MSG-168 Lecture Series on Modelling and Simulation as a Service (MSaaS)

11. MSaaS Discovery Technical Approach

Dr. Keith Ford
Thales UK
Crawley, RH10 9HA
UNITED KINGDOM

keith.ford@uk.thalesgroup.com

ABSTRACT

This paper describes the results from the approach to Discovery developed by a Thales led team on behalf of the scientific branch of the UK MOD (Dstl). It identifies the key components for implementing a discovery capability, which are a registry, metadata and a repository, and describes how they are used.

1.0 INTRODUCTION

This paper describes the results from the approach to MSaaS Discovery developed by a Thales led team on behalf of the scientific branch of the UK MOD (Dstl). MSaaS provides a new approach to delivering simulation capability based on the principles of Service Oriented Architectures (SOA). It provides:

- A fully transparent and integrated method of moving from an operational requirement to an executable simulation that can deliver part or all of that requirement;
- On-demand composition of simulations re-using existing capability where possible and integrating new if required;
- Deployment and execution of simulations to be decoupled from specific hardware and infrastructure to enable flexible use;
- Sharing of paid-for capability – hardware, software, services, infrastructure - across an enterprise.

For these elements to exist in their most exploitable form, the following requirements need to be met:

- All types of M&S resources are available via a persistent and independent facility decoupled from delivery of specific M&S systems to meet existing requirements;
- A wide range of users can interact with this facility on their own terms and to meet their own needs.

The two requirements above are the main reasons why MSaaS differs from current simulation provision. A driving principle behind MSaaS is that for reuse of M&S resources to become the rule rather than the exception, they need to exist and be widely accessible independently of any particular project, programme or portfolio. Current initiatives address some of the underlying technical requirements to deliver these things, but do not offer a complete solution. The MSaaS research shows what a full solution would look like.

Experience from many domains has shown that it is usually more efficient to modify something rather than create it new from scratch. This approach is encouraged by the MSaaS concept. The entry point into the MSaaS ecosystem depends on how the requirements for the simulation environment are specified. These may be provided as an operational scenario or as a list of user needs. The way the MSaaS ecosystem is used will depend on the role and experience of the person using it and what they want to do.

1.1 Definitions

Catalogue: Contains metadata identifying, describing and/or summarising associated resources and provides mechanisms to support discovery and retrieval of the associated resource.

Registry: An authoritative catalogue that provides additional management mechanisms such as access control, versioning, audit trails and workflows to ensure the content is unique and properly authorised.

Metadata: data about data.

Discovery: The process of finding content in a catalogue via browsing, simple or complex searches (including natural language searches) that was previously unknown. It can also be used to find items outside the scope of the initial search scope via associations between items.

In the context of MSaaS, discovery differs from searching as with searching the person knows what they are looking for but doesn't know where to find it.

2.0 DISCOVERY CONCEPT

The discovery concept is shown in Figure 1 and is described in more detail in the 'MSG-168 Overview of MSaaS Concept to Ecosystem' lecture. A Simulation Integrator provided with an operational scenario may initially search the Registry for Scenarios or Compositions that contain the required entities. They may also supplement the Registry queries with additional specific simulation requirements e.g. Exercise Control. The Simulation Integrator may also manually search for individual M&S Assets or M&S Services.

