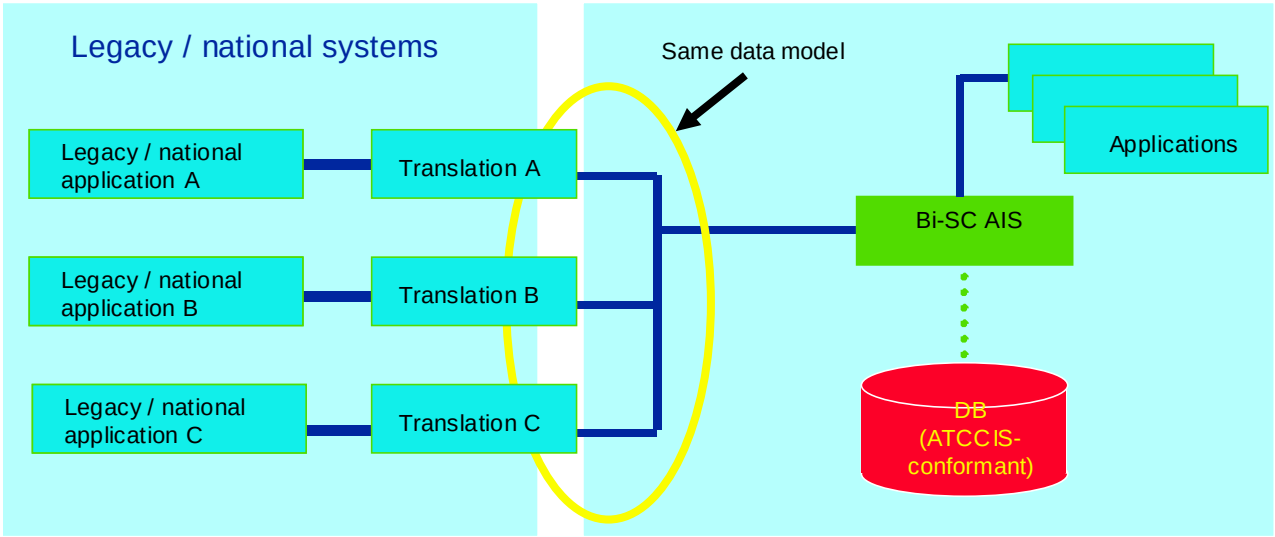
On the disadvantage side, the technology has not yet been tested in a full NATO operational environment, but a four-system demonstrator has been produced, and the technology is scalable to encompass a very large number of systems. In particular, the technology ensures that the management problems remain at the one-system level, and therefore do not grow as the number of systems being integrated expands.

7 ALTERNATIVE TECHNIQUES

There are two techniques available to implement the IDE function, Data Mediation and Data Translation. Data Mediation works by first making associations of the meta-data of the data sources and the data sink, and then automatically converting source data to the sink on the basis of these pre-determined associations. In principle, this is a very powerful technique; however, at the present time the technology is still in the research stage, with academic institutions producing small-scale demonstrations. No proposals for a full-scale demonstration have come to our notice at this time. The technology is thus considered to be far too immature to be considered for introduction to NATO at the present.

By contrast, Data Translation is a very much more mature technology which has been in use in commerce for some time. Most of those applications have been for data warehousing applications, but some applications have been for genuine data integration applications. Where the translation process is carried out on a one-translator-per-system basis, there are very few problems about scaling to multiple systems. The scaling problems are mainly associated with the suitability of the sink data model for the spread of data types to be found in the source systems; in this respect, the highly generic nature of the ATCCIS data model is of immense benefit in minimising such risks. Finally, it must be emphasised that both techniques act on the conversion of data on a one-for-one basis. Data aggregation, data fusion and other application-level functions are outside the scope of both technologies.

8 THE IDE ARCHITECTURE



This diagram gives a very simplified overview of the IDE architecture resulting in the use of translation techniques on a Translator-per-System basis. Data from each legacy or national system is processed by its own local translation process to the target (sink) data model and added to the data model of the target system by normal database update techniques. The translation mechanism is a process, implemented as a software package; although for simplicity it is shown in this slide as though it were a separate system, it could equally well be hosted on the legacy system if that were to prove to be the preferred option. However, to emphasise the “No Impact” concept, we always show it as a separate system.

Because the translator process will only translate data about which it has been provided with appropriate translation data (which is another form of meta-data), it acts as a simple form of guard against the accidental translation of data which is not to be released. However, the translator process makes no claims to be an approved guard, and additional security devices would normally be expected to be fitted by national authorities to protect national systems which may contain nationally-sensitive data. These would typically be positioned between the national system and the translator.

Both the initial configuration of the translator, and any subsequent upgrades or changes to a national

system will require detailed analysis of the source system in order to specify the translation meta-data. For this reason, the configuration of the translators is expected always to be done by the nation concerned. The diagram thus shows the translators residing in the national management domain, with the exception of the specification of the output format (ATCCIS conformant) which is essentially public domain.

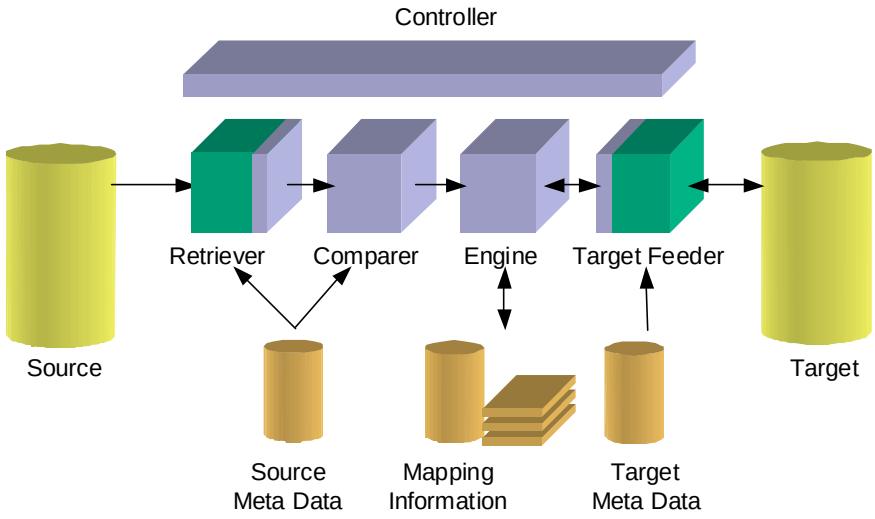
9 WORK DONE BY NC3A

The preliminary study on data mediation carried out in 1998 showed that the technique held potential for complex translation situations, and for the tracking of changes to databases. A simple demonstration system was created, using the most rudimentary meta-data, which was shown at JWID-99. Much interest was demonstrated by visitors at the ability to show data from three different systems out of a common database in response to a single query, with the consequential ability to provide for integrated data solutions.

Evaluation of a contractor report made clear that, although the concepts behind the Data Mediation technology were both powerful and useful, the technology was very immature with no commercially available implementations of a data mediator product, and little prospect of any such products appearing in the market for some considerable time. Data mediation may have benefits for special situations in the future, yet to be assessed and proven.

At the same time, an investigation was made of other products, all of which proved to be Data Translator systems, and it was determined that this offered a better approach for the near term. A contract was let for the development of a demonstrator using translation technology for display at JWID 2000. Problems with the suitability of the translation proposed by the contractor meant that only a very limited demonstration could be mounted at that time, but a very good tool has since been developed by the contractor as a COTS product, which has proven to be very successful and very flexible. A demonstration held at NC3A in late November 2000 showed the capabilities of this tool, and the design gives confidence for its use in many other situations, including message-oriented environments. A major demonstration is planned for JWID-01 at SHAPE.

The diagram below shows the architecture of the translator box produced by the tool.



10 THE SELECTED DATA MODEL FOR IDE

The selection of the ATCCIS data model, in the form known as the SHAPE Land Command and Control Information Exchange Data model (LC2IE DM) proved to be a sound choice. The complex nature of this data model means that the specifications of the translations are themselves more complex, but no instances were found in the work on the four NATO legacy systems where translations could not be specified with alacrity and accuracy.

The NATO NDAG Reference model is also based on the ATCCIS model, and is under strict Configuration Management; the LC2IE DM should similarly be placed under CM while it is being used as an interim measure before the full availability of the NATO RDM. At the same time, some of the work of the NDAG could usefully be retro-fitted to the LC2IE DM to make it into a Joint product, a JC2IE DM; the experience of NC3A and their contractor suggests that the minimal changes for the interim product would be small and easy to define and implement. For the November 2000 demonstration mentioned above it was necessary to add only four low-level entities (Naval unit, Air unit, Naval facility and Air facility) and to extend the range of a set of domain values to cover maritime and air factors. The total work took less than a couple of days; to repeat this work under full CM control would take less than one week.

11 THE TOOLS USED FOR IDE DEVELOPMENT

Mention has already been made of the shortcomings of the original analysis tools proposed by the contractor. These tools were designed for data warehousing applications where the primary focus of the tools was to analyse data – often dirty data – for which a data model did not exist. In the IDE situation, data models existed and were well documented (although there were some instances where the semantics of the data were not fully defined). Additionally, in data warehousing applications, the emphasis on fitting all source data into a single data model in the destination system does not apply. It is thus not surprising, with the benefit of hindsight, that the tools were found to be unsatisfactory for the IDE situation.

The analytical process involved in determining the translations required is both a skilled process and one which requires time. An analyst familiar with both the source system and the destination data model can complete several source tables each day if the source data model is “clean” and the semantics are fully defined and supported by exemplar data samples. Loose source data models, or a lack of semantic definition, or a lack of sample data, will slow the process to a considerable extent. The tool developed by the contractor provides considerable assistance in converting the results of the analysis into translation rules; future versions are expected to provide some additional assistance to the analysis itself, but cannot fully replace the need for analysis or the analyst.

12 SUMMARY

NATO and the nations still have a plethora of incompatible data systems which are likely to remain in service for many years. A fusion approach is not appropriate, and is likely to prove unmanageable and unaffordable.

The Integrated Data Environment provides a response to this information management challenge that is both manageable and affordable, and is eminently suitable for an incremental growth approach.

Commercial off-the-shelf tools are available which support IDE and thus support Coalition interoperability, NATO to NATO interoperability, NATO to nations interoperability, and Coalition HQ to nations interoperability.

