SINTEF A6538

REPORT

Structuring guidelines for the technical reference architecture for information and communication technology (ICT) in the Norwegian Defence

Brian Elvesæter
Arnor Solberg

April 2008

SINTEF

SINTEF ICT

Address: NO-7465 Trondheim, NORWAY
Location: Forskningsveien 1
Telephone: +47 22 06 73 00
Fax: +47 22 06 73 50

Enterprise No.: NO 948 007 029 MVA

SINTEF REPORT

TITLE

Structuring guidelines for the technical reference architecture for information and communication technology (ICT) in the Norwegian Defence

AUTHOR(S)

Brian Elvesæter, Arnor Solberg

CLIENT(S)

Norwegian Defence Communication and Information Services Division (NDCISD)

REPORT NO.	CLASSIFICATION	CLIENTS REF.	
SINTEF A6538	Open	Camilla Bårnes Roark	
CLASS. THIS PAGE	ISBN	PROJECT NO.	NO. OF PAGES/APPENDICES
Open	978-82-14-04392-1	90B256	33
ELECTRONIC FILE CODE		PROJECT MANAGER (NAME, SIGN.)	CHECKED BY (NAME, SIGN.)
NDCISD_Report_v12.doc		Brian Elvesæter	Svein G. Johnsen
FILE CODE	DATE	APPROVED BY (NAME, POSITION, SIGN.)	
	22.04.2008	Bjørn Skjellaug, Research Director	

ABSTRACT

This report presents structuring guidelines for the technical reference architecture for information and communication technology (ICT) in the Norwegian Defence. The overall structure highlights the different areas that needs to be addressed as part of the technical reference architecture that is going to be established and maintained. The areas are:

- *IT services and applications* which focuses on the structuring and description of software systems.
- *Communication infrastructure* which focuses on the structuring and description of communication infrastructure that enables software systems to be distributed and communicate.
- *Non-functional aspects (NFAs)* which determine important qualities of the ICT system cut across the IT services and applications and the communication infrastructure. The two aspects *management* and *security* are part of the overall structure.
- *Interoperability* which focuses on structuring and description of systems interoperability, i.e., the ability of two or more systems or components to exchange information and to use the information that has been exchanged.
- *Deployment platform* which focuses on the structuring and description of capabilities and constraints of physical assets such as weapon systems, sensors and hardware platforms.

KEYWORDS	ENGLISH	NORWEGIAN
GROUP 1	Information technology	Informasjonsteknologi
GROUP 2	Information infrastructure	Informasjonsinfrastruktur
SELECTED BY AUTHOR	Technical reference architecture	Teknisk referansearkitektur

TABLE OF CONTENTS

1 Introduction

1.1 Context and background

The Norwegian Defence Logistics Organisation (NDLO)[1] is responsible for the establishment and management of a technical reference architecture for the adaptation and use of information and communication technology (ICT) in the Norwegian Defence. The technical reference architecture should be applicable for target architectures for ICT in the Norwegian Defence for the next five to eight years. The technical reference architecture should ensure that all relevant ICT projects and initiatives are streamlined and result in interoperable ICT services that address the requirements and needs of the Norwegian Defence.

This report presents structuring guidelines for the technical reference architecture for ICT in the Norwegian Defence. The guidelines address different parts of the ICT architecture, namely IT services and applications, communication infrastructure, management, security, deployment platforms and interoperability. The technical reference architecture needs to be populated with concrete descriptions for each of these parts in order to give more specific guidelines for the development of specific target architectures.

The focus of this report is only on the structure of the technical reference architecture and not its additional contents.

1.2 Target audience

The target audience of this report are the following groups:

- The Norwegian Defence Communication and Information Services Division (NDCISD)[2] architecture group which is responsible for establishing and managing the technical reference architecture for the Norwegian Defence.
- Technical groups/departments in NDCISD which are responsible for specialised areas of the ICT in the Norwegian Defence.
- Other relevant groups/departments in NDLO that have concerns or interests related to the technical reference architecture.
- ICT projects and initiatives that deals with the specification, procurement, development and/or management of ICT services and target architectures for the Norwegian Defence.
- The architecture forum in NDLO.

1.3 Input sources

This report reflects the current state-of-the-art in IT system architectures and systems interoperability and embraces modern architectural trends such as service-oriented architecture (SOA). In addition to the general knowledge of modern system architectures addressing aspects such as systems interoperability, the specific content of this report is based on the following sources:

- NATO Architecture Framework (NAF) [1] and the NATO C3 Technical Architecture [2].
- Input from domain experts at NDCISD, specifically Camilla Bårnes Roark, Espen Gjøs and Asbjørn Steinsvik.
- Norwegian reference model for the defence information infrastructure [3-6].

[1] Forsvarets logistikkorganisasjon (FLO)
[2] Forsvarets logistikkorganisasjon/Informasjons- og kommunikasjonstjenester (FLO/IKT)

- OASIS "Reference Model for Service Oriented Architecture" [7].
- OSI reference model "Open Systems Interconnection – Basic Reference Model" [8].

2 Architecture concepts and definitions

2.1 Architecture terminology

Even if it is used by many, the term "architecture" has no well established definition. Nevertheless in the field of software engineering there is no shortage of more or less overlapping definitions [9]. The IEEE 1471-2000 recommended practice for architectural description [10] (see Appendix C) include definitions of important terms and relate these in a conceptual model of architectural description.

In this report we adopt the terms from the general terminology (see Appendix A) in chapter 7 of the NATO Architecture Framework (NAF) [1]. NAF defines *architecture* as the fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution. NAF defines a set of *views* and *subviews* (see Appendix B) for how to establish *models* that describe different aspects of an architecture.

In addition to the NAF terminology, NATO has a set of glossaries [11] that may be considered relevant for the work on describing technical architectures such as AAP-31 CIS Glossary, AComP-1 Communications Glossary, ADatP-2 Information Technology Glossary, AAP-6 NATO Glossary of Terms and Definitions, AAP-15 Glossary of Abbreviations used in NATO documents and ACMP-6 NATO Configuration Management Glossary.

2.2 Types of NATO architectures

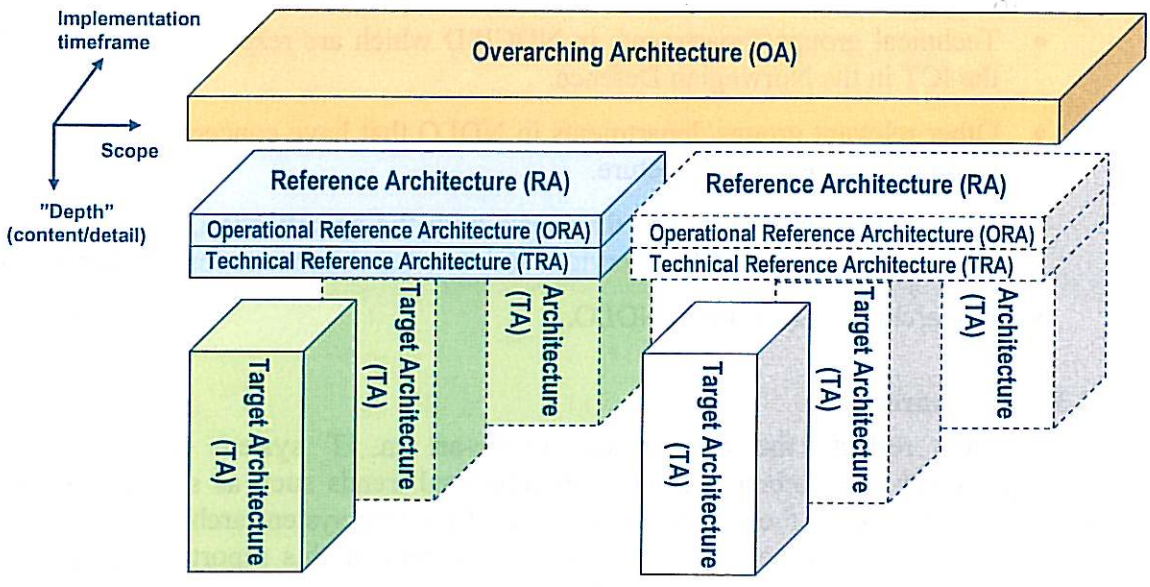

Figure 1: Relationship between different architecture types

NATO defines three different types of architectures in NAF [1]:

- The *overarching architecture (OA)* contains an overview of all related systems and their interoperability.
- The *reference architecture (RA)* is more specific and spans a timeframe of about 5 years.
- Each RA applies to multiple *target architectures (TA)* that are implementation snapshots in time of their related RAs.

In this report we further divide the RA into two:

- The *operational reference architecture (ORA)* focuses on the more operational parts of the RA and is largely covered by the capability, operational and programme views of NAF.
- The *technical reference architecture (TRA)* focuses on the more technical parts of the RA and is largely covered by the service-oriented, systems and technical views of NAF.

The structuring guidelines described in this report apply specifically to the technical part (TRA) of the reference architecture for the Norwegian Defence that spans a timeframe of 5-8 years. However, the structuring guidelines are general and can also be applied to other ICT reference architectures.

The content of the technical reference architecture for the Norwegian Defence should be described according to the views and subviews defined in NAF. As a starting point one may use existing architectural descriptions available in NDLO that describe the "as-is" situation and thus can be used to define a *baseline architecture (BA)*. Ideally, the BA should be described according to NAF, however such formal architectural descriptions may not be available during the initial architecture work.

3 Guidelines for structuring the technical reference architecture

3.1 Overall structure

This report presents structuring guidelines for the technical reference architecture for information and communication technology (ICT) in the Norwegian Defence. The overall structure highlights the different areas of an ICT architecture that needs to be addressed as part of the technical reference architecture that is going to be established and maintained.

