14th ICCRTS: C2 and Agility

Concept of a Portal for the Integration of COP-Objects from Heterogeneous Sources

Topic 9 - C2 Architecture and Technologies

Dr. Thomas Nitsche and Dr. Andreas Wotzlaw

FGAN/FKIE, C2IS Department

Neuenahrer Straße 20,

D-54343 Wachtberg-Werthhoven, Germany

Tel.: +49 228 9435 505

{nitsche, wotzlaw}@fgan.de

Concept of a Portal for the Integration of COP-Objects from Heterogeneous Sources

Abstract

In this paper we describe a new concept of an integration portal that, firstly, combines entire or partial operational pictures of different heterogeneous C2 services into a single operational view and, secondly, supports user business processes intervolving multiple services. We aim at integrating at the presentation layer, that is: services provide the integration portal with selected parts of their business objects in a form of visualizable elements which then are integrated and shown to the user in a single common operational picture. By using various visualization components we can visualize potentially arbitrary kinds of business objects. The integration portal can thus easily be adapted to new future services and visualization modes.

Our approach has the advantage that neither every application to be integrated has to provide its own GUI for a straightforward but unsatisfactory integration on the "pixel level", nor a much more complex and expensive semantic integration on the business data level is required.

Finally, our integration idea allows for a step-wise migration of C2 services towards the integration portal. Here, more and more services are integrated into the portal with the visualization of their business objects, while some applications can still be handled as in conventional portals.

Keywords: C2SI Architecture, Integration, Portal, Heterogeneity, COP Visualization

1. Introduction

The current existing command and control information systems (C2IS), but more specifically their various applications, services, and subsystems, have usually evolved from completely different development and application environments. They include, e.g., commercial and military off-the-shelf (COTS/MOTS) systems as well as special purpose applications. This leads to a high degree of heterogeneity.

The paradigm of network-centric warfare (NCW, see [1], [2], [8], [20], and [22]) is a powerful concept to employ the opportunities of modern IT-systems. It aims to improve military effectiveness by achieving information superiority [1] by connecting all military systems, ranging from sensors (e.g., reconnaissance systems) over the command and control systems to actors (effect systems). In doing so the integration of heterogeneous systems becomes a complex and time-consuming challenge, especially by integrating those parts of the C2IS which are responsible for the generation of the common

operational picture (COP). The data formats which describe the business objects of different applications differ strongly in their syntax and semantics from each other. Moreover, lots of domain-specific knowledge is not directly accessible but encoded within legacy applications [21]. Thus the realization of a common and homogeneous information space is rather a difficult task. It is often not possible to adapt the logical business data models of those heterogeneous systems due to the restricted accessibility to the internals of legacy applications, the impossibility to change them, or because of significant costs of such modifications.

1.1. Portal Technology

The use of portal technology (see, e.g., [13]) offers a relatively easy and inexpensive way to perform the integration. Here the specific graphical user interfaces (GUIs) of every application are presented to the user "as they are" in parallel on the screen. Consequently it is the user who has to merge the information that is shown in the separate windows (see Figure 1) and interpret it correctly.

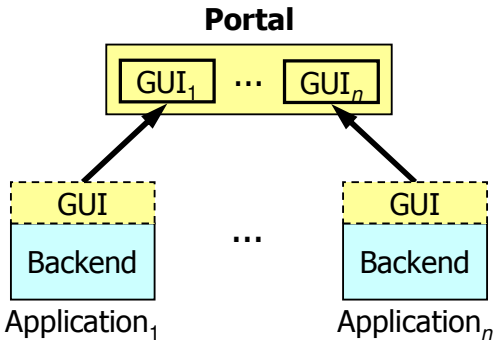

Figure 1. Idea of the portal technology.

The application of portal technology to the area of C2IS possesses unfortunately a number of disadvantages crucial for the user:

1. *Lacking business process support.* In a portal all activities are "application-driven". Thus the user must adapt his method of operation to the available functions of the different applications offered to him within the portal. From the user perspective the resulting business processes become complicated and not intuitive. The complexity of the system is transparent to the user and he is forced to find an appropriate application in each of his working steps. It must therefore be the goal to hide the complexity of the IT-system from the user and to provide him automatically with the necessary functions in each step of a business process.

2. *Lacking adaptation to the current operational picture.* It is very difficult to adapt the functions and the user interfaces of the applications integrated within the portal to the current information requirements of the user, e.g., to adapt it to the current mission or to the available communication resources. The applications to be integrated are in general not designated for such adaptations.

The following drawbacks correspond to the visualization of spatial data in portals. They are caused by the fact that each application which should be integrated provides its own specific user interface for the visualization of spatial data.

3. *Inhomogeneity of views.* Different applications can use different components for the visualization of spatial data, e.g., different geographical information systems (GIS). Thereby the presentation of such data can vary from one application-specific user interface to another. The user must therefore mentally compare and merge the data presented to him in a various manner, e.g., using different symbolization. This requires much more professional knowledge for the user, induces additional work, and can cause misinterpretation of data.

4. *Fragmentation of the overall view*. Even though all applications may use the same components for the visualization of spatial data, the user must still visually match the symbols and graphics as well as their positions (and potential movements) from the various application-specific visualizations in order to obtain an overall view of the current operational picture. For example, a certain unit may be shown in multiple windows from different applications. The user than has to mentally identify that their corresponding tactical symbols actually represent the same unit. This visual matching can again be a source for possible misunderstandings and misinterpretations.

1.2. Integration Portal

To eliminate the drawbacks of the portal technology discussed above, we propose a new portal technique which we call *Integration Portal*. Our Integration Portal combines (entire or partial) operational pictures of different heterogeneous C2 applications and services into a single operational view. Moreover, it supports business processes intervolving multiple services. The goal of the Integration Portal is to provide the user with a uniform and consistent view at the operational picture. By its reduced complexity it consequently simplifies decision-making for the user and decreases the possibilities for misunderstandings (see [17]).

