NATO Communications and Information Agency
Agence OTAN d'information et de communication

IKM TOOLS

IFB-CO-15079-IAS

BOOK II - PART IV SOW Annex A

SYSTEM REQUIREMENTS SPECIFICATION (SRS)

Document History

Edition	Date	Description
1.0	04/10/2019	Export from DOORS repository

Table of Contents

Figures

Tables

1 Introduction

1.1 Purpose

This System Requirement Specification (SRS) describes the external behaviour of the system to be delivered under the IKM Tools project. It also describes non-functional requirements, design constraints and other factors necessary to provide a comprehensive description of the requirements for the system.

1.2 Scope

This SRS supports the NATO Capability Package (CP) 9C0150 Core Information Services for Command and Control that identifies the capability and resources required to provide information services that need to be accessible to all users, regardless of Community of Interest (COI). The Capability Package reflects the minimum requirement to support the command and control of all military functions.

The basic SRS as contained in the requirement management tool (DOORS) contains the currently identified set of requirements for the IKM Tools (NIP, EDMS and TT+) covered by:

- Project CP9C0150 0IS03095 (“Provide Information Administration Services”), which covers the design of an initial enterprise-level core data management service capability to the NATO Command Structure that is consistent with the requirements of Deployable CIS, development and implementation across the enterprise of the initial capability to provide workflow and workspace services on the NATO Secret (NS) and NATO Restricted (NR) networks;

1.3 Background

The purpose of this project is to deliver the underlying IKM Tools that will enable NATO to rapidly deliver integrated information systems. The IKM Tools will allow other projects to use consistent, coherent and proven solutions to common problems, allowing them to focus on delivering real business value to NATO. Together with the Infrastructure as a Service (IaaS) delivered by the ITM project, this project will change the way NATO builds and procures functional area services (FASs) and user applications. The reuse of existing services and components will ensure that NATO achieves a greater return on investment on information systems. At the same time, the ability of different Communities of Interest to share and distribute information in a controlled way to a broader range of partners will ensure that the Alliance meets its “Responsibility to Share”.

1.4 Acronyms and Abbreviations

The acronyms and abbreviations used in this SRS are defined in Annex E of the Statement of Work.

1.5 Definitions

The definitions used in this SRS are defined in Annex E of the Statement of Work.

1.6 References

The abbreviated document titles given in square brackets, […], are used to refer to documents in the reference list.

1.7 Overview

This SRS comprises 4 sections:

- Section 1 provides an introduction and the use of this document.
- Section 2 provides an overall description of the IKM Tools Services, roles involved and project constraints.
- Sections 3 and 4 contain the Functional and Non-Functional Requirements comprising the required functionality and their associated requirement attributes.

During the system requirements analysis and design (Design Stage), further elaboration of IKM Tools Information Objects will be required, and, almost certainly, additional Information Objects, products, attributes and relationships will be identified to implement the requirements.

1.8 SRS Conventions

The system requirements, defined in this document, are individually identified by a unique number which shall be used at all times as the specific reference for each.

No meaning is associated with the order of serial numbering. There could be gaps in numbering and requirement identifiers in a group do not have to be sequential.

Requirement identifiers are encapsulated in square brackets.

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [IETF RFC 2119, 1997].

1.9 Standards and Specifications

Changes to identified standards and specifications (e.g., Interface Control Documents, STANAGs) are anticipated prior to the deployment of the IKM Tools capability. In cases in which changes to the standards/specifications are not major and are identified to the Contractor prior to the end of the Detailed Design Phase, no additional cost shall accrue to the change in standard/specification. Changes which shall not be considered major include those in which:

- The same technology is used (e.g., Web Services)
- There is no significant increase (< 20%) in the number of information objects/attributes supported
- There is no significant increase (< 20%) in the number of classes/methods supported
- There is no significant increase (< 20%) in the number of required settings (e.g., security settings)
- No specialised COTS hardware or software components are required to implement the changes

Some given references are for draft versions of documents (e.g. "Edition 4 Ratification Draft"). Where this is done, the Contractor shall use the draft document as given with the expectation that it shall be ratified without any substantial change.

Changes in standards/specifications as part of a normal technological GEN/maintenance path (e.g., Windows Vista to Windows 7, .NET Framework 3.5 to.NET Framework 4.0) shall not be considered to be major.

1.10 Verification Methods

The requirements in this SRS will be verified through qualification, herein defined as an endorsement with a guarantee and supporting documentation that the item being qualified satisfies the specified requirement(s). The different verification methods applicable to the requirements herein are described in the following paragraphs.

Note: In some cases, more than one verification method might be required in order to verify fulfilment of a requirement.

1.10.1 Inspection

Inspection is the visual examination of an item (hardware and software) and associated descriptive documentation. Verification is based on the human senses (sight, touch) or other means that use simple measurement and handling methods. No stimulus is necessary. Passive resources such as meter rule, gauge may be used.

For Non-Developmental Items (NDI), Modified NDI and Developmental Items, hardware inspection is used to determine if physical constraints are met, and hardware and/or software, inspection is used to determine if physical quantity lists are met.

1.10.2 Analysis

Analysis is the review and processing of design products (documentation, drawings, presentations, etc.) or accumulated data obtained from other qualification methods, such as manufacturer's tests of a product to be mass-produced, to verify that the system/component design meets required design criteria.

1.10.3 Testing

Testing is the operation of the system, or a part of the system, under controlled and specified conditions, generally using instrumentation, other special test equipment or specific test patterns to collect data for later analysis. This verification method usually requires recorded results to verify that the requirements have been satisfied.

2 General System Description

2.1.1

2.1 Product Perspective

2.1.1 Purpose of the IKM Tools

Intended Use

Requirement ID: IKM-SRS-1

The IKM Tools shall be enabled for the ON and the PBN with equivalent functionalities. The target audience are users in need of collaboration by

sharing information, executing tasks and alike cooperation activities accessible via web browser application.

Other users will be services and FAS applications that directly interact with the underlying IKM Tools information platform and as a result benefiting from the collaboration features IKM Tools provides.

Finally basic information exchange will be allowed between IKM Tools in different domains limited by the policies and technology available between Data Centers.

Verification Method: Test

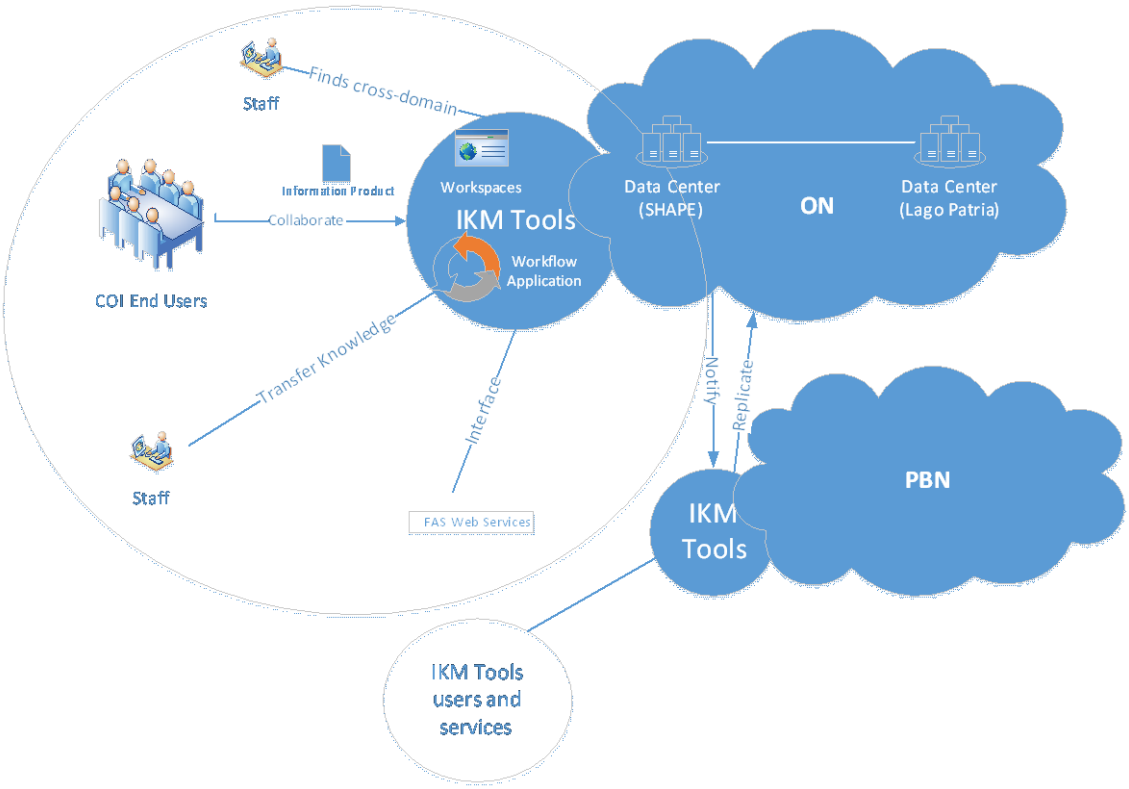


Figure 1 IKM Tools Operational Concept

2.1.1.2

Interoperability

The IKM Tools services will be interoperable with similar services in the deployed and federated domains, through clearly defined interfaces based on agreed open standards.

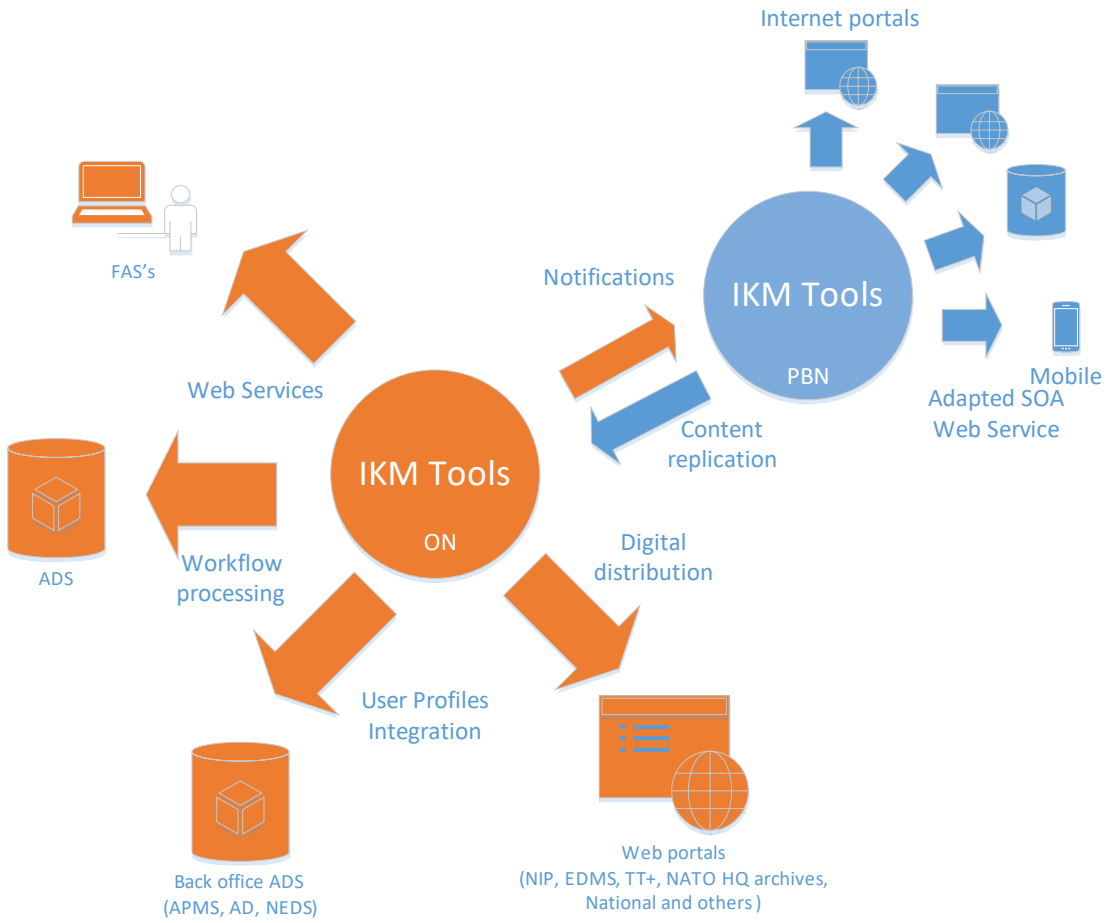


Figure 2 External Interfaces

IKM Tools will interface with similar systems in both ON and PBN networks. However in the ON with more restrictive access control to information.

In the PBN as it will use the ITM infrastructure it will be installed in the NR domain to be able to cross communicate with the IKM Tools in the ON.

The IKM Tools services will be consistent with the requirements of Deployable CIS (DCIS), see section NATO Bi-SC AIS Deployable CIS.

The implementation of comprehensive cross domain services is outside of the scope of this project. However, this project will leverage any existing border protection and cross domain services where these offer mechanisms for information exchange across security domains (both with NATO and external domains). This project will also provide the information products in the needed format for the cross-domain solution. The coordination between this project and CP 9C0150 Project 2012/0IS03102, which will implement additional cross-domain solutions, is within scope of this project.

This IKM Tools will support the integration of FASs and user applications via standard interfaces and API as web services, e-mail or File Transfer Protocol (FTP).

The IKM Tools may interface with NATO Nations CIS provided they implement standard protocols and interfaces as IKM Tools and that the border protection mechanism is configured.

Modes of operation

The technical solution to be provided for the IKM Tools will be able to support modes of operation including peacetime, crisis, and exercise, from the static locations through network connectivity.

Requirement ID: IKM-SRS-2

2.1.1.3

IKM Tools shall be deployable to the MIR and to disconnected network nodes.

Verification Method: Test

Mission Secret capacity provisioning is not within the scope of this project. Installation of the IKM Tools on the Deployable CIS provided by CP0A0149 is outside of the scope of this Project.

2.1.2 Scope

An internal logical architecture of the IKM Tools is presented in Figure X.