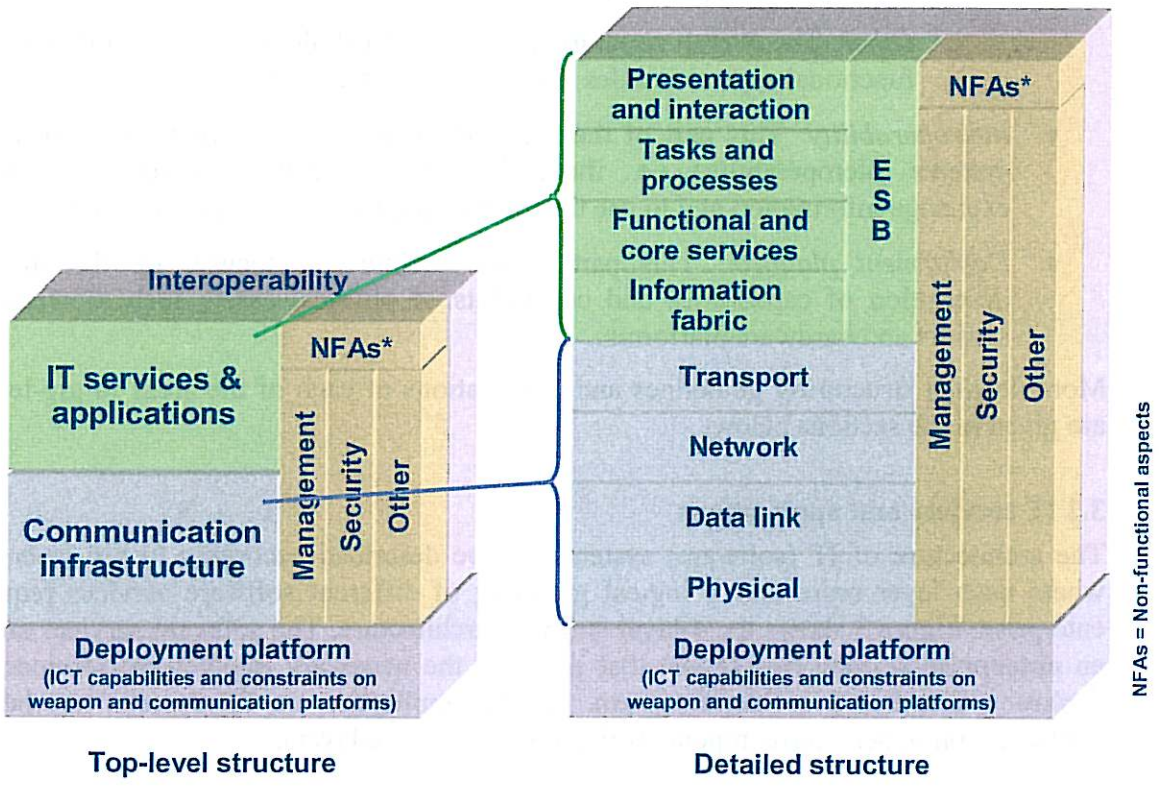

Figure 2: Overall structure (top-level view & detailed view)

Figure 2 shows two different representations of the overall structure. The *top-level structure* identifies the areas IT services and applications, communication infrastructure, the non-functional

aspects (NFAs) management and security, interoperability and deployment platform as the main parts of a technical ICT architecture. In the *detailed structure* the two areas IT services and applications and communication infrastructure have been further divided into subareas.

A short description of the main areas of the top-level structure is given below:

- *IT services and applications:* This part of the architecture focuses on the structuring and description of software systems.
    - In the detailed structure this part is further divided into *presentation and interactions*, *tasks and processes*, *functional and core services*, *information fabric* and *enterprise service bus (ESB)*.
- *Communication infrastructure:* This part of the architecture focuses on the structuring and description of communication infrastructure that enables software systems to be distributed and communicate.
    - In the detailed structure this part is further divided into *transport*, *network*, *data link* and *physical* layers.
- *Non-functional aspects (NFAs):* Describing the functionality offered by an ICT system is a key part of an architectural description. However, the non-functional aspects determine the quality of the ICT system. Non-functional aspects are typically cross-cutting aspects that cut across the logical layers of the IT services and applications, and the communication infrastructure.
    - *Management:* Management focuses on the life-cycle of ICT services.
    - *Security:* Security focuses on security policies and the mechanisms required enforcing these policies.
    - *Other:* The overall structure includes a placeholder for adding other important non-functional aspects besides management and security.
- *Interoperability:* This part of the architecture focuses on structuring and description of systems interoperability, i.e., the ability of two or more systems or components to exchange information and to use the information that has been exchanged.
- *Deployment platform:* This part of the architecture focuses on the structuring and description of capabilities and constraints of physical assets such as weapon systems, sensors and hardware platforms.

More detailed structuring guidelines and explanations of each of the areas in top-level structure are given in the sections below.

3.2 IT services and applications

The architecture of IT (software) systems can be described according to a 4-layer architecture where each layer represents a logical grouping of different software services required by the enterprise. Figure 3 shows the 4-layer software architecture. The software services are coupled to an **enterprise service bus (ESB)** that provides the necessary middleware services required to deploy a distributed software system. An IT application or IT system can be seen as a configuration of services/components that resides in these layers.

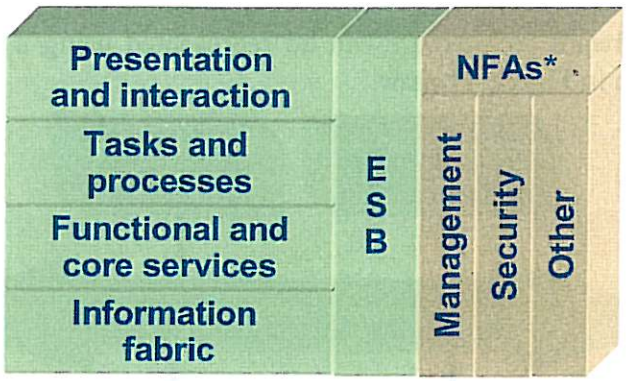

Figure 3: 4-layer software architecture

The four layers are as follows:

1. **Presentation and interface layer** provides presentation and user dialog logic. This will typically contain Web clients and application clients. Application clients can typically accommodate a richer graphical user interface and support offline work better than what can be provided by Web clients.
2. **Tasks and processes layer** provides the enterprise process model that focuses on the semi-automated execution of enterprise processes to support the enterprise operations or more specifically the work tasks of the end-users.
3. **Functional and core service layer** provides distributed and network-visible enterprise-level services, and is responsible for protecting the integrity of enterprise resources. A service in this layer can be seen as an application function packaged as a reusable component which can be used to support user tasks and enterprise processes.
4. **Information fabric layer** provides global persistence services. Resource adapters, e.g., Open Database Connectivity (ODBC) and JDBC[3], provide access, search and update services for using database management systems (DBMS). Typically, integration of legacy information systems, e.g., enterprise resource planning (ERP), mainframe transaction processing and database systems, is done at this layer.

3.2.1 Service-oriented architecture (SOA)

Service-oriented architecture (SOA) refers to the latest trend in system architectures. Technically, SOA can be seen as a software architecture that defines the use of loosely coupled software services to support the requirements of the business processes and software users. In a service-oriented environment, resources on a network are made available as independent services that can be accessed without knowledge of their underlying platform implementation. According to W3C, SOA specifies a set of components whose interfaces can be described, published and discovered and thus enabling services to be invoked over a network.

SOA aims to promote software development in a way that leverages the construction of dynamic systems which can easily adapt to volatile environments and be easily maintained as well. SOA enables flexible connectivity of applications or resources by representing every application or resource as a service with a standardized interface. This enables them to exchange structured information quickly and flexibly. This flexibility enables new and existing applications to be easily and quickly combined to address changing business needs, and the ability to easily combine and choreograph applications allows IT services to more readily reflect business processes.

[3] JDBC is a trademarked name and is not an acronym. Nevertheless, JDBC is often thought of as standing for "Java Database Connectivity".

The 4-layer software architecture can be represented using a service bus software architecture pattern as shown in Figure 4 below.

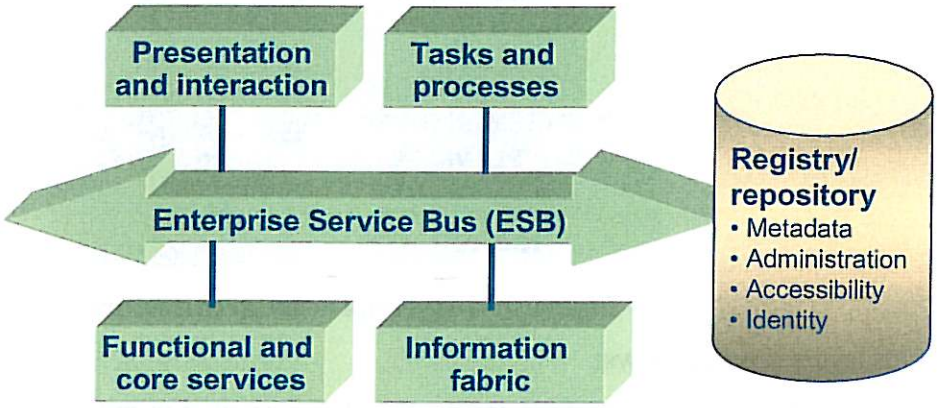

Figure 4: Service-oriented software architecture

The OASIS reference model for SOA [7] (see Appendix D) defines SOA as *"a paradigm for organising and utilising distributed capabilities that may be under the control of different ownership domains."* The reference model defines seven principal concepts as the basis for describing service-oriented architectures. A service is the means by which the needs of a consumer are brought together with the capabilities of a provider. A *service* offers one or many *functions* that result in *real world effects* if being invoked. 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 in the *behaviour model* and the *information model* of the *service description*.

The document "Policy for military adaptation and use of information and communication technology in the Norwegian Armed Forces" [5] defines a reference model called the *information infrastructure (INI)* and contains a brief explanation of the services defined in this reference model. Appendix E classifies the identified services of the INI according to the structure proposed in this report. Please note that the INI reference model is currently under revision.

3.2.2 Enterprise service bus (ESB)

An enterprise service bus (ESB) is a pattern of middleware that unifies and connects services, applications and resources within a business. Registry, repository, management and security services can be part of the middleware services offered by an ESB. The report "Forsvarets tekniske arkitektur for integrasjonsmellomvare" [12] further details the architecture of an ESB for the Norwegian Defence.

The ESB represents a new way of looking at how to integrate applications, coordinate resources and manipulate information. Unlike many previous approaches for connecting distributed applications, for example remote procedure call (RPC) and distributed objects, the ESB pattern enables the connection of software running in parallel on different platforms, written in different programming languages and using different programming models. The ESB may also provide middleware services such as composition, mediation, matchmaking and transformation that enable interoperability between software systems.

The ESB provides the open, standards-based connectivity infrastructure for a service-oriented architecture. Each of the interactions with the ESB ideally makes use of a WSDL-based service definition, invoking the required transport services and quality of service.

