Work-Centered Decision Support

Dr. Michael J. Young
Air Force Research Laboratory
Human Effectiveness Directorate
2698 G Street
Wright-Patterson AFB, OH 45433
USA

Dr. Robert G. Eggleston
Air Force Research Laboratory
Human Effectiveness Directorate
2255 H Street
Wright-Patterson AFB, OH 45433
USA

SUMMARY

The ongoing revolution in information technology is creating digital nervous systems which connect all of an enterprise's information assets together into an information grid. Accessing data no longer requires one to connect to several specific systems in serial fashion; instead, one plugs into a grid. The development of digital nervous systems is making more and more information available to decision-makers in real-time. The sheer volume of information threatens to overwhelm users. Surprisingly, there has been very little research into how this revolution will affect the human-computer-interface (HCI). The Air Force Research Laboratory has pioneered the development of a new class of job performance aid, called the Work-Centered Support System (WCSS). A WCSS is a stand-alone interface client that shifts the HCI paradigm from the desktop metaphor to a work support metaphor. Traditional displays and controls, to include current user-centered HCIs, are machine- or system- focused, providing direct connections to machine resources and direct manipulation support for application tools. Further, a traditional HCI provides access to, or is interfaced to, only one system. Users must often log into and out of multiple systems to find the data they need in order to make decisions or perform a task. This is an increasingly clumsy set up that wastes time, increases cognitive load, and can lead to mistakes. In contrast, a WCSS is work focused at the scale of work-threads, which encompasses planned tasks, interruptions, and opportunistic behavior. It represents the user's problem workspace, and acts to amplify and facilitate user decision-making, problem solving, and overall work performance. In addition, machine automation is exploited in the form of intelligent agents that can work as team players in tandem with the user. This includes the ability to dynamically plug into the information grid and to find, fuse, format, and present information to the user in a manner relevant to the current context. In short, a WCSS allows the interface to make best advantage of the greatly increased volume of information. In this paper, we discuss the revolution in information technology that is enabling the WCSS work aid, and illustrate the WCSS concept through examples.

INTRODUCTION

Information systems are evolving from stand-alone applications having their own data sets and sources into distributed networked applications which share data in real-time. This ongoing evolution in information technology is creating digital nervous systems which connect all of an enterprise's information assets together into an information grid. Accessing data no longer requires one to connect to multiple systems in serial fashion; instead, one plugs into a grid. The development of digital nervous systems (e.g., Enterprise Resource Planning, Business Enterprise Models, Network Centric Warfare, and Joint Battlespace Infosphere) is making more and more information available to decision-makers in real time.

Paper presented at the RTO HFM Symposium on "The Role of Humans in Intelligent and Automated Systems", held in Warsaw, Poland, 7-9 October 2002, and published in RTO-MP-088.

This evolution of information systems is ongoing and consists of several development trends. An early trend was the development of enterprise resource planning (ERP) systems, like those pioneered by SAP (the German software company). ERP systems are complex sets of interlinked software applications that use a unified database to share data. More recently, there has been a trend to develop enterprise application integration (EAI) software. This class of software connects machines to other machines (as opposed to linking applications through a data base), and lets businesses embed business process information and procedures into the information flow. In addition, EAI software often comes with graphical tools, which enable management information system professionals to quickly reprogram and change the flow and transformation of information within the networked software as situations dictate.

In addition to the development of specialized software, there has been extensive development of web-enabled software, including the development of information exchange standards such as the Hypertext Markup Language (HTML) and Extensible Markup Language (XML). Such standards make it relatively easy for disparate computer systems and applications to share data. The result of these developmental trends is that more and more information is becoming available to users in real time.

How has the evolution of networked information systems affected the human computer interface (HCI) and what type of new HCI capability has been created to help users deal with the tremendous increase in information? Real-time enterprises usually opt to provide users “digital dashboards”, so called because the display changes in real-time in a manner similar to an automobile display. A digital dashboard can be thought of as a spreadsheet that displays continually updated enterprise information. Individual cells within the spreadsheet show the status of a specific metric of interest to the user. Different cells are often linked by macros (i.e., sub-routines), which integrate and transform data.

Figure 1 shows the Integrated Management Tool (IMT) employed by Flight Managers (FM) at the United States Air Force Tanker Airlift Control Center (TACC) to command and control air mobility flights (i.e., the movement of cargo and personnel). The IMT provides real time data on planned and ongoing missions, and is an example of digital dashboard technology. Figure 2 is a screen shot of one of the IMT windowpanes. Columns within this pane provide real-time information on mission numbers, call signs, aircraft type, tail number, departing airfield, arriving airfield, estimated time of arrival, and so on. The full IMT currently has eighty plus columns of data. Individual cells are continually updated, as new data enters the enterprise system. If a specific parameter goes out of bounds, say an aircraft is delayed, the appropriate cell changes color alerting the FM of the potential problem.