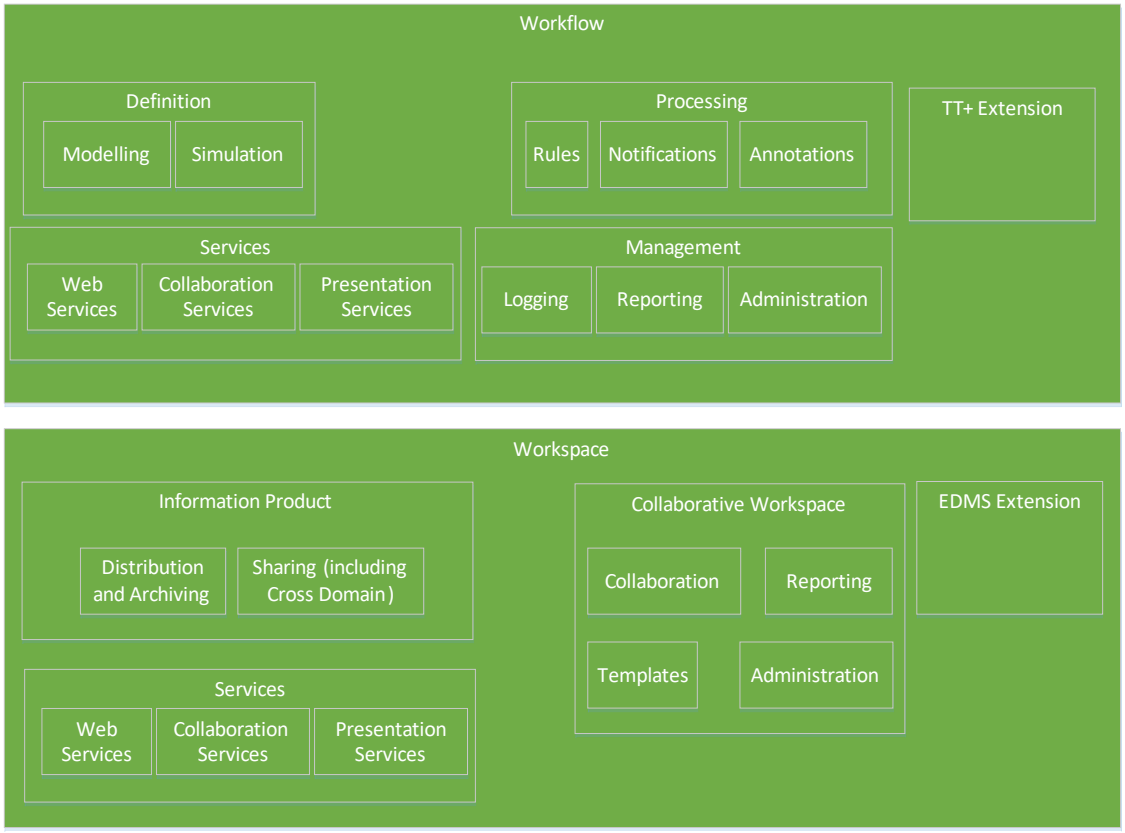


Figure 3 NSOV-1 Service Taxonomy

The main categories in this logical architecture are:

The **Workflow:** consists of all components IKM Tools will develop to enable a workflow capability for process automation. Dependant components are not included like IKM Tools Framework Runtime or Security modules that will be delivered by other projects.

Supports the synchronisation of directory data between the different NATO directories

The **Workspace:** consists of all components IKM Tools will develop to provide collaboration workspaces for user cooperative work. As the Workflow category it does not contain dependant modules.

2.1.3 Users and Roles

The IKM Tools will provide a wide range of functionality and benefits to various users:

NATO End Users will interact directly with the IKM Tools services via the Graphical User Interface (GUI), typically using a web browser. The IKM Tools will expose the services in a friendly and consistent manner to improve access to information and increase user mobility. At the same time it'll hide the complexity of the underlying services and will show a modern, industry-standard user experience (UX).

Solution Architects and Developers will indirectly interact with the IKM Tools services, designing their systems or FASs to integrate with IKM Tools and benefiting from its functionality. The ability to leverage existing IKM Tools services will mean that new portals will be quicker and more cost-effective to develop and integrate within NATO CIS. At the same time it will standardized IKM Portals with the inherent benefits of sharing similar UI, technical knowledge and support; resulting in an overall increase of Return On Investment (ROI) of NATO systems.

External Partners will benefit from federation capabilities offered by the IKM Tools services. There will be a consistent way to integrate federated systems, so that information extends across NATO and partner networks (NATO Enterprise, Alliance Enterprise and Coalition Enterprise). The IKM Tools will provide a common web platform delivering IKM services, both military and CIMIC. This will provide extensive information superiority (“Responsibility to Share”), while simultaneously ensuring that information is released only to those with a “Need to Know”.

And the user roles envisaged in the IKM Tools aligned with the current NIP roles are:

Business role: to provision IKM services like workflows and workspaces

IKM Manager role: to control and govern the information in the IKM Tools by setting up managed metadata, taxonomies, content types, policies for retention and disposition, naming conventions and all related IKM management activities.

Operational role: the ones to operate the IKM Tools day to day, as End User by acceding and using the IKM Tools user facing features or as an Administrator with higher privileges to manage and maintain the technical functions of IKM Tools.

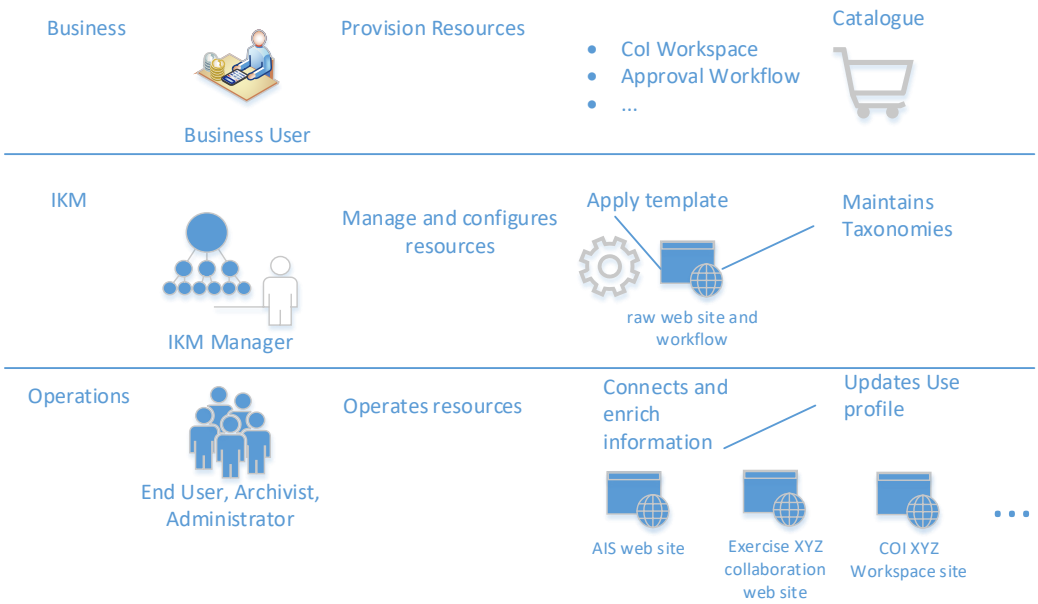


Figure 4 Roles in IKM Tools

2.2 General Constraints

2.2.1 Relationships with other programs

The IKM Tools is part of the Bi-SC AIS as defined in the Bi-SC AIS Reference Architecture [NAC AC/322-D(2005)0037, 2005].

The Bi-SC AIS is NATO's Command and Control Information System used throughout the NATO Command Structure, in NATO Command Deployments and in NATO Exercises. The Bi-SC AIS is in turn one element of NATO's overall CIS Capability, which includes a number of strategic sub-systems such as the NATO IKM Tools Framework (aka NIP Core), Communications Network, Planning, Intelligence, Theatre Missile Defence, and Deployable CIS and so on.

Requirement ID: IKM-SRS-2b

The capability shall comply with the NATO Policies [C-M(2015)0041-REV2].

The capability to be acquired under the Capability Package CP 9C0150 is to be a fully integrated element of the Bi-SC AIS.

In this context, the IKM Tools will co-exist and may interact with Intelligence Functional Services (Intel-FS), Logistics Functional Services (LOG FS), Maritime Functional Services (MCCIS), Land Functional Services (LC2IS), Air Functional Services (AirC2IS), Planning Functional Services (TOPFAS) and other functional services (such as Common Operational Picture, JCOP or NCOP; or Joint C2 Functional Services, JC2IS) as they become available.

2.2.2 Existing Capabilities

Related Projects

2.2.2.1.1 IT Modernisation (ITM)

The ITM project will transform the way IT services are provided to users across the NATO enterprise, including the NCS, the NATO Headquarters (NHQ) and NATO agencies. It will provide Infrastructure as a Service and an Enterprise Service Management and Control Service. ITM is the amalgamation of the three CP 9C0150 Projects: OIS03091; OIS03092, and OIS03101. The IKM Tools will be primarily deployed in the Data Centres delivered by the ITM project. This means that the IKM Tools will be able to take advantage of the ITM infrastructure, thus reducing the amount of Hardware that will need to be specifically procured.

2.2.2.1

The IKM Tools project relies on the timely delivery of the IT infrastructure by the ITM project.

The IKM Tools project will need to integrate the Service Management tooling with the SM&C framework as provided by ITM.

2.2.2.1.2 NIP

The NATO Information Portal (0IS03031 under CP 5A0050/9B0020) is the official NATO portal in ON domain and available to the majority of the NCS Commands. It provides Enterprise Search Services, Portal Services and Metadata Services (e.g. NIP term store).

Requirement ID: IKM-SRS-3

These mentioned services are to be used whenever possible by IKM Tools. In addition IKM Tools shall take into account these services consistency when proposing new IKM Tools GUI design and operatively.

Verification Method: Analysis

2.2.2.1.3 Email and IKM resilience (UR 2015)

Current IKM tools (NIP, EDMS, TT+) and Email are known to run on single location instance installations that include Single Point of Failure components, one Data Center. Potential outages/disasters could impact business for a significant time. NCIA has started a project to provide an interim resilience until ITM delivers the wave 3 (e.t.a. 2021).

2.2.2.1.4 IAS Step 1: EDMS and TT+

The IAS Step 1 has implemented the EDMS and TT+ applications hosted in the IKM Tools Framework Share Point infrastructure. Around 20 Commands are in use of these applications.

The EDMS was an upgrade of previous the DHS application and provides a Document Management System for the Commands, to work with documents organized in different Organizational Units (OU) as Site Collections.

The IAS Step 2 is to improve some missing EDMS features and to interact with it to enhance the workspace capabilities.

The TT+ was the update of the previous TTE and provides a staff tasking functionality where staff are assigned tasks and collaboratively contribute in workspaces to fulfil them.

The IAS Step 2 is to improve the TT+ with extra features.

2.2.2.1.5 NATO Enterprise Directory Services (NEDS)

The NATO Enterprise Directory Services will provide Directory synchronisation between the various NATO directory and data repositories. The IAS Step 2 project is intended to further integrate the NEDS system via Microsoft Identity Manager (MIM) service, with additional data sources (e.g. NPKI and Physical Access Control Systems) and to deploy it onto to the PBN.

The IAS Step 2 project relies on the NEDS project to deliver the system according to its current scope (i.e. ON).

2.2.2.1.6 NATO Public Key Infrastructure

The NPKI project will deploy a PKI environment for NATO that fully complies with the NPKI Certificate Policy [NAC AC/322-D(2004)0024-REV2, 2008].

The IKM Tools project will provide a number of services, including the integration into the authentication and authorisation services that depend on the implementation of a PKI capability across the NATO Enterprise.

2.2.2.1.7 FAS implementation projects

It is expected that the majority of FASs will take advantage of the services offered by the IKM Tools. This will mean that all FASs will have an interdependency with this project. The components and services that are provided will be an integral part of new FASs.

The fact that the implementations of different FASs that could make use of IKM Tools services have already started challenges the IKM Tools project. This issue will be addressed in several ways:

Projects that are supposed to be implemented in several increments, can postpone usage and integration with IKM Tools services until future increments.

Projects can implement FASs compatible with standards and interfaces identified for the IKM Tools [NAC AC/322-N(2011)0205, 2011], but postpone their use until the availability of IKM Tools. When services provided by the IKM Tools become available, the FASs will be reconfigured to make use of them. Because of implementation of the same standards and well defined interfaces, no additional implementation effort will be required. Detailed descriptions of selected interfaces to be provided by the IKM Tools are available in form of SIPs [NAC ADatP-34(G)-REV1, 2013].

Projects can implement their own IKM Tools services, compatible with the existing IAS Step 1, with the intention of replacing them with the enterprise IKM Tools services, when they become available.

In limited cases, projects will not wait for the IKM Tools but implement their own alternative solutions, often using different technology or based on a dedicated Information Knowledge Management Service.

2.2.2.1.8 Deployable CIS (CP0A0149)

Requirement ID: IKM-SRS-4

The IKM Tools shall be consistent with the requirements of Deployable CIS (see section 4.3.2.2).

Verification Method: Analysis

2.2.2.1.9 Provide Information Exchange Services (2012/0IS03102 under CP 9C0150)

The IAS Step 2 project will leverage any existing border protection and cross-domain services where these offer mechanisms for information exchange across security domains (both with NATO and external domains).

The IAS Step 2 project will coordinate in that respect with the CP 9C0150 Project 2012/0IS03102, which will implement additional cross-domain solutions.

Solution/Development Constraints

2.2.2.2 The IKM Tools will provide the framework upon which many FASs will be built. Consequently the system will need to support the concept of multi-tenancy, where multiple users, processes and activities can access the services simultaneously, while being logically isolated from each other.

In accordance to NATO policy [NAC AC/317-D/71 (Revised), 1996] COTS software should be used in preference to the development of new software unless it evidently contains major disadvantages (e.g. cost effectiveness, negative impact on existing IT infrastructure, security, market stability).

The implementation of physical components on special purpose networks (exercises/training/mission) is outside the scope of this project. These networks will benefit from the services provided under this project through network connection, in accordance with security policy.

Implementation of comprehensive cross domain services are outside of the scope of this project.

Implementation of components in Nations, if required for interfacing purposes, is out of scope of this project.

2.2.3.1

2.2.3 Operating Environment

Node Types

The IKM Tools services will be installed on computing nodes provided by the ITM as depicted in Figure 5 Node types:

Data Centre Nodes – ITM project plans to deliver Data Centres to serve as Main Computing Facilities (MCF) The MCFs will provide the complete suite of IKM Tools services.

Enhanced Nodes (Standalone) will provide a scaled down set of IKM Tools services, with reduced HW footprint for situations where agreed service levels cannot be met through the MCFs. It is anticipated that there will be different service requirements for different types of Enhanced Nodes.

Standard Nodes will be receiving all their services from the MCFs. No IKM Tools services will be installed there.

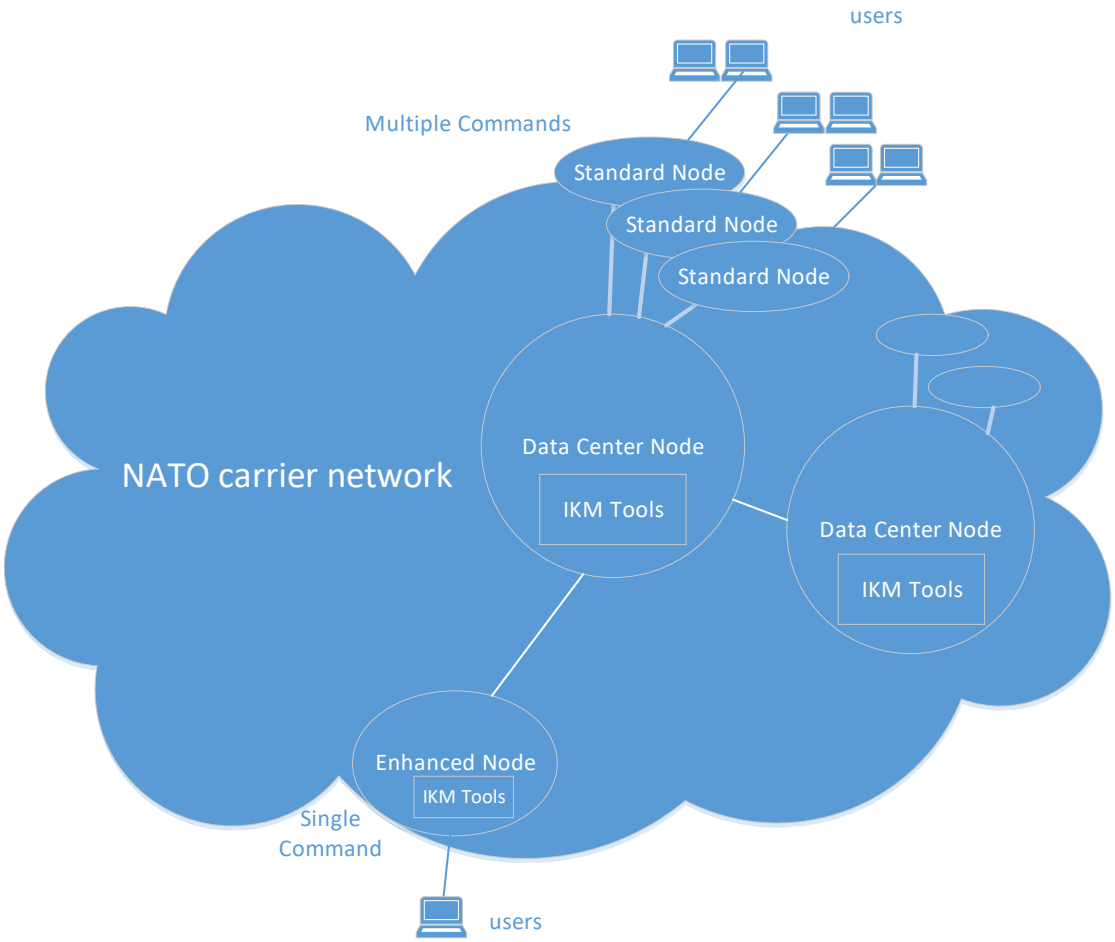


Figure 5 Node types

An **Integration and test facility**, as for example in the Programme Management and Integration Capability (PMIC) environments, will be provided to enable other projects already at their software development phase consistent integration with the IKM Tools services.

