High tempo battlefield requirements, rapidly evolving communications/information technologies and the need to include legacy components, call for a System-of-Systems engineering approach in the development of Data Fusion Enabled Networks (DFEN). System-of-Systems engineers are concerned with large scale interdisciplinary issues combining multiple, heterogeneous, distributed systems that are embedded in networks at multiple levels and multiple domains. These issues are also of concern to Data Fusion Engineers because the networks being engineered to resolve the issues are the environment in which future data fusion systems must perform successfully. This paper is a description of a System-of-Systems approach to the development of DFEN's. Applying System-of-Systems techniques, a DFEN software and hardware infrastructure is being developed. The infrastructure is being developed as a capability within a network enabled C4ISR infrastructure, using a Service Oriented Architecture (SOA) and a DFEN specific ontology. Within this framework, important aspects of Data Fusion system development can be addressed in a process-driven, scalable and evolvable manner.

Keywords: System design, Distributed systems, System-of-Systems, Ontology, Architecture, Systems engineering.

1 Introduction

The mix of high tempo battlefield conditions, rapidly evolving technologies, and the need to include legacy components, calls for a System-of-Systems (SOS) engineering approach in the development of data fusion enabled networks (DFEN) (illustrated in Figure 1).

The concept the US calls Network Centric Warfare (NCW) and the UK knows as Network Enabled Capability (NEC), is defined as "an emerging theory of war in the information age that seeks to translate an information advantage into a competitive warfighting advantage through the robust networking of well informed geographically dispersed forces allowing new forms of organizational behavior". Military operations are now conducted jointly, with multiple branches, allied and coalition forces, requiring increased levels of interoperability.

Traditional "stovepipe" C4I systems are no longer adequate for this highly connected, collaborative environment. This is the domain of SOS engineering. SOS brings with it a number of non-trivial, challenging requirements.

- Rapidly integrate data from distributed, heterogeneous, dynamic entities; present the relevant info with high usability for command decision makers; formulate an integrated response; and bring the appropriate forces to bear.
- System configuration on the fly, as dictated by evolving mission needs.
- A collaborative networking infrastructure supporting secure near-real-time manipulation and sharing of massive amounts of increasingly complex information, collected and fused from diverse sources, is required to facilitate ad-hoc teams.

A fundamental issue specific to data fusion is becoming clear. That is, the numerical and heuristic techniques used to perform data fusion are embedded in a significant set of non-fusion, support functions. When developing information fusion (IF) and resource management (RM) systems, engineering the support infrastructure detracts
from the main IF and RM focus. This is even more acute in the SOS domain. The level of cost and complexity needed for supporting SOS ancillary functions is greater than in previous isolated systems.

SOS-based C4ISR architectures are being researched and developed. As these SOS architectures and infrastructures emerge, enabling robust and affordable data fusion capabilities at the architectural level will prove highly beneficial.

This paper describes how a specific C4ISR infrastructure that encourages component re-use in a SOS is evolving to provide a structured development environment for DFEN systems. The resulting DFEN infrastructure inherits capabilities necessary to support operation in an ad-hoc, tactical, operational and inter-operational network. It’s also recognised that evolving software development methodologies that allow for support of SOS considerations are going to be needed for efficient development of DFEN systems. Evaluation of these tools and an approach to combine them with DFEN specific functional paradigms to enable DFEN system development will be presented.

2 DFEN Infrastructure Overview

The DFEN infrastructure under development seeks to achieve the following goals: open structured architecture; distributed computing; fault tolerance; flexible configuration; timely and efficient data distribution; data source independence; application independence; security; modularity; scalability; and the use of interface and integration standards.

The DFEN infrastructure builds from an evolving C4ISR Infrastructure that has come into being as a result of lessons learned providing communications, networking and data management to a number of countries’ armed forces. Using key structural and process features of this infrastructure, important fusion aspects can be addressed at the architectural level. Some of the main issues currently being addressed are:

- ontology-based representation of the information in the infrastructure
- provision of ancillary support functions (i.e., communications, message routing and processing, database management, man/machine interface, and sensor management)
- integration of legacy data fusion applications and systems
- where and when fusion is performed in the processing flow
- selection of appropriate fusion algorithms
- the role of the man-in-the-loop

The first three items deal with the required architectural structure for DFEN system development and component reuse. The remaining items deal with the development of DFEN development methodologies and processes to enable efficiency through solution pattern re-use. Section 3 discusses the DFEN architectural structure, while Section 4 presents the applied processes that enhance DFEN system development.