3.2.3 Addressing IT services and applications in NAF v3

IT services and applications are mainly addressed by the *NATO Service-Oriented View (NSOV)* and the *NATO Systems View (NSV)*. However, some relevant subviews from the *NATO Operation View (NOV)* may also be used.

- The *information fabric* can be described using the *Information Model (NOV-7)* with further detailing in the *System Data Model (NSV-5)*.
- Functional and core services SOA can be described using the subviews *Service Taxonomy (NSOV-1)*, *Service Definitions (NSOV-2)*, *Service Orchestration (NSOV-4)* and *Service Behaviour (NSOV-5)*.
- The *processes and tasks* can be described using the *Operational Activity Model (NOV-5)* with further detailing in the *Services to Operational Activities Mapping (NSOV-3)*.

3.3 Communication infrastructure

The communication infrastructure can be divided into a set of layers. The two most common reference models for structuring these layers is the Open Systems Interconnection (OSI) reference model [8] and the TCP/IP reference model. The OSI model defines 7 layers and the TCP/IP model defines 4 layers. However, both the OSI model and the TCP/IP model have some weaknesses as further explained in Appendix F.

In this report we propose to use a 5-layer hybrid reference model recommended by Tanenbaum in [13]. Figure 5 shows the bottom 4 layers of the hybrid reference model. Layer 5 of the hybrid model is called the *application layer* and corresponds to the area *IT services and applications* in the overall structure proposed in this report (cf. Figure 2). The application layer supports application and end-user processes. From a communication infrastructure point of view, it is the communication-centric services, e.g., network management and transport protocols, which are important to consider. Such communication services could be part of the middleware services that are offered by enterprise service bus (ESB) technologies. The application layer is deliberately omitted in Figure 5 to better align with the figures presented earlier in this report.

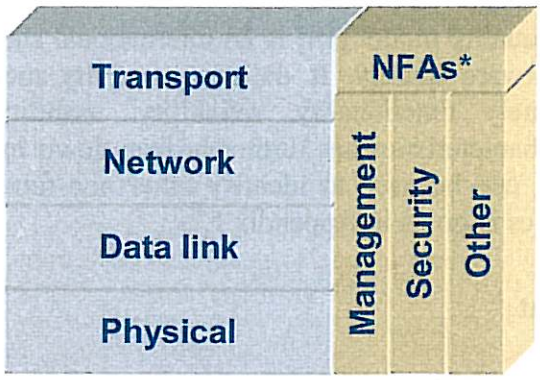

Figure 5: Communication infrastructure layers

The hybrid reference model defines the following four layers:

- **Transport layer:** This layer provides transparent transfer of data between end systems, or hosts, and is responsible for end-to-end error recovery and flow control. It ensures complete data transfer.
- **Network layer:** This layer is concerned with controlling the operation of the subnet. It provides switching and routing technologies, creating logical paths, known as virtual circuits, for transmitting data from node to node. Routing and forwarding are functions of this layer, as well as addressing, internetworking, error handling, congestion control and packet sequencing.

- **Data link layer:** At this layer, data packets are encoded and decoded into bits. It furnishes transmission protocol knowledge and management and handles errors in the physical layer, flow control and frame synchronization. The data link layer is divided into two sublayers: The Media Access Control (MAC) layer and the Logical Link Control (LLC) layer. The MAC sublayer controls how a computer on the network gains access to the data and permission to transmit it. The LLC layer controls frame synchronization, flow control and error checking.
- **Physical layer:** This layer is concerned with transmitting raw bits over a communication channel. It conveys the bit stream, i.e., electrical impulse, light or radio signal, through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data on a carrier, including defining cables, cards and physical aspects. Fast Ethernet, RS232, and ATM are protocols with physical layer components.

3.3.1 Addressing communication infrastructure in NAF v3

The communication infrastructure is covered by the *NATO Systems View (NSV)* and in particular the *System Interface Description (NSV-1)* and *Systems Communication Description (NSV-2)* which is further divided into the subviews *System Port Specification (NSV-2a)*, *System to System Port Connectivity (NSV-2b)*, *System Connectivity Clusters (NSV-2c)* and *Systems Communication Quality Requirements (NSV-2d)*.

3.4 Non-functional aspects

Non-functional aspects determine the quality of the ICT system. Non-functional aspects are typically cross-cutting aspects that cut across the layers of the IT services and applications, and the communication infrastructure areas. For some specific ICT systems there may be other non-functional aspects, such as performance, maintainability, reliability and usability that are important to describe as part of the technical architecture. Furthermore, since non-functional aspects typically have interdependencies, there may also be tradeoffs between certain aspects that one want to describe as part of the technical reference architecture.

The ISO/IEC 9126 quality model [14] is an international standard that characterises software for the purpose of software quality. It defines a quality model that identifies 6 main quality characteristics, namely functionality, reliability, usability, efficiency, maintainability and portability. These characteristics are further broken down into sub-characteristics. The ISO/IEC 9126 quality model can be used to identify other non-functional aspects. An overview of the defined characteristics is given in Appendix E[4].

3.4.1 Management

3.4.1.1 Service management

IT service management is a discipline for managing IT systems and is regarded as a primary enabler for IT governance. Governance is the combination of processes, practices and tools which facilitate life-cycle of enterprise services and provide means for creating, communicating, enforcing and managing compliance to corporate policies regarding the non-functional service characteristics that are of importance to business today. Governance can occur at:

[4] Please note that the ISO 9126 quality model defines both security and interoperability as qualities of a software system. In this report we consider security and interoperability as such important aspects of the architecture that they have been highlighted in the overall structure.

- *Design-time*, e.g., ensure all service definitions are stored in a registry and that no new services can be developed whose responsibilities significantly overlap with existing services.
- *Run-time*, e.g., a service should authenticate users and should not accept a user name token with unencrypted password or self-issued certificates.
- *Operations*, e.g., when this service's response time exceeds 5 seconds, an alert should be sent to an administrator/monitoring person.

Additionally, governance can relate to compliance with policies at both a management and technical level:

- *Corporate*, e.g., ensure we are compliant with the emerging EuroSOX directives [15].
- *Technical*, e.g., ensure all Web services are WS-I Basic Profile [16] compliant.

3.4.1.2 Network management

Network management refers to the activities, methods, procedures, and tools that pertain to the operation, administration, maintenance, and provisioning of networked systems [17].

- *Operation* deals with keeping the network (and the services that the network provides) up and running smoothly. It includes monitoring the network to spot problems as soon as possible, ideally before users are affected.
- *Administration* deals with keeping track of resources in the network and how they are assigned. It includes all the "housekeeping" that is necessary to keep the network under control.
- *Maintenance* is concerned with performing repairs and upgrades, e.g., when equipment must be replaced, when a router needs a patch for an operating system image, when a new switch is added to a network. Maintenance also involves corrective and preventive measures to make the managed network run "better", such as adjusting device configuration parameters.
- *Provisioning* is concerned with configuring resources in the network to support a given service, e.g., setting up the network so that a new customer can receive voice service.

Functions that are performed as part of network management accordingly include controlling, planning, allocating, deploying, coordinating, and monitoring the resources of a network, network planning, frequency allocation, predetermined traffic routing to support load balancing, cryptographic key distribution authorization, configuration management, fault management, security management, performance management, bandwidth management, and accounting management.

A large number of access methods exist to support network and network device management. Examples of access methods are Common Management Information Protocol (CMIP) [18], Simple Network Management Protocol (SNMP) [19], NETCONF [20], and Java Management Extensions (JMX) [21].

Schemas include the Web-Based Enterprise Management (WBEM) [22] which is a set of systems management technologies developed to unify the management of distributed computing environments. WBEM is based on Internet standards and Distributed Management Task Force (DMTF) open standards such as the Common Information Model (CIM) [23]. CIM is an open standard that defines how managed elements in an IT environment are represented as a common set of objects and relationships between them.

3.4.1.3 Addressing management in NAF v3

NAF refers to the ISO/IEC 15288 standard on system life-cycle processes [24]. System life-cycle process management concern the tasks and responsibilities to assure that effective system life-cycle processes are available for use by the organisation. These life-cycle processes are guaranteed to be consistent with the organisation's goals and policies, that are defined, and adapted and maintained in a consistent way in order to meet the nature of individual projects, and that are capable of being applied using effective, proven methods and tools.

Since management is a cross-cutting aspect it needs to be addressed from all of the relevant views chosen to describe the ICT architecture. Furthermore, relevant management standards should be listed in the *NATO Technical View (NTV)*.

3.4.2 Security

Security is a key aspect of defence systems. Security management concern the tasks and responsibilities to guarantee availability, confidentiality, integrity, as well as secure communication throughout the architecture. The security aspect needs to be considered throughout any architecture development effort. This implies identification of security aspects in the mission space and handling of the security issues accordingly in the ICT architecture.

3.4.2.1 Information and communication security

For the ICT reference architecture the information and communication security are the major concerns. The nature of information and communication security (e.g., availability, integrity and confidentiality) is briefly described below. More elaborated descriptions can be found in standards and security frameworks, e.g., the NC3A's Architecture Engineering Methodology (AEM).

Information integrity: In order to establish the integrity of the information used by the organisation, each information object has four characteristics that define whether the information object correctly represents what it is supposed to represent. This fact is independent of the way the information object is used in any communication. The characteristics that are dependent of the manner of communication are:

- *Criticality:* The bearing the information object has on the organisation.
- *Currentness:* The (absolute or relative) moment in time the information object reflects.
- *Accuracy:* The level of detail of the information object.
- *Priority:* Indicating the level of precedence to another information object.

Information confidentiality: In order to establish the trustworthiness of the information object two characteristics need to be established:

- *Authority of origin:* The trustworthiness and mandate of the organisation of the producer of the information object content. There can be multiple origins for different properties of the information object. All of them must be identified.
- *Data adaptability:* To what extent the information object can be used in different contexts, without changing its meaning, i.e., the risk of misinterpretation.

The communication security is the companion to the information security. Communication characteristics determine the *availability* of the right (combination of) information objects to the right actor, at the right time (and in the right format).

Communication integrity: In order to determine the integrity of the exchange of information the following characteristics need to be established:

- *Timeliness:* The time window in which the content of the information object remains valid to the actor involved.
- *Criticality of arrival:* The level of certainty that the information object is available at its destination.
- *Frequency and periodicity:* The number of occurrences of the communication in a certain timeframe and the intervals of the occurrences.