Martin Krick, who graduated from Imperial College, London, with a First Class honours degree, is a Principal Scientist in the Information Systems Division at the NATO C3 Agency. He has been specialising in the interoperability of C2 systems, and the associated problems of the definition of data, since 1979, initially at the UK MOD and later during two spells at NC3A.

From 1984 to 1994 he was a member of the ATCCIS Study Group, and chairman of its Technical sub group.

More recently he has been leading investigations into ways of implementing mechanisms for providing for interoperability between non-compatible systems which have led to the development of the IDE concept and the creation of tools to support the concept'.

This page has been deliberately left blank

Page intentionnellement blanche

Providing the Common View of the Situation – The WASP Approach

Niclas Bergman
SaabTech Systems AB
Nettovägen 6
SE-175 88 Järfälla
Sweden

Klas Wallenius
Royal Institute of Technology
SE-100 44 Stockholm
Sweden

1. INTRODUCTION

New commercial network technology brings new cost-effective approaches to information distribution and information sharing. By sharing information, not only does planning and engagement become more efficient, but the information resources themselves also become better utilized. Information sharing allows for wider flexibility in the organizational structure, it decentralizes the decision making, giving faster reactions to new scenarios and unpredicted situations. Moreover, the strive towards a higher interoperability between coalition partners with more frequent and greater joint peace keeping activities also demands a higher level of information sharing.

This paper presents the techniques and methodologies for information sharing that has been developed and evaluated within the WASP-project (Wide Area Situation Picture). The WASP system provides a solution based on a distributed and non-hierarchical infrastructure utilising commercial network technology. The encapsulated software labelled the WASP Correlator Unit (WCU) is designed for easy integration into present systems and for co-operation with standardised NATO links.

The paper gives a brief technical overview of the WASP concept and highlights the system performance and functionality by means of results from simulations. The system features include automatic distributed track correlation with optional manual interaction, accurate estimation of data quality, and bandwidth control. The system also provides functionality for selective subscription to information with respect to its geographical origin and data accuracy. Data selection is obtained by use of multicast services in the network, giving each subscriber to the network the part of the global information that is of perimere importance from his horizon. Simultaneously, each subscriber feeds the network with all information that will contribute to the global picture concerning both coverage and quality.

The system provides global target numbering distributed to all subscribers. The quality of the common situation picture available in the network is naturally affected by the tracking performance at each subscriber feeding the network, but also by the network delay and target manoeuvrability. The data quality is therefore constantly estimated and monitored by each WCU. The system is fully scalable regarding the number of connected subscribers and due to the generality of the encapsulated WCU, the subscribers can be of fundamentally different types.

2. BACKGROUND

A Revolution in Military Affairs, RMA, is the label often used to indicate the vast changes that are foreseen in modern armed forces around the world [3]. The revolution will enforce completely new doctrines and organisations for warfare. New ways of running business in the commercial sector, exploiting the fast development of information technology, will influence this revolution. The concepts of warfare leveraged by the new technology are sometimes described as Network Centric Warfare [1]. The network centric view, as opposed to the traditional platform centric view, will imply that information obtained somewhere in the organisation can be shared by anyone else that is connected to the network, assuming this individual is authorized the handle the information. The Network Centric view enables decentralised decision making and a reduced number of levels in the management hierarchy. This will, in turn, give much faster reactions to

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

unexpected events in the battle-space. Another advantage with the network centric viewpoint is the oppurtunity for easier integration between allied partners who may operate in coalition using an extended network baring the common situation picture. The Swedish Armed Forces has extensive intentions to develop an RMA-like concept [3]. One of the efforts made in this direction is the research on a *Mobile Joint Command and Control Function*, with the Swedish acronym ROLF (Rörlig Operativ LedningsFunktion) [5].

A basic necessity for the mobile C2 function is the ability have a good situation awareness, described in detail below. A research programme to find methods and concepts for a Wide Area Situation Picture, WASP, is performed in co-operation between SaabTech Systems AB and FMV. The WASP programme has developed a prototype system for demonstrating how to implement a network centric functionality where a common situation picture is maintained and distributed within a network structure. Current research aims at evaluating the principles by integrating this prototype system with a state-of-the-art air surveillance C2 center.

The views presented in this article are mainly based on experiences from the WASP programme. However, there are numerous other activities in the RMA arena where SaabTech Systems AB takes part [6].

3. Situational Awareness and the Battle-Space Model

The concept of good *Situational Awareness* usually requires that the following aspects are known by the decision-maker: (1) the states of own and enemy forces, (2) the environment, and (3) the relationship between the forces and the environment [2].

Officers in the Swedish Armed Forces use a well-proven decision scheme, in order not to forget any important aspects before making a decision [4]. The decision scheme requires that the purpose and goals of the task and knowledge of own and enemy forces are considered to maintain the decision alternatives. The knowledge of the forces includes a comparison of the capacity under present terrain and weather conditions. We can see that these aspects map well on the definition of situational awareness. By taking these aspects into account, operators will make their decisions based on a well-founded awareness of how to optimally exploit their own resources.

By ‘awareness’ we mean the *mental* model of how the Battle-Space is characterised. This mental model is typically achieved by the operators’ interaction with the command and control system that hosts the *technical* model of the Battle-Space. The technical model, on the other hand, is the representation of information and data from sensors, intelligence forces, human reporters, and other information sources, arranged in a manner that is comprehensible for the decision-makers. The process to assemble the technical model from the data and information is known as *Data Fusion*. The Data Fusion process can be performed manually or (more or less) automatically.

It is important that the technical model is able to support all the important aspects of situational awareness.

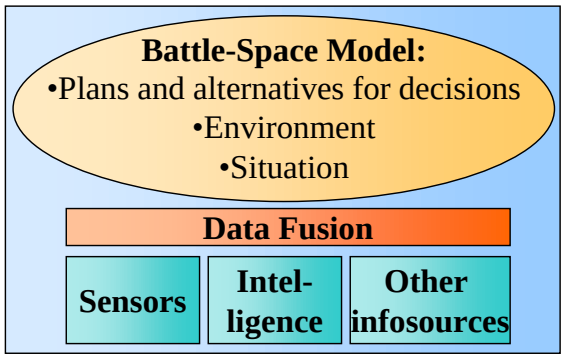


Figure 1. ***A Battle-Space model is fused from data and information collected by sensors, intelligence forces and other sources.***

Referring to the decision scheme above, we see three major categories of information needed in a complete *Battle-Space Model* (see Figure 1):

Situation. Estimated states of objects in the battle-space. The states include identity, type, kinematics, and logistic status of the objects.

Environment. Information on phenomena that cannot be affected by own or enemy actions, e.g. weather, terrain, and doctrines.

Decisions. Purpose and goals for the missions, together with both rejected and selected decision alternatives ('plans').

4. A Network Centric Info-Structure – the Common Battle-Space Model

Traditionally, 'systems of systems' for military organisations have been *platform centric*. The communication bandwidth between the units (command and control centres, aircraft, tanks, etc.) has been very limited compared to the communication within each unit. The information flows between different units have been explicitly defined and thus very inflexible. The information exchange in such a platform centric info-structure is naturally message based, i.e. certain message formats are predefined for different kinds of reports and orders. Different units have historically used their own sensors and data processing capabilities to maintain their own local Battle-Space Models. This as a result of the difficulty they have had sharing information.

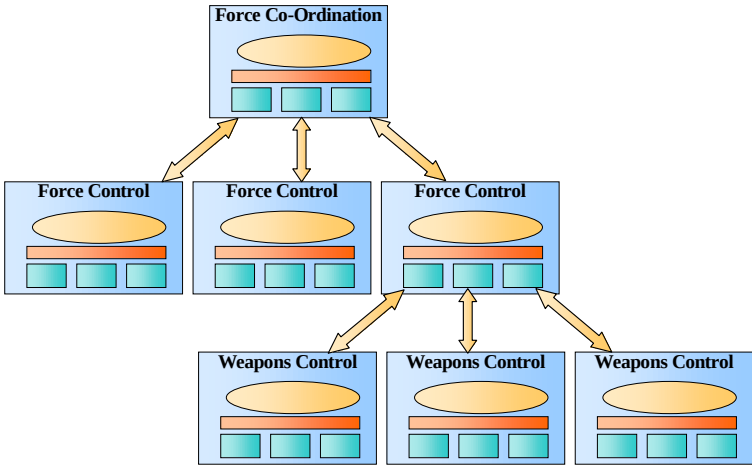


Figure 2. ***In a traditional platform centric info-structure, each C2-unit keeps it's own model of the battle-space, assembled by use of own data and information sources.***

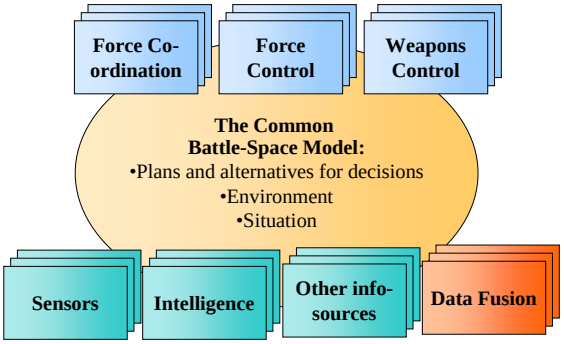


Figure 3. ***In the proposed network centric info-structure, the Common Battle-Space Model is shared among all the decision-makers. This also implies that the information resources are shared between the decision-makers.***

There has been little or no means to keep these models consistent between the platforms (see Figure 2).

By the development of modern network technology, communication bandwidth has increased significantly. At the same time, standardised communication protocols have made it considerably cheaper to offer services across all units connected to a network. Thus, the view can be changed from defining information flows between the platforms to instead defining the common information model available to all units connected to the network. By standardiszing this common information information can be shared also between coalition partners working in a coalition structure. We strongly advocate that a system of systems for Command and Control should be built around a *common* Battle-Space Model. This model should include all aspects of situational awareness for all users. The model should include the pieces of information relevant to all decision-makers on all different levels in the military organisation (see Figure 3). The traditional platform centric view will then be replaced by a *network centric* view, where the information model is important rather than the geographical location.