A **Reference facility** maintained by the NCI Agency will be used to ensure proper change management and evaluation before installation of NATO information systems integrated with the IKM Tools into operational environment. This will not be on the main ON and PBN networks themselves, but are expected to be located on equipment of the same specification in the Data Centres.

Deployment on the **DCIS** is explicitly out of scope for this project. However all services will be required to be able to run on the DCIS platform. The services that are actually provisioned on a DCIS node will depend on the nature of the mission involved. It is anticipated that a single DCIS node will vary between requiring the full suite of IKM Tools services (comparable to a Data Centre), selected IKM Tools Services (comparable to an Enhanced Node) or none (a Standard Node).

Operational Network and Protected Business Network Connectivity

The IKM Tools services will be installed for two major networks (as defined in [SHAPE 3050/SH/CCD CIS/CAR/335/13-301388, 2013]):

2.2.3.2 The PBN providing IT services at the NU and NR classification level in support of administrative business processes, appropriate operational processes and those processes requiring interaction over the Internet;

The ON providing IT services at NS level in direct support of war fighting processes, processes requiring higher levels of assurance and processes of military and political communications.

As presented in Figure 6, both networks will be implemented with the same logical components, but with separated physical components. The two physical infrastructures will be connected to support required cross-domain information exchanges compliant with, and within the limits of, applicable NATO security policy.

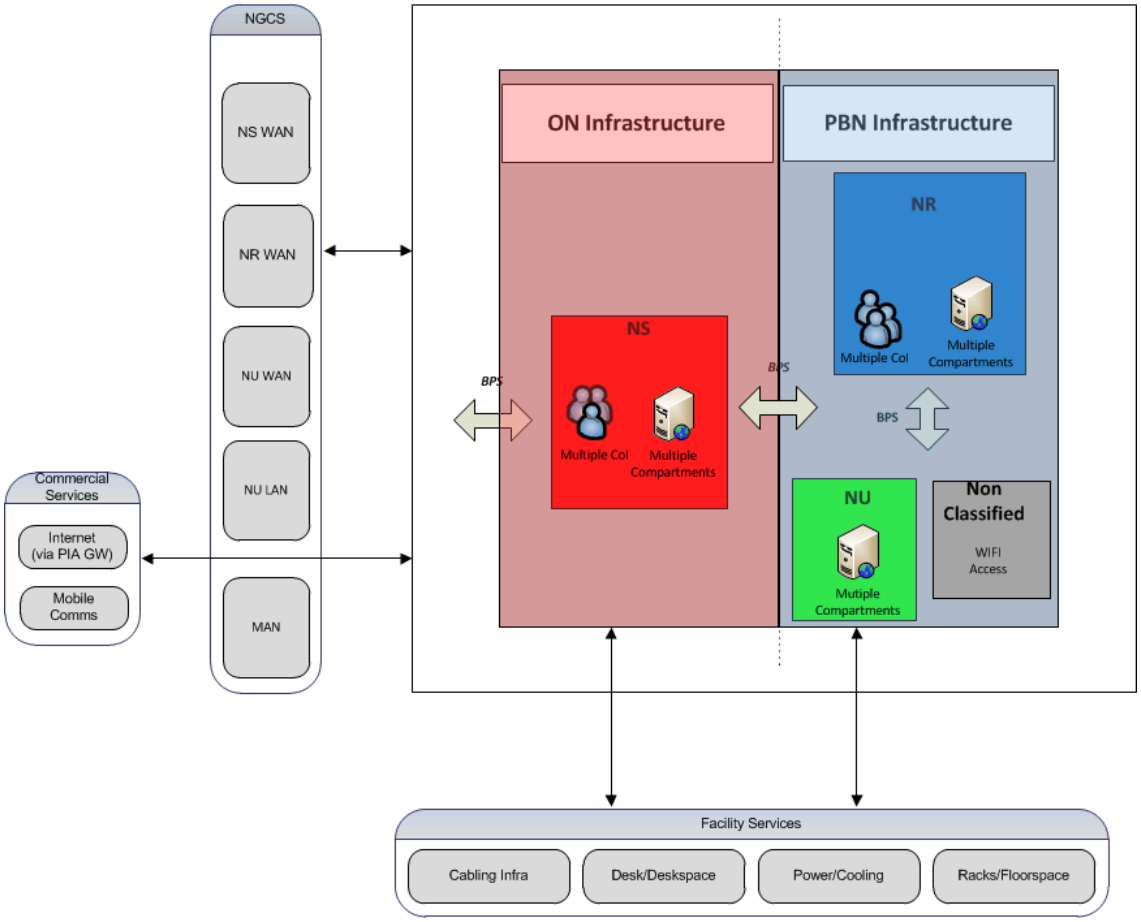


Figure 6 ON and PBN interconnectivity (ref. [CO-13703])

The IKM Tools will be hosted in the NS red box (NS) and in the Business blue box (NR). The communication will be through the Border Protections System (BPS) offered by ITM.

Initially there will be no IKM Tools in the NU green box, nor inside it in the DMZ area. So no Internet facing IKM Tools are foreseen.

The NS and Business domains will both have one security domain respectively so current multiple NS domains will be migrated into one in the target ITM NS. The same accounts for the NR/NU domains.

While ITM strives to unify all members of NATO enterprise into one Enterprise Network, federation capabilities must be implemented and maintained to allow federation of NATO Enterprise and NATO nations to support C4ISR services. Federation is essential to support Exercise Planning, Air Policing and other key NATO capabilities.

However this is the initial situation and it is expected to change over time by including other NATO domains and Nations. As an example, ITM out of scope sites that may need federation are:

- MCCE Eindhoven NLD (NS) hosted in DC Mons
- NFIU BGR/EST/HUN/LTU/LVA/POL/SVK (NS/NU) hosted in DC Mons
- COMMZ Thessaloniki GRC (NS) hosted in LP
- KFOR HQ Pristina RKS (NS) hosted in DC Mons
- MNCNE Szczecin POL (NS/NU) hosted in DC Mons
- MNCSE Bucharest ROM (NS/NU) hosted in DC Mons
- NECCCIS NOR/ISL/LTU

Therefore the IKM Tools shall be federated ready for the above eventuality.

2.2.3.3

Service Provision

From the business perspective the IKM Tools services will be provided through the NCIA Service Catalogue as any other service offered services in the Enterprise.

As from the technical perspective as the IKM Tools services will be used by other projects and systems it'll be needed to early assess the integration and compatibility testing. Therefore, integration testing will be conducted on the IV&V Test Facility and preparation for deployment (including Change Management and Change Evaluation) will be conducted within the Reference Facility.

The system will be centralised as much as possible, mostly at the Data Centres, and with a reduced footprint in the Enhanced Nodes as delivered by the ITM Project.

The Standard Nodes will receive all of their services from the Data Centre Nodes. In exceptional cases, services with special requirements (e.g. regarding availability or performance) that cannot be met when provided from the Data Centre Node will be provided locally by on-site installation, but typically still will be administered remotely.

2.3 Assumptions

It is assumed that the following projects have been delivered, or at least are available for integration to the IKM Tools project:

- ITM

- NPKI
- SOA and IdM
- IAS Step 1
- NIP

It is assumed that future NATO FASs and other information systems integrated by the IKM Tools will be centralized in the Data Centres provided by the ITM project. In exceptional cases, systems not installed in the Data Centres will have sufficient network connectivity (bandwidth and latency) to integrate through the IKM Tools.

The modifications of any other systems (beyond those covered by the pilots) are out of scope of this project.

3 Functional Requirements

3.1 General

3.1.1 Integration

Requirement ID: IKM-SRS-5

The IKM Tools shall leverage the ITM provided Chat and Presence Services, being Microsoft Skype for Business or Microsoft Teams.

Verification Method: Analysis

Requirement ID: IKM-SRS-6

The IKM Tools shall integrate with the ITM Share Point delivered platform, including with the ITM subservices for:

Verification Method: Test, Analysis

- User Spaces: MS My Sites and MS One Drive
- Backups and replication: Metalogix
- Security labelling: Titus Classification Enterprise Suite (Military Edition), ensuring the classification is consistent with the IKM Tools and within the file content
- Performance: Metalogix SharePoint Diagnostic Manager
- Service management: ITM SMC Service
- MS Office products (MS Word, MS Excel, MS Outlook, MS Visio): client and browser based (aka Office Online Server)

Requirement ID: IKM-SRS-7

The IKM Tools shall integrate with Microsoft Exchange Services for sending emails, events, meetings, notifications and appointments to any account within the NATO Enterprise.

Verification Method: Test

Requirement ID: IKM-SRS-8

The IKM Tools shall integrate with MIM for Identity Management

Verification Method: Test

Requirement ID: IKM-SRS-9

The IKM Tools shall integrate with Microsoft Active Directory (AD):

Verification Method: Analysis

- compatible with the Enterprise Directory Services Service Interface Profile (SIP)
- access via Secure LDAP

Requirement ID: IKM-SRS-10

The IKM Tools shall integrate with the 'NIP Core' application layer consisting of a set of templates, services, metadata, content types and COTS: Layer 2 for automatic metadata tagging and Sparqube for cross-site look up

Verification Method: Analysis

Requirement ID: IKM-SRS-11

The IKM Tools shall support authentication from external Identity Provider (IdP) to allow the following authentication mechanisms:

Verification Method: Analysis

- Category I: NATO users who can authenticate to the Active Directory;
- Category II: NATO users who cannot authenticate to the Active Directory but can present a SAML token issued by a trusted Security Token Service;
- Category III: NATO users who cannot authenticate to the Active Directory and cannot present a SAML token issued by a trusted Security Token Service;
- Category IV: Non-NATO users (e.g. national users) who cannot present a SAML token issued by a trusted Security Token Service.

Requirement ID: IKM-SRS-12

The IKM Tools shall support Microsoft Active Directory Federated Services (ADFS) Trusted Identity Provider (TIP).

Verification Method: Analysis

Requirement ID: IKM-SRS-13

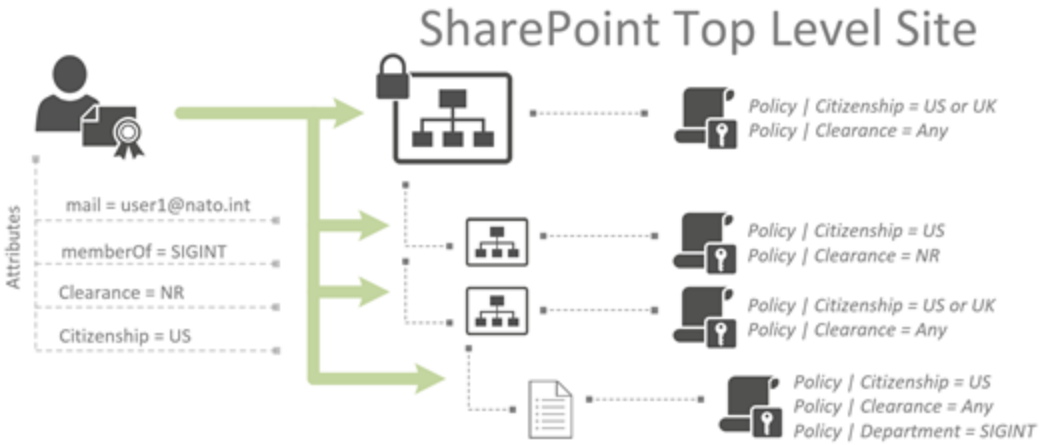
The IKM Tools shall be claims aware adhering to the Enterprise Single Sign On (ESSO) provided by ITM. So that users in a federated domain (e.g. using ADFS) can seamlessly access the IKM Tools without the need of re-authentication.

Verification Method: Test

Requirement ID: IKM-SRS-14

The Authorization mechanism shall relay on the SAML claim token attributes to access the web applications, Attribute Based Access Control (ABAC), and further coarse-grained by Role Based Access Control (RBAC) to access the Share Point sites. See following figure ITM Share Point External and Internal Authorization, see following diagram:

Verification Method: Test



Requirement ID: IKM-SRS-15

The IKM Tools shall support execution in logically separated environments within ON, PBN and Mission domains, so that two installations don't interfere with each other. Specifically for Training purposes, Mission Execution, Exercises and Experimentation.

Verification Method: Analysis

3.1.2 Migration

The ITM has performed a migration of the IAS Step 1 to the ITM centralized datacentres. This migration is in reality a “move” of the current IAS preserving the Share Point 2013 platform and topology configurations.

The IKM Tools is to finish the migration by updating the IAS Step 1 applications and dependant services to ITM Share Point along with the content migration (See section ref IAS Step 1 Upgrade).

Requirement ID: IKM-SRS-16

The IKM Tools shall upgrade the NIP application, content and dependant services into the provided ITM Share Point infrastructure in ON, PBN and Standalone networks (complete list needed)

Verification Method: Test

Requirement ID: IKM-SRS-17

The IKM Tools shall upgrade the TT+ application, content and dependant services into the provided ITM Share Point infrastructure in ON, PBN and Standalone networks (complete list needed)

Verification Method: Test

Requirement ID: IKM-SRS-18

The IKM Tools shall upgrade the EDMS application, content and dependant services into the provided ITM Share Point infrastructure in ON, PBN and Standalone networks (complete list needed)

Verification Method: Test

Requirement ID: IKM-SRS-19

The IKM Tools shall upgrade the current NIP Core in Share Point 2013 to the ITM provided Share Point platform in ON, PBN and Standalone networks

Verification Method: Test

3.1.3 Presentation Services

These services provide the IKM Tools visual interaction with the user by depicting the IKM Tools system operations and results on web pages.

As described in the C3 Taxonomy:

"The Web Presentation Services allow combining rich content from different data sources into a single client web page or desktop, using a combination of Web 2.0 technologies such as HTML snippets, scripting code (JavaScript), on demand code (AJAX, JSON), web service calls and proprietary code."