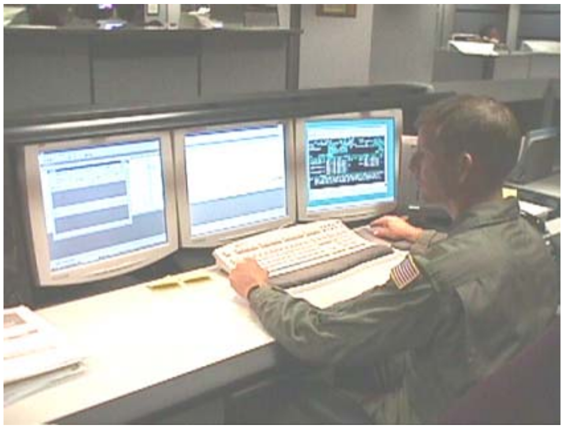
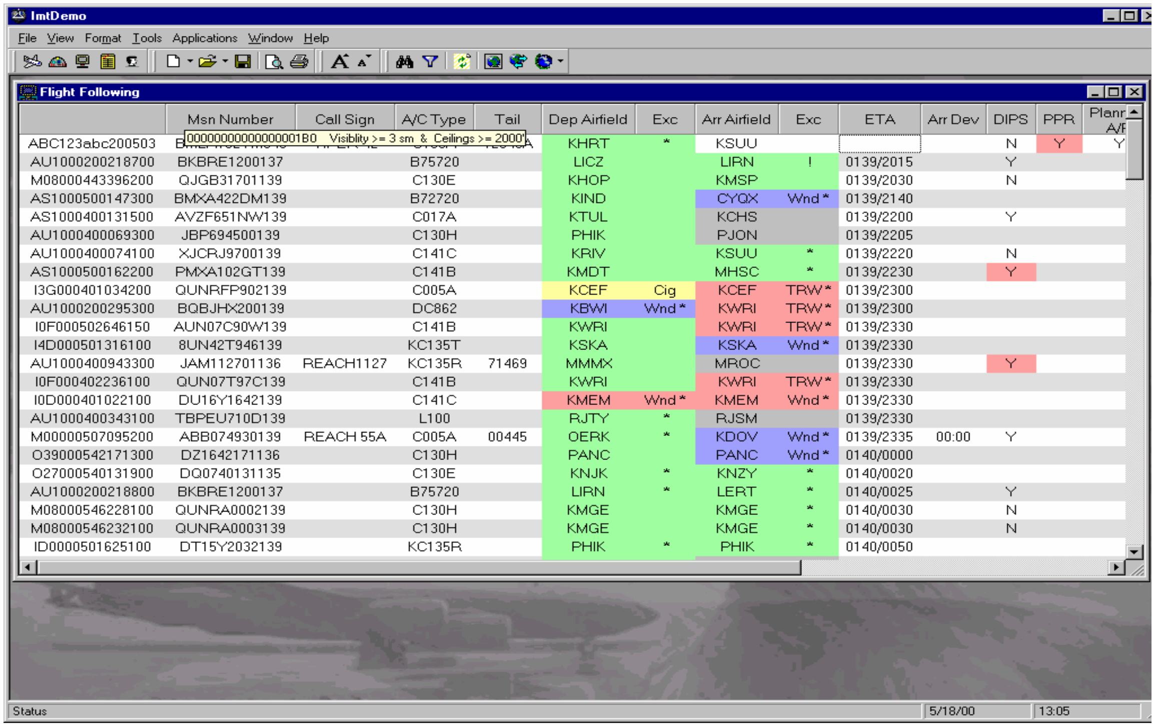


Figure 1: Integrated Management Tool.

The IMT is an amazing engineering achievement that highlights the power of web portal technology. It serves as a user client that brings together data from many divergent sources, organizes its presentation within a work-oriented framework, and provides ready access to a variety of work tools. Prior to its deployment, FMs had to search multiple databases to find and retrieve information about aircraft status. Further, there were not any means to automatically notify FMs when a value had gone out of bounds. (They were typically notified via a phone call well after the problem occurrence). Now, the FM needs only scan the IMT spreadsheet to see if there are problems.



00000000000000001B0 Visiblity >= 3 sm & Ceilings >= 2000'

	Msn Number	Call Sign	A/C Type	Tail	Dep Airfield	Exc	Arr Airfield	Exc	ETA	Arr Dev	DIPS	PPR	Planr A/F
ABC123abc200503	B[illegible]	[illegible]	[illegible]	[illegible]A	KHRT	*	KSUU				N	Y	Y
AU1000200218700	BKBRE1200137		B75720		LICZ		LIRN	!	0139/2015		Y		
M08000443396200	QJGB31701139		C130E		KHOP		KMSP		0139/2030		N		
AS1000500147300	BMXA422DM139		B72720		KIND		CYQX	Wnd *	0139/2140				
AS1000400131500	AVZF651NW139		C017A		KTUL		KCHS		0139/2200		Y		
AU1000400069300	JBP694500139		C130H		PHIK		PJON		0139/2205				
AU1000400074100	XJCRJ9700139		C141C		KRIV		KSUU	*	0139/2220		N		
AS1000500162200	PMXA102GT139		C141B		KMDT		MHSC	*	0139/2230		Y		
I3G000401034200	QUNRFP902139		C005A		KCEF	Cig	KCEF	TRW *	0139/2300				
AU1000200295300	BQBJHX200139		DC862		KBWI	Wnd *	KWRI	TRW *	0139/2300				
I0F000502646150	AUN07C90W139		C141B		KWRI		KWRI	TRW *	0139/2330				
I4D000501316100	8UN42T946139		KC135T		KSKA		KSKA	Wnd *	0139/2330				
AU1000400943300	JAM112701136	REACH1127	KC135R	71469	MMMX		MROC		0139/2330		Y		
I0F000402236100	QUN07T97C139		C141B		KWRI		KWRI	TRW *	0139/2330				
I0D000401022100	DU16Y1642139		C141C		KMEM	Wnd *	KMEM	Wnd *	0139/2330				
AU1000400343100	TBPEU710D139		L100		RJTY	*	RJSM		0139/2330				
M00000507095200	ABB074930139	REACH 55A	C005A	00445	OERK	*	KDOV	Wnd *	0139/2335	00:00	Y		
O39000542171300	DZ1642171136		C130H		PANC		PANC	Wnd *	0140/0000				
O27000540131900	DQ0740131135		C130E		KNJK	*	KNZY	*	0140/0020				
AU1000200218800	BKBRE1200137		B75720		LIRN	*	LERT	*	0140/0025		Y		
M08000546228100	QUNRA0002139		C130H		KMGE		KMGE	*	0140/0030		N		
M08000546232100	QUNRA0003139		C130H		KMGE		KMGE	*	0140/0030		N		
ID0000501625100	DT15Y2032139		KC135R		PHIK	*	PHIK	*	0140/0050				

Figure 2: Integrated Management Tool Screen Enlargement.