The implementation of the Common Battle-Space Model would imply a tremendous improvement of decision performance. Information resources can be shared by all the platforms, giving views for each decision-maker not restricted to own sensor coverage. Better and faster situational awareness together with a larger flexibility would lead to a much better utilisation of available resources.

5. The MST and the WASP Concepts – a General Approach to Utilise Different Data Sources

SaabTech Systems has developed the methods needed to meet the requirements for a common description of the situation, regarding moving objects in the battle-space. WASP – the Wide Area Situation Picture is truly network centric, and is designed to

Give information on moving objects for different users, with common target numbers.

Be robust to attacks, jamming and technical disturbances,

Fully utilise currently available information resources: sensors, C2 centres, data links, and data fusion capabilities (all from many different suppliers).

Always indicate the resulting data precision due to the currently available information resources.

Be fully scalable, allowing for thousands of both users and contributors.

Be easy to integrate, even with existing systems and data links.

The achievement of these goals within WASP is performed on two levels: on the *sensor data level*, and on the *track data level.*

On the sensor data level, active and passive sensors may be of fundamentally different kind, measuring from one to three or even higher dimensions. A measurement could be a one-dimensional bearing to the target or a complex data record including bearing, range, elevation and Doppler information. A measurement may also give indications that can be used to estimate the object's type or identity.

Data fusion is performed to combine these sensor data to estimate what objects there are, and where they are in the battle-space. SaabTech Systems AB has the solution for this – the MST, with the future add-on of automatic type identification (MST+). The MST is capable of using many different sensors from any supplier as long as there is output on the sensor data level.

Theoretically, data fusion is best performed when data from as many sensors as possible are being utilised in one tracker – *centralised* sensor data fusion. In some cases it is actually necessary to feed data from more than one sensor to the tracker, for instance when only passive sensors are used. Thus it would seem appropriate to have only one data fusion node in the network, using data distributed from all available sensors.

There are, however, several obvious reasons why data fusion on the sensor data level will be performed at more than one place and why there has to be means to combine data also on the track level. First of all, the usage of one single MST in the entire system would imply a very vulnerable solution. Redundancy will be needed to meet the requirements for robustness. Secondly, some sensors only give track data because of the construction of the sensor (with *local* sensor data fusion) or because of the available bandwidth. Thirdly, the connection of centres and systems belonging to other organisations will give tracks in most cases.

By the combination of MST and the WASP, data fusion can be performed on both the sensor data level and the track data level. Together they make an excellent approach to solve the trade-off problems regarding local or centralised data fusion. The combination of the concepts also meets the requirements for generality regarding connected data sources from different vendors, see Figure 4.

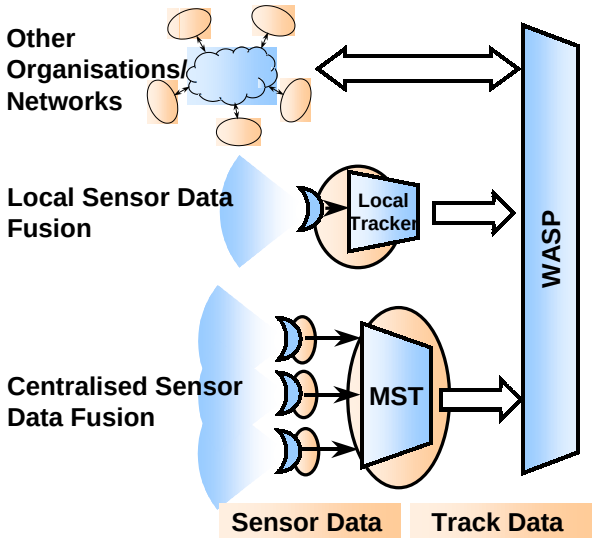


Figure 4. ***Data from both the sensor level and the track level can be utilised by the combination of MST and WASP.***

6. The WASP CORRELATION UNIT (WCU)

The *WASP Correlator Unit* (WCU), is used to perform the track correlation required in the WASP concept. The participating track sources (C2 units, data fusion nodes and/or networks belonging to other organisations) are connected to the network, and to each other, via the WCUs. All software necessary to establish the WASP is encapsulated in these WCUs. Thus, the adaptation effort for each type of track source is kept at a minimum, see Figure 5.

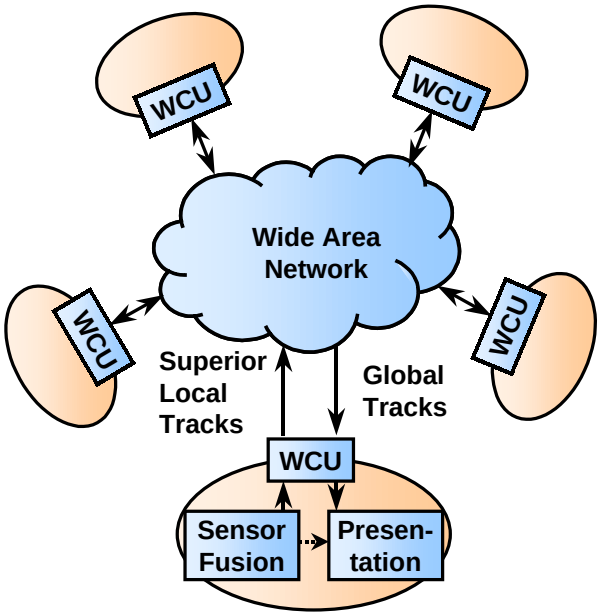


Figure 5. ***The WASP Network. Participating track sources are connected to each other via the WCUs.***

The local track data is correlated to the global track data, received from other WCUs (via the network), to determine the tracks that correspond to the same real objects. This correlation process is fully automatic, although it will be possible to perform manual interaction.

Special care is taken in the WCUs to estimate the precision of the tracks. Local data that is either unique or of better precision than the global data, is reported to the other WCUs. In this way the network will distribute only the best track data that is available for each object. Finally, the actual Wide Area Situation Picture is assembled for presentation to the local operators.

To reduce bandwidth consumption, the total surveillance area is divided into smaller subscription areas. For each such area there are several bandwidth levels that could be subscribed to by using multicast services in the network. On the lowest bandwidth level, information on all objects in the subscription area is reported, although at a rather low update rate. Higher accuracy is achieved by subscribing to higher levels, thus requiring more bandwidth in the network.

The WASP concept will be fully scalable, due to the selected correlation algorithm and the use of multicast. Thousands of C2 units can be connected and there will be no limit on the total surveillance area. The survivability of the WASP will be outstanding. Although sensors, data links and C2 units may come and go, each operator will still achieve the best quality and consistency from the WASP.

7. The WASP Prototype System

A WASP prototype system has been developed at SaabTech Systems to demonstrate the concept in an air surveillance application. In the prototype system a number of computer nodes are connected to each other via an Ethernet LAN. The computer nodes are hosting several C2 units. Each C2 unit consists of one MST, one WCU, and one display unit. There is also a combined air traffic and radar simulator, and a network simulator, see Figure 6.

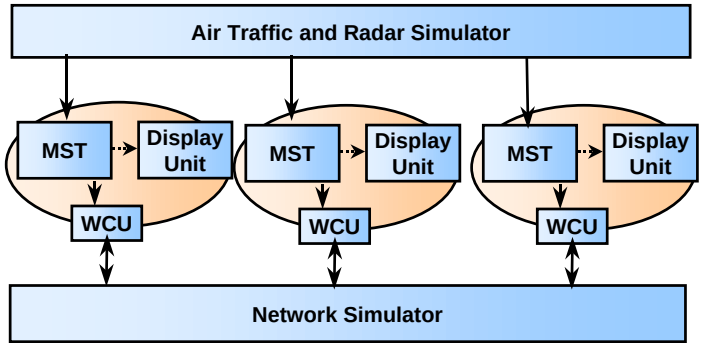


Figure 6. ***The WASP Prototype System***

A large number of radars, observing the same air traffic scenario, can be simulated. Radar plots are thus produced which are used by the MSTs to assemble situation pictures that are local to each C2 unit. The local situation pictures are forwarded to the WCUs which, in turn, communicate via the simulated network to establish the WASP.

The resulting situation picture presented in each C2 unit is compared to the “truth” in the air traffic simulator to estimate the target accuracy. The estimated accuracy takes into account the performance of available radars, MSTs and data links. In addition to accuracy estimation, the system provides evaluation of the consistency between the WASP inforamtin in the C2 units on the basis of the available track numbers. The performance of the WCU software can thus be evaluated under severe conditions.

Different sets of data links, sensors, trackers, and C2 units can be evaluated with the prototype system. Used in such manner, the WASP Prototype System serves as an excellent tool for simulation based acquisition (SBA).

8. The Example

As an example, the prototype system has been set up using three C2 Units situated in the southern part of Sweden:

Unit 1 is a surveillance centre, using two surveillance radars with range, bearing and elevation measurement capabilities. The radar stations are simulated with typical performance parameters. The rotation time is 10 seconds, and the radars cover a range of 300 km each. Unit 1 is connected to the network with a high bandwidth.

Unit 2 is also a surveillance centre, using two radars situated 300 km east of the radars used by Unit 1. These radars only measure range and bearing, meaning that Unit 2 is lacking information on the target altitudes. Unit 2 is also connected to the network with a high bandwidth.

Figure 7. ***Screen-shots taken from the prototype system at a time when there is no communication between the three C2 units.***

Unit 3 is a Surface-To-Air Missile Unit. It uses a surveillance radar that covers a much shorter range, 75 km. The radar has, on the other hand, a much faster rotation time – 1 second. This radar is also lacking elevation measurements, which means that the center has no altitude information in its local situation picture. The network connection for unit 3 is very poor, the bandwidth is limited to only 1200 bits/s.

Figure 7 shows screen shots of the situation pictures displayed in the different units at a moment when the simulated network is completely jammed out. Since there is no communication, only targets within the range of the unit's own radars can be seen on the screens. The radar coverage is partially overlapping. Some targets can thus be observed in more than one unit. The target numbers are indicated as three digit numbers followed by the letter "I" on the label of each target, the capital I stands for internally generated target number. Comparing the situation pictures it is obvious that the target numbers are inconsistent between the units due to the lack of communication between them. Furthermore, the altitude information can only be seen in Unit 1 (indicated by the second line of the target labels).

