Allied Data Publication 34

(ADatP-34(D))

NATO Interoperability Standards and Profiles

Volume 6

Annexes

Date: 12 January 2010

C3 CCSC NATO Open Systems Working Group

Table of Contents

This page is intentionally left blank

1. ANNEXES

001. This document has been developed and agreed (AC/322(SC/1-WG/4)N(2010)0002-AS1, 24 Mar 10) by the NATO Open Systems Working Group (NOSWG) under the authority of the NATO Consultation, Command and Control Board (NC3B). Under AC/322-N(2010)0038-AS1, the NATO Consultation, Command and Control Board noted ADatP-34(D) and approved the standards and profiles in Volume 2 as mandatory for use in NATO common funded systems in accordance with the NATO networked C3 Interoperability Policy.

002. This NISP volume contains additional relevant appendices.

This page is intentionally left blank

A. SERVICES AND INTEROPERABILITY POINTS IN PLATFORM ORIENTED AND SOA ENVIRONMENTS

A.1. BACKGROUND

003. To paraphrase William Shakespeare [1] "What's in a name? That which we call a service by any other name would mean the same". The problem is that the meaning of service does not always mean the same thing; it is dependent upon the context in which it is used. A messaging service in a client/server or platform oriented environment is not the same as a messaging service in a SOA environment. Many use this confusion to indicate that they are providing a service in a SOA environment, when in fact the service is actually provided in a platform oriented environment.

004. The NNEC FS introduced the terms Service Interoperability Point (SIOP) and Service Interface Point (SIP).

- SIOP: A reference point within an architecture where one or more service interfaces are physically or logically instantiated to allow systems delivering the same service using different protocols to interoperate

- SIP: A set of attributes [technical specification] that specifies the characteristics of a service interface between interoperable systems in the NII

005. TACOMS Post 2000 describes an Interoperability Point (IOP) which is the point where nations agree to use STANAGs to interconnect their national systems to achieve interoperability of services. Similarly, The Information Exchange Gateway (IEG) concept describes an Interoperability Point where NATO and Nations interconnect their respective systems.

006. The NISP contains Profiles which contain a set of standards to be used to exchange services and may imply profiles for platform centric or SOA environments.

007. All these efforts describe service interoperability without going into the detail of the services being platform centric or SOA. This annex to the NISP Volume 1 will clarify the meaning of services in platform oriented an SOA environments, and their respective interoperability points.

008. The following diagram (Figure A.1) depicts generic interface points and service interoperability that is applicable for both platform centric and SOA environments. Details of platform centric and SOA services are described in the following sections.

[1] "O! be some other name: What's in a name? that which we call a rose By any other name would smell as sweet"

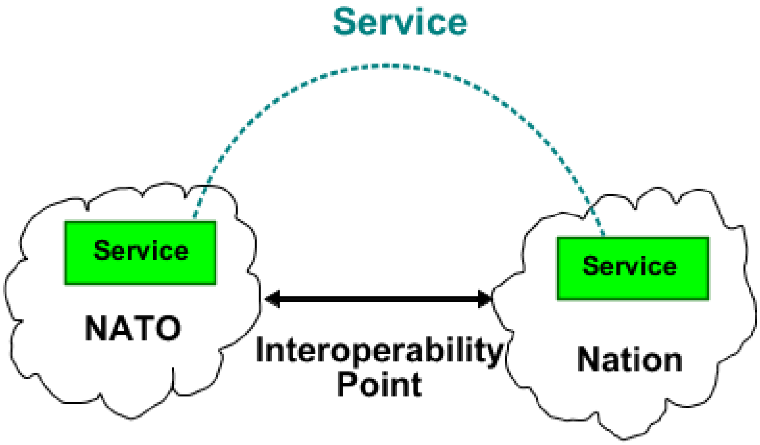


Figure A.1. Generic Interoperability Point and Service Interoperability

A.2. PLATFORM ORIENTED ENVIRONMENT

009. Volume 1, Appendix C of the NISP describes the NATO Technical Reference Model (NTRM) and NATO Common Operating Environment (NCOE) Component Model (NCM). These models can be used in describing services in a platform oriented environment.

A.2.1. NTRM

010. As stated in Volume 1, Annex E of the NISP, the NTRM focuses on separating data from applications and applications from the computing platform. The NTRM provides the definitions necessary for designing and defining architectures and related service components. It also identifies service areas (i.e., capabilities that have been grouped together by functions - see Figure A.2), as well as their interfaces.

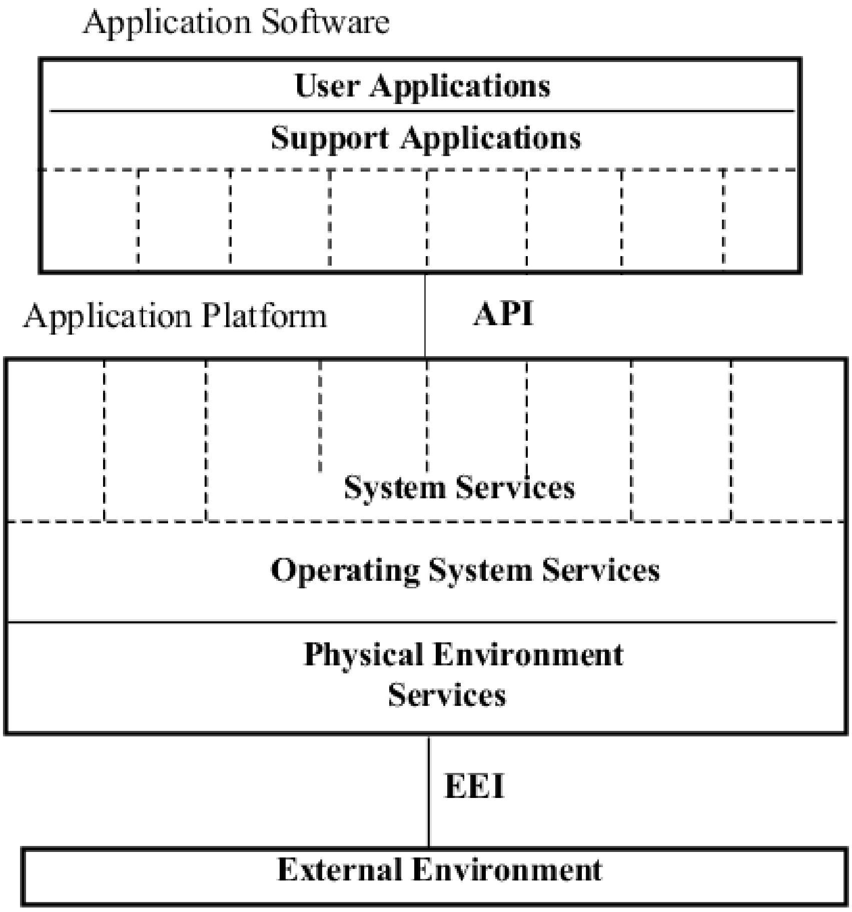


Figure A.2. NTRM Service View

011. The Application Platform Entity is structured in the following 12 Application Platform Service Areas:

- **User Interface Services.** These services define how users may interact with an application. The term user interface in this context means a graphical user interface (GUI). Standards are not only required for setting up and managing graphical windows, but also for the toolkit and generic 'look and feel'.

- **Data Management Services.** The management of data is central to most systems. To improve interoperability, data should be defined independently from the processes that create or use it, and should be maintained and shared among many processes.

- **Data Interchange Services.** These services provide support for the interchange of data between applications. They are designed to handle data interchange between applications on the same or on heterogeneous platforms.

- **Graphics Services.** These services provide functions required for creating and manipulating graphics.

- **Communication Services.** These services provide distributed applications support for data access and applications interoperability in heterogeneous or homogeneous networked environments.

- **Operating System Services.** These services are the core services needed to operate and administer the application platform and provide an interface between applications software and platform. Application programmers will use operating system services to obtain operating system functionality.

- **Internationalization Services.** Within the context of the NTRM, internationalization provides a set of services and interfaces that allow a user to define, select, and switch between different culturally related application environments supported by the particular implementation. Character sets and data representation services include the capability to input, store, manipulate, retrieve, communicate, and present data independently of the coding scheme used. This includes the capability to maintain and access a central character set repository of all coded character sets used throughout the platform.

- **System Management Services.** Information systems are composed of a wide variety of diverse resources that must be managed effectively to achieve the goals of an open system environment. While the individual resources (such as printers, software, users, processors) may differ widely, the abstraction of these resources as managed objects allows for their treatment in a uniform manner.

- **Security Services.** Different groups of individuals within and across the various NATO applications need to work with specific sets of data elements. Access to these sets of data elements is to be restricted to authorized users. Satisfaction of this requirement has traditionally been accomplished by the implementation of separate information systems. Organizations cannot continue to afford to implement separate information systems to satisfy this requirement, nor is it effective to require the user to change interface components every time the need arises to operate with a different restricted data set. Significant benefit will accrue when an individual information system can effectively support the needs of different groups of users and data sets.

- **Distributed Computing Services.** These services provide specialized support for applications that may be physically or logically dispersed among computer systems in a network, but yet wish to maintain a co-operative processing environment. The classical definition of a computer becomes blurred as the processes that contribute to information processing become distributed across a facility or a network. As with other cross-cutting services, the requisite components of distributed computing services typically exist within particular service areas.

- **Software Engineering Services.** The procedural aspect of an application is embodied in the programming languages used to code it. Additionally, professional system developers require methods and tools appropriate to the development and maintenance of applications.

- **Common C2 Applications Services.** These services provide the ability to view data (i.e., share) in a common way across the network. Common C2 Applications Services promote interoperability among diverse functional mission area domains and may be executed between both individual and multiple functional application domain areas.

012. In the NC3TA each of these service areas was further defined into one or more functional classes, with each class mapped to one or more standards. For example, communication services has a class called messaging with Simple Mail Transfer Protocol (SMTP) being one of the protocols used to support the communication service.

A.2.2. NCM

013. The NTRM provides the structural basis for defining the NCOE (NATO Common Operating Environment) Component Model (NCM). The NCOE provides the set of building blocks and guidance necessary for effective maintenance of open system design, development, implementation and integration. The NCM is shown below in Figure A.3.

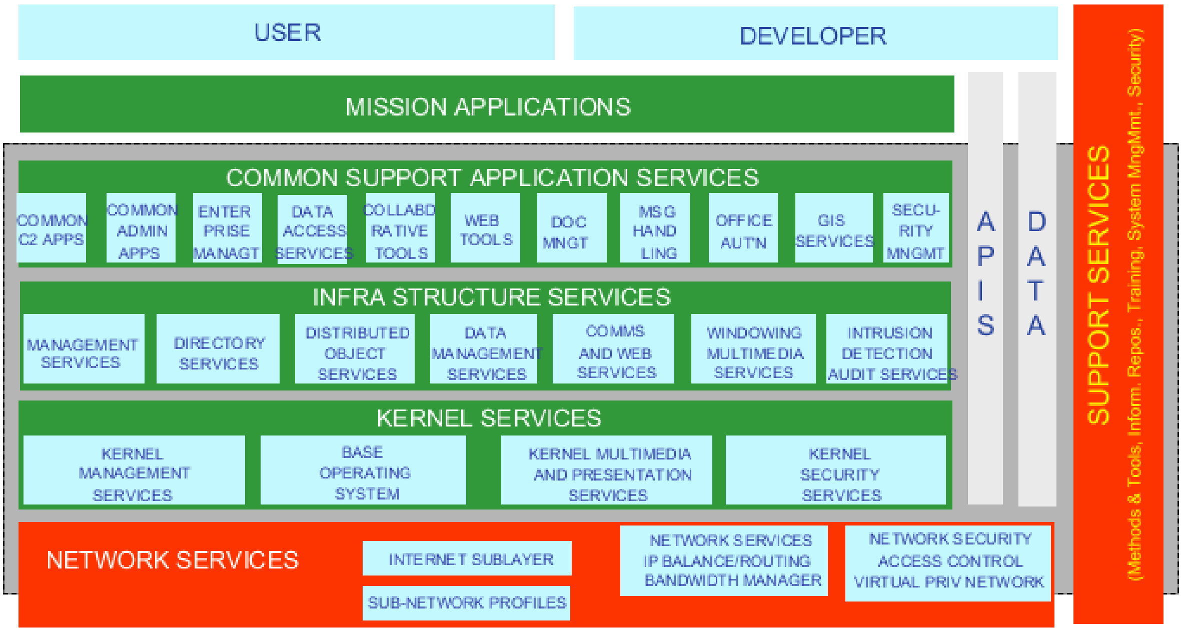


Figure A.3. NCOE Component Model

014. The principal components of the NCM include:

- **Network Services.** The NCOE Network Services constitute the basic interface between the platform and the underlying networking infrastructure and include the Internet sub-layer services.

- **Kernel Services.** The Kernel Services are that subset of the NCOE component segments, which are required for all compliant platforms. At a minimum, this sub-set would consist of the operating system, windowing software, security services, segment installation software and an executive manager.

- **Infrastructure Services.** Infrastructure services are those services that directly support the flow of information across NATO systems. Infrastructure services provide a set of integrated capabilities that the applications will access to invoke NCOE services.

- **Common Support Application Services.** Common Support Application Services provide services to process and view data in a common way (share data) across the network. The NCOE common support application services promote interoperability among various Mission Applications.

- **Application Programming Interfaces.** Applications are integrated into the NCOE through a common set of Application Programming Interfaces (APIs). The APIs are invoked by the applications and services as required.

- **Data Component Definition.** The data component refers to the way in which data is taken into account in the NCOE and is related to the main components of the NCOE (Common Support Application Services, Infrastructure Services, Kernel Services) and even, out of NCOE components, in the strictest sense, to Mission Applications.

- **Support Services.** The NCOE Support Services include Methods and Tools, Information Repository, Training Services, System Management and Security.

A.2.3. OSI Protocol Stack and Services

015. The OSI protocol stack (as well as the Internet stack) is based on the concept of layering as depicted below (Figure A.4):

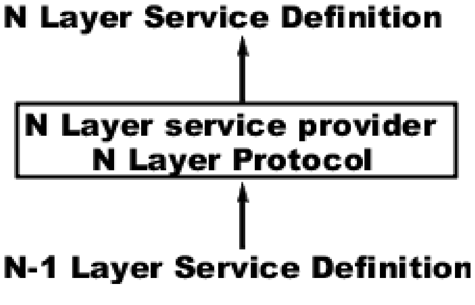


Figure A.4. Service Layering

016. A key aspect of the layer principle is layer independence. The service user is not concerned with the specifics of the protocol used by the service provider to provide the service. The user of the N Layer service uses defined service primitives to use the services provided by the N Layer. The N Layer uses the N Layer protocol and services provided by the N-1 Layer to provide the N Layer services. The N Layer protocol definition describes the rules which each N Layer service peer uses when communicating with its other service peers. As long as there are no changes

to the service interface, the service user at that layer is completely unaffected by changes in the underlying layers or by the protocol used within the layer. Protocol layering is key to the development the profiles contained in the NISP.

A.3. SOA ENVIRONMENT

017. Volume 1 Annex E of the NISP also describes SOA[2]. Following are some of the SOA highlights describes in Volume 1:

- Service Oriented Architecture (SOA) is a paradigm for organizing and using distributed capabilities that may be under the control of different ownership domains.
- Visibility, interaction, and effect are key concepts for describing the SOA paradigm.
- For a service provider and consumer to interact with each other they have to be able to 'see' each other. Visibility needs to be emphasized because it is not necessarily obvious how service participants can see each other to interact.
- The service interface is the means for interacting with a service. It includes the specific protocols, commands, and information exchange by which actions are initiated that result in the real world effects as specified through the service functionality portion of the service description.
- SOA is commonly implemented using Web services, but services can be made visible, support interaction, and generate effects through other implementations.

018. The SOA concepts are best depicted in Figure A.5.

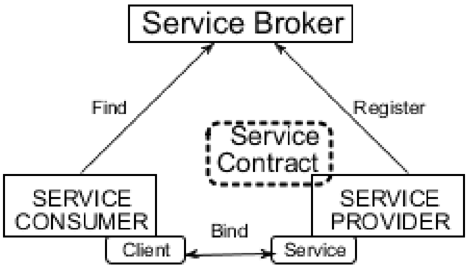


Figure A.5. Conceptual Roles and Operations of a SOA

- Service: The means by which the needs of a consumer are brought together with the capabilities of a provider.
- Service Description: The information needed in order to use, or consider using, a service.
- Service Provider: Makes the service available and publishes the contract that describes the interface to the service and registers the service with a service broker

[2] OASIS Reference Model for Service Oriented Architectures

- Service Consumer: Queries the service broker and finds the desired service.
- Service Broker: Gives the service consumer directions on where to find the service and its service contract.

A.4. SERVICE AND INTEROPERABILITY POINTS

019. This section deals with service and interoperability points connecting domains under separate management and control, such as National or NATO Systems. In a platform oriented environment two parties agree "a priori" on the services and standards at static interoperability points. In A SOA environment provider parties advertise a service with a public interface which may be discovered by a consumer requiring the use of that service.

A.4.1. Platform Centric Service and Interoperability Points

020. A typical example of a service and interoperability point implementation in a platform centric environment is the Information Exchange Gateway (IEG) shown in Figure A.6[3].

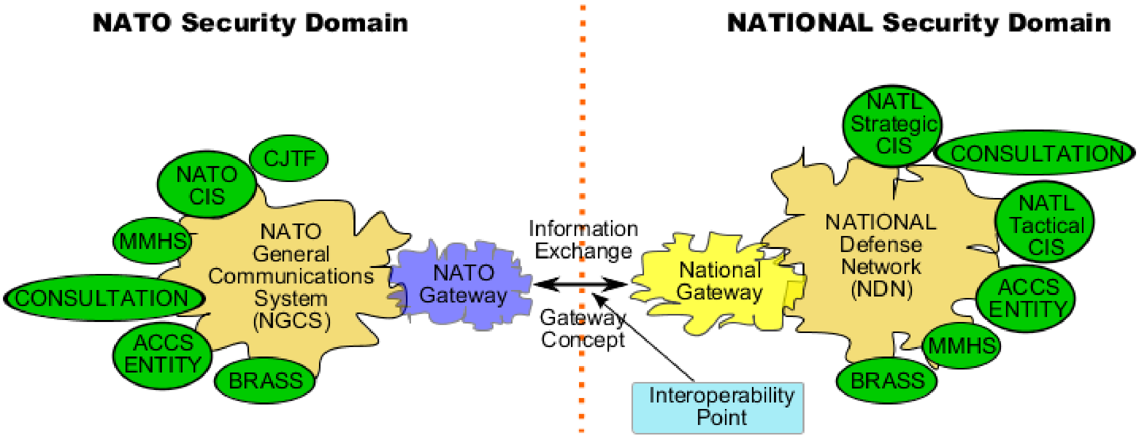


Figure A.6. NATO IEG Concept

021. The IEG Concept is designed to satisfy information requirements by providing information exchange capability between different CIS communities, while protecting the same CIS communities. The IEG Concept provides:

- Information Exchange Services (IES) for the exchange of information between NATO and other CIS communities, and
- Boundary Protection Services/Boundary Protection Component (BPS/BPC) to protect NATO and national CIS communities by implementing the stated security principles like self protecting node principle and defence in depth to support the security objectives of confidentiality , integrity and availability.

[3] AC/322-D(2005)0054

022. The Interoperability Point depicted in Figure A.6 may be considered both a SIOP and a SIP. The services at this interface point are taken form the services described in the NTRM. Example services are messaging and document exchange. These services are initial services supported by the IEG. Interoperability for these services is obtained by using the mandatory protocols (e.g. SMTP) and profiles contained in Volume 2 (the NISP profiles can be considered equivalent to SIPs).

023. The communications SIOPs and SIPs (e.g. tactical data links, IP) described in the NNEC FS are compliance with the NTRM.

A.4.2. SOA Service and Interoperability

A.4.2.1. SOA Characteristics

024. One of the highlights of a SOA is the degree of documentation and description associated with it. The use of a service without the service consumer needing to know the details of the service implementation, the service description makes available critical information that a consumer needs in order to decide whether or not to use a service. In particular, a service consumer must possess the following items of information:

- That the service exists and is reachable;
- That the service performs a certain function or set of functions;
- That the service operates under a specified set of constraints and policies;
- That the service will (to some implicit or explicit extent) comply with policies as prescribed by the service consumer;
- How to interact with the service in order to achieve the required objectives, including the format and content of information exchanged between the service and the consumer and the sequences of information exchange that may be expected.

025. The service description is part of the service contract depicted in Figure A.5. A service contract needs to have the following components[4]:

- Header
    - Name - The service name. Should indicate in general terms what it does, but not be the only definition
    - Version - The version of this service contract
    - Owner - The person/team in charge of the service
    - RACI

[4] Wikipedia

        - Responsible - The role is the person/team responsible for the deliverables of this contract/service.
        - Accountable - Ultimate Decision Maker in terms of this contract/service.
        - Consulted - Who must be consulted before action is taken on this contract/service. This is 2-way communication. These people have an impact on the decision and/or the execution of that decision.
        - Informed - Who must be informed that a decision or action is being taken. This is a 1-way communication. These people are impacted by the decision or execution of that decision, but have no control over the action.
    - Type - This is the type of service to help distinguish the layer it resides.
        - Data
        - Process
        - Functionality
        - Presentation
- Functional
    - Functional Requirement (From Requirements Document) - Indicates the functionality in specific bulleted items what exactly this service accomplishes. The language should be such that it allows test cases to prove the functionality is accomplished.
    - Service Operations - Methods, actions etc. Must be defined in terms of what part of the Functionality it provides.
    - Invocation - Indicates the invocation means of the service. This includes the URL, interface, etc. There may be multiple Invocation paths for the same service. We may have the same functionality for an internal and external clients each with a different invocation means and interface. Examples:
        - SOAP
        - REST
        - Event Triggers
- Non-Functional
    - Security Constraints - Defines who can execute this service in terms of roles or individual partners, etc. and which invocation mechanism they can invoke.

- Quality of Service - Determines the allowable failure rate.
- Transactional - Is this capable of acting as part of a larger transaction and if so, how do we control that?
- Service Level Agreement - Determines the amount of latency the service is allowed to have to perform its actions.
- Semantics - Dictates or defines the meaning of terms used in the description and interfaces of the service.
- Process - Describes the process, if any, of the contracted service.

026. SIOPs and SIPs for SOA be compatible with Figure A.5 and will have to be compliant with NATO agreed standards for service descriptions and contracts. Since it is expected SOA services will be implemented with web services, SIOPs and SIPs will be implemented with web service standards (e.g. WSDL, UDDI, XML, SOAP).

A.4.2.2. Recommendations

027. To clarify the meaning of service interoperability in platform centric and SOA centric environments it is recommended that:

- For platform centric environments the terms Interoperability Point (IOP) and Interface Point (IP) be used, and
- For SOA environments the terms Service Interface Point (SIP) and Servcie Interoperability Point (SIOP) be used.

This page is intentionally left blank

B. NATO, NATIONAL AND INDUSTRY NEC IMPLEMENTATION APPROACHES

B.1. NATO NETWORK ENABLED EFFORTS

B.1.1. Transformational Areas

028. In August 2004 the NATO Strategic Commanders described the following framework for transformation shown in Figure B.1 which is structured around Transformation Goals and Objectives. A central tenet of this Bi-Strategic Command (Bi-SC) strategic vision is that future operations will be effects-based in that they will involve all instruments of Alliance power, political, diplomatic, economic and military, exercised in an integrated fashion to create a desired effect in order to achieve a strategic objective. Regarding the transformation goals, NATO must be able to achieve Coherent Effects, Decision Superiority and Joint Deployment and Sustainment. In order to achieve these goals, the NNEC Vision & Concept identified specific areas where NATO needs to conduct research and develop concepts to improve capabilities. These are entitled Transformation Areas. NNEC is one of them and is seen as a key enabler to nearly all the others.

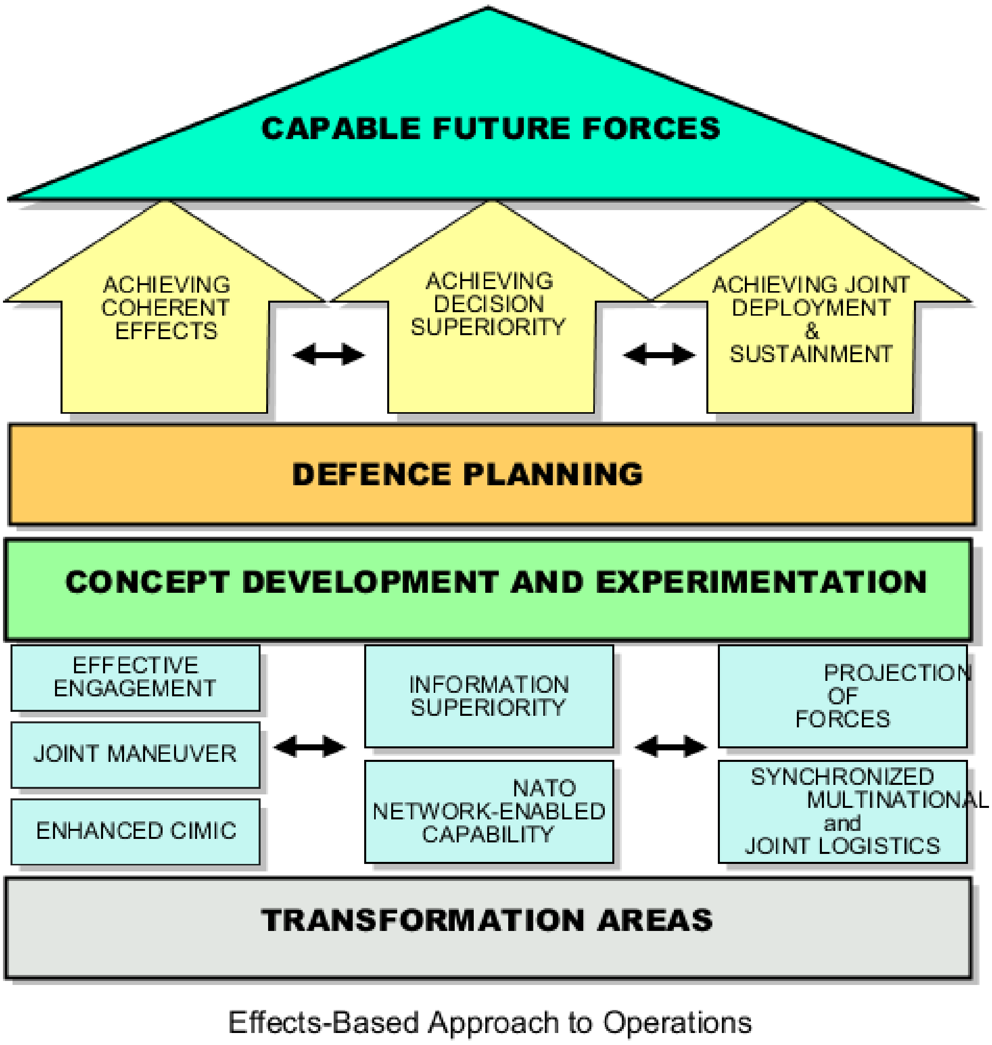

CAPABLE FUTURE FORCES
ACHIEVING COHERENT EFFECTS
ACHIEVING DECISION SUPERIORITY
ACHIEVING JOINT DEPLOYMENT & SUSTAINMENT
DEFENCE PLANNING
CONCEPT DEVELOPMENT AND EXPERIMENTATION
EFFECTIVE ENGAGEMENT
JOINT MANEUVER
ENHANCED CIMIC
INFORMATION SUPERIORITY
NATO NETWORK-ENABLED CAPABILITY
PROJECTION OF FORCES
SYNCHRONIZED MULTINATIONAL and JOINT LOGISTICS
TRANSFORMATION AREAS
Effects-Based Approach to Operations

Figure B.1. Framework for Transformation

B.1.1.1. NATO Network Enabled Capability (NNEC) Feasibility Study

029. In November 2002 the NATO C3 Board (NC3B) agreed that there was a need to develop NATO NEC. NNEC is based on adapting national initiatives like the U.S. Network Centric Warfare (NCW) and the UK Network Enabled Capability (NEC).

030. In January 2004, 12 Nations, through the NC3B, sponsored the NNEC Feasibility Study. Version 1.0 of the Feasibility Study was delivered to the sponsoring nations and version 2.0 was delivered to all NATO nations in October 2005. In December 2005 the NC3B endorsed the NNEC Feasibility Study recommendations and its release to Partners and industry.

031. The aims of the Feasibility Study are listed below:

- Support further development of the NNEC concept
- Identify types of C2ISR capabilities required to enable NATO Network Centric Operations
- Develop a strategy and roadmap for realization of a Networking and Information Infrastructure (NII)

032. Key conclusions in the Feasibility Study included the need for a highly interconnected CIS to support future operational needs of the Alliance; and the need for a NII based on a federation of NATO and National systems. The NISP must be structured to support the development of the NII.

033. Another major conclusion of the NNEC FS is that change is a constant. In order to make change manageable a model for NNEC Maturity Levels has been developed along the structure of four maturity stages. The NISP must be restructured to align with these stages. As a first step alignment with four stages was not yet feasible.

034. The NNEC FS also concluded that the required flexibility for federating NATO and national systems into an NII can only be reached by adopting the service-oriented approach. In practice, this leads to a need for identification of a coherent and comprehensive services framework, including strict interface definitions. The framework should be provided as a result from architecture work (typically Overarching Architecture). The solution patterns for services and interface definitions need to be developed using time-phased standards and technology profiles. Again the NISP needs to be structured to provide the required inputs to this process.

B.1.1.2. Networking and Information Infrastructure / Federation of Systems

035. The Networking and Information Infrastructure (NII) strategy assumes that the NII will be implemented as a Federation of Systems (FoS), involving the use of Service Oriented Architectures (SOAs).

B.1.1.2.1. Networking and Information Infrastructure (NII)

036. The NII can be viewed as an evolving, multi national military "Intranet" - a "Federation-of-Systems" (FoS) - interoperating seamlessly to provide information to anyone, anywhere, anytime; if appropriately authorised. Similar to the Internet-driven Information Age transformation that is occurring world-wide, so will the NII support the transformation of mission capabilities of NATO nations, NATO, and coalition partners. The NII can be defined as: "*A federation of systems, formed by the synergistic amalgamation of a dynamic set of globally interconnected, multi-national, autonomous systems, each comprised of networking and information infrastructure components, providing information capabilities, associated processes, and personnel for collecting, processing, storing, disseminating and managing information to authorised users on demand, on an end-to-end basis.*"

B.1.1.2.2. Perspectives

037. The NII can be looked at from three points of view.

1. At a conceptual level the NII captures the vision of the future shared information environment for NATO nations, NATO, and coalition partners.

2. At the planning level, the NII architecture defines the structure of NII systems and components, their relationships, and the principles and guidelines governing their design, operation and evolution over time. The NII Architecture is used to determine interoperability and capability requirements, advance the use of commercial standards, accommodate accessibility and usability requirements, and implement security requirements both within NATO and between federated systems.

3. At the physical level and in the day-to-day environment, the NII will provide information and communication services vital to the effective conduct activities and is the foundation for allowing NATO nations and NATO to achieve their network enabled capabilities.

B.1.1.2.3. Composition and Ownership

038. It is thought that the NII will be comprised of both owned and leased communication an computing systems, information services, software, data, security services, and other associated capabilities necessary to achieve Information Superiority for the Nations and organisations contributing to the NII. The NII includes service interfaces between participants including NATO nations, NATO agencies, coalition partners, and non-NATO users including potential participation from Non-Governmental Organisations (NGOs), civilian police, emergency services, and local governmental groups. The NII provides capabilities from all operating locations (e.g. government facilities, headquarters, posts, camps, stations, facilities, mobile platforms, deployed sites, and agencies).

039. The NII will also be made of service-based systems provided by multiple sovereign nations as well as NATO itself and will operate on a co-operative basis. It encompasses not only the physical and personnel resources necessary to plan, implement, and operate the actual infrastructure, but also includes the necessary agreed processes to ensure continued coherent evolution and effectiveness of the NII. The infrastructure will be a dynamic heterogeneous entity with configuration changing with the environment and technological advances. This environment is considerably different from a national-only environment and will therefore require greater emphasis on standardisation and interoperability, not only for equipment but also in management methodologies including Service Level Agreement (SLA) management.

040. Inclusion of a service-based system within the NII does not imply ownership or control by any entity other than the contributor. Contributed systems are independently managed and controlled by their owners within the framework of the Federation-of-Systems (FoS), which is a concept explained later in the Chapter. However inclusion of a system within the NII does imply that the contribution will comply with agreed NII community policies, security requirements, and procedures.

B.1.1.2.4. Strategy for the Networking and Information Infrastructure (NII)

041. The strategy for developing the networking and information sharing aspects of NNEC focuses on the 'joining together' of networking systems and core information systems from NATO and NATO nations, to form a Federation-of-Systems (FoS) capability that implements the NII. The FoS concept is used here to refer to a set of different systems, which are not centrally managed, but are so connected or related so as to produce results beyond those achievable by the individual systems alone. In effect, the NII is to be made up of a combination of national Networking and Information Infrastructures segments and a NATO Networking and Information Infrastructure (NNII), which together will provide capabilities that no one system can provide by itself. The need for the NII is consistent with the tenets of NNEC, which have evolved from earlier concepts[1], which are outlined below:

- A robustly networked force improves information sharing.
- Information sharing enhances the quality of information and shared situational awareness.
- Shared situational awareness enables collaboration and self-synchronisation, and enhances sustainability and speed of command.
- These, in turn, dramatically increase mission effectiveness.

B.1.1.2.5. Federation-of-Systems (FoS)

042. Definition and term: **Federation-of-Systems (FoS)**: *A Federation-of-Systems is a System-of-systems but one managed without central authority Constituent systems are independently managed and have a purpose of their own.*

B.1.1.2.6. Federation-of-Systems Approach

043. The strategy for developing the networking and information sharing aspects of NNEC focuses on the "joining together" of networking systems and core information systems from NATO and NATO nations, to form a Federation-of-Systems (FoS) capability that implements the NII. The FoS concept is used here to refer to a set of different systems, which are not centrally managed, but are so connected or related so as to produce results beyond those achievable by the individual systems alone. In effect, the NII is to be made up of a combination of national Networking and Information Infrastructures segments and a NATO Networking and Information Infrastructure (NNII), which together will provide capabilities that no one system can provide by itself.

044. The Internet is the best known example today of a Federation-of-Systems. There is no central control, and participation is through collaboration and co-operation to meet the objectives of

[1] Network Centric Warfare : developing and leveraging information superiority /David S. Alberts, John J. Garstka, Frederick P. Stein. p. cm. -- (CCRP publication series) Includes bibliographical references.ISBN 1-57906-019-6

the federation. A FoS is characterised by a greater degree of autonomy, heterogeneity, and distribution than is found in a system-of-systems (SoS) approach, which also involves the "joining together" of systems, but allows for a degree of centralised control. Although it is thought that the NII will operate as a FoS, it is also possible that there may be times when nations may wish to allow for the centralised management of segments of their systems, in order to meet shared operational objectives. It is important that the implementation of the NII allow for this possibility.

045. The need to support national autonomy in the implementation and operation of the NII forces the adoption of participation models capable of supporting a flexible, adaptable approach to participation in the NII. It is clear that the successful implementation of the NII will require architectures, technical solutions, and operational procedures beyond that which any single nation will require or can provide.

B.1.1.2.7. Service Oriented Architecture

046. Definition and term: **Service Oriented Architecture (SOA)**: *An architecture within which functions are defined as independent services with well-defined interfaces which can be called separately or in defined sequences to form business processes. The interface is the focus and is defined in terms of the required parameters and the nature of the result when the service is invoked. A SOA enables services to be published, discovered and utilized.*

B.1.1.2.8. Strategy for Future C3 Capabilities

047. The strategy for developing future C3 capabilities is based on the Capability-Based Planning (CBP) approach. A capabilities-based paradigm focuses more on how an adversary might fight than on who the adversary might be and where a war might occur. It requires us to identify capabilities that military forces will need to deter and defeat a particular type of adversary. This approach involves a functional analysis of operational requirements, leading to the identification of capabilities required to accomplish a mission. Once the required capability inventory is defined, the most cost effective and efficient options to satisfy these requirements are sought. This process involves the mapping of operational capability requirements to a supporting set of system functional requirements, which identifies the system capabilities required to support a mission.

048. The CBP methodology builds on the CBP approach introduced as part of the 2003 NATO Defence Requirements Review processes and the CBP planning approach used within the NATO Overarching Architecture and provides a new approach to conducting NATO C3 planning.

049. While the concept of CBP is key to the identification of required system functionality, the concept of Service Oriented Architectures (SOAs) is key to meeting those requirements and is an essential part of the overall strategy.

050. SOAs provide a flexible modular approach for implementing system functional requirements in the form of services. This strategy for developing future C3 capabilities responds to the need to for a flexible, modular approach for meeting future Consultation and C2 requirements.

051. The use of SOAs has emerged as a major trend within the commercial sector and among nations developing NNEC type capabilities, because of the flexibility they provide in sharing information and information processing capabilities. SOAs provide mechanisms for using existing information services as well as providing a basis for developing new more advanced information services, utilising existing services. Such mechanisms will allow many Consultation and C2 needs to be satisfied by linking together existing information services in a modular, flexible fashion that can be readily adapted to changing operational needs. The flexibility provided through the use of SOAs is particularly well suited to supporting the needs of coalition based Network-Centric Operations.

B.1.1.2.9. Key Information and Integration Elements

052. The Information and Integration component of the NII is characterised by the use of Service Oriented Architectures to expose software functions as consumable services that can be discovered and invoked across the network. The use of SOAs ease application and data sharing and provide a flexible mechanism for reusing existing services to enable the development of new, value-added information services.

053. A primary goal of the SOA approach is to make information resources available to all consumers on the network and support the efficient discovery and delivery of that information to the consumer.

054. The use of the SOA approach requires that we adopt a common Net-Centric Data strateg to ensure that we make information visible, accessible, understandable and interoperable with other sources of information. Trusting the information we get and trusting that the information we supply will be handled correctly will be a key success factor. The ability to provide flexible, secure, role-based, information access that can be quickly configured to support changing policy is foundational to the long-term success of the NII.

055. Realising the benefits of the SOA approach will require that we agree on a standardised set of foundational services covering such areas as service discovery security, metadata management, identify management, service management and mediation.

B.1.1.2.10. SOA, Loose Coupling and Federated Services

056. The need for improved system interoperability is clearly spelled out in the latest versions of the NATO Interoperability Management Plan (NIMP) and NATO Architecture Framework (NAF).

057. In order to achieve the principle of flexible interoperability, a change of focus is required: from the idea of standalone, stovepipe systems (i.e. platform-oriented) to the idea of shareable, universal information (i.e. service-oriented).

058. This "service oriented" architectural view indicates that rather than information being inextricably tied to a particular system, it should be available to all who need it (and have rights

to it). Information would still have "consumers" and "producers" but the assignment of these roles would be dynamic and would cross system boundaries.

059. The guiding principle of information services is the adoption of service oriented architecture (SOA) and where appropriate, the implementation of web services.

060. A simple definition of an SOA is "a broad set of concepts that enable units of functionality to be provided and consumed as 'services'. This essentially simple concept can and should be used, not just for web services, but also at each tier of the architecture, in order to compartmentalise and provide flexibility". For example, an SOA allows the use of 'thin clients' by tailoring the provided services to the needs and capabilities of a consumer. This allows a more flexible use of hardware resources and is more cost-effective (i.e. a reduced number of software licenses).

061. In an SOA, loosely coupled systems view one another as "services", accessible via a standard interface without knowledge of the underlying implementation of the service. A "registry" manages and directs the interactions between services. Data is exchanged in a common format using standard protocols, which helps to ensure compatibility. All this allows user-to-system or direct system-to-system interactions that have not previously been generally feasible.

062. The concept of every system viewing others as "services" in a loosely coupled manner is coherent with the concept of NNEC being a FoS. From an SOA perspective, it means that IIS is to be thought of as a federation of services, where any NATO or national information system will be autonomous and provide specific services by means of implementing a standardised service interface.

063. Thus, it will be necessary not only to define the standards to regulate the implementation of such interfaces but also a set of service interoperability points that will facilitate the interoperation of similar ser ices provided by different nations in a seamless manner.

B.1.1.3. NNEC Strategic Framework

064. In June 2005 the Military Committee tasked Allied Command Transformation (ACT) to develop the NNEC Strategic Framework as a means of describing NNEC developmental activities. The Strategic Framework will consist of the following documents: NNEC Vision and Concept, NNEC Roadmap(s), NNEC Business Case(s), Compendium of NNEC related Architectures, and detailed Plan. Although the NNEC Feasibility Study was not identified as Strategic Framework document, it is recognized as providing the technical foundation for NNEC through the NII. The development of the Strategic Framework is depicted in Figure B.2 below.

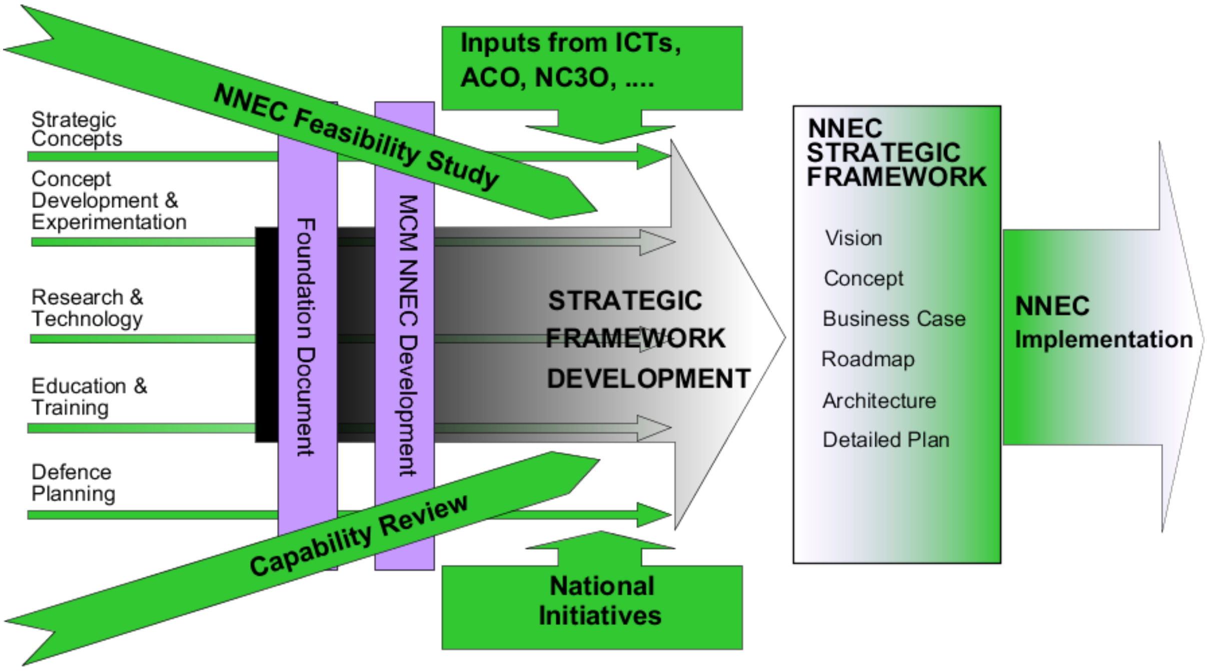


Figure B.2. NNEC Strategic Framework

B.2. NATIONAL NETWORK ENABLED EFFORTS

065. As stated previously NNEC is based on national initiatives. Many NATO and Partner Nations have initiated network enabled programs in their respective nations. NNEC will be a federation of NATO and National service based systems, with the Nations contributing around 90 percent of the capabilities, it is important that the standards and profiles described in the NISP take into account the ongoing developments in the nations.

B.2.1. Finland

066. The C4 development started in Finland in the 1980's. At that time, all systems were developed separately for the Army, Navy, and the Air Force, thus resulting in a situation with a large number of distributed stove-piped systems. Although these systems have served us well in the past, their lifecycle is coming towards an end, and new systems need to be developed to serve our current and future needs. We face the same challenges as many other militaries as we realize that it is not affordable to develop new systems on top of the old ones by patching and bridging gaps. Furthermore, there is a limit to which technical, data, and application integration can take us. Therefore we have decided to give up most of our old systems by 2010. This decision became the starting point for the development of our IC4I and forms the basis for our Network Enabled Defence (NED), which comprises a domestic roadmap to which crisis management is aligned.

067. The Finnish view on NED is that it is a working title for a concept that uses the principles of Network Enabled Capabilities for providing total defence (also called homeland security). Fin-

land has a long tradition in total defence, where the armed forces cooperate with other governmental agencies and industry to secure the functions vital to the society. NED is defined to describe how future networks with improved and integrated information and weapon systems can enable command and control in joint and territorial operations with multiple partners. The basic requirement in our system development is interoperability with the international development. Thus, there is no significant difference whether the crisis is in Finland or somewhere else, nor is there a technical difference whether the coalition is lead by the UN, EU, NATO or a nation.

068. The Finnish approach to NED is very pragmatic and based on incremental development. We place high emphasis on innovation opportunities that are technically feasible, cost-efficient, and organizationally achievable. NED is not seen as a goal, but as a journey in the right direction together with partners. Especially industry partnerships are considered valuable, as they have the research and production capabilities needed to turn innovations into solutions and applications - again, real implementations, not just visions and theories. One example of such partnership is the establishment of a Network Centric Operations Centre of Excellence (NCO CoE) together with IBM in Finland.

069. The first task that we did was to rationalize system development in order to cut the number of overlapping systems. This lead to system centralization, and the idea of "the network as the computer" emerged, which in practise means that thin clients capable of doing local processing are used, but the applications and data storage are performed via the network. We also decided to take a leap into a SOA-based approach. The two main issues in this first phase are the reliance on COTS as well as our architecture development. In our architecture development, we have found the usage of overarching and reference architectures as drivers for change, and ended up with an overarching architecture and two reference architectures (C4I and administrative). In the near future, there will be two additional reference architectures, which are shared with other agencies for collaboration purposes. As for using COTS, we have taken a very pragmatic approach in building and deploying our systems. For example, for voice communication we have developed a TETRA based system, which was gradually deployed in Finland as well as in international exercises and operations, such as in the Finnish battalion in Kosovo (2002), the Nordic PfP Exercise (2003), MN Brigade (C) in Kosovo (2003), and MN TF Althea (2004). During the years 2001-2004 we developed civilian crisis management systems, which were based on the idea that whatever can be deployed in our national defence can be used in international operations as well. Another developed COTS product was the Deployed COTS Network (DCN), which connects virtually any digital and analogue network together. In less than four months we procured, tested trained, and deployed a brigade level DCN to KFOR.

070. The second step involves a functional change and is concerned with information sharing. At this stage, SOA is used as the basis for the architecture. At the moment, we have three main experiences developed or in development for the second phase. One is the Network Enabled Operations Centre (NEOC), where we tested the centralization of information powerhouses into the core network. The concept of information powerhouses basically means that all information is centralized into a few server hotels, which are connected to the backbone and accessible through the access networks. The demonstrator proved increased survivability, better situational awareness, and brought new processes and a new C4 structure. The second issue is SECNET,

the interagency networking concept. This is still under development, but we are getting there in a pragmatic way, developing services that can be shared between authorities. The third concept is METO, which is a Sea Surveillance Information System (SSIS) shared by the Navy, Border control, and Maritime traffic safety, and thus coordinated by the defence ministry, ministry of transportation, and the interior ministry.