3 Architectural Structures for Re-Use

The following architectures attempt to resolve the DFEN issues of representing information, providing ancillary support functions and integrating legacy system components. Use of architectures is a way to encapsulate and organize expert knowledge so that solution patterns are more easily recognized and applied, even by non-expert developers.

3.1 Ontology-Based Representation of the Information in the Infrastructure

The representation of information within the DFEN infrastructure is a critical system characteristic. The success of heterogeneous, distributed systems inter-operating, depends on a common understanding of shared data. A generalized information representation aids in the development of generalized interfaces and provides additional context to help maintain the semantic content of the system’s information.

Ontologies are becoming increasingly important because they provide the critical semantic foundation for many rapidly expanding technologies such as software agents, e-commerce and knowledge management [5]. Ontologies are generally recognized as an essential tool for allowing communication and knowledge sharing among distributed users and applications, by providing a semantically rich description and a common understanding of a domain of interest. Ontologies include class/subclass hierarchies, relationships between classes, class attributed definitions and axioms that specify constraints.

A DFEN specific ontology is currently under development. Figure 2 shows the highest level of this ontology. The ontology is intended to incorporate distributed fusion concepts as well as battlespace entity representations. The addition of the former will be needed to aid in the DFEN process orchestration and provide the necessary background information for distributed fusion processing.
The implementation of distributed fusion systems must deal, to a large extent, with providing non-fusion specific functions. These functions include communications, message routing and processing, database management, man/machine interface, and sensor management. As shown in Figure 3 typical data fusion systems can be more than 90% ancillary support functions [2].

To manage costs, the ad hoc one-of-a-kind way in which fusion systems are developed must cease [3]. Using a network enabled, C4ISR SOS infrastructure and a DFEN ontology, a DFEN software and hardware infrastructure is being developed. This infrastructure allows both the supporting functions, and IF and RM software modules to be re-used. The SOS approach is taken by applying a Service Oriented Architecture to the applications software within the infrastructure.

**Figure 2: Highest Level of DFEN Ontology**

Other knowledge representation approaches and technologies are being studied to determine the most suitable one or combination for DFEN. Examples include Conceptual Spaces [6], and integrated semantic/spatial object representations using region quad-trees [1]. Attention to knowledge representation is considered a critical enabling factor to successful collaborative interaction of DFEN components.

### 3.2 Provision of Ancillary Support Functions and Integration of Legacy Systems

The engineering of a system of systems differs from the engineering of a single system. Inter-operability and collaboration mechanisms are of the utmost importance because the overall system capabilities can only be achieved by the collaborative interaction of the multiple systems. Apart from the need to interact, systems need to also consider how the systems around them can impact them, both positively and negatively.

The SOS impact on single system developers is that their systems can no longer be developed as standalone systems but, instead, as a part of larger systems that are delivering complex, synergistic capabilities. Interfaces, architecture and collaboration need to be considered with the highest priority. Impacts of external systems must be considered.

The GD Canada C4ISR infrastructure is being developed to answer the SOS challenges posed by the joint collaborative NCW/NEC vision. The issues are large scale inter-disciplinary ones with multiple, heterogeneous, distributed systems that are embedded in networks at multiple levels and multiple domains. The scope of the infrastructure is shown in Figure 4.

**Figure 3: Percentage Composition of Software Application Types in Data Fusion Software**

**3.2.1 System of Systems Approach to Data Fusion**

SOS engineering is the engineering of the capabilities of a mix of legacy and new systems into a system of systems capability that is greater than their simple sum. The main objective of a system of systems is to provide capabilities that can only be achieved with a mix of multiple, autonomous and collaborating systems.
It incorporates a SOS perspective and applies a Service Oriented Architecture (SOA) to provide software architectural solutions to interfacing and collaboration.

3.2.2 Application of Service Oriented Architectures

The SOA aims to provide enterprise operational solutions that can extend or change on demand. SOA solutions are composed of reusable services, with well-defined, published and standards-compliant interfaces. The SOA provides a mechanism for integrating existing legacy applications regardless of their platform or language.

Conceptually, there are three major levels of abstraction within SOA:

- **Business Processes**: A set of actions or activities performed with specific business goals in mind.
- **Services**: Represent logical groupings of operations.
- **Operations**: Analogous to object-oriented methods, these represent atomic logical units of work (LUW). Many method-like characteristics are present in an operation, like structured interfaces and returns.