As powerful as the portal and digital dashboard technologies are, we argue that they do not go far enough in utilizing the capabilities that digital nervous systems provide. The development of digital nervous systems provides an opportunity to reconceptualize and redesign the human computer interface. In this paper, we describe our research to create a new class of job performance aid that we call the Work-Centered Support System (WCSS). A WCSS is a stand-alone interface client that changes the HCI paradigm from the desktop metaphor to a work support metaphor. We begin by discussing traditional interfaces, then introduce the WCSS concept from a software perspective, and describe it from a human computer interface perspective, showing how it facilitates the direct support of work better than traditional interfaces do. We conclude by discussing some of the issues associated with creating WCSS technology, to include the role of agents in the interface and required extensions to cognitive task analytic techniques.

WORK

Let us begin by considering a traditional stand-alone application program. Every application program has a separate human computer interface which enables the operator to utilize the system's functionality (e.g., to search a database for some specific information, or generate a document). Further, the interface, per se, is designed around a tool metaphor: The system is an intricate tool and the interface is how one controls the tool. As a simple example, consider a word processing program. Menus provide access to the program's functionality. Specific operations or utilities (e.g., opening a file, inserting a table) are grouped by function. File operations are a part of one menu; table operations are part of another, for instance. The user employs the tool (i.e., application program) to create a product (e.g., a conference paper).

Now consider work. As an example we will use an abstracted description of the flight planning function performed by a FM. In addition to monitoring missions, FMs also plan missions and produce mission papers for the flight crews. At the start of the shift, a FM receives a list of missions to plan. For each mission, they must request an aircraft from the "barrel" function (i.e., an organization that assigns aircraft to missions) and request cargo weight information from the unit being moved. They also have to propose and get approval for a flight plan from the appropriate air traffic control agencies. Once they have this information (aircraft type & cargo weight) and flight plan approval, they can notify the wing which will fly the mission of its planned start time, calculate the fuel requirement, request a weather forecast for the mission route, and begin generating mission papers. Several different information sources may need to be consulted, and data may need to be checked and rechecked as rough estimates are refined and new situations develop. Once all required information has been collected and route-planning decisions made, a set of mission papers are generated and sent to the wing which will fly the mission. The whole process usually takes a few hours, but it can become more protracted and involve considerable collaborative interaction. A FM is normally planning several missions simultaneously.

Planning a mission is an example of a *work- thread*: a connected series of work activities directed toward accomplishing a goal. Each activity in the work-thread usually consists of multiple steps. Some steps are purely cognitive; others involve actions through multiple modalities. Work activities generally require the user to frequently mentally shift frames of reference, for example, from mentally formulating a course of action using stored cognitive knowledge about typical flight plans, to searching a data base by invoking knowledge of passwords to gain access and system commands to find and retrieve data, to then accessing additional data or requesting some service or approval (e.g., Diplomatic Clearance) from a different system. These mental shifts significantly tax human cognitive abilities, such as short-term memory, and are probably the point where human error is most likely to occur. One of our goals in creating new class of job performance aid is to keep the user more focused on the task at hand (e.g., planning a mission) by reducing the time they are off task thinking about how to locate information and integrating data mentally into situational awareness.

How does a contemporary human computer interface directly support the user in accomplishing his/her work? It usually does not. Nearly all human computer interfaces are designed to provide access to information and general work tools, but they are not conceived of as applications that center on the evolving nature of work. Once again, work typically consists of a variety of work-threads, sets of activities directed towards specific goals. Further, in complex jobs, a user is frequently required to be engaged in several work-threads simultaneously and is frequently interrupted at inopportune times by telephone calls, urgent emails, or individuals appearing at his/her desk. Traditional interfaces deigned around the desktop metaphor do not provide direct support for work in this dynamic and to some extend unpredictable work situation. They do not help the user manage the manifold ongoing work-threads, nor do they help the user keep track of where they are in a specific work-thread. With a traditional interface, the user is forced to manage work via their short-term memory, the use of external aids such as post-it notes, and by creatively using the tools provided in the tool kit. (See figure 1).

THE WORK-CENTERED SUPPORT SYSTEM CONCEPT

The opportunity exists to radically reengineer the human-computer interface to directly support work. Our approach to achieving this is to create what we call a Work-Centered Support System (WCSS). A WCSS is a stand-alone interface client that provides multiple forms of support (e.g. decision support, product development support, collaborative support and work management support) within an integrated work-centered framework. Support is provided using both direct aiding, through the use of intelligent automation, and indirect aiding, through the use work centered frames and data forms. Both the direct aids and the structural form properties of the interface must be suitably coordinated to achieve the twin goals of reducing cognitive complexity while maximally supporting flexible problem understanding and action taking. We have approached this task by developing intelligent interface agents that meet the dictum of being “team players” (Roth, et al., 1997) and embedding them in work-centered frames and data forms (Eggleston, et al. 2000; Young et al., 2000). In current instantiations, we have used intelligent agents to automatically find, fuse, format, and present information. In this section we describe the WCSS from a software perspective. The next section discusses design considerations.