Therefore enables the user to customize the web pages by combining widgets (portlets, web parts and alike web components). The purpose is to achieve a better User eXperience (UX) via a flexible and standard User Interface. For example adhering to the UX aspects for a proper UX design:

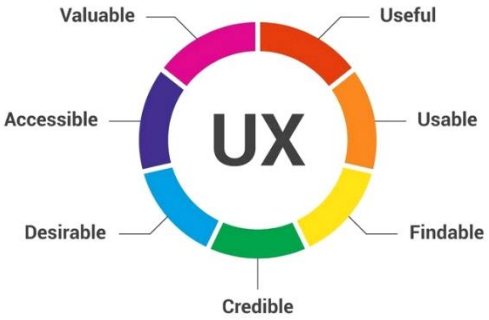


Figure 7 The 7 Factors that Influence User Experience (ref: Interaction Design Foundation)

The combination of widgets that Share Point offers facilitates this purpose. For the IKM Tools to deliver the required UX some more features need to be implemented following the industry standards.

Requirement ID: IKM-SRS-20

The IKM Tools shall provide the functionality to support the addition, update, deletion and visualisation of web widgets.

Verification Method: Test

Requirement ID: IKM-SRS-21

The IKM Tools shall provide the ability to customize the web widgets for the look and feel, security and similar functional aspects.

Verification Method: Test

Requirement ID: IKM-SRS-22

The IKM Tools shall be able to discover web widgets and make them available for use.

Verification Method: Inspection

Requirement ID: IKM-SRS-23

The IKM Tools shall provide functionality to manage the complete web widgets lifecycle. This includes deployment, modification, maintenance, un-deployment and archiving.

Verification Method: Analysis

Requirement ID: IKM-SRS-24

The IKM Tools shall provide inter-web widgets communication and remote web widgets communication.

Verification Method: Analysis

Requirement ID: IKM-SRS-25

The IKM Tools shall provide the functionality to render the visualization of the web widget according to user customization and user input.

Verification Method: Test

Requirement ID: IKM-SRS-26

The IKM Tools shall follow the Web 2.0 standards when applicable:

Verification Method: Analysis

- NISP Standard - HTML version 5.0 - "HyperText Markup Language (HTML), Version 5.0, Reference Specification"
- NISP Standard - RFC 2616: HTTP v1.1 - "HyperText Transfer Protocol (HTTP), version 1.1"
- NISP Standard - RSS 2.0 - "RSS 2.0 Specification"
- RFC 2318 - "The text/css Media Type"
- RFC 4287 - "The Atom Syndication Format"
- RFC 4329 - "Scripting Media Types"
- RFC 4627 - "The application/json Media Type for JavaScript Object Notation (JSON)"
- RFC 5023 - "The Atom Publishing Protocol"
- RFC 3986, 2005 - URI to identify resources
- CSS3 : CSS2.1 http://www.w3.org/TR/CSS2/ , CSS Style Attributes, Media Queries Level 3, CSS Namespaces, Selectors Level 3, CSS Color Level 3

Requirement ID: IKM-SRS-27

The IKM Tools shall comply with the NATO Visual Identity Guidelines [ref. NATO Visual Identity Guidelines v. 3 (on-line)]

Verification Method: Inspection

Requirement ID: IKM-SRS-28

While it is advisable that the IKM Tools GUI should be not so far from the current IKM Tools so that the user sees a similar and familiar look and feel (e.g.: colors, layout and navigational behaviour), the Contractor shall propose a new modern UX and UI design that will be approved by the Purchaser during the Design Phase.

Verification Method: Inspection

Requirement ID: IKM-SRS-29

The IKM Tools shall provide all capability responsive for desktop devices and also for mobile devices (aka tablet and mobile phone).

Verification Method: Inspection

Requirement ID: IKM-SRS-30

The IKM Tools shall provide all capability optimized for the device is operating on.

Verification Method: Inspection

3.1.4 Web Services

In order to enhance the flexibility of the IKM applications a modular design needs to be considered to logically separate the applications from the underlying services they use.

This separation will guarantee the replacement of the applications or update of services without much impact between each other.

Requirement ID: IKM-SRS-31

For that whenever possible the contractor shall use web services instead of native calls which are tightly bounded to the particular version of the API. Specifically the following interfaces depicted in the Figure 8 Internal Interfaces:

Verification Method: Analysis

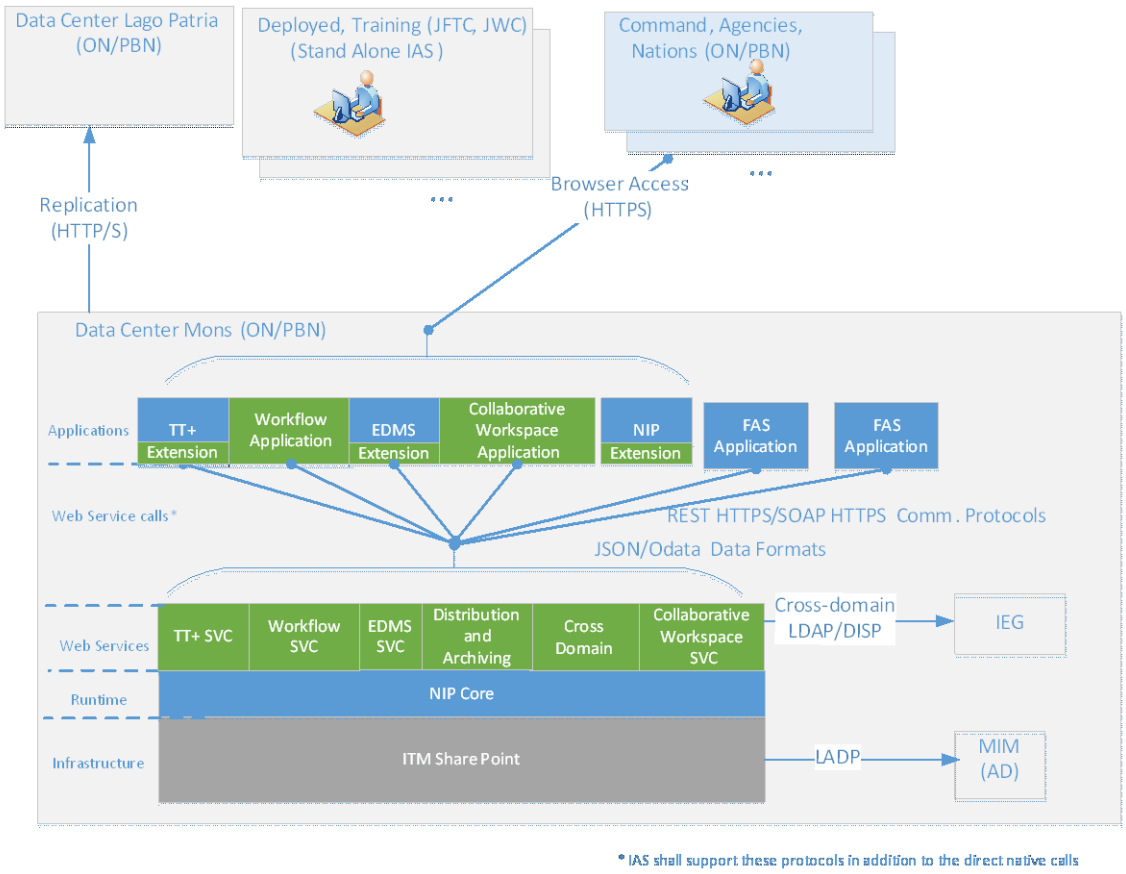


Figure 8 Internal Interfaces

Requirement ID: IKM-SRS-32

The IKM Tools Web Services using the REST Style when invoking other Web Services shall conform to the SIP for REST Messaging (see [NCIA AI 06.02.07, 2015]).

Verification Method: Analysis

Requirement ID: IKM-SRS-33

The IKM Tools Web Services using the SOAP Style when invoking other Web Services shall comply with the SIP for SOAP Messaging (see [NCIA AI 06.02.06, 2015]).

Verification Method: Analysis

Requirement ID: IKM-SRS-34

For each API Component the Contractor shall provide a “Web Service Reference Guide” to fully document the interface, including:

Verification Method: Analysis

- Server Application (the application that is exposing the service)
- Client Application (the application that is invoking the service)
- Authentication method
- Logging mechanism
- Mechanisms for securely invoking the API
- Available methods and functionality
    - Name
    - Description
    - Inputs
        - Name
        - Description
        - Type
        - Mandatory/Optional
        - Eventual enumeration values
        - Example of correct/incorrect values
    - Error messages
    - Elaboration rules

Requirement ID: IKM-SRS-35

The IKM Tools shall expose the APIs using open standards or widely accepted industry standards.

Verification Method: Analysis

Requirement ID: IKM-SRS-36

The IKM Tools API mechanism shall limit access of authenticated and Authorised Users/Systems to information products and workspaces required to perform the authorised function.

Verification Method: Analysis

Requirement ID: IKM-SRS-37

The IKM Tools GUI's shall make use of web services API to interact with a service whenever possible and when performance is not degraded below the accepted KPIs (see performance section).

Verification Method: Analysis

Requirement ID: IKM-SRS-38

The IKM Tools GUI's using web services API shall be designed to allow for future GUI enhancement or replacement with a modular and component design

Verification Method: Analysis

3.1.5 Search

The IKM Tools will leverage existing search services provided by ITM when present, improve the current IKM Tools Search service and also integrate with the SOA and IdM Platform Information Search Services.

On one hand the ITM Search Service uses the underpinning Share Point Search Engine as well as the IKM Tools (aka NIP Search) Service. As the ITM Search focuses on Share Point sources in ON and PBN domains, the NIP Search focuses on the Bi-SC domain in ON.

The intent is to consolidate all these search services into the IKM Tools Search service and managed independently.

Requirement ID: IKM-SRS-39

The IKM Tools Search service shall be upgraded as separated component (aka. dedicated Share Point Farm) so that the failure of other IKM Tools services doesn't impact this service and reverse..

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-39b

The IKM Tools Search service shall be capable to, at least, search 18 million items in ON and 7 million items in PBN, adhering to the prescribed KPIs (see NFR section Performance).

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-40

The IKM Tools shall provide search functionality leveraging the ITM Search capability if present by reusing existing services and extending them as needed, including the needed additional underlying Hardware and Software.

Verification Method: Analysis

Requirement ID: IKM-SRS-41

The IKM Tools shall be able to easily (with less than a working day) incorporate new information sources and extend the search index with the new data sources. The system uses OOTB SharePoint functions to ensure that the tool uses industry and FMN standards.

Verification Method: Analysis

Requirement ID: IKM-SRS-42

The IKM Tools shall provide the means for a user to discover the existence of information products to which they have access rights within the Enterprise.

Verification Method: Test

Requirement ID: IKM-SRS-43

The IKM Tools shall provide the means for a user to retrieve discovered information products to which they have access rights.

Verification Method: Test

Requirement ID: IKM-SRS-44

The IKM Tools shall provide the means to aggregate discovered information products by normalizing, de-duplicating and scoring them by relevance

Verification Method: Test

Requirement ID: IKM-SRS-45

The IKM Tools shall provide the means to sort discovered information products

Verification Method: Test

Requirement ID: IKM-SRS-46

The IKM Tools shall allow the user with means to specify search options, parameters and filters. At least filtered by: Command, Office, Keywords, Topics, Author, Date, Information Type and number of likes.

Verification Method: Test

Requirement ID: IKM-SRS-47

The IKM Tools shall allow the user to refine the search (faceted) based on one or more metadata fields (at least: topic, originating office, author, TT number, the command, the date, key word, area, sub area)

Verification Method: Test

On the other hand the SOA and IdM Information Search Services provides semantic search (i.e SPARQL queries search). The IKM Tools search will incorporate this service as another data source and integrate it as a federated service. For that it will pass on the search query in Natural Language to that service and collect back the results and present them to the user in the IKM Tools GUI:

Requirement ID: IKM-SRS-48

The IKM Tools shall provide the means to utilize natural language for the purposes of aggregating the discovered information products from the SOA and IdM Information Search Service

Verification Method: Test

Requirement ID: IKM-SRS-49

The IKM Tools shall provide with a Enterprise Directory workspace where to lists, search and browse users in the Enterprise with configurable views.

Verification Method: Test

Requirement ID: IKM-SRS-50

The IKM Tools Enterprise Search Directory shall enable to search users by: name, nationality, location, organizational unit and keywords

Verification Method: Test

Requirement ID: IKM-SRS-51

The IKM Tools Enterprise Search Directory shall list users depicting picture, name, organizational unit, location, nationality and job title, and allow filtering and ordering by name, nationality, organizational unit and location

Verification Method: Test

Requirement ID: IKM-SRS-52

The IKM Tools shall support searching physical media based on its metadata.

Verification Method: Test

3.1.6 Analytics

The Analytics functionality will provide a more precise, extended and relevant insights based on the information products (documents, articles...etc) reachable by the IKM Tools. It is targeted to the IKM arena and does not foresee the analysis of other data like FASes internal data types.

This functionality will provide analytical services to support the decision-making needs of the enterprise, using information produced by gathering, consolidating, cross-referencing and enhancing information from various sources. Different types of analytics can be applied: Descriptive analytics looks at past performance and understands that performance by mining historical data to look for the reasons behind past success or failure. Predictive analytics is an area of data mining that deals with extracting information from data and using it to predict trends and behaviour patterns. It is trying to answer the question what will happen. The Analytics Services enable the development, management, generation and dissemination of reports from identified information sources in a format most readily understood by the target reader and possibly based on specified templates.

The IKM Tools will support processing of structured and unstructured information, as streams or stored in persistent stores like databases.

There is a dedicated data scientist team that produces the analytic reports based on the Microsoft Power BI and KNIME tools. The Contractor will make available their output into the IKM Tools.

The IKM Tools will allow to publish analysed results in a variety of forms, as web content, MS Office documents and database stores

Requirement ID: IKM-SRS-53

The IKM Tools shall leverage the current Microsoft Power BI and KNIME technologies to provide the analytic services

Verification Method: Analysis

Requirement ID: IKM-SRS-54

The IKM Tools shall seamlessly integrate the analytic reports produced by the Power BI and KNIME tools, using the same look and feel, similar navigation and SSO without the need to re-authenticate.

Verification Method: Test

Requirement ID: IKM-SRS-55

The IKM Tools shall allow to interact with the published analytic reports as per user permissions, so that users can drill down, see hints and extended graphical information, export reports to Excel, Power Point and CSV, and collaborate with other users (sharing or analyse them in Excel).

Verification Method: Test

Requirement ID: IKM-SRS-56

The IKM Tools shall allow users to analyse the use of individual information items and documents

Verification Method: Test

Requirement ID: IKM-SRS-57

The IKM Tools shall allow users to analyse and display large amounts of data. The IKM User creates reports using the analysis tools. Then the IKM user reads the reports and identifies trends and patterns to improve NATO business processes

Verification Method: Test

Requirement ID: IKM-SRS-58

The IKM Tools shall allow users to present the results of the analysis in a wide variety of ways so that the reports are presented in the most suitable format for user interaction: as tables, charts, diagrams, video and images.

Verification Method: Test

Requirement ID: IKM-SRS-59

The IKM Tools shall allow users to analyse both structured and unstructured data.

Verification Method: Test

Requirement ID: IKM-SRS-60

The IKM Tools shall allow users to store and reuse analysis that they have already been performed, either on the same or different sources of data.

Verification Method: Test

3.2 IAS Step 1 Upgrade

The IAS Step 1 is responsible for moving the original IAS into the ITM infrastructure by transferring the IAS underlying virtual machines (including the IAS EDMS/TT+ applications) into the ITM HW infrastructure.