A business process consists of a series of operations that are executed according to a set of business rules. The sequencing, selection, and execution of operations, is termed service or process choreography. Typically, choreographed services are invoked in order to respond to business events. So a key feature of SOA, due in no small part to the service being the basic abstraction, is the ability to incorporate user business processes into the system.

The way in which a specific military user performs the process of decision making could be elicited and designed into the system. The service layer implementation of this process would incorporate services from the DFEN infrastructure. In operation, the user’s decision making process would choreograph the service interactions and performance, including the DFEN services that are needed. The power then comes when a new decision making process must now be used, but the same group of DFEN services are simply choreographed in a new way to provide the desired fusion capability.

The Data Infrastructure Services and Applications Framework (DISAF) shown in Figure 5 is the SOA being applied. The DISAF provides an open structured architecture-based development environment. It also offers some specific capabilities developed as a result of lessons learned in working with several of the world’s modern military forces. These include: efficient, policy-based data distribution even in low bandwidth, high loss, high latency, wireless network environments; the provision of multiple, independent levels of security; and end-to-end Quality of Service awareness and control.

The DFEN infrastructure is being incorporated into DISAF by identifying and developing DFEN services. Using the DFEN specific ontology to represent information throughout DISAF enables intelligent access to information and intelligent collaboration by the services.

4 Methodologies and Processes for Re-Use

To further harness the architecture just discussed in Section 3, methodologies and processes will be applied. This provides a common map for finding or expressing system solutions. This common map will allow prior pieces of the solution (i.e., patterns) to be recognized and organized for re-use even by non-expert developers.

This approach allows key fusion aspects to be addressed by the DFEN infrastructure, maintaining the specialized
considerations of SOS development. The two that will be discussed are:

- Section 4.1- Where and When Fusion is Performed in the Processing Flow
- Section 4.2 – Selection of Appropriate Fusion Algorithms

4.1 Where and When Fusion is Performed in the Processing Flow

A fundamental issue in developing a data fusion system is the choice of the fusion architecture. A key question is where to combine or fuse the data in the processing flow of two or more information sources (sensor or non-sensor generated). The solution must also take into account SOS considerations which deal with combining system capabilities to provide an enhanced capability. This design choice affects the quality of the fused product, the nature of the algorithms or techniques that may be used, the complexity of the processing logic, and the bandwidth of the communications required between the sensor and the fusion system. The choice of architecture is not arbitrary; instead it depends on the nature of the sensors involved as well as the nature of the inferences sought and available infrastructure [2].

In a SOA, a big part of this answer is to identify and create appropriate services. As mentioned previously, the DFEN infrastructure would include a group of DFEN services available in the service layer. Given that this is done, user business process modeling could be used to elicit and model the user's decision making process. This would be implemented in the service layer which in turn is implemented in the software component layer. This process is where Service Oriented Analysis and Design (SOAD) would be applied.

To serve the domain specifics of data fusion, a functional paradigm that is fusion specific would be applied. An example of such a paradigm is given by [3]. The Data Fusion Tree (DFT) paradigm and its dual paradigm the Resource Management Tree provides a method for identifying and organizing fusion solution patterns. This pattern illumination and organization offers fusion specific design and development the benefit of efficiency through solution pattern identification and re-use.

The combination of SOAD and a fusion specific functional paradigm is a strong enabler of efficient fusion architecture determination, design and development.

4.1.1 Service Oriented Analysis and Design (SOAD)

SOAD is a methodology which provides a systematic format for development in the SOA environment. SOAD spans multiple design and development disciplines and adds other relevant system considerations in an integrated manner. Figure 6 illustrates how SOAD spans the main application areas of three of today's existing modeling approaches, Enterprise Architectures (EA), Business Process Modeling (BPM), and Object Oriented Analysis and Design (OOAD) [7].

Figure 6: Service Oriented Analysis and Design [7]

In thinking in terms of services, the business or operational processes of an organization must now become incorporated into the design of the system. This explicit incorporation of the Business domain provides the potential to better meet user requirements by making their operational processes explicit components in the system. For military based fusion systems, this means that the warfighter's operational methodologies become part of the system. How a decision maker gathers, processes and then uses information to make a decision can be part of the system.