The basic idea behind the Integration Portal is the integration of operational pictures of different C2 applications at the visualization (or presentation) layer. Our approach has the advantage that neither every application has to provide its own GUI for a straightforward but unsatisfactory integration on the "pixel level" (see Figure 1), nor a much more complex and thus expensive semantic integration of business data (i.e., on the business object layer) is required. The Integration Portal can be located between those two extremes.

More specifically, in our approach it is not necessary to adjust (or even integrate) the different data models of the participating applications and services. According to our integration idea, only those parts of the business objects that are relevant for the visualization are sent to the Integration Portal where they are presented to the user in a homogeneous way (see Figure 2).

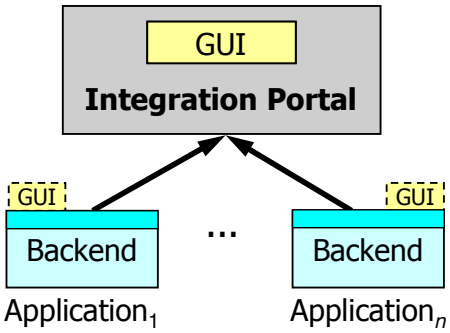

Figure 2. Main idea of the integration portal.

In the context of the Integration Portal, a *service* is an application which provides some specific functionality. In the military domain we have, among others, services for managing distributed data, operational pictures (COP-service, see [5] and [6]), or decision support systems (see [11] for an overview). For instance, a BlueForce-Tracking-Service can be seen as a service that provides information about the actual positions of own and hostile troops.

Our approach enables the services to send the visualizable parts of their business objects in a form of *visualization elements* (see Section 2) to the Integration Portal (see Figure 3).

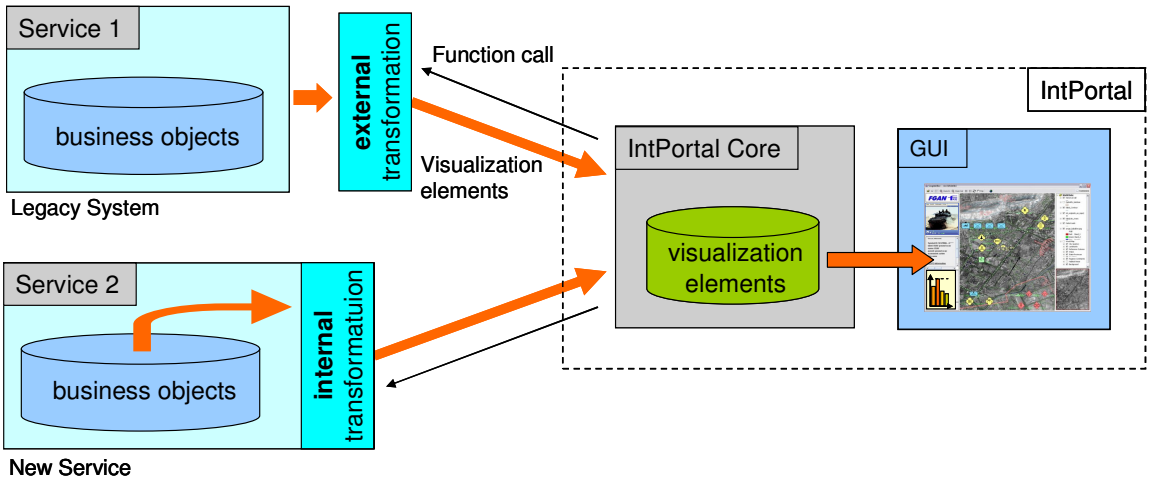

Figure 3. Transformation of business objects into visualization elements.

The portal is provided with a graphical user interface (GUI) for a combined presentation of the visualization elements of all services. In addition, the user can call functions relevant to his business processes directly from the central GUI. These functions are provided by the services and can be called in the context of selected visualization elements but are executed by the services. Thus business processes spanning multiple applications can be realized in a more intuitive way for the user.

In order to provide an individual view onto the C2IS, the Integration Portal is instantiated for each user separately. To this end, a specific *session profile* is used for each user of the system. Besides the user-specific access rights, it also contains his *area of interest* determined by his current role or mission. Furthermore the session profile also includes information about the type and capacity of the user device (see [9]).

Moreover, for a better integration and reduction of redundancy, the Integration Portal shall encapsulate as much of those features and C2IS components that are relevant for the visualization and manipulation of spatial data as possible. Here, especially parts of geographical information systems (GIS) like, e.g., components for efficient scaling and presentation of different kinds of spatial data (e.g., raster or vector graphics), or commercial components for the generation of tactical symbols and graphics, are of crucial interest. Access rights to manipulate spatial data are granted according to the current role or mission of a user. Besides spatial data, also other kinds of data can be visualized in the Integration Portal, provided suitable visualization methods exist. The current implementation includes, e.g., media data, office automation files, or (formatted and unstructured) messages, besides the map presentation of the operational pictures.

Finally, our integration idea allows for a step-wise migration of C2 applications and services toward a better integration. Here, more and more services will visualize their business objects by the Integration Portal, whereas other applications can still be embedded into the portal in a conventional way. Gounin and Guyard describe in [4] for their knowledge portal as an example, how the integration of applications can be realized using web-services.

The concepts presented here have some similarities to data or business mashups, where data from one or more sources is combined into a single presentation tool (e.g., mashup-based web-applications based on Google Maps or Flickr APIs.). However, our ideas are more general and are not tied to any concrete technologies. For more details on the mashup concept we refer to [10].