The colour of the targets indicates the estimated quality, in terms of resulting accuracy due to used sensors and limitations of the data links. White targets are of good quality (the resulting accuracy is better than 500 m) while grey targets are of bad quality (the resulting accuracy is worse than 500 m). Most targets in Figure 7 are of good quality. A few targets of bad quality can be found far from the sensors, though.

Figure 8 shows the same example but at a later time, when the jamming of the network has ceased. Interaction between the WCUs in the different C2 units is now enabled and the following effects may be noticed:

Targets that are observed in any C2 unit are displayed also in the other units.

The targets have the same numbers in all the different C2 units – the WCUs have interacted to agree upon a common target numbering.

Altitude information is now displayed in all units for targets that are observed by radars in Unit 1.

Targets in Unit 3 that are not observed by the local radar are displayed at a rather low accuracy (indicated by the grey colour). This is due to the limited bandwidth.

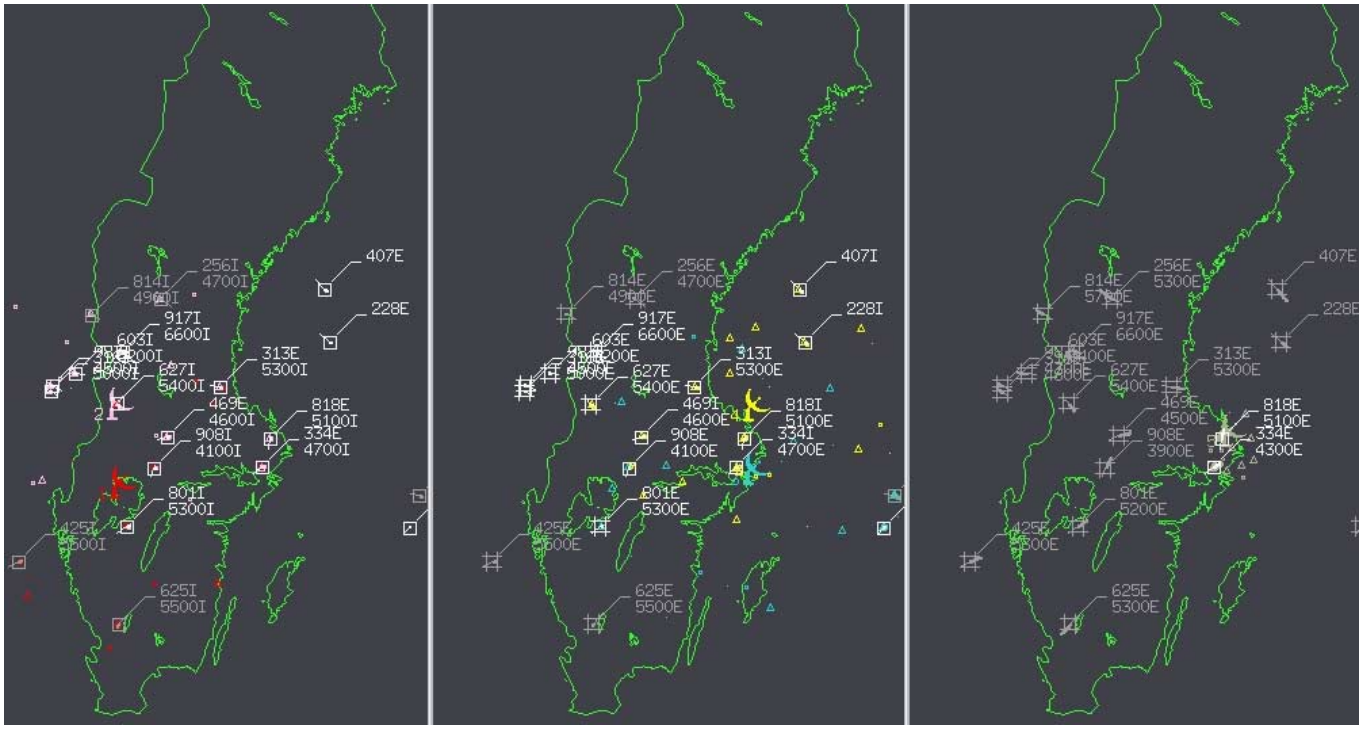


Figure 8. ***Screen-shots taken from the prototype system at a later time when the communication is in operation.***

A rather good accuracy is required to engage the weapon system from a Surface-To-Air Missile Unit. To avoid using their own radar, the decision-makers in Unit 3 would try to increase the accuracy. The WASP concept offers several alternatives to consider: The decision-makers can try to obtain more bandwidth. They can also use prioritisation to give more bandwidth to important targets, or they can restrict the subscription of external targets to a much smaller area. If it is decided not to improve the accuracy by any of these alternatives, the Surface-To-Air Missile unit still has the overview of the situation outside the range of its own radar, and the common track numbers is greatly facilitating co-operation with other units.

9. Conclusions

The use of a Common Battle-Space Model will improve the decision performance dramatically. The combination of MST and WASP will contribute substantially to such a model, regarding moving objects. MST and WASP can make flexible use of very different kinds of data sources. Both these components already exist, the MST as a heavily tested product sold and integrted into several systems, and WASP on the prototype level, sucessfully integrated into a state-of-the-art air-surveillance centre. By the methods to correlate tracks, to manage bandwidth, to manage data subscription, and to estimate data precision, we already are on the way to accomplish a Common Battle-Space Model.

REFERENCES

[1] D.S. Alberts, J.J. Gartska, and F.P. Stein. *Network Centric Warfare.* U.S. D.o.D., 1999.

[2] V. Gawron, *Situational Awareness* (Lecture notes), H. Silver and Associates, UK, 1997.

[3] HKV. *RMA – A new foundation for defense forces development* (in Swedish). Technical Report HKV 09 100:63046, Swedish National Defence, 1999.

[4] HKV. *StabsR 1 Fu* (in Swedish), Swedish National Defence, 1996.

[5] C. Sundin, and H. Friman. *Rolf 2010 – a mobile joint command and control concept.* Technical Report ISBN 91-87136-33-3. ISSN 1403-2120, Swedish Defence College, 1998.

[6] D.Wengelin. *Notes on how progress in information technology spurs revolution in military affairs.* TechNet Europe 2000, 21st AFCEA Europe Symposium & Exposition, Prague, Czech Republic 2000.

This page has been deliberately left blank

Page intentionnellement blanche

Data Fusion and the Coalition Common Operating Picture

Gp Capt James Stewart
MOD DSc(BMD)
Northumberland House
Northumberland Avenue
London
WC2N 5BP
England

Mr Alan Collinson
Collinson Systems Limited
25 Thornton Road
Pickering
North Yorkshire
YO18 7HZ
England

Dr Leslie Pierre
BMDO
Pentagon
7100 Defense
Washington DC
DC 20301-7100
USA

Dr Brian Shand
Advanced Systems Architectures Limited
North Block
Bentley Hall
Blacknest
Alton
Hampshire
England

Mr Paul James
DSTL
St Andrews Road
Gt Malvern
Worcs
WR14 3PS
England

1. ACKNOWLEDGEMENTS

The authors wish to acknowledge the contributions made to this paper of the following individuals and organisations, Mr Felner of the Ballistic Missile Defense Organisation (BMDO), Mr Morgenstern of John Morgenstern Associates, Mr David Himelright of the White Sands Missile Range, Mr Mclaren of Science Applications Incorporated, Mr Rhodes of Rhodes Research, Mr Bailey of the Ministry of Defence, Directorate of Science, (Ballistic Missile Defence) MOD DSc(BMD) and Mr Gordon Evans of Vanguard Research Inc.

2. BACKGROUND

BMDO and the UK Defence Evaluation and Research Agency (DERA) under the direction of the MOD DSc(BMD) have undertaken a series of technical demonstrations to investigate data fusion applied to development of an operational picture. BMDO and MoD DSc(BMD) are concerned with the development of ballistic missile technology. However, ballistic missiles are only part of the threat in theater. The aim must be to integrate BMD into theater defense and use the common infrastructure. Hence, despite the focus on BMD, the work described has been undertaken in both ballistic targets and air breathing threat environments. Some further work is already being planned to be carried out later this year at Joint Exercise Roving Sands. This presentation describes the scope of the work undertaken and presents some of the most interesting results.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

3. THE DATA FUSION ENVIRONMENT

3.1 A Data Fusion Model Data Fusion has been analysed by the Joint (US) Directors of Laboratories, Data Fusion Working Group (JDLDFWG) [1, 2, 3]. The JDLDFWG have developed a functional model to represent the data fusion process. The model is illustrated in Figure 1. Five hierarchical levels of functionality have been identified.

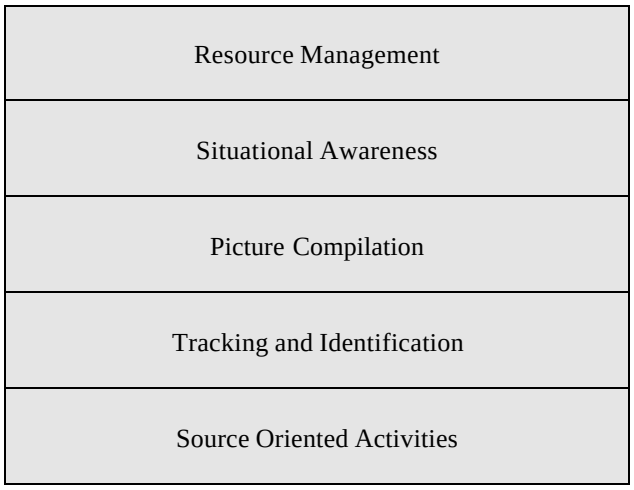


Figure 1 Joint Directors of Laboratories Data Fusion Working Group Model

3.2 Source Oriented Activities (Level 0) Source oriented activities are the lowest, least "abstract" functions carried out to achieve Data Fusion. These activities are associated with the source of the information, viz the sensors. The sub-processes which occur at this level are as follows.

a. Formatting of messages from the sensors

b. Coordinate conversion

c. Time alignment

d. Bias correction

e. Registration (Gridlock)

These processes were discussed in detail by a Joint US/UK Multi-Sensor Tracking and Fusion Forum (MTAFF) in March 1999. It was recognised during the MTAFF workshop that these functions were critical to optimal data fusion.