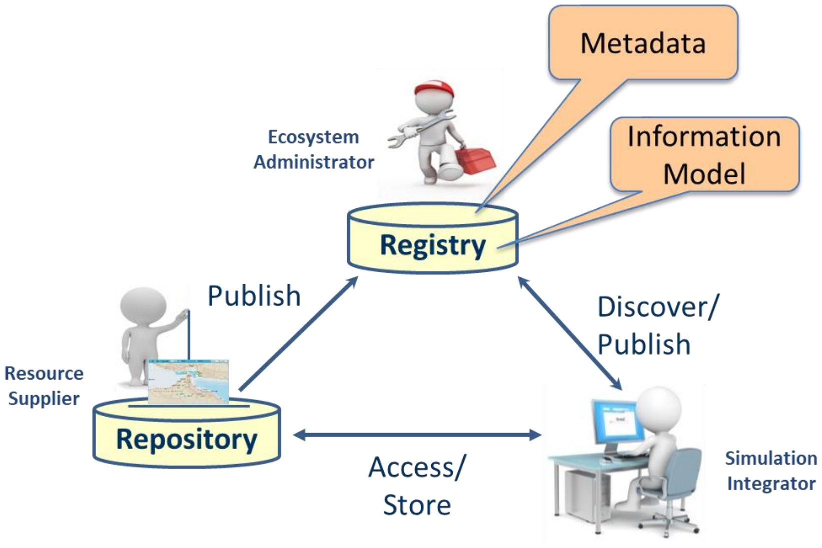

Figure 1: Discovery concepts deliver some elements of MSaaS, making use of registries and repositories to manage and deliver content

If the person knows of a Simulation Environment they want to rerun, they can go straight to the Deployment for it. This can be searched for in terms of the Event name, date when the previous Event was run or the name of the person previously responsible for running the Event, etc.

The Simulation Integrator will evaluate the metadata and supporting information for the discovered Registry Objects to determine their suitability. Ideally a Composition will be discovered that completely satisfies the requirements. In this case the Simulation Integrator can use the power of the Registry's Information Model to identify Scenarios and Deployments associated with the Composition to see if they can be reused as is. If the Composition, Scenario or Deployment doesn't completely satisfy the requirements, the Simulation Integrator can modify them as required. A new Blueprint for the Event is created that links all the objects associated with the Event, which is published to the Registry/Repository so that they are available for future reuse.

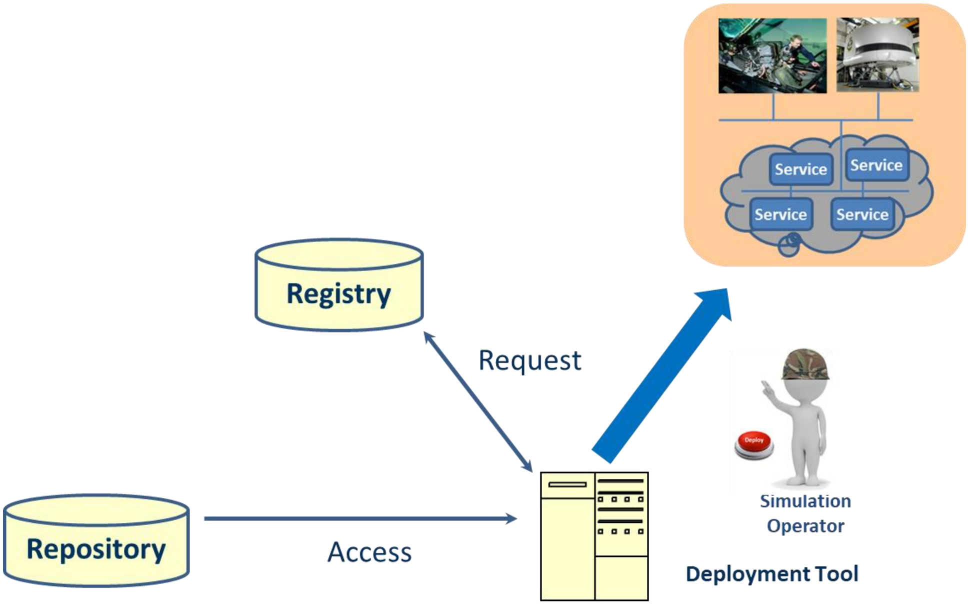

Figure 2: Runtime Discovery

During Runtime[1], the Simulation Operator (which for training could be an instructor) will deploy the simulation to the infrastructure. They will use a Deployment tool to search the Registry for the Deployment object for the required exercise and at a push of a button, deploy the simulation to the infrastructure. Using the information specified in the Deployment object, the Deployment tool accesses the Registry to determine which Repository the services can be downloaded from or to obtain a link for those services running 24/7. For services that have to be installed, the Deployment tool creates virtual machines and containers on the infrastructure to install the executable code. It also creates the network for connecting the services and any simulators that may be used. The deployed services can then be accessed by the Simulation Operator or User. As an example, a Simulation User who is a trainee may search the Registry to obtain on-demand training. This approach hides the complexity of the deployment process to non-technical users.