A WCSS can be depicted as a software client having three layers or components. Each layer employs intelligent agents to accomplish a specific characteristic of work support. The back-plane layer consists of acquisition agents which embody knowledge of how to find and retrieve data. Their function is to automatically monitor and access data sources for the user. In an enterprise where all data is available through a digital nervous system, the agents only need knowledge of one system, including system ontology, details of the communication language, and appropriate transportation protocol. Regrettably, most enterprises today are not fully integrated into a digital nervous system, they are somewhere between stand alone and fully integrated data systems. As a result, in a typical WCSS the acquisition agents need to embody a variety of data retrieval knowledge (e.g., how to access and read a web page, generate SQL commands to retrieve data from a relational database, and publish and subscribe to a data network). Acquisition agents notify other agents when new data has been retrieved or received.

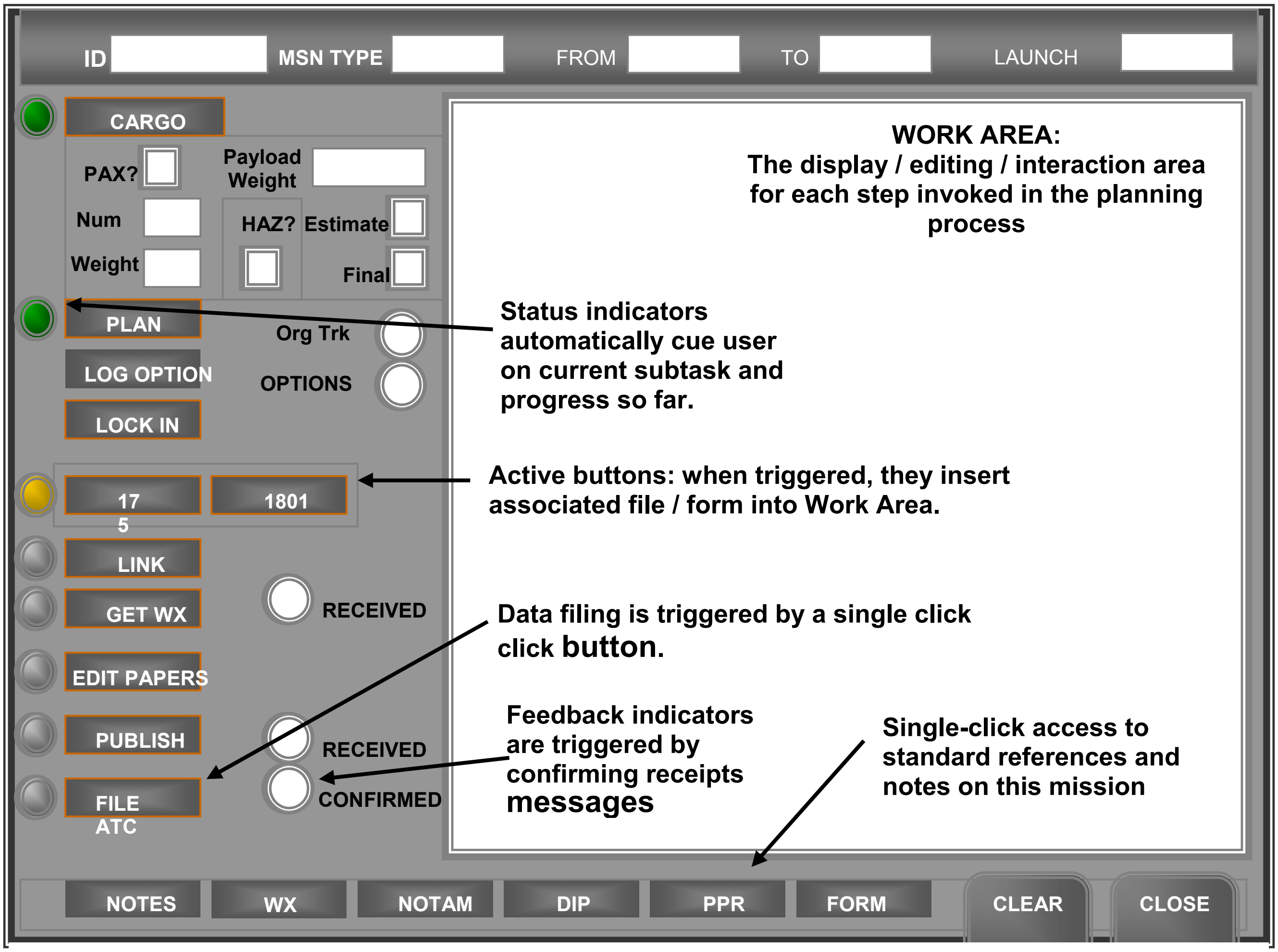
The mid-plane layer consists of analysis agents which embody the knowledge required to turn data into actionable information. Their function is to reason over data and fuse data in support of work activities. There is substantial evidence that human experts possess schema, or complex memory structures, that enable them to perceive patterns in the environment and which link those patterns to specific actions that must be performed. Without a WCSS, in order to create the situational awareness needed to trigger schema the human user must search multiple data systems to identify the pieces of the puzzle and then use his/her short-term memory and external memory aids (e.g., post-it notes) to fuse the pieces into a comprehensive situational assessment, which then activates the appropriate behavioral schema. Within a WCSS, analysis agents operate to create the patterns of information the user needs to assess the current situation. They use the data collected by the acquisition agents to continually appraise the situation, proactively identifying possible problems, and dynamically generating a prioritized list of potential user work actions.