3.3 Tracking And Identification (Level 1) The source oriented processes might be viewed as the precursor to data fusion. The tracking and identification level deals with the fusion of sensor data which has been optimised by the "Level 0" processes. The purpose of the level 1 activities are to build a description of each

[1] Bedworth and O'Brien (1999), "UK Data Fusion System Development", Mark Bedworth and Jane O'Brien, Proceedings of Military Data Fusion 1999, London
[2] Hall and Llinas (1997), "An Introduction to Multisensor Data Fusion", David L. Hall and James Llinas, Proceedings of IEEE, Vol 85, No 1, January 1997
[3] Anthony (1995), "Principles of Data Fusion Automation", Richard T. Antony, Publisher: Artech House, ISBN: 0-89006-760-0

entity in the field of regard of the overall system (the operational picture). In general, the description deals with both kinematic and classification of objects.

3.4 Picture Compilation (Level 2) After the Level 1 processes have built a description of each entity, the Level 2 processes must build a picture which can be presented to operators which represent the entities. These functions include representing the data from Level 2 in a way which can be best absorbed by operators. The types of algorithm which must be utilised at this level of activity are those which deal with, for example, the length of tracks which should be reported, i.e. the filtering of the track data.

3.5 Situational Awareness (Level 3) The situational awareness functionality aims to interpret the information provided in the picture in the context of the overall situation. Additional data must be fused to provide the contextual situation which might include intelligence sources of data and the fusion of pieces of data from the picture level, for example that platforms are related. Situational Awareness has been defined as when one possesses the knowledge or cognizance of objects and their locations, places, events, status, orders, conditions, and states relative to the viewer's environment. It is primarily concerned with sharing common, accurate, unambiguous information among appropriate policy and operational indites with sufficient timeliness to assess and influence actions. In the simplest of terms good Situational Awareness is achieved when one's perception of his current environment mirrors reality!

3.6 Resource Management (Level 4) Resource Management is concerned with the development an execution of a plan based on the situational awareness. In other words Level 4 is concerned with Command and Control.

4. THE DATA FUSION ENGINE

The work carried out by DERA used the Target Oriented Tracking System (TOTS) as the fusion engine. TOTS is a real-time, multi-target, tracking and sensor data fusion system developed by Advanced System Architectures Ltd with US and UK Government support. It employs a unique, autonomous, multiple model, multiple hypothesis paradigm to provide accurate, low-latency tracks from data provided by one or more sources.

TOTS accepts and fuses asynchronous time-tagged measurements from any mix of active and passive sensors and can simultaneously track all target types i.e. aircraft, ballistic and cruise missiles and interceptors. TOTS offers:

- Low processing latency
- High precision, continuous tracking even through the abrupt changes in target behaviour which are typical in air and missile defence environments
- Implementation scalability such that systems can be easily configured for different threat scenarios
- Intrinsic support for a wide range of collateral information to be attached to a track

TOTS provides intrinsic multi-source fusion of position and kinematic data. Although it does not provide direct fusion of track identity, tracks can inherit identity and other source-derived attributes from contributing source data (e.g. Link 16).

5. ASCIET

Most of the results presented in this paper were established at an All Services Combat Identification Evaluation Team (ASCIET) exercise. However, some more general results will also be discussed. The J-ATOM (Joint ASCIET TOTS Operational Measurement) was a joint US/UK experiment sponsored by the BMDO. J-ATOM was designed to evaluate the performance of kinematic composite tracking in a typical operational environment using TOTS and standard air defense assets.

During the 1991 Gulf War considerable media attention was focussed on the coalition losses due to fratricide. ASCIET was established to identify the causes and ways of preventing such losses. Between 28 February and 10 March 2000, an ASCIET exercise comprising air, ground and sea activities took place around Savannah, Georgia. Naval, Marine, Air and Land forces from both the US and the UK took part. A large number of sensor systems participated in the exercise and were operating under realistic combat conditions. Many of the platforms were instrumented to allow their true position to be broadcast to participating agencies. Therefore, ASCIET provided a means of evaluating the tracking and fusion performance of the TOTS in an operational, instrumented, multi-national air and missile environment, with the post trial addition of real Theatre Ballistic Missile (TBM) data to the air tracks.

The specific objectives of JATOM were to address: -

- **Link 16 Issues:** To provide insight into the potential for TOTS to process and resolve duplicate and broken Link 16 tracks.

and

- **Sensor Fusion:** Provide insight into the capabilities of TOTS as a plot correlation and fusion device using unmodified operational sensors.

6. J-ATOM SENSORS

J-ATOM collected 2D and 3D primary and IFF data from two AN/TPS-75 and two AN/TPS-59 radars providing us with updates approximately every 10 second. Real time connection to the AN/TPS-59 radars was not authorized during the exercise. However, plot and IFF data recordings from the AN/TPS-59 radars were later made available and used in post trial analysis. Data was collected over a region of radius 250 Nm, (450 km) which included the exercise area together with considerable amount of additional coverage

The system was also required to take Link 16 traffic from the ASCIET Data Net (ADNET) and combine this with the radar data. The Link 16 network included some 50 participants reporting position and air tracks. The ASCIET exercise area was adjacent to several major airports and a number of major air lanes. Thus a significant part of the air traffic comprised civil air traffic movements, which contributed significantly to the total ‘track load’. Typically, the ASCIET exercise traffic was of the order of 25 tracks at any one time, but the civilian air traffic load could increase this by a factor of ten. ASCIET traffic comprised fighter aircraft, helicopters, surveillance assets and cruise missile surrogates. The high density and distribution of targets encountered during the exercise was easily handled in real-time by the J-ATOM system.

There were two TOTS systems at the heart of J-ATOM, one deployed at Hunter Army Airfield and the other at Wright Army Airfield. Either of these could take and fuse the data from any of the radars along with Link 16 data. Track displays were collocated with the TOTS systems. Management of the system was from a console based at Hunter Army Airfield.

7. REGISTRATION

An essential pre-requisite of positional and kinematic data fusion is that the data must be represented in a common spatial and temporal frame. Several processes are required to carry this out: the first process we shall consider is sensor registration

7.1 Spatial Registration It was observed from the data gathered from the TPS-75 radars during the exercise set-up period that the tracks were being formed on individual targets with noticeable spatial offsets between them. Analysis of this offset using the TOTS bias estimation tool revealed a 6.9-degree azimuth alignment difference between the two radars. Subsequently, one of the radars was re-aligned to true north, a correction of 1.3 degrees. The residual 5.6-degree error was found to be due to the other radar having been aligned

to geo-magnetic North whereas the AN/TPS-59 radars and Link 16 sensors had been correctly aligned to true north.

In any network of systems, timing differences between individual elements are likely. However, in order to achieve optimum kinematic data fusion, it is essential to establish a common measurement time reference.

7.2 Temporal Registration Although real-time data fusion was successfully achieved throughout the exercise period, the post trial analysis work revealed the presence of measurement time biases. It was possible during analysis to use compensation which left no residual time bias, and data fusion produced a generally clearer picture with longer tracks and fewer breaks.

In operational service, corrections for time bias will have to be made in real-time. Techniques to achieve this have yet to be identified. During post trial analysis work, prototype tools for measuring fixed time biases were used to verify that there were no time biases between the radars. We believe there is there is a great deal of scope for further work to develop the tools to allow the war fighter to identify and offset timing errors rapidly.

It should be noted that fixed time biases only represent part of a wider problem. Time-variant timing errors, such as drift, also need to be considered. Reporting of time on data links presents another significant problem. In the worst case, some data links do not provide measurement time. A common problem with distributed sensor systems is message latency and indeed for some missions, such as fire control, latency can be a limiting factor. Latency results from the following causes: -

The time taken by the sensor to produce a report from a measurement

The time taken by communications to convey a report to its destination

The effect of network loading

It is important to note that latency may not be constant and has to be allowed for. During JATOM a range of latencies were observed.

J-ATOM was designed to investigate the effects of latency on picture compilation in a distributed tracking system. The architecture included two tracking nodes at well-separated sites, where latencies from the sensors would be different. Restrictions at the exercise prevented exploitation of this feature. We believe that it is important that the work started at ASCIET 00 on latency and its effects is carried on in the future.

8. SENSOR CHARACTERISATION

In order to achieve optimum tracking and kinematic data fusion, it is necessary to understand the measurement error characteristics of the contributing data sources. During the exercise initial values were derived from sensor specifications. While not ideal, this was sufficient to allow the data to be fused during the exercise.

During the post trial analysis work, these values were refined by detailed examination of the recorded data but, Clearly, what is needed for the future are tool which are capable of characterising all the sensors.

Similar characterisation issues also apply to Link 16. The following issues exacerbated by at least three problems: -

Firstly, the Link 16’s Track Quality Indicator which is represents the source system’s confidence in its track report. However, as it is generally implemented, it is not ideally suited to represent track quality for a subsequent data fusion process. Further, the use of TQI is inconsistent between different sensors, and also sensors reported by other data links e.g. Link 11.

But secondly, it is also worth noting that TQI is an active component in establishing Reporting Responsibility. Erroneous TQI assignment with the resulting incorrect assumption of reporting responsibility can prevent the best quality data from being distributed on the net. For example, high quality AWACS reports may be suppressed by a weapon system that is reporting a higher TQI on the (not always correct) assumption that it has better data. We see here an example of this problem. Notice how initially JTIDS reports closely follow the TPS-75 reports of the platform altitude. At the point indicated, R2 is taken over by a sensor reporting a higher TQI. However, as can be seen, the altitude track is considerably degraded from that point onwards!

And thirdly, for the purposes of kinematic data fusion, knowledge is required of the quality of the data being reported. Unfortunately, as we have seen, TQI reported in Link 16 air tracks cannot be relied on to provide this. A possible substitute would be to fall back on the quality of the originating sensor measurement. Conceptually, *a priori* knowledge of the sensor platform performance, linked to its position as reported in a PPLI message, could be used to derive some better approximation to the data quality. However, the J-ATOM data set does not include PPLI messages from the deployed sensor platforms which preclude the use of method. Instead for JATOM we took the approach of treating Link 16 as pseudo sensors a set of values were established to represent the uncertainties of the Link 16 pseudo sensor. These values proved adequate to meet the requirements to identify duplicate and broken Link 16 tracks. Through a process of trial and error, pre-fusion measurements facilitated the objectives of the experiment

For the future, we believe it is important to identify a means of addressing, and hopefully solving this problem.