[1] Equivalent to Distributed Simulation Engineering & Execution Process (DSEEP) steps 5 and 6

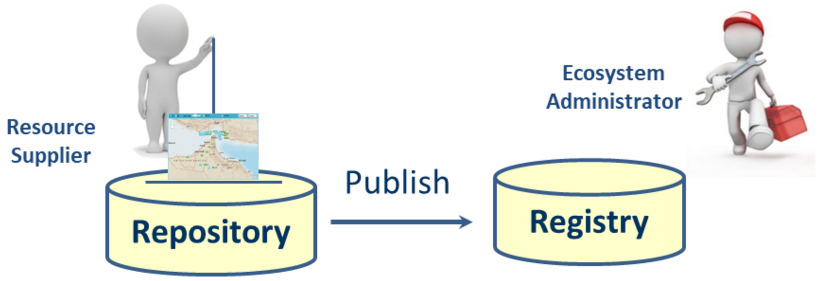

Figure 3: MSaaS Maintenance Activities

The Ecosystem Administrator is responsible for performing several maintenance activities. The most important is to confirm that information about the M&S Resources to be added to the ecosystem is compliant with the required metadata standards and accurately reflects the functionality being offered by the Resource Supplier. They also need to verify that the M&S Resource performs as described.

The Ecosystem Administrator manages all aspects of the Registry including software upgrades and federating with new Registries as they become available. They also need to verify, where applicable, the network connectivity for new Repositories that are added to the ecosystem. The Ecosystem Administrator would also be responsible for access control for the Registries and Repositories and accreditation of the MSaaS ecosystem.

3.0 INFORMATION MODEL

3.1 Discovery and the MSaaS Information Model

The function of the MSaaS information model is to support delivery of MSaaS via the ecosystem. It is part of the set of functionality termed ‘discovery’, which is in turn supported by the registry-repository model for managing information and objects.

The UK approach began with understanding who will interact with the facility and how they will do this. Use cases were developed to describe interactions, and based on this, the team considered how information needs to be structured to enable these interactions to occur. It was concluded that a very modular approach is required in order to enable the flexible use and reuse of services for a range of purposes and as part of multiple simulations.

Defined objects include:

- Objects that allow users to specify a implementation-independent set of requirements (scenario, conceptual model, simulation environment specification);
- Objects that define and describe services of various types;
- Objects that specify a core reference set of abstract entities that all simulation specifications, compositions and services can be linked to define their functionality;
- A simulation composition object which references a set of specification objects and then provides a list of services that will be used to deliver that specification;
- A simulation deployment object which references a simulation composition and specifies how those services will be deployed;
- Objects that define physical assets needed for deployment and their properties;
- An object that defines an event, which is a particular execution of a defined simulation deployment;

This type of structure is required in order to provide the flexibility needed to be able to use the MSaaS ecosystem to its full potential. The UK research developed these concepts within an Enterprise Architect model. The high-level diagram below shows the main properties of these objects and the relationships between them.