071. In the third phase, we are concerned with a cultural change that involves collaboration based on the SOA approach. To mention two examples, we have build proof-of-concepts for a joint Common Operational Picture (COP) service. In the first PoC, we tested the scalability of having a "mother database" with all information. This database was reachable from the Defence Staff through a common portal. Information was collected from various local environments, which all had a picture of their own environment. The PoC showed us that the technology worked. In the second PoC, we tested what we call the "child's architecture", where some of the local environments are detached from the "mother environment". The idea is that the detached unit is able to operate on its own even without a network connection, seeing only its own environment. However, as the network connection is resumed, it should be possible to rejoin to the common environment. The focus was on the scalability of SOA, information handling when coming back online, and integrity issues. In this PoC we scaled down to brigade and battalion level, but it is possible to scale it further down to team or even individual fighter level.

072. We have learnt many lessons on our still ongoing journey towards NED. We believe that our pragmatic and incremental approach together with strong industrial partnership, where development and deployment lifecycles can be brought down to 18 months, is a good way to go ahead. Collaboration with NATO in architectural development, using the idea of overarching and reference architectures, and working together on open service interface descriptions has proven valuable. Also testing the systems in exercises and deploying them on the field in various operations have given us a lot of experience.

073. Still, several challenges remain. The main challenge is IA, which often is a hot potato that nobody wants to address. This is the next challenge that we will really focus on solving. The IA challenge is especially difficult at the moment due to the cultural change related to the transition from "need to know" to "need to hide" and "duty to share". Whereas communication security has been addressed by various security protocols, network and content security are still open questions. For example, the concept of CBIS/MLIS has suffered from a standstill for several years.

074. Another challenge is the difficulty in integrating differ nt approaches from NATO, EU, the government and industry. However, we are hoping that the industry will help us solve the problem, and hence we will continue with our strong partnership approach. Furthermore, future services are still unknown, thus leading to "ad hoc system design". The best we can do at the moment is to try to develop a common and generic environment, in which to fit future services in a standard way. Also decomposing current systems to services provided by applications using core and common services is a challenge.

075. The biggest challenge, perhaps, is not technical, but cultural. The user needs to feel "a return of investment" for his efforts to learn new ways of doing business. Training and educating users

is one thing, but the real challenge is to get people to truly adopt the concept of collaboration in a network-centric environment using unified process workflows.

B.2.2. France

B.2.2.1. Background

076. The objective of the present document is to describe the French orientations to rationalize the interconnection between the French CIS and NATO CIS.

B.2.2.1.1. France involvement in NATO force structures

077. France is involved in many NRF rotations:

- 2006 : NRF7 French participation in the Eurocorp at LCC level
- 2008 : NRF10 MCC
- 2008 : NRF11 LCC
- 2009 : NRF12 JFACC
- 2009 : NRF13 CJSOTF
- 2010 : NRF14 MCC and LCC (Eurocorp)

078. Besides, France has already 2 HRF HQ located in France: a maritime HQ in Toulon (southern part of France) and a LCC HQ in Lille (northern part of France).

079. Each time France is on alert for HRF or NRF purposes, it is mandated to setup a special interconnection between NATO HQ and these headquarters. In order to optimise those interconnections from a personnel perspectives and a cost point of view (equipments, leased lines..), the intention is to setup a permanent network. Furthermore, this permanent network will facilitate the exchange of information with NATO and will enable France to be in line with the NNEC perspectives to offer the available services to all the military users.

B.2.2.1.2. French participation in NRF5

080. France has participated as the JFACC during the NRF5 rotation. That was the starting point of the rationalisation work which is made in the logical continuity from this participation. The Figure B.3 displays the roles played by the French units and the information systems involved during the certification exercise ALLIED ACTION 05.

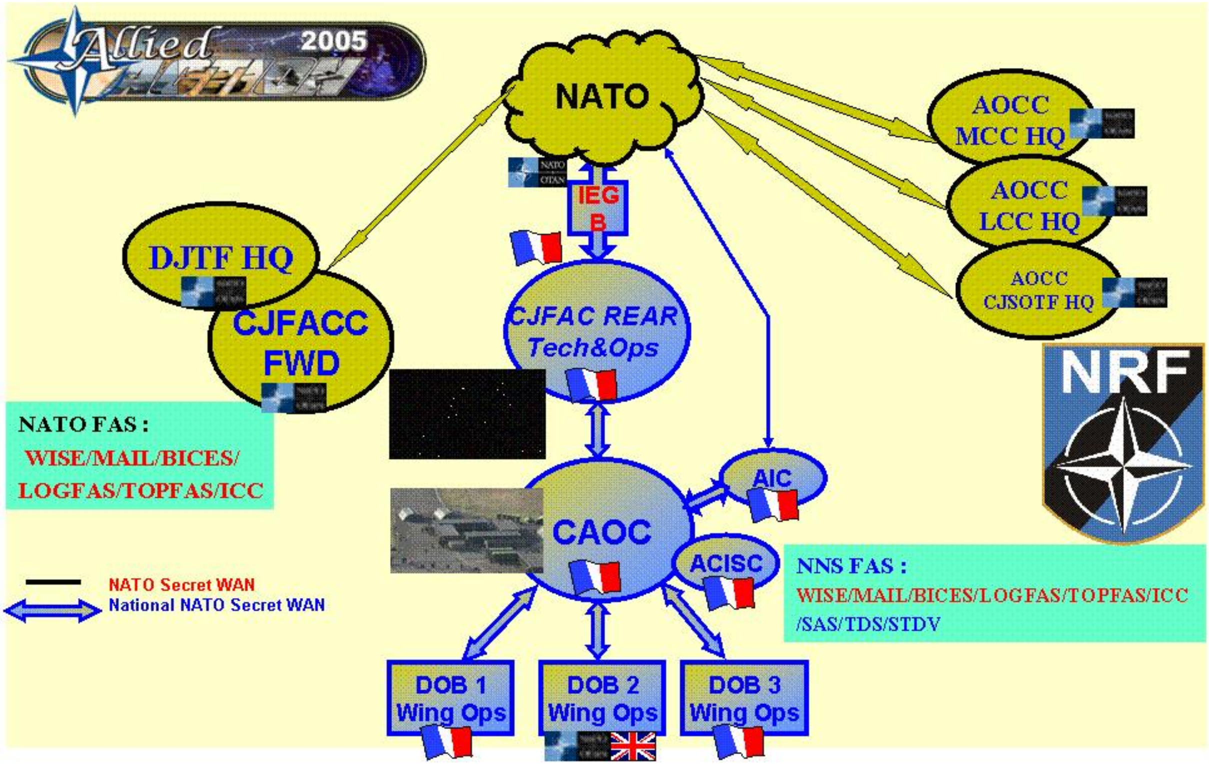


Figure B.3. French participation in Allied Action 05

081. The French systems used during this exercise were the following:

- SCCOA (command and control system for air operation)
- Document management system
- STRADIVARIUS : French integrated air picture
- IRIS : AdatP3 formatting tool
- TDS : friendly force awareness
- Messaging system
- Videoteleconferencing

082. The Lessons learnt from NRF5 participation can be sorted in three points:

- First of all, from the user's point of view, for maximum efficiency it would be easier to have only one workstation to access directly to the right information instead of using a swivel chair and disquettes (or USB keys). Another aspect of this point of view is to adopt the maxim "train as you fight", and as such it is highly desirable to have the ability to use NATO and French tools on the same network.

- From CIS engineers' point of view, the complexity the Information Exchange Gateway (IEG) imposes the need for a specific CIS team dedicated to the support of this gateway. It is not easy to deploy and configure it. On the other hand, using only one IP network between NATO and France has eased the overall deployment. The compromise that should be made is to avoid the deployment of those gateways on theatre where the skilled human resource is rare.

- From the interoperability point of view, it is much easier to have only one network to enable people to access the services they need.

B.2.2.2. Situation today

083. The actual situation is the result of successive deployment made when the different systems were available. Each time the participation of a French unit was required an extension of the NATO system was establish to connect them. The result is that each functional system has added direct link and specific network to enable the required exchange of information. The number of transmission links has grown and equally the number of gateways and crypto devices. One of the links might be overloaded at a period of the day but we are not able to use another link that might be free at the same time. The teams in charge of these equipments have no centralized management tool and no possibility to improve the quality of service for the different communities of users.

B.2.2.3. Optimization and rationalization of NATO links

084. The first step proposed is to rationalize the transmission links. For the moment, direct connections between each French location to the NATO network have been established. Hence the number of leased lines is important and to allow a communication at NATO secret level between Brest and Toulon for NATO purposes the communication is done via SHAPE. The optimization is not done on the different links and the level of redundancy for each location is relatively low. The proposed way to improve the actual situation is to:

- Have 2 BME for permanent purposes (1 NATO Point Of Presence close to Paris and 1 for ACCS)

- 1 BME for each HRF headquarters

- Establish 2 transmission links between Paris, Lyon and NATO

- Create links within the French network to avoid having to go back to the NGCS for communications between 2 sites, working for NATO purposes, located in France.

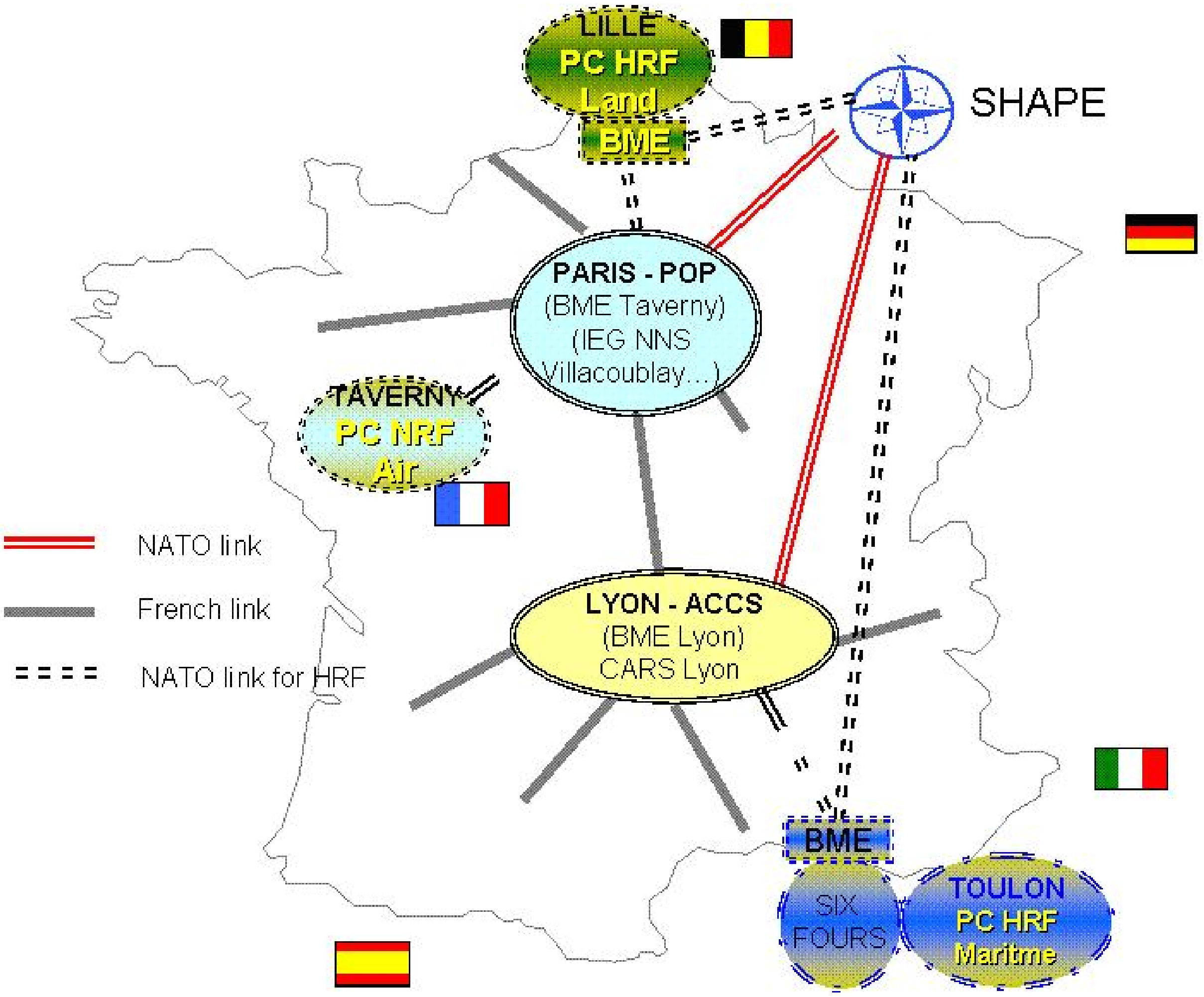


Figure B.4. Future Interconnection

B.2.2.4. Setting up of a unique logical network

085. The main objective for the NATO Secret network (FR NNS) is to give the opportunity for the end-user to have only one computer on his desk accessing all the needed services. The first step is to take into account the messaging and webaccess services to enable the access to ICC, MCCIS, BICES-webserver, WISE and CRONOS. All the "supporting" services like DNS, PKI, NTP will be available for the different systems. The flows of information between the national side and the NATO sides will be filtered in the appropriate gateway.

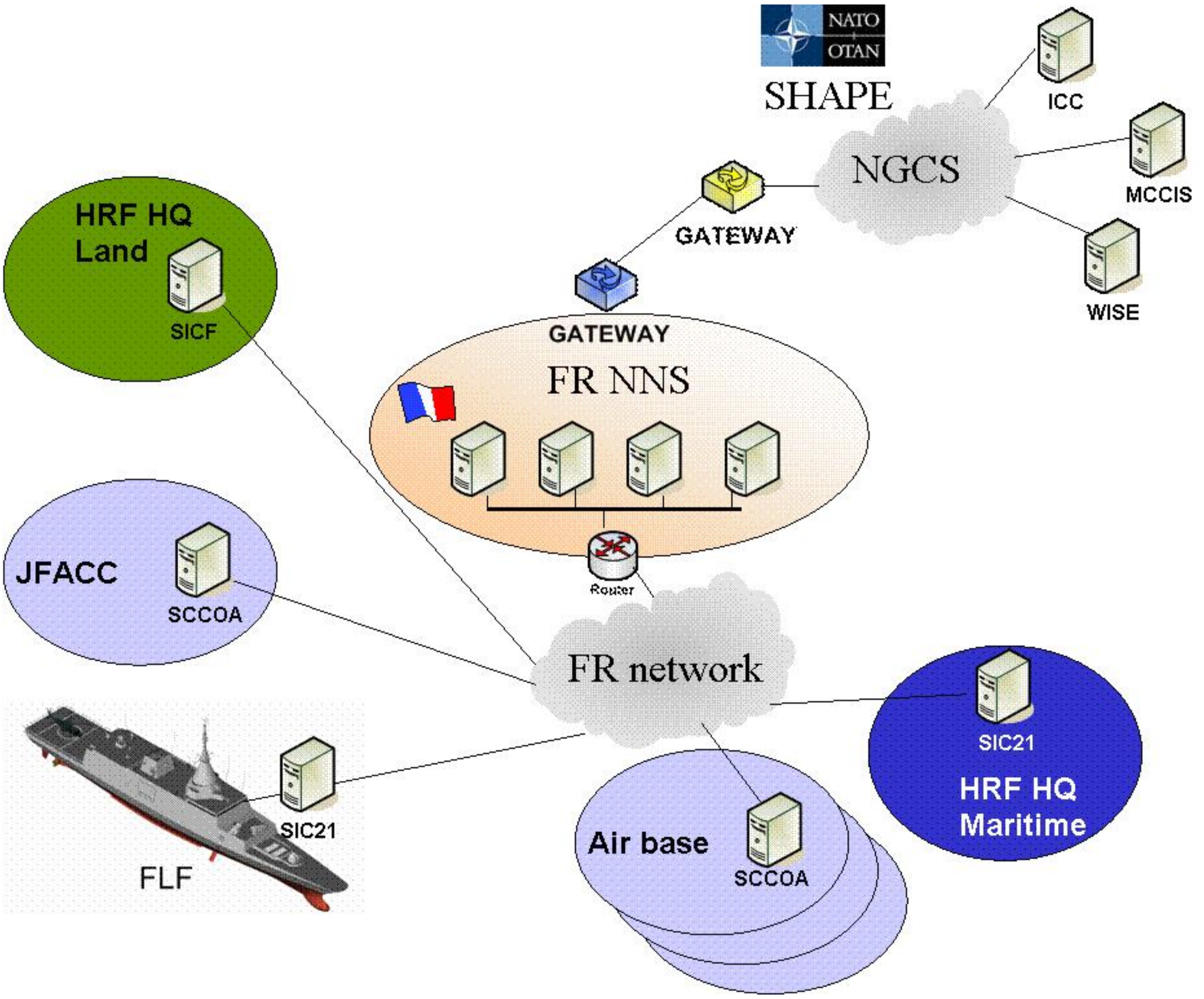


Figure B.5. IP Interconnectivity

086. The target is to share the access also for other systems like ADAMS, LOGREP, LOGDATA,..and facilitate the evolution towards NMS.

B.2.2.4.1. Short term solution

087. The availability of the FR NNS is needed for the validation phase of the new maritime system that France is developing: SIC21. This system will have to exchange information with MCCIS and several others NATO systems. As this validation phase will start during summer 2007, it is envisaged to use the IEG that was used during the NRF5 rotation with some minor changes (authorization of MCCIS flows). At the same period of time, the air community will start to move on the NNS: the CASPOA (Analysis and Simulation centre) for air operations preparation, which is a NATO Centre of Excellence, located in Taverny (close to Paris), the CCOA (Air Operation Centre) located in Lyon and the CDAOA (Air Defence and Air Operations Command) located in Paris will be connected.

088. At the end of 2008, all the ICC and MCCIS clients located in France should be able to work directly from their computers on the NNS. This means on the contrary that the people located on the theatre will still require direct links to NATO WAN.

B.2.2.4.2. Mid term objective

089. Two aspects will be considered after the starting up of this network. On one side new clients will have the opportunity to access those services and on the other side new services will be added on the network. The new clients concerned by this migration on NNS are the squadrons and air headquarters that can participate in NRF/HRF such as flying units, ARS, Ships and headquarters and the new headquarters created in France The pace for moving the units will be based on the next rotation of NRF where France is playing a component command role.

090. For NRF 11, France will be Land Component Command and still keeps two options for its interconnection. The first would be to use the FR NNS and the second to have a direct connection to NSWAN. The choice is depending on the organization of the Joint Force Command leading the rotation.

091. For the NRF 12 France, Air Component command, it is assumed that the FR NNS will be used as it was done during the rotation 5. The only difference would be the availability of more services.

B.2.2.4.3. CRONOS integration in FR NNS

092. The actual network is obsolescent. It was built in 1999 and some client workstations were deployed in various locations. France has contracted a support with a company to install and maintain the various computers. This contract is coming to its end, thus there is an opportunity to re-analyse the relevance of such a contract and the associated cost. Some clients will not be connected with the contractor because there are out of the boundary of the initial contract. France has made the decision to int grate this "service" in the FR NNS. No more upgrade of the actual network is planned; all the new users will be connected via the FR NNS.

B.2.2.4.4. ICC integration in FR NNS

093. France will use the opportunity of the movement of the different headquarters during the 2007-2008 period to setup an architecture that will support all the workstations spread out in many places in France. The upgrade of ICC from the actual version to 2.7.1, then 2.8 will give an ideal integration period to migrate towards the new approach. After that migration it is assumed that network upgrades will be necessary for the air community for the arrival of ACCS. All the network evolutions will be taken into account by the FR NNS network manager.

B.2.2.5. Ongoing work

B.2.2.5.1. Interfaces between NS, NNS and MS

094. During operations, usually a Mission Secret network is deployed. To enable interconnection between network handling NATO Secret information and network handling Mission Secret

information, it will be necessary to develop an IEG Case C. The location of this gateway is still under discussion and need to be further investigated. The main objective for this interconnection is to offer all the NATO IP services to the different communities of interest in the theatre.

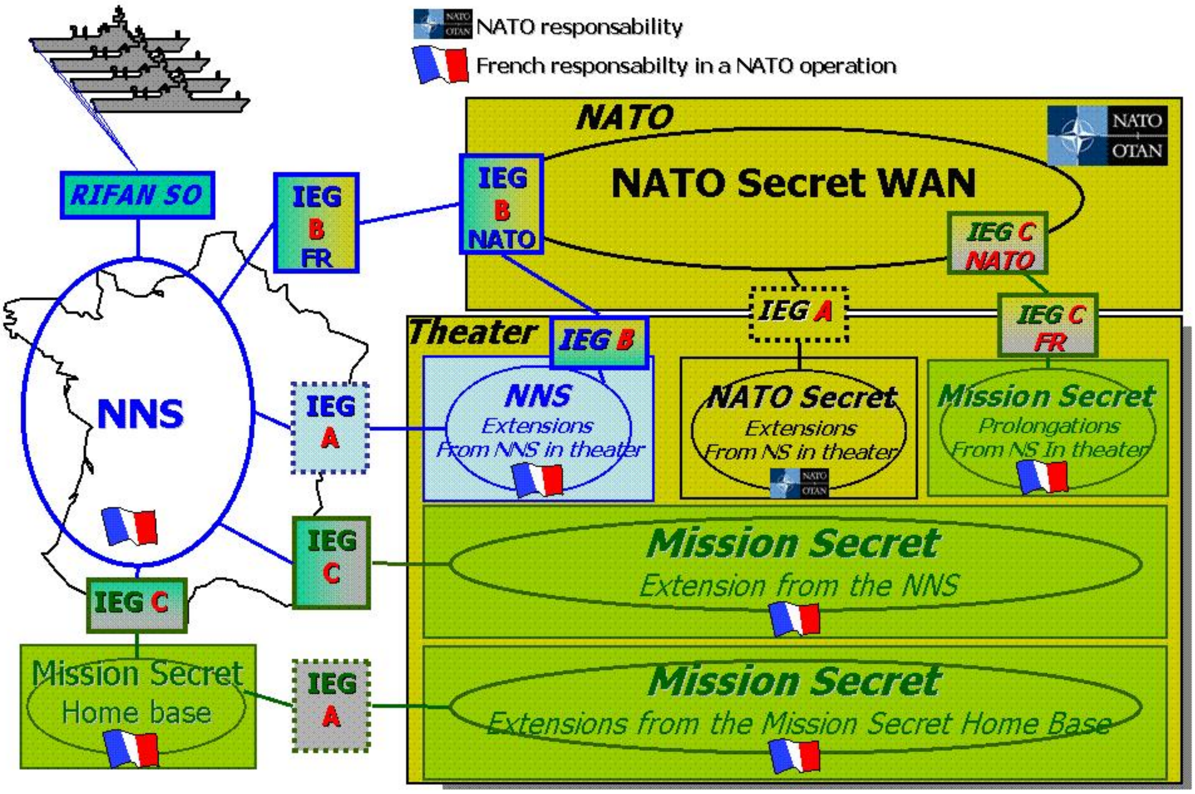

NATO responsability
French responsabilty in a NATO operation
NATO
NATO Secret WAN
NATO OTAN
RIFAN SO
IEG B FR
IEG B NATO
IEG C NATO
IEG A
IEG C FR
Theater
IEG B
NNS
NNS Extensions From NNS in theater
NATO Secret Extensions From NS in theater
Mission Secret Prolongations From NS In theater
IEG A
IEG C
IEG C
Mission Secret Extension from the NNS
Mission Secret Home base
IEG A
Mission Secret Extensions from the Mission Secret Home Base

Figure B.6. Interconnecting MS Network

095. The management of such a network composed of systems managed by NATO bodies and national teams will need to be worked out to develop Service Level Agreement between the network providers (NATO and national) so that availability seen from the users' point of view is maximal.

096. The other challenge will be to ease access to the right information. Having authorized the interconnection of the NATO and the national systems does not mean that the various people in the different Headquarters are accessing the information of interest to them.

B.2.2.5.2. IP networks not on NNS

097. Some systems are currently under development and a more accurate analysis must be undertaken to verify how and when they could go on the national network. That is the case for ACCS and Bi-SC-AIS. This analysis should be followed by a testing phase to ensure that the system continue to provide the services it is supposed to provide.

098. For other services, like videoteleconferencing, France will setup a separate network because of bandwidth constraints. When the quality of service will be spread on all the IP networks, and operational priorities agreed within the nations, this could evolve towards a unique network on which all the services would be made available.

B.2.2.5.3. Non IP networks

099. If there is a clear thread to use an IP network for exchanging most of the data needed for command and control functions, it must be recognize that non-IP networks will last for at least 5 years. Thus it is important for the services that they support to guarantee that the interoperability between the non-IP systems within NATO nations and PfP nations will continue until the transition phase to IP is accomplished. The services concerned are real-time service (e.g. Link 16, circuit between radar and CAOC,...), telegraphy, circuit switched and packet switched services.

B.2.2.5.4. NNEC perspectives

100. To foster and federate initiatives to improve military actions effectiveness within a coalition force, information is the key element. Information management and effect control to reach the desired end-st te have to be taken into account joint, inter-agency and international aspects development of interoperability, using state of the art methods will accelerate adaptation towards using enterprise services across the nations and appropriate networking. The necessity to accelerate the acquisition of the systems and to optimise spending will tailor the experimentation process. In order to take advantage of all the technology opportunities, a structured but pragmatic approach will be taken to enable the different program managers to take into account the interoperability challenges in an international context. In particular, studies need to be done to identify the criteria to determine when Service Oriented Architecture is the right approach and, when appropriate, propose a migration plan in accordance with the other nations involved.

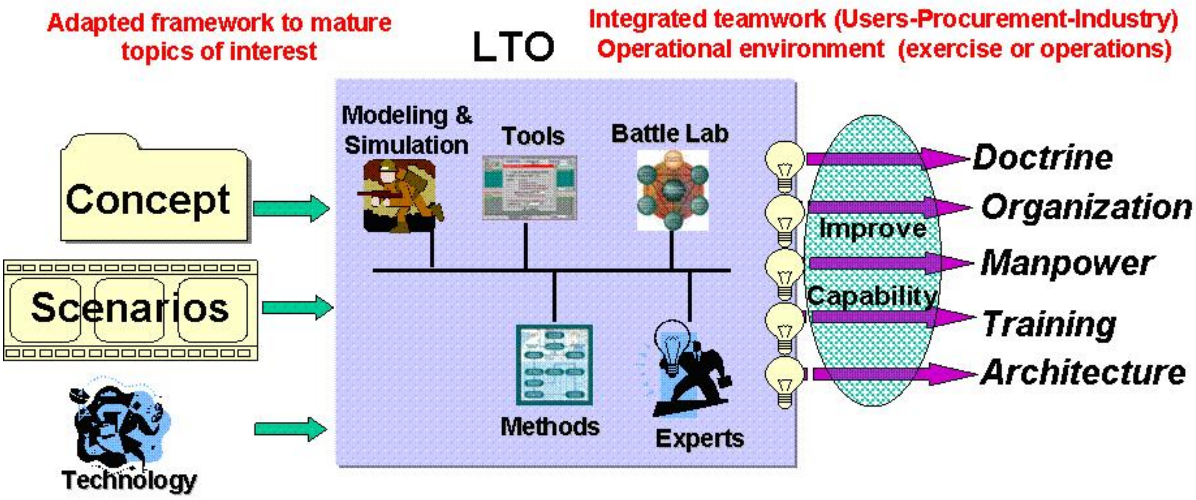


Figure B.7. French Battlelab

B.2.3. Netherlands

B.2.3.1. Policy and direction

101. NEC is essential for the innovation of military operations and recognised as a priority in Defence policy. This policy is reflected in the three courses that are followed to promote the specific implementation of NEC in the armed forces:

- Improvement of the adaptability of the armed forces for participation in multinational, joint, combined and interagency coalitions;
- Improvement of the ability of the armed forces to innovate;
- Contribution to the implementation of an integrated multinational NII.

102. Due to the priority given to NEC, the Secretary-General established an NEC steering group. This group is tasked with the promotion of innovation of military action by giving direction to the development of NEC. The Director of Operational Policy, Requirements and Plans of the Defence Staff provides central direction to the development of NEC and manages the NEC steering group. Nearly all parties involved in the transformation of the Defence organisation are represented in the steering group. The group is chaired by the Deputy Director for Requirements (of the Defence Staff). The tasks of the steering group include:

- Promoting the development of innovative military concepts;
- Promoting the adaptability and interoperability of Dutch capabilities in multinational (NATO, EU, ad hoc) coalitions;
- Monitoring the implementation;
- Informing and advising the civilian leadership about relevant subjects;
- Formulating, on an annual basis, an NEC plan as an input to the Defence plan, to be able to monitor the progress in the development of NEC and to lay the foundation for future requirements; and
- Contributing to the preparation and harmonisation of Dutch positions in meetings of relevant international consultation forums.

B.2.3.2. Implementation of national NII

103. The Netherlands armed forces will be tasked with implementing the national segment of NII, the Dutch NII, which can be integrated into larger coalitions. In this process, the main areas of attention are:

- **Design**

  The NII must be designed in such a way that it can support networked operations. It requires agreements on common architecture, the use of technologies and standards, but also on basic data and the management of these data.

- **Realisation**

  Through investment programmes and concrete projects, the NII hardware, software and services components, as well as the interfaces between those components, will be implemented.

- **Implementation and maintenance**

  This mainly concerns agreements and measures to ensure that the new technical possibilities are in fact being used. Everything depends on the willingness to share information. This area of attention also includes information management and control.

- **Protection**

  Arrangements must be agreed, measures must be taken and investments must be made to ensure the security of NII as well as the reliability of the information supply.

B.2.3.3. The development of NEC

104. NEC cannot be bought. Furthermore, the realisation of NEC is an evolutionary process without a strictly defined beginning or end. It is a process of growth, comparable to the process of change that many organisations go through in order to stay up-to-date in the field of information and communication technology. Doctrines, processes, command and control, organisations, personnel and materiel must develop in a coherent and evolutionary way. That, therefore, is the approach that the Defence organisation will use towards the development of NEC. As a result, NEC will be introduced into the Netherlands armed forces in a gradual and manageable way. This approach is characterised by the following principles:

- combination of bottom-up activities aimed at the development of the necessary information infrastructure and top-down activities aimed at concept development with regard to the future capabilities;

- a step-by-step approach, i.e. a learning path where experiments are used, in close cooperation with developers and users, to reduce the risks as much as possible and make users aware of the new possibilities of NEC as soon as possible;

- co-evolutionary approach: the balance among the various aspects of development is constantly monitored.

105. For several NEC systems, the Defence organisation takes itself the lead in the development. This approach is different to the approach used by many other nations, who develop specifica-

tions and delegate the actual development to industry based upon these specifications. TITAAN and ISIS, for example, are being developed by the Dutch Army itself, with support of industry.

106. The Defence organisation distinguishes three types of innovation on the path towards NEC:

- **Technological innovation**

  The development, application and implementation of new technology bring about changes in the physical domain.

- **Process-driven innovation**

  The adaptation of organisational processes and the implementation of new operational concepts also causes changes in the information domain.

- **Organisational innovation**

  The structure, standards and culture of the organisation and the individual change, causing the cognitive social domain to change too.

107. Taking into consideration the three types of innovation, there are five NEC levels, if we also include the zero situation - the platform-centric level. Using these levels can help the Defence organisation define its ambitions and requirements clearly and can also provide insight into the adaptability and interoperability of a military capability. These levels are respectively Isolated (1), Deconfliction (2), Coordination (3), Collaboration (4) and Coherent effects (5).

B.2.3.4. Critical success factors for the implementation of NEC

108. Critical factors on which a successful implementation of NEC in the Netherlands armed forces will depend, include:

- **Requirements evaluation and procurement**

  In the processes of requirements evaluation and procurement, NEC-related requirements must be taken into account structurally. This means that attention must be paid not only to the capability that is acquired, but also to the future role of that capability and its integration in the network. Flexible and incremental procurement strategies, in which the insight into the functional demands and the best way to meet those demands gradually increase, tie in with this. This also requires a more intensive cooperation with the industry. In the development of new capabilities, NEC-requirements will be taken aboard in the design, just as is the case now with the Joint Strike Fighter.

- **Information Management development under architecture**

  In order to establish the Dutch NII and to steer the development of the required Information Management infrastructure in the right direction, there must be an architecture available which is used effectively. The Defence Information Management Architecture (DIVA) will be developed further for this purpose.

- **Scientific research and operational experiments**

  This is necessary to be able to evaluate the usability of new concepts and technologies, thus reducing the risk of making high costs that turn out to be unnecessary and irreversible choices that turn out to be wrong.

- **The development of NEC competencies and the design of C2 processes**

  It is clear that NEC will have a considerable impact on the human element in the Netherlands armed forces. The success of the implementation of new technology, systems and concepts and the improvements in performance that these should bring about is wholly dependent on the presence of the willingness and the competencies to actually use them.

- **Central coordination and international cooperation**

  The ideal situation would be that national and international agreements are made so that the various NEC processes will be able to reinforce one another. In order to achieve this, there must be central coordination and international cooperation.

B.2.3.5. Examples research and experiments

B.2.3.5.1. Experimenting - JPOW

109. CHANIA (CRETE) - In Joint Project Optic Windmill (JPOW), the computers of all participating countries are linked together, for interoperability is an integral concept throughout JPOW. "Three, two, one, fire", a Greek officer, headphones over his coalblack hair, counts down. A trembling explosion; a grey white cloud of smoke like a small cauliflower. An orange-coloured rocket races up from the launch site and hisses its way into a cloudless sky. Shortly before the projectile intercepts a small target aircraft, it is destroyed from the ground, and its shattered remains fall into the ice-blue Mediterranean waters off the coast near the NATO shooting range on the rocky island of Crete. This pandemonium is repeated three times. It is D-Day in the international air defence exercise Joint Project Optic Windmill. As a highlight, the Guided Missile Group of the De Peel airforce base shoots four ageing rockets into the firmament. JPOW is somewhere in between Star Wars, Star Trek and the US Strategic Defence Initiative. The foundation of the project was laid by the Netherlands, motivated by the first Gulf war (Source: Defensiekrant (MOD periodical), 20 April 2006).

B.2.3.5.2. Experimenting - SENECA

110. In 2005, the Defence organisation, along with TNO and THALES, carried out the first Dutch Joint NEC experiment. This experiment involved setting up a distributed experimenting environment which linked several locations together. Simulated operational sea, land and

air systems successfully networked their actions in a single scenario. The experiment demonstrated the added value of the development of a Simulation Environment for NEC Assessment (SENECA) that provides opportunities for frequent experimenting.

B.2.3.6. Examples NEC in practice

B.2.3.6.1. Optimal use of information

111. The maritime dimension traditionally has a strong multinational orientation. Individual units must be interoperable to be able to function in a multinational flotilla. A recent example (late 2005 - early 2006) was the Dutch contribution to Task Force 150, which was responsible for so-called maritime security operations in an area stretching from the coast of Somalia to the Arabian Gulf. TF 150 consisted of a great variety of navy vessels (including an air defence and command frigate), maritime patrol vessels, unmanned aircraft and shipborne helicopters. All these means contributed to the joint mission: identifying and detecting suspect vessels, boarding and searching vessels, collecting information and halting suspect vessels and arresting their crew. Through use of the American CENTRIX network and the Coalition Forces Intelligence Cell, the vessels could exchange information (such as photographs) with each other and with shore authorities directly by e-mail and chat, thus ensuring that the waiting time for merchant ships could be reduced to a minimum.

B.2.3.6.2. ISIS

112. Task Force Fox was a NATO force that was intended to protect the international observers who were tasked with monitoring the implementation of the peace plan for Macedonia in 2001-2002. In this multinational coalition with Germans, Frenchmen and Italians under Dutch leadership, the language barrier was, considering the operational circumstances, a real problem. Because information was shared with the coalition partners using ISIS, and this information was also represented graphically, there would soon be a shared understanding of the operational situation and emerging crises could be overcome swiftly and effectively. It is also thanks to ISIS that Task Force Fox has gone down in history as a successful NATO mission.

B.2.4. Spain

B.2.4.1. Policy and Direction

113. National policies and directives support the bringing in of the NEC model nationally as they do its contribution to the multinational efforts. Both, information superiority and interoperability are considered a force multiplier/booster at the Spanish Military Strategy. The National Defence Act speaks of a call to "urge a transformation of our Armed Forces in line with a new model which would give them advanced technological capabilities". The Organic Law of National Defence and the Military Planning Directive requires that "Armed Forces are to be organized and trained in such an advanced technological manner that they will thus enjoy combat advantage, fewer losses and less damage and be networking operative"

114. Aware of the very high number of actors and lines of work involved on Spanish NEC, in order to assure the proper co-ordination, CHOD set up a NEC Study and Developing Board (CEDENEC). This commission comes under the Chief of the Joint Staff responsibilities and is presided over by the Head of the CIS Division drawing its members from Defence staff HQ Spain and the HQs of each of the Services along with the Materiel and Armaments Board (DGAM), the CIS Inspectorate (IGECIS) and the Policy Board (DIGENPOL).

115. This organization gives the specific structure set up for the improving of the initiatives coherence that bring in other pertinent bodies, and for the generating proposals including those approaches required to make of Spanish NEC a reality and its contribution to NATO NEC a fact. This working whole structure, that includes a working groups subestructure and is supported by the Spanish NEC Office, is charged with giving leadership to NEC during its initial phases by answering questions as to its definition, objectives, requirements, criteria as to its functional structure, information infrastructure, interoperability and any other matter that might have an effect on the development of NEC. It is also to promote the NEC knowledge within the Spanish Armed Forces, specially to all the involved bodies in its definition and development and to manage NEC related information.

B.2.4.2. Shaping up Spanish NEC

B.2.4.2.1. Pre-emption: Strategic Framework and Complexity Management

116. Spain is aware that there is no single sequence of long-term planned baselines, but an approximate direction of progress, frequently reviewed and adapted to changing technologies, requirements and constraints. Therefore, CEDENEC believes that the best planning strategies under a high level of uncertainty and risk are no longer those that deliver optimum results in the very specific circumstances anticipated, but those that are robust enough to perform acceptably in a variety of potential scenarios. As such a strategic framework to answer the what-for, the what, the how and the when of the undertaking should be and should allow a balance among stability and flexibility.

117. The Spanish NEC Strategic Framework is under CEDENEC's development and covers what Spain must achieve in order to fulfil the operational requirements; Lays down the strategic lines of work including those related with conceptual development, methodology and best practices, design and implementation and shared awareness; Establishes the global plans which include the strategic lines of work and; Goes into details as to short term plans. As it is to be the case for any rolling strategy, operational and technical analysis and studies (including those lessons learned) are to be continuously improving upon this strategic framework. Last but not least in importance is the human dimension including morale, leadership, training, education and, doctrine. The strategic framework is also aimed to this aspects.

118. This strategic framework is to be considered within the Spanish Military Planning Process as well as within those efforts made of Defence bodies having technical responsibilities, research and development competences along with acquisitions responsibilities. In this situation

the efforts towards the rolling orientation of running capabilities, programs and initiatives are going to be more important than ever.

B.2.4.2.2. Operational and Technical Interoperability Awareness

119. It is more important than ever to keep alive the operational and technical interoperability awareness that Spain has held essential for so many years and backed by her participating in different demonstrations, exercises and experiments.

120. It is to note Spanish technical experimentation at CWID initiatives. Spanish industry, universities and military bodies with technical competences play a part in the CWID demonstration. A Pilot model for SOA interoperability services, the Spanish Military Messaging System, Spanish web map services and web feature services or COP services have been put to work within latest NATO CWID. As it happened to be within other nations, Spain has its own national CWID. As such, middleware for sensors at C2 networks, or broad band mobile communications for tactical deployments, or tactical mini terminals for satellite band X communications have also been given demonstration airings.

121. Spain is also working on supporting infrastructures for experimentation, training and exercises. Noteworthy here is that Spain is working towards the setting up of a Laboratory for NEC experiments (CENEC) where any practical solutions put forward are to be measured against those operative demands. Advanced joint simulation facilities are to be keep financed next year. NEC related prototypes have been delivered to our Operational Post Command.

122. Following the NATO approach the development of our systems is in keeping with NC3TAv7 and NAFv2. Spain is analyzing its architectures views evolution in keeping with NAFv3 and NISP.

B.2.4.2.3. Follow-on

123. The main effort within the short to medium term must be towards achieving the Spanish NEC strategy for the managing of this complex endeavour and its concomitant experimental support demands.

124. In the medium to long term Spain expects to be able to have the run of a fully operative working whole structure, this taking into its scope civilian organizations and other ministries.

125. If there were something that could be called an international net-ready certification Spain could be deeply involved in it. Systems processes, capabilities and even acquisitions programmes could be verified on the basis of net-ready criteria or, to put it in another way: systems, processes, capabilities and acquisition programmes associated with network enabled capabilities could have what it might be call a "net-ready stamping". An interoperability authenticity stamp which would be, but of course, anchored upon the international determination it is concerned with. Which all means that an evaluation of existing systems will be called for to check out whether the net-ready criteria have been satisfied be it nationally or within any given coalition.

126. The Spanish acquisition process is aligned with NATO Phased Armaments Programming System (PAPS). This process came up for review under a new proposal. Among other aspects, it's under consideration a set of criteria that allows PAPS documents go further in the process, in order to enable rationalization as well as the necessary evolution in aspects such as technology.

127. Spain's ultimate vision is of a real federation of military and civilian services and its cross utilization for crisis management and current operations. Civilian and military bodies would natuarally provide any that were to be needed, services that would thus endow us with powerful dual capabilities for defence and security.

B.2.4.3. Implementation of the Spanish Core NII: The Spanish Military Command and Control System (SMCM)

128. In order to solve the interoperability and information flow problems caused by a fragmentary approach, the solution adopted is to define and implement a new C2 system, the SMCM (Joint C2 system), which will become the basic CIS infrastructure on which the specific services will be built. Of course this new system integrates the military telecommunications that have been integrated in a single joint system for many years (the Military Telecomunication System - STM) and is designed from the beginning to solve the shortfalls and problems of the existing information capabilities.

129. Its Military Information System (SIM) incremental process has been divided into 3 increments and a previous definition phase. In this definition phase, it has been developed and approved the Concept of Operation of the System, the System Planning and Funding. The first increment objective is to implement and provide a set of common services, like both PKI and Directory Services and the priority services like Military Messaging System. The system is to be deployed in the principal nodes (Joint, Land, Air and Maritime Headquarters, etc.) and is to be accredited to manage information upon National SECRET, NATO CONFIDENTIAL and CONFIDENTIAL UE level. The second increment includes the implementation of the specific functional services, the accreditation to NATO SECRET and SECRET UE, and it is planned to deploy the system to 400 nodes approximately. The last increment it is planned to provide all the functional area services and evolve the system to comply with the Network Enabled Capability. Of course, our will to set up this increments as schedule is tied to the available budgeting and funding.

B.2.4.4. POC

130. ESPNEC@oc.mde.es

B.2.5. Sweden

131. Authors:

- Peder Blomqvist, Swedish Defence Material Administration

B.2.5.1. The Swedish Network Based Defence (NBD) concept

132. The Swedish Armed Forces has adopted the concept of Network Based Defence (NBD) as its main strategy to achieve Network Ready Capabilities. NBD is also the method for Swedish force transformation to Joint Operations and Capabilities and thus a fast, flexible and co-ordinated deployment of Reaction Force Units within national boundaries as well as a partner in an international mission. The NBD concept is the Swedish approach to retain the benefit of the Revolution in Military Affairs (RMA) and is similar to the Network Enabled Capabilities (NEC) and the Net-Centric Operations Warfare (NCOW) concepts.

B.2.5.2. Development of a Command and Control System within the NBD concept

133. The Swedish Armed Forces has within the concept of NBD since 2001 been developing frameworks, concepts, principles, rules and recommendations, as enablers and guidance for design, construction and operation of the methods, services and technology solutions that should form the foundations and infrastructure of a future Command and Control System. The goal is to deploy the target architecture and infrastructure of the C2-system within the year of 2010. The C2-system is being developed with interoperability and the usage of open standards and commercial products (COTS) in focus. Gradually the C2-system infrastructure will develop into a full-featured Net-centric C4ISTAR-system with incorporated functionality for other legacy systems as well as new systems.

134. The Swedish Armed Forces has during 2006 established a NBD Development Centre under the command of Chief of Development.

135. Sweden has adopted a Concept Development and Experimentation (CD&E) process based on international examples. One of the purposes of this process is to guarantee or secure interoperability.

136. Sweden has conducted experiments in three different military command levels and also in Multi-National Experimentation (MNE) events.

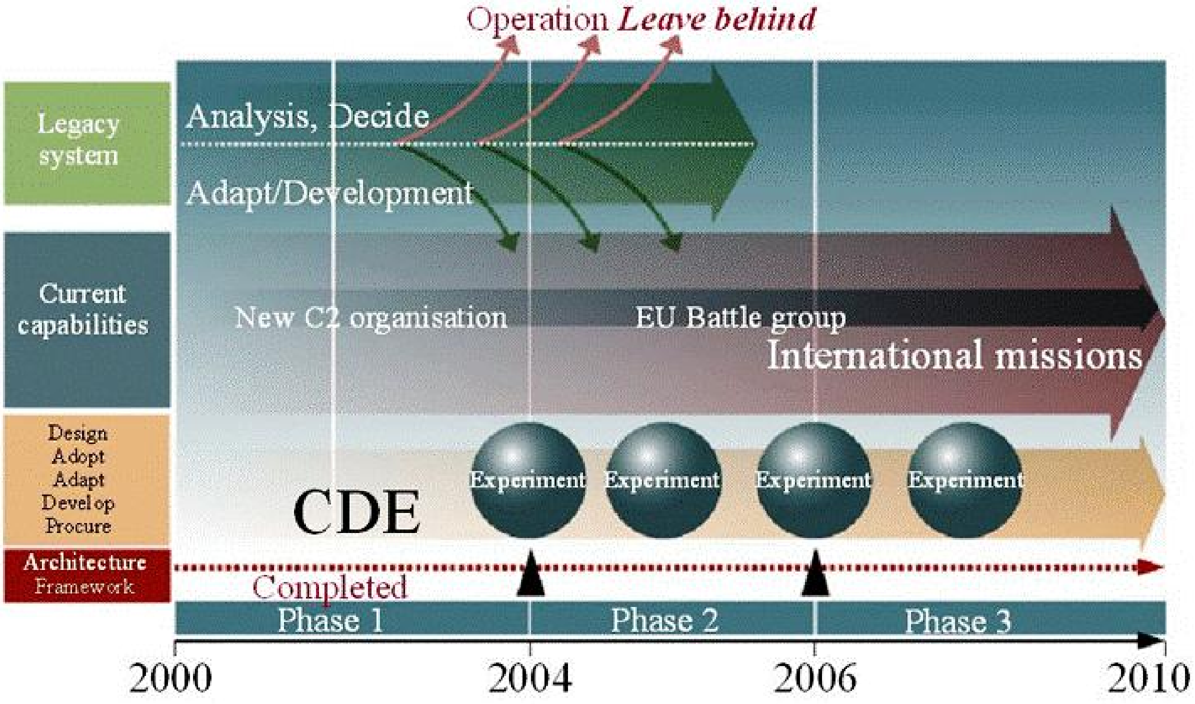


Figure B.8. Transformation Roadmap

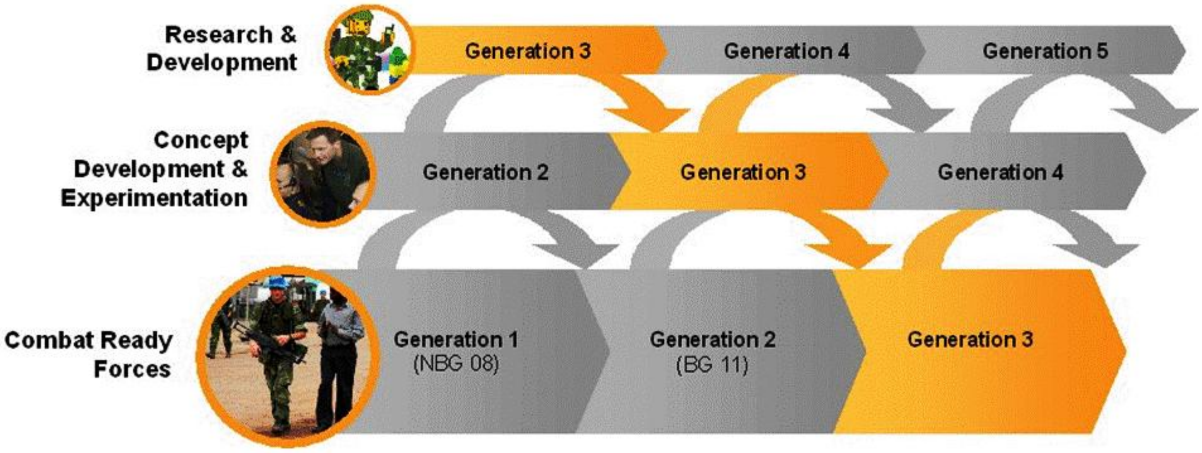


Figure B.9. CD&E Model

B.2.5.3. Development Strategy and Methodology

137. The development is a joint effort between the Swedish Armed Forces, the Swedish Defence Materiel Administration (FMV), the Swedish Defence Research Agency (FOI), the National Defence College (FHS), the industry and other partners. The development is based on an overarching architecture approach with concepts and principles for System, Services, Informa-

tion and Life-Cycle management. The development strategy is to use an evolutionary development methodology with continuous and parallel research and development of methods, knowledge, organisation and technology. The results are tested, verified and validated in experiments and demonstrators, and then carefully packaged in form of generic principles, design rules and methods. These will gradually build up the target architecture with re-usable foundations and reference architectures for further development of service-oriented systems and components for net-centric environments. The Swedish Armed Forces development of the NBD joint Command and Control Information System (SWECCIS) results in the knowledge and erudition of building NBD operational systems.