9. SYSTEM PERFORMANCE

9.1 Track Initiation If an object is to appear in the operating picture, it must first be detected and a track must be initialized. Data fusion provides some extremely useful attributes for these first stages of picture generation. Once a sensor has made an initial detection of an object and a track has been initiated, that track information can be used by other sensors to enhance their own detection processes. In other words, it might be possible to cue other sensors. There is, however, a further benefit which should not be underestimated. If multiple sensors are able to view an object and their data is fused, generally, it would be expected that the uncertainty associated with the fused position and velocity of the object would be better, i.e. reduced, compared with any of the single sensors. There are some exceptions to this general rule, for example if one or more of the sensors is heavily biased. Where by heavily we mean the bias is large compared with the sensor uncertainty. We have observed that this improvement in sensor uncertainty is achieved immediately that the second and subsequent sensors start to contribute observations to a track. This is a very useful attribute. Whereas, without fusion, each time an object is tracked by a sensor the track accuracy takes some time to converge, with fusion an already established track is improved. This can greatly extend battlespace.

9.2 Completeness There are many instances where use of multiple sensors can extend the operating picture to cover a greater volume of the battlespace or make it more complete. In both instances this is of particular importance in a world with more sophisticated threats such as cruise and ballistic missiles. To illustrate the point, in post-trials analysis from ASCIET we examined the contributions which primary radar information can make to IFF data. In an ideal world, every radar would provide accurate 3 dimensional reports on every aircraft. In practice, the picture would be generated entirely from combined primary and IFF plots as these provide to provide more accurate measurements of the target height than do the primary radar plots. However, for lots of reasons the IFF replies may not be available, for example when an aircraft banks and the IFF antenna is obscured. If reports are ignored if they do not have IFF the picture can be degraded. We addressed this issue by using data from the TPS-75 radars located at Hunter and the CRTC.

If data is used which consists only of primary plus IFF there can be a large number gaps in the reported target position. We identified gaps of greater than 40 seconds. Worse, these gaps tend to occur close to maneuvers, at precisely the time when a good, complete, picture is most needed. These gaps occur because the secondary radar was unable to obtain IFF returns from the aircraft during at these times. Although track is maintained through the gaps, an air picture based on this data could not be regarded as timely and accurate. However, if the primary only

data is used to update the track at these times the gaps disappear. This is a simple example but we feel it illustrates the point we wish to make. In addition, the same point which was made earlier about the effects of enhanced track uncertainty are still appropriate in this case.

10. HANDLING LINK-16 DUALS

Now onto the performance of the system in identifying Link 16 duals. It is generally accepted that the assignment of R2 in the JTIDS net often leads to the more than one platform reporting on a single target. After tracking the sensor data from ASCIET 00 taken on the 7th March, the one hundred longest TOTS tracks were extracted and examined for dual and multiple reporting over Link 16. As each track produced by J-ATOM contained a record of each instant that a Link 16 track report had been fused with it. Of the 100 tracks examined, 61 contained reports with a unique JTIDS track number throughout their lifetime. Eighteen tracks had a single JTIDS unit reporting at any one time but the JTIDS track number changed at least once. Fourteen tracks had one or more simultaneous units reporting different track numbers on the same targets; sometimes the same platform would be providing two track numbers.

10.1 Elimination Of Duals However, TOTS was demonstrated as being able to produce a single continuous track on an object for which a number of Link 16 track fragments were reported simultaneously. From this, it is apparent that the J-ATOM system was capable of identifying a number of track duals and track breaks occurring in the Link 16 picture. Clearly, a good next step would be to add a process to perform this automatically.

11. CONCLUSIONS AND WAY FORWARD

The principle improvements we were able to demonstrate in the air picture were as follows.

We were able to identify Link 16 duals successfully and demonstrated our ability to track ballistic and air breathing targets simultaneously

We showed the potential of fusion to improve the air picture

We also showed how we can support sensor registration

And finally, we demonstrated some contributions to interoperability by demonstrating how non JTIDS participating sensors could contribute to the air picture

As to the future, there is a great deal of scope for further work to develop the tools to allow the war fighter to identify and offset timing rapidly. For datalink data to be used in conjunction with non-JTDS participating sensor units, further work into the use of sensor measurement uncertainties to represent air track positional uncertainties needs to be undertaken. The potential benefits to be gained from an ability to make use of the identity data in the fusion process and also to aid in the track initiation process needs to be investigated. Latency remains a critical issue and it is recommended that the planned experiment be carried out in a future exercise. To exploit this capability a means of automating the process of identifying the duals would need to be developed. Finally, we believe that tools must be developed to characterise all the sensors which contribute to the picture.

This page has been deliberately left blank

Page intentionnellement blanche

Coalition Requirements for Shared Situational Awareness

Gp Capt James Stewart
MOD DSc(BMD)
Northumberland House
Northumberland Avenue
London
WC2N 5BP
England

Mr Alan Collinson
Collinson Systems Limited
25 Thornton Road
Pickering
North Yorkshire
YO18 7HZ
England

Dr Leslie Pierre
BMDO
Pentagon
7100 Defense
Washington DC
DC 20301-7100
USA

Mr Gordon Evans
Vanguard Research Inc.
1725 Jefferson Davis Highway
Arlington
Virginia
VA 22202
USA

Wg Cdr Clive Harrison
DSTL
St Andrews Road
Gt Malvern
Worcs
WR14 3PS
England

1. BACKGROUND

The U.S. Ballistic Missile Defense Organization (BMDO) and the UK Ministry of Defence, Directorate of Science (Ballistic Missile Defence) (MOD DSc(BMD)) have undertaken a series of three bilateral Policy-Military Seminars focusing on Theater Ballistic Missiles (TBM). Two additional seminars are being planned. This presentation describes the challenges that are the basis of these seminars, how they are conducted, their most important findings and, in particular, what these seminars have taught the sponsors about the requirements for shared situational awareness which, ultimately, forms the context to interpret the common operating picture.

2. THE CHALLENGES

2.1 Regional Conflicts The U.S. and UK are seeking to reorient their forces from the missions that preoccupied them during the Cold War: from the defense of Europe, defense against Soviet aggression, and regional conflicts to the ability to prosecute and win regional conflicts with their allies as they did in Desert Storm.

In future, our two countries will wish to pursue our strategic national interests, and those of our allies, in various regions of the world. However, in future regional conflicts we may be confronted with Weapons of Mass Destruction (WMD) delivered by ballistic missiles. Therefore, looking ahead and preparing our militaries and policy staffs for that future, it is necessary for us to give some thought and make some preparations for how to contend with these special weapons in the hands of regional opponents.

Paper presented at the RTO IST Symposium on "Information Management Challenges in Achieving Coalition Interoperability", held in Quebec, Canada, 28-30 May 2001, and published in RTO MP-064.

Desert Storm was a wake up call! Recall the surprise when Scud-like ballistic missiles, loaded only with high explosives, although not very effective militarily, were very significant politically. When used against Israel, these weapons threatened to change the character of the war by drawing in other parties or breaking down the coalition aligned against Saddam Hussein. What would the effect have been if WMD had been involved?

Our increased understanding of this new world of regional conflicts helps us identify offensive and defense-related equipment needed to support our forces. We are learning that a conflict involving WMD essentially is a different kind of war and we must reevaluate and fight differently. In any region to which our forces may deploy, the Theater Ballistic Missile (TBM) and other weapons of mass destruction environment will profoundly affect coalition operations.

2.2 The Scope of the Challenge The potential introduction of TBMs into a conflict is likely to affect the following aspects of our independent policy and military decision making.

- Our Government policy
- Planning and execution of both political and military options
- Homeland support
- Host nation support
- Military doctrine
- Tactics, Techniques and Procedures
- Military logistics

2.3 Coalition Affairs To achieve true international interoperability, nations must overcome military constraints, existing political agreements, international treaties and protocols, the force mixes, foreign equipment, and differing national operational concepts which combine to create a host of potential impediments. Formal and informal political and military relationships between and among the government, military, paramilitary, and civilian organizations within each national entity with whom the coalition partners interacts will have considerable bearing on what is desirable and feasible for coordinating effective and timely situational awareness within the region. Relationships are often dictated by cultural characteristics, historical precedent or submerged ethnic groupings that might have little or no bearing on the specific political or military problem being faced.

It is our belief that coalition partners, even as close to one another as the U.S. and the UK, as well as host nations would likely find the following factors in play:

- National interests versus coalition interests
- National sensitivities and strength of political support for participation
- Desired command, control, communications, computers and intelligence (C4I) systems and battle management behavior
- Centralized vice decentralized control of information systems and processes
- Responsibility (which partner) for information systems' overall performance

- Assignment of responsibility and the relationship and paths of communication & interaction between coalition partners and host nation

- Objectives postulated for theater missile defense within the host country under consideration, from the standpoint of that country and the other partner

- Differences that might affect C4I and battle management relations (i.e., de facto U.S. control vs. shared responsibilities, degree of sophistication in operating and using advanced electronic-based systems, mutual recognition, language barriers, and cultural characteristics)

The political and military relationship of both coalition partners with the host nation will be most likely challenging and, at times, stressful. The issue of what and when the partners share with the host nation is a most sensitive and critical one and must be addressed early in the deployment decision process. Partners will be reluctant to release more than the absolute minimum required to support their operations and maintain host nation support. Release of certain categories of information is often not granted. This hampers the host nation's active participation in the day-to-day coalition affairs. It is important that what is shared with the host nation is accurate, unambiguous and with sufficient timeliness to assess and influence operations.

The deployed coalition force has a responsibility to provide the counterforce, active defense, passive defense, and C3I capabilities to deter TBM dangers, and protect their forces. Each of the partners will be contributing to all or some of those pillars according to their respective force mix. All aspects of the kill chain must be aggressively addressed in the planning of the TBMD.

3. WHAT IS A POL MIL SEMINAR AND HOW IS ONE CONDUCTED?

To conduct a Policy-Miltary seminar we assemble a group of experts drawn from civil service policy makers and the military and for two days facilitate their discussions of a series of challenging scenarios and events. Considering the truism that "what is good for the U.S. forces may not be good for all nations involved with the U.S. in a multinational endeavor", assured that the seminars' discussions would be spirited. To ensure success, the sponsor's first task is to determine the objectives. Ideally, the objectives are settled nine months to a year before the seminar. Once the sponsors have nominated the objectives, the process of conducting the seminar is handed over to the seminar team. BMDO and MOD DSc(BMD) believe that one pre-requisite of a successful seminar is a good team of domain, that is BMD, experts.