This means to continue using the SPS 2013 platform and current NS AIS domain.

Requirement ID: IKM-SRS-61

In this IAS Step 2 the Contractor shall upgrade to the ITM Share Point platform. By upgrading the IKM Tools, deploying them into the ITM Share Point platform along with the respective data, services and into the target security domains: NS for the IKM Tools in ON, NR for the IKM Tools in PBN.

Verification Method: Test

See the following diagram:

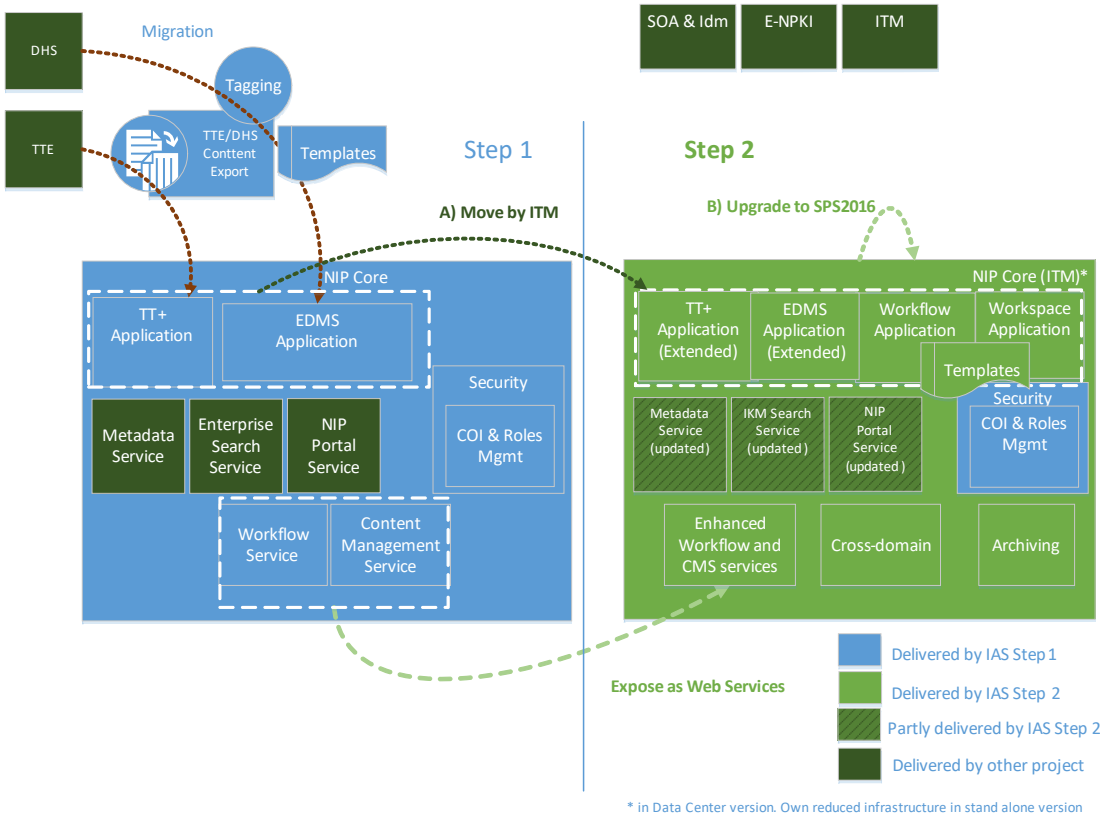


Figure 9 IAS Step 1 Upgrade and Migration

3.3 Workflow

3.3.1 Definition

The workflow definition it's configuration in the design phase to set up the workflow steps, activities, roles and actors. This phase occurs when new workflow needs to be set up but also for existing workflows that need modifications. Normally happens in a development environment and is executed by the Workflow Designer role. This role has access to all the needed tools to graphically set tasks and decision points, identify actors and flows between activities. The result is a workflow definition or a template that can be used for subsequent workflows definitions.

Once the definition is set, it is tested thoroughly and eventually published into the IKM Tools after Service Owner approval.

Modelling

3.3.1.1 As the first action for a designer Modelling involves the retrieving of the workflow requirements generally obtained by analysing existing business flows and interviewing participants.

These requirements are then "translated" into the workflow model by the Workflow Designer by using the IKM Tools workflow design tool. In here the Workflow Designer can select actions, flows, activities and other related components and configure them, by connecting them with arrows as representing the process flow.

Requirement ID: IKM-SRS-62

The IKM Tools Workflow shall be available on the Operation Network (ON) and on the Protected Business Network (PBN). See SOW ANNEX D for the complete list of sites.

Verification Method: Inspection

Requirement ID: IKM-SRS-63

The IKM Tools shall allow the user to manage the workflow definition with operations: create, delete, modify, export, import, backup and restore.

Verification Method: Test

Requirement ID: IKM-SRS-64

The IKM Tools shall enable the creation of workflows, its steps, including Exception, Alternative and Extension paths; defining the activities/tasks, their inputs, actors (users/services), conditions (triggers, rules, etc.) under which they will operate and their expected outputs.

Verification Method: Test

Requirement ID: IKM-SRS-65

The IKM Tools shall allow to model workflows routing: parallel, sequential, forks and reverse flows.

Verification Method: Test

Requirement ID: IKM-SRS-66

The IKM Tools shall provide a workflow functionality for document processes, including but not limited to: item approval, collect signatures, get feedback, document tracking, content publishing and document archiving and distribution

Verification Method: Analysis

Requirement ID: IKM-SRS-67

The IKM Tools shall provide a workflow functionality applicable at least to the following information products: documents, events, images, video, audio, list items and related IKM information products

Verification Method: Analysis

Requirement ID: IKM-SRS-68

The IKM Tools shall allow to scope workflows to Sites, Lists and Items

Verification Method: Analysis

Requirement ID: IKM-SRS-69

The IKM Tools shall provide the means to manage and reuse data structures (workflow templates, forms templates, documents, etc.) for use within another workflow

Verification Method: Test

Requirement ID: IKM-SRS-70

The IKM Tools shall include a graphical workflow modelling, composition and orchestration with other integrated and federated services or workflows; and conforming with the standard Business Process Model and Notation (BPMN).

Verification Method: Inspection

Requirement ID: IKM-SRS-71

The IKM Tools shall support the following workflows patterns:

Verification Method: Analysis

- Digitalized Process: workflows defined from external analogue source (i.e.: paper)
- Intelligence Business Operations (IBO): workflows based on analytics (part of IKM Tools) and decision management technologies integration that produces a transaction

Requirement ID: IKM-SRS-72

The IKM Tools shall allow the user to create, delete and modify an activity.

Verification Method: Test

Requirement ID: IKM-SRS-73

The IKM Tools shall allow the user to create, delete and modify a decision point.

Verification Method: Test

Requirement ID: IKM-SRS-74

The IKM Tools shall allow the user to order and connect activities and decision points as linear, parallel or nested flows.

Verification Method: Test

Requirement ID: IKM-SRS-75

The IKM Tools shall allow the user to drill down into the details about a workflow and activity.

Verification Method: Test

Requirement ID: IKM-SRS-76

The IKM Tools shall allow the user to select events that trigger the workflow, including but not limited to:

Verification Method: Analysis

- Item added, modified or deleted
- Item status changed

Requirement ID: IKM-SRS-77

The IKM Tools shall allow the user to select the triggered action mode: manual by the user or automatic by an event

Verification Method: Analysis

Requirement ID: IKM-SRS-78

The IKM Tools shall allow the user to select the means by which an event is notified:

Verification Method: Test

- by the GUI as a user interaction
- by a web service call
- by receiving an E-mail
- by adding an item in a folder in the File System
- by the IKM Tools workspace or workflow

Requirement ID: IKM-SRS-79

The IKM Tools shall provide the means to define and manage roles of users and services, their access and participation within a workflow and the secure execution of that workflow. The secured execution includes the access to the workflow activities as well as its related information products (i.e.: workflow documents)

Verification Method: Analysis

Requirement ID: IKM-SRS-80

The IKM Tools shall allow workflows and dependant information (information products, participants and alike) to be indexed by the NATO Search so that they are visible to the Enterprise Users following the concept of "Responsibility to Share", and allow to restrict sensitive workflows to a close hold group of participants (i.e.: public workflow attribute to allow all users read access and close-hold attribute to define a limited group of users).

Verification Method: Test

Requirement ID: IKM-SRS-81

The IKM Wools shall allow users to model business processes for repeated execution as "workflows" so that after workflow completion users can re-start the workflow again from the beginning in an iterative manner.

Verification Method: Test

3.3.1.2

Simulation

After the workflow definition is completed the simulation phase helps to verify the workflow adequacy and trustfulness against the business requirements and validate that is fit for purpose and error free.

The simulation needs to represent a realistic workload in terms of user interactions and information products outputs. The purpose is to reassure it will operate as expected in the real scenario when is published to production environment.

Requirement ID: IKM-SRS-82

The IKM Tools shall support the simulation and validation of workflows

Verification Method: Test

Requirement ID: IKM-SRS-83

The IKM Tools shall allow workflow processes improvement by iteratively Design, Model, Execute, Monitor and Optimize workflows

Verification Method: Analysis

Requirement ID: IKM-SRS-84

The IKM Tools shall support the profiling of workflows and consumers (users and services) using metadata and supporting taxonomies (controlled, semi-controlled and open), ontologies, data models and standard data elements. This includes versioning of workflows and the state of execution of specific instances.

Verification Method: Analysis

Requirement ID: IKM-SRS-85

The IKM Tools shall provide a GUI for debugging workflows. The GUI shall allow the user to set breaks points in activities and related script code, view workflow status variables, continue the process, step-in inside a sub-activity and stop the workflow.

Verification Method: Test

3.3.2 Processing

This phase occurs during the workflow operational usage. The day to day workflow execution is described here.

3.3.2.1 Rules

The rules are the governing actions a workflow is adhered to such ways of executing steps, scripts and activities. The rules specify what a workflow can do on a given time and restraint to the workflow definition. The Workflow Engine applies the rules by reading an input, processing it and producing an output.

Requirement ID: IKM-SRS-86

The IKM Tools shall allow the user to initiate a workflow according to the workflow definition.

Verification Method: Test

Requirement ID: IKM-SRS-87

The IKM Tools shall automatically update the new workflow instances upon a workflow definition change, without affecting the current workflow instances being executed.

Verification Method: Test

Requirement ID: IKM-SRS-88

The IKM Tools shall enable activities to run code (Scripts) to extend the activity's actions

Verification Method: Test

Requirement ID: IKM-SRS-89

The IKM Tools shall allow the user to record activity status.

Verification Method: Test

Requirement ID: IKM-SRS-90

The IKM Tools shall allow the user to delegate an activity to another user according to the workflow definition and set permissions

Verification Method: Test

Requirement ID: IKM-SRS-91

The IKM Tools shall allow the user to take over an activity from another user according to the workflow definition and set permissions

Verification Method: Test

Requirement ID: IKM-SRS-92

The IKM Tools shall allow to preserve the current execution state in case of an interruption and resume the execution without any data and status loss.

Verification Method: Test

3.3.2.2

Notifications

An important aspect of a workflow is the user's participation and awareness of the workflow status. The notifications mechanisms allows to send information to the participants of a workflow for status updates, to remind user's needed action or simply for information.

Although normally email is a standard method other methods can also be provided such as SMS or web announcements in a portal.

Requirement ID: IKM-SRS-93

The IKM Tools shall allow the user to manage personal workflow notifications (read, register and un-register) with parameters such as frequency, delivery methods, and triggered events

Verification Method: Test

Requirement ID: IKM-SRS-94

The IKM Tools shall allow an authorized user to manage (assign, remove) user's workflow notifications (read, register and un-register) with parameters such as frequency, delivery methods, and triggered events

Verification Method: Test

Requirement ID: IKM-SRS-95

The IKM Tools shall allow the user to filter notifications he receives by at least the following parameters:

Verification Method: Test

- Workflow actions
- Frequency
- Notification format: email, SMS or alert message in IKM Tools GUI.

Requirement ID: IKM-SRS-96

The IKM Tools shall be able to generate notifications of workflow status changes to the participants.

Verification Method: Test

Requirement ID: IKM-SRS-97

The IKM Tools shall automatically inform the participant for an activity that a new activity has to be performed.

Verification Method: Test

Requirement ID: IKM-SRS-98

The IKM Tools shall automatically inform the participant for an activity that a related decision has been made.

Verification Method: Test

Requirement ID: IKM-SRS-99

The IKM Tools shall automatically inform the user for any definition changes in the workflow he designed

Verification Method: Test

Requirement ID: IKM-SRS-100

The IKM Tools shall provide the mechanism to send SMS notifications to participants being individual users or a group of users and according to the defined notification parameters

Verification Method: Test

Requirement ID: IKM-SRS-101

The IKM Tools shall provide the mechanism to send e-mail notifications to participants being individual users or a group of users and according to the defined notification parameters

Verification Method: Test

Requirement ID: IKM-SRS-102

The IKM Tools shall allow to customize the notification message content: text, links and other information so that it is personalized and adapted to the workflow and the target receiver audience. This customization can be easily changed by the authorized user and associated to different workflow steps (i.e.: upon approval the participant receives "activity 2 approved by the manager with message: all good").

Verification Method: Test

Requirement ID: IKM-SRS-103

The IKM Tools shall be able to send email notifications with the participant's required responses enabled in the message so that participant can directly select his response without the need to go to the IKM Tools Workflow GUI, that is, within the user's email client (i.e.: participant receives an email message in Outlook containing action buttons for Approve, Reject responses).

Verification Method: Test

Annotations

3.3.2.3 Annotations is a feature to enrich the workflow understanding by adding side notes into its description, that is, during the definition phase.

But also serves to complement the workflow decisions by explaining why a decision was taken in a decision point and therefore acknowledge the rationale.

Requirement ID: IKM-SRS-104

The IKM Tools shall allow the user to annotate a workflow or an activity with notes, links, files, images and/or keywords, at workflow definition and during workflow execution (instance)

Verification Method: Test

Requirement ID: IKM-SRS-105

The IKM Tools shall allow the user to manage annotations: add, delete, modify and define keywords.

Verification Method: Test

Requirement ID: IKM-SRS-106

The IKM Tools shall allow the user to preserve (record and store) annotations about a decision (i.e.: keep history)

Verification Method: Test

3.3.3 Services

Web Services

Requirement ID: IKM-SRS-107

3.3.3.1 The IKM Tools shall provide an open standard workflow functionality API for FASes integration (i.e. based on OData), so that FASes can perform web service calls to the IKM Workflow. Specifically integration with Microsoft Power BI for workflow analytics reporting.

Verification Method: Analysis

Collaboration Services

Requirement ID: IKM-SRS-108

3.3.3.2 The IKM Tools shall provide the means to distribute information products to and from participants within a workflow instance and their collaborative environment over integrated and federated services.

Verification Method: Test

Requirement ID: IKM-SRS-109

The IKM Tools shall support a collaborative environment and means for check-in/out, versioning, storage and share in Collaborative Workspace and import and export of workflows information.

Verification Method: Test

Requirement ID: IKM-SRS-110

The IKM Tools shall integrate with Share Point framework for creating, moving, modifying and deleting items in a list, for sending and receiving items to workspaces and for adding and changing items metadata. The integration consists of user interaction (access, operations) via the Share Point GUI and it's underlying services.