At design and implementation time, two additional communication characteristics become relevant:

- *Form(at):* The appearance of the information. The communication channel may require special form(at) of the communicated data, e.g., required packaging related to communication protocols, compression of transformed data, encryption of transformed data etc.
- *Volume:* The size of "one communication exchange" determines the time needed for "one communication exchange", i.e., influence on availability.

Communication confidentiality: In order to determine the trustworthiness of the exchange of information, four characteristics need to be established:

- *Authority of originator:* The trustworthiness and mandate of the organisation of the originator of the information. The applied authentication and authorization schemes used for the originator impact the communication confidentiality.
- *Authority of recipient:* The trustworthiness and mandate of the organisation of the recipient of the information. The applied authentication and authorization schemes used for the originator impact the communication confidentiality.
- *Traceability/security of communication:* The trustworthiness of the path (physical network) that provide the communication of the data. This typically involve risk analysis and treatments for instance to determine the need for recording in order to detect anomalies or the need for encryption etc.
- *Communication adaptability:* The possibility that the communication can follow different paths, i.e., to ensure availability.

3.4.2.2 Security mechanisms and security categories

As part of identification of security aspects in the mission space and handling of the security issues, formal risk assessment is needed at the architectural level leading to security provisions that match the aspect. Some main security mechanisms are:

- *Identification and authentication (IA):* Identification is the process whereby a network element recognizes a valid user's identity. Authentication is the process of verifying the claimed identity of a user. A user may be a person, a process, or a system (e.g., an operations system or another network element) that accesses a network element to perform tasks or process a call.
- *Authorisation and access (AA) control:* Authorisation and access control mechanism allows you to restrict access based on criteria unrelated to the identity of the user (e.g., username and password, host name or host address of the machine requesting information etc.).

- *Security auditing (SA):* A manual or automated systematic measurable technical assessment (of a system or application). Manual assessments include interviewing staff, performing security vulnerability scans, reviewing application and operating system access controls, and analyzing physical access to the systems. Automated assessments include system generated audit reports or using software to monitor and report changes to files and settings on a system etc.
- *Security of communication (SC):* Communications security includes crypto security, transmission security, emission security, traffic-flow security, and physical security (physical measures necessary to safeguard classified equipment, material, and documents from access thereto or observation thereof by unauthorized persons).
- *Non-repudiation (NR):* The concept of ensuring that a party in a dispute cannot repudiate, or refute the validity of a statement or contract. Although this concept can be applied to any transmission, by far the most common application is in the verification and trust of signatures.

These security mechanisms are typically linked to security categories. Security categories are used to categorise the nature and sensitivity of information objects and interactions (e.g., *Top Secret, Secret, Confidential, Unclassified*). These categories are then again used to identify which security mechanisms that should be applied in order to satisfy the defined security level.

3.4.2.3 Addressing security in NAF v3

NAF emphasises the importance of the security and governance for all architectural aspects where information and communication security is important. A central reference standard is the ISO/IEC 17799 standard on information security management [25].

In NAF, security management includes the support of NATO security accreditation processes in the architectural design stage, management of access rights and control of access to architecture infrastructure and architecture repository, and addressing architecture product classification.

Since security is a cross-cutting aspect it needs to be addressed from all of the relevant views chosen to describe the ICT architecture. NAF stresses to provide a comprehensive specification of how systems are connected at a detailed infrastructural level, what interfaces each system exposes (ports), the hardware interfaces used, and the protocols that govern transmission of data across the interface. Furthermore, relevant security standards should be listed in the *NATO Technical View (NTV)*.

3.5 Interoperability

Definitions on interoperability have been intensively reviewed [26, 27]. Generally speaking, interoperability is the ability or the aptitude which two systems have to understand one and the other and to function together. The word "inter-operate" implies that one system performs an operation for another system. From computer technology point of view, it is the faculty for two heterogeneous computer systems to function jointly and to give access to their resources in a reciprocal way. In the context of networked enterprises (extended, virtual...), interoperability refers to the ability of interactions (exchange of information and services) between the enterprise systems. The IEEE definition [28] of interoperability is:

> *"the ability of two or more systems or components to exchange information and to use the information that has been exchanged"*

An important remark is to consider interoperability as a problem of compatibility which is not only concerned with ICT aspect, but other aspects of enterprise as well. Thus developing

interoperability means to develop knowledge that removes incompatibilities that may exist between any two heterogeneous systems.

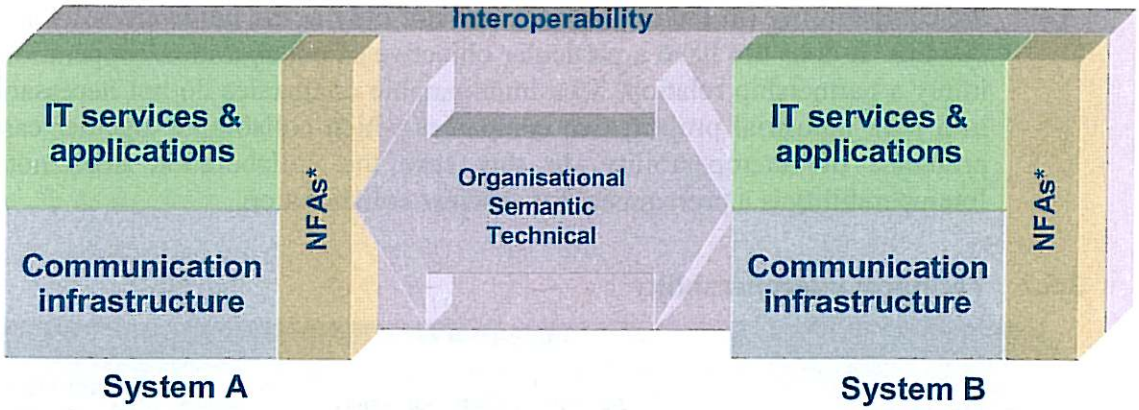

Figure 6: Technical, semantic and organisational interoperability

The European Interoperability Framework (EIF) [29, 30] describes three aspects of interoperability, namely technical, semantic and organisational.

- *Technical interoperability* is the linking up of computer systems through the agreement on standards for presenting, collecting, exchanging, processing and transporting data. This includes key area such as multi-channel access, open interfacing, and data integration using various methods, including implementation of XML-based standards.
- *Semantic interoperability* is concerned with ensuring exchanged data share the same meaning at both their origin and their destination to enable systems to combine and process received information from other resources. In part, this involves the furtherance of coordinated common semantics on the basis of XML and other standards.
- *Organisational interoperability* relates to the inter- and intra-organisational exchange of data and information and requires relevant organisation of business process and internal structures of the organisation (e.g., responsibilities, authorisation) for better exchange, understanding and usage of those data. Most particularly, this requires relevant modelling and harmonising of business processes at their points of exchange.

Please note that the NAF definition of interoperability as defined in Appendix A takes the point of view of organisational interoperability which is outside the scope of this report. The interoperability focus of the technical architecture is on technical and semantic interoperability.

3.5.1 Clarifications

Although there exist many definitions on interoperability, but those definitions do not yet provide a clear understanding. To define the domain of interoperability, it is necessary to clarify some confusing concepts around the notion of interoperability.

- *Interoperability vs. integration:* Generally, interoperability has the meaning of coexistence, autonomy and exchange, whereas integration refers to the concepts of coordination, coherence and uniformisation. From the point of view of degree of coupling, the "tightly coupled" indicates that the components are interdependent and cannot be separated. We are in the case of an integrated system. The loosely coupled means that the components are connected by a communication network; they can exchange services while continuing their own logic of operation. It is the case of interoperability. Thus two integrated systems are interoperable; but two interoperable systems are not necessarily an integrated one.

- *Interoperability vs. collaboration:* The concept of interoperability is also different from the concepts "collaboration" and "cooperation". Interoperability is a property relating to the compatibility (in the broad sense and not only at the hardware/software level) of two systems. It does not have a particular objective of collaboration/cooperation and does not imply a partnership relation. Two interoperable companies do not necessarily collaborate in a joint industrial project; two companies which collaborate together can have serious problems of interoperability. In this case the collaboration does not really exist. Interoperability is a prerequisite for efficient collaboration.

3.5.2 Technical interoperability

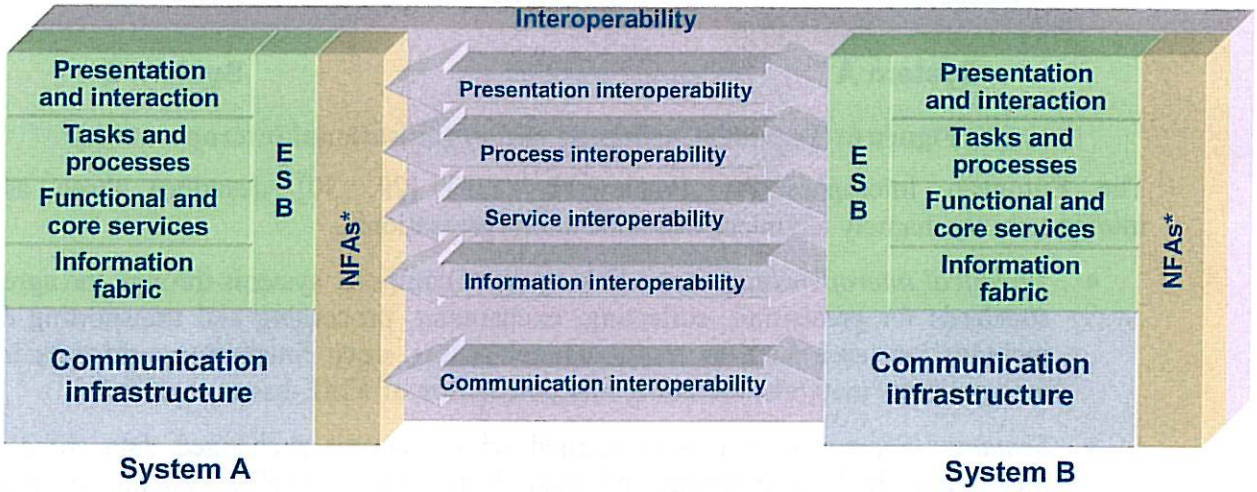

Figure 7: Technical interoperability at different levels

Technical interoperability can be seen at the different layers of the ICT architecture:

- Interoperability of **presentation and interaction** is related to how to connect various (graphical) user interface components together in order to form a consistent workplace environment (e.g., a Web portal) towards the end-users.
- Interoperability of **processes** aims to make various processes fit together. A process defines the sequence of the services (functions) according to achieve some defined outcome that can be either business-oriented (e.g., a business process) or human-oriented (e.g., a task).
- Interoperability of **services** is concerned with identifying, composing and executing various software services (designed and implemented independently). Services are an abstraction and an encapsulation of the functionality provided by an autonomous entity.
- Interoperability of **information/data** is related to the management, exchange and processing of different documents, messages and/or structures by different collaborating entities so that each party in the collaboration interprets the data in the same manner (i.e., the information is shared).
- Interoperability of **communication** is related to interconnectivity of infrastructures.