2. Integration Concept

The heterogeneity of the applications and services to be integrated implies that a large number of different business objects have to be integrated and visualized. With regard to the integration idea (see Section 1.2), the heterogeneous business objects of the services are maintained by the services themselves. Only those parts of the business objects that are relevant for the visualization are handed over to the portal where they are integrated and visualized (see Figure 3).

2.1. V-Objects as Visualization Elements

Each service connected to the Integration Portal can send the visualizable parts of its business objects to the portal in the form of abstract defined objects which we call *v-objects*. These objects are not tied to any concrete data type, thus allowing for a wide flexibility in terms of visualization and implementation. Practically, it means that we can represent with our v-objects almost any kind of data types. Especially geospatial

information can be represented directly in standardized formats such as the XML-based Geography Markup Language (GML, see [15]) or Keyhole Markup Language (KML, see [16]).

The v-objects are visualized in the presentation layer of the Integration Portal according to their type (see Figure 4), provided the Integration Portal includes means (i.e., appropriate visualization components, see Section 3.1) for visualization of those data types.

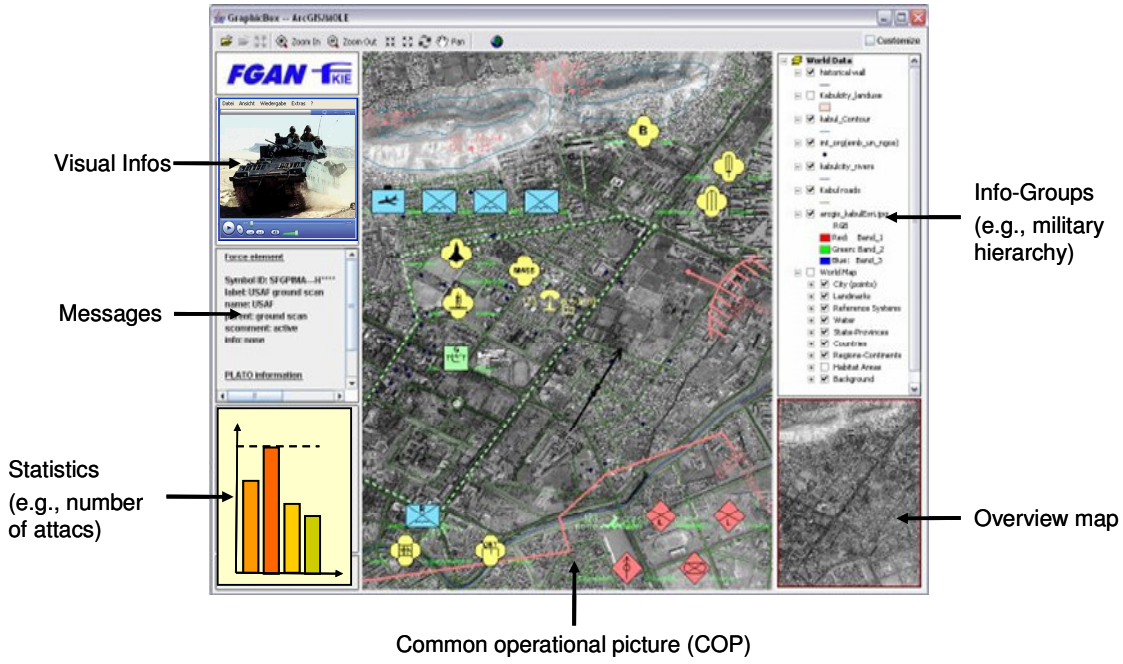

Figure 4. The different panels of the Integration Portals can visualize different kinds of objects.

A typical example includes military units that can be represented as spatial v-objects (described, e.g., by its position, unit name, and STD-MIL-2525 Symbol-ID, see [19]) and then be visualized in the Integration Portal as tactical symbols on the map showing the COP. In such a way also other data can be handled flexibly, e.g., weather data, statistics, or logistic information. However, the visualization of such different kinds of v-objects (representing the business objects of the integrated services) can differ substantially. Here we might display tree structures, text messages, function graphs, diagrams, videos, or other kinds of visualizations.

Basically, every service decides by himself which parts of its business objects should be visualized in the Integration Portal. To this end, a service has to transform those objects that are to be visualized into suitable v-objects. In the case of COTS systems where we do not have access to their source code, the transformation into v-objects has to be

realized by means of an additional *wrapper* component (see the external transformation component in Figure 3, or the wrapping of the Mail Service in Figure 5). Thus, the adaptation of the existing systems to the interface of the Integration Portal (see Section 3.2 below) or the development of appropriate wrapper-components is the price which has to be paid for a better integration.