Verification Method: Test

Requirement ID: IKM-SRS-111

The IKM Tools shall provide a workflow functionality integrated with the following IKM portals if available: NIP Portal, EDMS and TT+. So that the activities and information products of the workflows can seamlessly interact with these applications as they do with the Share Point framework

Verification Method: Test

Requirement ID: IKM-SRS-112

The IKM Tools shall provide a search capability on the workflows' content: descriptions, status, activities, participants, keywords and other metadata, leveraging the IKM Search if available.

Verification Method: Test

Presentation Services

Requirement ID: IKM-SRS-113

3.3.3.3 The IKM Tools shall offer a web Graphical User Interface (GUI) for all the user interactions: participate, design, monitor, debug and report workflows and activities.

Verification Method: Inspection

Requirement ID: IKM-SRS-114

The IKM Tools workflow web GUI shall allow the user to graphically design workflows with elements representing activities, flows, decision points, actions and all necessary components of the workflow.

Verification Method: Inspection

Requirement ID: IKM-SRS-115

The IKM Tools workflow web GUI shall allow the user to interact with the workflow via rich data collection forms, that is with form controls supporting photos, geo location, maps, barcodes, signatures and alike, adhering to the standards [ref HTML5, CSS...]

Verification Method: Analysis

Requirement ID: IKM-SRS-116

The IKM Tools workflow forms shall allow validation of user's inputs by field types: phone, email, country and other defined Managed Metadata in IKM Tools whenever possible. And to set mandatory filled fields visually displayed (i.e.: with a red star).

Verification Method: Test

Requirement ID: IKM-SRS-117

The IKM Tools shall allow configurable dashboards to present the information product's metadata and workflow's information in multiple views

Verification Method: Analysis

Requirement ID: IKM-SRS-118

The IKM Tools workflow forms shall allow to connect to back end Authoritative Datasources (AD), being in the Enterprise, in the Federation or in the Cloud, to

retrieve information and seamlessly using the needed authorization method (i.e.: use ADFS if user information is in a federation).

Verification Method: Test

Requirement ID: IKM-SRS-119

The IKM Tools shall visualize the workflow status with indicators to at least depict: workflow Defined (not initiated), workflow In-progress, workflow Paused, workflow Error and workflow Completed

Verification Method: Inspection

Requirement ID: IKM-SRS-120

The IKM Tools shall allow to show the workflow status and dashboard reports as separate components views so that can be re-used and placed in different IKM Tools sites and web pages (i.e.: a report web part connected to a workflow that can be placed in multiple pages).

Verification Method: Test

Requirement ID: IKM-SRS-121

The IKM Tools shall allow to create views based on IP metadata (i.e. documents with tagged 'likes').

Verification Method: Test

3.3.4 Management

In order to be able to properly control a workflow lifecycle a management feature is needed. It consists of all activities surrounding a workflow administration. The purpose is to ensure the proper execution, with the correct participants and information inputs, the Workflow Manager role overviews this with the help of the IKM Tools monitoring and alerting features. Takes proper actions and also informs the Service Owner for service deviations in terms of performance, availability and other parameters set in the SLA.

Requirement ID: IKM-SRS-122

3.3.4.1 The IKM Tools shall offer a comprehensive framework to graphically identify issues and actions and also to take O&M actions such as exporting or importing workflow definitions, maintaining the workflow versions and auditability.

Verification Method: Inspection

Logging

As part of the Management feature the logging helps to audit the workflow steps for future forensic analysis but also for Continuous Service Improvement (CSI) of the workflow itself.

Logging has to be detailed enough to capture all the execution steps, with special emphasis in error detecting and prevention via warning logs.

Requirement ID: IKM-SRS-123

The IKM Tools shall log workflow execution events categorized so that they can be easily identified at least as information, warning and error.

Verification Method: Analysis

Requirement ID: IKM-SRS-124

The IKM Tools logs shall be accessible by an authorized user via the IKM Tools GUI (i.e.: logs as a list with restricted access)

Verification Method: Inspection

Requirement ID: IKM-SRS-125

The IKM Tools shall keep the history of the changes that occurred to the workflow with metadata, including but not limited to: name of the user, date, time, nature of the change.

Verification Method: Inspection

Requirement ID: IKM-SRS-126

The IKM Tools shall provide statistical and auditing capabilities over the workflow instances for NATO governance and policy compliance purposes [ref NIMP AC/322-D(2017)0027] and for workflow optimization.

Verification Method: Test

3.3.4.2 Reporting

This feature permits to gather workflow information in formats that are suitable for the users to depict multiple aspects of the workflow instances. Reports are normally established by the Workflow Designer and triggered automatically by the IKM Tools or manually by the user via the IKM Tools GUI.

Reporting allows to collate workflow information on various form and perform some basic analysis to produce a visual output being a list of events, some performance diagrams or a summary of the entire workflow status.

Requirement ID: IKM-SRS-127

The IKM Tools shall report information associated with workflows, their instances and related collaborative environments.

Verification Method: Inspection

Requirement ID: IKM-SRS-128

The IKM Tools workflows reports shall at least include the following report types: Workflow Overview listing all instances, Activity Status showing activities execution times, and User usage reports depicting user's contributions in workflows.

Verification Method: Inspection

Requirement ID: IKM-SRS-129

The IKM Tools workflows reports shall be in the most suitable form for each report type: graphs, diagrams, tables or texts

Verification Method: Analysis

Requirement ID: IKM-SRS-130

The IKM Tools workflow reports shall be produced in widely accepted data formats, specifically compatible with Microsoft Excel for tables and Microsoft Word for texts or diagrams.

Verification Method: Test

Administration

3.3.4.3 Administering a workflow means to warrant it works appropriately and that any modifications asked by a user or taken by the Workflow Administrator is applied and executed.

Exporting, importing and backing up workflows are necessary actions to preserve proper O&M and execution control. Also to allow to expand the IKM Tools functionality by incorporating new workflows defined elsewhere that can be applied to the IKM Tools.

Also entails to un-lock workflows in the event of errors, misuse or just speed it up by enforcing steps completion or by delegating activities to other users.

Requirement ID: IKM-SRS-131

The IKM Tools shall support the maintenance and sustainment of workflows, their execution (instances) and exception handling in accordance with a defined Model, their content and of their supporting systems (in conjunction with SMC and IA services) throughout the workflow and instance lifecycles.

Verification Method: Test

Requirement ID: IKM-SRS-132

The IKM Tools shall allow the user to export workflows with a button-click and in a format that can be easily stored, interpret by a user and imported back to the IKM Tools

Verification Method: Test

Requirement ID: IKM-SRS-133

The IKM Tools shall allow workflows to be backed-up and archived recording the: workflow id, workflow name, workflow definition, workflow version, workflow status, back up date time and user who triggered the backup.

Verification Method: Test

Requirement ID: IKM-SRS-134

The IKM Tools shall allow the user to manage activities as components of the workflow definition.

Verification Method: Test

Requirement ID: IKM-SRS-135

The IKM Tools shall allow authorized users to un-lock activities by superseding the participant, discarding participants' action or by delegating the action to another participant.

Verification Method: Test

Requirement ID: IKM-SRS-136

The IKM Tools shall allow the user to manage the workflow execution by start, pause, cancel and monitor the workflow instances

Verification Method: Analysis

Requirement ID: IKM-SRS-137

The IKM Tools shall allow the user to manage (Create, Read, Update and Delete) activities.

Verification Method: Test

Requirement ID: IKM-SRS-138

The IKM Tools shall allow the user to manage decision points.

Verification Method: Test

Requirement ID: IKM-SRS-139

The IKM Tools shall allow the user to change the decision status.

Verification Method: Test

3.3.5 TT+ extension

The IAS Step 1 project delivered the TT+ application with some missing features that were decided to include in the next IKM Tools project.

This features are very specific and are meant to accurately make TT+ fit for purpose for the Commands.

Therefore they are to be included in the existing TT+ application and not in the workflow application.

These are specifically captured in the following requirements:

Requirement ID: IKM-SRS-140

TT+ shall provide a configurable Prioritisation indicator on Taskers.

A Priority column shall be added to enable Tasker prioritisation.

Default values: Urgent, High, Routine

This priority column shall be configurable (i.e. possibility to amend default values or add new ones) and associated to a configurable Traffic Light indicator (an icon), i.e. it shall be possible to change that icon. The TTE icons shall be reused as default icons.

In addition, this Traffic Light indicator shall have a tooltip that provides the actual Priority in full text (i.e. "Urgent", "High", "Routine"). When the user selects the prioritisation of the tasker the Tasker priority is displayed with arrows (up for high, down for low and no arrow for normal). Tasker times (new for issued in the last 24 hours, green until 80% of the suspense date had expired, yellow for when 20% remains, red for tasker overdue).

Verification Method: Test

Requirement ID: IKM-SRS-141

TT+ shall provide a configurable "Office inbox".

The Office inbox provides a list of all actions assigned to a specific office.

The user shall be able to flip through offices inboxes seamlessly.

The TTE Office inbox shall be used as the reference.

Verification Method: Test

Requirement ID: IKM-SRS-142

TT+ shall allow a user with appropriate permissions to create a Tasker from a document stored in EDMS:

When the EDMS is configured to be connected to TT+, the user shall be able to create a Tasker from any document. The User must be have the appropriate permissions to create a tasker.

The tasker hence created shall retain the reference to the "initiating document" by simply storing a link under the 'Documents' tab.

Verification Method: Test

Requirement ID: IKM-SRS-143

TT+ shall allow external services to store the URL of the tasker initiation document. (Create field to foresee previous requirement):

When workspace is created (or links list in workspace), add link to the initiating document

Verification Method: Test

Requirement ID: IKM-SRS-144

TT+ shall provide a configurable "My tasks page":

The 'My tasks' page shall provide a list of all the incomplete tasks that are assigned to a user (same approach as office inbox, with configuration list).

The user has a my tasks view which is solely for the logged on user. The user can sort the view of taskers by priority & date. (aka: TT+ 'My Tasks' shall be used as the reference).

Verification Method: Test

Requirement ID: IKM-SRS-145

The user shall be able to create and manage a favourite users list as follows:

Verification Method: Test

- to add users from Active Directory user account or groups, and SharePoint groups.
- to remove users
- to sync users with AD and grey out the deleted users accounts (non present users).

Requirement ID: IKM-SRS-146

TT+ sub tasking shall leverage Favourite users:

When creating / editing a task (sub task) the user shall be able to cherry-pick amongst her favourite users as well as enter new AD user accounts manually. TT+ shall rely on an extended people picker control.

See the TTE 'sub tasking' functionality as the reference.

In addition, deleted ad account accounts should be greyed out (+ no checkbox)

Verification Method: Test

Requirement ID: IKM-SRS-147

TT+ 'adding a member' shall leverage Favourite users:

When adding a new member to a Tasker, the user shall be able to cherry-pick amongst her favourite users as well as enter new AD user accounts manually. TT+ shall rely on an extended people picker control.

See the TTE 'add Users' functionality under 'Members' tab as the reference.

In addition, deleted ad account accounts should be greyed out (+ no checkbox)

Verification Method: Test

Requirement ID: IKM-SRS-148

TT+ shall provide a centralised Help Centre:

For the centralized implementation (NIP scenario), all commands shall share the same centralised Help Center (no need to install a separate Help Center for every command).

For standalone implementations, the TT+ instance shall have a dedicated Help Center.

The user shall be able to access a centralised help centre site regardless of the command. The user shall be able to search common help questions, find related SOPs, find the contact numbers and email addresses for the help centre. The user should also be able to speak to a real competent person.

Verification Method: Test

Requirement ID: IKM-SRS-149

TT+ shall allow a user to send a Tasker Product to the external documents landing zone:

When the system is configured to be connected to an external documents landing zone, the user shall be able to send the final Tasker product to the configured external documents landing zone.

Verification Method: Test

Requirement ID: IKM-SRS-150

The selected task shall be highlighted on the sub tasks tree:

on the sub tasks page (within a Tasker, on the 'Tasks' tab), the selected task shall be highlighted on the tasks tree on the left hand side.

Verification Method: Test

Requirement ID: IKM-SRS-151

TT+ shall provide a configurable Business Intelligence (BI) Reporting component:

For the TT+ baseline, it shall be configured to provide the default pie chart reports that came with TTE:

Verification Method: Test

- Taskers by office (exclude archived taskers)
- Taskers by status (exclude archived taskers)

Requirement ID: IKM-SRS-152

The TT+ shall allow to print the taskers list by exporting from the tasker list, the office inbox, any view, and subtasks into an Excel or PowerPoint file for printing.

Verification Method: Test

Requirement ID: IKM-SRS-153

The TT+ shall hide or disable Site Content button from users so that only functional administrator, and site collection administrators are able to see the site contents.

Verification Method: Test

Requirement ID: IKM-SRS-154

The TT+ shall Ensure all tasker details are hidden from users who are not members of close hold taskers

Verification Method: Test

Requirement ID: IKM-SRS-155

The TT+ shall prevent the user to add dates in the past in the tasker for the following date fields: suspense date, contribution date, target completion date, coordination date, date due to HQ, validation date , COS action date, Note that dates are different depending on the site.

Verification Method: Test

Requirement ID: IKM-SRS-156

The TT+ shall allow to add field description for each metadata element in the tasker so that subsequent users receive information on what should be included in each field along with an example if necessary. (i.e. greyed out text or hint)

Verification Method: Test

Requirement ID: IKM-SRS-157

The TT+ shall pre-set the Suspense Date to "today plus 3 weeks" value in the tasker as default but user can change it afterwards.

Verification Method: Test

Requirement ID: IKM-SRS-158

The TT+ shall ensure all actions and changes to tasker to be listed in History (i.e.: change in Instructions field)

Verification Method: Test

Requirement ID: IKM-SRS-159

The TT+ shall mark archived taskers in their metadata so that they can be filtered in views.

Verification Method: Test

Requirement ID: IKM-SRS-160

The TT+ shall remove the Upload Multiple documents button if feature not supported

Verification Method: Inspection

Requirement ID: IKM-SRS-161

The TT+ shall provide an option for preventing uploading documents in regular Taskers' document libraries but allowing uploading links instead. So that when the option is enabled, the tasker prevents the user from uploading the document in the tasker. Instead the user has to upload the link. (User must previously store the document in EDMS). The upload of documents must always be possible for Close Hold taskers.

Verification Method: Test

Requirement ID: IKM-SRS-162

The TT+ shall ensure no folders are created in Links Libraries

Verification Method: Test

Requirement ID: IKM-SRS-163

The TT+ shall implement the Category metadata (Product, Point Paper...etc.) in the tasker Links

Verification Method: Test

Requirement ID: IKM-SRS-164

The TT+ shall implement a traffic light feature for subtasks as they are for tasks

Verification Method: Inspection

Requirement ID: IKM-SRS-165

The TT+ shall implement an Alert Subscription system to tell any contributor when sub task status reaches subscribed status

Verification Method: Test

Requirement ID: IKM-SRS-166

The TT+ shall prevent the creation of subtasks on closed taskers

Verification Method: Test

Requirement ID: IKM-SRS-167

The TT+ shall include additional tasker metadata in the notification email when subtask is issued. A configurable email shall include it

Verification Method: Test

Requirement ID: IKM-SRS-168

The TT+ shall provide a Mail to All members function in the Tasker

Verification Method: Test

Requirement ID: IKM-SRS-169

The TT+ shall guarantee the consistency of the Taskers dates so that Creation Dates are always earlier than Suspense and End Dates

Verification Method: Test

Requirement ID: IKM-SRS-170

The TT+ shall support up to 5000 open taskers per Command

Verification Method: Test

3.4 Workspace

Requirement ID: IKM-SRS-171

The IKM Tools shall provide the means to share an information product among other selected users for simultaneous editing as in EDMS and NIP. The user can select the permissions levels to grant to other contributors based on existing security policies and permissions leveraging the IKM Tools Framework Runtime.