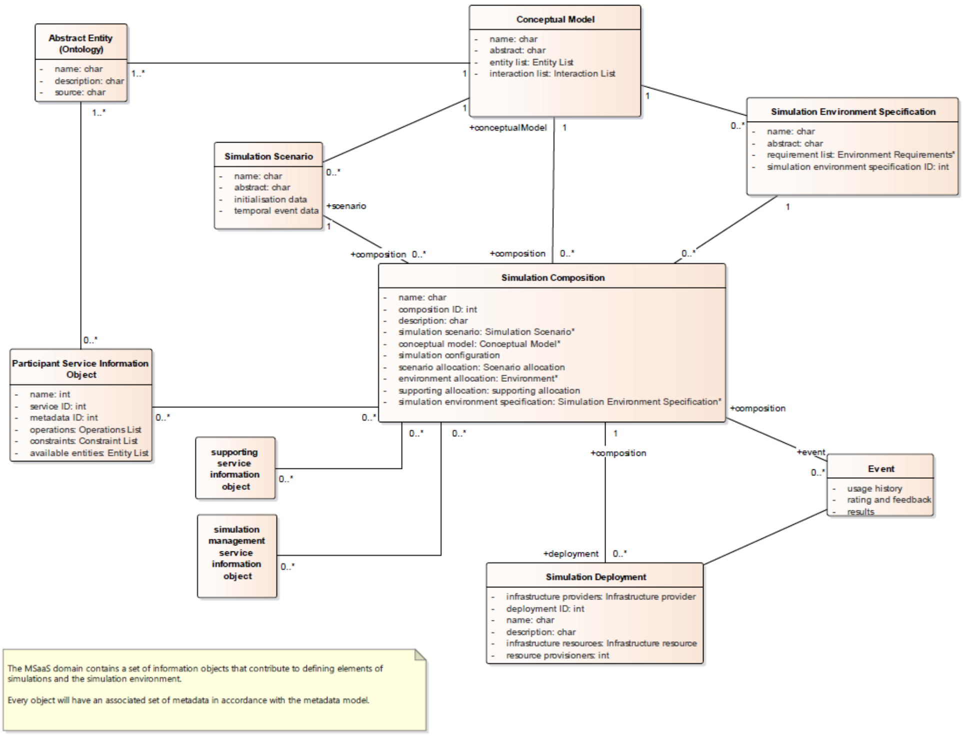

Figure 4: The MSaaS information layer consists of a number of defined M&S information objects that exist within the ecosystem.

The guiding principles for the MSaaS information model are that the model must:

- Be sufficiently flexible to permit the users to incorporate (within reason) data from a range of sources based on different underlying standards;
- Enable users to move from high-level overview down to detailed technical implementation within the same facility (although almost certainly not via the same toolsets and applications)
- Be based on the ecosystem principles as described in standards on SOA such as those from the Open Group and OASIS.

3.2 Information Layers

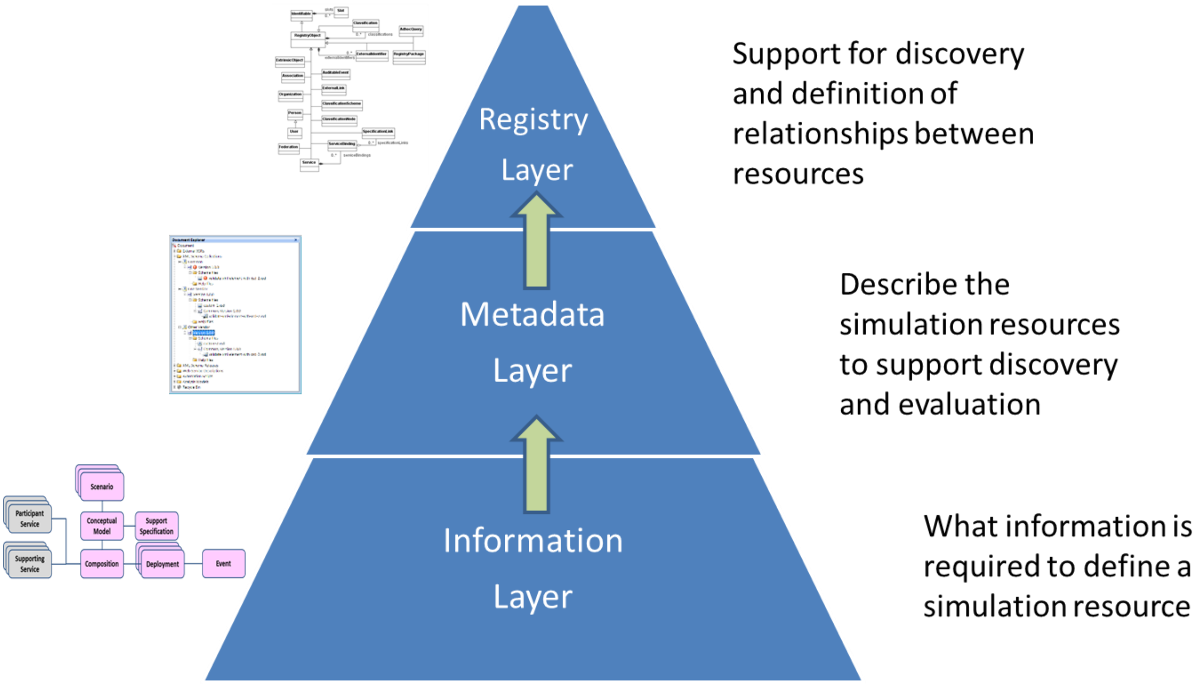

Figure 5: Information Model Layers

The UK approach to discovery was to split the MSaaS information model containing the information, data and metadata needed within the ecosystem into three distinct layers:

- The **MSaaS Information Layer** contains the M&S information objects defining particular constructs within the ecosystem needed to specify, compose, deploy and execute simulations. An information object can either be a construct in its own right (i.e. a composition or deployment object) which contains details of a particular functional element, or it can be associated with something that has a physical existence (for example, a description of a service).
- The **Metadata Layer** contains all metadata related to M&S assets. The purpose of metadata is to enable discovery functionality to be implemented, including machine-to-machine and ecosystem querying elements. The UK definition of discovery supports the entire specify-compose-deploy-execute cycle; it is the full set of functions that enable users to find and reference all other objects within the ecosystem. Metadata is a distillation of the information objects; it always contains a reference to the actual information so does not need to duplicate its entire contents, but only the elements that are needed for discovery functionality.
- The **Registry Layer** is realised using the Registry Information Model (RIM) and is a further distillation of the metadata. Its purpose is to enable implementation of the type of registry required by the UK model, which is one that does not just catalogue, but that actively maintains an up-to-date record not only of objects that exist (both real and conceptual) but also the relationships between them to enable much more sophisticated querying than would otherwise be available. The referenced repository component then contains a full copy of the metadata file and associated annotations.

Together, these three layers comprise the MSaaS information model.

4.0 METADATA

4.1 Overview

Discovery on its own is insufficient to determine if a simulation resource is suitable for employment in a particular exercise. Metadata must also be provided to enable the resource to be intelligently evaluated to assess its suitability for a particular purpose. The term metadata is typically described as 'data about data'. However, a more useful way to define metadata with respect to MSaaS is 'information about a resource (M&S information object, data or services)'. It should be noted that Metadata only provides detailed information about a resource i.e. it is not actually the resource.

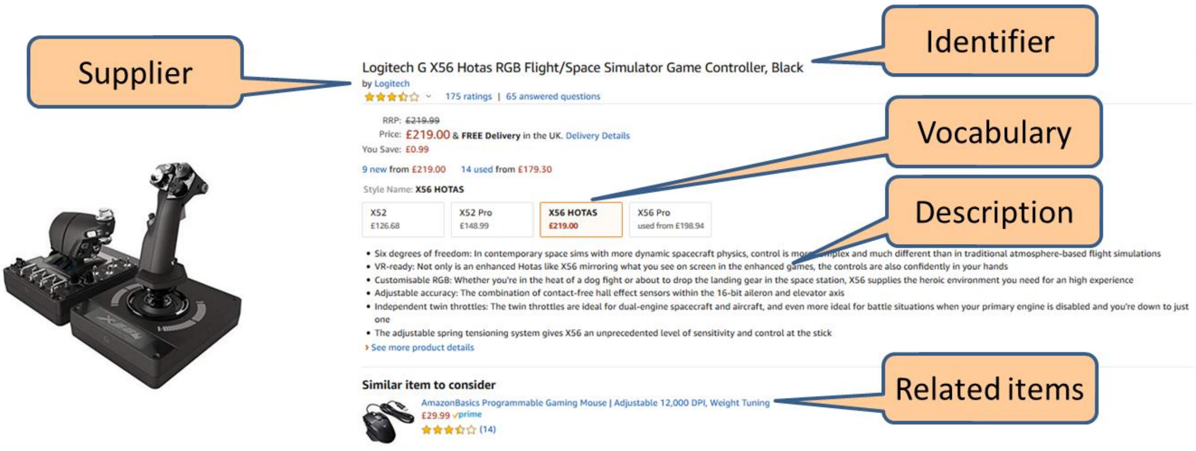

Image courtesy of Amazon

Figure 6: Example Metadata

Metadata is used frequently in many areas in supporting a user in determining if a resource is appropriate for their needs. This can include a typical Amazon entry as shown above.