Front-plane, or presentation, agents embody knowledge on how to employ graphical user interface (GUI) elements. They control what information is on the screen, respond to user requests, and alert the user to potential problems and opportunities. They know how to aggregate or disaggregate information based upon who the user is and what are his/her current requirements. Further, actionable information is made available in the user’s ontology and may be contextualized in different ways to accommodate a variety of work approaches adopted by a specific individual. In addition, front-plane agents are similar to back-plane agents in that they interface to external software, but in the case of front-plane agents the software is usually commercial GUI building software.

DESIGN CONSIDERATIONS

In creating WCSS technology, another of our goals is to create ecological interfaces. From our perspective, an ecological interface must provide user support on two dimensions. First, it must serve as a memory aid, allowing the user to relieve short-term memory of the burden of remembering the status of a given activity. Concurrent with this function, it must provide situational awareness for a work-thread or task, enabling the user to quickly pick the activity back up and immediately discern any potential problems. Second, it must afford action; it must make it readily apparent what action is required and what are the constraints on action for any specific task, at any point in time.

Figure 3 is a proposed FM planning display from a design study by Whitaker and Eggleston (2001). It is used in conjunction with a mission palette (not shown) which provides status information on all missions, those that are ongoing, those being planned, and those that will be planned later.



WORK AREA:
The display / editing / interaction area for each step invoked in the planning process

Status indicators automatically cue user on current subtask and progress so far.

Active buttons: when triggered, they insert associated file / form into Work Area.

Data filing is triggered by a single click click button.

Feedback indicators are triggered by confirming receipts messages

Single-click access to standard references and notes on this mission

Figure 3: Flight Planning Display.

To work on a specific mission, the FM selects it from mission palette by clicking on it. Intelligent agents then populate the data fields of the display shown in figure three by locating all mission relevant information. The display provides immediate situational awareness to the FM. The information along the top summarizes the mission, while the information along the left side provides status of the activities or tasks that make up the mission planning work-thread. A FM can tell at a glance the work state for this mission. The display thus relieves the FM of the cognitive burden of remembering where he/she is in the planning process for a specific mission. Because each mission is contained in a separate work pallet, the FM is able to quickly discern his or her total mission planning work state during a shift. This ability to directly see the work situation is particularly valuable at shift change over time, when a new FM is taking over partially completed work. Currently, both the out-going and in-coming FMs have to spend considerable time getting up to speed on the status of operations and the open work requirements as part of the new shift work. In the future, when this display is fully implemented, that time will be substantially reduced.

The flight planning display illustrates five main design principles for WCSSs described by Eggleston, Young, and Whitaker (2001):

- Use a work domain ontology as the organizing framework across the decision support tool set
- Maximize explicit reference to task domain elements in on-screen information display
- Provide tailored and context-sensitive support
- Maximize effective fusion of data from multiple databases that need to be consulted to complete work tasks
- Minimize perceptual, cognitive, and motor task demands associated with identifying, seeking, or interpreting relevant information and producing work artefacts

While the terminology displayed on the flight planning display may seem arcane to outsiders, it is the terminology used by the flight planners; it matches their existing frame of reference or ontology. The screen also brings together in one place all the elements of the task domain; the FM does not need to access any other computer or data source to plan a mission. All of the data, which actually resides in several systems, is retrieved automatically and fused into an appropriate format by intelligent agents. This automatic retrieval and fusion minimizes the user's cognitive burden by eliminating the need to shift frames of reference, from a planning mode, to a data collection mode, and back, keeping the user focused on the task at hand.

While this display has not yet been fully implemented by the TACC, components of it have been implemented in the Integrated Management Tool. Our next example has been implemented in the TACC in prototype form. We will describe it to highlight some of the functionality that reside not in the GUI the user sees, but with the acquisition, analysis, and presentation agents that comprise the nucleus of software processing in the WCSS.

As mentioned above, a second duty of the FM is flight following, monitoring ongoing missions for problems. One class of problems they have to deal with is the effect of changing weather on ongoing and planned missions. For example, at different times of the year the jet stream can move into a flight path providing a significant head-wind of 200 or more miles per hour. On a long flight this can result in slowing the aircraft sufficiently so that it no longer has enough fuel to complete the mission. In addition, unexpected thunderstorms and severe turbulence can also be a problem for an aircraft planning to land. Prior to the development of the WCSS for Global Weather (WCSS-GW; described below), the FM would typically only find out about such weather issues after they had become a problem for a specific mission. The problem was not that the data was not available; it was that the necessary information was spread among several systems and that there was not any way to easily fuse the data and determine the potential impacts. One of the goals in developing the WCSS-GW was to provide the TACC a way to proactively identify potential weather impacts, well before they become a problem for specific missions. The FM is assisted within the TACC by a Weather Officer (WXO). A WXO's responsibilities include generating mission forecasts for missions, preparing general forecasts for world regions of interest, maintaining global situation awareness of weather (and its potential impacts), and working with FMs to determine appropriate responses when adverse weather affects missions (e.g., delay a flight, reroute it, make changes to planned fuel load or cargo).