Verification Method: Test

Requirement ID: IKM-SRS-172

The IKM Tools shall provide the means to uniquely identify the content leveraging the IKM Tools Framework Runtime

Verification Method: Analysis

Requirement ID: IKM-SRS-173

The IKM Tools shall support the play-back of audio, and display of video and textual information. (i.e. documents, presentations, recordings, etc.) as NIP and leveraging the IKM Tools Framework Runtime.

Verification Method: Test

Requirement ID: IKM-SRS-174

The IKM Tools shall support the use of metadata (profile) information of users, processes and services to control access and permissions to workspace information leveraging the IKM Tools Framework Runtime

Verification Method: Analysis, Inspection

Requirement ID: IKM-SRS-175

The IKM Tools shall support the integrity, completeness, security, compliance and reuse of information products by supporting their review, assessment, retention and disposition. This includes support for the management of the information product's lifecycle at the "Planning" and "Disposition" phases with policies and governance processes.

Verification Method: Analysis

3.4.1 Information Product

In general IKM Tools defines a higher abstract entity called Information Product (IP) that consists of: the content (image, document…etc.), possibly a signature and core metadata as depicted in the following figure:

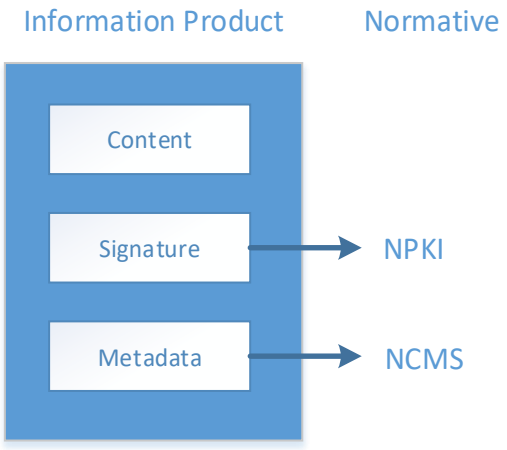


Figure 10 Information Product in IKM Tools

The information product base implementation in Share Point is the 'item' content type.

Requirement ID: IKM-SRS-176

If the Enterprise NATO Public Key Infrastructure (NPKI) is available the IKM Tools shall support the signature of the IP using the NPKI to guarantee the IP authenticity and integrity. Else using the already existing NPKI.

Verification Method: Test

Requirement ID: IKM-SRS-177

The IKM Tools shall adhere to the Confidentiality Labelling for Information Sharing [Ref. AC/322(CP/1)N(2015)0021] for IP security metadata tagging and labelling.

Verification Method: Analysis

Requirement ID: IKM-SRS-177b

The IKM Tools shall adhere to the NATO Core Metadata Specification (NCMS) [Ref. STANAG 5636] for IP metadata tagging and labelling.

Verification Method: Analysis

Requirement ID: IKM-SRS-178

The IKM Tools shall allow the user to tag the IP with metadata available in the system (Taxonomy, Managed Metadata, Keywords and Likes) and also new metadata introduced by the user.

Verification Method: Test

Requirement ID: IKM-SRS-179

The IKM Tools shall automate as much as possible the tagging of IP metadata based on the IP content. That is if the IP contains metadata fields which values are within the IP content the IKM Tools shall automatically populate them.

Verification Method: Test

Requirement ID: IKM-SRS-180

The IKM Tools shall provide a mechanism for identifying the authoritative source (i.e. author) of the IP.

Verification Method: Test

Requirement ID: IKM-SRS-181

The IKM Tools shall allow a mechanism for multiple users to be able to simultaneously update IPs so that all user's modifications are taken into account and do not interfere.

Verification Method: Test

Requirement ID: IKM-SRS-182

The IKM Tools workspace shall support the entire lifecycle of an information product: creation, capture, update, storage, transformation, distribution and disposal compatible with EDMS and NIP, and leveraging the IKM Tools Framework Runtime. Supporting the correspondent templates and metadata and applicable to, at least these information products:

Verification Method: Test

- MS Office products (documents, workbooks, presentations etc.)
- Open Standard office product
- Audio recordings
- Videos recordings
- Emails
- PDFs XML files (that may include battle-space objects and definitions)
- Instant Messages recordings.
- Tasks (activities, processes) Events
- Unformatted text (generic content types, etc.)

Requirement ID: IKM-SRS-183

The IKM Tools shall ensure a consistent metadata tagging so that the metadata within the file is the same as the metadata depicted to the user in the portal, i.e.: a file with property Classification and value NATO UNCLASSIFIED is shown in a list with the column Classification NATO UNCLASSIFIED. And this consistency is maintained throughout the entire file life-cycle (i.e.: if there is a

modification in the Share Point column list metadata, it automatically updates the correspondent metadata in the file properties, and reverse).

Verification Method: Test

Requirement ID: IKM-SRS-184

The IKM Tools workspace shall support the capture of information using information templates.

Verification Method: Analysis

Requirement ID: IKM-SRS-185

The IKM Tools workspace shall support the metadata provisioning (profiling) of workspace(s) and information.

Verification Method: Analysis, Inspection

Requirement ID: IKM-SRS-186

The IKM Tools workspace shall ensure mandatory metadata (profile) information of workspace(s) and information is provisioned and sustained throughout their respective lifecycles.

Verification Method: Analysis

Requirement ID: IKM-SRS-187

The IKM Tools workspace shall ensure mandatory metadata (profile) information of processes is provisioned and sustained throughout their respective lifecycles.

Verification Method: Analysis

Requirement ID: IKM-SRS-188

The IKM Tools workspace shall support the creation of information products. (i.e. documents, presentations, spreadsheets, other file types, etc.) through collaborative processes.

Verification Method: Test

Requirement ID: IKM-SRS-189

The IKM Tools workspace shall support processes for the capture of metadata (profile) information of workspace users, processes and information.

Verification Method: Analysis

Requirement ID: IKM-SRS-190

The IKM Tools shall support the profiling of content and consumers (users and services) using metadata and supporting taxonomies (controlled, semi-controlled and open), ontologies, data models and standard data elements. This

includes support for content versioning. This applies to the entire lifecycle for that content leveraging the IKM Tools Framework Runtime

Verification Method: Analysis

Distribution and Archiving

As the objective is that the NATO Enterprise will adhere to the “Responsibility to share” principle and as such enable IP to be easily shared among stakeholders the Distribution feature plays a fundamental role.

3.4.1.1

Requirement ID: IKM-SRS-191

The IKM Tools shall be able to share or transfer IP between different networks and domains with the same security domain (a.k.a. security classification) in a controlled manner, in line with in place security policies.

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-192

The Archiving process for long term preservation shall follow the same approach as distributing IP’s regarding signature and metadata preservation

Verification Method: Analysis

Requirement ID: IKM-SRS-193

The IKM Tools shall support the export and import of information products to and from Content Management Systems (CMS) that comply with Content Management Interoperability Services (CMIS). This also includes support for metadata and other pertinent governances related to that content to those associated systems.

Verification Method: Analysis

Requirement ID: IKM-SRS-194

The IKM Tools shall be able to export and import information products that comply with the IKM Tools archiving and exporting specifications (ref INFORMATION SHARING WITHIN NATO,AC/322-N(2010)0026-REV1) without data and metadata loss, that is, preserving original information.

Verification Method: Analysis

Requirement ID: IKM-SRS-195

The IKM Tools shall export the information products following the Industry and NATO standards maintaining the information product's format whenever possible:

Verification Method: Test

- Standards: Open Packaging Convention (OPC) ECMA-376, STANAG-4774, STANAG-4778
- TN-1491 Edition 1: "Profiles for binding metadata to a data object" , using the profile that best suits the information product format

Requirement ID: IKM-SRS-196

The IKM Tools shall be able to export and archive the non- OPC compliant information products with their metadata included in their format container's properties:

Verification Method: Test

- Images: JPEG, GIF, JPG
- Audio and video: NISP Standard - MPEG-4, MPEG-2
- ASCII text-formatted data: txt
- Portable Document Format, ISO 32000-1: PDF

For an Information Product to be distributed, or send outside the IKM Tools system while preserving it's metadata a particular process must be followed. This process involves existing services with well documented guidelines.

The intent is to guarantee the IP contains the enriched information for another system to take advantage of. Conversely a reverse process is also envisaged when importing an IP with metadata from another system.

Requirement ID: IKM-SRS-197

When an IP is to be exported it shall use a NATO Metadata Binding Service (NMBS) to attach the IP metadata to the exported file (bind). Conversely when an IKM Tools compatible IP is imported into the IKM Tools the Binding Service shall detach the IP metadata (unbind) and store it appropriately in the destination IKM Tools library. See Figure below:

Verification Method: Analysis

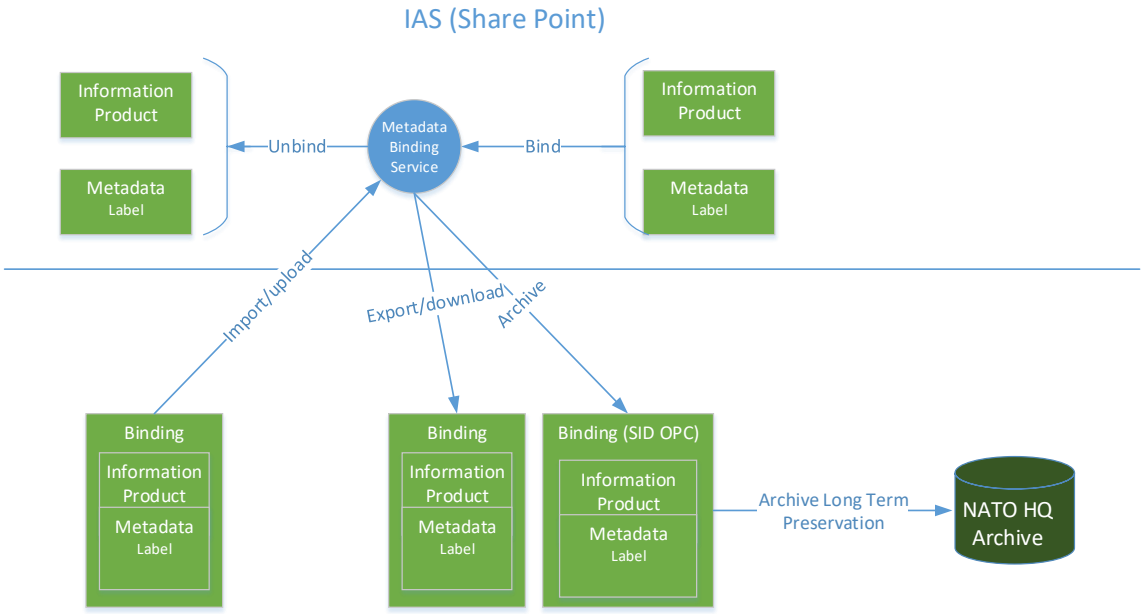


Figure 11 Distribution and Archiving

Requirement ID: IKM-SRS-198

The IKM Tools shall use the NMBS following the specification STANAG 4778 to bind the information product metadata (stored in the file properties space or Share Point space) with its content

Verification Method: Analysis

Requirement ID: IKM-SRS-199

The IKM Tools shall use the NMBS to extract the information product metadata and store it in the appropriate space in the IKM Tools (Share Point or other) so that the information product metadata is recognized in the IKM Tools as any other information product

Verification Method: Test

Requirement ID: IKM-SRS-200

The IKM Tools shall use the NMBS to atomic sign the metadata with its related content (applicable to all IP) to guarantee the integrity of the tagged content (not to confuse with the integrity of the entire IP).

Verification Method: Test

Requirement ID: IKM-SRS-201

The IKM Tools shall use the NBMS to incorporate the signature in the proper location within the IP following the specification TN-1491.

Verification Method: Analysis

Requirement ID: IKM-SRS-202

An imported IPs may be subjected to metadata change, when mapping original metadata with the available agreed metadata in the IKM Tools. Or it may change its original metadata along the IPs lifetime. In such cases the original author signature will be invalid and therefore an IKM Tools signature shall apply from then on when exporting the IP.

Verification Method: Test

Requirement ID: IKM-SRS-203

The IKM Tools shall sign the IP with a pre-defined signature established for all outgoing IPs that may supersede the original signature when the IP was imported.

Verification Method: Test

Requirement ID: IKM-SRS-204

The IKM Tools workspace shall support the publication, distribution, subscription and delivery of information.

Verification Method: Test

Requirement ID: IKM-SRS-205

The IKM Tools workspace shall support passive and active subscriptions (aka push and pull) so that users can subscribe to events in the workspace (new document added, modified, deleted) and also an Administrator can subscribe users on their behalf (i.e.: to get news notifications).

Verification Method: Test

Requirement ID: IKM-SRS-206

The IKM Tools shall provide the ability to send notifications from the Operational Network to the Protected Network, as required by business processes.

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-207

The IKM Tools shall provide support publication and distribution of information products to and from integrated and federated services. This includes support for the management of the content lifecycle at the "Retrieval, Use, Accessibility and Transmission" phase.

Verification Method: Analysis

Requirement ID: IKM-SRS-208

The IKM Tools shall support the long-term preservation and restoration of archived content within the organisation and with its parent organisation (from staff elements through to the NATO Archivist). This includes support for the

management of the content lifecycle at the "Planning", "Retrieval, Use, Accessibility and Transmission" and "Disposition" phases.

Verification Method: Analysis

Requirement ID: IKM-SRS-209

The IKM Tools shall support retention and disposition IPs policies so that an authorized user can set up retention times, metadata workflow changes in the IP, IP location changes and other related archiving mechanisms to ensure a consistent and durable IP archiving.

Verification Method: Test

Requirement ID: IKM-SRS-210

The IKM Tools shall implement an archiving mechanism following the Industry accepted standard framework Open Archival Information System. Notably implement the two packages:

Verification Method: Analysis

- Submission Information Package (SIP): for the information products sent to the archive (compatible with NATO Docs system)
- Dissemination Information Package (DIP): for the information sent to a user when requested

Requirement ID: IKM-SRS-211

The IKM Tools shall be able to automatically replicate information from one portal to another in line with predefined rules.

Verification Method: Test

Requirement ID: IKM-SRS-212

The IKM Tools shall support multiple domain publishing so that user can publish information in various domains within the same security domain and being commensurate with the user permissions.

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-213

3.4.1.2 The IKM Tools shall provide mechanism to record physical media information with metadata, including their physical location.

Verification Method: Analysis

Sharing

To empower users in the NATO Enterprise to seamlessly work together in workflows a Sharing capability is envisaged in IKM Tools. This capability enables users to set and get notifications as part of the workflows actions regardless where they sit, if in the ON or in the PBN.

As sharing also entails IP collaboration allowing multiple users to simultaneously edit a document, add comments and review modifications.

Requirement ID: IKM-SRS-214

The IKM Tools shall provide the ability to replicate selected information products from the Protected Business Network to the Operational Network using the ITM services. This information shall be made them available as required by business processes and security permissions.