In the geospatial community metadata has been widely used to capture information about charts and maps.

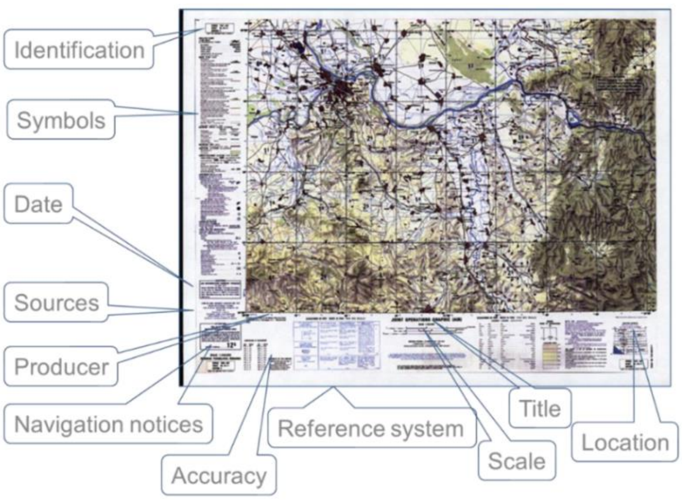

Figure 7: Topographical Map showing metadata

Such metadata as shown in the examples above are focused on evaluation by a human user; in a digital environment it is also necessary that metadata will be evaluated by computer to support automated and semi-automated selection of resources based on user needs.

The use of metadata as a means of understanding what resources are available to an end-user is a well-accepted approach in many domains and is considered to be a key facilitator in the re-use of resources by MSaaS. By providing descriptive information about a resource, metadata can facilitate data discovery and initial evaluation to support improved resource re-use.

4.2 Metadata Requirements

Metadata typically provides a variety of metadata categories to enable the end user to establish the most useful resource to support a particular role or task.

- **Discovery Metadata**: This is the minimum amount of information that needs to be provided to convey to the inquirer the nature and content of the data resources. It essentially answers the 'what, why, when, who, where and how' questions about resources.
- **Use Metadata**: This is the information required to access, transfer, load, interpret and apply a resource in the end application where it is exploited. This often includes details of a data dictionary, schema, classifications and other parameters that are useful to both humans and machines in enabling the proper use of the resources.
- **Management Metadata**: This is the information required to support the control of the data by managing agencies or infrastructure. Examples are original identifiers and source, runtime instances of simulations, and their compositions and deployments, as well as to track versions, distribution caveats, citations, etc.

Although the majority of metadata standards address most of the categories presented above, each domain may have a different profile that has been tailored to their particular needs. An example collection of domain specific metadata standards can be found in the Digital Curation Centre (DCC) web site[2]. Other examples can be found on Wikipedia[3] and other sites[4]

4.3 Metadata Standards

A key driver for the UK research was not to create a new metadata standard specific to MSaaS but to reuse and extend if necessary, existing standards. A quick look at potential metadata standards and profiles was performed to identify ones that best supports the needs of MSaaS.

Assessment was based on the following criteria:

- Support for the primary Dublin Core elements (see below);
- Extensible to address the future needs of the M&S community;
- A suitable encoding mechanism for machine interpretation.

These properties are considered essential for any metadata standard that is selected for use within an MSaaS ecosystem.

[2] http://www.dcc.ac.uk/resources/metadata-standards/list

[3] https://en.wikipedia.org/wiki/Metadata_standard

[4] http://www.dcc.ac.uk/resources/briefing-papers/standards-watch-papers/what-are-metadata-standards

Dublin Core is a widely used metadata standard to describe digital resources (video, images, web pages, etc.), as well as physical resources such as books or CDs, and objects like artworks[5]. The Dublin Core schema defines the elements typically used in the context of an application profile which constrains or specifies their use in accordance with local or community-based requirements and policies. However, it does not define implementation detail.