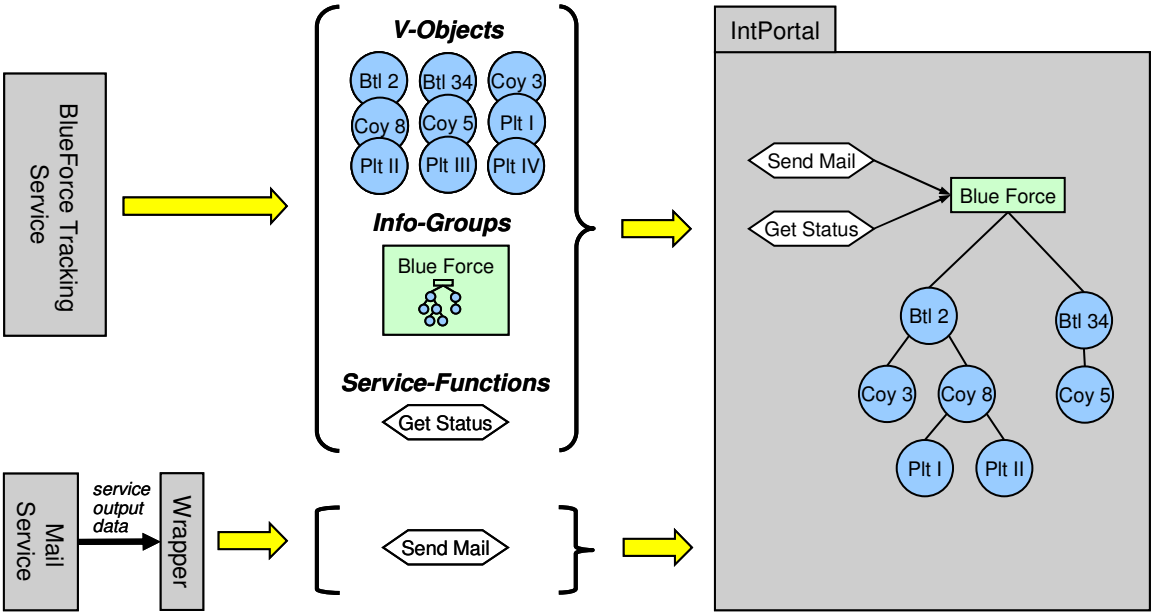

Figure 5. Sending of v-objects, info-groups and service-functions to the Integration Portal.

The task of the Integration Portal is to efficiently manage all those heterogeneous v-objects from different services and to visualize them correctly in an ergonomic way.

2.2. Information Groups

Note that the number of v-objects can be in some situations rather large. Moreover, their corresponding business objects are often related to each other and can be partitioned into subgroups. Typical examples include sets of all own forces, hostile forces, or spatial objects like streets, respectively. Since these relations can be of particular importance for the user, they are explicitly represented in the Integration Portal as *information groups* (*info-groups*). Info-groups enable both a better classification of the v-objects, as well as a better overview of the different types of v-objects.

Moreover, we often have some structure within a group. Therefore we let organize the v-objects within an info-group as a tree where the nodes correspond to v-objects. Each info-group can also include other info-groups. Additionally, we introduce grouping nodes that do not correspond to any specific v-object but to a whole group of them (like, e.g., a set of streets).

For a small example let us consider a set of v-objects representing own forces. Their hierarchical structure can be presented to the user as a tree. Figure 5 shows a corresponding information group called "Blue Force". It is defined by the BlueForce-Tracking-service and is used to describe the hierarchical relations between the military units it includes.

A more complex example is shown in Figure 6. Here the above-mentioned "Blue Force" group (green node) is part of another information group "Patrol" (see the magenta node on the top of Figure 6) that consists of own forces as well as some spatial data (here streets). The blue nodes in Figure 6 correspond to v-objects representing military units or streets, while the yellow nodes correspond to grouping nodes.

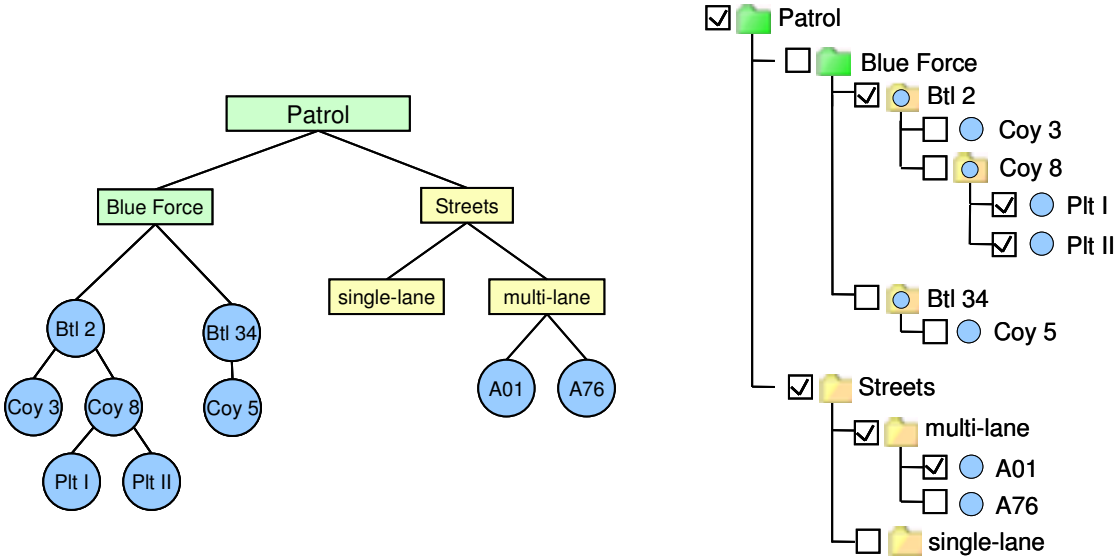

Figure 6. An information group "Patrol" with complex structure. The left part of the figure shows the concept of the hierarchical group structure, while on the right hand side its (explorer like) realization in the Integration Portal can be seen.

2.3. Service Functions

To support business processes that may involve multiple applications, we introduce *service-functions*. These functions are defined by the services and associated with one or more info-groups, and hence with all the v-objects within that groups. If a user calls such a service function from the Integration Portal, this function call is forwarded to the corresponding service and executed there accordingly.

As an example let us consider a service-function "Send mail", which we associate with all own forces, i.e., with all v-objects of the info-group "Blue Force". This function can now be used for sending mails from the Integration Portal to the military units visualized on the map. To do so, the user first selects certain units (that are represented by v-objects of the info-group "Blue Force") directly on the map and then calls the associated "Send mail" function from a popup-menu (see Figure 7). The selected units correspond directly

to the mail-receivers, so the user only has to type the message text and to send the mail. The complete message is then forwarded to the Mail Service which sends it to the specified receivers.