Non-functional aspects such as management and security that cut across these layers must also be taken into account when addressing interoperability.

3.5.3 Semantic interoperability

To overcome the semantic barriers which emerge from different interpretations of syntactic descriptions, precise, computer processable meaning must be associated with the models

expressed on the different levels. It has to be ensured that **semantics** are exchangeable and based on common understanding in order to enhance interoperability. This can be achieved using **ontologies** and an annotation formalism for defining meaning in the exchanged models.

Semantic interoperability cuts across all layers of the ICT system and relates how information represented at the different layers should be understood in a common and consistent manner.

3.5.4 Addressing interoperability in NAF v3

Interoperability is closely linked to the use of standards which is covered by the *NATO Technical View (NTV)* which defines the subviews *Technical Standards Profile (NTV-1)*, *Technical Standards Forecast (NTV-2)* and *Standard Configurations (NTV-3)*. The proposed structuring of ICT architectures in this report can be used to support classification of standards in these subviews as illustrated in Figure 8.

Layer/aspect	Technical standards
Presentation and interaction	HTML, Java ServerPages (JSP), Java ServerFaces (JSF)
Tasks and processes	BPEL, WS-Rules
Functional and core services	WSDL 1.1, WSDL 2.0, OWL-S
Information fabric	XML, XSD, SQL, ODF, OWL
ESB	CORBA, Web Services Architecture (WSA), HTTP, SOAP
Communication infrastructure	TCP, UDP, IPv4, IPv6, Ethernet
Management	SNMP, JMX, ebXML Registry Information Model (RIM), UDDI v2, UDDI v3, WBEM, Cmoon Information Model (CIM), ISO/IEC 15288
Security	PKI, HTTPS, WS-Security, ISO/IEC 15288
Other NFAs	ISO/IEC 9126
Interoperability	Nato Interoperability Standards and Profiles (NISP), WS-I Basic Profile 1.2

Figure 8: Structuring standards

For instance service-oriented systems can be developed using Web Services Architecture [31] standards such as SOAP [32], XSD [33], WSDL [34] and BPEL [35]. Some management concerns are supported by UDDI [36] and there exists a set of security standards for Web services [37]. Technical interoperability of Web services is prescribed by the WS-I Basic Profile [16], and semantic interoperability can be handled specifically by information model and exchange standards, or more general by adopting semantic technologies such as OWL [38] to precisely define the semantics in the information being exchanged and processed. In addition, NATO has a set of specific standards that must be incorporated. The website [39] provides a library of NATO standardization agreements for procedures and systems and equipment components, known as STANAGs.

The work on structuring standards should be related to the ongoing work on NATO Interoperability Standards and Profiles (NISP) [40]. NISP provides a finer-grained classification

for standards than those presented in the overall structure presented in this report. Furthermore, NISP defines a further timeframe/scope dimensions for classifying standards:

- *Emerging long term:* A standard is considered emerging long term if it deals with technology that is expected to be useful in the long term to NATO.
- *Emerging mid term:* A standard is considered emerging mid term if it is sufficiently mature to be used within the current or next planned systems.
- *Emerging near term:* A standard is considered Emerging near term if it is mature enough to be used within 0-2 years.
- *Mandatory:* A standard is considered mandatory if it is mature to be used immediately. This means that it may both be applied within existing systems and in within future mid term planned systems.
- *Fading:* A standard is considered fading if the standard is still applicable for existing systems. The standard however is becoming obsolete or will be replaced by a newer version or another standard. Except for legacy systems or interoperability with legacy systems, the standard may not be used any more.
- *Retired:* A standard is considered retired if the standard, that has been used in the past, but is not applicable any more for existing systems.
- *Rejected:* A standard is considered rejected if, while it was still emerging, it is considered unsuitable for use within NATO.

3.6 Deployment platform

The area *deployment platform* in the overall structure is mainly introduced as a placeholder for describing physical assets such as weapon systems, sensors and hardware platforms that may not be captured precise enough in architecture frameworks. Thus, additional descriptions may be introduced and added to the technical reference architecture. Examples of characteristics that one may want to describe are radiation, shielding, environmental properties, heat radiation, dimension and other capabilities and/or tolerance levels.

3.6.1 Addressing deployment platform in NAF v3

The deployment platform is covered by the *NATO Systems View (NSV)*. NAF defines *physical assets* (such as weapons systems, sensors and platforms) and *capability configurations* (combining the concept of a system with the concept of a role that is using that system) that can be described in the subviews *System Interface Description (NSV-1)* and *Service Provision (NSV-12)*.

4 Conclusions

This report presents structuring guidelines for the technical reference architecture for information and communication technology (ICT) in the Norwegian Defence. The guidelines address different parts of the ICT architecture, namely *IT services and applications*, *communication infrastructure*, the non-functional aspects (NFAs) *management* and *security, interoperability* and *deployment platform* as the main parts of a technical ICT architecture.

The overall structure highlights the different areas of an ICT architecture that needs to be addressed as part of the technical reference architecture that is going to be established and maintained. The technical reference architecture needs to be populated with concrete descriptions for each of these parts in order to give more specific guidelines for the development of specific target architectures.

5 References

[1] NATO, "NATO Architecture Framework version 3", NATO, 2007.

[2] NATO, "NATO C3 Technical Architecture - The NNEC Service Oriented Architecture", NATO. http://194.7.80.153/website/book.asp?menuid=15&vs=0&page=volume2%2Fch03s03.html (last visited 2008).

[3] Forsvarsdepartementet, "Beskrivelse av programområde informasjonsinfrastruktur: Plan for perioden 2006-2009+, Versjon 1.01", Forsvarsdepartementet. http://www.regjeringen.no/upload/kilde/fd/bro/2006/0020/ddd/pdfv/277792-informasjonsinfrastruktur_programomrade.pdf

[4] Forsvarsdepartementet, "Policy for militær tilpasning og anvendelse av informasjons- og kommunikasjonsteknologi i Forsvaret", Forsvarsdepartementet, 1 September 2005. http://www.regjeringen.no/upload/kilde/fd/bro/2006/0020/ddd/pdfv/277246-ikt_policy.pdf

[5] Forsvarsdepartementet, "Policy for military adaptation and use of information and communication technology in the Norwegian Armed Forces", Forsvarsdepartementet, 1 September 2005. http://www.regjeringen.no/upload/FD/Reglement/Policy_military_adaption_IKT.pdf

[6] Forsvarsdepartementet, "Konsept for styring av elektronisk informasjon i Forsvaret", Forsvarsdepartementet, 1 September 2005. http://www.regjeringen.no/upload/FD/Reglement/Policy_military_adaption_IKT.pdf

[7] OASIS, "Reference Model for Service Oriented Architecture 1.0", OASIS, OASIS Standard, 12 October 2006. http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf

[8] ISO, "Information Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model", International Organization for Standardization (ISO), ISO/IEC 7498-1, 1994. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=20269

[9] SEI, "How Do You Define Software Architecture?" Software Engineering Institute (SEI). http://www.sei.cmu.edu/architecture/definitions.html (last visited 2008).

[10] IEEE, "IEEE Std 1471-2000: IEEE Recommended Practice for Architectural Description of Software-Intensive Systems", IEEE, October 2000.

[11] NATO, "NATO Glossaries", NATO. http://www.nhqc3s.nato.int/NATOGL/glossary.asp (last visited 2008).

[12] FLO/IKT, "Forsvarets tekniske arkitektur for integrasjonsmellomvare", 2008.

[13] A. S. Tanenbaum, "Computer Networks", 3rd ed., New Jersey, Prentice-Hall, Inc., 1996, ISBN: 0-13-394248-1.

[14] ISO, "Software engineering - Product quality - Part 1: Quality model", International Organization for Standardization (ISO), ISO/IEC 9126-1, 2001. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749

[15] The Sox Institute, "What an IT manager needs to know about Eurosox". http://www.grccontrollers.com/files/IT-Security_UK_20070916.pdf (last visited 2008).

[16] WS-I, "Basic Profile Version 1.2", Web Services Interoperability (WS-I) Organization, 28 March 2007. http://www.ws-i.org/Profiles/BasicProfile-1.2.html

[17] A. Clemm, "Network Management Fundamentals", 1st ed., Cisco Press, 2006, ISBN: 978-1587201370.