The data that the WXO uses are spread among several systems. Some of these systems are Air Force owned, others government owned, and others university owned. The WXO accesses this data through a computer by going to various web sites and by receiving automatic electronic updates from some of the systems. These data are then integrated into a weather picture mentally by the WXO and by drawing select details on a map. The primary cognitive challenges facing the WXO are integration and interpretation. Data are often ambiguous, and data from different sources are often conflicting. Further, the WXO is usually trying to predict future weather from these ambiguous, conflicting data: A very challenging task. Good decisions can save thousands of dollars and precious time and crew resources.

The WCSS-GW was developed to support the WXO and FM[1]. It provides a variety of job performance aiding techniques to help them make decisions, generate products, collaborate, and manage work. Acquisition agents within the WCSS embody knowledge of how to access weather related data. Some of the agents are capable of going to a university web site to retrieve data, others know how to request flight plans from the advanced computerized flight planning system, and still others know how to receive and process electronic updates that are sent from air traffic control systems and other corporate data bases. There is one agent for each data source. The data they retrieve includes published weather alerts, pilot reports (PIREPS), wind and turbulence from the Aircraft Communication Reporting System (ACARS), Significant Meteorological Information bulletins (SIGMETS), satellite imagery (providing cloud and wind data), meteorological observations at airfields, lighting strike reports, planned aircraft routes and refueling orbits, and status information on ongoing missions.

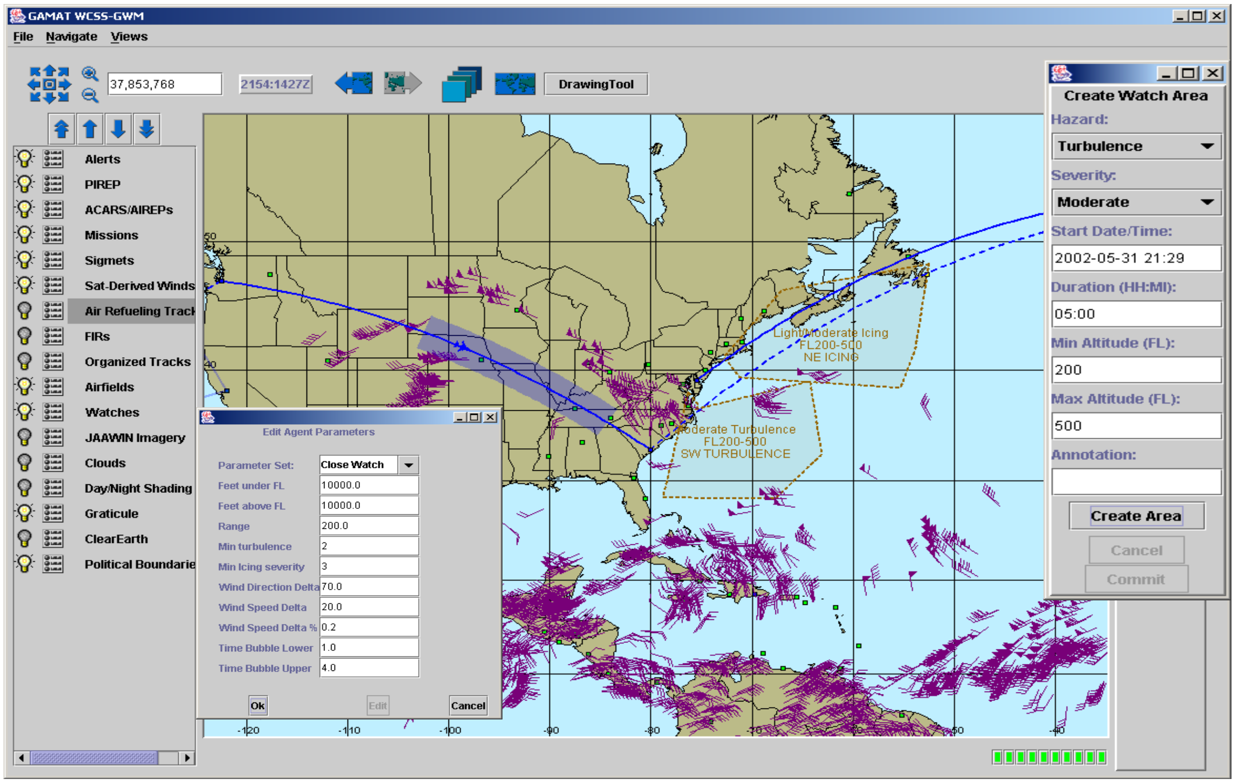
Analysis agents embody knowledge on how to fuse weather information, both geospatially and temporally. There are three types. Forecast analysis agents are created when weather forecasters are developing a forecast; they look for observations that support the forecast. Region analysis agents are created when a WXO decides to monitor a specific geographical area; they watch for weather observations that match criteria defined by the WXO. Mission analysis agents are automatically generated by planned missions; they watch for weather reports that would significantly affect a mission.

Presentation agents influence or control GUI elements. They compose display features and respond to user requests. Their principal function is to fuse disparate data sources together within a display. They use an open-source geospatial display toolkit called OpenMap to present information. OpenMap is a sophisticated programmer's toolkit that enables a developer to quickly build applications and Java applets that access and display data from legacy databases and applications. It provides the means to enable users to see and manipulate geospatial information.