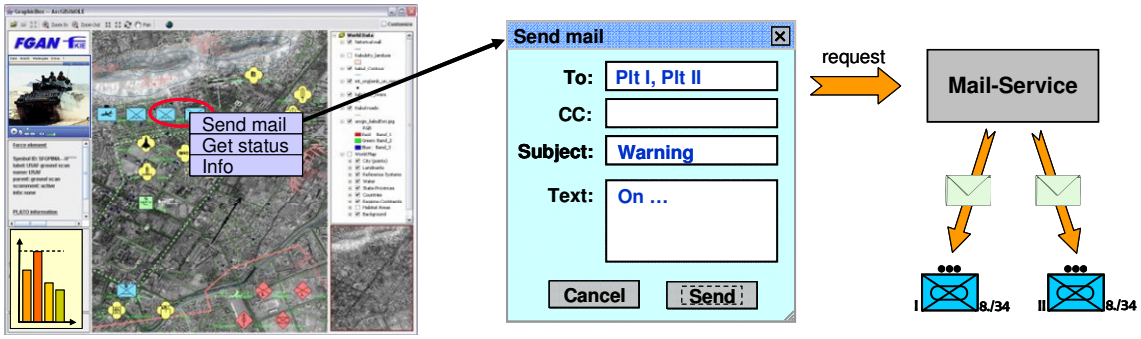

Figure 7. Call of the service-function "Send mail" in the Integration Portal.

The visualization elements of the Integration Portal thus comprise three types of objects we have introduced before:

- V-objects that correspond to the visualizable parts of the heterogeneous business objects from different services,
- Information groups that are used to describe structures and dependencies between v-objects, and
- Service functions that correspond to the functions of the C2 system.

3. Architecture of the Integration Portal

We now describe the general architecture of the Integration Portal. In its development, the following requirements were taken into account:

- Efficient maintenance of large sets of visualizations elements.
- Development of a flexible interface for the communication between the Integration Portal and the services. The interface can be used by the services to send the visualization elements to the Integration Portal, and to modify or to delete them afterwards.
- Development of conflict-solving strategies for the access to common resources in the Integration Portal. Since all services have access to the same GUI, we have to ensure that the data sent from the services is always kept consistent. Moreover,

the Integration Portal must not be overloaded with the data processing, whereas important data, e.g., warnings, should be displayed promptly to the user.

- Efficient and appropriate presentation of various types of visualization elements (e.g., tactical symbols for military units or sheets with tables for statistical data) in the user interface of the Integration Portal. The presentation of data in the user interface should also be appropriate for mobile devices.

- Encapsulation of external components of C2IS, like GIS.

3.1. Components of the Integration Portal

As described in Section 1.2, the Integration Portal is instantiated for each user with a separate *instance*. Each instance consists of the following three main, user-specific software components (see Figure 8 and [18]):

1. *Visual Object Manager* to preserve a consistent processing of service-requests and their visualization elements,

2. *Display Object Manager* to manage efficiently the visualization elements and to prepare their presentation to the user, and

3. *Graphical User Interface (GUI)* to present ergonomically the visualization elements to the user and to interact with him.

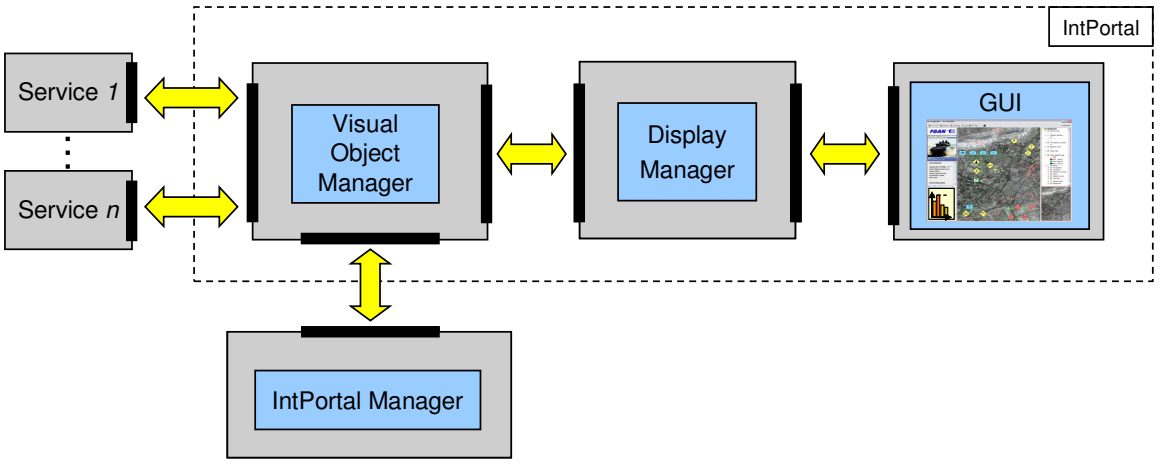

Figure 8. Components of the Integration Portal and its connection to services.

The instantiation of the Integration Portal for each user is accomplished by the *IntPortal Manager*, the forth component of our portal (see Figure 8). The IntPortal Manager is created at the start of the C2IS and is managing all user-specific instances of the Integration Portal. Besides the creation and initialization of each Integration Portal instance, the IntPortal Manager also manages the user preferences (i.e., settings of a

specific user). The user preferences are stored by the IntPortal Manager when the user finishes his session (i.e., by logging off). During the start of the Integration Portal, the IntPortal Manager initializes the new instance with the current user session profile and, if available, with the previously stored user preferences from his last session.

The advantage of partitioning the Integration Portal into four main components is the possibility to distribute them in the network in order to run them separately on different computers as shown in Figure 9. This distribution of the portal can be done according to the available resources like, for instance, the communication bandwidth, or the capacity of the user device. While the GUI of the portal normally runs directly on the user device, the remaining components of the Integration Portal can be relocated from the user device onto other available servers if required. This holds especially for the Visual Object Manager, which handles the various requests from the integrated services (see Section 3.2 below).