The team must design a series of "vignettes" which will expose the points of interest to the sponsors. Each vignette is approximately an hour long and over the course of a two day seminar there may be as many as eight separate vignettes.

A particularly challenging aspect of the design of the seminar is the choice of the scenario, that is, the epoch and location on which the players must focus. For example, if the epoch is set too close to the present day, players may find their own experience tells them an event is so unlikely as to be incredible. Conversely, if the epoch is too far in the future, the scenario becomes too far removed from players' experience and the outcome may be random. The art of scenario design is not unlike choosing the setting for a play, it must be sufficiently credible to "suspend disbelief". We have discovered that ten years hence is about right. As for the location, we have made some interesting discoveries about that and this will be addressed later in the paper.

Having been presented with a set of objectives and a scenario selected, the seminar team must then devise a series of events that will exercise the players' minds. This brings us to the players. There can be no doubt that the players are the most important part of a seminar, they will be drawn from the decision makers, or those near them, the people who make a difference. The players will be drawn from all three armed services and different policy agencies or departments. For example, the UK has provided civil servants from the Ministry of Defence, the Foreign and Commonwealth Office and staff from the British Embassy in the U.S.. The U.S. has provided policy staff from the State Department, the Office of the Secretary of Defense, USSPACECOM, and the Air Staff. Each

nation provides approximately nine or ten players drawn from this mix of policy and military staff. We have found it expeditious to use a mixture of ab-initio and experienced players, that is players who have played in a previous seminar. Also, the seminar team has found that it is wise to invite more players, usually between ten and twelve, than are expected to take part. We have discovered that there are always some who have to drop out at the last minute due to unforeseen and urgent operational or policy matters. Incidentally, this is how we know we are finding the right type of players!

We have discovered there are some important rules and guidelines for conducting seminars.

a. All discussions are carried out without attribution - thus the players can speak freely. Their names will never be disclosed. This is our most important rule.

b. Everyone must speak his or her mind

d. It is important that the players realize they are not being examined

e. There are no right answers or pat solutions

f. There is no such thing as a stupid question

f. And finally, Players must enjoy the seminars - if we didn't make the experience rewarding we would not be able to get people to come along to a second seminar

Whilst not a rule, we have discovered an important tool that helps each seminar along. To allow the group to get to know each other better and allow the discussion to continue we have found it a good idea to hold a seminar dinner overnight between the two seminar days - sometimes the best thoughts come after a beer! We call this the Samuel Adams Factor!

The facilitators lead the discussion during each vignette as it is played out. The use of domain experts means that advice can be given on TBMs and TBMD, if required. We deliberately split the group into teams for some vignettes and sometimes hold plenary sessions. Supporting the facilitators, each nation provides analyst/rapporteurs who record the key points raised in the discussion. Following each seminar the facilitators, analysts, and the rapporteurs will spend equally as much time as was spent during the seminar drawing together and ordering the key points to form the final report.

One factor we have discovered is that often the UK players feel they are entering the seminar at a disadvantage to their U.S. colleagues. No one likes to feel they are speaking from a position of ignorance and to address this the UK holds a pre-seminar BMD Primer day, what might be termed, "BMD 101". At the primer day, the facilitators introduce the UK players to BMD through a series of lectures and structured exercises. These primer days have proven to be popular and we have found we regularly have more attendees than will play in the seminars.

4. SOME SEMINAR FINDINGS

So have we learned anything of any value at these seminars? It is probably best to discuss some of the general findings before considering the more specific findings related to the common operating picture.

4.1 The Strategic Missile The first finding simply reconfirmed our intuitive belief that was observed in Desert Storm. The consensus of all our players has been that the TBM is essentially a strategic weapon which is used to attempt to alter the course of a campaign and that the use of a missile may have a large political effect. The military effect of use of the TBM was seen as less important. It may impact the choice of a port of debarkation and it may have a small impact on the tempo of a campaign but it was not generally considered to have a major military impact.

4.2 Public Opinion The flight time of a TBM can be short, in fact very short, say 5 minutes or less. And a TBM might well carry WMD warheads. Thus, in less than five minutes, the whole nature of a campaign could change. This may give rise to a whole series of knock on effects. One effect which our players thought was extremely important was the effect this may have on the general public. Unless they had been carefully prepared beforehand, and perhaps not even then, the public would very easily be swayed by sight of the effects of a missile impact, especially if that were to include the effects of WMD. In short the "CNN effect".

We determined that the public opinion about a campaign would, in general, assume one of three postures. The public might be broadly in support of the coalition, but after an impact had occurred, public sentiment might swing towards a clamor for a massive retaliatory strike or, alternatively, towards a desire to withdraw from the coalition effort. It was believed that the use of WMD would be likely to swing public opinion towards the latter states and that missile defence would tend to restore general confidence and support.

It might be worth adding as a rider that, although there was considerable discussion on the subject, no firm conclusions were reached on the relative tolerance of the UK and U.S. public. We must investigate whether there are thresholds of tolerance at a future seminar but it was believed possible that no threshold can be determined in advance. Expressing this technically the threshold may be chaotic. An alternative way of thinking about this is that it is a matter of emotion.

4.3 Longer Range Missiles – The Theatre or the Region Perhaps the most important findings were all concerned with the effect of introducing longer range missiles into a scenario. We have all been "brought up" on the belief that a major military consideration is to constrain conflict within a theater. However, if the range of a TBM increases only marginally, some interesting effects occur. As soon as a TBM can overfly a country, third party states begin to get involved. This situation was seen during Desert Storm with the attempt to introduce Israel into the war. More generally, it should be remembered that "Red" has friends as well as enemies. With longer range missiles, if things go badly for Red, their friends may be tempted to influence the conflict by using their own missiles. Neither should we forget the proliferation chain. Red may not have obtained their missiles from indigenous development. If the next country in the chain becomes involved constraint gets progressively harder.

We have come to believe that the effect of the longer range missile is to make the concept of the theater redundant. Indeed, you may have already noticed that throughout this paper we have tended to use the term "region" instead of theatre.

4.4 Longer Range Missiles – The Homeland Just as was the case with the discussion of the theater or the region, we have certain preconditioned expectations of how to prosecute warfare. Another example of this may be our attitude to consideration of use of a pre-emptive strike. Our every instinct is not to consider pre-emptive strike partly because there is the likelihood that world opinion would not favour such a move but also, in truth, probably because it is such a difficult military option. However, our players were of the opinion that if the homeland was threatened, as our players put it when the enemy can "reach out and touch the homeland", there may be situations when pre-emptive strike was worthy of consideration. This situation was particularly the case when there was a possibility of the use of WMD. One final observation before leaving this point, the host nation is someone's homeland as well.

4.5 Longer Range Missiles – Strategic National Interests To summarize the special influence of longer range missiles, our players believed that, while it was inevitable that Strategic National Interests would be pursued in theaters (regions) around the world, the ability of long range missiles to reach the homeland was likely to lead us to review our Strategic National Interests and adopt a more strongly focussed attitude to interventionism.

4.6 On Deterrence For the U.S. and the UK, the cornerstone of Cold War defence doctrine was deterrence based on nuclear forces. The seminars suggest a wider spectrum of deterrent responses have to be considered. A particularly important deterrent was the potential for massive conventional response. However, it is interesting to

note that the players also believed that Ballistic Missile Defence (BMD) also had deterrent properties. This deterrent capability arises because the obvious deployment of BMD provides a clear and overt signal that the coalition intends to remain in the region and intends to pursue the strategic national interests against the threat of the TBM.

4.7 The Framework of Our Operations Success is achieved when the Policy Objectives of the coalition are met. The tools available to coalition leaders to meet their objectives are drawn from Political, Military, and Economic options. Only too often the options are drawn up independently of each other. However, all these options are intertwined and should not be considered in isolation. For example, military options should consider the economic and policy implications of a given course of action.

5. SITUATIONAL AWARENESS

The seminars have also provided us with a better understanding of the requirements for establishing effective coalition situational awareness and some of which may even flow down to requirements for the Common Operating Picture.

The players believed that the first priority for establishing situational awareness as the coalition entered a region was is to step up surveillance and reconnaissance to achieve the most complete intelligence product possible. In attempting to define the meaning of a complete intelligence product, the group consensus was that it must be robust enough to deal with <u>any</u> aggressive act. However, a particular concern is the ability to deal with ambiguous acts. It was regarded as imperative to establish the intent of the aggressor even in the earliest stages of a campaign. Further, a way needs to be found to express the level of uncertainty associated with information.

The players considered that it was essential to have a clearly defined information exchange policy prior to the start of conflict and separate policies for each coalition member and for the coalition members with the Host Nation. There was considerable discussion about the information exchange policy. Initially, the views of the players tended towards the exchange of as much information as possible. As the discussion continued the merits of various caveats were appreciated, for example, the protection of sources. Finally, a consensus was reached that only the minimum essential information should be passed to a host nation and that would always exclude the source of the information. We must investigate this further in future seminars.

Perhaps the most important and far reaching of the findings of these seminars are those concerned with the effects of the increasing range of missiles. The rest of the findings determine the way in which the common operating picture is interpreted but range has a direct effect on the requirements for the common operating picture. Our operating picture may be quite inadequate unless its scope covers the whole of a region.

6. SUMMARY

We believe our players have already given us some profound insights, and we believe taken these insights back to their desks, but we also realise that we are only starting to scratch the surface. If a single lesson was to be restated which represents perhaps our most important finding to date, it has to be the effect of longer range missiles, within a region and, perhaps more importantly when they reach out to the homeland. The effects of these missiles when coupled with a WMD capability are profound.

As we have already said, our sponsors have already given us the direction to hold at least another two seminars and these are currently in the planning stages. What are our plans for these seminars? Our strategy will be to explore these issues in a systematic way, addressing each of the pillars of BMD. However, as the opportunities arise we shall also investigate some issues in more depth.

In conclusion, we believe the policy-military seminar has proven to be a valuable tool which our respective leaderships are employing to guide us forward in learning how to deal with the effects of ballistic missiles and their capability to deliver strategic effects within regional warfare.