The main screen of the WCSS-GW is shown in figure 4. On the left hand side of the figure are listed the types of weather and other information of interest to the WXO. Clicking on a topic results in the information on that topic which is applicable to current geospatial coordinates being displayed on the map. The user can choose to display as many or as few items of interest as desired; a lit light bulb depicts a "turned-on" information

[1] The WCSS-GW is being developed for AFRL by BBNT. See Scott, Roth, Malchiodi, Deutsch, Kazmierczak, Eggleston, Kuper, and Whitaker, 2002 for additional information on the project.

source. The map display is fully controllable by the user. The controls along the top allow the user to change view, pan, zoom in or out, etc. Several different map projection schemes are available. As the user changes the map view, agents update it appropriately with whatever information is “turned on.”



Edit Agent Parameters

Parameter Set: Close Watch
Feet under FL 10000.0
Feet above FL 10000.0
Range 200.0
Min turbulence 2
Min Icing severity 3
Wind Direction Delta 70.0
Wind Speed Delta 20.0
Wind Speed Delta % 0.2
Time Bubble Lower 1.0
Time Bubble Upper 4.0

Create Watch Area

Hazard: Turbulence
Severity: Moderate
Start Date/Time: 2002-05-31 21:29
Duration (HH:MI): 05:00
Min Altitude (FL): 200
Max Altitude (FL): 500
Annotation:

Figure 4: Work-Centered Support System for Global Weather Main Screen.

This display greatly reduces the workload and cognitive burden on the WXO. Instead of the WXO having to go to different systems and manually retrieve data and then attempt to fuse it either in his/her head or on a map, the system does all of that automatically, greatly improving the WXO’s weather-related situational awareness. In addition, it affords decision support by enabling the officer to quickly look for converging evidence from multiple sources, and it enables the user to dramatically increase his/her time on task, answering weather related questions.

Next consider how the display supports work management and collaboration. When a mission is planned, an analysis agent is automatically spawned to monitor the flight. This agent requests an acquisition agent to go to the flight planning system and retrieve the appropriate flight plan. The analysis agent then defines a “spatial tube” around the mission that extends ahead, behind, above, and below the aircraft, and moves with it once it takes off and begins its flight. (See the middle of Figure 4 for a depiction of a tube). The analysis agent monitors the weather within the tube by subscribing to weather updates from acquisition agents for the area it is watching. It primarily looks for any deviations from forecasted weather, but also looks for significant meteorological events. The boundaries of the tube have default values, but are also adjustable by the user. In Figure 4, the Edit Agent Parameters Screen shows where the user would change the default values of the “tube” to other values.

This automatic monitoring and data fusion aspect of the system enables both work management and collaborative support. One problem for the WXO is what to focus on next. Prior to the WCSS-GW there was not at easy way to see if any missions were going to encounter weather-related problems. The WXO would have to scan the various data sources, build up an awareness of where there were potential problems, and then go check with the FM to see if any missions were flying into those areas within the relevant time window. Because of the time required to carry out these steps, this rarely was done; the WXO focused on other tasks such as regional forecasts and forecasts for planned missions. The result was that the TACC was in a reactive mode. The FM and WXO would respond to problems when radioed in by a flight crew. With agents now automatically monitoring weather along the flight paths and alerting the WXO to potential problems, they can both better manage their time. They are alerted to potential problems, usually well in advance, and can schedule time to work problem resolution with minimal impact to their ongoing work activities. Further, prior to the development of the WCSS-GW, there was not a good way to visualize flight path and weather information together. The FM and WXO used different displays. The WCSS-GW enables them to share a display which integrates flight path, weather, and base information enabling them to better communicate and to more quickly devise a solution, often before the problem occurs.

Finally, the WCSS for Global Weather also enables the user to create custom watch areas. The palette on the right of figure 4 enables a user to create an intelligent agent to monitor for specific weather occurrences such as high turbulence or lightning. The user first clicks on Create Area and draws an area on the map. The user then fills in information on the palette, to include the temporal duration of the watch area, the altitudes that will be watched, and the type of weather phenomena that will be monitored. Clicking Create Area triggers the instantiation of a new analysis agent that will monitor the area. This component of the WCSS-GW provides product development support by eliminating work steps and provides decision-support by identifying and fusing converging information, enabling a WXO to more quickly develop a region forecast or achieve situational awareness.

RESEARCH ISSUES

The two most important research areas for the future development of WCSS are the role of agents in the interface and improving the creation of analysis agents by developing extensions to cognitive task analysis techniques. An intelligent agent is a software thread of execution that accomplishes a specific task. This thread of execution runs within the WCSS client process and has computational primitives that enable it to send and receive data to and from other agent threads (including those that interface to the "external world"). A major line of the research on the role of intelligent agents within a human computer interface focuses on providing them a persona so they can be better "social actors" with which the user can interact. We have deliberately chosen not to display the agents as social actors, opting instead to embed them within a direct manipulation framework. This reflects our design goals to use a work domain ontology as the organizing framework for work support and to maximize explicit reference to task domain elements on screen. By building the GUI around domain objects, the intelligent agents naturally fall into the background. The user may be aware there are intelligent agents operating, but the user's focus remains on the work-thread and task elements, and not the social aspects of the agent.

We believe that the introduction of social agents could potentially reintroduce unnecessary mental shifts into work, as the user shifts from focusing on the task to "socializing" with the agent. There is some support for this position from the research literature. Milewski and Lewis (1997) note that the use of agents as the interface (as opposed to having behind the scenes agents) usually involves a delegation model, where the user delegates activities to the agents. Delegating activities to agents (or other humans) requires that the delegator assess the agent’s competency, communicate the desired outcome and possible strategy, and monitor work progress. All of theses activities require types of knowledge other than domain task knowledge. We suspect this would introduce unnecessary shifts in focus away from the task at hand as the user endeavors to activate and employ “social knowledge” to commission the agent.