Dublin Core Metadata Element Set		
Title	Contributor	Source
Creator	Date	Language
Subject	Type	Relation
Description	Format	Coverage
Publisher	Identifier	Rights

Figure 8: Dublin Core Metadata Element Set

The UK research reviewed the following Metadata Standards:

- ISO 15836 – Dublin Core
- ISO 19000 series – Geographic Information – Metadata
- NATO Core Metadata Specification (NCMS)
- MoD Geospatial Metadata Profile (MGMP)
- UK GEMINI 2.2 Metadata Standard (GEMINI2)

ISO 19000 series was selected as the family of standards to be used in the MSaaS experiment. These satisfy the assessment criteria in that they support all the Dublin Core elements, they are formally extensible (see MGMP/GEMINI2) and have an XML encoding standard to support machine interpretation. It also aligns well with the NCMS, though no formal mapping exercise was undertaken. As well as the Dublin Core elements, the standards support sophisticated geographic elements, data quality, distribution information, and in the case of 19119, supports detailed descriptions of Services as resources.

4.4 Controlled Vocabulary

The UK concept has a core requirement to relate many of the concepts used within it back to authoritative reference sources such as controlled vocabularies, ontologies or similar data of software used to generate specific results sets. This is embedded into the model, with the concept of abstract entities existing as information objects within the ecosystem being an example of the use of this type of data to enable conceptual interoperability (i.e. that users and stakeholders can exchange objects knowing that they are referencing the same underlying information). This not only essential to achieve machine-to-machine communication, but also even human readability is limited if there is no way of understanding other interpretations of the concepts involved. For example, a service that provides a simulated aircraft may be useful for a range of purposes – but not if there is no way of formally specifying the behaviours, properties and level of fidelity that it provides such that others can evaluate whether it meets their own requirements.

[5] https://www.dublincore.org/

The concept of a Controlled Vocabulary is important to ensure that terms are used consistently in the MSaaS ecosystem. A Controlled Vocabulary provides a way to organize knowledge for subsequent retrieval. It is used in subject indexing schemes, subject headings, thesauri, taxonomies and other forms of knowledge organization systems.

The ISO 19115 standard supports extension to the terminology through the use of codelists and thesauri.

A codelist provides a constrained list of allowable values that provide unambiguous comparable definitions. Colloquially, we need to compare 'apples with apples'. An example of a codelist is the International Aid Transparency Initiative (IATI) Country Codes[6].

A more relevant example is the extension to the ISO19115 MD_KeywordTypeCode codelist defined by the National Oceanographic and Atmospheric Administration (NOAA)[7]. This shows how NOAA have extended the standard codelist to include new keyword types that are appropriate to their community.

This approach has been used to allow the MSaaS community to describe the resource using appropriate, relevant and consistent terms.

Thesauri augment the use of codelists by providing unambiguous definitions for words, phrases and concepts within a specialised domain. The EIONET General Multi Lingual Environmental Thesaurus[8] (GEMET) is a widely used thesaurus in the meteorological and geospatial domains. For example this defines an emergency shelter as being a building that is a 'Shelter given to persons who are deprived of the essential needs of life following a disaster'[8].

For the UK MSaaS research, the 'SISO Reference for Enumerations for Simulation Interoperability[9]' was taken as the starting point for the development of a simulation specific thesaurus.

5.0 REGISTRY

5.1 Overview

Metadata provides descriptive information about an individual resource that allows it to be evaluated to determine if the resource is fit for a specific purpose. Metadata also provides the core information to support search and discovery but does not directly implement search and discovery capabilities.

Search and discovery requires a catalogue or registry to collect, store, and to manage and interpret the metadata information for the disparate types of resources they describe. The UK research utilised the Envitia GeoRegistry product to provide a registry capability. This is an open-standards implementation of the Open Geospatial Consortium (OGC) CSW-ebRIM specification.

The following definition of a registry has been provided by the OGC CSW-ebRIM documentation

- A registry can be considered to be similar to a library.
- The repository is like the bookshelves in the library.

---

[6] http://iatistandard.org/202/codelists/Country/

[7] https://geo-ide.noaa.gov/wiki/index.php?title=ISO_19115_and_19115-2_CodeList_

[8] http://www.eionet.europa.eu/gemet

[9] https://www.sisostds.org/ProductsPublications/ReferenceDocuments.aspx

- The repository items in the repository are like book on the bookshelves. The repository items can contain any type of electronic content just like the books in the bookshelves can contain any type of information.
- The registry is like the card catalogue. It is organized for finding things quickly.
- A RegistryObject is like a card in the card catalog. All RegistryObjects conform to a standard just like the cards in the card catalog conform to a standard.
- Every repository item MUST have a RegistryObject that describes it, just like every book must have a card in the card catalog.