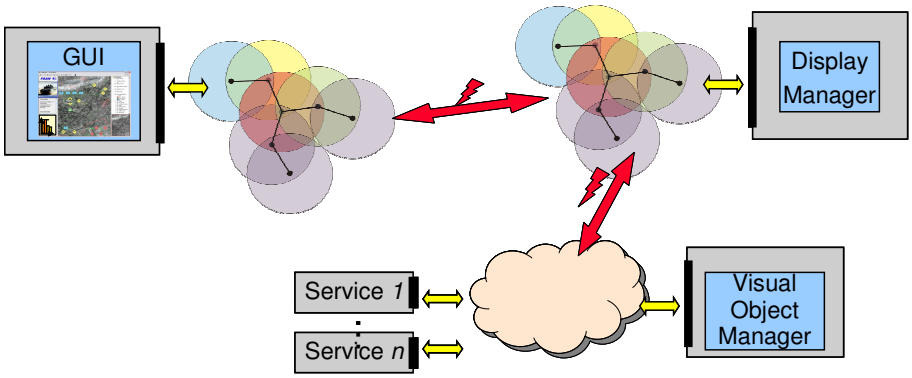

Figure 9. Distributing the portal components in the network.

The visualization elements that are sent to the Integration Portal from various heterogeneous services can contain different types of data and may have to be visualized in different ways. While the structure of information groups can be shown as a tree, the v-objects require much more specific visualization methods. Military units, for instance, can be visualized as 2D tactical symbols on a map, whereas weather data, statistics, or logistic information require other kinds for their presentation, e.g., in the form of a diagram. Furthermore, it is possible that some v-objects can be visualized in multiple ways, according to the type of data they contain.

To handle such diversity of visualization possibilities in the Integration Portal, different *visualization components* are used within the integration portal to visualize various v-objects (or parts of data included therein) according to their type. An example for a visualization component is described in [7]. It provides GIS-functionality by wrapping existing geographical information systems. Figure 10 shows how the various types of v-objects are visualized by different visualization components.

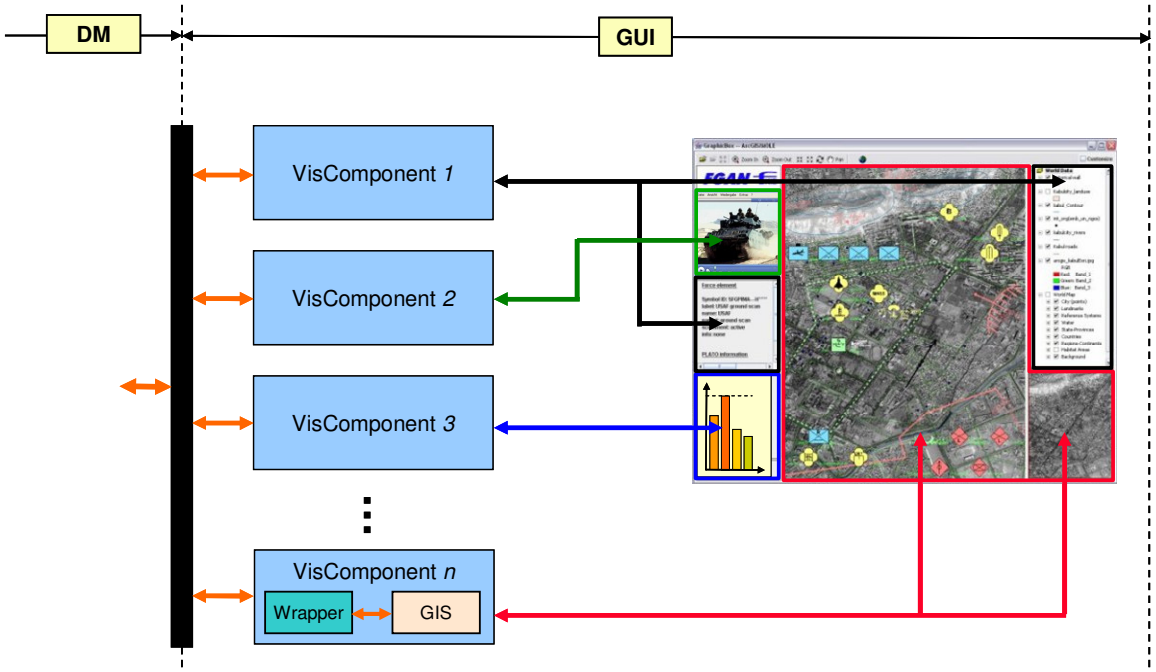

Figure 10. The visualization components fill their corresponding panels.

By the concept of visualization component, our Integration Portal is well prepared for visualization of future forms of business objects and thus can be flexible extended with new visualization methods if required.

3.2. Integrating C2-Services

The main task of the Integration Portal is to integrate and present the visualization elements coming from different services. This is accomplished by using *requests*. The goal of a request is to carry out a task assigned to it. An example is the update of some specific v-objects in the Integration Portal, e.g., due to a movement of units, which requires that their changed positions are sent to the portal for an updated operational picture.

Each request has its producer (e.g., a service) which generates the request and sends it to its recipient (e.g., to the Integration Portal). Furthermore, an *answer* is associated (in a request-reply-pattern style) with each request. This answer is sent by the recipient of the request to its producer after the request processing has been finished. In case the processing of a request has failed, the associated answer can include information for the producer about the reason of the failure.

There are three kinds of requests which can be processed or produced by the Integration Portal:

1. *Service-requests* are sent by the services to the Integration Portal. The services use them to communicate with our portal (see Figure 8), i.e., to send their visualization elements to the portal as well as to update or to delete them.