B.2.5.4. Development Milestones for the Command and Control System 2004-2010

138. Autumn 2004: New development opportunities

139. Autumn 2005: Common situational information

140. Spring 2006: Common situational awareness

141. Autumn 2006: New opportunities for co-ordination

142. 2007-2010: Final design and procurement of the Command and Control System Information Infrastructure[2]

B.2.5.5. Interoperability

143. In autumn 2004 Sweden made a strategic decision to adopt current NATO standards and frameworks for C3-systems interoperability and to participate in and contribute to NATO's development of NNEC-based standards and frameworks, where possible as a partner nation. Lessons learned from the adoption of NATO standards and open standards and solutions create an environment that increases interoperability among crisis management agencies.

B.2.5.6. Main objectives of the Network Based Defence (NBD)

144. *Interoperability* - Ability to act in Joint Operations together with national and international partners, both military and civil. This can be achieved through standardisation and harmonisation of concepts, procedures and architecture.

145. *Situation Adaptiveness* - Flexible configuration and management of units, for given tasks and new opportunities. Dynamical configuration of Situation Adopted Systems, from a Baseline Architecture of resources and services. Well defined suites of Capability Packages for different scenarios.

[2] If decided by the Swedish Government

146. *Common Situational Awareness* - Common Situational Information adopted for each user needs. Role-based situation picture and information access. Location and terminal independent distribution of data. Automated as well as on-demand collection, aggregation, fusion, processing, analysis, predictions and presentation of data and information from multiple sources.

147. *Command Superiority* - Swedish NBD experiment has showed that information that are fast distributed through out the Network and not compiled in centralised nodes empowers the lower echelon's to seize opportunities according to commander's intent as they come along.

148. *Cost effectiveness* - Flexible configuration will give us the ability to reuse and reconfigure systems and services in minutes instead of months/years. This together with the ability to bring along legacy systems should mean that the risk of building systems that are obsolete when operational also reduces the risk of wrongly spent money.

149. *Precision Engagement* - Co-ordinated and well balanced engagement, precise in time and location. This can be achieved by obtaining decision superiority.

B.2.5.7. The Service Demonstrator

150. The Swedish Defence Architecture project together with FM *Ledsyst* developed a set of concepts that supports a Service Based Systems approach where services can be assembled on demand into SitSyst, Situation adapted systems. A SitSyst is dedicated to solve a particular problem area or support a specific mission. These ideas and concepts were implemented during 2003 in a platform called *The Service Demonstrator*. *The Service Demonstrator* is a test system developed within Ledsyst in order to support the development towards a network centric defence. The purpose of the demonstrator is primarily to support the methodology development for network centric warfare.

151. *The Service Demonstrator* have continuously been updated and improved up until now. More system elements have been added and improvements have been made concerning stability, user functionality, application integration etc. Central concepts for *The Service Demonstrator* are SitSyst, dynamic addition and removal of services, creation of roles and authorization for solving specific tasks.

152. Built on experiences from the Service Demonstrator and lessons learned the development of a more extensive service based system, *BasePlatform*, began in 2005. The current version of *BasePlatform* specifies a platform design that addresses core functionality such as service infrastructure, security, scalability, flexibility etc.

B.2.5.8. FMA- An Enterprise Architecture for Future Systems in the Swedish Armed Forces

153. SAF Enterprise Architecture (FMA) for the 21st century is under development. Swedish Armed Forces needs to co-ordinate and synchronise different resources (personnel, technology

and information) enabling them to perform right activity, in right time frame and in right area. These resources are co-ordinated and synchronised within an organisation. To synchronise all these mutual dependent systems and their development over time is the complex challenge that Swedish Armed Forces Enterprise Architecture shall support Swedish Armed Forces to master. The purpose of FMA is to obtain tools for enabling interoperability in co-operation with society in general and international co-operation partners as well. Another important aspect is to give conditions for re-use of resources within the frame of NBD. SAF engagement systems must therefore be developed against a harmonisation and towards one common structure of present and new structures. This development shall be supported through supplying an Architecture Framework (FM AR) containing overarching rules for:

- How to describe a system
- Methods for build and use systems
- Competencies/roles in these systems

B.2.5.9. National NNEC Approaches

154. Some Swedish examples of key strategic programmes project experiences, products and best practice, lessons learnt, that can perhaps be of interest for the NATO and PfP nations.

- FM LedsystT FMLS2010 Network Based Defence experiences
  - Documentation experiences (SwAF Architecture Framework, NAF v2)
  - SOA implementation experiences and best practice
  - Service Demonstrator experiences and best practice
  - NBD concepts, definitions and best practice
    - System concept
    - Service concept
    - Situation Adopted System concept (SitSyst)
    - Information concept
    - Life-cycle management concept
  - NBD Systems Development methodology work and
- SwAF Enterprise Architecture development and NATO NAF 3 adjustment effort.
  - FM AR Concepts and models

  - NAF v2 and NAF v3 (pragmatic approach) documentation experiences and best practice

- SWECCIS Reference Architecture experiences (Swedish Command and Control Information System), target architecture for SWECCIS 2008.

  - NAF v3 (pragmatic approach), documentation experiences and best practice

- The SwAF ERP project (PRIO) and the service oriented requirement specification work.

B.2.6. The UK Network Enabled Capability

155. The UK has adopted the term Network Enabled Capability (NEC) as it captures it's intent to invest in networks as a means of maximizing the capability inherent in current and future platforms. The UK is using the term "Network Enabled Capability" since it gives due weight to both the platform capability and the network, and is consonant with their work on Effects Based Operations, Knowledge superiority and Decision Superiority.

156. The NEC goal is to radically enhance our operational capability by improving the way we share and use information , with the key objective of providing a coherent conceptual and technical framework to link sensors, decision-makers and effectors.

157. The US concept of Network Centric Warfare, put into practice in Afghanistan, is a key driver. In the same way that precision guided munitions demonstrated in the Gulf war acted as a driver for change in UK and other forces, the success of the US' strategy, implement through a tight linkage between sensors, communications, information, and weapons systems is causing the UK to look at how it would fight the next conflict.

158. Lessons learned have emphasized the importance of interoperability in the joint and coalition environment.

159. It is important to understand that the path to NEC is an evolutionary one that builds on existing equipment programmes, concepts and structures but delivers increased joint capability. The parallel evolution of the other UK Lines development (Concept & Doctrine, Structure, Equipment & Technology, People, Training, Sustainment) is an essential part of the progress towards a full Defence capability Existing processes will provide the links.

160. The diagram below shows the underlying principles in the UK in building its NEC.

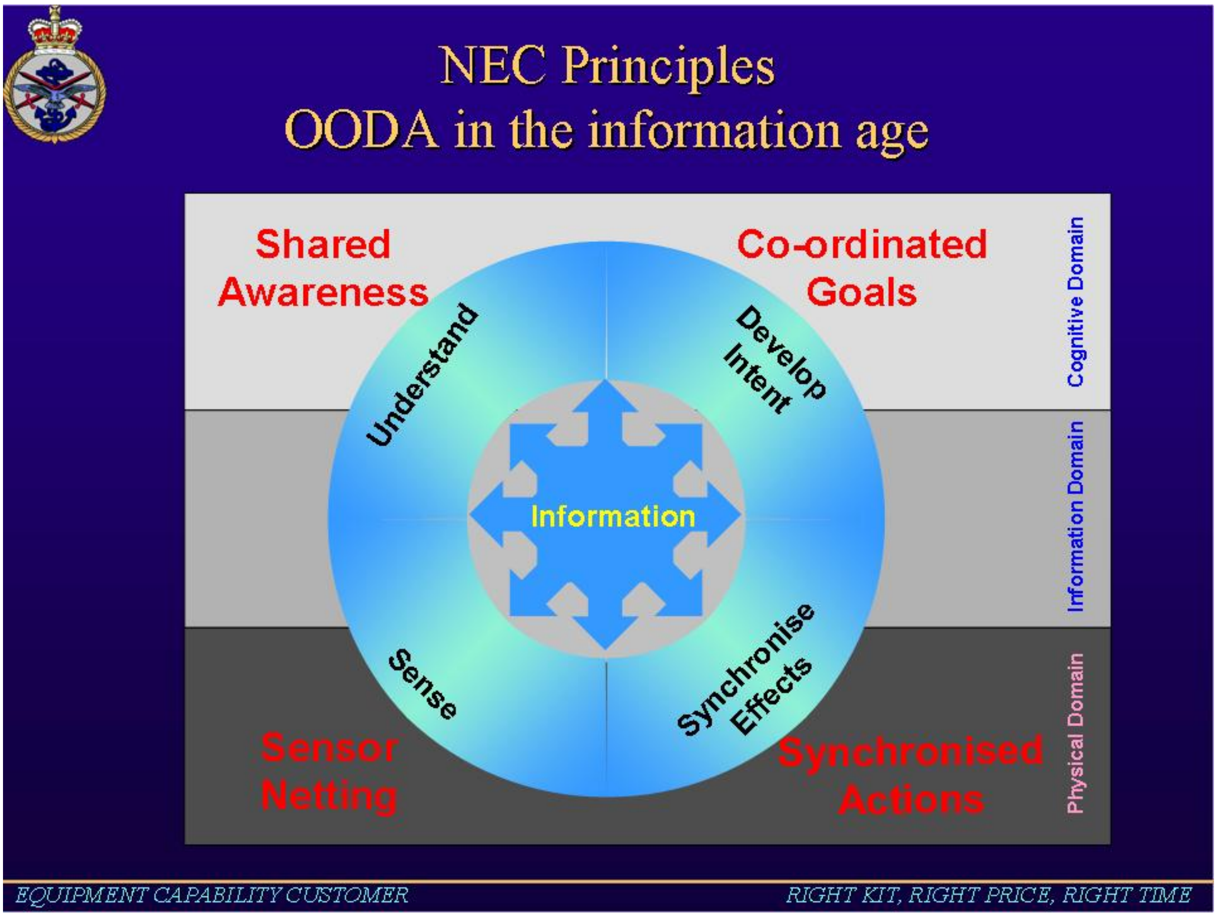


Figure B.10. NEC Themes

161. The table below summaries the key themes of the UK's NEC

Effects Synchronization	Achieving the desired effects through the synchronisation of activities within and between mission groups.
Agile Mission Groups	Enabling the dynamic creation and configuration of task oriented mission groups that share understanding and that employ and co-ordinate available assets to deliver the desired effect.
Dynamic Collaborative Interworking	Enabling agile command and control within and between mission groups through the ability to concurrently plan and execute operations in a way that is dynamic, continuous and synchronised. Thus, it allows all entities (including non-frontline MoD bodies, Other Government Departments, industry, academia and public services as well as the military) to work

	together dynamically to meet changing mission needs.
Shared Understanding	Enabling each user to generate an understanding of the battlespace that is appropriate and adequate to their task and consistent with the understanding of other users. This understanding covers the interpretation of the situation (current situation, its history, and potential developments of all battlespace participants) and of Command Intent (the effects and outcomes higher command wants to achieve.
Full Information Accessibility	Enabling users to search, manipulate and exchange relevant information of different classifications (respecting security constraints) captured by, or available in, sources internal and external to the battlespace.
Resilient Information Infrastructure	Enabling information is managed coherently across the battlespace and that the potential for secure and assured connectivity is provided to all battlespace users.
Inclusive Flexible Acquisition	Co-ordinating processes across MOD, OGDs and industry that promotes the rapid insertion of new technologies, facilitates coherence between acquisition programmes and provides an incremental approach to delivering and sustaining 'net-ready platforms'

Table B.1. UK NEC Themes

162. Further information can be found at: http://www.mod.uk/DefenceInternet/AboutDefence/CorporatePublications/ScienceandTechnologyPublications/NEC/

B.2.7. United States

B.2.7.1. Overview

163. The United States' Department of Defense (DOD) pioneering theory of Net-Centric Operations and Warfare (NCOW) has fundamentally changed how the United States plans for and engages in military operations. NCOW seeks to translate an information advantage into a competitive war-fighting advantage through the robust networking of well informed geographically dispersed forces allowing new forms of organizational behavior. This "networking" utilizes information technology via a robust network to allow increased information sharing, collaboration, and shared situational awareness, which theoretically allows greater self-synchronization, speed of command, and mission effectiveness.

B.2.7.2. Net Centric Efforts

B.2.7.2.1. The Global Information Grid (GIG) Bandwidth Expansion Program

164. The Global Information Grid Bandwidth Expansion (GIG-BE) Program was a major Department of Defense (DOD) net-centric transformational initiative executed by DISA. GIG-BE created a ubiquitous "bandwidth-available" environment to improve national security intelligence, surveillance and reconnaissance, information assurance, as well as command and control. Through GIG-BE, DISA leveraged DOD's existing end-to-end information transport capabilities, significantly expanding capacity and reliability to locations worldwide.

165. This program provided increased bandwidth and diverse physical access to approximately 87 critical sites in the continental United States (CONUS), Pacific Theater, and European Theater. These locations are interconnected via an expanded GIG core.

166. GIG-BE provides a secure, robust, optical terrestrial network that delivers very high-speed classified and unclassified Internet Protocol (IP) services to key operating locations worldwide.

167. After extensive component integration and operational testing, implementation began in the middle of the 2004 fiscal year and extended through calendar year 2005. The initial implementation concentrated on six sites used during the proof of initial operational capability, achieved on Sept. 30, 2004. On Dec. 20, 2005, the GIG-BE program achieved the milestone of Full Operational Capability.

B.2.7.2.2. Transformational Communications Satellite System (TSAT)

168. The TSAT Program is actually just one node in a broad spectrum of programs known as the Transformational Communications Architecture (TCA). In 2001, the United States Department of Defense (DOD) initiated a Transformational Communications Study to accelerate the delivery of advanced capabilities with state-of-the art technology to the field.

169. The study concluded that the United States Military's existing program plan would not meet forecast communications requirements. It also suggested that there was a window of opportunity to provide an architectural framework for a compatible communications system across the Department of Defense and the intelligence community - one that could increase U.S. capabilities by a factor of ten.

170. Those conclusions, plus ongoing experience in the Global War on Terror and new technology developments like UAVs, helped shape the Transformational Communications Architecture (TCA).

171. TSAT is intended to provide internet-like capability that extends high-bandwidth satellite capabilities to deployed troops worldwide, and delivers an order of magnitude increase in available military bandwidth. Using laser communications inter-satellite links to create a high datarate backbone in space, TSAT will be one of the key enablers for the American vision of Network Centric Warfare.

172. A visual image from a UAV that would take 2 minutes to process with the Milstar II satellite system would take less than a second with TSAT. A radar image from a Global Hawk UAV (12 minutes), or a multi-gigabyte radar image from space-based radar (88 minutes), would also take less than a second with the TSAT network. Best of all, the recipient can be on the move with a relatively small receiver, anywhere in the world.

173. The TSAT system is currently scheduled to launch in 2013-2016.

B.2.7.2.3. Wide-Band Satellite Communications

174. Wide-Band Satellite communications provides ubiquitous communications with optical quality bandwidth to mobile and tactical users.

175. The Wideband Gapfiller Satellite program and the Advanced Wideband System will augment and eventually replace the Defense Satellite Communications System (DSCS) in 2009 or 2010. These satellites will transmit several gigabits of data per second-up to ten times the data flow of the satellites being replaced.

176. Protected communications will be addressed by a global extremely high frequency (EHF) system, composed of the Advanced Extremely High Frequency System and Advanced Polar System. These systems are expected to provide about ten times the capacity of current protected satellites (the Milstar satellites). Narrowband needs are supported by the UFO (Ultrahigh-frequency Follow-On) constellation, which will be replaced by a component of the Advanced Narrowband System.

177. Capacity gains in these systems will also be matched by improved features, such as multiple high-gain spot beams that are particularly important for small terminal and mobile users. Satellite, terminal, control, and planning segments will utilize emerging technology to ensure the best capability for the cost. Coordination among ground, air, and space segments and between government and commercial assets will help ensure deployment of the most efficient, effective, and affordable communications systems.

B.2.7.2.4. Net-Centric Enterprise Services

178. Net-Centric Enterprise Services (NCES) is a Department of Defense program, managed by the Defense Information Systems Agency (DISA), to develop information technology infrastructure services for future systems used by the United States military to support the broad range of applications and data used in a net-centric enterprise. There are nine core enterprise services defined in the Network Centric Operations and Warfare - Reference Model (NCOW-RM):

1. Storage
2. Mediation
3. User Assist
4. IA (Information Assurance)

5. ESM (Enterprise Service Management)
6. Messaging
7. Discovery
8. Application
9. Collaboration

179. NCES maps these nine services to four product areas:

1. Enterprise Service Oriented Architecture (SOA) Foundation
2. Content Discovery & Delivery
3. Enterprise Collaboration
4. Defense On-Line Portal

B.2.7.2.5. Horizontal Fusion

180. Horizontal Fusion (HF) program refers to the net-centric applications and content needed to provide analysts and war fighters with the ability to make sense of complex and ambiguous situations. Within HF, the first implementation of a Service Oriented Architecture (SOA) in the Department of Defense was achieved. The demonstration proved that the realization of a Net-Centric environment is technically feasible using legacy investments. Horizontal fusion is not just a single program, but a portfolio of net-centric initiatives using a common architecture and integration process. Recent initiatives include the following:

- **Department of State: Net-Centric Diplomacy**. Net-Centric Diplomacy is an initiative aimed at enhancing war fighters' ability to gain situational understanding about adversaries and their operating environment by providing a full range of diplomatic reporting from world-wide posts to the collateral space, provided upon demand via net-centric DOD information services accessible through the MARS portal.

- **Environment Visualization (EVIS)**. EVIS produces forecasted weather effects on tactical missions and makes these available and advertised through the enterprise, enabling a user to access high resolution, mission-tailored weather effects summaries and related map overlays, and do this within their tactical decision-making cycle.

- **Information Assurance / Certification and Accreditation Process**. IA/CA will provide streamlined certification and accreditation services; makes recommendations and advocates for policy changes to support the certification and accreditation of net-centric operations.

- **Joint Surveillance Target Attack Radar System (Joint STARS)/ Affordable Moving Surface Target Engagement (AMSTE)**. Joint STARS/AMSTE brings a sophisticated moving-object tracking capability to Horizontal Fusion that enables tracking more targets with greater accuracy. Joint STARS/AMSTE can push these tracks to a ground station using either a line of sight or beyond line of sight data link, with the ground station converting both the moving target indicator reports and tracks into an XML message and sending it to the MAJIIC initiative for publication.

- **Knowledge Management in a Net-Centric Environment (KMINCE)**. Based on the National Ground Intelligence Center's mission, KMINCE provides tactical data, intelligence reports, publishing/posting tools, and collaboration functions within a web-accessible, service oriented environment. KMINCE allows operational users to search tactical databases through a Federated Search interface, build and post intelligence products within the collateral space and collaborate within the Horizontal Fusion MARS portal.

- **Multi-Sensor Aerospace-Ground Joint ISR Interoperability Coalition (MAJIIC)**. MAJIIC enhances U.S. joint and coalition ISR data interoperability and MAJIIC enhances information sharing via the development, testing, and implementation of data standards, XML schemas, and leading edge Web-based enterprise services. MAJIIC will "post-before-processing" to the collateral space U.S. and coalition near-real-time ISR sensor data and mission situational awareness information for discovery and smart-pull by Mars Portal users and value-adding command, control, communications, computers (C4), ISR systems.

- **Net-Centric Geospatial-Intelligence Services (NGS)**. NGS is a portlet on the Mars portal that provides the nation's war fighters and senior policymakers with access to geospatial intelligence (GEOINT), the exploitation and analysis of imagery and geospatial information to describe, assess and visually depict physical features and geographically referenced activities on Earth.

- **Network Basic Language Translation Services (NetBLTS)**. Network Basic Language Translation System (NetBLTS) enables non-linguists to quickly triage foreign documents and provides a translation aid to linguists. NetBLTS provides Optical Character Recognition (OCR), machine translation, and document management and indexing. Users can save extracted keywords and phrases, document translations, and foreign documents to a database repository for future analysis. The repository is accessible through the Horizontal Fusion Federated Search application.

- **Non-Obvious Relationship Awareness (NORA)**. NORA discovers relationships among people and organizations to answer the question, "Who knows whom?" and can be accessed by the war fighter through the Mars Portal and collateral space.

- **Trusted Wisdom**. Trusted Wisdom provides secure, mobile, real-time posting of reporting from field collectors. Information is tagged and accessible in the collateral space, where communities of interest can host rapid analysis and fusion of field collector reporting and technical collection data, and serve as venue for field collector, analyst, and war fighters to interact and rapidly develop fused actionable intelligence.

- **Visual Enterprise Monitoring (VEM)**. VEM provides a "window to the information flow" within a network to increase commanders' and decision-makers' overall situational awareness.

- **Visualization/Information Dominance (V/ID)**. V/ID bridges traditionally separate analytic processes, including data preparation and exploitation, and Web-enables it through data extraction, analysis and tagging, via the commercial ClearForest ClearTags entity extractor. Visualization capability is enhanced/enabled using the commercial Starlight toolkit.

- **Coalition Shared Intelligence Networked Environment (COSINE)**. COSINE combines analysis/production-sharing and cross-coalition information management married to a CENTRIXS -like secure network structure for intelligence exchange and content-based information security and release management capabilities that will allow individual coalition domains to quickly connect secure coalition command, control, and intelligence systems, share information and coordinate with both allied and coalition partners in a timely (near real-time) secure manner, and dynamically alter access to information when the need arises.

- **Cooperative Engagement Capability (CEC)**. CEC webTracks initiative provides access to the CEC sensor network's real-time air track picture and make it accessible to a variety of clients via intelligent pull to allow tailoring of the data requested for either current or historical track data and underlying measurement data, or streaming data to constantly update track movements.

- **Extensible Tactical C4I Framework (XTCF)**. XTCF is an open, extensible, plug-and-play architecture that will transform command and control data management services by providing an architecture that will rapidly add new value-added services and get new content providers and consumers quickly onto the collateral space in a dynamic battle space.

- **Global Net-Centric Surveillance and Targeting (GNCST)**. GNCST is developing a capability to demonstrate model-based fusion of upstream data from multiple intelligence sources to detect, locate and identify time-critical targets and targets of national interest, and to distribute target reports to tactical users via collateral networks in tactically relevant timelines.

- **Integration of Non-Traditional Information Sources (INTIS)**. INTIS uses non-traditional sensors, the F/A-18 Hornet and the AH-64 Apache, to provide secure, rapid delivery of hostile surface-to-air missile and anti-aircraft electronic intelligence to the war fighter and the intelligence community, aiding in fast updates of the common operating picture and more accurate targeting information.

- **Naval Research Lab (NRL) Sensor Node**. Provides an airborne node on the collateral space for target location and detection to support ground troops and joint strike forces directly, by posting "sensor products" (e.g., imagery, data, reports) and alerts for immediate use in operational planning by various Web-enabled users.

- **Ocean Surveillance Information System Evolutionary Development (OED)**. As the only operational command, control, communications, computer, and intelligence system trusted to provide multi-level secure capability, OED supports the customers of US and partner Joint Intelligence Centers with information tailored to their clearance level, area of interest, and need to know.

- **Secure Mobile Networks**. Provides the war-fighter with secure, robust voice and data communication networks which enable collaboration even in highly dynamic, unpredictable, mobile wireless environments. Secure Mobile Networks are intelligent, resilient, and self-configuring networks that allow access to global assets in the field even when direct links with reach back communications are not available.

- **Trusted Workstation (TWS)**. TWS provides intelligence analysts and operational war fighters with on-demand simultaneous access to common and mission-critical desktop applications running at multiple security domains from a single ultra-thin-client workstation.

- **Ubiquitous Automated Information Manager (U-AIM)**. U-AIM enables the aiming of external information resources to automatically discover, access, associate, and prioritize intelligence and information products, and focus and allocate resources on high priority information needs through a simple Web application that allows the war fighter to continuously formulate target or event nomination and receive alerts, all tailored to the war fighter's role and mission.

- **Warrior's Edge**. The Army's Warrior's Edge represents a dynamic ad hoc networked local sensing environment comprising soldiers and unattended and robotic sensors, each providing a user-tailored perspective of the combat situation to the war fighter during changing conditions to maximize mission success.

B.3. INDUSTRY NETWORK ENABLED EFFORTS

181. NNEC will be a federation of NATO and National systems the utilization of common standards is paramount for achieving the goals of NNEC. The vast majority of standards implemented in NNEC are expected to be open standards; so working with industry has been recognized as providing benefits to both NATO and industry. NATO monitors and evaluates the work from these industrial organizations and if found relevant incorporates their concepts in the development of the NISP.

182. Industry is also recommending a time phased approach for implementing systems. For example, Gartner described a life cycle approach to using standards at a symposium it held in October 2004[3]. This approach is depicted in Figure B.11

[3] Gartner Symposium IREXPO 2004, Lake Buena Vista, Florida, 17-22 October 2004

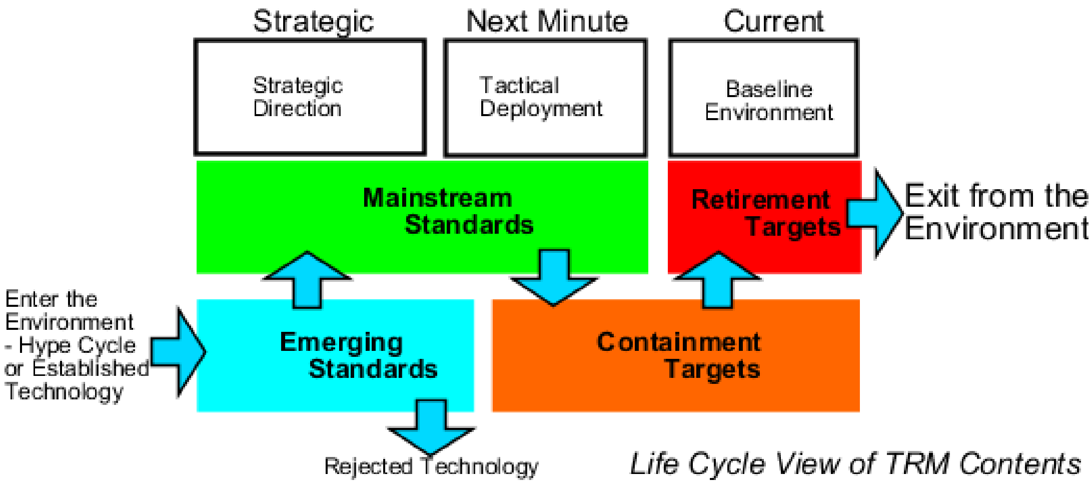


Figure B.11. Life Cycle Approach to Using Standards

183. The first phase is the emerging period, in which new technologies appear that may represent a significant advance (this is equivalent to emerging standards in the NISP). Once proven the standards may qualify for mainstream status, which means that they can be adopted without conditions (this is similar to mandatory standards in the NISP).

184. Eventually, newer standards emerge and mainstream standards move into the containment stage, that is, the standards are supported, but they are not suitable for new applications. Finally, t e retirement stage is reached, in which funding has been allocated to replace or retire the standards, and no further support will be provided.

C. REFERENCE MODELS

185. By definition, a reference model is an abstract framework for understanding the relationships among the entities within a specified environment. It enables the development of specific architectures using consistent standards or specifications supporting that environment.

186. A basic reference model will consists of the smallest set of unifying concepts, rules and relationships within a particular problem area, and is independent of specific standards, technologies, implementations, or other concrete details.

187. The relationship between the Reference Model and the implementation of a particular abstract concept is illustrated in Figure C.1.

188. Reference models are a standard definitive document or conceptual representation of a system or process. It provides a structure which allows the modules and interfaces of a system to be described in a consistent manner. In the context of near-future NATO environments, member nations could use these general models in designing architectures for net-enabled systems and platforms. These models represent the mid-term common framework that NATO nations can build to. A common framework is one of the keys to ensuring interoperability.

189. General reference models presented by industry do not take into account the typical Military or NATO unique situation. In the far-term, the NNEC approach will require us to establish a NATO Reference Model for services that tailored to complement the NATO mission.

190. Reference models are a standard definitive document or conceptual representation of a system or process. It provides a structure which allows the modules and interfaces of a system to be described in a consistent manner. In the context of future NATO environments, member nations can use these models in designing architectures for net-enabled systems and platforms.

191. Such assets can then participate in an NCO environment, by acting as interoperable nodes on a fixed or mobile ad hoc network. Once an element operates as a node, it can discover and register its needs and capabilities on a network, and use that network to communicate, interact and function with other nodes. The ultimate result is greater mission effectiveness.

C.1. PLATFORM ORIENTED ARCHITECTURE REFERENCE MODELS

C.1.1. The NATO Technical Reference Model (NTRM)

C.1.1.1. Purpose

192. Within the context of information systems, a Technical Reference Model (TRM) is a generally accepted construct that provides a basic set of concepts and a conceptual framework for identifying and resolving standards issues.

193. The main purpose of the NATO TRM (NTRM) is to structure the standards listed in the NATO Interoperability Standards and Profiles (NISP). As such, it should be a stable model and modifications should be made very carefully.

C.1.1.2. Structure

194. The NTRM focuses on separating data from applications and applications from the computing platform. This is a key principle when striving to attain a true open system environment. The NTRM provides the definitions necessary for designing and defining architectures and related service components. It also identifies service areas (i.e., capabilities that have been grouped together by functions), as well as their interfaces.

195. As indicated above, the NTRM is designed to decouple the application and external environment from the platform. This allows for portability of the application and independence from the external devices (e.g., disk, mouse, keyboard, LAN). This is accomplished by defining the application program interface (API) and external environment interface (EEI) accordingly.

C.1.1.3. Basic Entities and Interfaces

196. The basic elements of the NTRM are those identified in the POSIX OSE Reference Model. This model includes 3 classes of entities and 2 types of interfaces as follows:

- Application Software Entity
- Application Program Interface (API)
- Application Platform Entity
- External Environment Interface (EEI)
- External Environment Entity

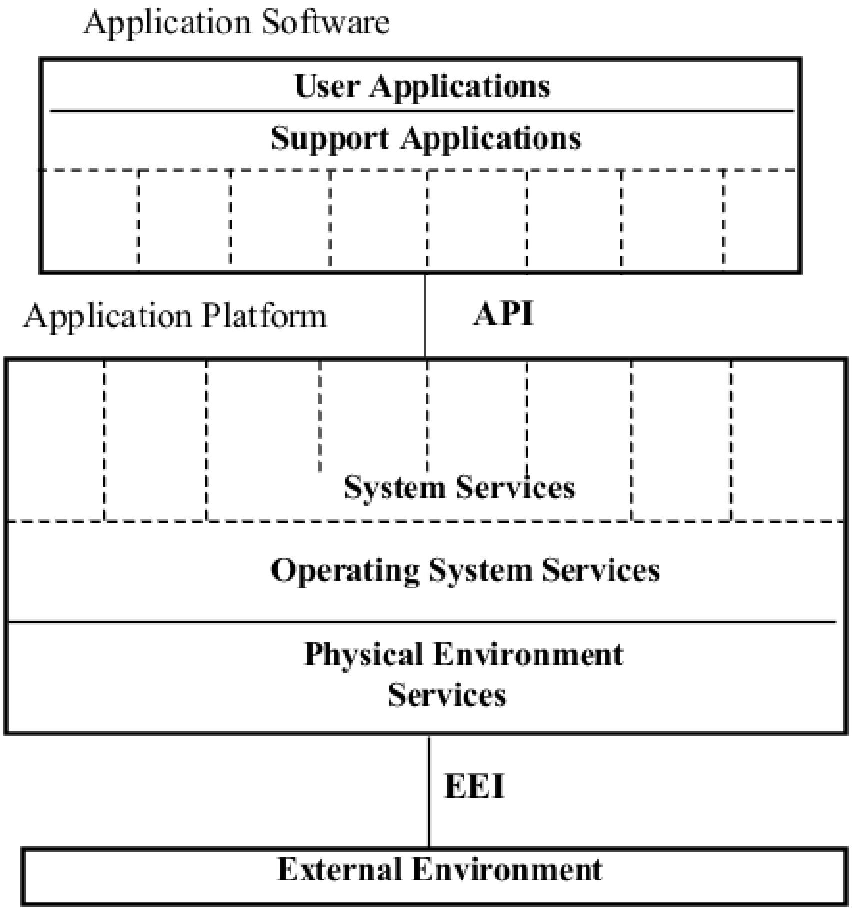


Figure C.1. NTRM Services View

C.1.1.3.1. Application Software Entity

197. The Application Software Entity includes both mission area and support applications.

- Mission-area or user applications implement specific user requirements (e.g., personnel, material, and management). This application software may be COTS, GOTS, custom developed, or consist of a combination thereof.
- Support applications (e.g., email and word processing) can be used to develop mission area specific applications or could be made readily available to the user. There are six support application categories:
  - Multimedia,
  - Communications,
  - Business Processing,
  - Environment Management,

- Database utilities,
- Engineering support.

C.1.1.3.2. Application Platform Entity

198. The Application Platform Entity contains the system services and the physical environment services. It is the second layer of the NTRM and includes the services in which information processing functionality is built.

- Service Areas are defined in Section C.1.1.4.
- Physical Environment services address the requirements for establishing the data interchange interface between physical resources and enable bus or communications link boards to address their peers in another node or system. They may also enable links to address processors or enable processors to address memory registers.

C.1.1.3.3. External Environment Entity

199. The External Environment Entity consists of external services that interact with the physical environment services of the application platform entity. These services are classified into the general categories of user services (e. g., mouse, display), information exchange services (e.g., memory stick) and communications services (e.g., LAN, WAN).

C.1.1.3.4. Interfaces

200. The Interfaces include both Application Programming Interfaces (API) and External Environment Interfaces (EEI)):

- The APIs are the interface between the application software entity and the application platform. They constitute a collection of standards-based interfaces,
- The External Environment Interfaces (EEI) are defined as the interfaces between the application platform and the external environment across which services are available, primarily in support of system and application software interoperability. User and data portability are provided directly by the EEI and provide the interfaces between the application platform entity and the external environment.

201. A concept of internal interfaces (IIs), not included in the previous version of the model, complements the notions of APIs and EEIs:

202. IIs are the interfaces within the system entities, both between sub-entities at the same level and sub-entities at different levels where sub-entities at a higher level make use of services offered by a lower level one. (A direct communication between two sub-entities at the same level is only possible at the lowest level).

203. The interfaces are in principle supported or realized by commonly defined data models and structures (e.g., ACP 133 B Schema Information for Directory Services).

204. Users should assess their own requirements and create a listing of services, interfaces, and standards that satisfy their own mission-area needs in conjunction with the NTRM accordingly.

205. In addition, the NTRM can accommodate a wide variety of general- and special purpose systems. From the perspective of the application software entity, these services are provided by an application platform whether the particular services are provided from the local platform or from remote platforms that may comprise one or more nodes of a larger distributed system. The NTRM can also be applied in a distributed environment and networked environment.

206. The objective of the NTRM is to provide a complete, as well as extensive set of features and capabilities. The NTRM provides consistency for user applications from a broader community in order to address interoperability, open systems, acquisition, and management issues associated with commercial off-the-shelf (COTS) and Government off-the-shelf (GOTS) products.

C.1.1.4. Service Areas of the NTRM

207. The System Services of the Application Platform Entity are used to structure the standards listed in the NISP. The classes and sub- classes described in the NISP are to be considered part of the NTRM. The 12 System Services areas are:

- **User Interface Services.** These services define how users may interact with an application. The term user interface in this context means a graphical user interface (GUI). Standards are not only required for setting up and managing graphical windows, but also for the toolkit and generic 'look and feel'.

- **Data Management Services.** The management of data is central to most systems. To improve interoperability, data should be defined independently from the processes that create or use it, and should be maintained and shared among many processes.

- **Data Interchange Services.** These services provide support for the interchange of data between applications. They are designed to handle data interchange between applications on the same or on heterogeneous platforms.

- **Graphics Services.** These services provide functions required for creating and manipulating graphics.

- **Communication Services.** These services provide distributed applications support for data access and applications interoperability in heterogeneous or homogeneous networked environments.

- **Operating System Services.** These services are the core services needed to operate and administer the application platform and provide an interface between applications software and platform. Application programmers will use operating system services to obtain operating system functionality.

- **Internationalization Services.** Within the context of the NTRM, internationalization provides a set of services and interfaces that allow a user to define, select, translate and switch between different culturally related application environments supported by the particular implementation. Character sets and data representation services include the capability to input, store, manipulate, retrieve, communicate, and present data independently of the coding scheme used. This includes the capability to maintain and access a central character set repository of all coded character sets used throughout the platform.

- **System Management Services.** Information systems are composed of a wide variety of diverse resources that must be managed effectively to achieve the goals of an open system environment. While the individual resources (such as printers, software, users, processors) may differ widely, the abstraction of these resources as managed objects allows for their treatment in a uniform manner.

- **Security Services.** Different groups of individuals within and across the various NATO applications need to work with specific sets of data elements. Access to these sets of data elements is to be restricted to authorized users. Satisfaction of this requirement has traditionally been accomplished by the implementation of separate information systems. Organizations cannot continue to afford to implement separate information systems to satisfy this requirement, nor is it effective to require the user to change interface components every time the need arises to operate with a different restricted data set. Significant benefit will accrue when an individual information system can effectively support the needs of different groups of users and data sets.

- **Distributed Computing Services.** These services provide specialized support for applications that may be physically or logically dispersed among computer systems in a network, but yet wish to maintain a co-operative processing environment. The classical definition of a computer becomes blurred as the processes that contribute to information processing become distributed across a facility or a network. As with other cross-cutting services, the requisite components of distributed computing services typically exist within particular service areas.

- **Software Engineering Services.** The procedural aspect of an application is embodied in the programming languages used to code it. Additionally, professional system developers require methods and tools appropriate to the development and maintenance of applications.

- **Common C2 Applications Services.** These services provide the ability to view data (i.e., share) in a common way across the network. Common C2 Applications Services promote interoperability among diverse functional mission area domains and may be executed between both individual and multiple functional application domain areas.

C.1.2. NCOE Component Model (NCM)

208. The NTRM provides the structural basis for defining the NCOE (NATO Common Operating Environment) Component Model (NCM).

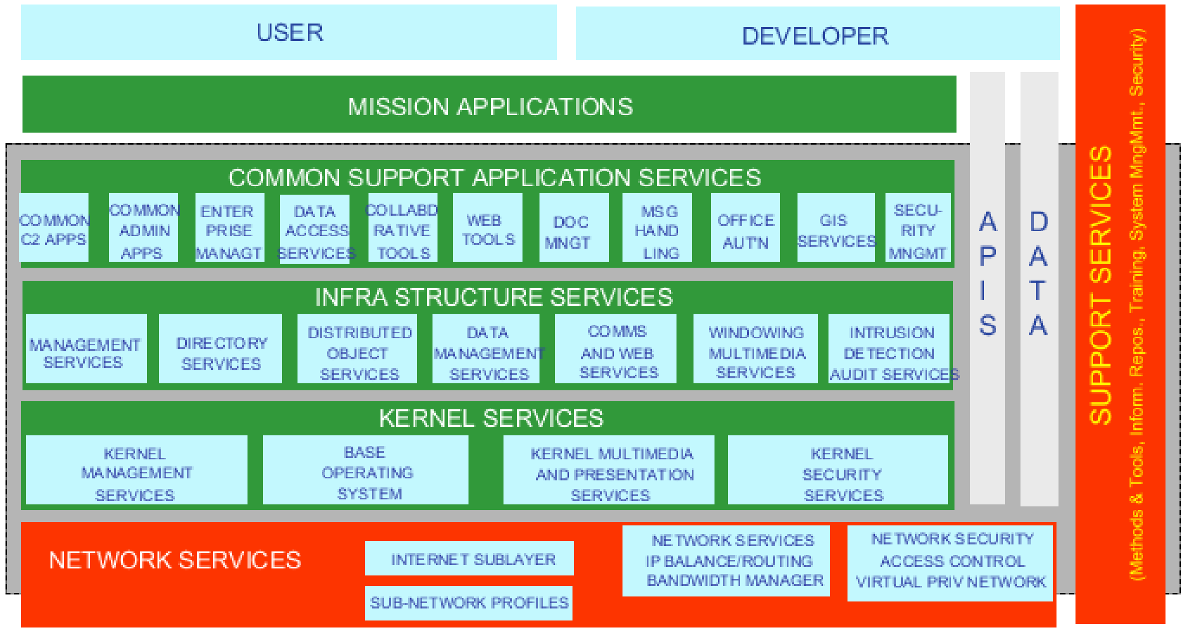


Figure C.2. The NCOE Component Model

209. The principal components of the NCM (see Figure C.2) include:

- **Kernel Services.** The Kernel Services are that subset of the NCOE component segments, which are required for all compliant platforms. At a minimum, this sub-set would consist of the operating system, windowing software, security services, segment installation software and an executive manager.

- **Infrastructure Services.** Infrastructure services are those services that directly support the flow of information across NATO systems. Infrastructure services provide a set of integrated capabilities that the applications will access to invoke NCOE services.

- **Common Support Application Services.** Common Support Application Services provide services to process and view data in a common way (share data) across the network. The NCOE common support application services promote interoperability among various Mission Applications.

- **Network Services.** The NCOE Network Services constitute the basic interface between the platform and the underlying networking infrastructure and include the Internet Sub-layer services.

[1] The term API is to be understood in a local sense (e.g. APIs between components interfaced on a user desktop), as well as in a distributed sense (e.g. interfaces from legacy or external components using an Object Request Broker (ORB) through IDL interfaces).

- **Application Programming Interfaces**. Applications are integrated into the NCOE through a common set of Application Programming Interfaces (APIs). The APIs are invoked by the applications and services as required [1].
- **Data Component Definition**. The data component refers to the way in which data is taken into account in the NCOE and is related to the main components of the NCOE (Common Support Application Services, Infrastructure Services, Kernel Services) and even, out of NCOE components, in the strictest sense, to Mission Applications.
- **Support Services**. The NCOE Support Services include Methods & Tools, Information Repository, Training Services, System Management and Security.

C.2. SERVICE ORIENTED ARCHITECTURE REFERENCE MODELS

C.2.1. What is SOA?

210. Service Oriented Architecture (SOA) is a paradigm for organizing and using distributed capabilities that may be under the control of different ownership domains. It is natural in such a context to think of one person's needs being met by capabilities offered by someone else or, in the world of distributed computing, one computer agent's requirements being met by a computer agent belonging to a different owner. There is not necessarily a one-to-one correlation between needs and capabilities; the granularity of needs and capabilities vary from fundamental to complex, and any given need may require the combining of numerous capabilities while any single capability may address more than one need. The perceived value of SOA is that it provides a powerful framework for matching needs and capabilities and for combining capabilities to address those needs.

211. Visibility, interaction, and effect are key concepts for describing the SOA paradigm. **Visibility** refers to the capacity for those with needs and those with capabilities to be able to see each other to interact. Visibility is typically enhanced through the use of metadata to describe such aspects as functional and technical requirements, related constraints and policies, and mechanisms for interaction. For maximum visibility, metadata must be in a form in which its syntax and semantics are widely accessible and understandable.

212. Whereas visibility introduces the possibilities for matching needs to capabilities (and vice versa), **interaction** is the activity of using the capability. Typically mediated by the exchange of messages, an interaction proceeds through a series of information exchanges and invoked actions. There are many facets of interaction; but they are all grounded in a particular **execution context** – the set of technical and business elements that together form a path between those with needs and those with capabilities and that permit information to be exchanged, actions to be performed and provides a decision point for any policies and contracts that may be in force.

213. The purpose of using a capability is to realize one or more **real world effects**. At its core, an interaction is "an act" rather than "an object" and the result of a interaction is an effect (or a set/series of effects).

214. The expected effects, together with relevant preconditions associated with those effects, should be made visible as part of the capability metadata and form an important part of the assessment as to whether a given capability matches similarly described needs. It is not possible to describe every possible effect of using a capability: indeed a cornerstone of SOA is that such knowledge is not necessary.

215. A concept that is considered central to SOA has not yet been mentioned – that of **service**. Both needs and capabilities exist outside of SOA. What distinguishes SOA is the perceived improvement in bringing needs and capabilities together. **In SOA, services are the mechanism by which needs and capabilities are brought together.** SOA is not the solution of domain problems but rather a way of organizing a wider array of possibilities to generate a domain solution. By itself, SOA does not provide a solution to a difficult domain problem where a satisfactory solution does not already exist. SOA can, however, provide an organizing and delivery paradigm that enables one to get more value from use of both solutions which are locally “owned” and solutions under the control of others. It also enables one to express solutions in a way that makes it easier to modify or evolve the identified solution or to try alternate domain solutions.

216. The concepts of visibility, interaction, and effect apply directly to services in the same manner as these were described for the general SOA paradigm. Visibility is promoted by the **service description** which contains the information necessary to interact with the service and describes this in such terms as the service inputs, outputs, and associated semantics. The service description also conveys what is accomplished when the service is invoked and the conditions for invoking the service. In general, entities (people and organizations) offer capabilities through services and act as **service providers**. Those with needs who make use of capabilities through their associated services are referred to as **service consumers**. The service description allows prospective consumers to decide if the service is suitable for their current needs and establish whether a consumer satisfies the requirements, if any, of the service provider to be permitted access.

217. Having described what is SOA, it is appropriate to note several things which are related but are not necessary attributes or restrictions.

218. SOA identifies necessary aspects of interactions involving multiple ownership domains; however, it does not directly embody concepts relating to ownership.

219. SOA is commonly implemented using Web services, but services can be made visible, support interaction, and generate effects through other implementations.

220. By following a Service-Oriented Architecture NATO nations can then participate in an NNEC environment, by acting as interoperable nodes on a fixed or mobile ad hoc network. Once an element operates as a node, it can discover and register its needs and capabilities on a network, and use that network to communicate, interact and function with other nodes. The ultimate result is greater mission effectiveness.

221. In February of 2005, the Organization for the Advancement of Structured Information Standards (OASIS) started standards work to define an SOA reference model (SOA-RM) by establishing a technical committee for that sole purpose.

222. The goal of OASIS SOA-RM technical committee is to "establish a Reference Model to encourage the continued growth of specific and different SOA implementations whilst preserving a common layer that can be shared and understood between those or future implementations."