Importantly the ebRIM Registry can be extended to support new:

- Complex associations between resources;
- ClassificationSchemes – Represents a controlled vocabulary used to classify registry objects in some manner (e.g. ontology, taxonomy, thesaurus);
- ClassificationNodes – Represents a term or concept that augments an existing canonical classification scheme to define a new object type, association type, data type, and so forth; AdhocQueries – Represents a parameterized query definition that provides a simple means of searching registry content without requiring detailed knowledge of ebRIM or the syntax of a particular query language;
- AdhocQueries – Represents a parameterized query definition that provides a simple means of searching registry content without requiring detailed knowledge of ebRIM or the syntax of a particular query language;
- ExtrinsicObject – Describes a repository item that represents an information resource such as a specification document or an application schema.

Using these extension capabilities the UK research has further developed the underlying registry information model (RIM) to support the complex requirements of the simulation resources identified by the information modelling work.

5.2 MSaaS Registry Information Model

The Registry Information Model (RIM) is effectively a distillation of metadata into relational databases to enable the end-user to search, discover, manage and enhance the M&S resources at the appropriate level. At the most basic level where a user wants to simply search on names, abstracts and recover the raw metadata then no extensions to the core RIM are required. However, if the user needs to search based on relationships or specific object classifications then the RIM needs to be extended to support these.

An MSaaS RIM Extension Package was created to reflect the findings of the information modelling work to accommodate the identified use cases and key attributes of the M&S resources.

New objectTypes were developed to represent the top level M&S resource object types e.g. SimulationResources-ParticipantServices. Differentiation into object types allows the user to ask questions such as ‘Find all SimulationCompositions that have a name like ‘JTAC training’?’

All objects have a common structure based on the base objects (Identifier, RegistryObject and ExtrinsicObject). Object type specific data can be provided by use of slots. For example the SimulationEvent can hold an eventDate. This allows the user to ask questions such as ‘Find all SimulationEvents where the EventDate is between May 2012 and April 2013?’

ClassificationNodes have been added to support the controlled vocabularies discussed above. This enables the user to ask questions such as ‘What SimulationEntities are available that have been classified as a SISO Fixed Wing Aircraft?’ In this circumstance the term Fixed Wing Aircraft is unambiguous and has the definition provided by the SISO terminology.

Typed associations can provided information on how M&S Resource are related to each other. This allows the user to ask questions such as ‘What SimulationDeployments have been created to support the SimulationComposition with the name MSG136-Comp1?’ or ‘What ParticipantServices have been used by the SimulationDeployment with the name ICOVICS-UKCloud?’ This extends the search and discovery capabilities of the Registry beyond a list of simple object types.

6.0 CONCLUSION

The complexity of the discovery capability required to discovery capability required to promote the reuse of simulation resources in MSasS is related to the amount of data that is to be managed. For low numbers of resources, as found in a bounded MSaAS implementation, this can be provided by the search function in a simple database or even a spreadsheet. When MSaaS is implemented across an Enterprise, a more sophisticated implementation is required based on the use of a Registry. The specification of an MSaaS information model and implementation in a registry is a very specialised topic and requires people with the required skills to implement it.

Development of the MSaaS Discovery process is still continuing and the following challenges need to be resolved to implement a sophisticated discovery capability:

- Champion/funding to maintain Registry;
- Human and machine discovery;
- Discipline for creating metadata;
  - Requires to be (semi)-automatic
- Information should be accurate and up to date;
- Approach needs to be scalable;
- Ability to federate registries;
  - Mapping metadata between different registry implementations.

7.0 ACKNOWLEDGEMENTS

The author would like to acknowledge the contribution made by Louisa Stewart (SEA), Brendan Mason (Envitia) and Louise Simpson (Thales) with developing the approach to MSaaS Discovery for Dstl in the UK.