2. *IntPortal-requests* are sent by the Integration Portal to the services or to the IntPortal Manager. The requests to the services are mainly used to forward calls to service-functions from the user to the appropriate services. Requests to the IntPortal Manager serve the administration of the Integration Portal (e.g., by storing the user preferences).

3. *Manager-requests* are sent by the IntPortal Manager to the Integration Portal. They also support the administration of the Integration Portal, especially the instantiation and initialization of each user-specific instance of the portal.

For a detailed description of these requests and the protocols for the communication of the services with the Integration Portal we refer the reader to [14].

3.3. Encapsulating GIS

For the presentation of the visualization elements we use a number of dedicated, data-specific visualization components. Existing and commonly used GIS are used to visualize and manipulate spatial data which is of crucial importance for the generation of the COP. In our approach the display features of those applications are encapsulated in a special software component called *GIS visualization component*.

Our goal was to provide a modular and product-independent integration of GIS. To this end a new generic interface has been developed which enables us to integrate various parts of GIS (e.g., libraries with basic and frequently used functions) into a C2IS in a flexible and service-independent way. With regard to the main idea of the Integration Portal (see Section 1.2) it is possible to successively eliminate all redundant, application-specific visualization modules from C2IS and to encapsulate them in the Integration Portal. This allows for a flexible exchange of GIS and consequently simplifies maintainability and extensibility of the system. Furthermore, the complexity of the C2IS architecture as well as that of the integrated services will be reduced since changes in the GIS have no or only a marginal influence (due to the modularity and the encapsulation of the GIS within the GIS visualization component).

A concept for a product-independent integration of GIS, or parts of it, into the Integration Portal is presented in detail in [7].

3.4. Solving Resource Conflicts

At runtime various resource conflicts can occur in the Integration Portal. They are mostly the result of a simultaneous and shared access of different services to the same GUI. The requests of all services have to be processed in such a way that the user device will not be overloaded. Especially important elements (like warning messages) have to reach the user with only a minimal time delay und their visualization should not be disturbed by processing other visualization elements. The strategies for solving the conflicts

mentioned above can be seen as a kind of optimization strategies. In the Integration Portal we distinguish between *priority strategies* and *update strategies* [14], which we now discuss in more detail.

Priority strategies

The processing order of the received visualization elements in the Integration Portal is modified dynamically such that elements more important for the user are processed (and delivered to the visualization) earlier than other, less important elements. The importance of the visualization elements is specified by assigning a suitable *priority* value to the service-request which contains the visualization elements. Every service has the possibility to specify a priority to each of its own requests according to the current priority strategy. Furthermore, the services can also be prioritized according to their role in the C2IS. The Visual Object Manager (see Figure 8) of the Integration Portal includes dedicated components called *mediators* which rearrange the processing order of the requests according to their priorities. More specifically, every mediator modifies the order of the requests in its waiting queue in such a way that requests with a higher priority will be visualized earlier than those with a lower priority. Thus, the actual priority strategy indirectly results from the priorities that some higher instance of the system (e.g., access control) has granted to the services, and they in turn to their requests.

Update strategies

Potentially each service can sent a huge amount of update requests for its v-objects to the Integration Portal. Normally only the current state of each v-object is relevant for the user. Thus, earlier, out-dated updates for v-objects which are still waiting for their visualization can be deleted and replaced by the update with the latest state of that object. This measure prevents the visualization components of the GUI from being overloaded by too many updates. In the Integration Portal every update request for v-objects undergoes an additional verification according to the current update strategy. Hence, it is guaranteed that all incoming requests are processed with regard to their importance (priority) and actuality.

Consider the case where a service sends its requests rather frequently, e.g., several times per second, to the Integration Portal. An example could be the already mentioned BlueForce-Tracking-Service (see Section 1.2, [5], or [6]), which has not been optimally set up. It will cause visualization problems if the user wants to be informed about all possible changes of the military units but the device he is using does not allow for such a high update rate. In such a case the service will try to inform the Integration Portal about lots of position changes. Unfortunately, the user device will shortly be overloaded and finally blocked with these new update information, due to its low performance possibilities. Consequently, it is of great importance to apply a suitable update strategy in such a case to ensure that the data processing in the Integration Portal will not be disturbed due to too much incoming data.

4. Summery and Conclusions

The Integration Portal described here combines (entire or partial) operational pictures of different heterogeneous C2 services into a single operational view. Moreover, it supports user business processes intervolving multiple services. In our approach we integrate at the visualization (or presentation) layer. Only those parts of business objects which are relevant for the visualization are sent to the portal by the services. The Integration Portal has the following advantages:

1. It provides a central user interface in which all visualization elements from different, heterogeneous services can be presented to the user in a combined and consistent way. In particular the spatial data from various services can be visualized onto a common operational picture shared by all services. The user can access it effectively by selecting the information groups or parts thereof that are to be displayed. This gives rise to some kind of user defined operational picture [12].

   Moreover, the integrated services do not have to provide their own GUIs. This allows for a successive elimination of redundant visualization parts from the various integrated applications that reduces the overall complexity of the C2IS.

2. It supports application-crossing business processes better than in standard portals. The user can call service-functions directly from the Integration Portal. The required arguments of the function can be specified by the user by selecting particular v-objects directly in the Integration Portal, e.g., by selecting them on the map of the COP. Consequently, the different applications and services themselves as well as the internal complexity of the IT-system are hidden from the user. It allows replacing application-driven business processes by more user-oriented ones that involve multiple service processes.

3. Encapsulation of GIS. By including various visualization components within the Integration Portal we can visualize various kinds of business data. Here it is possible to encapsulate functions from different, existing systems. Especially parts of GIS are of crucial interest here.