223. Achievement of their goals for this reference model will be done by defining the fundamental nature of SOA, and emerge with a common vocabulary and understanding of SOA. As such, it will provide a conforming reference that treats SOA as a powerful abstract model that is independent of the various inevitable technology evolutions.

C.2.2. The Benefits of Service Oriented Architecture

224. The main drivers for SOA-based architectures are the requirement to facilitate the manageable growth of large-scale enterprise systems, the requirement to facilitate Internet-scale provisioning and use of services and the requirement to reduce costs in organization to organization cooperation.

225. The value of SOA is that it provides a simple scalable paradigm for organizing large networks of systems that require interoperability to realize the value inherent in the individual components. Indeed, SOA is scalable because it makes the fewest possible assumptions, including about the network and also minimizes any trust assumptions that are often implicitly made in smaller scale systems.

226. An architect using SOA principles is better equipped, therefore, to develop systems that are scalable, evolvable and manageable. It should be easier to decide how to integrate functionality across ownership boundaries. For example, a large company that acquires a smaller company must determine how to integrate the acquired IT infrastructure into its overall IT portfolio.

227. Through this inherent ability to scale and evolve, SOA enables an IT portfolio which is also adaptable to the needs of a specific problem domain or process architecture. The infrastructure SOA encourages is also more agile and responsive than one built on an exponential number of pair-wise interfaces. Therefore, SOA can also provide a solid foundation for business agility and adaptability.

C.2.3. Overview of the Model

228. A key concept of SOA is that of **service** . In general, entities (people and organizations) create capabilities to solve or support a solution for the problems they face in the course of their business. SOA is a way to organize the world around this key concept of service. The noun “service” is defined in dictionaries as “The performance of work (a function) by one for another.” However, service, as the term is generally understood, also combines the following related ideas:

- The capability to perform work for another
- The specification of the work offered for another
- The offer to perform work for another

229. These concepts emphasize a distinction between a capability and the ability to bring that capability to bear in the context of SOA, where the capability exists independently of SOA. The term service should, therefore, be understood as a set of separate, yet interrelated and more precise concepts. These concepts are an offer, interaction and effect.

230. The concept of an **offer** follows directly from the dictionary definition of service: 'by one' and 'for another.' In general terms, an offer is a proposal; made by providers which may possess a capability that address a need. In order to use a service, it is necessary to know that it exists, what is accomplished if the service is invoked, how the service is invoked, and other characteristics. Collectively this is the service **visibility**. When given an explicit searchable form, this information allows, for example, prospective consumers to decide if the service is suitable for their current needs and establish whether a consumer satisfies any requirements of the service provider to be permitted access. This information constitutes the **service description**.

231. The convergence of a capability and a need results in an **interaction**. In an SOA, interaction is effected by exchanging information between service providers and consumers. Typically this is achieved by exchanging messages using a standardized protocol; however, there are many modalities possible for interacting with services.

232. At its core, an interaction is “an act” rather than “an object. Therefore, interaction is the focus of the interfaces and behaviour necessary to support the interaction. Recall that interaction may, and typically does, involve crossing ownership boundaries. SOA identifies some of the necessary aspects of interactions involving multiple ownership domains; however, it does not directly embody concepts relating to ownership.

233. The final key concept is the **real world effect** of using services; it is always the case that there is an intended purpose to providing a service and similarly to using a service. Given that there is often an ownership boundary between the service provider and consumer, there is a natural distinction to be drawn between the public interactions with a service and the private actions of both the service provider and consumer. This distinction maintains and encourages independence of each service participant which, in turn, greatly enhances the scalability and security attributes of SOA. Focus can be directed to the public aspects of using a service by examining the **conditions** of using a service and the **expectations** that arise as a result of using the service. Service conditions are loosely associated with the **service policies** and the expectations with **service contracts** .

C.2.4. The SOA Reference Model

C.2.4.1. Service

234. A **service** is a means to access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. A service is provided by one entity – the **service provider** – for use by others, but the eventual consumers of the service may not be known to the service provider and may demonstrate uses of the service beyond the scope originally conceived by the provider,

- Information returned in response to a request,
- A change to the shared state of defined entities, or
- Some combination of the above.

235. Note, the **service consumer** in (1) does not typically know how the information is generated, e.g. whether it is extracted from a database or generated dynamically; in (2), the service consumer does not typically know how the state change is effected. In either case, the service consumer would need to provide input parameters required by the service and the service would return information, status indicators, or error descriptions, where both the input and output are as described by the data model exposed through the published service interface. Note that the service may be invoked without requiring information from the consumer (other than a command to initiate action) and may accomplish its functions without providing any return or feedback to the consumer.

236. The service concept above emphasizes a distinction between a capability that represents some functionality created to address a need and the point of access to bring that capability to bear in the context of SOA. It is assumed that capabilities exist outside of the SOA. In actual use, maintaining this distinction may not be critical (i.e. the service may be talked about in terms of being the capability) but the separation is pertinent in terms of a clear expression of the nature of SOA and the value it provides.

C.2.4.2. Service description

237. The service description represents the information needed in order to use a service. It may be considered part of or the complete set of the metadata associated with a service. In any case, the service description overlaps and shares many common properties with service metadata. In most cases, there is no one “right” set of metadata but rather the metadata content depends on the context and the needs of the parties using the associated entity. The same holds for a service description. While there are certain elements that are likely to be part of any service description, most notably the data model, many elements such as function and policy may vary.

238. Best practice suggests that the service description should be represented using a standard, referenceable format. Such a format facilitates the use of common processing tools (such as discovery engines) that can, in turn, capitalize on the service description.

239. While the concept of a SOA supports use of a service without the service consumer needing to know the details of the service implementation, the service description makes available critical information that a consumer needs in order to decide whether or not to use a service. In particular, a service consumer must possess the following items of information:

1. That the service exists and is **reachable** (i.e., the service is **visible** to the service consumer and there are sufficient mechanisms in place for the service participants to be able to interact);
2. That the service performs a certain function or set of functions;

3. That the service operates under a specified set of constraints and policies;
4. That the service will (to some implicit or explicit extent) comply with policies as prescribed by the service consumer;
5. How to interact with the service in order to achieve the required objectives, including the format and content of information exchanged between the service and the consumer and the sequences of information exchange that may be expected.

240. Subsequent sections of this document will deal with these aspects of a service in detail but the following subsections will describe the relationship of these information items to the service description.

C.2.4.2.1. Service Reachability

241. A service description should include sufficient data to permit a service consumer and service provider to exchange information. This might include metadata (such as the location of the service and what information protocols it supports and requires) and information that allows the service consumer to determine if the service is currently reachable or not.

C.2.4.2.2. Service Functionality

242. Item 2 relates to the need to unambiguously express the function(s) of the service and the real world effects that result from it being invoked. This portion of the description needs to be expressed in a way that is generally understandable by service consumers but able to accommodate a vocabulary that is sufficiently expressive for the domain for which the service provides its functionality. The description of functionality may include, among other possibilities, a textual description intended for human consumption or identifiers or keywords referenced to specific machine-processable definitions. For a full description, it may be useful to indicate multiple identifiers or keywords from a number of different collections of definitions.

243. Part of the description of functionality may include underlying technical assumptions that determine the limits of functionality exposed by the service or of the underlying capability.

C.2.4.2.3. Policies Related to a Service

244. Items 3 and 4 from Section 2.2.4.2 relate to the service description's support for associating constraints and policies with a service and providing necessary information for prospective consumers to evaluate if a service will act in a manner consistent with the consumer's constraints and policies.

245. In some situations the consumer may similarly provide an indication of its constraints and policies to support a service' need to do a similar evaluation of suitability. Thus, both prospective consumers and providers are likely to use the service description to establish what Section 2.2.5.3 refers to as the **execution context**.

C.2.4.2.4. Service Interface

246. The service interface is the means referred to in Item 5 for interacting with a service. It includes the specific protocols, commands, and information exchange by which actions are initiated that result in the real world effects as specified through the service functionality portion of the service description.

247. The specifics of the interface should be syntactically represented in a standard referenceable format. These prescribe what information needs to be provided to the service in order to exercise its functionality and/or the results of the service invocation to be returned to the service consumer. This logical expression of the set of information items associated with the consumption of the service is often referred to as the service data model. It should be noted that the particulars of the standard referenceable format is beyond the scope of the reference model. However, requiring that mechanisms be available (in order to define and retrieve such definitions) is fundamental to the SOA concept.

C.2.4.3. Descriptions and Metadata

248. One of the hallmarks of a Service Oriented Architecture is the degree of documentation and description associated with it; particularly machine processable descriptions – otherwise known as metadata.

249. The purpose of this metadata is to facilitate integration, particularly across ownership domains. By providing public descriptions, it makes it possible for potential participants to construct applications that use services and even offer compatible services. Standardizing the formats of such metadata reduces the cost and burden of producing the descriptions necessary to promote reuse and integration.

C.2.4.3.1. The roles of description

250. An important additional benefit of metadata – as opposed to informal natural language descriptions – is its potential to facilitate automated software development. Both service providers and service consumers can benefit from such automation – reducing the cost of developing such systems.

251. For example, metadata can be used as a basis of discovery in dynamic systems. Metadata can assist in managing a service, validating and auditing usage of services which may also be simplified by rich metadata. It can also help ensure that requirements and expectations (regarding the content of any data interchanged) are properly interpreted and fulfilled.

C.2.4.3.2. The Limits of Description

252. There are well-known theoretic limits on the effectiveness of descriptions – it is simply not possible to specify, completely and unambiguously the precise semantics of a service. There will always be unstated assumptions made by the describer of a service that must be implicitly

shared by readers of the description. This applies to machine processable descriptions as well as to human readable descriptions.

253. Fortunately, complete precision is not necessary either – what is required is sufficient precision to enable required functionality.

254. Another kind of limit of service descriptions is more straightforward: whenever a repository is searched using any kind of query there is always the potential for zero or more responses. There may be many reasons why a multiplicity of responses is returned: there might be several versions of the service, there might be competing services that offer overlapping functionality and there might be services from multiple different providers.

255. In the case that there is more than one response, this set of responses has to be converted into a choice of a single service in order for a service consumer to ensure the required function performed. In a multi-provider scenario, that choice must also take into account real world aspects of the service – such as whether the service consumer can identify the provider, can or should trust the provider, and whether the provider is reliable and timely in delivering the service offered. It is unlikely that all such factors can be easily and securely encoded in descriptions and search criteria.

C.2.5. Interacting with Services

256. Interacting with a service involves exchanging information with the service and performing actions against the service. In many cases, this is accomplished by sending and receiving messages, but there are other modes possible that do not involve explicit message transmission. However, for simplicity, we often refer to message exchange as the primary mode of interaction with a service. The forms of information exchanged and understood, together with the mechanisms used to exchange information, constitute the **service interface** –.

257. The key concepts that are important in understanding what it is involved in interacting with services are the **data model**, the **process model** , the **execution context** and the **expectations** about the interaction.

C.2.5.1. Data Model

258. The data model of a service is a characterization of the information associated with the use of the service.

259. The scope of the data model includes the format of exchanged information, the structural relationships within documents and the definition of terms used. Typically, only information about, and data potentially included in, an exchange with a service are generally considered as being part of that service's data model.

260. There are two important aspects of a data model that are important in interpreting information exchange – the structure of the information and the meaning assigned to elements of the information. Particularly for information that is exchanged across an ownership boundary, the

interpretation of strings and other tokens in the information is a critical part of the semantics of the interaction.

C.2.5.1.1. Structure

261. Understanding the representation, structure and form of information exchanged is a key initial step in ensuring effective interactions with a service. There are several levels of such structural information; ranging from the encoding of character data, through the use of formats such as XML, SOAP and schema-based representations.

262. A described data model typically has a great deal to say about the form of messages, about the types of the various components of messages and so on. However, pure “typed” information is not sufficient to completely describe the appropriate interpretation of data.

C.2.5.1.2. Semantics and Ontology

263. The primary task of any communication infrastructure is to facilitate the exchange of information and the exchange of intent. For example, a purchase order combines two somewhat orthogonal aspects: the description of the items being purchased and the fact that one party intends to purchase those items from another party. Even if for exchanges that do not cross any ownership boundaries, exchanges with services have similar aspects: this is an update to the customer profile with these changes.

264. Especially in the case where the exchanges are across ownership boundaries, a critical issue is the interpretation of the data. This interpretation must be consistent between the participants in the service interaction. Consistent interpretation is a stronger requirement than merely type (or structural) consistency – the tokens in the data itself must also have a shared basis.

265. An ontology is a formal description of terms and the relationships between them in a given context. It will include information about how terms should be interpreted within a given context, constraints on and functions of valid values for the data and associated properties, as well as information about possible relationships of some terms to other terms (hierarchical, class/sub class, associative, dependent, etc.).

266. The role of explicit ontologies in an SOA is to provide a firm basis for selecting correct interpretations for elements of information exchanged.

267. Ontologies also provide a point of context to facilitate the reinterpretation of data. Such a reinterpretation is effectively represented as a particular traversal of the graph of concepts and relationships embodied in the ontology. How much automation of ontology walking is appropriate will depend on the nature of the service and the service participants.

268. Note that, for the most part, it is not expected that service consumers and providers would actually exchange ontologies in their interaction – the role of the ontology is a background one – it facilitates sound interactions. Hence ontology references are mostly to be found in **service descriptions**.

269. More specifically, and in order for a service to be consistent, the service should make consistent use of terms as defined in an ontology. Specific domain semantics are beyond the scope of this reference model; but there is a requirement that the service interface enable providers and consumers to identify unambiguously those definitions that are relevant to their respective domains.

C.2.5.2. Behavioural model

270. The second key requirement for successful interactions with services is knowledge of the process or temporal aspects of interacting with the service. Loosely, this can be characterized as knowledge of the actions on, responses to and temporal dependencies between actions on the service.

271. For example, in a security-controlled access to a database service, the actions available to a service consumer might include presenting credentials, requesting database updates and reading results of queries. The security may be based on a challenge-response protocol. For example, the initiator presents an initial token of identity, the responder presents a challenge and the initiator responds to the challenge in a way that satisfies the service. Only after the user's credentials have been verified will any action that queries and/or updates the database be accepted. The sequences of actions involved are a critical aspect of the knowledge required for successful use of the secured database service.

272. There are other aspects of the behaviour of services that are important. These include, for example, whether the service is transactional, idempotent or long running. As a particular example, a service that supports updating an account balance with a transaction is typically idempotent; i.e., the state of the account would not be affected should a subsequent interaction be attempted for the same transaction.

C.2.5.2.1. Action model

273. The **action model** of a service is about the individual actions that may be invoked against the service. Of course, a great portion of the behaviour resulting from an action may be private; however, the expected public view of a service surely includes the implied effects of actions.

C.2.5.2.2. Process Model

274. The **process model** characterizes the temporal relationships between actions and events associated with interacting with the service. It is fairly common to partition the process model associated with a service into two levels: the particular sequences of operations needed to achieve single service exchanges and longer term transactions. These two levels may be nested – a long running transaction is often composed of sequences of exchange patterns.

275. Note that although the process model is an essential part of this Reference Model, its extent is not completely defined. In some architectures the process model will include aspects that are not strictly part of SOA – for example, in this reference model we do not address the orchestration of multiple services – although orchestration and choreography may be part of the process

model of a given architecture. At a minimum, the process model must cover the interactions with the service itself.

C.2.5.2.3. Higher-order attributes of processes

276. Beyond the straightforward mechanics of interacting with a service there are other, higher-order, attributes of services’ process models that are also often important. These can include whether the service is **idempotent**, whether the service is **long-running** in nature and whether it is important to account for any **transactional** aspects of the service.

277. A service is **idempotent** if subsequent attempts to perform identical transactions are discounted. For example, it often important that a bank will only cash a check once – subsequent attempts to cash the same check should be ignored, rejected or initiate an alert process. Note that idempotency is not the same as effect-free or stateless: a service that always returns the same results is idempotent, but only by virtue of the fact that it does not change from invocation to invocation.

278. Idempotency is an important attribute of a service in an environment where there is a significant possibility that the interaction between the service provider and consumer may be interrupted – whether by a network issue or simply one of the parties dropping out. A strategy for recovering from such a breakdown is to attempt to repeat the interaction – an idempotent service is required to ignore such repetitions should the transaction have been completed beforehand.

279. A service is **long-running** if the activities engendered by an interaction are likely to persist beyond the immediate interaction itself. For example, a classic book selling service might be viewed as a long-running service: the activity started by the purchase of the book may take days or weeks to complete. It can be important to account for a long-running process as it has implications for the kinds of infrastructure needed – both by the service provider and by the service consumer – in order to be able to keep track of the progress of the interaction.

280. Often, once a business-level contract has been agreed on, it can be difficult or impossible to simply cancel the consequences of the agreement. This is particularly an issue when the agreement of several parties is necessary simultaneously. For example, booking a vacation may require a flight ticket as well as a hotel room – without either component the result is not a vacation. However, the airline typically will not have a relationship with the hotel. If there are no hotel rooms available for the proposed vacation then the airline ticket will need to be cancelled.

281. The process of reversing a previously completed transaction – backing out of the airline booking for example – is likely to be quite different to the process for the original transaction; possibly even involving a different service. Knowledge of such compensatory actions is a key aspect of interacting with **transactional** services.

C.2.5.3. Actualized Services

282. The **execution context** of a service interaction is the set of infrastructure elements, process entities and policy assertions that are deployed as part of the instantiated service interaction. In

effect, the execution context defines the point of contact between abstractions such as service descriptions which are mostly about the potential for interaction and an actually executing service. It is the point of measurement between the service description and reality, between theory and practice.

283. The execution context is not limited to one side of the interaction; rather it concerns the totality of the interaction – including the service provider, the service consumer and the common infrastructure needed to mediate the interaction.

284. The execution context is central to many aspects of a service interaction. It defines, for example, the decision point for any policy enforcement relating to the service interaction. Note that a policy decision point is not necessarily the same as an enforcement point: an execution context is not by itself something that lends itself to enforcement. On the other hand, any enforcement mechanism of a policy is likely to take into account the particulars of the actual service interaction.

285. The execution context also allows us to distinguish services from one another. Different instances of the same service – denoting interactions between a given service provider and different service consumers for example – are distinguished by virtue of the fact that their execution contexts are different.

286. Finally, the execution context is also the context in which the interpretation of data that is exchanged takes place – it is where the symbol grounding happens as it were. A particular string has a particular meaning in a service interaction in a particular context – the execution context.

C.2.6. Real World Effect

287. There is always a particular purpose associated with interacting with a service. Conversely, a service provider (and consumer) often has a priori conditions that apply to its interactions. The service consumer is trying to achieve some result by interacting with the service, as is the service provider. At first sight, such a goal can often be expressed as “trying to get the service to do something” This is sometimes known as the **real world effect** of using a service.

288. The internal actions that a service providers and consumers perform as a result of participation in service interactions are, by definition, private and fundamentally unknowable. By unknowable we mean both that external parties cannot see others’ private actions and, furthermore should not have explicit knowledge of them. Instead we focus on the state that is shared between the parties – the **shared state**. Actions by service providers and consumers lead to modifications of this shared state; and that in turn leads to modified **expectations** by the participants.

289. Note that there does not need to be a third party to act as a kind of escrow for the shared state between service providers and consumers. The elements of the shared state are inferred from the communication that has occurred between the participants together with other context as necessary. Of course, in the case where adjudication is a possibility, it becomes prudent to record the interaction – so that disputes can be arbitrated.

290. Although there is not necessarily a one-to-one correspondence, the natural container for the conditions applying to a service is the **service policy**. Similarly, the natural container for the expectations arising from a service is the **service contract**.

C.2.7. Policies and Contracts

291. A **policy** represents some constraint or condition on the use, deployment or description of an owned entity as defined by any participant. A **contract**, on the other hand, represents an agreement by two or more parties. Like policies, agreements are also about the conditions of use of a service; they may also constrain the expected real world effects of using a service. The reference model is focused primarily on the concept of policies and contracts as they apply to services. We are not concerned with the form or expressiveness of any language used to express policies and contracts.

C.2.7.1. Service Policy

292. A policy is a statement of the obligations, constraints or other conditions of use of a given service that expresses intent on the part of a participant. More particularly, policies are a way for expressing the relationship between the **execution context** and the **data** and **behaviour models** associated with the service.

293. Conceptually, there are three aspects of policies: the policy assertion, the policy owner (sometimes referred to as the policy subject) and policy enforcement.

294. For example, the assertion: “All messages are triple-DES encrypted is an assertion regarding the forms of messages. As an assertion, it is measurable: it may be true or false depending on whether the traffic is encrypted or not. Policy assertions are often about the way the service is realized; i.e., they are about the relationship between the service and its execution context.

295. A policy always represents a participant’s point of view. An assertion becomes the policy of a participant when they make it their policy. This linking is normally not part of the assertion itself. For example, if the service consumer declares that “Al messages are triple-DES encrypted”, then that reflects the policy of the service consumer. This policy is one that may be asserted by the service consumer independently of any agreement from the service provider.

296. Finally, a policy may be enforced. Techniques for the enforcement of policies depend on the nature of the policy. Conceptually, service policy enforcement amounts to ensuring that the policy assertion is consistent with the real world. This might mean preventing unauthorized actions to be performed or states to be entered into; it can also mean initiating compensatory actions when a policy violation has been detected. An unenforceable constraint is not a policy; it would be better described as a wish.

297. Policies potentially apply to many aspects of SOA: security, privacy, manageability, Quality of Service and so on. Beyond such infrastructure-oriented policies, participants may also express business-oriented policies – such as hours of business, return policies and so on.

298. Policy assertions should be written in a form that is understandable to, and processable by, the parties to whom the policy is directed. Policies may need to be automatically interpreted, depending on the purpose and applicability of the policy and whether it might affect whether a particular service is used or not.

299. A natural point of contact between service participants and policies associated with the service is in the service description. It would be natural for the service description to contain references to the policies associated with the service.

C.2.7.2. Service Contract

300. Where a policy is associated with the point of view of individual participants, a contract represents an agreement between two or more participants. Like policies, contracts can cover a wide range of aspects of services: quality of service agreements, interface and choreography agreements and commercial agreements

301. Thus, following the analysis above, a service contract is a measurable assertion that governs the requirements and expectations of two or more parties. Unlike policy enforcement, which is usually the responsibility of the policy owner, contract enforcement may involve resolving disputes between the parties to the contract. The resolution of such disputes may involve appeals to higher authorities.

302. Like policies, contracts may be expressed in a form that permits automated interpretation. Where a contract is used to codify the results of a service interaction, it is good practice to represent it in a machine processable form. This facilitates automatic service composition, for example. Where a contract is used to describe over-arching agreements between service providers and consumers, then the priority is likely to make such contracts readable by people.

C.2.8. Visibility

303. For a service provider and consumer to interact with each other they have to be able to ‘see’ each other. This is true for any consumer/provider relationship – including in an application program where one program calls another: without the proper libraries being present the function call cannot complete. In the case of SOA visibility needs to be emphasized because it is not necessarily obvious how service participants can see each other to interact.

304. Visibility is the relationship between service consumers and providers that is satisfied when they are able to interact with each other. Preconditions to visibility are awareness – typically the initiator in a service interaction must be aware of the other parties – willingness – the parties must be predisposed to interactions – and ability – the participants must be able to exchange information as part of a service interaction.

C.2.8.1. Awareness

305. A key aspect of visibility is awareness – both the service provider and the service consumer must have information that would lead them to know of the other’s existence. Technically, the prime requirement is that the initiator of a service interaction has knowledge of the responder.

The fact of a successful initiation is often sufficient to inform the responder of the other's existence.

306. Awareness of service offerings is often mediated by various discovery mechanisms. For a service consumer (say) to discover a service provider, the service provider must be capable of making details of the service (notably service description and policies) available to potential consumers; and consumers must be capable of finding that information.

307. Service discoverability requires that the service description and policy – or at least a suitable subset thereof – be available in such a manner and form that, directly or indirectly, an awareness of the existence and capabilities of the service can become known to potential consumers. The extent to which the discovery is "pushed by the service provider, "pulled" by a potential consumer, subject to a probe or another method, will depend on many factors.

308. For example, a service provider may advertise and promote their service by either including it in a service directory or broadcasting it to all consumers; potential consumers may broadcast their particular service needs in the hope that a suitable service responds with a proposal or offer or a service consumer might also "probe" a entire network to determine if suitable services exist. When the demand for a service is higher than the supply, then by advertising their needs, potential consumers are likely to be more effective then service providers advertising offered services.

309. One way or another, the potential consumer must acquire a sufficient description to evaluate whether the service matches their expectations and, if so, the method for the consumer to establish a contract and invoke the service.

C.2.8.2. Willingness

310. Associated with all service interactions is intent – it is an intentional act to initiate and to participate in a service interaction. For example, if a service consumer discovers a service via its description in a registry, and the consumer initiates an interaction, if the service provider does not cooperate then there can be no interaction. In some circumstances it is precisely the correct behaviour for a service to fail to respond – for example, it is the classic defence against certain denial-of-service attacks.

311. The extent of a service participant's willingness to engage in service interactions may be the subject of policies. Those policies may be documented in the service description.

312. Of course, willingness on the part of service providers and consumers to interact is not the same as a willingness to perform requested actions. A service provider that rejects all attempts to cause it to perform some action may still be fully willing and engaged in interacting with the consumer.

C.2.8.3. Reachability

313. A service consumer may have the intention of interacting with a service, and may even have all the information needed to communicate with it. However, if the service is not reach-

able, for example if there is not communication path between the consumer and provider, then, effectively, the service is not visible to the consumer.

314. Reachability is the relationship between service participants where they are able to exchange information as part of service interactions. Reachability is closely connected to the concept of execution context – an important requirement for an execution context is to establish that service participants can communicate with each other.

C.2.9. SOA Attributes

SOA Attribute	Feature	Year 2	Year 3	Year 4	Year 5	Year 6
Service Applications	Enterprise Management	-	-	0%	15%	30%
	Discovery	-	-	-	0%	20%
	Messaging	-	-	-	-	0%
	Mediation	-	-	-	-	-
	Collaboration	-	-	-	-	-
	Security	-	-	0%	15%	30%
	Storage	-	-	-	-	-
	Application	-	-	-	-	-
	Assistance	-	-	-	-	-
Service Catalogue	Description Language	0%	20%	40%	60%	80%
Service Instance	Definition Language	0%	20%	40%	60%	80%
Service Publications	Unicast	-	-	0%	25%	50%
	Multicast	-	-	0%	25%	50%
	Broadcast	-	-	0%	25%	50%
	Anycast	-	-	-	-	0%
Service Discovery	Registry	0%	20%	40%	60%	80%
	Directory	-	0%	30%	60%	90%
Service Data Model		-	-	-	-	0%
Service Contract	Low Level	-	-	-	-	-
	High Level	-	-	-	-	-

Table C.1. Implementation of SOA Attributes

315. The previous table, Table C.1, is a suggested generalized outline on when SOA attributes should be implemented within the mid-term time frame. A value of zero percent indicates that implementation should begin in that year. Conversely, a value of a hundred percent indicates that implementation should end in that year. Naturally, measurable metrics to quantify progress would have to be agreed upon by the NATO nations.

SOA Attribute	Feature	Year 6	Year 7	Year 8	Year 9	Year 10
Service Applications	Enterprise Management					
	Discovery					
	Messaging					
	Mediation					
	Collaboration					
	Security					
	Storage					
	Application					
	Assistance					
Service Descriptions	Description Language					
Service Definitions	Definition Language					
Service Publications	Unicast					
	Multicast					
	Broadcast					
	Anycast					
Service Discovery	Repository					
	Directory					
Service Data Model						
Service Contract	Low Level					
	High Level					

Table C.2. ***Far-term SOA Attributes***

C.2.9.1. Service Application

316. A service is a contractually defined behaviour that can be implemented and provided by a service provider of choice for use by service consumer. The term “services” does not imply web

services; although, web services are well known implementations of SOA. Other specialized implementations include J2EE[2] and .NET[3].

317. The following service group is called the NATO Information Infrastructure Core Enterprise Services (NII-CES). NII-CES services are available enterprise-wide and are independent of Cross COI Services. They are considered the building blocks upon which Cross COI Services are created.

318. Capability Areas

319. Service Management Control

- Enterprise Management
- Application
- Assistance

320. Information Assurance

- Security

321. Information and Integration

- Discovery
- Mediation
- Storage

322. Communication

- Messaging
- Collaboration

323. Community of Interests

324. Users and Missions

C.2.9.2. Service Catalogue

325. The service catalogue information consists of the constraints and policies that define the usage context and functionality of the service. This enables service consumers, human or another service, to examine the service description and evaluate the following questions:

- What does the service do?

[2] Sun Microsystems® Java 2® Platform, Enterprise Edition (J2EE) defines the standard for developing component-based, multi-tier, enterprise applications. http://java.sun.com/j2ee/

[3] Microsoft® .NET is a set of Microsoft software technologies for connecting information, people, systems, and devices. www.microsoft.com/net/

- Is it relevant to my mission?

326. The declaration may also include details about any implied process or other legal or business terms that occur when the service is invoked. For example, if a service consumer invokes a service that places a purchase order to the service provider, and the execution is successful, it may result in a financial responsibility to the service provider or some other legal entity.

C.2.9.3. Service Type

327. Service type information includes all information that is needed to know how to use or how to produce a service of a specific type. It includes information on service interfaces, protocols and achievements.

C.2.9.4. Service Instance

328. While the nature of the services themselves may vary, a common standard for declaring a service is desirable when building an infrastructure. The service instance consists of the technical parameters, constraints and policies that define the terms to invoke the service instance. Every service should include a service definition in a standardized format. This enables service consumers, human or another service, to examine the service definition and evaluate the following questions:

- How can I bind to a service?

- What security protocols (if any) must be used with it?

C.2.9.5. Service Publication

329. A service must communicate its service description in an accessible manner to potential consumers. It does so by using one of several advertising method; catalogue or webpage publishing, Pulling, or Pushing.

330. In the Pull methodology, potential service consumers request the service provider to send them the service description. This pull methodology may be invoked as a service itself.

331. In the Push methodology, the service provider, or its agent, sends the service description to potential service consumers. The push and pull methodologies may work together to facilitate advertising services through a third party in a pattern of publish and subscribe.

332. Different models for the push methodology include:

- Unicast (point-to-point) is a methodology where the service provider sends a message from a single source to a single destination.

- Multicast (point-to-multipoint) is a parallel communication pattern in which a source host sends a message to a group of destination hosts. This is different from sending multiple serial unicast (point-to-point) messages to each of the destination hosts.

- Broadcast (point-to-all points) is a methodology where the service provider sends a transmission to all message consumers on a fabric.
- Anycast (point-to-point-to-multipoint) is a methodology that assigns a private address to several message consumers on a fabric. The message sender does not know or care who consumes the message or the details of the message's distribution list.

C.2.9.6. Service Discovery

333. Discovery occurs when a potential consumer obtains information about the existence of a service, its applicable parameters and terms. Discovery does not constitute authorization to execute against the service; although these details may be included in the discovery pattern.

334. Service discovery may include the following steps:

- Finding and selecting achievements that suits the need of the service consumer
- Finding and selecting service types that produces the selected achievements
- Finding and selecting service instances of the selected service types

335. Finding and using services based on type can be easily automated, while finding and using services based on description or ontology is still difficult to automate.

C.2.9.6.1. Implementing advertising and discovery

336. The publishing and discovery components of SOA may be implemented in many ways, including using a registry/repository or a services directory. Although using these may make discovery easier, an SOA requires neither of them.

C.2.9.6.2. Registry/Repository

337. A Registry/Repository is a component where users can store and manage artefacts required for their enterprise to function. This includes artefacts that require sharing among more than one user (such as XML schemas and web-service descriptions). The repository component provides an intrinsic storage mechanism that is bound to the registry such that the registry knows about any auditable events to the artefacts in the repository.

C.2.9.6.3. Services Directory

338. A Directory is an interface that provides information to bind to artefacts. Those who own or control the artefacts may make an entry into the directory to reference the artefact and explain how to bind to it. Others may retrieve this information and use it to bind to the artefacts. The main drawback of the directory is that it has no control or notification if an artefact is deleted, changed or replaced, and the directory may not be able to indicate these events to users.

339. Both Registry/Repository and Directory implementations allow for federation (also called replication). This functionality allows content from one implementation to be replicated or referenced from within other implementations.

340. Several standards exist to constrain Registry/Repository and Directory implementations. The most prevalent standards are ISO/IEC 11179 Part 315 (an ISO standard for metadata registries), the OASIS ebXML Registry-Repository Technical Specifications16 and the OASIS Universal Description and Discovery Interface (UDDI) Technical Specification

341. The OASIS UDDI specification is available from the OASIS website under the technical committee's home page area at www.oasis-open.org/committees/tc_home.php?wg_abbrev=uddi-spec.

342. Standards such as Bluetooth TM use a broadcast-type methodology to advertise their services to other Bluetooth devices that are within range.

C.2.9.7. Service Data Model

343. When invoking a service, certain parameters may be required to help the service fulfil the service request. The service may also pass parameters back to the service consumer. To understand any required parameters serialization, an artefact is required to specify the associated data models for the services. Even if no parameters are used, SOA requires a base component to declare this in a format that is understandable to service consumers.

C.2.9.7.1. Known implementations

344. The W3C's WSDL10 can be used to express that related schema fragments constrain XML instance data being passed in and out of services. ebXML's Collaboration Protocol Profile11 also references a schema for instance data being used in a service or collaboration of services. Both of these implementations are not specifically dependent on the W3C's XML Schema19 format; yet most implementations use it.

C.2.9.8. Service Contract

345. A service contract is not an actual legal document. Instead, the service contract specifies a set of technical data, and possibly business information. The contract is between whoever is providing the Web service, and whoever is consuming the Web service. Put in the simplest terms, the contract provides details about the service being offered by the provider. By agreeing to a contract, both sides understand exactly what will be provided.

C.2.9.8.1. Low Level

346. A service contract can operate at a lot of different levels. A low-level contract expresses how you communicate, and what the expectations of communications are. But that low-level contract is not nearly as important as the higher-level contracts. At a minimum, will contain:

- Service description
- Service interface details
- Service inputs/outputs

C.2.9.8.2. High Level

347. This higher-level contract is far more important because these contracts frequently have business implications, not just technical implications. For example, a contract may include details of how the service will be authenticated, and so have details about authentication, encryption and authorization. A typical high-level contract could include the following:

- Description of service
- Details interface to the service
- Service inputs/outputs
- How service will be authenticated
- How service will be authorized
- What type of encryption will be used
- Service level agreements (Availability, response time, etc)
- Charges to use the service

C.3. NETWORK-ENABLED OPERATIONS (NEO-RM)

348. A NATO Network Enabled Operations Reference Model (NEO-RM) should be created during the mid-term time frame. The best course of action to create such a document would be build upon similar existing doctrine used at the national level by some NATO members.

349. The NEO RM, describes all of the activities required to establish, use, operate, and manage the network enabled enterprise information environment to include: the generic user-interface, the intelligent-assistant capabilities, the network enabled capabilities (i.e., core services, Community of Interest services, and environment control services), and the enterprise management components. It also describes a selected set of key standards that will be needed as the NEO capabilities of a seamless communications network backbone are realized.

350. The NEO-RM is to describe the evolving NATO enterprise aspects of an objective net-enabled information environment. The NEO-RM, as designed, serves as a common, enterprise-level, reference model for NATO's net-enabled operations, such as the expeditionary NATO Response Force (NRF), and for current and future acquisition programs to reference. A shared vision of the enterprise information environment will assist decision makers promote enterprise-wide unity of effort.

Figure C.3. NATO Network-Enabled Operations: The Future

C.3.1. NEO Attributes

NCOW Attribute	Features	Year 2	Year 3	Year 4	Year 5	Year 6
Internet Protocol	Automated Configuration	-	-	0%	25%	50%
	Large Address Space	-	-	0%	25%	50%
Secure & Available Communications	Hardened Against Denial of Service	-	0%	10%	20%	30%
	Core Network Encryption	-	-	0%	20%	40%
	Edge to Edge Encryption	-	-	-	0%	25%
Only Handle Information Once (OHIO)	Use Known Repositories	-	0%	25%	50%	75%
Post in Parallel	Post Data Before Processing	-	-	0%	20%	40%
Smart Pull		-	-	0%	20%	40%

Service Agreements	Service Contract	-	-	-	-	-
	Quality of Service	-	-	-	-	-
Data Centric	Metadata	-	-	0%	30%	60%
	Metadata Registry					
Application Diversity		-	0%	10%	20%	30%
Assured Sharing		-	0%	20%	40%	60%

Table C.3. Implementation of NEO attributes

351. The previous table, Table C.3, is a suggested generalized outline on when NEO attributes should be implemented within the mid-term time frame. A value of zero percent indicates that implementation should begin in that year. Conversely, a value of a hundred percent indicates that implementation should end in that year. Naturally, measurable metrics to quantify progress would have to be agreed upon by the NATO nations.

C.3.1.1. Internet Protocol (IP)

352. The Internet Protocol (IP) is a protocol used by source and destination hosts for communicating data across a packet-switched inter-network. Data in an IP inter-network are sent in blocks referred to as packets. No setup of "path" is needed before a host tries to send packets to a host it has previously not communicated with.

353. The Internet Protocol provides an unreliable datagram service (also called best effort); i.e. It makes almost no guarantees about the packet. The packet may arrive damaged, it may be out of order (compared to other packets sent between the same hosts), it may be duplicated, or it may be dropped entirely. If an application needs reliability, it is provided by other means, typically by upper level protocols transported on top of IP.

354. Inter-network routers, forward IP packets across interconnected layer 2 networks. The lack of any delivery guarantees means that the design of packet switches is made much simpler. (Note that if the network does drop, reorder or otherwise damage a lot of packets, the performance seen by the user will be poor, so most network elements do try hard to not do these things - hence the best effort term. However, an occasional error will produce no noticeable effect.)

355. In a network-enable NATO, there is a need for support for an unlimited number of site addresses for wireless communications devices, remote sensors, vehicles and precision-guided munitions. Therefore, with any large-scale operation, the current finite number of IP addresses becomes a resource that must be managed and closely monitored.

356. The ad-hoc nature of future NATO operations dictates the need for easily configurable networks. Since most of the devices connected to NATO networks will be mobile devices, a

device must be able to arbitrarily change locations on the Internet and still maintain existing connections.

C.3.1.2. Secure and Available Communications

357. Security requires that systems and users are protected against attack, disruptions, and threats. Data must also be kept private and free of malicious or corrupted content as it travels throughout the enterprise. And the network infrastructure itself must be protected against exposure to attacks that impacts internal servers and end user systems. In the mid-term time frame the initial goal is encryption for the core network, and the final goal is edge-to-edge encryption with a network that is hardened against denial of service.

358. Ensuring availability means that systems themselves are always available, and that information is readily accessible to users and other authorized individuals, especially as it relates to regulatory compliance and legal discovery. Migrating older, yet still useful, data poses an added challenge to availability, as users still demand immediate access and IT requires high degrees of ease and automation to achieve this.

C.3.1.3. Only Handle Information Once (OHIO)

359. A key feature of network-enabled NATO is the ability to provide individual soldiers and commanders with relevant timely information. But pushing information to users in the area of operation is difficult because the number of items that can link to the network exceeds the current messaging protocol’s ability to assign addresses. Technologies that allow wireless systems to plug into tactical and theatre networks seamlessly without straining resources may permit the military to deploy more network-enabled devices.

360. This capability fits into a concept known as “only handle information once” (OHIO), where an information producer posts data once but permits authorized users to access it. This approach differs from requiring the producer to know the address of every user that may want the information.

C.3.1.4. Post in Parallel

361. All NATO participants or business process owners make their data available on the network as soon as it is created. Posting data does not mean just making accessible, it also means tagging (describing) the data appropriately for its content. Processing of data will be done as needed by specialized web services invoked by the data consumer.

C.3.1.5. Smart Pull

362. The two solutions in use for data synchronization are called “Push” and “Pull”. Smart Pull is just a further refinement of the Pull solution.

- Push solutions involve the server notifying the device that data is available. Solutions to this usually require some type of infrastructure to manage the distribution of notifications to the network enabled-devices. For example, the solution might involve using the mobile phone service provider’s SMS system or might be a custom built system like that created by Research in Motion (RIM) for their Blackberry communications network.

- Pull solutions put the burden on the network-enabled device to go retrieve data. They’ve historically been simple implementations using techniques like calling the server on some regular time interval or maybe even relying on the user to initiate the data sync.

363. Generally the immediate notification of push is desirable, however when building our own applications, the simplicity of pull is more attractive. This is where Smart Pull comes in. An example of the concept of smart pull that has been gained a lot of attention since the announcement is the Microsoft Messaging and Security Feature Pack (MMSF). MMSF provides Windows Mobile 5.0 Smartphone with full connectivity to their Exchange server, keeping Outlook data up-to-date including receiving email as soon as it arrives at the server. The technique used to maintain synchronization is basically a long-running web service call.

364. The smart pull mechanism seems like the kind of solution that will address many common scenarios faced when developing mobile applications that require close synchronization with a server. Applications encourage discovery; users can pull data directly from the network or use value-added discovery services.

C.3.1.6. Data Centric Approach

365. In a data-centric approach the data separate from applications, or services. Communications between services occurs by posting data. The steps involved in a data-centric approach are:

- Analyze how the data is used and moved as it flows through the system to better understand how to store and use it.

- Describe data or tag data to facilitate a system level view. This involves describing data **and** describing the context under which it was collected or generated.

- Build services around data. Services are just the methods to manage data.

C.3.1.7. Application Diversity

366. Users can pull multiple services to access same data (e.g., for collaboration). This idea reinforces the underlying concept of the data being independent of the services that manipulate the data.

C.3.1.8. Assured Sharing

367. Assured sharing means trusted accessibility to net resources, such as: data, services, apps, people, collaborative environment, etc. Access is assured for authorized users, but denied for unauthorized users by maintaining thorough security policies.

C.3.1.9. Quality of Service

368. Data timeliness, accuracy, completeness, integrity, and ease of use.

C.4. INFORMATION SYSTEMS INTEROPERABILITY REFERENCE MODEL (ISIOP-RM)

369. A successful case of interoperability is demonstrated in NATO's efforts to standardize small arms. It was politically unattainable, for domestic industrial base reasons, to standardize on a common rifle and pistol for all NATO countries. But NATO did standardize on ammunition; the standard pistol round is 9mm. The interoperable ammunition solution gained interoperability -- the real need -- without imposing unpalatable commonality requirements. Such interoperability can also be achieved with information systems.

C.4.1. Interoperability Attributes

Interoperability Attribute	Year 2	Year 3	Year 4	Year 5	Year 6
Inter-Network					
Modularity					
Data					
Shared processes					
Interoperable procedures					
Cognitive interoperability					
Doctrinal interoperability					

Table C.4.

C.4.1.1. Inter-Network

370. Communications interoperability can be defined by the ability to inter-network. Because of the confederated nature of NATO, heterogeneous communications networks are necessary, and unavoidable; the key to interoperability is that they can be concatenated together using routers. Each discrete communications network is a routable network.

C.4.1.2. Modularity

371. Information systems are made up of sense, decide and act functions that rest upon the communications. This layer of the Model deals with the modularity of these functions and the complexity of information systems, which can be explained by nesting and chaining.

C.4.1.3. Data

372. Data element interoperability is a clear requisite to information system interoperability. This is the abstract layer where this discussion of data, meta-data and meta-meta-data belongs. A key issue is semantic equivalences.

C.4.1.4. Shared Processes

373. This is a software engineering concept. At its trivial level reusable code obviously enhances interoperability but that is a side effect of what is essentially an economy effort in code production. At a more mature level, mobile code and portable code are the pertinent issues.

C.4.1.5. Interoperable Procedures

374. Operating procedures is the level at which we tend to shift from systems engineering to human factors in the layered Model. This is the domain long inhabited by Standard Operating Procedures.

C.4.1.6. Cognitive Interoperability

375. Cognitive Interoperability has to do with shared situation awareness. Information systems are interoperable at this layer if decision makers in two different systems are seeing coherent pictures of the information presented.

C.4.1.7. Doctrinal Interoperability

376. This is a human factor that leads to coherency and uniformity of action. Different decision makers, when presented with the same information will be making similar decisions. The usual doctrinal tensions of uniformity versus creativity are still present and certainly not resolved by this Model. The Model only serves to illustrate the level of abstraction where such discussion belongs.

377. Where the reference models, described in the previous section, laid out the framework for a networked-enabled NATO, it is the technologies that are projected to be available during the mid-term period are going to enable the implementation of a network-enabled NATO, during this time frame.

C.5. BUSINESS PROCESS INFRASTRUCTURE FRAMEWORK (BPIF)

378. Within the wide area of relevant information technologies, this work has focused on a particular field referred to as Business Process Infrastructure Framework (BPIF). Although highly relevant, this is by no means a well defined or even well established field. Instead, an important part of this work has been to find a working definition and scope for the field.

379. As a starting point, the intent has been to describe technologies that in various ways support business processes. As almost any information technology available can be said to do this, we have focused on those technologies that most explicitly have to do with this kind of support. Furthermore, as service orientation is a known fundamental architectural concept for future

NNEC solutions, the definition of BPIF assumed here has been limited to that kind of solutions. From the outset, we have also seen technologies for service description and discovery, which also is part of the Swedish responsibility, as such a fundamental ingredient of BPIF, that it is included as a subfield within BPIF.

380. An important basis for this work has also been the FMA Technology Support Study, June 2005, that points out and describes some central areas of technology development, in particular within the field of distributed system architecture. Parts of the material from that report have also been reused here, although somewhat rearranged to fit the current scope.

381. Within this scope, this report presents the first steps toward a complete description of the field of BPIF, and includes

- a definition of the field,
- a structure that divides the field into a number of aspects,
- a brief overview of each of these aspects,
- a list of candidate technologies (with an emphasis of existing or emerging standards) with a bearing on BPIF, and
- A mapping of the candidate technologies onto the BPIF subfields.

382. Currently, the terms used in this report are those traditionally used in the general development of the various technologies that are presented. However, in the more specific context where the results of this report are to be applied, an adaptation of some terms to that context will probably be needed.

C.5.1. Overview

383. This section consists of two main parts. The first part is a definition of the field of business process infrastructure frameworks (BPIF), which divides the field into four main aspects. These aspects are described one by one. The descriptions both introduce some central concepts, and provide an overview of technology fields that support the aspects.

384. Many relevant technologies do not fit into a single of the aspects or technology fields addressed in the first part. The second part of the document is therefore a listing of potentially relevant technologies, along with brief descriptions of the technologies and which aspects of BPIF they support.

C.5.2. Definition

385. At operational level, business processes are key components for the realization of an organization’s tasks, in particular those tasks that represent the main and externally exposed

capabilities of the organization. A business process is a realization of a work flow that may involve several subtasks and decision points. Often, a business process also involves the collaboration of several organizations where each organization will contribute in parts of the operations. Setting up and running a business process is thus an important and often complex task in any operation.

386. Although business processes have traditionally been treated mostly in a commercial context, they are also relevant as means of organizing operations in a context like NNEC. However, to successfully make use of the concept of business processes, in particular in a very dynamic context, a framework for engineering and management of business processes is needed, i.e., a business process infrastructure framework (BPIF).