While the SOA approach reinforces well-established, general software architecture principles such as information hiding, modularization, and separation of concerns, it also adds additional themes such as service choreography, service repositories, and the service bus middleware pattern, which require explicit attention during modeling. The operational processes of the user must now drive the analysis and design rather than the established "use-case driven" approach of OOAD. The difference here is that in SOA especially within the larger SOS paradigm more than one system at a time must now be examined because a process (or even a single service) may span more than one system.

The architecture of the system in a SOA generally comprises stateless, fully-encapsulated, and self-describing services satisfying a generic business service that is closely mapped to the BPM as shown Figure 7.
These services might be composed of a number of collaborating or orchestrated services. SOA must also take into consideration, syntax, semantics and policies to support ad hoc composition, semantic brokering and runtime discovery.

In terms of developing a fusion architecture using the SOA environment, SOAD methodologies and processes will need to be used to identify and define DFEN services. These services will then be used as building blocks for future implementations of DFEN systems. The actual composition of these parts will be determined by the Business Layer.

4.1.2 Paradigm-Based Architecture

Reduced software costs are achieved through the encapsulation of knowledge with reusable DFEN services, patterns and components using SOA. These now need to be organized in a DFEN specific functional paradigm. The following is an example of a fusion specific functional paradigm [3]. The paradigm is called the Data Fusion Tree (DFT). Both it and its dual paradigm the Resource Management Tree (RMT) are shown in Figure 8.

![Figure 8: Data Fusion Tree and Resource Management Tree [3]](image)

The DFT specifies how the data is to be batched and the order that the batches are to be processed. The fusion of each batch of data is performed in its fusion node of the DFT. This paradigm is used in the SW system engineering process shown in Figure 9.

![Figure 9: Fusion SW Development Process Using DFT/RMT Functional Paradigms [3]](image)

The idea here is to identify and apply proven fusion specific software development approaches to the DFEN SOA Infrastructure. Doing so provides a template solution process that is specific to data fusion, but integrated with the SOS-based DFEN Infrastructure. Further research will be done to enhance and implement this approach.

4.2 Selection of Appropriate Fusion Algorithms

In a SOS, a mix of capabilities from both legacy and new systems, are engineered to deliver enhanced capabilities in concert. This applies to the fusion aspect of selecting appropriate fusion algorithms for a given information problem. Once a requirement for fusion has been determined through a proper analysis of the user’s operational procedures, available infrastructure and operational environment, determination of the type of fusion problem, or problems needs to be made. In the SOS world, the way in which the algorithms are introduced must take into consideration how they will interact with existing or new algorithms and where in the system they are to operate as discussed earlier in Section 4.1. Algorithms are not an isolated solution in themselves but part of a collaborative information processing mechanism that is fluid and operating in a resource restricted, extreme environment.

The problem of algorithm selection depends strongly on the nature of the inferences sought by the data fusion system, the specific application, and sensor data available [2]. Even with these factors accounted for, a given situation will likely result in having a number of competing algorithms that are applicable to the process. Some trade-offs can be made based on the mathematical nature of the algorithms and the availability or lack of availability of a priori information. Other trade-off...
considerations include throughput constraints, computer resource requirements of algorithms, and operational constraints. The SOS considerations that must also be considered include the support of standard interfaces, applicable capabilities of the algorithm and where and how they contribute to a larger system, and how the other systems and algorithms impact its performance.

A structured process can be applied to algorithm selection based on four separate perspectives [2]:

- systems analysis view
- operational view
- mathematical or numerical view
- end-user's view

To this list must be included:

- SOS analysis view

An important aspect of any algorithm selection process that will be applied is the categorization of both the fusion problems and fusion solutions. Fusion problems types could then be identified and mapped to fusion solutions that are suitable [1]. The SOS perspective must consider that the overall fusion problem is made up of numerous local fusion problems that must deliver results that flow to help solve the next fusion problem. This process flow is determined by the operator business process and could be refinement of the data inside a single JPL Fusion Process level or a promotion to the next level. This ability to break a fusion problem down to its constituent parts may provide emergent solutions not previously available.

5 Conclusion

A SOS approach to the provision of DFEN systems has been adopted and described in this paper. Using a network enabled, C4ISR infrastructure using a SOA development environment and DFEN ontology, a DFEN software and hardware infrastructure is being developed to provide a structure for re-use of both DFEN and supporting function system components. This structure is coupled with SOAD and fusion specific functional paradigms, to encapsulate and organize the domain solution patterns for re-use. How SOS is applied to different aspects of the fusion domain was also discussed.

References