The network-centric maturity model, presented in [3], defines five levels of maturity and a possible migration path for the implementation of network-centric capabilities in a C2 system. This model was created to clarify how C2IS can move logically and smoothly from traditional toward network-centric ones. It recognizes that collaboration among participants across location, function, and organization combined with shared information is the crucial element for this transformation. This enables to generate shared awareness in a system where the focus is not on the information but on what it means and implies.

We believe that our concept of the Integration Portal addresses these both aspects of the C2 transformation. The collaboration involved here is focused on the nature of the information being shared among services in order to identify and sort out inconsistencies and redundancies. Its combination with the global defined, shared data type v-object increases the quality of information and ensures better collaboration. Furthermore,

developing relationships and linkages between services by means of information groups and service-functions increases the effectiveness of shared information which leads towards shared awareness with richer interactions between the services and better understanding of shared data.

By hiding underlying complexity of the different services and applications from the user and allowing customized, user-specific views onto the shared and integrated handling of the COP-objects from heterogeneous services, the Integration Portals can help the military commanders and warfighter in successfully accomplishing their missions. Especially, the Integration Portal can support them to better understand the situational picture in a shared and consistent manner and adapt to changes more easily.

Acknowledgment

We would like to thank Marc Spielmann for many fruitful discussions and good remarks. This work was funded by the Federal Office for Information Management and Information Technology of the Bundeswehr.

5. References

[1] Alberts, D. S. and Hayes, R. E. *Power to the Edge*. Washington, DC: CCRP, 2003.

[2] Alberts, D. S. *Agility, Focus and Convergence: The Future of Command and Control.* In The International C2 Journal, Special Issue on the Future of C2, 1(1): 1-30, CCRP Publication Series, Washington, DC, USA, 2007.

[3] Alberts, D. S. and Hayes, R. E. *Planning: Complex Endeavors.* Washington, DC: CCRP, 2007.

[4] Gounin, D. and Guyard, A. B. *COP: From Knowledge Webs to Knowledge Portals*. In Proceedings of the 10th International Command and Control Research and Technology Symposium (ICCRTS), McLean, VA, 2005.

[5] Jansen, N. and Nitsche, T. *COP-Service* (*LaGe-Service).* Technical Report FKIE/ITF/2008/1, FGAN-FKIE, Wachtberg, Germany, 2008 (in German).

[6] Jansen, N. and Spielmann, M. *Architecture Concept for a Distributed Database and Data Management System for Disadvantaged Communication Networks*. In Proceedings of the Military Communications and Information Systems Conference (MCC), Cracow, Poland, 2008.

[7] Krämer, D. *Modular Connection of Geographical Information Systems to C2IS* (*Modulare Anbindung von Geoinformationssystemen an FüInfoSys).* FKIE-Report No. 170, FGAN-FKIE, Wachtberg, Germany, 2008 (in German).

[8] Kruse, J., Adkins, M., and Holloman, K. A. *Network Centric Warfare in the U.S. Navy's Fifth Fleet – Web-Supported Operational Level Command and Control in*

Operation Enduring Freedom. In Proceedings of the 38rd Hawaii International Conference on Systems Sciences (HICSS), IEEE Computer Society, 2005.

[9] Käthner, S. and Spielmann, M. A *base component for future C4ISR systems*. In Coalition C4ISR Architectures and Information Exchange Capabilities (RSO-IST-042-RSY-014). The Haag, Netherlands, 2004.

[10] Klocker, Ch. *Mashups: A New Concept in Web Application Programmin*g. Vdm Verlag Dr. Müller, 2008.

[11] Mulgund, S. (Ed.). *The International C2 Journal, Special Issue on Decision Support for Network-Centric Command and Control*, Vol. 6, No. 2, 2007.

[12] Mulgund, S. and Landsman, S. *User Defined Operational Pictures for Tailored Situation Awareness.* In 12th International Command and Control Research and Technology Symposium (ICCRTS), Newport, RI, 2007.

[13] NATO – North Atlantic Treaty Organization. *WISE Web Information Service Environment, System Architecture Guide. Version 1.1*. Norfolk, VA, USA: Allied Command Transformation, 2003.

[14] Nitsche, T. and Wotzlaw, A. *Concept of a Portal for the Integration of Data from Heterogeneous Sources (Konzeption eines Portals zur Integration von Daten aus heterogenen Quellen)*. FKIE- Report No. 167, FGAN-FKIE, Wachtberg, Germany, 2008 (in German).

[15] OGC – Open Geospatial Consortium. Geography Markup Language website, http://www.opengeospatial.org/standards/gml, 2009.

[16] OGC – Open Geospatial Consortium. OGC KML website, http://www.opengeospatial.org/standards/kml, 2009

[17] Ruff-Stahl, H.-J. *The Human Factor in Future Warfare – Situation Awareness and Decision Support in Complex Systems (Der menschliche Faktor im zukünftigen Kriegsbild – Situation Awareness und Entscheidungsunterstützung in komplexen Systemen).* Strategie und Technik, Vol. 6, pp. 40–46, 2008 (in German).

[18] Spielmann, M. *Visualization of Spatial Data in Command and Control Information Systems.* In Military Communication and Information Systems Conference (MCC 2006), Gdynia, Poland, Sept. 2006.

[19] U.S. Department of Defense. Interface Standard Common Warfighting Symbology MIL-STD-2525C, http://assist.daps.dla.mil/quicksearch/basic_profile.cfm?ident_number=114934, 2008.

[20] Villa, J. *Network Centric Warfare – A Tool or Hindrance to the Operational Commander.* Naval War College, Newport, RI, 2004.

[21] Whitt, L. *Why is C4I Software Hard to Develop?* In 12th International Command and Control Research and Technology Symposium (ICCRTS), Newport, RI, 2007.

[22] Wilson, C. *Network Centric Warfare: Background and Oversight Issues for Congress.* CRS Report for Congress RL32411, 2004.