[18] IETF, "The Common Management Information Services and Protocols for the Internet (CMOT and CMIP)", Internet Engineering Task Force (IETF), October 1990. http://tools.ietf.org/html/rfc1189

[19] IETF, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", Internet Engineering Task Force (IETF), December 2002. http://tools.ietf.org/html/rfc3411

[20] IETF, "NETCONF Configuration Protocol", Internet Engineering Task Force (IETF), December 2006 2006. http://tools.ietf.org/html/rfc4741
[21] JCP, "JSR 255: Java Management Extensions (JMX) Specification, version 2.0", Java Community Process (JCP). http://jcp.org/en/jsr/detail?id=255 (last visited 2008).
[22] DMTF, "Web-Based Enterprise Management (WBEM)", Distributed Management Task Force (DMTF). http://www.dmtf.org/standards/wbem (last visited 2008).
[23] DMTF, "Common Information Model (CIM) Standards", Distributed Management Task Force (DMTF). http://www.dmtf.org/standards/cim/ (last visited 2008).
[24] ISO, "Systems and software engineering - System life cycle processes", International Organization for Standardization (ISO), ISO/IEC 15288, 2008. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=43564
[25] ISO, "Information technology - Security techniques - Code of practice for information security management", International Organization for Standardization (ISO), ISO/IEC 17799, 2005. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39612
[26] D. Chen and F. Vernandat, "Enterprise Interoperability: A standardisation View, Enterprise Inter-and-Intra Organisational Integration", K. Kosanke et al. (ed.), Kluwer Academic Publishers, 2002, pp. 273-282.
[27] D. Chen and F. Vernandat, "Standards on enterprise integration and engineering – A state of the art", International Journal of Computer Integrated Manufacturing (IJCIM), vol. 17, no. 3, pp. 235-253, 2004.
[28] IEEE, "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries", Institute of Electrical and Electronics Engineers, 1990.
[29] IDABC, "European Interoperability Framework for Pan-European eGovernment Services, Version 1.0", IDABC, 2004. http://europa.eu.int/idabc/en/document/3761
[30] IDABC, "European Interoperability Framework for Pan-European eGovernment Services, Version 2.0", IDABC, 2007. Forthcoming at http://ec.europa.eu/idabc/
[31] W3C, "Web Services Architecture", World Wide Web Consortium (W3C), W3C Working Group Note, 11 February 2004. http://www.w3.org/TR/ws-arch/
[32] W3C, "SOAP Version 1.2 Part 1: Messaging Framework", World Wide Web Consortium (W3C), W3C Recommendation, 24 June 2003. http://www.w3c.org/TR/soap12/
[33] W3C, "XML Schema Part 0: Primer Second Edition", World Wide Web Consortium (W3C), W3C Recommendation, 28 October 2004. http://www.w3.org/TR/xmlschema-0/
[34] W3C, "Web Services Description Language (WSDL) 1.1", World Wide Web Consortium (W3C), W3C Note, 15 March 2001. http://www.w3c.org/TR/wsdl
[35] BEA Systems, IBM, Microsoft, SAP AG, and Siebel Systems, "Business Process Execution Language for Web Services Version 1.1", 5 May 2003. ftp://www6.software.ibm.com/software/developer/library/ws-bpel.pdf
[36] OASIS, "UDDI Version 3.0.2", Organization for the Advancement of Structured Information Standards (OASIS), UDDI Spec Technical Committee Draft, 19 October 2004. http://uddi.org/pubs/uddi_v3.htm
[37] OASIS, "Web Services Security Standard", Organization for the Advancement of Structured Information Standards (OASIS), April 2004. http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss
[38] W3C, "OWL Web Ontology Language Overview", World Wide Web Consortium (W3C), W3C Recommendation, 10 February 2004. http://www.w3.org/TR/owl-features/
[39] NATO, "Standardization Agreements". http://www.nato.int/docu/standard.htm (last visited 2008).
[40] NATO, "NISP Standards & Profiles for Coalition Interoperability".

6 Appendix A: NATO Architecture Framework (NAF) general terminology

The table below lists some important terms relevant for the work on technical architectures.

Term	Definition
Architect	The person, team, or organisation responsible for systems architecture.
Architecting	The activities of defining, documenting, maintaining, improving, and certifying proper implementation of an architecture.
Architectural description	A collection of architectural products to document an architecture.
Architectural product	A model representing an aspect of an architecture.
Architecture	The fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.
Aspect	A coherent and consistent set of characteristics of a system as seen from a given viewpoint.
Business process	A set of logically related tasks performed to achieve a defined business outcome.
Business rule	A statement describing a business policy or decision procedure.
Capability	The ability of one or more resources to deliver a specified type of effect or a specified course of action
Data	A reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing.
Data element	A unit or class of information which has a unique meaning and may embrace data items of distinct units or values.
Information	The knowledge concerning objects, such as facts, events, things, processes or ideas including concepts that within a certain context have a particular meaning.
Infrastructure	A part of a system necessary for the support of the system and which enables the system to perform its function.
Interoperability	The ability to operate in synergy in the execution of assigned tasks.
Metamodel	A model which describes a model.
Model	An abstraction of a real-world object or phenomenon.
Object	A physical or conceptual entity that may have one or more properties.
Process	A predetermined course of events defined by its purpose or by its effect, achieved under given conditions.
Service	A function, capability or behaviour that is provided by a producer to a consumer.
Stakeholder	An individual, team, or organisation (or classes thereof) with interests in, or concerns relative to, a capability.
Subview	A pattern from which to develop individual products by establishing the purposes and audience for a product and the techniques for its creation and analysis. (Note that this is called "viewpoint" in IEEE 1471-2000.)
System	A collection of components organised to accomplish a specific function or set of functions.
View	A set of subviews grouped by purpose.

7 Appendix B: NATO Architecture Framework (NAF) views and subviews

The table below lists all the views and their subviews defined in NAF v3.

View	Subview	Name	Description
NATO All View (NAV)	NAV-1	Overview and Summary Information	Provides executive-level summary information in a consistent form that allows quick reference and comparison between architectures.
	NAV-2	Integrated Dictionary	Define the terms used to describe the architecture.
	NAV-3a	Architecture Compliance Statement (Metadata)	To certify that the architecture meets the requirements of the NAF.
	NAV-3b	Metadata Extensions	To document any deviations of the architecture's subviews from the standard subview guidelines of the NAF.
NATO Capability View (NCV)	NCV-1	Capability Vision	Provide a strategic context for the capabilities described in the architecture.
	NCV-2	Capability Taxonomy	Provides a structured list of capabilities and sub-capabilities that are required within a capability area during a certain timeframe.
	NCV-3	Capability Phasing	Provides a representation of the available military capability at different points in time or during specific timeframes.
	NCV-4	Capability Dependencies	Describes the dependencies between capabilities. It also defines logical groupings of capabilities.
	NCV-5	Capability to Organisational Deployment Mapping	Shows deployment of resources in general, and systems specifically, in NATO Operational Commands and the ability between those resources to satisfy the military capability for a particular timeframe (or Epoch).
	NCV-6	Capability to Operational Activities Mapping	Describes the mapping between capability elements and the operational activities that those capabilities support.
	NCV-7	Capability to Services Mapping	Describes the mapping between capabilities and the services that these capabilities enable.
NATO Operational View (NOV)	NOV-1	High-Level Operational Concept Description	Provide a quick, high-level description of the architecture, and its functionality.
	NOV-2	Operational Node Connectivity Description	Depicts operational nodes with needlines between those nodes that indicate a need to exchange information.
	NOV-3	Operational Information Requirements	To identify and describe all information exchanges that make up all information needlines between operational nodes
	NOV-4	Organisational Relationship Chart	Identifies the key players in the operational domain that is subject to the architecture effort, and illustrates the organisational relationships among these key players.
	NOV-5	Operational Activity Model	Describes the operations that are normally conducted in the course of achieving a mission or an operational objective. It describes operational activities (or operational tasks) and Input/Output (I/O) flows between activities.
	NOV-6a	Operational Rule Model	Specifies operational constraints on an enterprise, a mission, or an operation, business, or on an architecture.

	NOV-6b	Operational State Transition Description	Describe the explicit sequencing of operational activities, especially in terms of the operational activities' life-cycles.
	NOV-6c	Operational Event-Trace Description	Provides a time-ordered examination of the information exchanges between participating operational nodes as a result of a particular scenario.
	NOV-7	Information Model	To analyze the information aspects of the operational domain and to guide the design of information systems.
NATO Service-Oriented View (NSOV)	NSOV-1	Service Taxonomy	To organise knowledge according to the service perspective, and to facilitate harmonization of services across multiple domains (or across multiple architectures).
	NSOV-2	Service Definitions	To strictly delineate and define services in order to understand the operational domain in terms of services supporting operational activities.
	NSOV-3	Services to Operational Activities Mapping	To provide traceability by illustrating which services support which operational activities.
	NSOV-4	Service Orchestration	To identify and describe how services in general3, and web services in particular, are utilized in the execution of operational activities, and how services are used, in conjunction, to support operational processes.
	NSOV-5	Service Behaviour	To specify the function and behaviour of individual services.
NATO Systems View (NSV)	NSV-1	System Interface Description	To illustrate which systems collaborate in which way to support the operational domain's information and information exchange needs as defined in the Operational View
	NSV-2	Systems Communication Description	To provide a comprehensive specification of how systems are connected at a detailed infrastructural level, what interfaces each system exposes (ports), the hardware interfaces used, and the protocols that govern transmission of data across the interface.
	NSV-2a	System Port Specification	To show the ports of a system and the protocols supported by each of those ports.
	NSV-2b	System to System Port Connectivity	To specify the physical, infrastructural nature of a connection between two systems.
	NSV-2c	System Connectivity Clusters	To define the connectivity requirements between nodes, and is used for estimating requirements for physical routing and bandwidth.
	NSV-2d	Systems Communication Quality Requirements	To specify specific quality requirements applicable to communications between systems.
	NSV-3	Systems to Systems Matrix	Provides detail on the interface characteristics described in the NSV-1 subview for the architecture, arranged in matrix form.
	NSV-4	System Functionality Description	Supports the development of system functional hierarchies and system functions.
	NSV-5	Systems Function to Operational Activity Traceability Matrix	Depicts the mapping of operational activities to system functions and thus identifies the transformation of an operational need into a purposeful responsibility assigned to a system.
	NSV-6	Systems Data Exchange Matrix	Specifies the characteristics of the system data exchanged between systems.
	NSV-7	System Quality Requirements Description	To communicate which quality characteristics are considered most crucial for the successful achievement of the mission goals assigned to the system.

	NSV-8	Systems Evolution Description	Describes plans for modernizing systems over time.
	NSV-9	Technology Forecast	To identify relevant emerging technologies, and to ensure that the architecture benefits from it, or is easily adapted to it.
	NSV-10	Systems Rules, Sequence & Timing Description	Dynamic behaviours concern the timing and sequencing of events that capture system quality characteristics of an executing system.
	NSV-10a	Systems Rule Model	To allow understanding of behavioural rules and constraints imposed on systems and system functions.
	NSV-10b	Systems State Transition Description	To describe the explicit sequencing of system interactions, using states and state transitions, brought on by triggers and events.
	NSV-10c	Systems Event-Trace Description	Provides a time-ordered examination of system data exchanges between participating systems (external and internal) or human roles, as a result of a particular scenario or situation.
	NSV-11	System Data Model	To enable analysis, design and implementation of the data presentation, handling and storage functionality of an information system.
	NSV-11a	Logical Data Model	Directly reflects the paradigm or theory oriented mapping from the information model to the data model.
	NSV-11b	Physical Data Model	Specifies how the logical data model will be instantiated in a particular product.
	NSV-12	Service Provision	To illustrate which systems contribute to the provision of which services.
NATO Technical View (NTV)	NTV-1	Technical Standards Profile	Provides a list of standards guiding and constraining the implementation of systems as defined in the various subviews of the NATO System View.
	NTV-2	Technical Standards Forecast	To identify emerging, obsolete and fragile standards, and to assess their impact on the architecture and its constituent elements.
	NTV-3	Standard Configurations	To describe standard configurations that are applied to or emerge from the architecture effort, used or encountered in any of the subviews developed in the architecture effort.
NATO Programme View (NPV)	NPV-1	Programme Portfolio Relationships	Details the relationships among projects within a major NATO programme, such as the Bi-SC AIS programme.
	NPV-2	Programme to Capability Mapping	Intended primarily to support the acquisition and fielding processes, including the management of dependencies between projects and the integration of all relevant project and programme elements to achieve a capability as defined by in NATO capability package (CP).