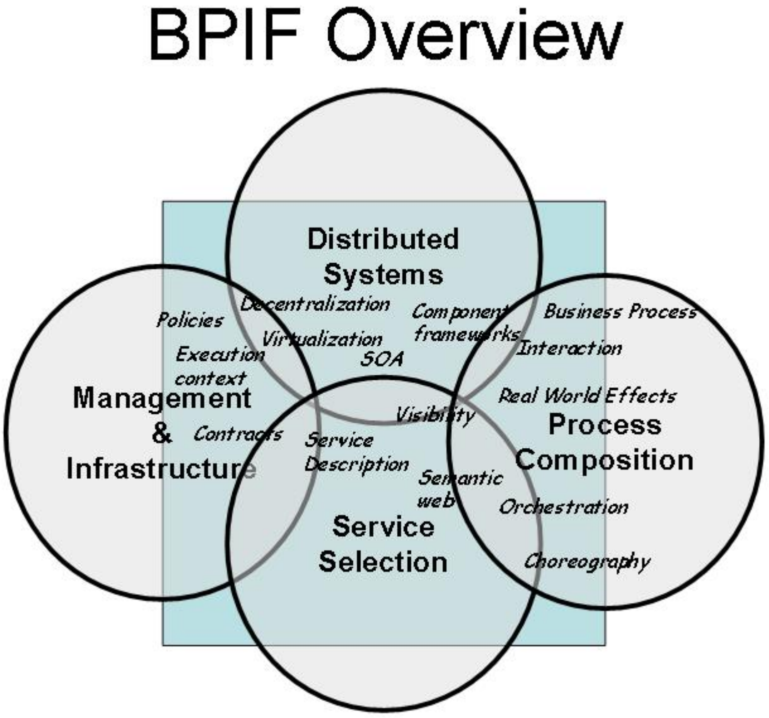
387. Today, there are technologies that provide at least partial support for a BPIF. Most of those technologies that provide the most explicit support for business processes are based on the concepts of services and service oriented architecture (SOA), and we will in fact limit our definition of BPIF in this context to include only service based solutions. However, few if any complete frameworks for service based business processes exist that provide the consistent technology reference model needed here. Therefore, we will address this field in terms of four aspects or concerns:

388. Distributed systems: Many practical business processes involve the collaboration of several organizations and systems. Therefore, a BPIF needs to be based on distributed system technologies. With service based solutions in focus, service oriented architecture (SOA) is an obvious field of distributed system technologies to consider, but there are also other trends in distributed system technologies that may provide essential support for the implementation of a BPIF.

389. Service selection: With services as key components for the realization of a business process, it is of course important that relevant services can be found. Normally, this is not just a matter of finding services that provide the right capabilities, but also to make sure that other requirements and constraints are fulfilled regarding, for example, quality of service (QoS) and policies. Specific concerns for this aspect are thus both the description of services, and means of finding and identifying the right services.

390. Process composition: Once suitable services are available, they need to be combined into sequences and decision points that represent the required work flow of the business process. Also it is necessary to make sure that service providers and consumers in the work flow interact properly in accordance with the intentions behind each service. Support for this aspect must include the description of business processes in these terms so that their setup and deployment can be made more or less automatic.

391. Management and infrastructure. This aspect concerns deployment, configuration, contracts, and policies for business processes. Typically, business processes will need to involve components that are distributed over heterogeneous environments and different organizations. This stresses the need for technologies that can provide a manageable infrastructure and execution context.



This is a simplified overview of the Business Process Infrastructure Framework. BPIF is indicated by the central box. The overlap between concepts and aspects should be noted, as well as the fact that there Are concerns in each aspect that is not specific to BPIF (in particular within distributed systems).

Figure C.4. ***BPIF Overview***

392. Each of these aspects can also be treated as a technology area where technologies already exist or development of new technologies is going on. However, much development in each area has a scope or orientation that is beyond the particular needs for a BPIF. Therefore our treatment of each area will not cover its full extent, but rather be limited to parts that are related to BPIF needs.

C.6. NNEC PROFILE

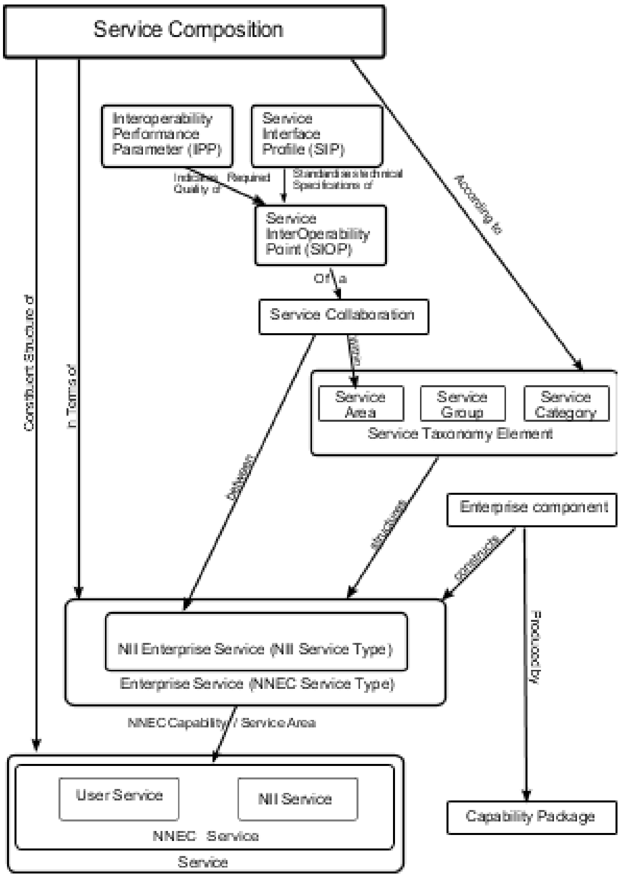

Service Composition
Interoperability Performance Parameter (IPP)
Service Interface Profile (SIP)
Indicates Required Quality of
Standardises technical Specifications of
According to
Service InterOperability Point (SIOP)
Of a
Service Collaboration
Constituent Structure of
In Terms of
Service Area
Service Group
Service Category
Service Taxonomy Element
Between
structures
Enterprise component
constructs
Procured by
NII Enterprise Service (NII Service Type)
Enterprise Service (NNEC Service Type)
NNEC Capability / Service Area
User Service
NII Service
NNEC Service
Service
Capability Package

Figure C.5. NNEC Profile

393. The items in the figure above are:

- Enterprise service synonym for NNEC Service Type
- ENTERPRICE SERVICE is a type of NNEC SERVICE classified by the capability (service) area in which it is delivering its services
- SERVICE CATEGORY is a TAXONOMY ELEMENT that structures ENTERPRISE SERVICES in a group with the same objectives
- SERVICE GROUP is a TAXONOMY ELEMENT that structures ENTERPRISE SERVICES in a group with the main objectives
- SERVICE LAYER is a TAXONOMY ELEMENT that structures ENTERPRISE SERVICES in a group of the same family
- SERVICE AREA is a TAXONOMY ELEMENT that structures ENTERPRISE SERVICES in a group of the same nature

- SERVICE COMPOSITION is the orchestration of a NNEC service in terms of Enterprise service
- ENTERPRISE service is a construction element of a ENTERPRISE SERVICE realizing a CIS part of a CAPABILITY PACKAGE
- CAPABILITY PACKAGE is a combination of national (military and civilian) and NATO funded infrastructure, operations, and maintenance, manpower, and associated costs that, with the military forces and other essential requirements, enable a NATO commander to achieve a specific military required capability.
- SIOP (Service Inter Operability Point) is a linking point that ensures that ENTERPRISE SERVICES of different making but with the same SIP can interact with each other
- SIP (Service Interface Profile) is a description of requirements of A SIOP that give guidance to engineers in realizing ENTERPRISE COMPONENTS
- IPP (Interoperability Performance Paramter) Is an indicator of the required quality of service of a SIOP according to the CM

C.7. TRANSITION FROM PLATFORM ORIENTED TO SERVICE ORIENTED MODELS

394. Information technology is undergoing a fundamental shift from platform-centric computing to network-centric computing. Platform-centric computing emerged with the widespread proliferation of personal computers and the global business environment. These factors and related technologies have created the conditions for the emergence of network-centric computing. This shift from platform to network is what enables the more flexible and more dynamic network-centric operation. The shift from viewing partners as independent to viewing partners as part of a continuously adapting ecosystem fosters a rich information sharing environment.

395. This shift is most obvious in the explosive growth of the internet, intranets, and extranets. Internet users no doubt will recognize transmission control protocol/internet protocol (TCP/IP), hypertext transfer protocol (HTTP), hypertext markup language (HTML), Web browsers (such as Netscape Navigator, and Microsoft's Internet Explorer), search engines, and JavaTM Computing. These technologies, combined with high-volume, high-speed data access (enabled by the low-cost laser) and technologies for high-speed data networking (hubs and routers) have led to the emergence of network-centric computing. Information "content" now can be created, distributed, and easily exploited across the extremely heterogeneous global computing environment. The "power" or "payoff" of network-centric computing comes from information-intensive interactions between very large numbers of heterogeneous computational nodes in the network, where the network becomes the dynamic information grid established by interconnecting partners participating in a collaborative, coalition environment. At the structural level, network-centric warfare requires an operational architecture to enable the common processes to be shared by all parties.

396. The fabric for enabling net-centric operations is Service-Oriented Architectures (SOA). SOA is enterprise architecture that leverages heterogeneity, and thus inherently platform-neutral. It is focused on the composition of Services into flexible processes and is more concerned with the Service interface and above (including composition metadata, security policy, and dynamic binding information), more so than what sits beneath the abstraction of the Service interface. SOA requires a different kind of platform, because runtime execution has different meanings within SOA. SOA is to enable business users and business process architects to compose Services into processes, and then manage and evolve those processes, in a declarative fashion. Runtime execution of such processes is therefore a metadata-centric operation of a different kind of platform --a Service-oriented composite application platform.

397. Network-centric warfare is characterized by new concepts of speed of command and self-synchronization.

398. The most important SOA within an enterprise is the one that links all its systems. Existing platforms can be wrapped or extended in order to participate in a wider SOA environment. NATO use of the NISP will provide a template for new systems development, as well as assist in defining the path for existing systems to migrate towards net-centric operations.

This page is intentionally left blank

D. ENTERPRISE SERVICE BUS (ESB) PROFILE IN THE SERVICE ORIENTED ARCHITECTURE (SOA) CONTEXT

D.1. INTRODUCTION

399. The aim of the document is to give at first an overview about Web Service Fundamentals with the focus to define after this a layer model for a common Enterprise Service Bus (ESB). Furthermore a draft proposal of an Enterprise Service Bus (ESB) / Enterprise Message System (EMS) Profile is defined based on this common ESB / EMS layer model.

400. At the end the document contains some hints related of the usage of an ESB in the military environment with a look-out on a federated ESB architecture.

D.2. REFERENCES

- [1] Open Group SOA Definition

  http://opengroup.org/projects/soa/doc.tpl?gdid=10632

- [2] OASIS SOA Reference Model

  http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm

- [3] OASIS: UDDI Version 3.0.2

  (UDDI Spec Technical Committee Draft, Dated 20041019)

  http://uddi.org/pubs/uddi_v3.htm

- [4] W3C: Web Services Description Language (WSDL)

  Version 2.0 Part 1: Core Language

  http://www.w3.org/TR/wsdl

- [5] W3C: SOAP Version 1.2

  W3C Recommendation (Second Edition) 27 April 2007

  http://www.w3.org/TR/soap/

- [6] W3C: XML Encryption Syntax and Processing

  W3C Recommendation 10 December 2002

  http://www.w3.org/TR/xmlenc-core/

- [7] W3C: XML Signature Syntax and Processing

  W3C Recommendation 10 June 2008

  http://www.w3.org/TR/xmldsig-core/

- [8] W3C: XML Key Management Specification (XKMS)

  http://www.w3.org/TR/xkms/

- [9] OASI: SAML specs and outreach info page

  http://www.oasis-open.org/committees/security

- [10] Security Assertion Markup Language

  (SAML) V2.0 Technical Overview

  http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.pdf

- [11] eXtensible Access Control Markup Language (XACML) Version 3.0

  Policy Distribution Protocol Use-cases and Requirements

  http://docs.oasis-open.org/xacml/access_control-xacml-3.0-distribution-requirements-wd-01.pdf

- [12] Basic Profile Version 1.0 (Final Material Date: 2004/04/16 19:06:16)

  http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html

- [13] Basic Profile Version 1.1 (Final Material 2004-08-24)

  http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html

- [14] Simple SOAP Binding Profile Version 1.0 (Final Material 2004-08-24)

  http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html

D.3. WEB SERVICE FUNDAMENTALS

401. This chapter gives an overview about the Web Service Fundamentals and about some Definitions.

D.3.1. SOA: Service Oriented Architecture

402. Service-oriented architecture (SOA) is a software architecture where functionality is grouped around business processes and packaged as interoperable services. SOA also describes

IT infrastructure which allows different applications to exchange data with one another as they participate in business processes. The aim is a loose coupling of services with operating systems, programming languages and other technologies which underly applications. SOA separates functions into distinct units, or services, which are made accessible over a network in order that they can be combined and reused in the production of business applications. These services communicate with each other by passing data from one service to another, or by coordinating an activity between two or more services.

403. The following figure shows an overview about the internal and external Web Service Architecture:

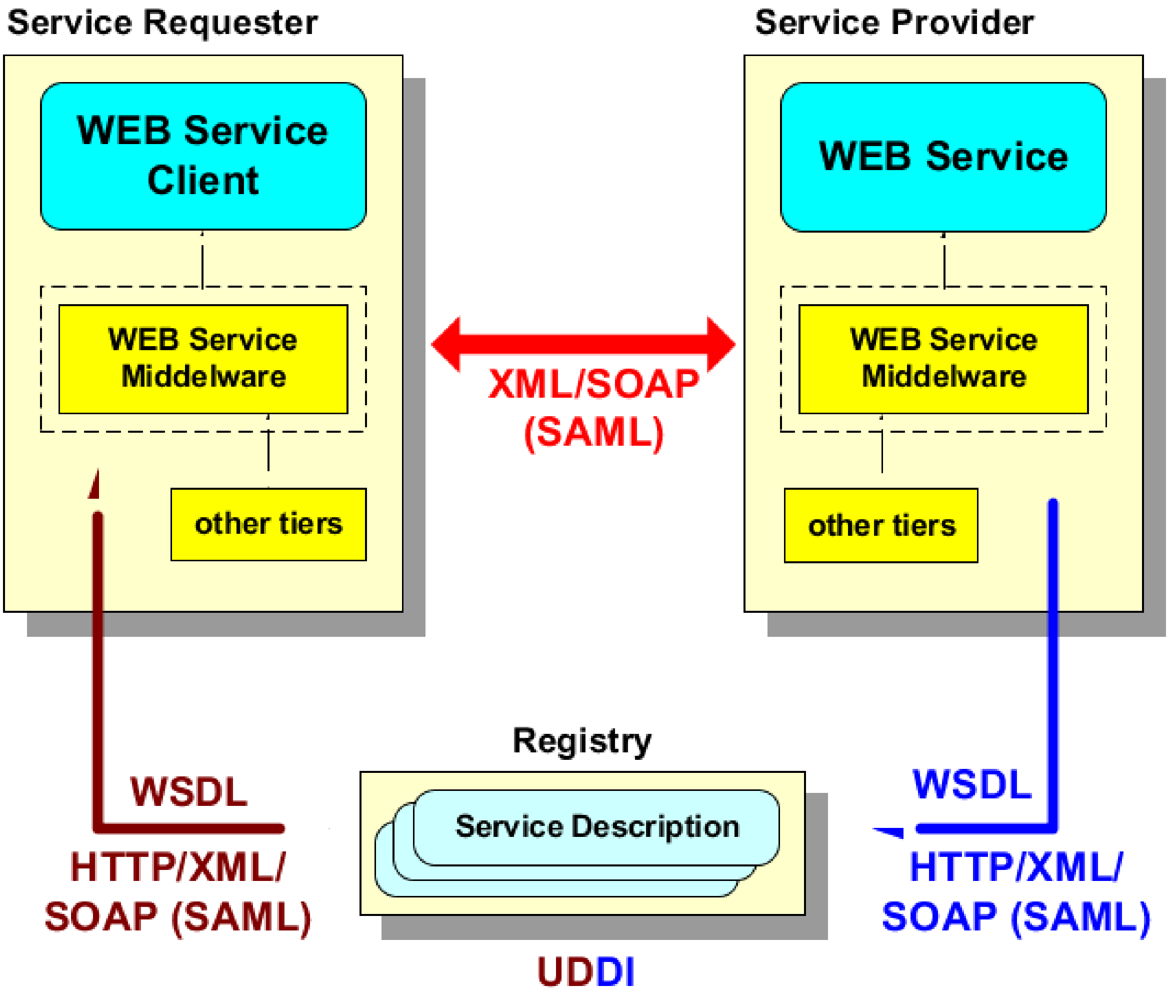


Figure D.1. Overview – Internal and External Web Service Architecture

404. There are multiple definitions of SOA. The OASIS group and the Open Group have created formal definitions which can be applied to both the technology and business domains.

- Open Group SOA Definition (SOA-Definition)

- OASIS SOA Reference Model (SOA-RM)

405. In addition, SOA is an approach to architecture whereby business services are the key organizing principles that drive the design of IT to be aligned with business needs.

406. The following figure shows the Web Service Specification in an OSI Stack Model.

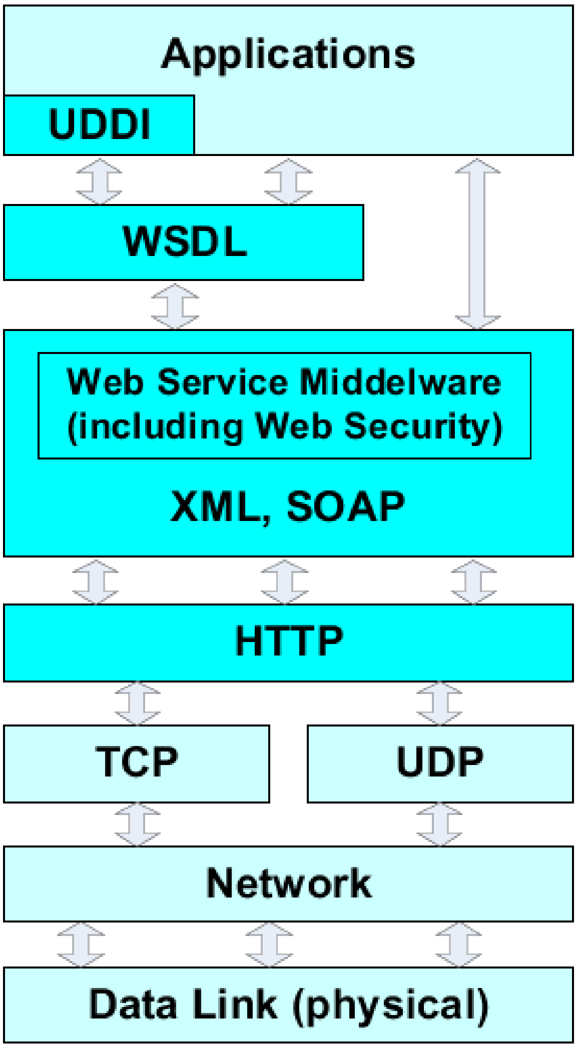


Figure D.2. Web Service Architecture – OSI Stack Model

407. On top of there are the applications (Server or Client applications). In the Web Service Architecture the UDDI (Universal Description Discovery and Integration) is signed as an application.

D.3.2. UDDI: Universal Description Discovery and Integration

408. The Universal Description, Discovery, and Integration (UDDI) protocol defines a standard method for publishing and discovering the network-based software components of a ser-

vice-oriented architecture (SOA). The standard specifies protocols for accessing a registry for Web services, methods for controlling access to the registry, and a mechanism for distributing or delegating records to other registries. In short, a UDDI registry provides a standard based approach to locate a software service, to invoke that service, and to manage metadata about that service. The following figure shows the principle of the UDDI Architecture with SOAP (messaging) for the methods of the "Inquiry and Publisher API".

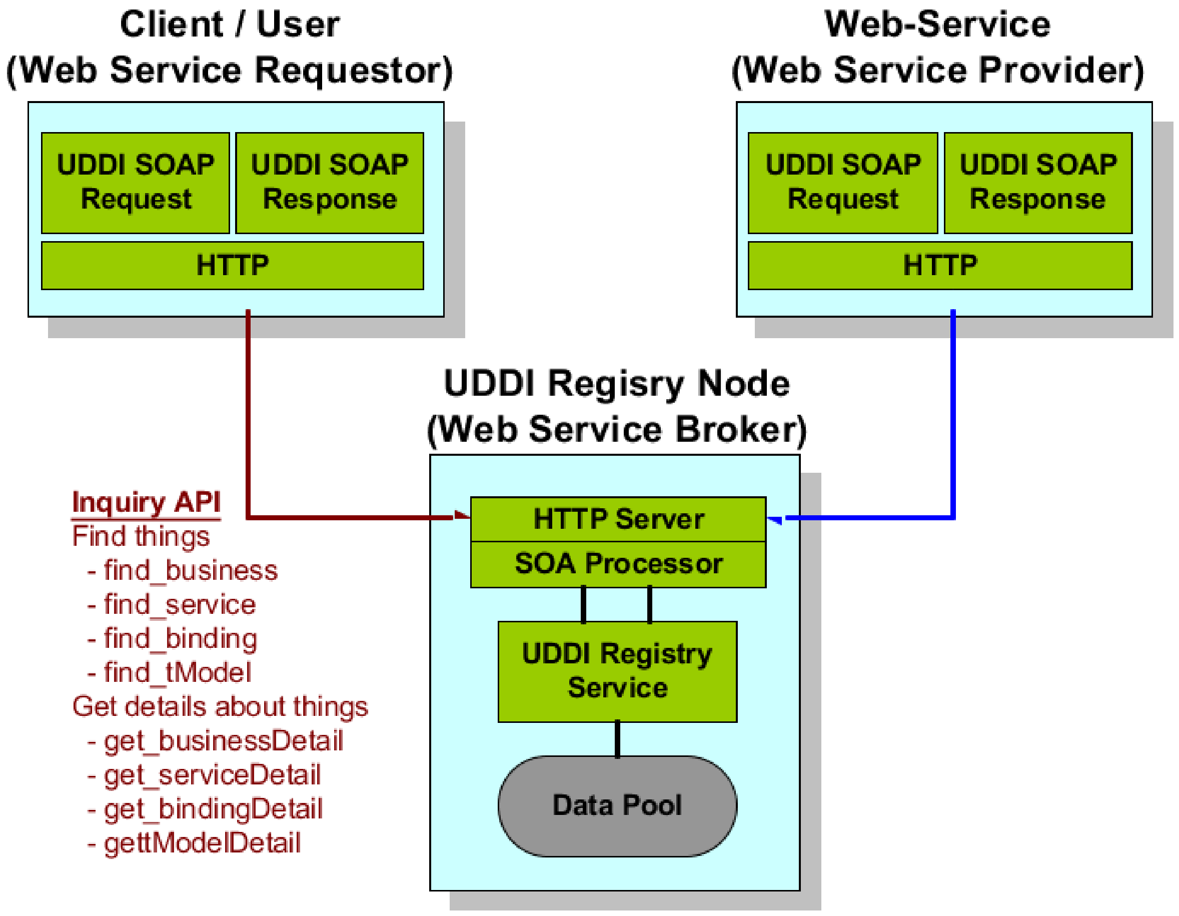


Figure D.3. UDDI Overview Architecture based on SOAP

409. The next figure shows the main extract of the UDDI Data Model (used in the "Data Pool").

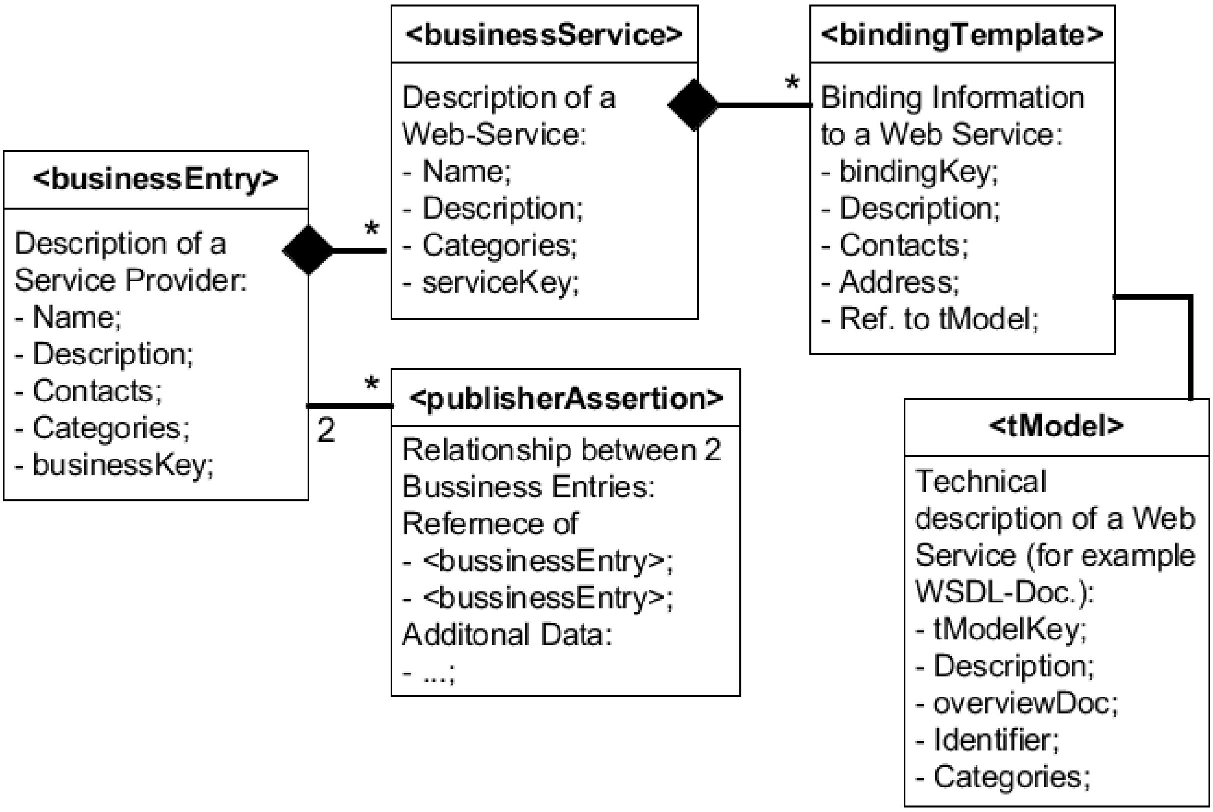


Figure D.4. Main extract of the UDDI Data Model

410. An other very important feature is to exchange UDDI Data between different UDDI Registries for example as show in the next figure. In addition, it is necessary to distinguish between internal services and external offered services.

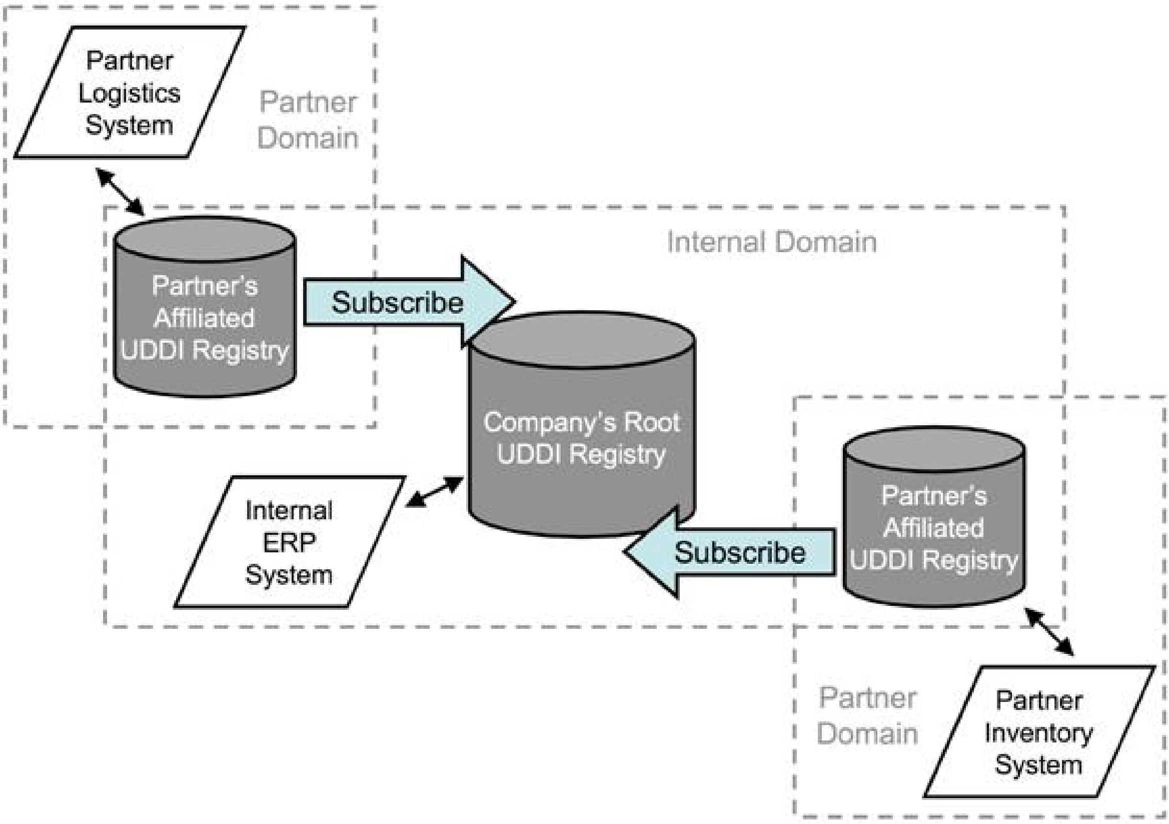


Figure D.5. UDDI illustration of Trading Partner Collaboration

D.3.3. WSDL: Web Services Description Language

411. Web Services Description Language (WSDL) is an XML-based Service / Interface Definition Language that separates function from implementation and enables design by SOA.

412. The value of WSDL is that it enables development tools and middleware for any platform and language to understand service operations and invocation mechanisms. For example, given the WSDL interface to a service that is implemented in Java, running in a HTTP-Server environment, and offering invocation through HTTP.

413. With SOAP, the WSDL specification is extensible and provides for additional aspects of service interactions to be specified, such as security and transactional.

414. The following figure shows the principle structure of a WSDL Document and its usage:

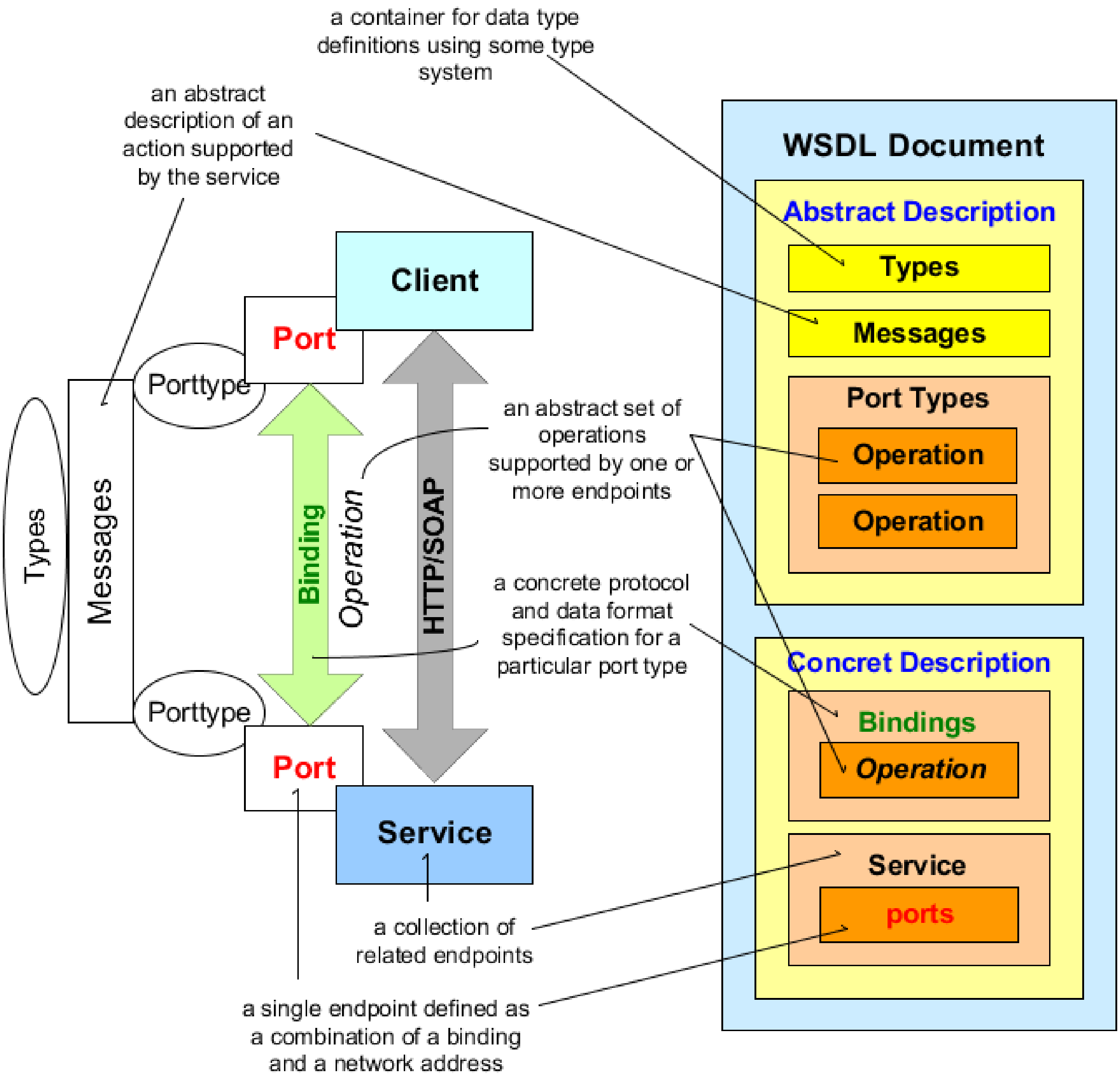

a container for data type definitions using some type system
an abstract description of an action supported by the service
WSDL Document
Abstract Description
Types
Messages
Port Types
Operation
Operation
Concret Description
Bindings
Operation
Service
ports
Client
Port
Porttype
Types
Messages
Binding
Operation
HTTP/SOAP
Porttype
Port
Service
an abstract set of operations supported by one or more endpoints
a concrete protocol and data format specification for a particular port type
a collection of related endpoints
a single endpoint defined as a combination of a binding and a network address

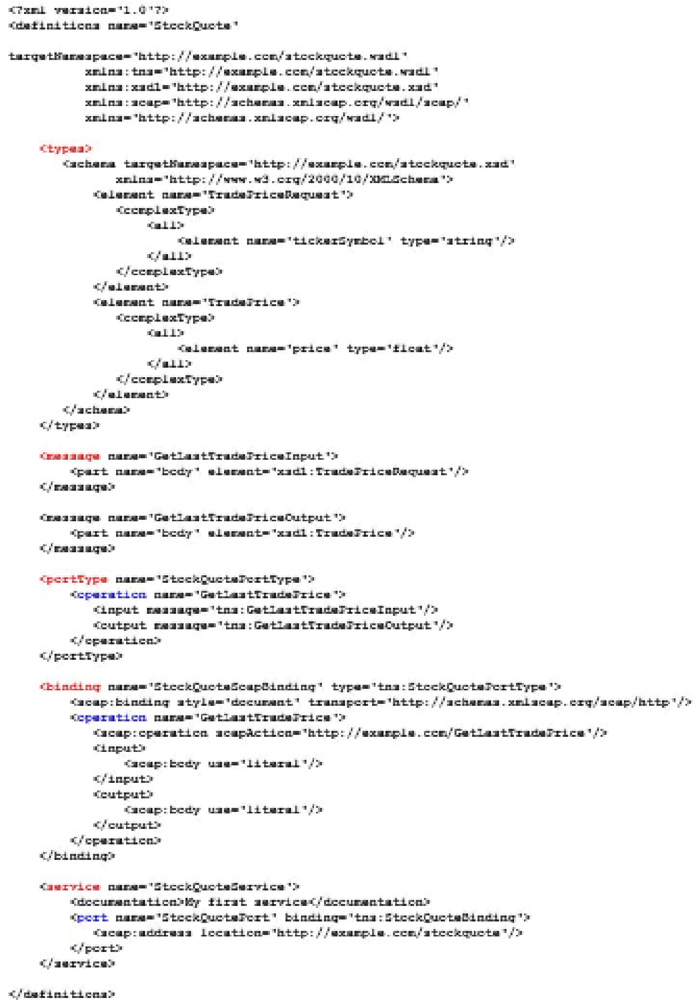
Figure D.6. WSDL Document Structure (with usage)

415. WSDL description contain:

- *Types* – a container for data type definitions using some type system (such as XSD).
- *Message* – an abstract, typed definition of the data being communicated.
- *Operation* – an abstract description of an action supported by the service.
- *Port Type* – an abstract set of operations supported by one or more endpoints.
- *Binding* – a concrete protocol and data format specification for a particular port type.

- *Port* – a single endpoint defined as a combination of a binding and a network address.
- *Service* – a collection of related endpoints.

416. The WSDL Document format is based on XML as illustrated in the example figure below.



```
<?xml version='1.0'?>
<definitions name='StockQuote'

targetNamespace='http://example.com/stockquote.wsdl'
          xmlns:tns='http://example.com/stockquote.wsdl'
          xmlns:xsdl='http://example.com/stockquote.xsd'
          xmlns:soap='http://schemas.xmlsoap.org/wsdl/soap/'
          xmlns='http://schemas.xmlsoap.org/wsdl/'>

    <types>
       <schema targetNamespace='http://example.com/stockquote.xsd'
              xmlns='http://www.w3.org/2000/10/XMLSchema'>
           <element name='TradePriceRequest'>
              <complexType>
                  <all>
                      <element name='tickerSymbol' type='string'/>
                  </all>
              </complexType>
           </element>
           <element name='TradePrice'>
              <complexType>
                  <all>
                      <element name='price' type='float'/>
                  </all>
              </complexType>
           </element>
       </schema>
    </types>

    <message name='GetLastTradePriceInput'>
        <part name='body' element='xsdl:TradePriceRequest'/>
    </message>

    <message name='GetLastTradePriceOutput'>
        <part name='body' element='xsdl:TradePrice'/>
    </message>

    <portType name='StockQuotePortType'>
        <operation name='GetLastTradePrice'>
           <input message='tns:GetLastTradePriceInput'/>
           <output message='tns:GetLastTradePriceOutput'/>
        </operation>
    </portType>

    <binding name='StockQuoteSoapBinding' type='tns:StockQuotePortType'>
        <soap:binding style='document' transport='http://schemas.xmlsoap.org/soap/http'/>
        <operation name='GetLastTradePrice'>
           <soap:operation soapAction='http://example.com/GetLastTradePrice'/>
           <input>
               <soap:body use='literal'/>
           </input>
           <output>
               <soap:body use='literal'/>
           </output>
        </operation>
    </binding>

    <service name='StockQuoteService'>
        <documentation>My first service</documentation>
        <port name='StockQuotePort' binding='tns:StockQuoteBinding'>
           <soap:address location='http://example.com/stockquote'/>
        </port>
    </service>

</definitions>
```

Figure D.7. Example of a WSDL Document Structure

D.3.4. SOAP: Simple Object Access Protocol

417. SOAP is an XML–based format for constructing messages in a transport PDU independent way and a standard on how the message should be handled. SOAP messages consist of an envelope that contains a header and a body. It also defines a mechanism for indicating and communicating problems that occurred while processing the message, which are known as SOAP faults.

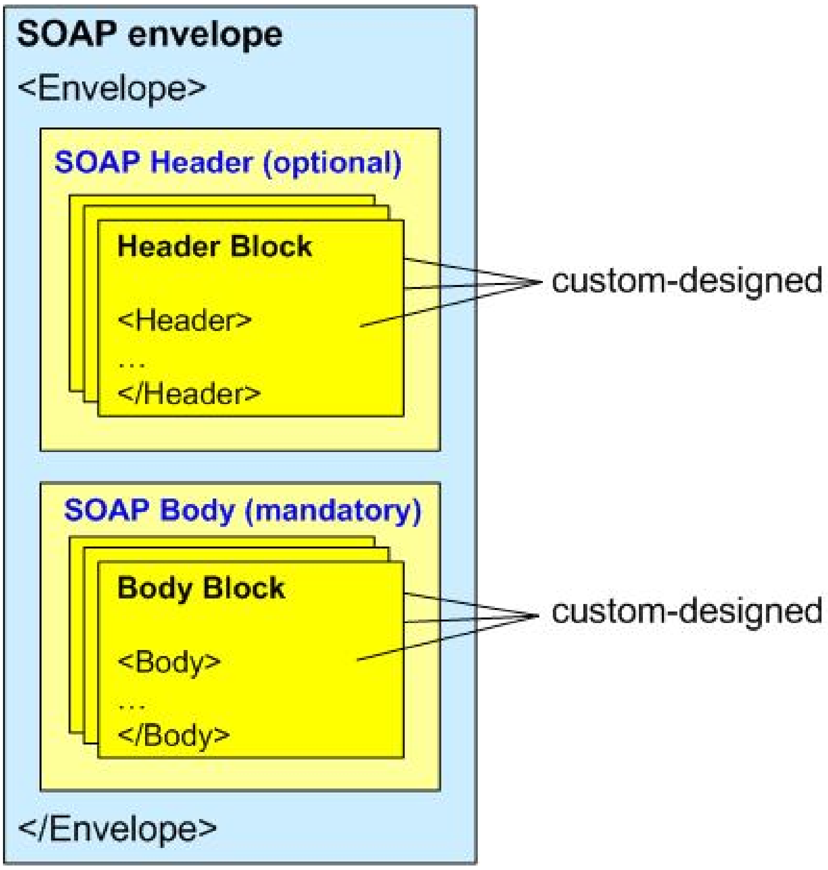
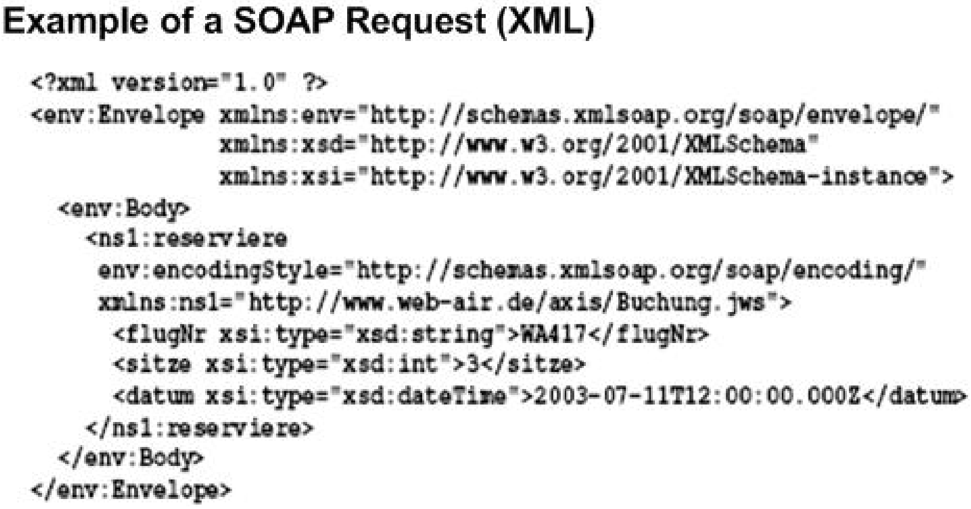


Figure D.8. SOAP Message Structure



Example of a SOAP Request (XML)

```
<?xml version="1.0" ?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
              xmlns:xsd="http://www.w3.org/2001/XMLSchema"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <env:Body>
    <ns1:reserviere
     env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
     xmlns:ns1="http://www.web-air.de/axis/Buchung.jws">
      <flugNr xsi:type="xsd:string">WA417</flugNr>
      <sitze xsi:type="xsd:int">3</sitze>
      <datum xsi:type="xsd:dateTime">2003-07-11T12:00:00.000Z</datum>
    </ns1:reserviere>
  </env:Body>
</env:Envelope>
```

Figure D.9. SOAP Message Example

418. The *headers* section of a SOAP message is extensible and can contain many different headers that are defined by different schemas. The extra headers can be used to modify the

behaviour of the middleware infrastructure. For example, the headers can include information about transactions that can be used to ensure that actions performed by the service consumer and service provider are coordinated.

419. The *body* section contains the content of the SOAP message. When used by Web services, the SOAP body contains XML – formatted data. This data is specified in the WSDL that describes the Web service.

420. *Remark:*The Header Block and the Body Block are in XML format, but contain customer designed content. SOAP doesn't define this content. For having standards it is necessary to define data models (including metadata of the content).

421. When talking about SOAP, it is common to talk about SOAP in combination with the transport protocol that is used to communicate the SOAP message. For example, SOAP that is transferred using HTTP is referred to as SOAP over HTTP or SOAP/HTTP.

422. The most common transport protocol that is used to communicate SOAP messages is HTTP. This is expected because Web services are designed to make use of Web technologies.

423. However, SOAP can also be communicated using JMS as a transport service. When using JMS, the address of the Web service is expressed in terms of a JMS connection factory and a JMS destination. Although using JMS provides a more reliable transport mechanism, it is not an open standard, requires extra and potential expensive investment, and does not interoperate.

424. SOAP is just XML and HTTP combined to send and receive messages over the Internet. It is not constrained by the application language (Java, C#, Perl) or the platform (Windows, UNIX, Mac), and this makes it much more versatile than other solutions.

425. There are many successful implementations of the basic Web services standards, particularly SOAP and WSDL but many aspects of service interaction and integration are not directly supported by basic standards, such as security, transactional, delivery assurance, and process modelling – for example.

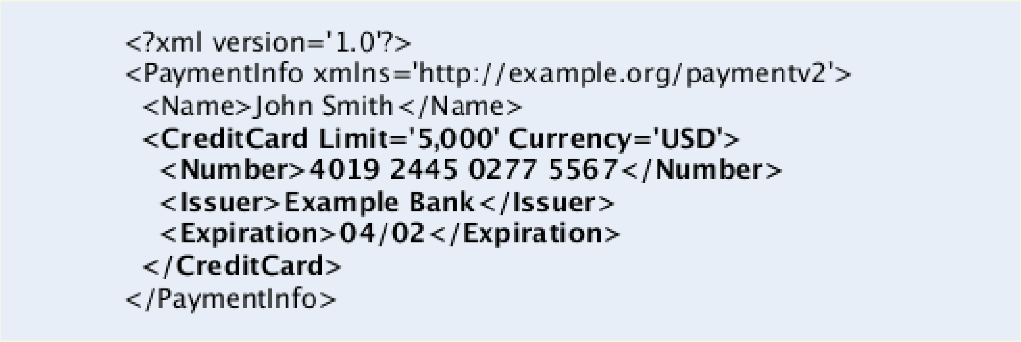
D.3.5. WEB Service Security

D.3.5.1. XML Encryption

426. XML Encryption is a W3C Standard to encrypt XML. It is done in such a way that the encrypted data remains and can be treated as XML. It uses both asymmetric and symmetric encryption algorithms, symmetric to encrypt the data and asymmetric to encrypt the symmetric session key. Both the session key and the cipher data are stored together in an XML element called EncryptedData. The EncryptedData element contains a series of child elements that describe the algorithms used during the encryption process, as well as containing key information and the cipher data.