We believe that a more important concern than agent persona is observability of agent activity. A danger of employing proactive agents is that common ground with the user may be disrupted. The classical case occurs when agent actions effectively change a mode that causes the support system to operate in a surprising manner. While front-plane agents in our current WCSSs are visible as work features, their parameters, while assessable, are generally hidden once set. As a result, a worker might forget these settings, especially if many agents have been deployed, and thus form an incorrect mental model of agent-based support. If various setting are used for a single type of agent, it may be necessary to augment the work-centered expression of that agent class to insure visibility of this meta-level information. The issues of agent persona and observability are empirical questions that warrant additional research.

With respect to task analytic techniques, one of our design goals is to craft WCSSs that are congruent with the user’s cognitive model of work. Research has shown that skilled decision-makers recognize situations; that is, they have developed schemas of typical situations that are quickly activated by environmental cues (Klein, 1999). These schemas are then tuned to the specific situation through rapid mental analysis and/or simulation. Ideally, we would like to have task elicitation techniques that anyone could apply to identify the schema and cues used by skilled decision-makers. We could then design the analysis agents to look for these patterns and have the presentation agents present a rank ordered set of potential problem situations to the user. Currently, without a WCSS, the users have to manually search for cues, which they then integrate mentally. Automating this process has the potential to greatly improve decision-making quality and reduce human errors.

There is a vast range of knowledge elicitation techniques. Cooke (1994), in a review article, documents fifty-six separate techniques. Unfortunately, theses techniques by themselves are not necessarily very good at eliciting and codifying a skilled decision-maker’s knowledge in a manner that would easily support the creation of more sophisticated analysis agents, agents which could identify the patterns that underlie schema, as opposed to monitoring for exception conditions like the WCSS-GW, for example. Part of the problem is that different users employ different ontologies. As one example, the term “mission” has three distinct uses within the TACC. For some users it is the first leg of a journey, say to a refueling track; for others it is the time until an aircraft first lands after having taken-off; for others, it is the time until the aircraft returns to its home station. Mapping out and codifying ontologies is a difficult problem due to the context specificity of most terms. We are not aware of any techniques that have solved this problem. Much research remains to determine which techniques are best at eliciting knowledge of schemas, and then determining how to best map this knowledge into analysis agents to maximize work support.

CONCLUSION

Information systems are evolving into digital nervous systems which tie all of an enterprise’s information into an information grid. As a result of this evolution in information systems, the human computer interface is becoming decoupled from application programs, and is evolving on its own into a stand-alone interface client. Currently, such clients are typified by web browsers which enable a user to “plug-into” application programs remotely. The web browser plugs into the web and provides a “fixed interface pane”, whose content changes with changes in the application software. More recently, there have appeared digital dashboards that make a vast amount of data available to the user in real-time. These are all examples of relatively thin clients, software that possesses minimal sophistication and complexity.

We have argued that the next step in the development of human-computer interfaces is the development of high-powered clients, what we call WCSSs, that will employ intelligent agents to find, fuse, format, present information, and respond to user requests. A WCSS changes the nature of the interface from accessing and managing information, via the desk-top metaphor, to work support. Work support entails providing decision support, product development support, and work management support at the level of work-threads. The primary conveyor of support is the ecological interface which provides situational awareness of the work-thread and affords action by making readily apparent what action is required and the constraints on that action.

The creation of WCSSs will dramatically alter the process of designing and building human computer interfaces. The human computer interface will become a stand-alone system, a user-interface client. The design process itself will no longer focus on the functionality of the application program and how a user might exploit it, but rather on the nature of the user's work and how the digital nervous system can support it. This reversal in focus promises to greatly improve the productivity of the user by designing and developing systems that are congruent with his/her cognitive processes and work style.

REFERENCES

Cooke, N.J. (1994). Varieties of Knowledge Elicitation Techniques. *International Journal of Human-Computer Studies*, 41, 801-849.

Eggleston, R.G., Young, M.J., & Whitaker, R.D. (2000). Work-Centered Support System Technology: A New Interface Client Technology for the Battlespace Infosphere. *Proceedings of NEACON 2000*, Dayton OH, 10-12 October, 2000, pp. 499-506.

Eggleston, R.G. & Whitaker, R.D. (2002). Work Centered Support System Design: Using Organizing Frames to Reduce Work Complexity. *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, Baltimore, ME, September 23-27, 2002.

Klein, G. (1998). *Sources of Power: How People Make Decisions*. MIT Press. Cambridge, MA.

Milewski, A.E. & Lewis, S.H. (1997). Delegating to Software Agents. *International Journal of Human-Computer Studies*, 46, 485-500.

Scott, R., Roth, E., Malchiodi, E., Deutsch, S.E., Kazmierczak, T, Eggleston, R.G., & Kuper, S., Whitaker, R.D. (2002). Using Software Agents in a Work Centered Support System for Weather Forecasting and Monitoring. *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, Baltimore, ME, September 23-27, 2002.

Whitaker, R.D., and Eggleston, R.G (2001). *Integrated Flight Management (IFM): Analysis and Interface Recommendations*. Draft Technical Report.

Young, M.J., Eggleston, R.G., and Whitaker, R. (2000). Direct Manipulation Interface Technique for Interacting with Software Agents. *Proceedings of the NATO/RTO Symposium on Usability of Information in Battle Management Operations*, Oslo, Norway, 10-13 April, 2000. RTO-MP-57 (AC/323/(HFM)TP/29), November 2000.