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	2. Originator's References	3. Further Reference	4. Security Classification of Document
	RTO-MP-064 AC/323(IST-022)TP/11	ISBN 92-837-1078-9	UNCLASSIFIED/ UNLIMITED

5. Originator Research and Technology Organisation
North Atlantic Treaty Organisation
BP 25, 7 rue Ancelle, F-92201 Neuilly-sur-Seine Cedex, France

6. Title
Information Management Challenges in Achieving Coalition Interoperability

7. Presented at/sponsored by
the RTO Information Systems Technology Panel (IST) Symposium
held in Quebec, Canada, 28-30 May 2001.

8. Author(s)/Editor(s)	9. Date
Multiple	December 2001
10. Author's/Editor's Address	**11. Pages**
Multiple	302 (text) 15 (slides)

12. Distribution Statement There are no restrictions on the distribution of this document. Information about the availability of this and other RTO unclassified publications is given on the back cover.

13. Keywords/Descriptors

C4I
CCIS (Command and Control Information Systems)
Coalition Interoperability
Common Operating Picture
Communications networks
COTS (Commercial Off The Shelf)
Criteria development
Data fusion
Information management
Information superiority
Integrated systems
Interoperability
Mobile software technologies
NATO operations
Natural language
Ontologies
Peacekeeping
Secure communication
Situational awareness

14. Abstract

This volume contains the Technical Evaluation Report and 25 unclassified papers, presented at the Information Systems Technology Panel Symposium held in Quebec, Canada from 28th to 30th May 2001.

The papers were presented under the following headings:

- Architectures and Standards: Fundamental Issues
- Information Management
- Mobile Software Technologies
- Interoperability Procedures and Practices
- Information Centric Warfare
- Coalition Common Operating Picture

This page has been deliberately left blank

Page intentionnellement blanche

NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION
BP 25 • 7 RUE ANCELLE
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int

DIFFUSION DES PUBLICATIONS
RTO NON CLASSIFIEES

L'Organisation pour la recherche et la technologie de l'OTAN (RTO), détient un stock limité de certaines de ses publications récentes, ainsi que de celles de l'ancien AGARD (Groupe consultatif pour la recherche et les réalisations aérospatiales de l'OTAN). Celles-ci pourront éventuellement être obtenues sous forme de copie papier. Pour de plus amples renseignements concernant l'achat de ces ouvrages, adressez-vous par lettre ou par télécopie à l'adresse indiquée ci-dessus. Veuillez ne pas téléphoner.

Des exemplaires supplémentaires peuvent parfois être obtenus auprès des centres nationaux de distribution indiqués ci-dessous. Si vous souhaitez recevoir toutes les publications de la RTO, ou simplement celles qui concernent certains Panels, vous pouvez demander d'être inclus sur la liste d'envoi de l'un de ces centres.

Les publications de la RTO et de l'AGARD sont en vente auprès des agences de vente indiquées ci-dessous, sous forme de photocopie ou de microfiche. Certains originaux peuvent également être obtenus auprès de CASI.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr, (FIZBw)
Friedrich-Ebert-Allee 34
D-53113 Bonn

BELGIQUE
Coordinateur RTO - VSL/RTO
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evère, B-1140 Bruxelles

CANADA
Services d'information scientifique pour la défense (SISD)
R et D pour la défense Canada
Ministère de la Défense nationale
Ottawa, Ontario K1A 0K2

DANEMARK
Danish Defence Research Establishment
Ryvangs Allé 1, P.O. Box 2715
DK-2100 Copenhagen Ø

ESPAGNE
INTA (RTO/AGARD Publications)
Carretera de Torrejón a Ajalvir, Pk.4
28850 Torrejón de Ardoz - Madrid

ETATS-UNIS
NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076-1320

FRANCE
O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GRECE (Correspondant)
Hellenic Ministry of National Defence
Defence Industry Research & Technology General Directorate
Technological R&D Directorate
D.Soutsou 40, GR-11521, Athens

HONGRIE
Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
H-1525 Budapest P O Box 26

ISLANDE
Director of Aviation
c/o Flugrad
Reykjavik

ITALIE
Centro di Documentazione Tecnico-Scientifica della Difesa
Via XX Settembre 123a
00187 Roma

LUXEMBOURG
Voir Belgique

NORVEGE
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25, NO-2007 Kjeller

PAYS-BAS
NDRCC
DGM/DWOO
P.O. Box 20701
2500 ES Den Haag

POLOGNE
Chief of International Cooperation Division
Research & Development Department
218 Niepodleglosci Av.
00-911 Warsaw

PORTUGAL
Estado Maior da Força Aérea
SDFA - Centro de Documentação
Alfragide
P-2720 Amadora

REPUBLIQUE TCHEQUE
DIC Czech Republic-NATO RTO
VTÚL a PVO Praha
Mladoboleslavská ul.
Praha 9, 197 06, Česká republika

ROYAUME-UNI
Dstl Knowledge Services
Kentigern House, Room 2246
65 Brown Street
Glasgow G2 8EX

TURQUIE
Millî Savunma Başkanliği (MSB)
ARGE Dairesi Başkanliği (MSB)
06650 Bakanliklar - Ankara

AGENCES DE VENTE

NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076-1320
Etats-Unis

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
Royaume-Uni

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council
Document Delivery
Montreal Road, Building M-55
Ottawa K1A 0S2, Canada

Les demandes de documents RTO ou AGARD doivent comporter la dénomination "RTO" ou "AGARD" selon le cas, suivie du numéro de série (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Des références bibliographiques complètes ainsi que des résumés des publications RTO et AGARD figurent dans les journaux suivants:

Scientific and Technical Aerospace Reports (STAR)
STAR peut être consulté en ligne au localisateur de ressources uniformes (URL) suivant:
http://www.sti.nasa.gov/Pubs/star/Star.html
STAR est édité par CASI dans le cadre du programme NASA d'information scientifique et technique (STI)
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
Etats-Unis

Government Reports Announcements & Index (GRA&I)
publié par le National Technical Information Service
Springfield
Virginia 2216
Etats-Unis
(accessible également en mode interactif dans la base de données bibliographiques en ligne du NTIS, et sur CD-ROM)

Imprimé par St-Joseph Ottawa/Hull
(Membre de la Corporation St-Joseph)
45, boul. Sacré-Cœur, Hull (Québec), Canada J8X 1C6

NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION
BP 25 • 7 RUE ANCELLE
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Telefax 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int

DISTRIBUTION OF UNCLASSIFIED RTO PUBLICATIONS

NATO's Research and Technology Organisation (RTO) holds limited quantities of some of its recent publications and those of the former AGARD (Advisory Group for Aerospace Research & Development of NATO), and these may be available for purchase in hard copy form. For more information, write or send a telefax to the address given above. **Please do not telephone**.

Further copies are sometimes available from the National Distribution Centres listed below. If you wish to receive all RTO publications, or just those relating to one or more specific RTO Panels, they may be willing to include you (or your organisation) in their distribution.

RTO and AGARD publications may be purchased from the Sales Agencies listed below, in photocopy or microfiche form. Original copies of some publications may be available from CASI.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Coordinateur RTO - VSL/RTO
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evère, B-1140 Bruxelles

CANADA
Defence Scientific Information Services (DSIS)
Defence R&D Canada
Department of National Defence
Ottawa, Ontario K1A 0K2

CZECH REPUBLIC
DIC Czech Republic-NATO RTO
VTÚL a PVO Praha
Mladoboleslavská ul.
Praha 9, 197 06, Česká republika

DENMARK
Danish Defence Research Establishment
Ryvangs Allé 1, P.O. Box 2715
DK-2100 Copenhagen Ø

FRANCE
O.N.E.R.A. (ISP)
29 Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GERMANY
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr, (FIZBw)
Friedrich-Ebert-Allee 34
D-53113 Bonn

GREECE (Point of Contact)
Hellenic Ministry of National Defence
Defence Industry Research & Technology General Directorate
Technological R&D Directorate
D.Soutsou 40, GR-11521, Athens

HUNGARY
Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
H-1525 Budapest P O Box 26

ICELAND
Director of Aviation
c/o Flugrad
Reykjavik

ITALY
Centro di Documentazione Tecnico-Scientifica della Difesa
Via XX Settembre 123a
00187 Roma

LUXEMBOURG
See Belgium

NETHERLANDS
NDRCC
DGM/DWOO
P.O. Box 20701
2500 ES Den Haag

NORWAY
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25, NO-2007 Kjeller

POLAND
Chief of International Cooperation Division
Research & Development Department
218 Niepodleglosci Av.
00-911 Warsaw

PORTUGAL
Estado Maior da Força Aérea
SDFA - Centro de Documentação
Alfragide
P-2720 Amadora

SPAIN
INTA (RTO/AGARD Publications)
Carretera de Torrejón a Ajalvir, Pk.4
28850 Torrejón de Ardoz - Madrid

TURKEY
Millî Savunma Başkanliği (MSB)
ARGE Dairesi Başkanliği (MSB)
06650 Bakanliklar - Ankara

UNITED KINGDOM
Dstl Knowledge Services
Kentigern House, Room 2246
65 Brown Street
Glasgow G2 8EX

UNITED STATES
NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076-1320

SALES AGENCIES

NASA Center for AeroSpace Information (CASI)
Parkway Center
7121 Standard Drive
Hanover, MD 21076-1320
United States

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
United Kingdom

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council
Document Delivery
Montreal Road, Building M-55
Ottawa K1A 0S2, Canada

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number (for example AGARD-AG-315). Collateral information such as title and publication date is desirable. Full bibliographical references and abstracts of RTO and AGARD publications are given in the following journals:

Scientific and Technical Aerospace Reports (STAR)
STAR is available on-line at the following uniform resource locator:
http://www.sti.nasa.gov/Pubs/star/Star.html
STAR is published by CASI for the NASA Scientific and Technical Information (STI) Program
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
United States

Government Reports Announcements & Index (GRA&I)
published by the National Technical Information Service
Springfield
Virginia 22161
United States
(also available online in the NTIS Bibliographic Database or on CD-ROM)

Printed by St. Joseph Ottawa/Hull
(A St. Joseph Corporation Company)
45 Sacré-Cœur Blvd., Hull (Québec), Canada J8X 1C6

ISBN 92-837-1078-9