427. The followings XML example illustrates a simple example of the XML Encryption:

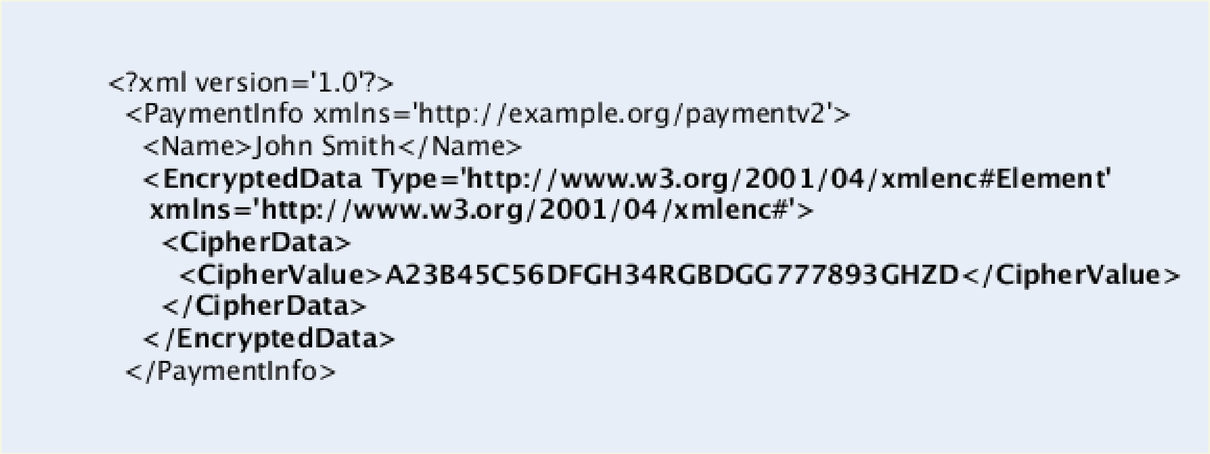
428. Consider the following fictitious payment information, which includes identification information and information appropriate to a payment method (e.g., credit card, money transfer, or electronic check):



```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
  <Name>John Smith</Name>
  <CreditCard Limit='5,000' Currency='USD'>
    <Number>4019 2445 0277 5567</Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

Figure D.10. XML Encryption

429. This markup represents that John Smith is using his credit card with a limit of $5,000USD. Smith's credit card number is sensitive information! If the application wishes to keep that information confidential, it can encrypt the CreditCard element:



```
<?xml version='1.0'?>
  <PaymentInfo xmlns='http://example.org/paymentv2'>
    <Name>John Smith</Name>
    <EncryptedData Type='http://www.w3.org/2001/04/xmlenc#Element'
     xmlns='http://www.w3.org/2001/04/xmlenc#'>
      <CipherData>
        <CipherValue>A23B45C56DFGH34RGBDGG777893GHZD</CipherValue>
      </CipherData>
    </EncryptedData>
  </PaymentInfo>
```

Figure D.11. XML Encryption

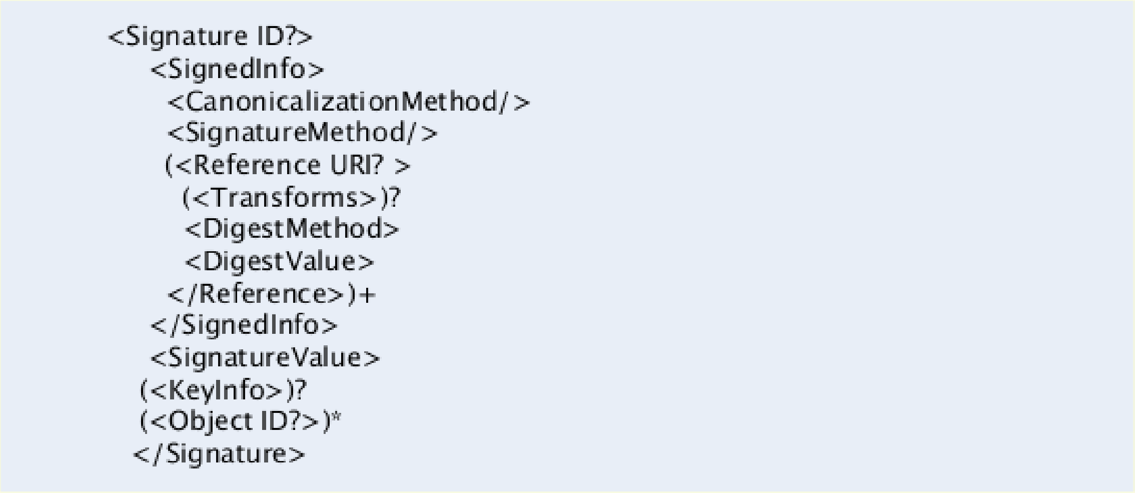
430. By encrypting the entire CreditCard element from its start to end tags, the identity of the element itself is hidden. (An eavesdropper doesn&apos;t know whether he used a credit card or money transfer). The CipherData element contains the encrypted serialization of the CreditCard element.

D.3.5.2. XML Signature

431. XML Signature (also called XMLDsig, XML-DSig, XML-Sig) is a W3C recommendation that defines an XML syntax for digital signatures. XML signatures can be used to sign data –a resource– of any type, typically XML documents, but anything that is accessible via a URL can

be signed. An XML signature used to sign a resource outside its containing XML document is called a detached signature; if it is used to sign some part of its containing document, it is called an enveloped signature; if it contains the signed data within itself it is called an enveloping signature.

432. XML digital signatures are represented by the Signature element which has the following structure (where "?" denotes zero or one occurrence; "+" denotes one or more occurrences; and "*" denotes zero or more occurrences):



```
<Signature ID?>
   <SignedInfo>
     <CanonicalizationMethod/>
     <SignatureMethod/>
    (<Reference URI? >
      (<Transforms>)?
       <DigestMethod>
       <DigestValue>
     </Reference>)+
   </SignedInfo>
   <SignatureValue>
  (<KeyInfo>)?
  (<Object ID?>)*
 </Signature>
```

Figure D.12. XML Signature

433. Within an XML document, signatures are related to local data objects via fragment identifiers. Such local data can be included within an enveloping signature or can enclose an enveloped signature. Detached signatures are over external network resources or local data objects that reside within the same XML document as sibling elements; in this case, the signature is neither enveloping (signature is parent) nor enveloped (signature is child). Since a Signature element (and its ID attribute value/name) may co-exist or be combined with other elements (and their IDs) within a single XML document, care should be taken in choosing names such that there are no subsequent collisions that violate the ID uniqueness validity constraint [XML].

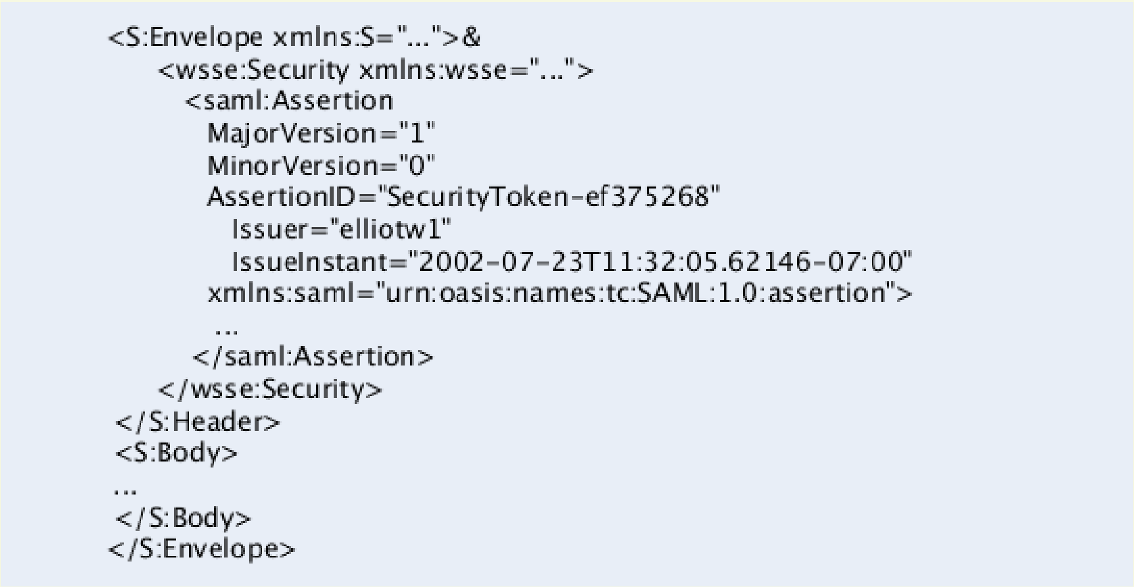
D.3.5.3. XML Token

434. A security token represents the user's claims and it's used by the Authentication Service for authenticate him. There are two kinds of security tokens:

- X.509 certificate
- SAML Assertion

435. In this document we discuss only the "SAML Assertion" in an additional point (not the X.509 certification).

436. The following XML example shows a SAML Token:



```
<S:Envelope xmlns:S="...">&
    <wsse:Security xmlns:wsse="...">
      <saml:Assertion
        MajorVersion="1"
        MinorVersion="0"
        AssertionID="SecurityToken-ef375268"
          Issuer="elliotw1"
          IssueInstant="2002-07-23T11:32:05.62146-07:00"
        xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion">
         ...
       </saml:Assertion>
    </wsse:Security>
 </S:Header>
 <S:Body>
...
 </S:Body>
</S:Envelope>
```

Figure D.13. SAML Token

437. The following figure shows the three XML security methods code in a SOAP (XML) Message.

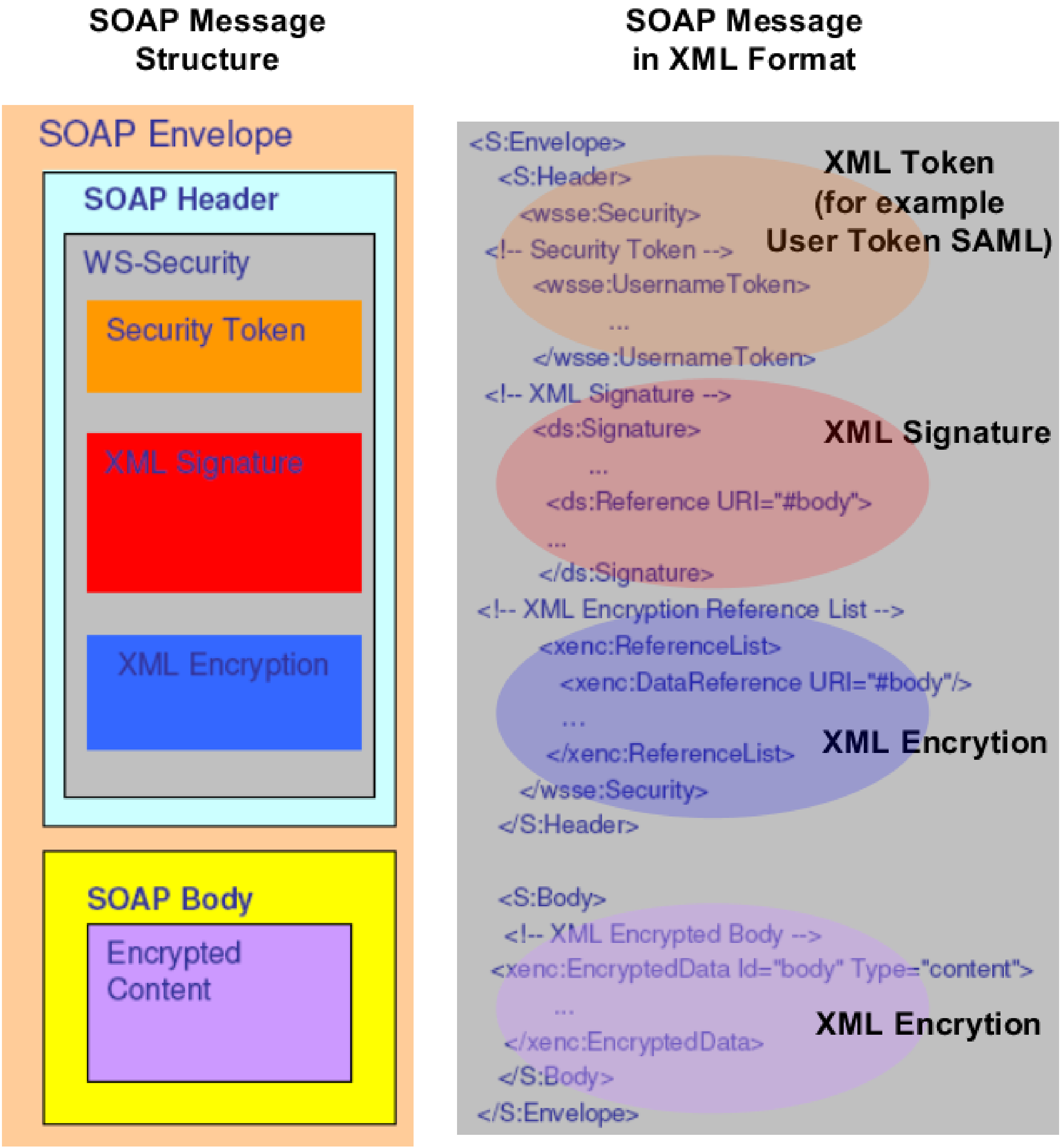


Figure D.14. SOAP Security based on XML Encryption & Signature & Token

D.3.5.4. XKMS: XML Key Management Specification

438. The XKMS specifies protocols for distributing and registering public keys, suitable for use in conjunction with the proposed standard for XML Signatures [XML-SIG] and an anticipated companion standard for XML encryption. The XML Key Management Specification (XKMS) comprises two parts – the XML Key Information Service Specification (X-KISS) and the XML Key Registration Service Specification (X-KRSS):

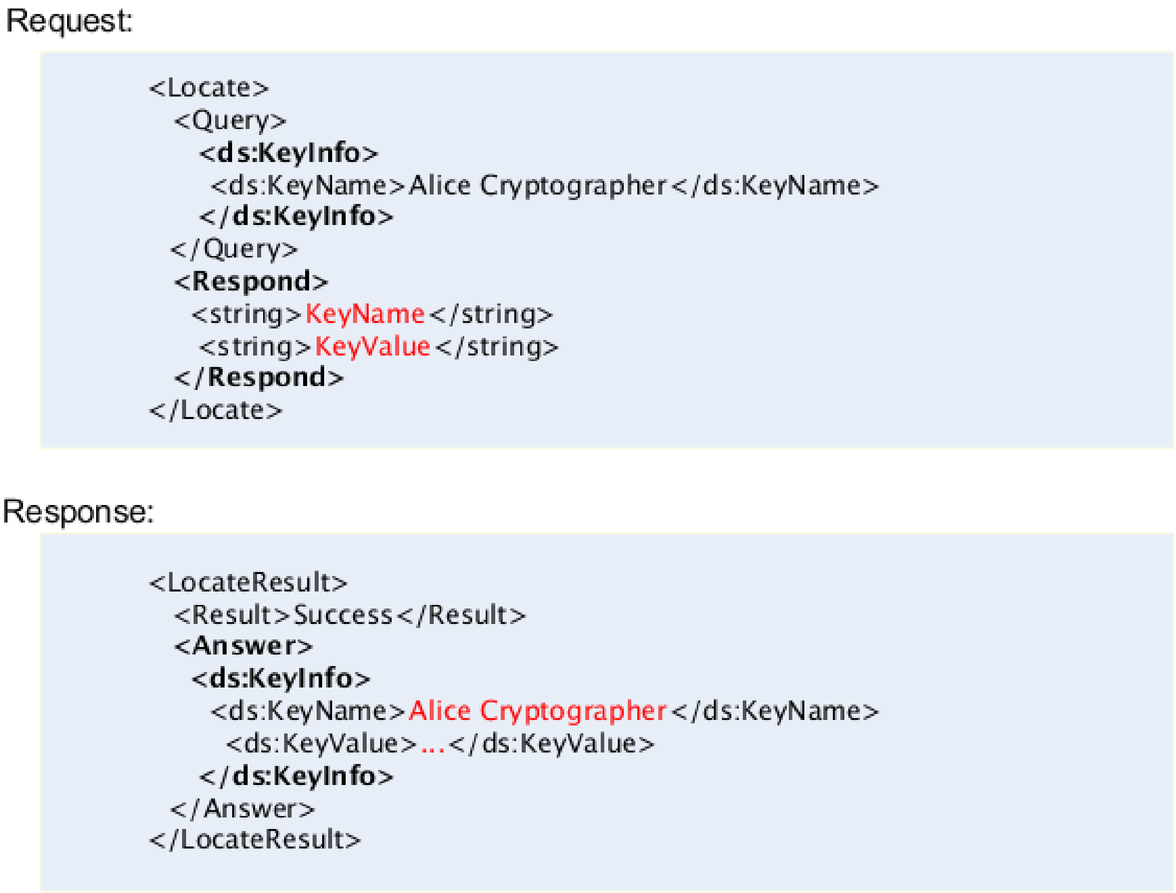
- *XML Key Information Service Specification (X-KISS)*: A protocol to support the delegation by an application to a service of the processing of Key Information associated with an XML signature, XML encryption, or other public key. Its functions include the location of required public keys and describing the binding of such keys to identification information.

- *XML Key Registration Service Specification (X-KRSS):* A protocol to support the registration of a key pair by a key pair holder, with the intent that the key pair subsequently is usable in conjunction with the XML Key Information Service Specification or higher level trust assertion service such as XML Trust Assertion Service Specification (XTASS).

439. *The underlying PKI may be based upon a different specification such as X.509/PKIX, SPKI or PGP – proposal: X.509/PKIX.*

440. Example for X-KISS:

441. The client is attempting to send an encrypted XML document and requires the public key encryption parameters of the recipient.



Request:

```
<Locate>
  <Query>
    <ds:KeyInfo>
      <ds:KeyName>Alice Cryptographer</ds:KeyName>
    </ds:KeyInfo>
  </Query>
  <Respond>
    <string>KeyName</string>
    <string>KeyValue</string>
  </Respond>
</Locate>
```

Response:

```
<LocateResult>
  <Result>Success</Result>
  <Answer>
    <ds:KeyInfo>
      <ds:KeyName>Alice Cryptographer</ds:KeyName>
        <ds:KeyValue>...</ds:KeyValue>
    </ds:KeyInfo>
  </Answer>
</LocateResult>
```

Figure D.15. SOAP Security based on XML Encryption & Signature & Token

D.3.5.5. SAML: Security Assertion Markup Language / SAML Architecture

D.3.5.5.1. SAML: Security Assertion Markup Language

442. Security Assertion Markup Language (SAML) is an XML-based standard for exchanging authentication and authorization data between security domains, that is between an identity provider (a producer of assertions) and a service provider (a consumer of assertions). SAML is a product of the OASIS Security Services Technical Committee.

443. The normal use case of SAML is support the Single Sign-On (SSO) issue, but SAML is also used for Identity Federation. SSO represents the ability of a user to authenticate in one domain and use resources in another domain WITHOUT re-authenticating. SAML is an XML Framework for exchanging security information over the internet. It enables different security services systems to INTEROPERATE.

444. The core SAML specification defines the structure and content of both assertions and protocol messages used to transfer this information. The next Figure illustrates the relationship between these basic SAML concepts.

445. SAML assertions carry statements about a principal that an asserting party claims to be true. The valid structure and contents of an assertion are defined by the SAML assertion XML schema. Assertions are usually created by an asserting party based on a request of some sort from a relying party, although under certain circumstances, the assertions can be delivered to a relying party in an unsolicited manner. SAML protocol messages are used to make the SAML-defined requests and return appropriate responses. The structure and contents of these messages are defined by the SAML-defined protocol XML schema.

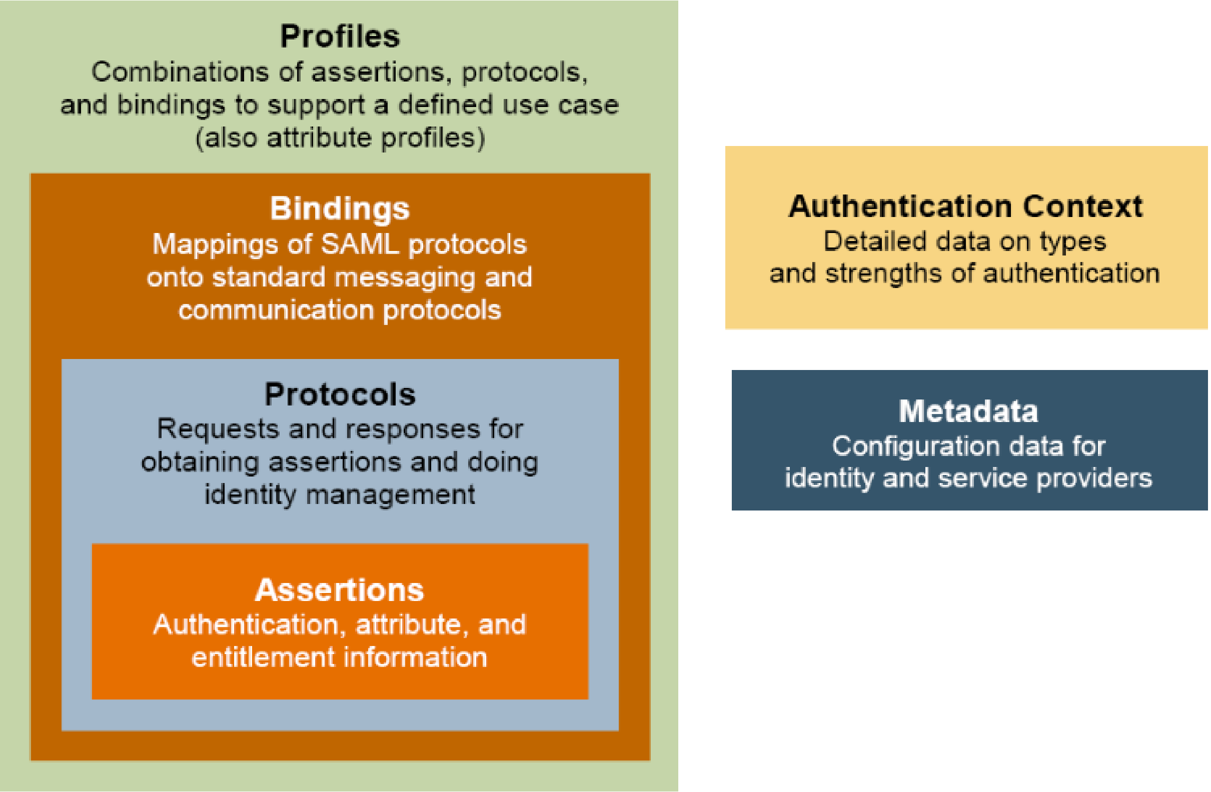

Profiles
Combinations of assertions, protocols, and bindings to support a defined use case (also attribute profiles)
Bindings
Mappings of SAML protocols onto standard messaging and communication protocols
Protocols
Requests and responses for obtaining assertions and doing identity management
Assertions
Authentication, attribute, and entitlement information
Authentication Context
Detailed data on types and strengths of authentication
Metadata
Configuration data for identity and service providers

Figure D.16. Basic SAML Concepts

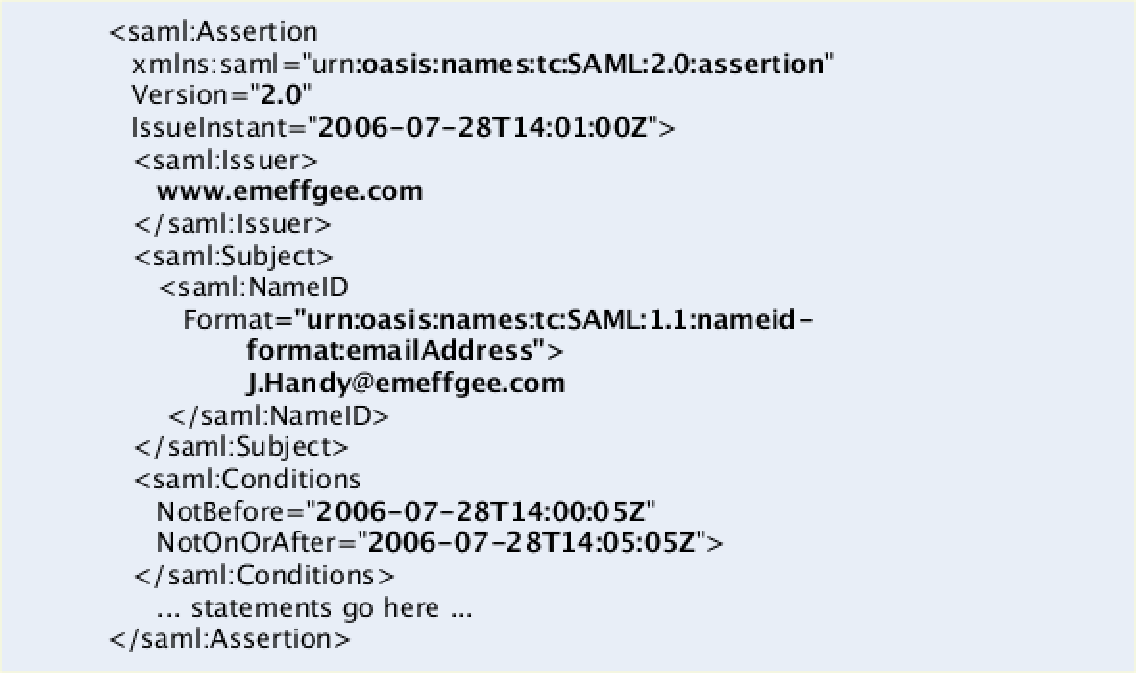
446. An assertion is a claim made by someone about someone else. SAML assertions are structured as a series of statements about a subject: Authentication, Attribute, Authorization Decision, or by an own customized statements. SAML defines three kinds of statements that can be carried within an assertion:

- *Authentication statements:* These are created by the party that successfully authenticated a user. At a minimum, they describe the particular means used to authenticate the user and the specific time at which the authentication took place.

- *Attribute statements:* These contain specific identifying attributes about the subject (for example, that user "John Doe" has "Gold" card status).

- *Authorization decision statements:* These define something that the subject is entitled to do (for example, whether "John Doe" is permitted to buy a specified item).

447. The following example shows a common portion Assertion:



```
<saml:Assertion
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  Version="2.0"
  IssueInstant="2006-07-28T14:01:00Z">
  <saml:Issuer>
    www.emeffgee.com
  </saml:Issuer>
  <saml:Subject>
    <saml:NameID
      Format="urn:oasis:names:tc:SAML:1.1:nameid-
           format:emailAddress">
           J.Handy@emeffgee.com
     </saml:NameID>
  </saml:Subject>
  <saml:Conditions
    NotBefore="2006-07-28T14:00:05Z"
    NotOnOrAfter="2006-07-28T14:05:05Z">
  </saml:Conditions>
    ... statements go here ...
</saml:Assertion>
```

Figure D.17. Assertion

448. An assertion contains one or more statements and some common information that applies to all contained statements or to the assertion as a whole. A SAML assertion is typically carried between parties in a SAML protocol response message, which itself must be transmitted using some sort of transport or messaging protocol.

449. The next Figure shows a typical example of containment: a SAML assertion containing a series of statements, the whole being contained within a SAML response, which itself is carried by some kind of protocol.

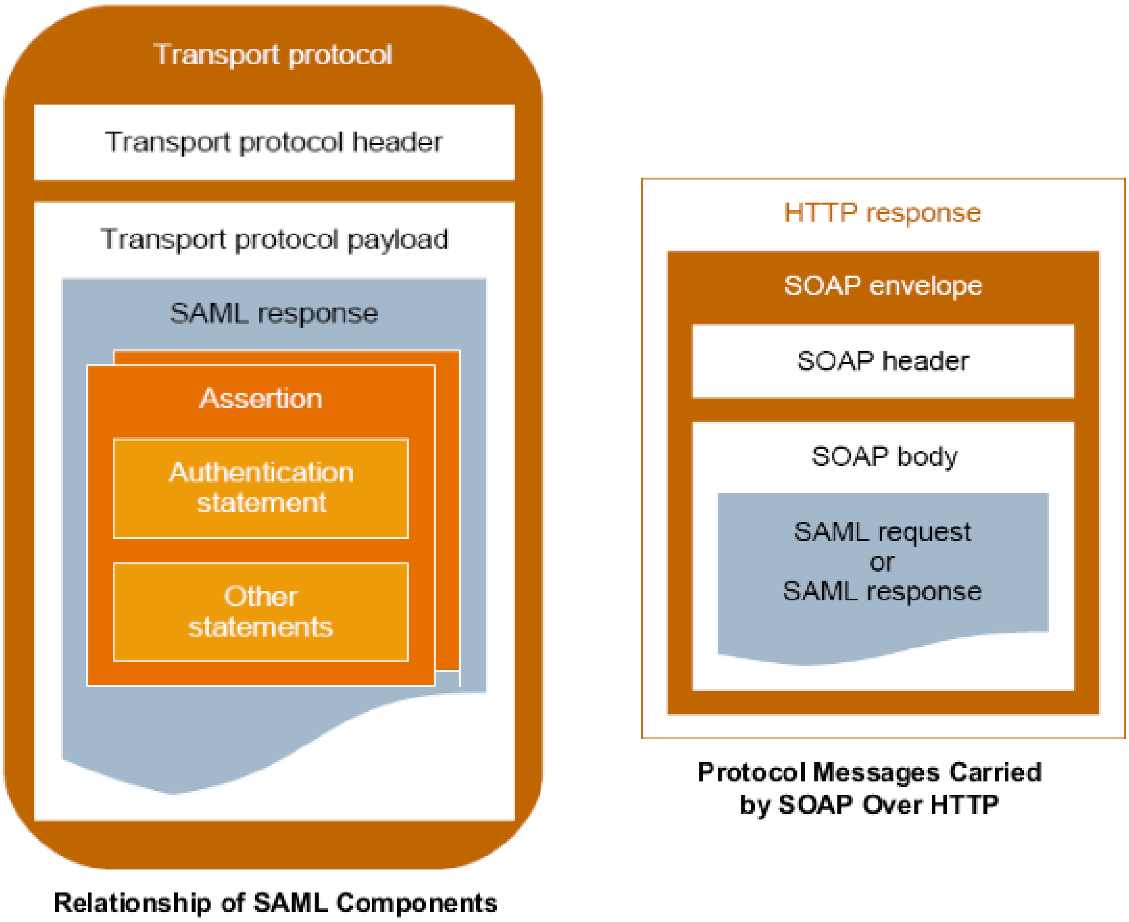

Transport protocol
Transport protocol header
Transport protocol payload
SAML response
Assertion
Authentication statement
Other statements
Relationship of SAML Components
HTTP response
SOAP envelope
SOAP header
SOAP body
SAML request or SAML response
Protocol Messages Carried by SOAP Over HTTP

Figure D.18. Relationship of SAML Components and Protocol Container

450. The means by which lower-layer communication or messaging protocols (such as HTTP or SOAP) are used to transport SAML protocol messages between participants is defined by the SAML bindings.

451. SAML defines a number of generalized request/response protocols:

- Authentication Request Protocol
- Single Logout Protocol
- Assertion Query and Request Protocol
- Artifact Resolution Protocol
- Name Identifier Management Protocol
- Name Identifier Mapping Protocol

452. SAML profiles are defined to satisfy a particular business use case, for example the Web Browser SSO profile. Profiles typically define constraints on the contents of SAML assertions, protocols, and bindings in order to solve the business use case in an interoperable fashion.

D.3.5.5.2. SAML Architecture (Web Security Architecture)

453. The following figure shows the chain of an access to a Service or to a Resource and the co-operation with the IT Security Services. The main SAML IT Security Services are:

- *Policy Enforcement Point (PEP):* A Policy Enforcement Point (PEP) at the resource provider formulates an authorization decision request (SAML or XACML) using the attributes and other information in the security context. The PEP sends this request to a Policy Decision Point.

- *Policy Decision Point (PDP):* The Policy Decision Point (PDP) combines the information in the request with policy obtained from a central policy store. The PDP renders an access control decision, which is returned to the PEP.

- *Policy retrieval point (PRP):* The component from which applicable policies may be retrieved. The communication protocol between PDP and PRP is XACML.

- *Policy Administration Point (PAP):* A Policy Administration Point (PAP) maintains authorization policy in a central location. The policy store is made available to the PRP for access control decisions.

- *Policy Information Point (PIP):* A Policy Information Point validates the specific attributes that are used for authorization.

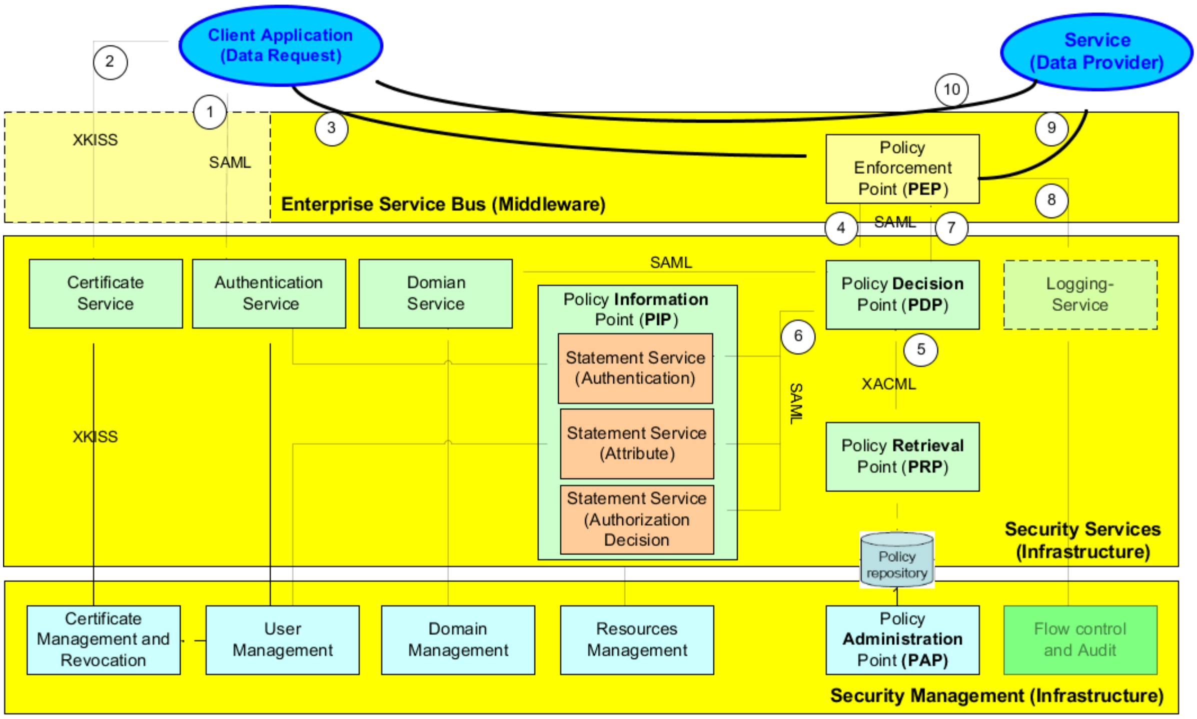

Client Application (Data Request)
Service (Data Provider)
1
2
3
4
5
6
7
8
9
10
XKISS
SAML
Enterprise Service Bus (Middleware)
Policy Enforcement Point (PEP)
SAML
Certificate Service
Authentication Service
Domian Service
SAML
Policy Information Point (PIP)
Statement Service (Authentication)
Statement Service (Attribute)
Statement Service (Authorization Decision
SAML
Policy Decision Point (PDP)
Logging-Service
XACML
Policy Retrieval Point (PRP)
XKISS
Security Services (Infrastructure)
Policy repository
Certificate Management and Revocation
User Management
Domain Management
Resources Management
Policy Administration Point (PAP)
Flow control and Audit
Security Management (Infrastructure)

Figure D.19. SAML Architecture (Overview)

454. The step sequence by an access of a client is:

- 1. Login based on a certificate;
- 2. Before an access the user needs may be some keys, which he can get via the Public Key Infrastructure using for example XKISS.
- 3. Service Consumer invokes a Service which was published earlier. The user (consumer) sends a request (SOAP message) via the Middleware Service Broker to the Provider. The SOA Middleware forwards the request to the PEP.
- 4. The PEP (Policy Enforcement Point) will capture the requirement for a service and pass the SAML onto the PDP for authentication and authorization validation of the obligation service.
- 5. The PDP receives the 'Authorization Decision Request' and requests a 'Policy Query' to the PRP. The PRP response with a 'Policy Statement'.
- 6. In addition the PDP checks some more, the User, Resource, and/or Context Attributes via some additional 'Statement Services'.
- 7. Based on the check results (Policy and Statements) the PDP decides the access (permit, deny, not applicable or indeterminate).
- 8. The PEP receives the access result, triggers the logging and the flow control.

- 9. If all access requests are valid, the PEP forwards the user request to the provider.
- 10. The Service Provider supports the user request.

D.4. ENTERPRISE SERVICE BUS / MESSAGING SYSTEM PROFILE

455. An *Enterprise Messaging System (EMS)* is a set of published Enterprise-wide standards that allows sending of semantically precise messages between computer systems. EMS systems promote loosely coupled architectures that allow changes in the formats of messages to have minimum impact on message subscribers. EMS systems are facilitated by the use of XML messaging, SOAP and Web services.

456. An *Enterprise Service Bus (ESB)* generally provides an abstraction layer on top of an implementation of an enterprise messaging system, which allows integration architects to exploit the value of messaging without writing code. Contrary to the more classical enterprise application integration (EAI) approach of a monolithic stack in a hub and spoke architecture, the foundation of an enterprise service bus is built of base functions broken up into their constituent parts, with distributed deployment where needed, working in harmony as necessary.

457. The following figure shows a layer model for the "Enterprise Messaging System (EMS) / Enterprise Service Bus (ESB)" based on the chapter: "Web Service Fundamentals".

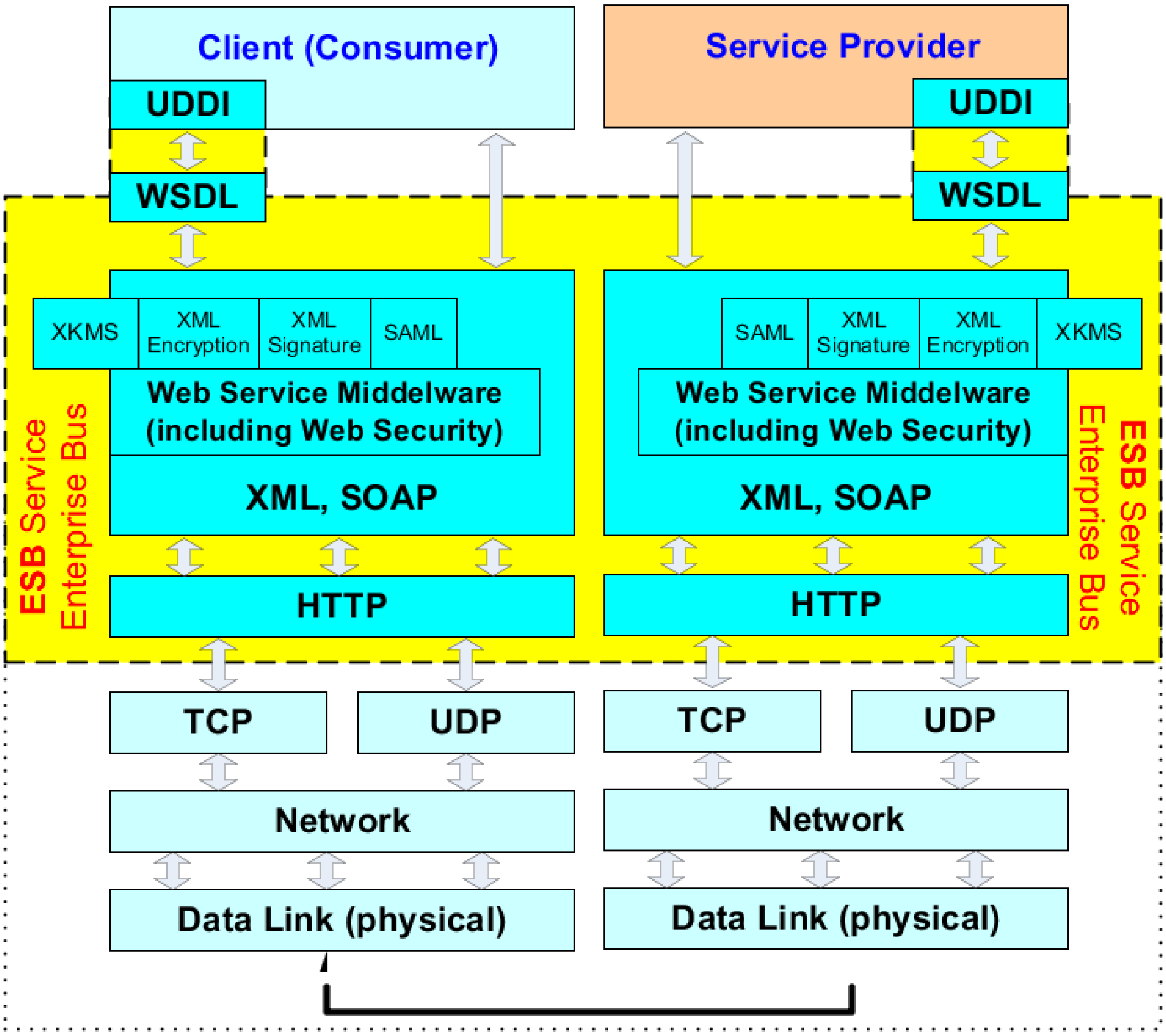


Figure D.20. ESB / EMS Layer Model

458. Based on the Standards (signed in an ESB / EMS Profile) every Client and Server can be integrated into the ESB system.

459. *WS-I Basic Profilee*

460. The WS-I Basic Profile V1.0 specifies a set of usage scenarios and Web services standards that can be used to integrate systems. It focuses on the core foundation technologies upon which Web services are based. Basic Profile V1.0 was approved unanimously on July 22, 2003, by the WS-I board of directors and members.

461. The WS-I Basic Profile V1.0 – Profile Specification consists of the following non-proprietary Web services related specifications:

- SOAP V1.1

- WSDL V1.1
- UDDI V2.0
- XML V1.0 (Second Edition)
- XML Schema Part 1: Structures
- XML Schema Part 2: Datatypes
- RFC2246: The Transport Layer Security Protocol Version V1.0
- RFC2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile
- RFC2616: HyperText Transfer Protocol V1.1
- RFC2818: HTTP over TLS
- RFC2965: HTTP State Management Mechanism
- The Secure Sockets Layer Protocol Version V3.0

462. UPDATE: A combined claim of conformance to the **WS-I Basic Profile 1.1** and the **Simple SOAP Binding Profile 1.0** is roughly equivalent to a claim of conformance to the *WS-I Basic Profile 1.0 plus published errata.*

463. Additional there are many successful implementations of the basic Web services standards, particularly SOAP and WSDL but many aspects of service interaction and integration are not directly supported by those basic standards, such as security, transactional, delivery assurance, and process modelling – for example WS-Security, WS-Trust, WS-Privacy, and WS-Policy. It also accommodated existing security technologies such as Kerberos, XML Digital Signatures, and XML Encryption.

464. Because of this an "ESB / EMS Profile" proposal is defined on the next page.

465. *ESB / EMS Profile proposal:*

466. A proposal for an ESB / EMS Profile could be based on the WS-I profiles:

- WS-I Web Service Basic Profile, v1.1:2nd ed. 2006
- WS-I Simple SOAP Binding Profile v1.0:2004

467. With the following parts of the WS-I profiles:

- Simple Object Access Protocol (SOAP) 1.1
- RFC2616: Hypertext Transfer Protocol – HTTP/1.1
- RFC2965: HTTP State Management Mechanism
- Extensible Markup Language (XML) 1.0 (Second Edition)

- Namespaces in XML 1.0
- XML Schema Part 1: Structures
- XML Schema Part 2: Datatypes
- Web Services Description Language (WSDL) 1.1
- UDDI Version 2.04 API Specification, Dated 19 July 2002
- UDDI Version 2.03 Data Structure Reference, Dated 19 July 2002
- UDDI Version 2 XML Schema
- RFC2818: HTTP Over TLS
- RFC2246: The TLS Protocol Version 1.0
- The SSL Protocol Version 3.0
- RFC2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile

468. And with the following additional parts included in the ADaP-34 NISP-Vol2-v2 Draft, but not in the WS-I profiles:

- TCP (IETF STD 7:1981, RFC0793:1981 updated by RFC3168:2001)
- UDP (IETF STD 6:1980, RFC0768:1980)

469. And with the following additional parts (not in the WS-I profiles and not in the ADaP-34 NISP-Vol2-v2 Draft):

- XML Encryption Syntax and Processing (W3C Recommendation 10 December 2002)
- XML Signature Syntax and Processing Second Edition (W3C Recommendation 10 June 2008)
- Security Assertion Markup Language, SAML v1.1 (OASIS)
- XKMS: XML Key Management Specification (W3C Note 30 March 2001)
- XACML eXtensible Access Control Markup Language Version 2.0 (OASIS Standard, 1 Feb 2005)

D.5. APPENDIX: ESB REQUIREMENTS IN A MILITARY ENVIRONMENT

470. This appendix describes an extract of some main requirements for an ESB in a military, mainly tactical, environment.

471. *Mobility and Availability of the ESB Infrastructure:*

472. One of the main requirements in the military sector is the mobility of a mobile client and service. The service and the client change the location and the ESB must support that location changes. Additional the ESB Infrastructure on the client and on the service must also be mobile. Independent from the status of the ESB environment the user should work in worst case locally on his equipment (using the local services).

473. Furthermore the ESB infrastructure must be on one site mobile and on the other site the ESB infrastructure must be also redundant in case of breakdown. For example the SAML Architecture (PEP, PDP, ...) and the PKI and the usage must be mobile and redundant. Therefore it is necessary to provide a replication of the critical and important data of the ESB infrastructure.

474. The current implementation of the ESB infrastructure looks like a more static environment with some redundancies.

475. *Bandwidth in a Military Environment:*

476. In a tactical, mobile environment low bandwidth is a major topic. Highly mobile military networks use for example radio communication (VHF, UHF) or Tactical Data Links (Link 16/22, VMF, JREAP, ...), SATCOM, directed antenna systems etc.. Contrary to this requirement the current ESB environment requires a high bandwidth and is IP based.

477. That means due to the design of an ESB the communication system must be able to fulfil its requirements. On one site some communication equipments must be improved like IP communication via radio, but on the other site the ESB design must take care about a low bandwidth (data rate). One improvement on the ESB design could be the usage of "Binary XML".

478. Binary XML, or Binary Extensible Markup Language, refers to any specification which defines the compact representation of XML in a binary format. How to involve Binary XML into SOA/SOAP and/or into the ESB Infrastructure and into the Client/Service Architecture (data model), is to analyze.

479. Remark: At the beginning it makes sense to start in an environment with higher bandwidth, but by the design of the ESB and the target of the ESB should be to support networks based on IP with lower bandwidth.

480. *ESB Security in a Military Environment:*

481. Because of the not included Security Standards, the security implementation in the ESBs isn't uniform and also some features are not implemented. Additional in the military environment the requirements related to the Security is different and especially from the nations.

482. Related to this it is necessary to define a security standard into the ESB / EMS Profile. Then based on this the implementation should be arranged in the ESB environment.

483. *Online and Offline ESB Management:*

484. In the military environment an ESB Management (including Service Management) is required. For example it is not only necessary to manage the access on a service at the first. Also

due to the runtime of an operation between a client and service it could be necessary to change the service profile – for example: role/priority based reduction or refusal of a service usage, or changing of the setting of the QoS (Quality of Service agreement).

485. Furthermore a flexible and mobile management and monitoring of the ESB Infrastructure together with the service provider is required.

486. *Interoperability: Non-interoperable ESB implementations*

487. Currently a lot of implementations for an ESB exist like SOPERA, WebSphere, ORACEL ESB (BPEL), Software-AG ESB. Every ESB implementation contains an own framework which should be included on the client and server application. In the most ESB implementations the Client/ESB interface and the Service/ESB interface is proprietary. That means, it isn't possible to contact with a Service or Client based on the ESB "A" implementation the ESB "B". This is only possible in this case, if the Client/ESB interface and the Service/ESB interface are implemented based on standards (for based on an ESB / EMS Profile).

488. Currently a lot of different ESB implementations exist and the current aim (workaround / first step) is to enable federation of two or more ESB architectures that conform to the common specification for ESBs. The Appendix chapter 6 explains an example of a Federated ESB Reference Architecture.

D.6. APPENDIX: FEDERATED ESB REFERENCE ARCHITECTURE

489. The following chapter is based on the document "ESB Interop Spec for Federation" from MoD UK, EDS, IBM and ORACLE (a proposal for federated ESB for ESBs based on different technology).

490. The ESB is an architectural component which provides a set of services (or capabilities). The component itself exposes a set of services which are characterised by a protocol, one or more addresses and specifics ways of handling invocations, such as security. It also uses a set of ports to integrate with service providers (such as application functions). The capabilities provided by the ESB may include message format transformation and protocol conversion, and it offers a number of interactions styles including request-response and publish-subscribe.

491. Currently exists a lot of different ESB implementations and the aim is to enable federation of two or more ESB architectures that conform to the common specification for ESBs.

492. The following list of high-level requirements has been used to give an overview for the (Federated) ESB Reference Architecture:

- It should be transparent to Service Consumers and Providers where the services they invoke are being delivered from (i.e. from which ESB).
- Services can be exposed for either internal or external consumption. Service Consumers external to an ESB will not have access to that ESB's internal Service Providers.

- Every message which emanates from a service should be identifiable and traceable back to its origin via a UUID. A component of this UUID is an identifier assigned to the domain when it joins the federation. (Alternatively time to live may need to be applied, so when set to 0 the message is private.) This prevents infinite loops.
- ESBs must provide a mechanism for authenticating Service Consumers and for controlling their access to the services the ESB exposes externally.
- ESBs must provide a facility for managing and publishing up-to-date service end-points for the services governed within the immediate zone and must be capable of storing service end-points for services offered by other governance zones.
- ESBs should audit the processing of a service that it offers and make available the audit records captured, upon request from the Service Consumer that invoked the service.
- Exceptions trapped by services invoked must be handled in a consistent manner across all ESBs within a federation. Whilst each ESB may implement exception handling differently, they must report errors to the Service Consumers following an agreed format and reporting mechanism.

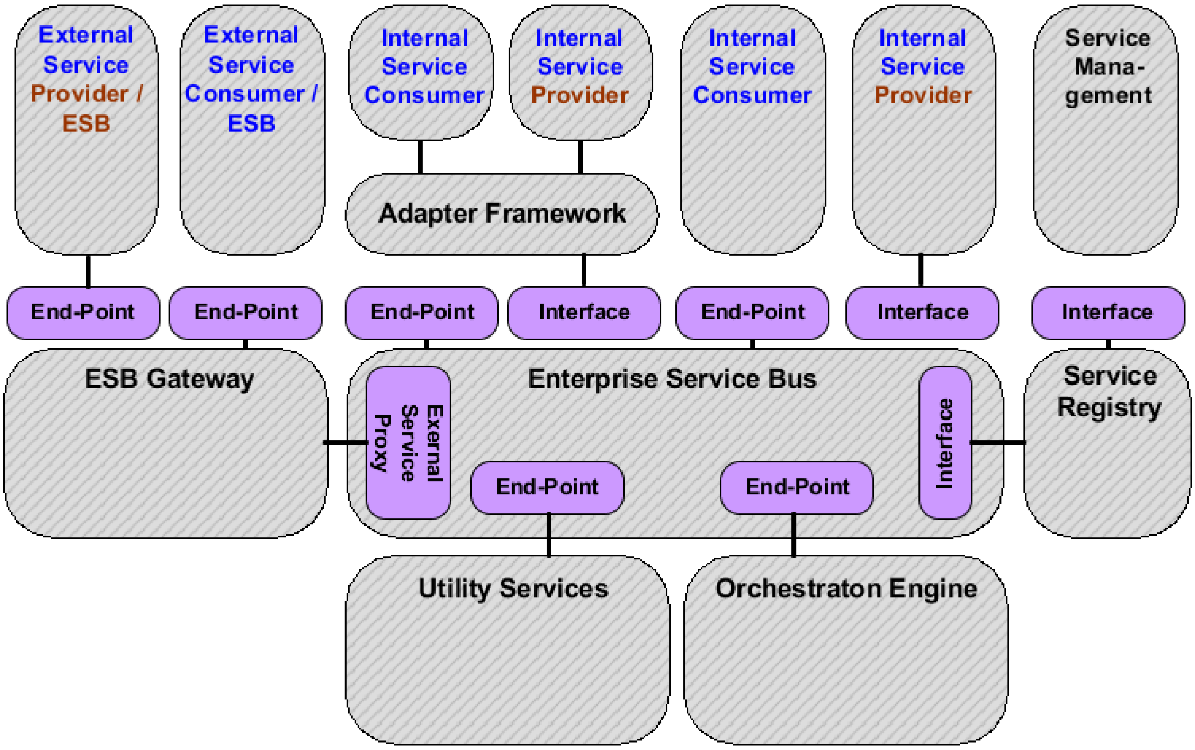


Figure D.21. Federation ESB Architecture Overview

493. The diagram above provides a logical overview of the subsystems that have been used to provide a Service Oriented Infrastructure for delivery of Service Orientated applications.

494. The ESB Gateway acts as a proxy to provide controlled access to the ESB. A principal use of the ESB Gateway is to expose services in a consistent manner across all governance zones. This node allows generic actions to be defined and performed on all calls to services such as logging, auditing, monitoring, security, and threat protection.

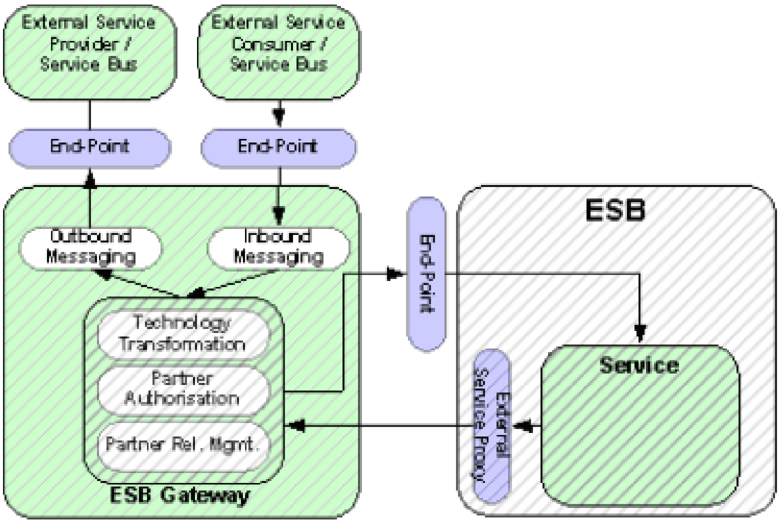


Figure D.22. ESB Gateway

495. The ESB Gateway provides the single-point of control for federated service invocations, that either originate with a local Service Consumer call to a remote service (service hosted on external ESB) or a remote Service Consumer call to a local Service Provider.

496. Local Service Providers and Service Consumers are shielded from the federated ESBs (their consumers and providers) by the ESB Gateway, which separates all aspects of external ESB interoperability from how services are provided and consumed locally.

497. ESBs are federated on the basis that each of the zone's ESBs is autonomous, and yet they all have knowledge of the wider enterprise-level services. The next figure shows a topology for federated ESBs in which any consumer can call services in any zone without necessarily having set up the communication paths in advance.

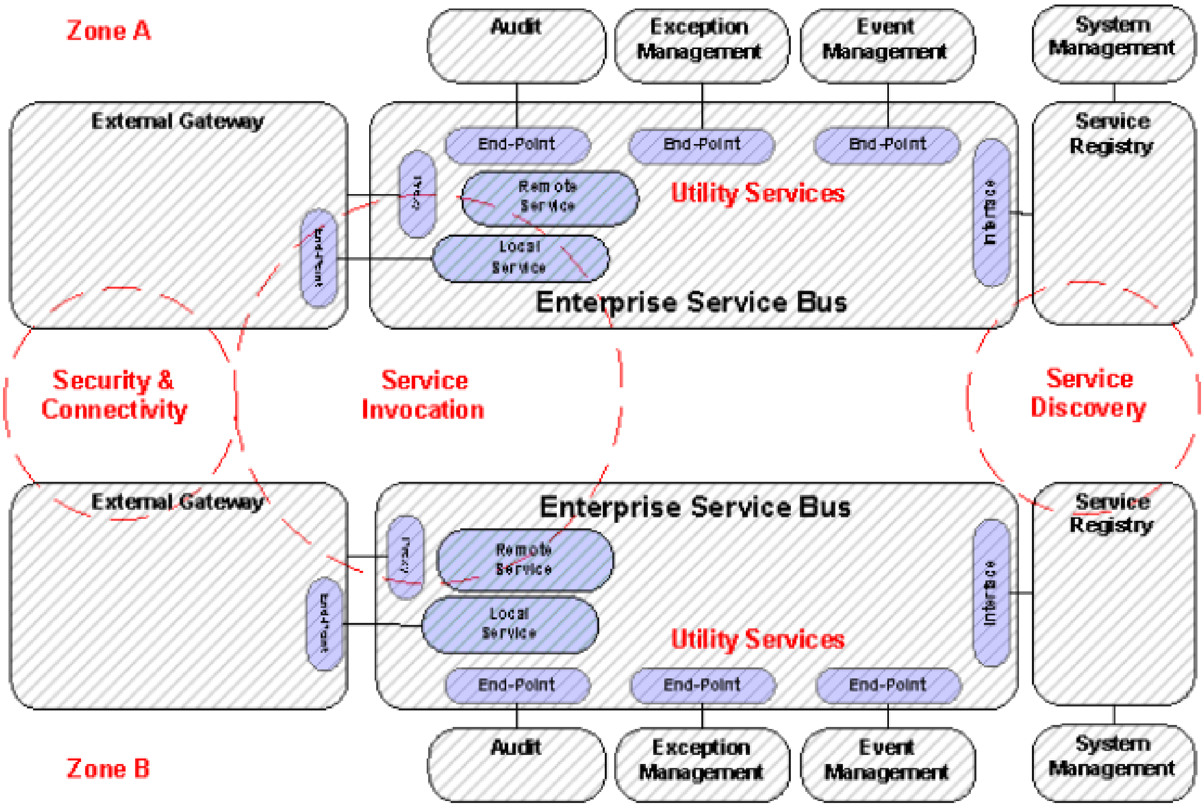


Figure D.23. Federated ESB

D.7. LIST OF ABBREVIATIONS

Abbreviation	***Item***
API	Application Programming Interface
EMS	Enterprise Messaging System
ESB	Enterprise Service Bus
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IP	Internet Protocol
IT	Information Technology
JMS	Java Message Service
MoD	Ministry of Defence
OSI	Open System Interconnection
PAP	Policy Administration Point
PDP	Policy Decision Point
PEP	Policy Enforcement Point

Abbreviation	***Item***
PIP	Policy Information Point
PKI	Public Key Infrastructure
SAML	Security Assertion Markup Language
SATCOM	Satellite Communication System
SSO	Single Sign-On
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
TCP	Transport Control Protocol
TLS	Transport Layer Security
UDDI	Universal Description Discovery and Integration
UDP	User Datagram Service Protocol
UHF	Ultra High Frequency
URL	Uniform Resource Locator
VHF	Very High Frequency
WSDL	Web Service Description Language
WS-I	Web Service Interoperability
XACML	eXtensible Access Control Markup Language
XKMS	XML Key Management Specification
XML	eXtensibke Markup Language
X-KISS	XML Key Information Service Specification
X-KRSS	XML Key Registration Service Specification

Table D.1. Acronyms

E. GUIDELINES USING DESIGN RULES IN NATO NEC FEDERATED ENVIRONMENT

Summary

498. This guideline document describes a concept and model for how knowledge of proven solutions can be documented and packaged in order to form a shared basis for supporting the development and the implementation of NNEC based systems for NATO.

E.1. INTRODUCTION

499. This document introduces the concept of design rules by describing what design rules are and how they shall be applied in a NATO Network Enabled Capabilities context.

500. Design rules are about reusing knowledge of proven solutions for reoccurring problems. Reuse of solutions that give NNEC-specific characteristics is particularly important. These solutions should solve frequent and/or difficult problems, promote important system characteristics and/or improve the quality of the resulting product in a cost effective way.

501. A design rule consists mainly of the following three parts:

- Context; describes under what circumstances the design rule is valid
- Problem/Opportunity; is a description of the problem it solves or the opportunity it exploits.
- Solution; is a description how the problem/opportunity shall/should be resolved in the given context

502. Design rules can give solutions on all levels, but it is anticipated that the produced design rules mainly takes care of the higher system levels (relating to the breakdown patterns in a system design) in order to avoid a cumbersome number of rules. If possible design rules shall be based on standards and/or NISP/NAF and will preferably be associated with as concept (generic concept of design).

503. The introduction of design rules in the NISP will also need to be integrated with other design related artefacts and frameworks within NATO such as the NATO Architectural Framework (NAF).

E.2. GENERAL

E.2.1. Target Group

E.2.2. Definitions, Abbreviations and Acronyms

Acronym	Explanation	Reference	Definition
DR	Design Rule	NOSWG	A standardized, reusable solution to a design problem in a specific context within

Acronym	Explanation	Reference	Definition
			a problem space that provides value to the user. Note: There are four (4) types of design rules: a. A development method that supports the life cycle perspective; b. A defined structure that supports descriptions of complex relations; c. A detailed description of suggested technical solutions; d. A proven and reusable solution for a generic problem.
DRP	Design Rule Package	NOSWG	A specific set of design rules that make up a solution package within a defined problem area.
SIOP	service interoperability point	EAPC(AC/322)D(2006)0002-REV1	A reference point within an architecture where one or more service interfaces are physically or logically instantiated to allow systems delivering the same service using different protocols to interoperate. Note: A service interoperability point serves as the focal point for service interoperability between interconnected systems, and may be logically located at any level within the components, and its detailed technical specification is contained within a service interface profile.
SIP	service interface profile	EAPC(AC/322)D(2006)0002-REV1	A set of attributes that specifies the characteristics of a service interface between interoperable systems in the Networking and Information Infrastructure. Note: A service interface profile is identified at a service interoperability point in an architecture system view.

E.2.3. References

Referenced documents

[1] C. Alexander et al. 1997 A Pattern Language, Oxford University Press, New York,

[2] E. Gamma, R. Helm, J. Vlissides 1995. Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA: Addison-Wesley

[3] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad and M. Stal. 1996. Pattern-Oriented Software Architecture, A System of Patterns. New York: John Wiley and Sons

[4] Designrules, in the commercial world. David B. Kim Clark

E.3. BACKGROUND

504. Packaging knowledge into something reusable is nothing new in the software engineering field of science. Almost ten years ago a book was published that made a huge impact on how software engineers look upon packaging and sharing knowledge of proven solutions. The Design Pattern-book gave the engineers a tool not only on how to describe, formalize, package and distribute their knowledge and experience but also a tool on how to discuss different possible solution alternatives to a specific problem. It enables efficiency in both the communication and the implementation of software design, based upon a common vocabulary and reference.

505. The design pattern concept described in this book was not an original idea but the adaptation of the ideas from a building architect, Dr Christopher Alexander, who wrote a book on patterns found when categorizing floor plans, buildings, neighbourhoods, town, cities, etc. In that book Alexander writes:

506. “Each pattern is a three-part rule, which expresses a relation between a certain context, a problem, and a solution.”

507. This is the central thing about being able to package our knowledge and experience. It is not enough to describe a solution. To make a solution useful you also have to state what problem the solution solves or what opportunity that the solution makes possible as well as the context in which the problem/opportunity – solution pair is valid. For instance, the optimal solution to the problem on how to enter and exit a building will be very different in the context of a building situated in Stockholm or somewhere in the arctic.

508. The design patterns from the Design Pattern-book are the type of patterns that have become most widely known. These patterns solve problems or makes opportunities possible at a analysis or design level of abstraction. However, this is not the only level of abstraction covered by patterns. 1996 an important piece of work regarding patterns was published dealing with patterns on an architectural level of abstraction. This book identified patterns for system architecture at a

higher level than the original design patterns. The patterns relate to the macro-design of system components such as operating systems or network stacks.

509. After this, patterns of higher and higher level of abstraction have been published, sometimes, but not very often, also on lower levels. A specific level of interest to us is the system level-of abstraction. System-level patterns identify and describe the overall structure and interactions that can occur between components of a system. Furthermore, Enterprise-level patterns are possible, showing how to efficiently organise ones enterprise and what type of services to offer to its clients.

510. Consequently, mechanisms similar to the design rules described in this guideline have been used in different contexts and at different levels of abstraction. In many cases they have been quite popular and proven practical. Thus, it can be assumed that the design rule concept can be an efficient means to provide reuse of knowledge within the future development of the NNEC.

E.4. DESIGN RULES SUMMARY

E.4.1. Introduction to design rules

511. Design rules are about reusing knowledge of proven solutions for reoccurring problems. Reuse of solutions that give NNEC-specific characteristics is particularly important. These solutions should solve frequent and/or difficult problems, promote important system characteristics and/or improve the quality of the resulting product in a cost effective way.

512. Design rules consist mainly of the following three parts:

- Context; describes under what circumstances the design rule is valid
- Problem/Opportunity; is a description of the problem it solves or the opportunity it exploits.
- Solution; is a description how the problem/opportunity shall/should be resolved in the given context

513. Design rules can give solutions on all levels, but it is anticipated that the produced design rules mainly takes care of the higher system levels (relating to the breakdown patterns in a system design) in order to avoid a cumbersome number of rules. If possible design rules shall be based on standards and/or NISP/NAF and will preferably be associated with as concept (generic concept of design).

514. A design rule package is a mechanism for packaging of design rules (by reference) within a certain domain or for a specific kind of system. The dependencies between design rules that are part of a design rule package shall be defined and minimized.

E.4.2. Benefits from using design rules

515. In today’s knowledge oriented organizations it is very important to make sure that the knowledge of people is preserved in the organization even if the people change positions or

leave the company. Design rules are important tools to be able to aid the process of managing this knowledge since they force documentation of knowledge in a structured way.

516. The use of design rules to document and package proven solutions is expected to speed up development, and reduce cost and risk, by reusing knowledge on how to solve recurring problems and by providing verified solutions to those problems.

517. Moreover, the use of design rules provide the means to coordinate development of different federated systems in order to make them network enabled and facilitate the evolvement of combined capabilities. Another important aspect is also that design rules aid organisations in creating a common understanding of the problems and challenges they are facing.

E.4.3. Consequences of using design rules

518. In order for design rules to have effect in an organization there must be a framework which describes what design rules are and how they shall be used, i.e. this document. Design rules will also affect the way solutions are described and must be an integral part of the architecture description framework.

519. Another important thing to remember is that design rules will affect the way we work, thus putting new requirements on the processes and people within our organization.

E.5. DESIGN RULES IN A NATO NEC FEDERATED ENVIRONMENT

520. This guideline document describes a concept and model for how knowledge of proven solutions in the form of design rules can be documented and packaged in order to form a shared basis for the future development of NNEC based systems for NATO.

521. The processes in which design rules are identified, produced and used are not described within this guideline.

E.5.1. Problems or opportunity description

522. In the development of large systems of systems or federated systems for the future needs of the NATO there are several problems to be solved as well as opportunities to exploit. The problems range from what methods to use for requirements capture and design to how to solve detailed technical matters.

523. In order to be able to establish a set of building blocks that can be used to meet the needs of the future NNEC, design regulations are absolutely essential if the building blocks shall be possible to be used together and combined in different ways, from a technical as well as from a business point of view.

524. Design regulations in this context are the descriptive or normative regulation work necessary for NATO nations to be able to implement, configure and use systems in a federated en-

vironment. This includes not only technical and business design, but also the ability to manage and maintain these regulations to be able to provide the NATO nations with flexible component based systems.

525. Moreover, there is a strong incentive to endorse reuse of proven solutions or implementations and thus get a more cost-effective solution. The overall quality is also expected to benefit from this kind of reuse.

526. In this document we will focus on the model for design rules, and the patterns for setting up the SIOP and SIP:s between federations, this in order to be able to exchange information services between parties.

527. Design rules patterns and knowledge for supporting NATO Nations in designing NNEC compliant components and services can also be retrieved from different Nations repositories as reference architectures, Sweden Design rules (releasable to NATO) will be included as one of the Partner nations reference architecture as recommended and proven patterns in order to achieve NNEC interoperability.

E.5.2. Solution

E.5.2.1. Design rules in the NNEC context

528. Design rules are about reusing knowledge of proven solutions. In the context of NNEC we are especially interested in reuse of solutions that provide typical NNEC characteristics. In addition to this, the use of design rules aim at making the development of NNEC more cost-effective and improve the quality in the resulting products.

529. As mentioned before, a design rule is in the most general description a three-part rule, which expresses a relation between a certain context, a problem or an opportunity and a solution.

530. Different design rules may be in conflict with each other, e.g. in that the solution of one design rule can be incompatible with the solution of the other.

531. Moreover, design rules can be singular or aggregates meaning that it either is an atomic rule or an aggregate of rules that together constitute the rule. The aggregate may include rules on how to combine the possibly conflicting aggregated rules in order to generate a rule according to the current priorities.

532. Design rules may be implemented for solutions on different levels. There may be design rules for specific technical design problems or rules, how to handle a major business opportunities. It is however anticipated that the majority of design rules valid for an NNEC-system will be focused on the higher levels.

533. Design rules can be used in order to meet functional as well as non-functional needs of the system of interest. It should be clear from all design rules which problem or opportunity it is supposed to solve.

E.5.2.2. General guidance for using design rules

534. The prime prerequisites for implementing a design rule are:

- The use of the design rule shall make the resulting design “NNEC-compliant”, i.e. the design rules shall provide essential NNEC-characteristics such as flexibility, interoperability, security and usability

- A design rule shall provide a solution to frequently shown problems, to enable reuse of solutions or implementations and thus get a more cost-effective solution.

- A design rule shall provide a solution to difficult problems, or explore an opportunity, i.e. be a part of the corporate or federated memory

- A design rule shall improve the quality of the resulting product relative a product solution not using the design rule.

535. At least one of the mentioned prerequisites should be fulfilled. There may of course be other valid prerequisites, which will be assessed and used to initiate the design of a design rule.

536. Design rules shall consist of either atomic rules or aggregates of rules that together shall constitute the rule. The aggregate may include rules on how to combine the possibly conflicting rules in order to generate a rule according to the priorities.

537. An atomic design rule must not contain solutions for more than one subject area, e.g. mixing of business and technical subjects shall be avoided. Detailed technical rules shall in the same way be separated from rules of information or logical nature.

538. Design rules shall where applicable be based on concepts and rules in an extended NATO Architecture Framework.

539. A design rule shall not be of too low granularity or too trivial in order to avoid an explosion in the number produced of design rules. To achieve the approved mandatory validity, a design rule shall specify the way to solve the problem it is intended for. Rules that can be expressed in single sentences are collected in general sections in the design rule solution part.

540. Great efforts shall be made to ensure that the design rule is maintainable. This is primarily achieved by limiting the problem area that the design rule is intended for. More complex problems or opportunities shall be supported by aggregates of rules.

E.5.2.3. Design rule model

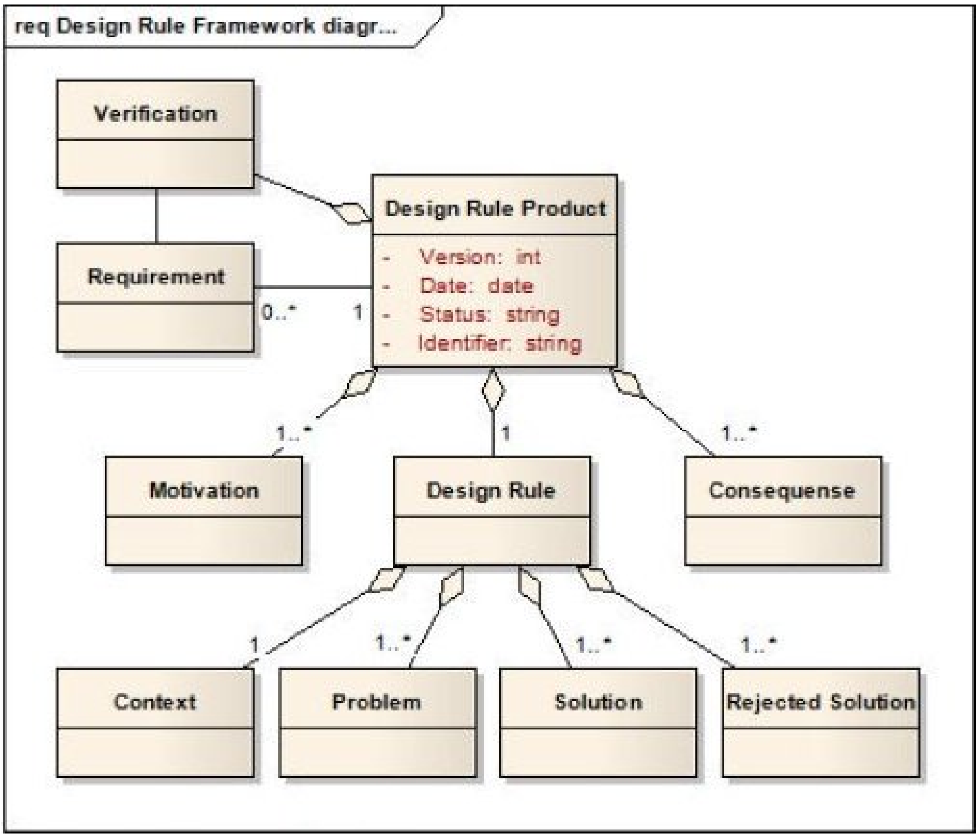


Figure E.1. Design rule model

541. The design rule product consists of:

- The basic design rule which, as already described, is a three part rule consisting of context, problem and solution. This shall also be complemented with one or more rejected solutions, i.e. solutions which shall not be used.
- An analysis and motivation why the solution fits the problem in the given context. This needs to be linked to direct business benefits such as cost savings or increased efficacy in operations.
- A description of the consequences from the proposed solution which is used to create an understanding at what cost the solution comes. This could include financial impacts, but also how people, processes or technology needs to be adjusted in order to achieve the solution. When describing the consequences from a design rule solution the impact on (at least) the following areas should always be considered:
  - Security
  - Interoperability

  - Cost
  - Usability
  - Flexibility and
  - Procedures
- Verification information which explains how the application of the rule can be verified.

542. A template for design rules, including guidelines, is defined in a separate document.

543. A design rule product is like Standards in the NISP related to near, mid and far term. A design rule can also exist in different versions with different status. The status of the design rule indicates which state of development the design rule is in.

- Candidates
- Approved
- Disposed

544. The solution described in a design rule may refer to other design rules to form an aggregate design rule. This may be the case for instance in a design rule describing a configuration to use in a specific context or for a specific type of system. If so, the validity of the referenced design rule within the current context shall be stated.

545. Each design rule is configured in one, and only one, Design Rule Package.

546. The status of a design rule indicates in which state of development it is.

547. Validity of a design rule is only used when referring as e.g. to form aggregates. The validity labels that can be used are defined in the table below.

Validity	**Description**
Mandatory	The rule shall be treated as a norm and is mandatory to use.
Optional	The rule gives good design principles and is recommended for use.
Candidate	The rule is planned for future use in this context. The design rule exist but is not appropriate to use due to reasons like cost, compatibility etc.

Table E.1. Rule validities

548. The lifecycle for a design rule must be coordinated with profiles and standards in the manner, following the NOSWG NISP model

E.5.2.4. Packaging of Rules (Rule Package)

549. Design rules are configured in packages named DRP, Design Rule Package. A DRP may also configure other DRPs, thus creating a hierarchy of packages. A design rule or DRP belongs to one, and only one, DRP.

550. DRPs are defined so that each DRP-structure covers rules that are specific to one particular domain defined for a specific subject area of norms.

551. Dependencies between DRPs shall be defined, and the dependencies shall be minimized. Circular dependencies must not exist. The visibility of design rules configured by a DRP may in addition be limited to the DRP only; default is however that only the DRP exposes the external visibility for a design rule.

552. No design rule shall be part of more than one DRP, if necessary cross-references between DRPs according to the rules for dependencies between DRPs shall be used. Common design rules must for this reason be allocated to higher levels in a DRP hierarchy.

E.5.3. Consequences

553. If the design rule concept is going to be successfully implemented, it is important to understand how they impact the other frameworks and processes used in design. These frameworks and processes also have to be adjusted so it becomes clear as to what is documented where and when.

E.5.3.1. Standards with the use of design rules

554. Standards is often about WHAT but not always about HOW. A vast number of standards are applicable for NNEC, what are applied where, how and together with what, does not always mean that complex system will work. In order to support profiling development when using NISP, Designrules is adopted by NATO as a complementary set of tools for :

- Helping to choose the right standard
- How to apply the standard on a specific problem
- Understanding the relations between different standards
- Applicability in different domains
- Helping with best practice and good patters in order to speed up the development of a profile.

E.5.3.2. Profiling with the use of NAF and Standards and Designrules in the NISP

555. The relations between the NISP and NAF objects in focus. The following picture shows the relations between the NISP objects Profile, Standards and Designrules. For more information about Profile guidance document.

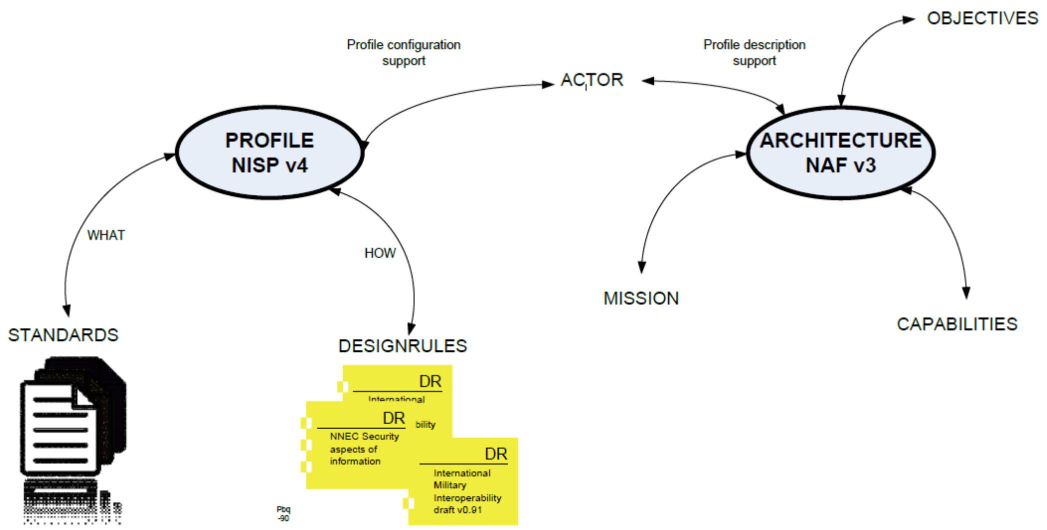

OBJECTIVES
Profile configuration support
ACTOR
Profile description support
PROFILE NISP v4
ARCHITECTURE NAF v3
WHAT
HOW
MISSION
CAPABILITIES
STANDARDS
DESIGNRULES
DR
International
bility
DR
NNEC Security aspects of information
DR
International Military Interoperability draft v0.91
Pbq -90

Figure E.2. Releationship between NISP objects Profiles, standards and Designrules

E.6. REFERENCE ARCHITECTURE - NATIONAL DESIGN RULES

E.6.1. The Swedish Design rules contributions

FMLS Architecture Framework Designrules

LT9O P05-0486 Executive Summary 1.0

Leif Nyberg, JV Network Based Defence, Framework Service Description LT1K P04-0320 Version 7.0 December 2006.

LT1K P05-0074 Overarching Architecture 4.0

LT1K P05-0075 Systems Engineering Vision FMLS 2010 5.0

LT1K P05-0026 - SOA for NBD Principles 3.0

LT1K P05-0507 Architecture Description Framework 2.0

LT1K P06-0025 Integrated Dictionary for FMLS 2010 Technical Systems rev 1.0

FMLS Generic Designrules

LT1K P04-0438 Definition of service Service Registry 3.0

LT1K P05-0235 Definition of service User Registry 2.0

LT1K P05-0446 NERE metadata specs for tech and softw syst 2.0

LT1K P06-0036 SD Provide Report 2.0

LT1K P06-0039 SD Access COP Information 2.0

LT1K P06-0061 Definition of Service SW and Data Distribution 1.0

LT1K P06-0064 Definition of Service Configuration 1.0

LT1K P06-0102 Definition of Service GetRevocation 1.0

LT1K P06-0269 Definition of Service TimeStamp 1.0

LT1K P06-0272 Definition of Service ComBroker 1.0

LT1K P06-0298 D3C 1.0

LT1K P05-0034 Infrastructure Overview 3.0

LT1K P05-0236 Definition of service Organization Registry 2.0

LT1K P05-0557 Design Target Architecture NERE 2.0

LT1K P06-0037 SD Process intelligence 2.0

LT1K P06-0059 Definition of Service Policy 1.0

LT1K P06-0062 Definition of Service Action 1.0

LT1K P06-0091 COPS Information model 1.0

LT1K P06-0134 Definition of Service DNS 1.0

LT1K P06-0270 Definition of Service AccessControl 1.0

LT1K P06-0274 Definition of API data validation 1.0

LT1K P05-0035 Communication Infrastructure Overview 4.0

LT1K P05-0443 NCES Reference Architecture 2.0

LT1K P06-0035 SD Provide Streaming Data 2.0

LT1K P06-0038 SD Support COPS 2.0

LT1K P06-0060 Definition of Service Log 1.0

LT1K P06-0063 Definition of Service Monitoring 1.0

LT1K P06-0095 NCES Management Information and Data models 1.0

LT1K P06-0145 Design Overview 1.0

LT1K P06-0271 Definition of Service NereRegistryAdmin 1.0

LT1K P06-0279 Definition of Service Network Time synchronization 1.0

FMLS Technical Designrules

LT1K P05-0217 - DR Data Incest Prevention 2.0

LT1K P06-0049 DR Risk management 2.0

LT1K P06-0106 Design Rule Mobility 2.0

LT1K P06-0350 DRP Flexibility 1.0

LT1K P05-0547 - DRP Common Operational Picture 2.0

LT1K P06-0050 DR Flexibility 2.0

LT1K P06-0108 DR security aspects of information 1.0

LT1K P06-0351 DRP Interoperability 1.0

LT1K P06-0008 Design Rule Legacy Integration 1.0

LT1K P06-0051 DR Interoperability 2.0

LT1K P06-0321 DR Scalability 1.0

LT1K P06-0352 DRP Security 1.0

E.6.2. Nation x …

This page is intentionally left blank

F. INTERNATIONAL MILITARY INTEROPERABILITY FOR INFORMATION EXCHANGE IN THE NNEC CONTEXT

Summary

556. This design rule describes how military organisations can develop and implement the ability to exchange information and services with military organizations from other nations to become interoperable. It touches on, but does not fully address the problems related to organizational structures and behaviour when multiple organisations collaborate in a federative manor in a mission.

F.1. GENERAL

F.1.1. Unique Identity

557. [An identifier that uniquely identifies the design rule. (Product ID)]

F.1.2. Target Group

558. This design rule targets any military organization that plan or foresee that it will participate in a mission where exchange of information and services with other military organizations is vital.

559. Within these organizations, the intended users are requirement analysts, architects and high-level designers of NNEC compliant systems.

560. This document defines patterns for enabling information exchange between parties in federations, and is to be used by architects designing SIOPs and SIPs according to NISP and the NATO C3 System Architecture Framework [6].

F.1.3. Definitions and abbreviations

CIA	Confidentiality, Integrity and Availability. Aspects which are to be considered when performing security analysis.
COI	Community Of Interest.
Design rule	A standardized, reusable solution to a design problem in a specific context within a problem space that provides value to the user.
ESB	Enterprise Service Bus. An ESB refers to a software architecture construct, implemented by technologies found in a category of middleware infrastructure products usually based on Web services standards that provides foundational services for more complex service-oriented architectures.
IEAT	A concept for Information Exchange Architecture and Technology developed within the frame of Multinational Experiment 5 with Sweden as lead nation.

IEG	Information Exchange Gateway. A technical system which is used to protect information assets. IEG are described in the IEG concept [10].
IEM	An Information Exchange Model (IEM) is a specification of the information which is exchanged between operational nodes. IEMs are used when deciding which information objects are to be exchanged in service interactions.
IER	Information Exchange Requirement, a specification of the required information exchanged between operational nodes which are described in an architecture.
IES	Information Exchange Service, a part of an IEG.
Information Zone	Information Zones is a concept identified and defined [11] to achieve confidentiality with high assurance, for a gathering of information within a defined perimeter, and interactions to its surrounding with a number of services and nodes inside the zone.
IPS	Information Protection Service, a part of an IEG.
NAF	NATO Architectural Framework.
NEC	Network Enabled Capabilities.
NNEC	NATO Network Enabled Capabilities.
NISP	NATO Interoperability Standards and Profiles [8].
NPS	Node Protection Service, a part of an IEG.
Operation	An operation where actors from multiple national system is tasked in a federation of system.
Service	In this context a technical mechanism which allows access to one or more capabilities in order to enable service interaction.
SIOP	Service Interoperability Point. A reference point within an architecture where one or more service interfaces are physically or logically instantiated to allow systems delivering the same service using different protocols to interoperate [6].
SIP	Service Interoperability Profile. A set of attributes that specifies the characteristics of a service interface between interoperable systems in the Networking and Information Infrastructure. A SIP is identified at a SIOP in an architecture system view [6].
SOA	Service Oriented Architecture. An architectural style which aims at a loose coupling of services with operating systems, programming languages and other technologies which underlie applications.

Bibliography

Steering documents

[1] Design Rule Framework, See NATO NISP DR guidance document

References

[2] DR Interoperability Sweden proposal, P06-0051 rev 3.0

[3] IEAT Concept, MNE-5 initiative

[4] Design Rule Flexibility, Sweden P06-0050 (NATO doc ?)

[5] Design Rule Security aspects of information, Sweden P06-0108 (NATO doc ?)

[6] NATO C3 System Architecture Framework, EAPC(AC/322)D(2006)0002-REV1

[7] Federated Governance of Information Sharing Within the Extended Enterprise, AFEI Information Sharing Working Group, Nov 17 2007

[8] NISP Volume 1, Version 3

[9] NATO Architecture Framework (NAF), Version 3. AC/322-D(2007)0048

[10] Guidance Document on the Implementation of Gateways for Information Exchange between NATO and External CIS Communities, AC/322(SC/4)N(2007)0007

[11] Swedish FMLS Security Architecture Overview, http://www.fmv.se/upload/Bilder%20och%20dokument/Vad%20gor%20FMV/Uppdrag/LedsystT/Övergripande%20FMLS-dokument/Generiska%20designdokument/LT1K%20P04-0385%20Security%20Architecture%20Overview%205.0.pdf , 33442/2006 Version 5.0, May 4 2007

[12] NISP Volume 3, Version 3

[13] TACOMS: TACOMS Post 2000 Profile, STANAG 4637

F.2. DESIGN RULE

561. This design rule is developed for use in NATO Interoperability Standards & Profiles (NISP) version 4. It is based on experiences from the Swedish Network Based Defence initiative where it extends the design rule for Interoperability [2] and the IEAT concept developed within the frame of Multinational Experiment 5[3]. The design rule also considers the NATO Information Exchange Gateway (IEG) concept[10].

562. The design rule is applicable for collaborative federations in the coming 2-6 years which means that it covers both existing systems which won't be replaced as well as new systems which are developed and implemented during this time period.

563. The technical scope for the design rule is the highlighted areas of Figure F.1. The design rule does not describe how to achieve interoperability on the Transport/Network level. Furthermore, it does not cover interoperability on the Community of Interest level. However, when design rules for these levels are created, this design rule will be used as the basis for enabling information exchange via services.

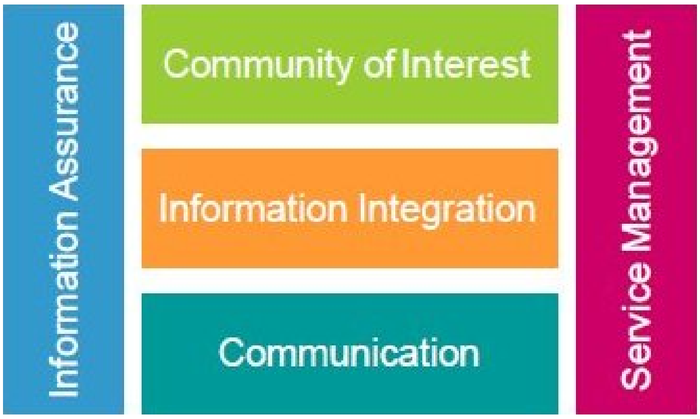


Figure F.1. Simplified NNEC Technical Services framework with design rule scope

F.2.1. Context

F.2.1.1. Introduction

564. The design rule should be used when there is a need for several different military actors to cooperate in a federative manor in order to solve a common mission. The key capabilities that this design rule will help enable are:

- Collaborative planning between multiple actors in a federation
- Collaborative synchronization of execution between multiple actors in a federation
- Collaborative assessment between multiple actors in a federation

565. The design rule does not address the operational activities needed to achieve the above capabilities, nor does it address the Community Of Interest (COI) technical services which supports these activities. Instead the design rule describes a set of principles, technologies and activities needed to create a technical platform which enables information exchange between the actors and can act as a foundation for the COI specific technical services when these are to be developed and deployed.

566. Since the design rule captures knowledge from previous experiences in this area it can save time and money for the involved actors. If the design rule is applied when defining the profile for such a mission, less time will be spent on getting to agreement on which services and underpinning technologies shall be used in the mission.

567. Many of the activities and technologies described in this design rule can also be applied when exchanging information and services with other actors than military organizations. However, there are specific aspects of collaborating with this type of organizations which are not covered by this design rule.

568. A suitable definition of interoperability in this design rule context (i.e. technical context) is: The ability of technical systems and/or organizations using technical systems to operate together by making (necessary) data & information and/or services produced by one system or organization available to the others, in an agreed format.

F.2.1.2. The International Military Federation

569. There are many challenges that have to be overcome in order to make collaborative work and knowledge sharing among the actors in an operation successful. In Section F.2.3 of this design rule mainly addresses the technical aspects of the establishment of federation in which collaborating actors can exchange information. However, organizational, process and legislation aspects must be covered to some extent since all of these needs to be harmonized in order to make the collaboration effective. Therefore, a number of non-technical issues are described in Section F.2.2.

570. The federation, depicted in Figure F.2, is where the collaborating actors provide services which the other actors can consume. To create a federation, the actors need to create a federation agreement which defines the rules of the federation, such as which data formats, information classifications should be used. Rules regarding information ownership and service levels (including quality of service) are also included in the federation agreement.

571. Collaboration in multilateral operations has previously been based on bi-lateral agreements between all participants, but in order to achieve the speed and flexibility needed today, there is a need to establish a baseline federation agreement which can be used as a starting point when creating new missions.

572. Actors which participate in the federation connect networks and systems within their responsibility (i.e. domain) to other actors in order to be able to exchange information. To protect the internal information and control which information is being exchanged one or more Information Exchange Gateways (IEG) are stood up between the federation and the actors' network. In the IEG, one or more service interfaces are physically instantiated. This is referred to as a Service Interoperability Point (SIOP) according to the NATO C3 System Architecture Framework [6].

573. Within an actor's domain there can be one or more networks where information is stored. The decision which internal networks shall be connected is taken by each actor (Federation member) independently of the other actors. In Figure F.2 two example networks are depicted, one federation network which holds information only relevant to the federation and one which is the actors' internal network. In this case, the IEG handles information exchange between these two networks as well as information exchange with other actors IEGs.

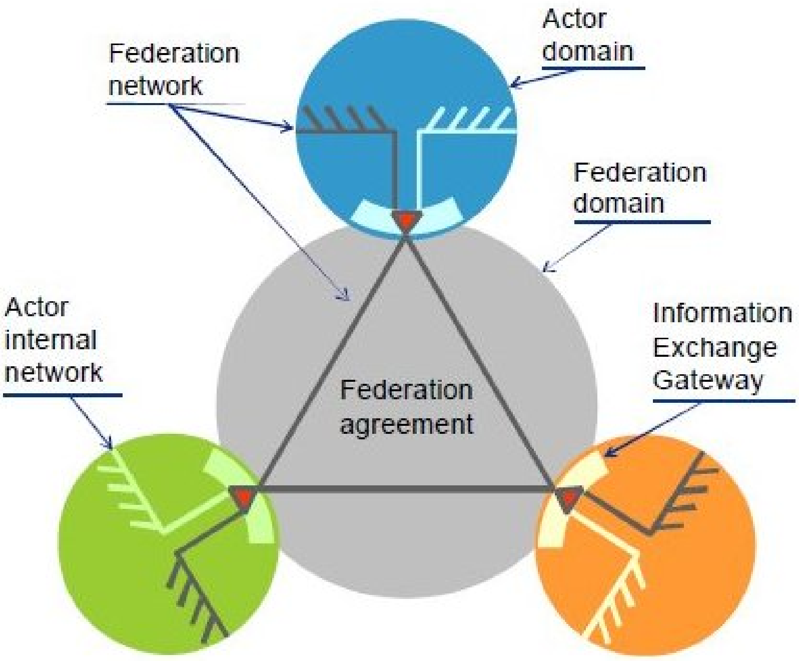


Figure F.2. Federation Overview

574. The remainder of the design rule describes the challenges the actors face and how they can cooperate in order to create a federation to exchange information in a secure manor.

F.2.1.3. Related design rule areas

575. Interoperability is closely linked to the following other design rule areas:

576. **Flexibility**: The requirements on interoperability will change over time. Also, in some situations, very limited time will be available for making the necessary modifications of the system in order to fulfil the new requirements. This means that the organization, security and technical systems need to be very flexible with respect to configuration and modifiability in order to be able to adapt to changing and extended interoperability requirements. For more information, refer to [4].

577. **Information security**: With interoperability follows information security risks that must be handled. The connection of external systems must be done in such a way that the information security of each nation or organisation is not compromised. However information security mechanisms cannot be allowed to be static. In each specific case the need to protect information must be balanced against the possible consequences from not sharing the information. The three aspects of security; confidentiality integrity and availability, must all be considered.

F.2.2. Problem

578. There are several challenges to the effort of creating a federation for collaboration between military partners, both related to technology, but also related to how organizations, humans and legislation systems work.

579. This chapter summarizes the basic requirements for the federation and identifies the challenges which must be overcome in order to establish the federation. The issues identified for these challenges are given an answer to in Section F.2.3.

Basic requirements for information exchange

580. The intent of this section is to identify a few of the most elementary (information exchange) requirements which are set on all international military federations. This is not a complete list, but these requirements acts as a driver for identifying the basic set of technologies needed in a federation.

[IER 1] People from the different organisational actors SHALL be able to communicate with each other using voice or text communication.

[IER 2] It SHALL be possible to discover and retrieve information (i.e. search) provided to the federation by different actors.

Challenges based on international agreements and regulations

581. Information and services exchange between nations and organisations (e.g. unclassified, restricted, secret and top secret classification) is based on government agreement between nations and organisations. Qualified information and services exchange can only take place if such agreement exists. To achieve this agreement is a lengthy process that often takes many months to finalize. It has also been proven complicated to negotiate and sign such an agreement between more than two nations and organisations at a time (multilateral). Nations are willing to share more information and services with some parties and less with others. This creates complicated situations during multilateral operations.

[Issue_1] How can a common, agreed description for analyzing and describing international military interoperability be created?

Challenges based on national law, national integrity and regulations

582. Differing laws, rules and regulations together with different cultures regarding information sharing are likely to impact willingness to share information and slow down process of getting agreements on what to share.

583. Parties participating in a multilateral operation are likely to have different requirements and priorities which will imply different scope and granularity of information exchange for each party. The parties will be required to protect their national integrity while sharing information

with the other parties. By this, it is likely that the parties wish to get access to more information and services than they are willing to provide themselves. It is also so that the parties will need to limit the possibilities for others to control how and what information is provided.

[Issue_2] How can the impact of national laws and regulations in coming to agreement of what information to share be minimized in order to support the requirements of flexibility and ability to change?

[Issue_3] How can parties participating in a multilateral operation protect their national integrity by using mechanisms to protect internal information and be able to control what information is released to others?

[Issue_4] How can the parties in a multilateral operation jointly come to agreement of what information shall be exchanged, how it shall be exchanged and how it shall be handled by receiving parties?

Challenges based on interpretation of information content

584. Semantic differences, i.e. differences in languages and the meaning of words and expressions, are likely to be an issue when exchanging information. If the collaborating parties cannot understand the information being communicated, the information will not be of any use and the trust of accessible information will be challenged. There is a need for the parties to eventually meet in a combined opinion, a common and agreed set of descriptions in order to reach wanted effects.

585. In order to solve the semantic challenge there is a need to understand the content of information and services exchanged between different systems/actors to be able to come to an agreement of the meaning of the information. However, the increasing requirements of the ability to rapidly change directions of the flow of information, as well as the actual content, means that the work with defining models and requirements for information exchange must be done continuously and during the whole lifecycle of an operation.

[Issue_5] How can the parties in a multilateral operation agree on what information shall be exchanged?

[Issue_6] How can differences in semantics and information models be handled in order to minimize the risk of the parties not understanding each other?

[Issue_7] How can it be ensured that the work with understanding others semantics and information models is done in all stages of the development lifecycle?

Challenges based on technical issues

586. Architecture and technical implementations of information systems will be different in most of the cases. The complete technical system will probably not be homogenous, rather a federation of heterogeneous systems and therefore hard to govern and manage.

587. Agreeing on standards, formats and mechanisms for information exchange is a critical success factor, however the sovereignty of the parties will increase the complexity of this task since there is no governing organ that can make the decisions.

588. A common understanding and agreement on the architecture and design for the federation is vital in order to succeed with agreeing on how information shall be exchanged. A major challenge in this perspective is that the maturity of using architecture and design as governing tools is likely to vary greatly among collaborating parties, thus slowing down the agreement process.

589. Since each actor has huge amounts of data of various kinds within their internal networks there is a need to have the means to organize and prioritize what to share. Also, when information has been shared within the federation, there must be mechanisms to be able to verify the authenticity, track usage of and prevent that the information is used by actors which are not meant to use it.

[Issue_8] Which architecture can enable governance and structure to mechanisms for information exchange between heterogeneous systems?

[Issue_9] Which standards, formats and mechanisms for information exchange should be used?

[Issue_10] What does a common architecture description framework for multilateral operations contain?

[Issue_11] What mechanisms shall be used in order to control what information to make available to partners in an international military operation?

[Issue_12] What mechanisms can be used to maintain information security and system safety, e.g. weapon safety, when external systems are connected to a nation’s internal network?

Challenges based on culture, lack of trust and organizational issues

590. Even if we have solved “challenges based on international agreements and regulations” we will still most likely hesitate to share information since the organizational culture does not foster incentives to share information[7]. This is understandable, but not very efficient from an operational perspective. We have to overcome these limitations and see the goal of the operations as more important than the individual organizations ego.

591. Today’s military organizations are experienced and usually organized around various stovepipe principles. This is a convenient, straight forward way of defining requirements, responsibilities and timetables for implementing new and enhanced systems. Operations were information is expected to be exchanged between both organizations and technical systems will set new requirements on the procurement process, working methods and the organizations working those issues.

[Issue_13] Data are not generally created to support enterprise needs. There are typically technical and political boundaries that inhibit this. To "line" applications development organizations, enterprise-level requirements for data are typically viewed as "external", as their direct customers, and typically the sponsor of the application, is not rewarded for serving the greater good, but for locally optimizing the performance of their organization[7].

F.2.3. Solution

F.2.3.1. Architecture for interoperability

592. The most important instrument in resolving the issue of creating a description for analyzing and describing international military interoperability as described in [Issue_1] is to create an architecture. This design rule outlines an architecture which provides the means to create a foundation for the federation in which information exchange among parties can take place.

593. The architecture is described by:

- Governing aspects (design principles and rules) used to explain and develop architectural principles and structures in important areas of the architecture.
- Common terminology & definitions.
- Structure. How systems, aspects and terminology/definitions are organized and grouped.
- Systems in terms of mission and/or technical systems.
- Services which describe how systems interact.

594. It is absolutely vital that the architecture addresses both operational and technical aspects so that there is a clear description of what purpose the technical implementation has [Principle_4].

F.2.3.1.1. Service Oriented Architecture

595. The Architecture outlined in this Design rule is Service Oriented [Principle_5]. The aim of this is to achieve a loose coupling of services with underlying systems, whether it is mission or technical systems. So, instead of describing interaction directly between systems, the systems use services to interact with each other. By specifying a contract for information exchange, a service definition [Principle_6], the inside of a system can be replaced or modified without having to change other systems which interacts with it. Thereby the issue of enabling information exchange between heterogeneous systems [Issue_8] is resolved.

596. Services used or provided by technical systems should as far as possible be expressed in a common way and contain formal descriptions suitable for IT processing.

597. The Service description shall contain:

- The allowed service protocols (process) to be used for information exchange.
- The interfaces (or message types) that are used to exchange information between a service consumer and a service producer.
- The definition of the data types that are used in the interfaces (messages) and therefore are in the information exchange model.
- The properties that consumers can use to distinguish between different implementations of a service.

598. To enable systems to find and connect to each other, information about services shall be published and accessible for the collaborating parties' IT systems.

F.2.3.1.2. Architecture description framework

599. In order for all parties to obtain a common "language" on how to describe their systems and the services they bring to the federation this design rule also covers an architecture description framework. The architecture description framework does not describe the architecture itself, but rather guides how the architecture shall be structured and what it should describe.

600. The current valid description framework within NATO is the NATO Architectural Framework (NAF) version 3[9] which provide the rules, guidance, and product descriptions for developing, presenting and communicating architectures which includes both operational aspects as well as technical aspects [Principle_4].

601. In the Framework, there are seven major perspectives (i.e., views) that logically combine to describe the architecture of an enterprise. These are the NATO All View (NAV), NATO Capability View (NCV), NATO Programme View (NPV), NATO Operational View (NOV), NATO Systems View (NSV), NATO Service-Oriented View (NSOV) and NATO Technical View (NTV). Each of the seven views depicts certain architecture attributes. Some attributes bridge several views and provide integrity, coherence, and consistency to architecture descriptions.

602. To support the creation of views and make sure they are consistent, NAF v3 defines a metamodel. The NATO Architecture Framework Metamodel (NMM) defines the relationships between the different components of the framework. It defines the architectural objects and components that are permitted in NAF v3 views and their relationships with each other.

603. There are certain views which are more important when designing architectures for multinational operations where interoperability is in focus [Issue_10]:

604. **NATO All-Views (NAV)** which capture aspects which overarch all other views. These views set the scope and context of the architecture, such as goals and vision, scenario and environmental conditions as well as time.

605. **NATO Capability View (NCV)** which explain what capabilities are needed in order to fulfil the strategic intent for the mission. Specifically, capabilities related to interaction between actors are important to identify in these views. If produced correctly, these views can already say a lot of which services are needed to fulfil the business needs. In particular, the NCV-2, Capability Taxonomy and NCV-7, Capability to Services Mapping views are important.

606. **NATO Operational View (NOV)** which is a description of the tasks and activities, operational elements, and information exchanges required to accomplish NATO missions. To design for interoperability all of these views do not have to be complete, but it is important to know which operational nodes exist and how they interact (NOV-2). Also, the information model defined in the NOV-7 view is important, especially for such information for which there are no or unclear standards to rely on. When going into more details of the architecture, the requirements on information exchange (NOV-3) are necessary to understand.

607. Currently, the operational views in NAF does not fully support modelling of services. The authors of this design rule recommends that future versions of NAF are complemented with the capabilities of using services to describe interaction between operational nodes instead of needlines.

608. **NATO Service-Oriented View (NSOV)** focuses strictly on identifying and describing services. The view also supports the description of service taxonomies, service orchestrations and a mapping of services to operational activities. The service description (NSOV-2) is a key component of a Service Oriented Architecture [Principle_6]. It is used to detach the functionality provided by a system (or services provided by an organizational unit) from the actual system. A service description includes information on how to interact with the service, what requirements a system must fulfil if it implements the service and what information model the services uses. Within NSOV-2 a SIOP can be depicted as a higher-level service interface. The detailed technical specification of a SIOP is contained within a Service Interoperability Profile (SIP). SIPs are addressed in NTV-1 Technical Standards Profile.

609. In the **NATO Systems View (NSV)**, the NSV-1 view is the most important since it describes how the different systems interact to fulfil the operational needs. The system descriptions should be kept on a black-box level, i.e. it is not relevant to describe the internals of the systems.

F.2.3.2. Key Principles

Sovereignty of collaborating parties

610. The sovereignty of the collaborating parties is fundamental; organizational right to use organic information systems and working methodology with various support tools shall in all situations be respected. The decision to publish information to the federation is the responsibility, and right, of each actor. Information content and possible restrictions will always be any actor's sovereign decision.

[Principle_1] Each collaborating party decides which information to publish into the federation.

View on information

611. Information shall be regarded as an operations wide asset and not be exclusive to any single operational area or function, with exceptions for agreed confidentiality. Collaborating parties should avoid over-classification of information. Information should be provided as a published service.

[Principle_2] Information published into the arena is available to all parties, if no restrictions have been agreed.

Agreements for Information Exchange

612. Agreements to facilitate Information Exchange shall exist for the operation and between the collaborating parties. The agreements includes which information is required to be exchanged, models for how exchanged information shall be structured, how information can be translated between models and the format of the exchanged information.

[Principle_3] Requirements, models, translations and format for information exchange in the arena are regulated by agreements.

Architecture

613. Establishment of a consistent and understandable architecture should be supported by a common terminology and a common architecture description framework. In order to ensure that the technical architecture fully supports the operational needs, there is a need for a joint architecture.

[Principle_4] The operational and technical aspects of the architecture are described using a common description framework.

614. The architecture of the federation must support exchange of information between many heterogeneous systems in order to fit all actors’ needs. A Service Oriented Architecture (SOA) achieves this by separating information exchange capabilities from business logic and system specific implementations.

[Principle_5] The technical architecture for information exchange follows the tenets of the Service Oriented Architecture concept.

615. OASIS (organization) defines Service as "a mechanism to enable access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description."

[Principle_6] Technical services for information exchange are specified in a service description.

Technology

616. Open and accepted international standards, both civilian and military should be used. Bespoke and proprietary standards shall only be considered when it delivers significant higher value.

[Principle_7] Technical services for information exchange uses open standards whenever possible.

Security

617. To achieve information exchange in a secure way using services, a set of principles which guides the use of security functions is needed:

[Principle_8] Service consumers and service providers use a common methods for authentication and authorization of users and services.

[Principle_9] There is a common method to obtain integrity by which a service consumer can check that the data sent from another part is not changed by a third part.

[Principle_10] There is a common method to guarantee the confidentiality of the information exchanged. This means that it is possible to prevent outsiders from getting access to the information that is exchanged.

618. It is important to remember that these principles only apply between the borders of the actors in the federation, not end-to-end between users. The reason for this is that it is very hard and cost driving to govern how security mechanisms shall be implemented within an actor.

F.2.3.3. The information aspect

619. In order to meet operational needs for information exchange and to build a federation, supported by technical systems serving as operational nodes, a number of areas must be addressed:

- Information Exchange Requirement specifications
- Information Exchange Models within collaboration areas and their relation to international standards, domain Community Of Interest (COI) models, semantic structures etc
- Translation specifications and translation mechanisms
- Specification of information exchange mechanisms in the federation e.g. common data management services, mediation services and bridges to external systems

620. Documenting the above according to [Principle_3] address issues [Issue_1], [Issue_2], [Issue_4], [Issue_5], [Issue_6] and [Issue_9] by creating agreements of what information is to be exchanged, how to interpret the information and which mechanisms are utilised to enable the information exchange.

621. This chapter covers the definition aspect of information, technologies which implement these definitions, like for example mediation, are covered in Section F.2.3.

F.2.3.3.1. Information Exchange Requirements

622. An Information Exchange Requirement (IER) is a specification of the required information exchanged between operational nodes. IERs are identified in the business modelling process and specify the elements of the user information used in support of a particular activity. The specification is done according to the NOV-3 view of NAF[9].

F.2.3.3.2. Information Exchange Models

623. An Information Exchange Model (IEM) is a specification of the information which are exchanged between operational nodes. IEMs are used when deciding which information objects are to be exchanged in service interactions. The specification is done according to the NOV-7 view of NAF[9].

624. An IEM is constructed top-down based on model elements from other existing Information Models e.g. standards as well as bottom-up based on information requirements specifications from Operational Concepts and Requirements Implications (OCRI)[8].

625. When designing Information Exchange Models several different approaches exist:

- Model based, e.g. JC3IEDM, ISO19100 series
- Ontology based e.g. Semantic web
- Message based e.g. ADatP-3

626. Given the timeframe for this design rule, a model based approach is the best approach considering what the technology can handle and results from ongoing modelling work. The ontology based approach can be adopted at a later stage when the technology and methods are more mature while the message based approach is to be avoided if possible since it cannot handle the complexity of integrated models.

F.2.3.3.3. Translations

627. There may be a large number of translations between two information models. Each translation is based on thorough analysis and is documented in a translation specification together with estimates of information loss.

628. There are different approaches to making translations between the models:

- Manual model mapping, that is when two models are compared and decision are made at element level on how to map and/or translate to the other models. This is often the case when the models to compare are documented according to different standards regarding ontological metadata notation, modelling style etc.

- Rule based model mapping that is when two models are compared and mapped to each other based on formalized rules. Automated translation has the potential to be applied in runtime, thus increasing flexibility in information exchange.

629. Technologies which perform automated translation between information models is not yet available to any greater extent. Therefore, the translation technologies described in Section F.2.3.5.6 focuses on supporting translation rules which are based on manual model mappings.

F.2.3.3.4. Information Exchange Objects

630. An information object is a set of data elements that are contained and treated as one unit. The content structure may vary in complexity from the simplest form with a number of data elements and an identifier to complex data structures and large quantities of data elements. Examples of information objects are documents, messages and data sets such as geographical data sets.

631. Information objects are created, processed, stored and moved/exchanged via services. An information exchange object is a standardised view, or an excerpt from, an information exchange model which from a technical point of view is suitable to exchange as a coherent set. Thus information exchange objects is a subset of all information objects which are meant to be exchanged via services.

F.2.3.3.5. Services and the information aspect

632. In a Service Oriented Architecture [Principle_5], information objects are created, processed, stored and moved/exchanged via services. Therefore it is important to understand the architectural relationship between services and information. I.e. how are services and information specified in order to enable the implementation of a service oriented architecture.

633. As depicted in Figure F.3, a service has operations. They are used for specification of how a consumer can interact with the service, for example create, read, update, delete. An operation requires one or more information objects to be exchanged between the consumer and provider, for example a message or a document. These exchange objects are excerpts from an information exchange model.

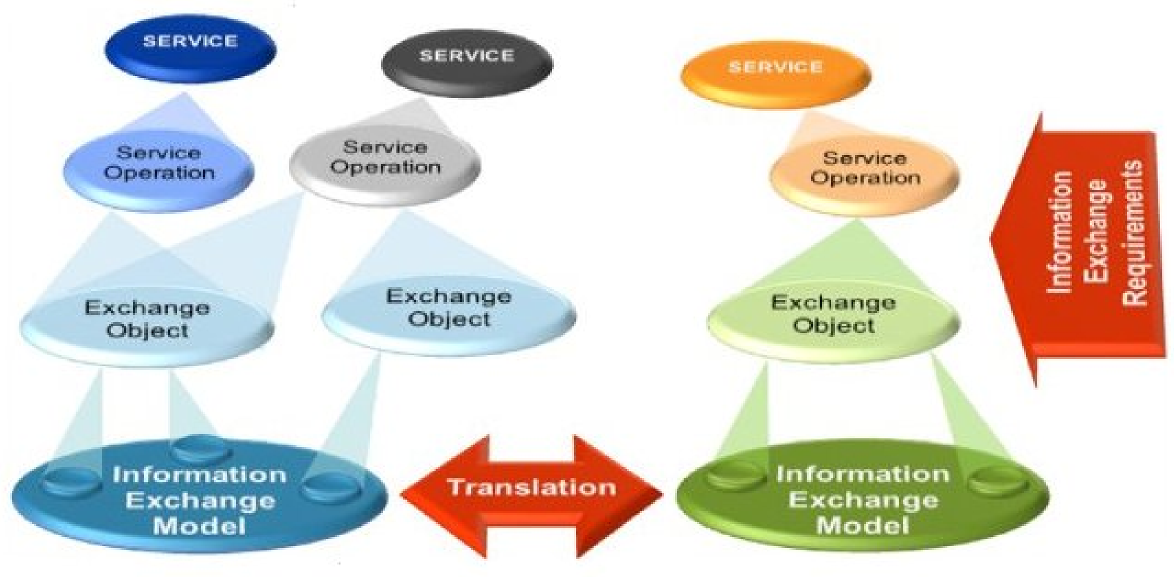


Figure F.3. Services and the information aspect

634. Translations are use to describe how information exchange models relate to each other and can also be used by mechanisms to automatically translate exchange objects from different information models. Information exchange requirements are set on service operations and exchange objects, i.e. what functionality shall the service provide and what information shall it handle.

F.2.3.4. The security aspect

635. When determining appropriate security solutions for a federation it is of outmost importance to analyse the information which needs to be assured. This is important in order to avoid a “too secure” solution, thus introducing higher costs and more difficult procedures than needed. The flexibility which is introduced by the NNEC concept requires a constant analysis of the need for information confidentiality, integrity and availability (CIA). Also, time needs to be considered in these analyses, i.e. how long does the information need to be protected.

636. This design rule does not cover how to perform CIA analyses, but it is certain that there is a need to be able to handle different levels of security in the federation. A set of scenarios has been defined in the IEG concept[10] which are used in this design rule to handle difference in security levels.

The Information Exchange Gateway Concept

637. Information Exchange Gateways (IEGs) are used to protect information assets of the participants in the federation. Since each participant provides an IEG to protect their assets there is a need to standardise the services and the architecture of IEGs in order to enable sharing of IEG components between the participants and use of commercially available technology. The NATO IEG concept[10] describes that each IEG has three major services:

638. “The first is the Node Protection Service (NPS). The NPS provides protection to the infrastructure; its purpose is to protect the physical assets of the “node” or nation being protected by the IEG.”

639. “The second major component/service is the Information Protection Service (IPS). NATO and each nation are responsible for protecting the flow of information out of its area (node or network). The mechanisms used to protect the information flow must satisfy the organization (nation or NATO) that the IEG is protecting.”

640. “The third major component/service is the Information Exchange Service (IES). The IEG must facilitate the flow of information between the protected node/network and the external organizations that are authorized (by the Information Protection Service).”

641. Together these services provide the solution to issues [Issue_3], [Issue_11] and [Issue_12]. More details on the implementation of IEGs can be found in Section F.2.3.5.7.

Information zones

642. Information Zones is a concept identified and defined to achieve confidentiality with high assurance, for a gathering of information within a defined perimeter, and interactions to its surrounding with a number of services and nodes inside the zone. The concept gives the advantage to separate assurance on security mechanisms to meet external and internal threats.

643. In a federative approach such as the one described in this design rule, each federation participant (actor) is to be considered as (at least) one information zone. The reason for this is that there is a clear responsibility for information and information management within each actor. At the border of the information zones there are Information Exchange Gateways (IEG) which protects the information within the zone and allows controlled sharing of information between information zones. See Figure F.4.

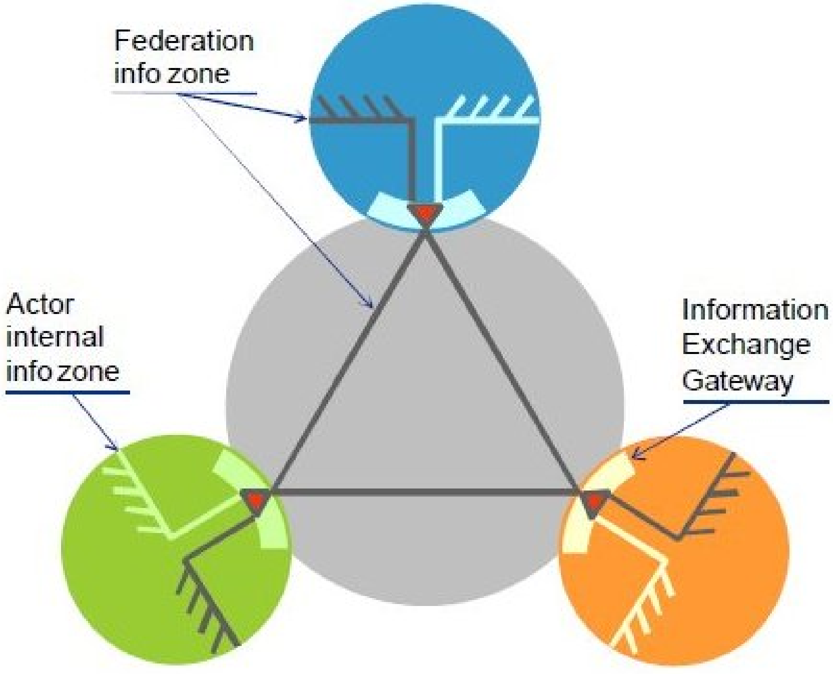


Figure F.4. Informationzones in the federation

644. The information classification level in each zone will differ and therefore the information assurance level needs to be adjusted accordingly. I.e. the more sensitive information within a zone, the more protection and dissemination control is needed.

645. By basing the security on information zones with boundary protection and controlled information flow and access to the zone, it is made easier to achieve high assurance since only a few mechanisms, i.e. the IEG, needs to be inspected/evaluated to meet the security requirement.

646. In the federation there may be several information zones depending on the classification of exchanged information. However, the number of information zones should be kept to a minimum in order to avoid unnecessary costs and complexity for implementation and maintenance of the federation.

F.2.3.5. Technology and profiles

647. As mentioned in Section F.2.1.2, there is “a need to establish a baseline federation agreement which can be used as a starting point when creating new missions”. The technology described in this chapter supports the creation of such an agreement by addressing [Issue_9] > “Which standards, formats and mechanisms for information exchange should be used?”

648. In other terms, the standards, formats and mechanisms defined in this chapter shall serve as the baseline for an international military federation.

649. There are two basic user requirements defined in Section F.2.2 which acts as drivers for the technology defined in this chapter. These requirements are:

[IER 1] People from the different organisational actors SHALL be able to communicate with each other using voice or text communication.

[IER 2] It SHALL be possible to discover and retrieve information (i.e. search) provided to the federation by different actors.

650. To be able to fulfil these requirements, a set of technical capabilities are needed. First of all, there must be network (IP) connectivity between the actors in the federation; however this is not covered by this design rule. Once network connectivity is established, the technical systems of the actors need to be able to publish and find the services which are to be used. Of course, all communication in the federation network must be secured by relevant security mechanisms.

651. In order to fulfil [IER 1], users first need to be able to find each other and once they have done that they can start collaborating.

652. To fulfil [IER 2] the Information Discovery Services are used to find relevant information. To retrieve the information, Messaging Services can be used. In some cases the information models used by the different actors does not match and then the Translation Services are used to translate the content.

653. Lastly, it is important for the actors in the federation to know the status of the services in the federation, especially if there are mission critical services which are provided by other actors.

654. The following chapters describe the above in more detail giving advice how to implement the technologies needed to provide these services.

F.2.3.5.1. Discovery services

Service Discovery Services

655. The Service registry enables the technical systems to discover each other. The service registry is a vital part needed for enabling the loose coupling between systems since it provides functionality for the systems to find each other , with such registry the relationships between the systems does not need to be hard coded into the systems. This means that it will be easy to add or remove participants and services from the federation.

656. The Service registry SHALL be implemented using UDDI v3 according to NISP[12]. In order to achieve high availability and allow each participant to be able to publish services, the Service registry shall be implemented using a replication pattern. I.e. the service registry is replicated between all participants in the federation.

657. The Service registry SHALL include the following information (metadata):

Service provider

- Unique id, Name, Description

Service type

- Unique id, Name, Description, Version

Service instance

- Unique id, Name, Description,Service interfaces (bindings e.g. WSDL) and applicable security mechanisms, Endpoint (e.g. URL), Owner - both service provider and human user owning the service, Security Classification - UNCLASS, RESTRICTED etc

Information Discovery Services

658. Each actor in a federation holds information which might be relevant to other actors. Therefore it is of outmost importance that there are mechanisms to discover information across actors. These mechanisms have to include the capability for an actor to decide which information shall be available to others according to [Principle_1] and [Principle_2].

659. There are mainly two ways of making the information discovery happen. One is to copy information between actors and let each actor make the information searchable, but this is not very efficient since it requires a lot of bandwidth and makes it hard to keep track of which information has been copied.

660. The other way of enabling information discovery is to use a federated search pattern where each actor provides a search interface to its information. This is much more efficient from a data distribution point of view, but requires that all actors come to agreement on the search interface. There are initiatives ongoing to standardise the ability to perform federated search, the most prominent one is the OpenSearch initiative[1]. Even though OpenSearch is not a formal standard it is well on its way to be adopted by many of the major tool vendors.

661. In either case, the actors in the federation must implement search engines which can index information (if the have any) and search clients which can access the search engines. A search client is in most cases an ordinary web browser, but can also be a more complex application if there are specific needs.

F.2.3.5.2. Repository Services

Metadata Registry Services

662. A metadata registry is a database which contains information about information which is useful for enabling information discovery. For example, search engines create metadata registries when they index content. But there are also other applications for metadata registries, like when an actor has sensitive information which needs to be able to be discovered. Say that there is a database which contains classified analyses of some sort. The analyses are of very good quality and can be of use to many, but it is impossible to publish them to everyone in the

[1] http://www.opensearch.org/

federation. So in order to make other actors aware that the analysis exists, unclassified analysis metadata, like what the analysis looks at and who has done it, can be published in a metadata repository. Now the other actors can discover that there is an analysis and contact the author to get approval for getting the contents.

663. To be able to store the metadata, the NATO Discovery Metadata Specification (NDMS) SHALL be used. This specification is based on the international standard ISO 15836 the Dublin Core (DC) Metadata Element Set.

F.2.3.5.3. Directory Services

Enterprise Directory Services

664. Sharing information about users is key to a federation since it enables people to find each other. The user directory holds information which enables authentication of users by certificates and public keys, authorization of users by roles and discovery of users by contact information which enables collaboration.

665. Each actor in the network shall provide information about the users that represents them. However, it is preferable if the federation has one point of access to all user directories. Therefore, the implementation of user directories in a federation shall follow the federated database pattern. This means that each actor provides their own database, but one actor provides a single entry point to all databases.

666. For the user registry LDAP shall be used according to NISP[12]. Products which can provide the single entry point to multiple LDAP databases are often referred to as Virtual LDAPs.

F.2.3.5.4. Collaboration Services

Audio based conference service

667. For voice communications standards SHALL be applied as according to TACOMS[13]. Streaming voice and video communication cannot be handled by the IEGs, TACOMS describes how to implement this functionality without the use of IEGs.

F.2.3.5.5. Messaging Services

Server-to-server e-mail messaging service

668. E-mail has become one of the most important applications for any business or organization of today. The main challenge for using e-mail in a federation is to be able to control that no classified information is embedded or attached to e-mails going out from an actor and protecting the actors from malicious software, such as viruses. This means that the IEG needs to be able to scan and filter incoming and outgoing messages.

669. Extra care needs to be taken for outgoing information where confidential information can be hidden in document history and inside images. Therefore, only text-based attachments (like OpenDocument Format or Office Open XML, see NISP[12]) without inserted code or images shall be allowed through the IEG.

670. It is also vital to have a manual inspection capability in the IEG to be able to assess the degree of confidentiality of the e-mail messages leaving an actor.

671. As described by NISP[12], SMTP according to RFC 2821 and others SHALL be used for e-mail. To secure communication between SMTP agents, TLS according to RFC 3207, SHOULD be used.

Instant messaging service

672. For instant messaging XMPP (IETF RFC3920:2004 -3923:2004) SHALL be used according to NISP[12]. XMPP is an XML based publish/subscribe protocol which is used by most of the dominant tool vendors. Using XML enables possibility for inspection and control of messages in IEGs which is very important in a federation.

673. There is one important aspect of XMPP which is not covered by the current standard specification; there is no security tagging options available which is needed when messages shall be passed between information zones with different security classifications. So if this is required a custom extension to XMPP needs to be defined.

674. Another thing which must be considered in a federation is routing of messages. Currently there are no XMPP servers which support routing of XMPP messages. This consequence of not being able to route messages is that the IEG has to be implemented as a transparent proxy, i.e. the systems on the outside of the IEG need to know about the systems on the inside. Even though the IEG can be used for inspection and filtering of messages in this case; it is not always a preferred solution from a security perspective. So, if the security requirements say that the IEG needs to act as a non-transparent proxy, the XMPP server needs to be modified to be able to act as an XMPP server and be able to route messages between XMPP domains.

Message passing service

675. In order to achieve an efficient exchange of information between the actors in a federation there is a need to be able to route and distribute messages. This type of capability is often included in the Enterprise Service Bus (ESB) concept.

676. An ESB refers to a software architecture construct, implemented by technologies found in a category of middleware infrastructure products usually based on Web services standards that provides foundational services for more complex service-oriented architectures.

677. An ESB generally provides an abstraction layer on top of an implementation of an Enterprise Messaging System which allows integration architects to exploit the value of messaging without writing code.

678. The ESB shall enable endpoints to interact in their native interaction modes through the bus. It shall support a variety of endpoint protocols and interaction styles. These interaction patterns are the least which shall be supported:

- Request/response: Handles request/response-style interactions between endpoints. The ESB is based on a messaging model, so a request/response interaction is handled by two related one-way message flows -- one for the request and one for the response.
- Request/multi-response: A variant of the above, where more than one response can be sent. Is often referred to as a subscription pattern.
- Event propagation: Events may be anonymously distributed to an ESB-managed list of interested parties. Services may be able to add themselves to the list.

679. When passing messages in the above patterns, the ESB SHALL be able to perform the following:

- Route: Changes the route of a message, selecting among service providers that support the requester's intent. Selection criteria can include message content and context, as well as the targets' capabilities.
- Distribute: Distributes the message to a set of interested parties and is usually driven by the subscribers' interest profiles.

680. The ESB SHALL be able to handle the following formats and protocols:

- SOAP over HTTP for Web Services
- JMS for Java messages
- XMPP for Instant messaging and XML based Publish subscribe messaging

681. When implementing the ESB concept in federations there are some things which must be considered. First, the products which realize the messaging and mediation capabilities needs to be the same everywhere since there are very small chances of realizing integration between two different products due to a lack of standardization. This means that the federation agreement must include which product to use.

682. Secondly, the management of rules for transformation of messages needs to be considered. ESB and messaging products are often built for central management of transformation rules, thus enabling a better control over the messaging capabilities in an enterprise. However, this can be problematic in a federative approach since all actors need to agree on the transformation rules or appoint one actor which has the authority to manage these.

F.2.3.5.6. Mediation Services

Translation Services

683. Translation is about manipulating messages in-flight between a service provider and a consumer (requests or events). This means that messages dispatched by a requester are transformed into messages understood by a slightly incompatible provider selected from a set of potential endpoints.

684. Translation services are often considered being a part of the ESB concept.

685. The patterns which translation products SHALL be able to handle are:

- **Protocol switch**: Enables service requesters to dispatch their messages using a variety of interaction protocols or APIs, such as SOAP/HTTP and JMS. Transcodes requests into the targeted service provider's format. Can be applied at the requester or the provider end of an interaction, at both ends, or anywhere in between.

- **Transform**: Translates the message payload (content) from the requester's schema to the provider's schema. This may include enveloping, de-enveloping, or encryption.

- **Enrich**: Augments the message payload by adding information from external data sources, such as customization parameters defined by the mediation, or from database queries.

- **Correlate**: Derives complex events from message or event streams. Includes rules for pattern identification and rules that react to pattern discovery, for example, by generating a complex event derived from content of the triggering event stream.

686. Also see Section F.2.3.5.5 for details in ESB implementation.

F.2.3.5.7. Information Assurance Services

687. As a minimum baseline for IEGs in a federation, the following shall be implemented in order to fulfil [Principle_8], [Principle_9] and [Principle_10]:

688. The IEGs shall include a Information Protection Service (IPS). This shall provide the following services:

- Authentication to verify the identity of users and systems sending/receiving data

- Authorization to verify rights for users and systems to send/receive data

- Content encryption/decryption capabilities to assure confidentiality and integrity of the data

- Information dissemination control to be able to control which data is passed through the IEG.

689. To be able to inspect the data flowing through the IEG, the data must be unencrypted. The IEG can send and receive encrypted data, but encrypted data must be decrypted by the IEG before it can be inspected and decrypted again for further transport.

690. The Information Exchange Service (IES) which the IEG shall be able to handle is described in the other technology sections of Section F.2.3.5.

691. The requirements for Node Protection Service (NPS) is not determined by this design rule, however some type of node protection is always needed. Since this design rule does not cover the communication layer, there is a need to create a design rule which describes this.

F.2.3.5.8. Service Management Services

692. Service management can be divided into managing, where the technical systems and services are being controlled, and monitoring where information regarding the status of the technical systems and services are shared.

693. In a federation, the participants may be able to managed systems and services provided by other participants, but this is unlikely due to information responsibility of organizations. I.e. a participant which is responsible for the information within its information zone will not let another actor have administrative privileges to the system where this information resides.

694. However, sharing monitoring information between the participants is essential if the Service Level Agreements (SLAs) shall be fulfilled. These SLAs are included in the agreements for information exchange as specified by [Principle_3].

695. Monitoring information is to be provided using the Simple Network Management Protocol version 3 (SNMP v3) standard according to NISP[12]. Using a non-XML based format for monitoring, like SNMP, will require a special filtering engine in the IEG IPS (see chapter Section F.2.3.5.7).

696. It is important to set the monitoring scope properly when implementing the monitoring solution in order to avoid dissemination of to much information into the federation. Therefore, monitoring information SHALL only be provided regarding the services which are provided by an actor. Important metrics to provide monitoring information about are:

- Availability of services, both past, current and future (planned outages)
- Performance in the form of response times and throughput
- Capacity, like for example maximum number of users or used storage space

F.2.3.6. Summary

697. To summarize, Figure F.5 depicts all the technologies mentioned in the chapters above. Together these technologies provide the foundation for secure information exchange in a multilateral collaboration federation in the NNEC context.

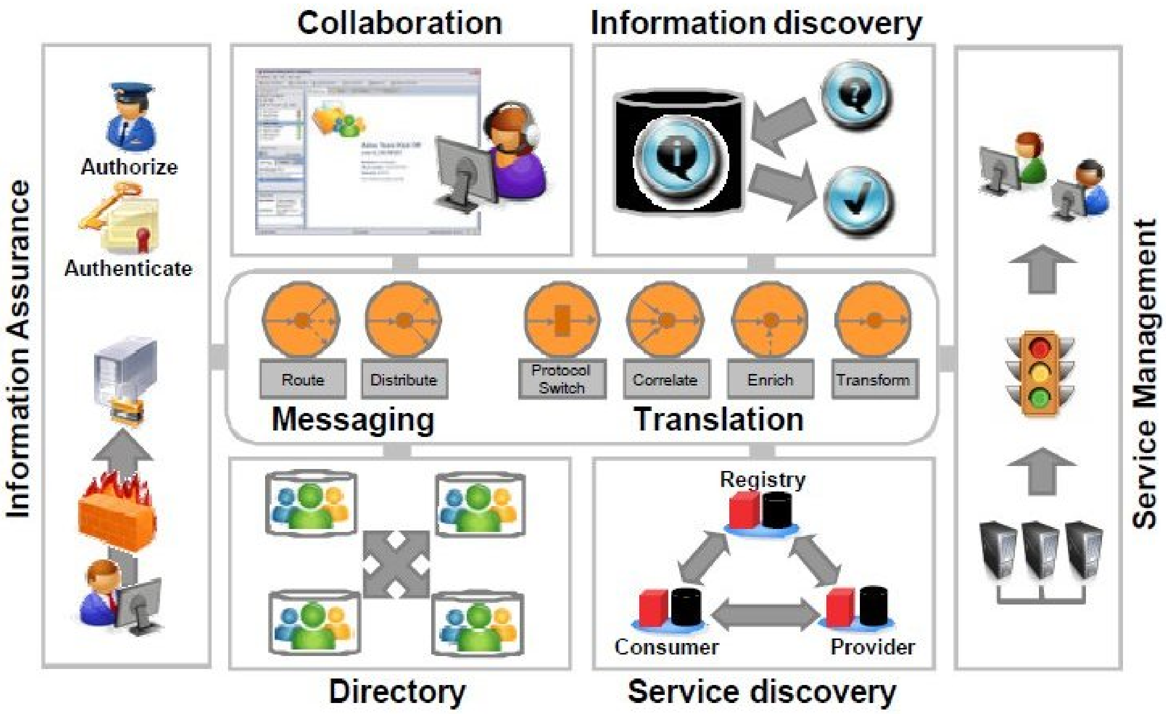


Figure F.5. Technology Overview

F.2.4. Rejected solutions

F.3. MOTIVATION

698. The NATO Network Enabled Capability (NNEC) Feasibility Study[2] highlights that “at their meeting in November 2002, the NATO C3 Board (NC3B) agreed that there was a need to develop a NATO concept to adapt national initiatives such as the U.S. Network Centric Warfare (NCW) and the U.K. Network Enabled Capability (NEC) to the NATO context. This NATO concept is referred to as (NNEC) …. The NNEC must provide for the timely exchange of secure information, utilising communication networks which are seamlessly interconnected, interoperable and robust, and which will support the timely collection, fusion, analysis and sharing of information”.

699. One of the key milestones along the route towards realising the NNEC strategy has been set out in the NATO Networked Consultation, Command and Control Interoperability Policy[3] refers.

700. In particular, the policy states that “It is the intent of NATO that measures shall be put into effect by the Organisation and by individual nations to ensure that information sharing requirements are met securely and expeditiously. This intent requires that appropriate interoperability

[2] EAPC(AC/322)N(2005)0007

[3] AC/322-D(2008)0041 (INV) dated 30 October 20008

solutions and procedures to match IOR over time shall be identified/developed with the nations and documented by the NC3B."

701. This design rule satisfies the above requirement of the NATO Networked C3 Interoperability Policy by identifying the high level design rules required for exchange of information services.

702. Information services are the primary mechanism for information interchange in a NATO environment. This is highlighted in the NATO Networked C3 interoperability policy: "This policy identifies NATO's intent for NNC3 interoperability, and identifies the principles and responsibilities for ensuring the development and effective use of systems to provide interoperable services supporting the sharing of information across the physical, information and human domains".

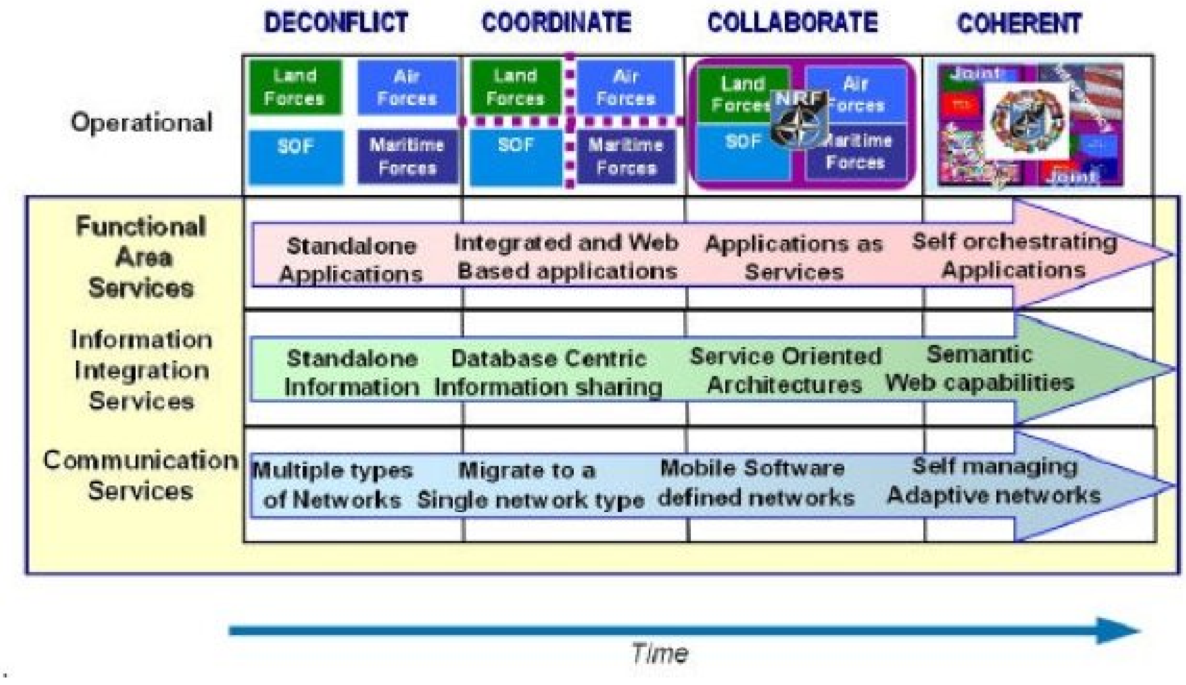

DECONFLICT
COORDINATE
COLLABORATE
COHERENT
Operational
Land Forces
Air Forces
SOF
Maritime Forces
NRF
Joint
Functional Area Services
Standalone Applications
Integrated and Web Based applications
Applications as Services
Self orchestrating Applications
Information Integration Services
Standalone Information
Database Centric Information sharing
Service Oriented Architectures
Semantic Web capabilities
Communication Services
Multiple types of Networks
Migrate to a Single network type
Mobile Software defined networks
Self managing Adaptive networks
Time

Figure F.6. Evolving C3 Requirements and Technology Trends for NNEC

F.4. CONSEQUENCES FROM THE SOLUTIONS

703. SOA offers a mechanism for achieving the agility required for NNEC. Whereas the current stove-piped way of doing business is rigid and difficult to adapt because business functions and the supporting IT are so tightly coupled, an SOA exploits newly available software components and web standards that can be reconfigured easily and quickly. SOA translates capabilities, processes and functions into services which can be invoked by a user through an interface. This requires the services to be available and the user to know the "what, how, how much and when" of accessing them. How the services work is of no consequence to the user but is important to

designers and architects. The underlying principles are not new, but the web services and related technology to bring it to life are; reinforced by their wide acceptance.

704. The predominant precept is that SOA is business driven. This puts designated defence Process Owners in the driving seat because they place requirements for service provision. If SOA is to be successful it means that they must truly understand what drives the capability they are entrusted to deliver so that they are in a position to inform/drive how it can be delivered to users in the most effective and efficient manner possible. New technology enables much looser coupling between business processes and the IT systems which support them and so overcome one of the key drivers of cost in most IT deployments – tight coupling i.e. changes in one area requiring a cascade of other required changes in order to work; with familiar cost, time and performance penalties. To support this, a high level governance structure is essential to enforce data and quality of service standards which enable reuse of services.

705. There are many benefits to SOA. They include access to previously unavailable information, the design of reusable services, the ability to make up new services from existing ones, the ability for businesses to make changes without costly IT expenditure, and so on. Moreover, the issues subtending from the use of legacy systems and the requirement to leverage as much value for money as possible from their continued use, becomes much less difficult by adopting a service perspective. For those who embrace SOA and see it through, the prospect of a working NNEC becomes realisable for the first time.

706. SOA is already here and any new major system provided by any one of the leading industry vendors is likely to have an SOA capability embedded in it. However, it should be noted that the federated model of SOA described in this design rule is still an emerging concept which will take time to reach maturity.

F.5. EXAMPLES

707. The diagram below shows the concept of federated SOA using a simplified model with participants of Organisation A and Organisation B. Organisations are required to build SOA enterprise scale systems that conform to the NATO Overarching Architecture. The organisations' SOA are connected in a federated manner providing maximum scalability and interoperability.

708. The actual physical connection between the SOAs is at the communications layer. The point of interconnection is called the Service interoperability point (SIOP). The standards used to connect at the SIOP are documented in a Service interoperability profile or SIP.

709. There are also logical connections at the Core Services layer and COI Services layers. These connections also have associated SIPs.

710. An example of the Core Services SIOP is currently being investigated and demonstrated by UK MOD.[4]

[4] Federated ESB Interoperabilty Specification - version dated 1 April 2008.

711. There is also a logical connection at the COI Services layer. The ability to share COI services is where the main benefit is realised as these are the business services used to undertake missions. Using the guidelines outlined in this design rule, organisations can interoperate by sharing COI services to perform business tasks. For example the UK MOD SOA pilot project has demonstrated a “logistics demand service” which follows a business process to fulfil a request for a store item or spare part.

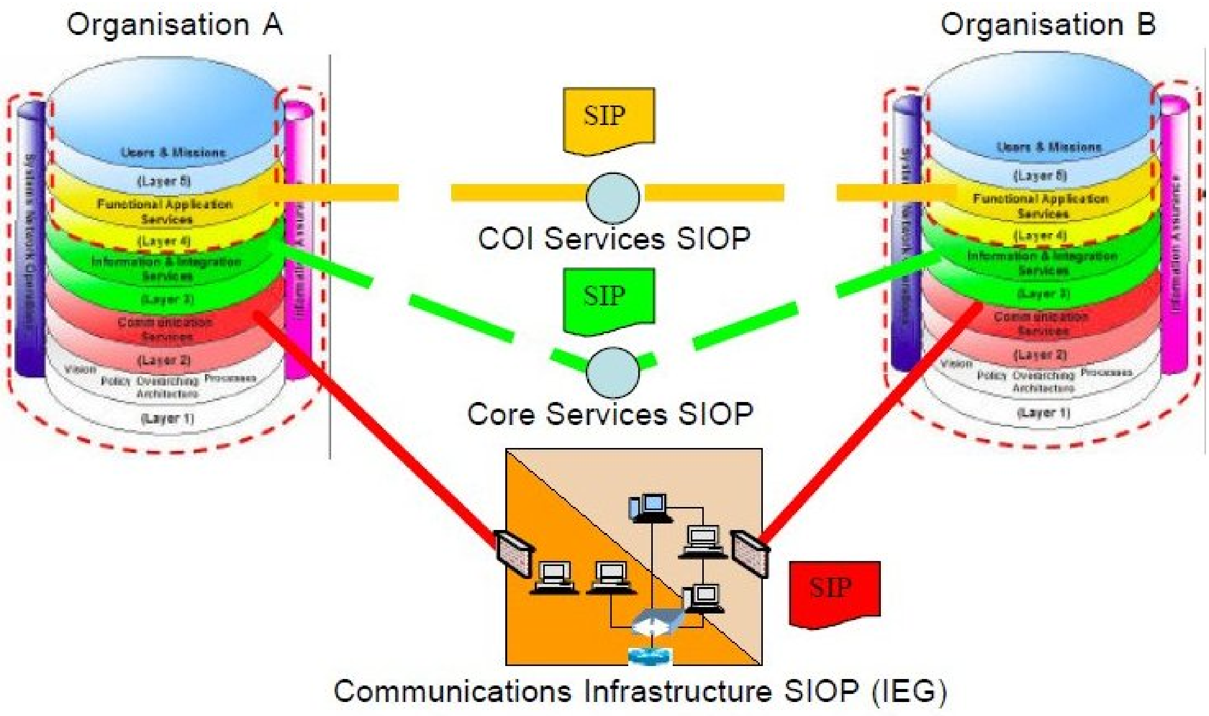


Figure F.7. Service Interoperability Points and their relationship to the Overarching Architecture

F.6. META DATA

F.6.1. Keywords

712. Interoperability, partner, national, international, external, interface,

F.6.2. Associated design rules

Assoc. #	DR ID	DR Product Name & Solution Reference	Release	Validity
1.				
2.				