8 Appendix C: IEEE Std 1471-2000 for architectural descriptions

The IEEE 1471-2000 recommended practice for architectural description [10] include definitions of important terms and relate these in a conceptual model of architectural description. According to the recommended practice, software-intensive systems are those complex systems "where software contributes essential influences to the design, construction, deployment and evolution of the system as a whole". The purpose of IEEE Std 1471-2000 is to facilitate the expression and communication of architectures and thereby lay a foundation for quality and cost gains through standardisation of elements and practices for architectural description of software-intensive systems. Figure 9 shows the conceptual model of architectural description as defined in IEEE Std 1471-2000.

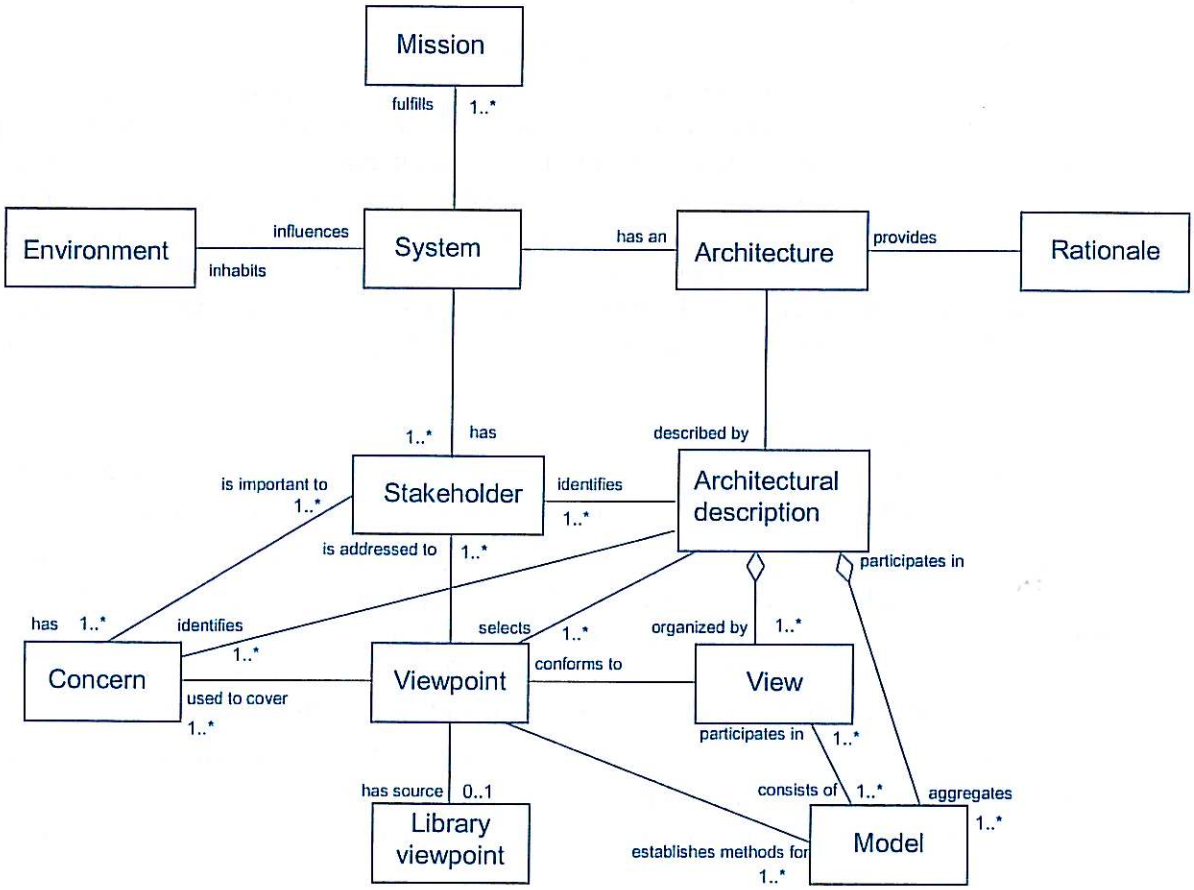

Figure 9: IEEE 1471-2000 conceptual model

Starting with *system*, it is defined to be "a collection of components organized to accomplish a specific function or set of functions." For the purposes of the recommended practice, "the term system encompasses individual applications, systems in the traditional sense, subsystems, systems of systems, product lines, whole enterprises, and other aggregations of interest." From this it follows that anything can be a system as long as it fulfils some purpose, i.e., accomplishes function(s), and one chooses to view it as a whole.

A system inhabits an *environment*, while the environment of a system can influence that system. The environment, sometimes referred to as the context, "determines the settings and circumstances of developmental, operational, political, and other influences upon that system. The environment can include other systems that interact with the system of interest, either directly via interfaces or indirectly in other ways. The environment determines the boundaries that define the scope of the system of interest relative to other systems".

Essentially, one draws a line between the system of interest and anything outside that system that influences it in some way. This line is the interface between the system and its environment. A system has one or more *stakeholders*. A stakeholder has one or more *concerns* relative to the system. Concerns are "those interests which pertains to the system's development, its operation or any other aspects that are critical or otherwise important to one or more stakeholders." Typical concerns a stakeholder can have relative to a system are functionality, performance, security, reliability, safety, etc.

A system exists to fulfil one or more *missions* in its environment. The existence of a system has a purpose; it should meet one or more objectives of one or more stakeholders. Often some of these objectives coincide with enterprise objectives so that using the system is an efficient use of resources in the enterprise.

A system has an *architecture* and this can be described in an *architectural description*. Note the distinction between the architecture of a system, which is conceptual, from the description of this architecture, which is concrete. Architectural description is defined as "a collection of products to document an architecture". The architectural description can be divided into one or several *views*. Each view covers one or more stakeholder concerns. View is defined as "a representation of a whole system from the perspective of a related set of concerns". A view is created according to rules and conventions defined in a *viewpoint*. Viewpoint is defined as "a specification of the conventions for constructing and using a view. A pattern or template from which to develop individual views by establishing the purposes and audience for a view and the techniques for its creation and analysis".

In addition to information described in views, an architectural description may contain other information such as system overview and system rationale. This information is not described according to a viewpoint definition, but may follow other organisational documentation practices.

An architectural description selects one or more viewpoints for use. This choice depends on the concerns of the stakeholders that need to be addressed by the architectural description. The IEEE Std 1471-2000 does not prescribe any particular viewpoints. A viewpoint may be defined with the architectural description, but it may also be defined elsewhere and only used in the architectural description. Such externally defined viewpoints are termed *library viewpoints*.

A view may consist of one or more *models* and a model may participate in one or more views. Each such model is defined according to the methods established in the corresponding viewpoint definition. The architectural description aggregates the models, organised into views.

9 Appendix D: OASIS reference model for SOA

The OASIS reference model for SOA [7] defines an abstract framework that can be used to understand significant entities and relationships between them within a service-oriented environment. It can be used to develop consistent standards or specifications supporting service-oriented environment. It is based on unifying concepts of SOA and may be used by architects to develop specific service-oriented architectures. The reference model not directly tied to any standards, technologies or other concrete implementation details.

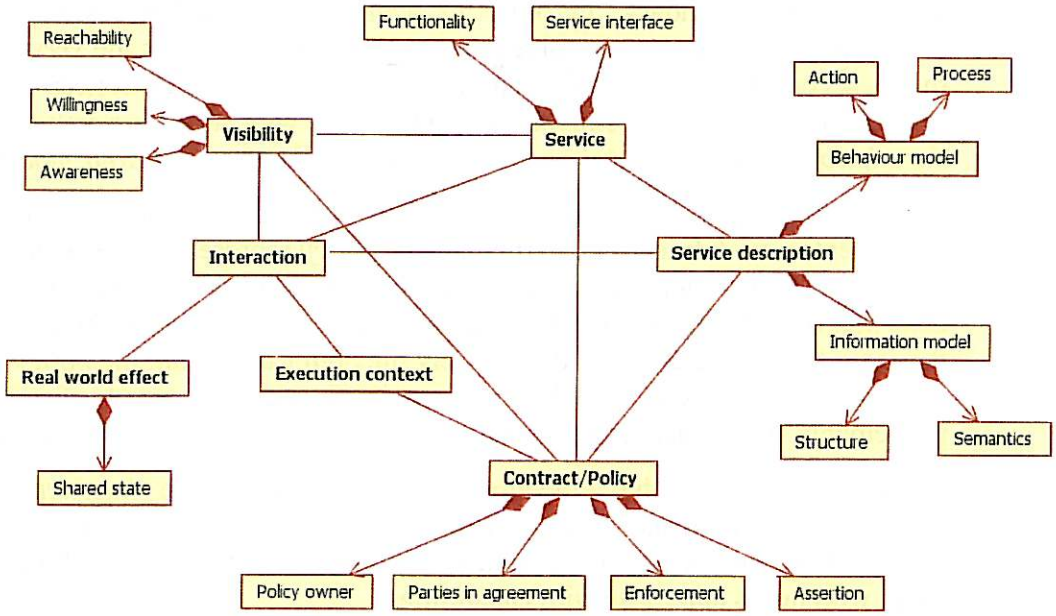

Figure 10: Conceptual model for technical architecture (service-oriented architecture)

The OASIS reference model defines SOA as *"a paradigm for organising and utilising distributed capabilities that may be under the control of different ownership domains"*. The reference model defines seven principal concepts, highlighted with bold font in Figure 10, as the basis for describing service-oriented architectures. Each of these concepts are elaborated below.

- *Service*: The means by which the needs of a consumer are brought together with the capabilities of a provider. A *service* offers one or many *functions* that result in *real world effects* if being invoked. 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 in the *behaviour model* and the *information model* of the *service description*.

- *Service description*: The information needed in order to use, or consider using, a *service*. The *information model* is a characterisation of the information that may be exchanged with the *service*, such as the *structure* and the *semantics* of the information. The *behaviour model* is a characterisation of *actions* invoked against the *service* and the *process* or temporal aspects of interacting with the service.

- *Visibility*: The capacity for those with needs and those with capabilities to be able to interact with each other. *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*, *willingness* and *reachability*. *Awareness* allows for service providers and service consumers to know of the other's existence and requires that the *service description* and *contract/policy* be available. *Willingness* is the intentional act to

initiate and to participate in a service *interaction*. *Reachability* is the relationship between service participants where they are able to interact though a communication infrastructure.

- *Real world effect*: The actual result of using a service, rather than merely the capability offered by a service provider. A *real world effect* can be the response to a request for information or the change in the *state* of some defined entities *shared* by the service participants.
- *Interaction*: The activity involved in making using of a capability offered, usually across an ownership boundary, in order to achieve a particular desired real-world effect. Interacting with a *service* involves performing actions against the *service*. In many cases, this is accomplished by sending and receiving messages. Key concepts that are important in understanding what it is involved in interacting with *services* revolve around the *service description* which references an *information model* and a *behaviour model*.
- *Contract/Policy*: A statement of obligations, constraints or other conditions of use of an owned entity as defined by a participant. 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. Conceptually, there are three aspects of *policies* namely the *policy assertion*, the *policy owner* and *policy enforcement*. Whereas a *policy* is associated with the point of view of individual participants, a *contract* represents an *agreement* between two or more participants.
- *Execution context*: The *execution context* of a service *interaction* is the set of infrastructure elements, process entities, *policy assertions* and *agreements* that are identified as part of an instantiated service *interaction*, and thus forms a path between those with needs and those with capabilities.

10 Appendix E: Norwegian Defence reference model for the information infrastructure

The document "Policy for military adaptation and use of information and communication technology in the Norwegian Armed Forces" [5] defines a reference model called the *information infrastructure (INI)* and contains a brief explanation of the services defined in this reference model. Appendix E classifies the identified services of the INI according to the structure proposed in this report. Please note that the INI reference model is currently under revision.

Classification	Service	Description
Functional (decision support) services	Command, control and management services	Services for planning, management and control of Armed Forces activities. For example, services for the development of plans, orders and missions, as well as for simulation and analysis.
	Manouver services	Services for conduct of military activities, i.e., in support of the various forms of operation (land operations, air operations, maritime operations, amphibious operations, air and missile defence, information operations, special operations and crisis management).
	Intelligence and surveillance services	Services for building relevant operational pictures, for example intelligence, reconnaissance, surveillance and sensor control.
	Fire control services	Services for controlling and synchronising various types of fire. For example, services for localisation and target processing, target engagement, choice of weapon and effect analysis.
	Protection services	Services for NRBC, fortification and protection measures.
	Logistic services	Services for acquiring and maintaining combat capability (materiel).
	Personnel services	Services for recruiting, development, utilisation and discharge of personnel.
	Structural services	Services for planning, realisation and evaluation of organizational structures.
	Works services	Services for management of defence property, buildings and installations. For example services to support the establishment and unrigging of camps.
	Financial services	Services for pay and accounting.
	Ad hoc adapted services	This type of service is included in order to indicate that we must have flexibility to assemble specially adapted groups of services to meet operational needs as they arise.
(Common) core services	Geographic services	Services for administration and use of geographic information. For example, a map engine capable of displaying military symbols, overlay handling and basic tracking.
	Collaboration services	Services for audio and video telephony and other online interaction.
Information fabric (services)	Information exchange	Standards and solutions for information exchange nationally, with allied forces and coalition partners and with other appropriate civil agencies. Examples of this type of service include military message handling, e-mail, data links and replication.
	Information management	Services for the capture, storage, fusion and correlation, recovery and use of information.

Management (services)	Registry services	Administration and provision of the services in the information structure, for example a look-up service ("electronic yellow pages").
	Service management	Services such as system monitoring, availability assurance and various kind of service desks.
Security (services)	Secure platforms	Secure runtime environments with standard support tools (for example FISBasis Secret/NATO Secret and FISBasis Restricted/Unclassified).
	Information security	Public Key Infrastructure (PKI), IP encryption and other types of service to ensure confidentiality, integrity and availability.

11 Appendix F: OSI and TCP/IP reference models for networking

The communication infrastructure can be divided into a set of layers. The two most common reference models for structuring these layers of the communication infrastructure is the Open Systems Interconnection (OSI) reference model [8] and the TCP/IP reference model. The OSI model defines 7 layers and the TCP/IP model defines 4 layers as shown in Figure 11.

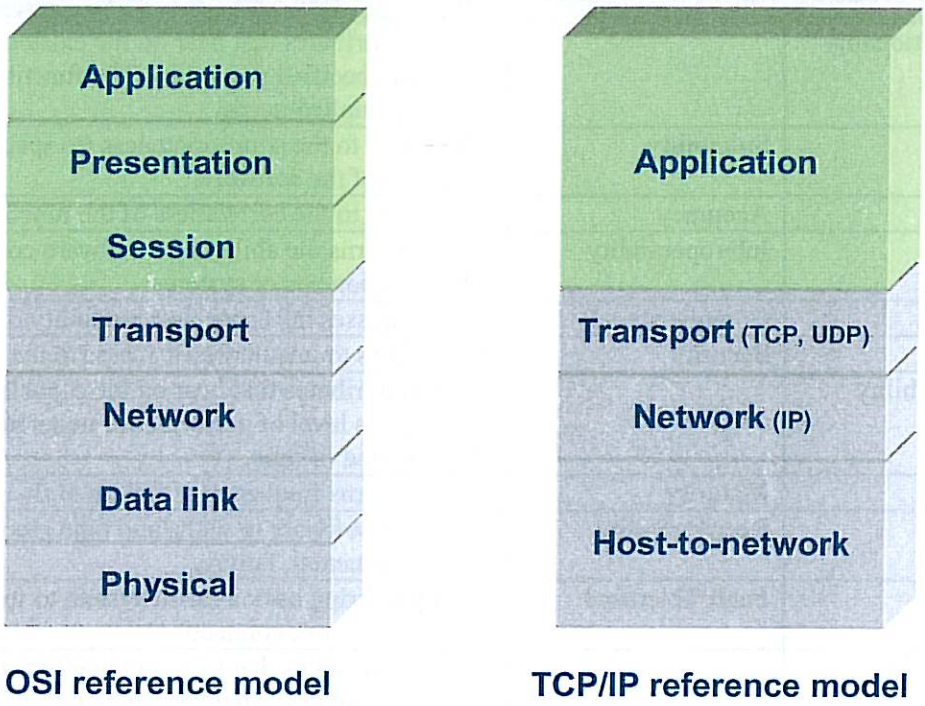

Figure 11: OSI and the TCP/IP reference model

The OSI and TCP/IP reference models have much in common. Both are based on the concept of a stack of independent protocols. Also, the functionality of the layers is roughly similar. The TCP/IP model does not have session or presentation layers. No need for them was perceived, so they were not included. Experience with the OSI model has proven this view correct as they are little use to most applications.

Three concepts are central to the OSI model, namely services, interfaces and protocols. The TCP/IP model did not originally clearly distinguish between these concepts.

- The *service* definition tells what the layer does, not how entities above it access it or how the layer works. A service is formally specified by a set of primitives (operations) available to a user or other entity to access the service.
- A layer's *interface* tells the processes above it how to access it. It specifies what the parameters are and what to results to expect.
- A *protocol* is a set of rules governing the format and meaning of the frames, packets, or messages that are exchanged by the peer entities within a layer.

Neither the OSI model and its protocols nor the TCP/IP model and its protocols are perfect. The OSI model is considered by many to be too complicated and to a large extent unimplementable, and was eventually eclipsed by the TCP/IP model. However, The TCP/IP model is not at all general and is poorly suited to describing any protocol stack other than TCP/IP. Furthermore, the host-to-network layer is not really a layer at all.

12 Appendix G: ISO/IEC 9126 quality model for software quality

The ISO/IEC 9126 quality model identifies 6 main quality characteristics, namely functionality, reliability, usability, efficiency, maintainability and portability. These characteristics are further broken down into sub-characteristics. The table below lists the characteristics defined.

Characteristic	Sub-characteristic	Description
Functionality		A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs.
	Suitability	This refers to the appropriateness (to specification) of the functions of the software.
	Accuracy	This refers to the correctness of the functions.
	Interoperability	This concerns the ability of a software component to interact with other components or systems.
	Compliance	This addresses the compliant capability of software.
	Security	This relates to unauthorized access to the software functions.
Reliability		A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time.
	Maturity	This concerns frequency of failure of the software.
	Recoverability	Ability of software to withstand (and recover) from component, or environmental, failure.
	Fault Tolerance	Ability to bring back a failed system to full operation, including data and network connections.
Usability		A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.
	Learnability	Learning effort for different users, i.e., novice, expert, casual etc.
	Understandability	Determines the ease of which the systems functions can be understood, relates to user mental models in Human Computer Interaction (HCI) methods.
	Operability	Ability of the software to be easily operated by a given user in a given environment.
Efficiency		A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions.
	Time Behaviour	Characterises response times for a given throughput, i.e., transaction rate.
	Resource Behaviour	Characterises resources used, i.e., memory, CPU, disk and network usage.
Maintainability		A set of attributes that bear on the effort needed to make specified modifications.
	Stability	Characterises the sensitivity to change of a given system that is the negative impact that may be caused by system changes.
	Analyzability	Characterises the ability to identify the root cause of a failure within the software.
	Changeability	Characterises the amount of effort to change a system.
	Testability	Characterises the effort needed to verify (test) a system change.
Portability		A set of attributes that bear on the ability of software to be transferred from one environment to another.
	Installability	Characterises the effort required to install the software.
	Replaceability	Characterizes the plug and play aspect of software components, that is how easy is it to exchange a given software component within a specified environment.

	Adaptability	Characterises the ability of the system to change to new specifications or operating environments.
	Conformance	Similar to compliance for functionality, but this characteristic relates to portability.