Verification Method: Test

Analysis

Requirement ID: IKM-SRS-215

The IKM Tools shall mark in the information product metadata to indicate whether it has been send or replicated to another domain: from PBN to ON or other

Verification Method: Inspection

Requirement ID: IKM-SRS-216

The IKM Tools shall support the controlled access and use of content (ACL, OLP, DRM, etc.) and thereby ensure managed security, access and protection of content. This applies to the entire content lifecycle from creation through to final destruction, particularly at the "Storage and Protection" phase.

Verification Method: Analysis

Requirement ID: IKM-SRS-217

The IKM Tools shall include the addressee user(s) or user group within the information product's metadata for the destination domain to rightfully distribute it. (e.g.: "To" field)

Verification Method: Inspection

Requirement ID: IKM-SRS-218

The IKM Tools shall retrieve the transferred information product from the ITM stored location and upload it into a library in IKM Tools accessible only to the addressee(s)

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-219

The IKM Tools shall uniquely identify an information product by a "Durable Link" that is fix along the time regardless the information product's change of location or name. Specifically for TT+ or EDMS documents being moved to a record center

Verification Method: Test

Requirement ID: IKM-SRS-220

The IKM Tools shall make use of the ITM available cross domain services for transferring an information product from PBN to ON

Verification Method: Test, Analysis

Requirement ID: IKM-SRS-221

The IKM Tools shall provide a mechanism to support knowledge transfer between staff: information such as emails, documents, links and relevant data. Specially indicated for staff rotation so that the replacement staff can easily access formal staff information, preserving access control and allowing Administrators to oversee the information.

Verification Method: Analysis

3.4.2 Collaborative Workspace

The collaboration workspaces are very important for a comprehensive user experience when cooperating to achieve defined goals. The workspaces are shared areas among a set of participants with functional elements that enables work together, on line and also offline.

These areas are sites in the IKM Tools portal with defined governance, information organization, look and feel and functionalities.

As part of the IKM Tools they leverage the existing underlying services provided by ITM (for storage and information transfer) and ITM Share Point platform for the User experience (GUI and end user functionality).

The concept of workspace spans from the Share Point Out of the Box templates to the NIP templates (Collaboration Sites) and IKM Tools templates (see section 3.4.2.3 Templates).

Some of these workspaces behaviour may change according to where they are deployed. For instance a Collaboration Site will not hold the same user community and roles if it's on the ON or in the PBN facing internet where additional roles may be defined).

3.4.2.1

Collaboration

There are various types of collaboration according to these factors:

- Security: with different degrees of openness a workspace can be fully open for anyone to contribute to semi open with a restricted area for members to a closed hold for only members access and contributions
- Timeframe: whether it is intended for a short time collaboration and then disposal (e.g.: an event workspace) or for long lasting period (CoI collaboration workspace).

The IKM Manager should take the appropriate actions to coordinate when site creation, during transition and operation and disposal. To maintain the quality and availability of information during the workspace lifecycle.

Also to ensure established policies of NATO Commands are enforced for the correct workspace operations.

Requirement ID: IKM-SRS-222

The IKM Tools shall support synchronous and asynchronous communication between users of the workspace.

Verification Method: Analysis

Requirement ID: IKM-SRS-223

The IKM Tools workspace shall support the search and discovery of workspace information leveraging the existing Search Services in the IKM Tools. This means the workspace information is indexed and searchable by Search Services and also available in the workspace as another Search Scope (or Search Data source).

Verification Method: Analysis

Requirement ID: IKM-SRS-224

The IKM Tools shall leverage the Microsoft OneDrive provided by ITM, a synchronized area where to sync user's local machine information. This information is therefore accessible offline and synchronized with the IKM Tools whenever the user's device becomes reachable to IKM Tools.

Verification Method: Analysis

Requirement ID: IKM-SRS-225

The IKM Tools shall provide the ability to create shared workspaces based on topics (i.e. Community Sites) for authorized users to manage, share, upload and store information related to a topic.

Verification Method: Analysis

Requirement ID: IKM-SRS-226

The IKM Tools shall allow authorized users to configure and customize the workspace for look and feel, content sharing and collaboration notifications (alerts, RSS)

Verification Method: Analysis

Requirement ID: IKM-SRS-228

The IKM Tools shall provide users with a mechanism to leave messages or comments addressed to other users in the workspace or information product. The messages shall include the actual message content plus the author and creation time and only visible to the addressee.

Verification Method: Test

Requirement ID: IKM-SRS-229

The IKM Tools shall ensure the IPs can be protected from unauthorised modification, dissemination, viewing and printing. So that an authorized user can set up Information Rights Management (IRM) restrictions to a particular IP.

Verification Method: Test

Reporting

Templates

3.4.2.2 Aside from the Share Point out of the box templates, there will be the NATO Enterprise Portal Templates which will be tailored Share Point templates for specific use cases. These templates will be already implemented by another project and the IKM Tools will integrate them by making them available in the Catalogue for its instantiation.

3.4.2.3

Requirement ID: IKM-SRS-230

The IKM Tools shall enable to instantiate the NATO Enterprise Templates from the Catalogue:

Verification Method: Test

- Document Collaboration
- Extranet/Intranet Front Page
- Exercise
- Tasking Workflow
- Project Workspace
- Event Template
- Document Publishing
- Wiki

Requirement ID: IKM-SRS-231

These templates shall be available for the IKM Managers to instantiate following the same NIP acquisition process in the NIP Catalogue when procuring services.

Verification Method: Test

The exact definition of the templates content is derived from the Bi-SC AIS users and Command user's common needs.

Once instantiated they can be modified to better fit the needed functionality by the user (e.g.: changing title label).

Requirement ID: IKM-SRS-232

The IKM Tools shall allow the creation of workspace templates from existing workspaces or as new (from no template)

Verification Method: Test

Requirement ID: IKM-SRS-233

The IKM Tools shall provide a Project Collaborative Workspace template with the following features:

Verification Method: Test

- a workspace where to share information among the project members: administrative details, milestones, deliverables, working documents, minutes and alike
- access adhered to RBAC policies set up by the Project Collaborative Workspace manager role
- connected and synchronized with Microsoft EPM if available so members can track the activities and issues with different views
- different views organizing the project information products: Miscellaneous, Start-up, Initiation, Execution, Closure, Logs, Deliverables, Project Documents, Reporting
- all project documents tagged automatically with the project code and name
- a list of all project members and their correspondent roles and contact details: email, phone and organizational element
- a list of Tasks connected to EPM (if available)
- a list of Issues
- a list of Risks
- template with a look and feel consistent with the existing IAS

Requirement ID: IKM-SRS-234

The IKM Tools shall provide an Event Collaborative Workspace template with the following features:

Verification Method: Test

- a workspace where to share information of the event and accessible to the event members
- access adhered to RBAC policies set up by the Event Collaborative Workspace Manager role
- tag the event with applicable NCMS metadata (i.e. security classification, active/in-active)
- Configurable event type categorization with at least: seminar, exercise, training, conference, visit, meeting categories
- a calendar depicting the event information
- a registration mechanism for participants to register the event with notification emails to and from the requester and the Event Collaborative Workspace Manager role
- a view where to see the participants and their registered metadata
- a mechanism to manage the participants with at least approve, reject, lock and notify operations
- template with a look and feel consistent with the existing IAS

Requirement ID: IKM-SRS-235

The IKM Tools workspace shall provide Community of Interests (CoI) aligned with NIP and EDMS: following similar lifecycle process for definition, procurement, implementation and maintenance. The CoI are workspaces related to a community of users and with a medium-long lasting life unlike Community Sites. The CoI are focused on a capability area (JSR, C2...etc.), therefore with need of document sharing capabilities, versioning and user's cooperation via comments, notes and alike. CoI's are controlled sites to the specific purpose of the community so a IKM Manager role is needed to control access and content management, and in general to set the particular policies of the CoI.

3.4.2.4

Verification Method: Test

Administration

Administering the workspaces is similar to workflows as it is focused on maintaining the correct functioning of the workspace functionality and that will follow the same procedures as described in the workflow section.

Requirement ID: IKM-SRS-236

The IKM Tools workspace shall provide the usage, administration, maintenance and sustainment of web content, information products and metadata throughout their respective lifecycles as in EDMS and NIP, and leveraging the existing IKM Tools Framework Runtime

Verification Method: Test

Requirement ID: IKM-SRS-237

The IKM Tools workspace shall provide the administration, maintenance and sustainment of workspace(s) throughout their respective lifecycles, similar to EDMS and leveraging in the existing IKM Tools Framework runtime

Verification Method: Test

Requirement ID: IKM-SRS-238

The IKM Tools shall allow the creation of workspaces with different degrees of collaboration audiences: closed (for members only), semi closed (with a public area and protected area for members only) and open (for every authenticated user collaboration).

Verification Method: Test

Requirement ID: IKM-SRS-239

The IKM Tools shall allow the creation of workspaces to authorized user via a web GUI.

Verification Method: Test

Requirement ID: IKM-SRS-240

The IKM Tools shall support the management of workspace(s) throughout their respective lifecycles.

Verification Method: Test

Requirement ID: IKM-SRS-241

The IKM Tools workspace shall support Information Management Policies in accordance to The primary directive on Information Management (C-M(2008)0113 (INV)) and the ones defined in the existing IKM Tools Framework runtime. So that policies can be applied to workspaces, its content and imported and exported.

Verification Method: Test

Requirement ID: IKM-SRS-242

The IKM Tools shall grant the user interactions with the workspace in accordance to the user's role (RBAC)

Verification Method: Test

Requirement ID: IKM-SRS-243

The IKM Tools workspace shall support processes for the review and qualification of workspace information for retention, disposition, declassification and destruction.

Verification Method: Test

Requirement ID: IKM-SRS-244

The IKM Tools workspace shall support processes for the backup, archiving and restoration of workspace information.

Verification Method: Test

Requirement ID: IKM-SRS-245

The IKM Tools shall provide a Catalogue of services in the GUI for authorized users to procure services and templates (i.e.: Collaboration Workspace).

Verification Method: Test

Requirement ID: IKM-SRS-246

The IKM Tools Catalogue shall follow a process by which a user can request a new service, a new feature or a Change Request and adheres to the IKM Governance Body process (ref: BICMB)

Verification Method: Test

Requirement ID: IKM-SRS-247

Services in the Catalogue shall include at least the IKM Tools (NIP, EDMS, TT+) and templates defined in section 3.4.2.3 Templates.

Verification Method: Test

3.4.3 Services

The Workspace Application needs to be available for external systems and FASs to provide collaboration spaces integrated with their systems. For that the IKM Tools must offer interfaces that are well known and adhered to recognized standards to facilitate the implementation.

Requirement ID: IKM-SRS-248

The IKM Tools workspace shall comply with the Content Management Interoperability Services v.1.0 (CMIS) standard when exposing its functionality as web services

3.4.3.1

3.4.3.2 *Verification Method:* Analysis

Web Services

Collaboration Services

Requirement ID: IKM-SRS-249

The IKM Tools shall integrate with the SOA and IdM Information Discovery Services to retrieve information products as a federated search

Verification Method: Test, Analysis

Presentation Services

Requirement ID: IKM-SRS-250

The IKM Tools shall support the navigation and browsing of the workspace and workspace information as in EDMS and NIP, and leveraging the IKM Tools Framework Runtime

3.4.3.3

Verification Method: Inspection

Requirement ID: IKM-SRS-251

The IKM Tools shall support the physical reproduction of still imagery and textual information as in EDMS and NIP

Verification Method: Inspection

Requirement ID: IKM-SRS-252

The IKM Tools shall allow a configurable presentation of information products, lists and mash-ups according to a selected criteria as in EDMS and NIP and leveraging the IKM Tools Framework Runtime.

Verification Method: Inspection

Requirement ID: IKM-SRS-253

The IKM Tools shall support configurable views and representations of information products and workspace(s).

Verification Method: Inspection

Requirement ID: IKM-SRS-254

The IKM Tools shall allow views and pre-views of information products within the web GUI.

Verification Method: Test

3.4.4 EDMS extension

Similarly to the TT+ extension the EDMS extension is to deliver identified capabilities not implemented in the first IAS Step 1.

Requirement ID: IKM-SRS-255

The IKM Tools Workspace shall be available on the Operation Network (ON) and on the Protected Business Network (PBN)

Verification Method: Test

Requirement ID: IKM-SRS-256

The IKM Tools shall support the management, administration, maintenance and sustainment of content and of their supporting systems (in conjunction with SMC, IA and other services) throughout the content lifecycle. This includes content modelling and mapping, supportive workflows, content de-duplication,

correction and enrichment, master data, monitoring, analytics, reporting and logging. This includes support for the management of the entire content lifecycle.

Verification Method: Analysis

Requirement ID: IKM-SRS-257

The EDMS navigation shall be as the NIP navigation, e.g.: Use the backward and forward navigation functionality like in the NIP "breadcrumb trail".

Verification Method: Test

Requirement ID: IKM-SRS-258

The EDMS navigation shall allow the user to navigate forward and backwards, similar to the NIP navigation. i.e.: Go to the Record Center and return back to the initial page.

Verification Method: Test

Requirement ID: IKM-SRS-259

The EDMS shall depict the name Records Centre when referring to Record Centre in all pages

Verification Method: Inspection

Requirement ID: IKM-SRS-260

The EDMS shall be able to move and copy minor and major versions documents by keeping the versioning and permissions.

Verification Method: Test

Requirement ID: IKM-SRS-261

The EDMS shall mark the documents that have been send to the Records Centre in its metadata field 'Send to' with value 'Record Centre'

Verification Method: Test

Requirement ID: IKM-SRS-262

The EDMS shall change title Related tasker(s) to Related Tasker

Verification Method: Inspection

Requirement ID: IKM-SRS-263

The EDMS shall remove 'Follow' button from the Office inbox page

Verification Method: Inspection

Requirement ID: IKM-SRS-264

The EDMS shall allow the user to upload multiple files

Verification Method: Test

Requirement ID: IKM-SRS-265

The EDMS shall allow approval and validation process in each step of the workflow: publishing, recording (convert to PDF) and archiving (store in Record Centre), so that an authoritative user can approve/validate the document in each of these steps.

Verification Method: Test

3.5 Current IKM Tools (NIP, EDMS and TT+) functionality

The current IKM Tools delivered functionality is specified here to clarify and further extend the code the Purchaser handed out to the Contractor.

If any function is unclear or not well defined the provided IKM Tools functionality shall prevail, i.e.: by executing Create new Tasker in TT+ function will expose what's required by the Contractor, the implementation specifics can vary but the outcome and functionality flow shall be the same.

Requirement ID: IKM-SRS-266

The Contractor shall deliver, at least, the same functionality as the current IKM Tools.

Verification Method: Test

3.5.1 NATO Information Portal (NIP)

3.5.1.1

General

3.5.1.1.1 Introduction

The NATO Information Portal (NIP) is the Web Hosting service on the Bi-SC AIS high side network (Operational Network, ON). Its technology is based on MS SharePoint 2013 delivered from Bi-SC AIS secure centralised Data Centres geo-redundant (Mons and Naples).

The NIP business model is “cloud-oriented” with the purpose of being the single point of access (SPOE) on NS Network.

The NIP aim is to decommission NATO Web Information Services Environment (WISE) portal capabilities scattered around the Bi-SC AIS

The NIP service is providing interface to:

- Formal Workspaces and Records: EDMS
- Formal Taskers: TT+

It is developed as a web portal to fulfil the following functions:

- Web Content Management
- Search

- Social Computing and Networking
- Information and Knowledge Publishing
- Metadata Management

3.5.1.1.2 Integration

3.5.1.1.2.1 Search

Requirement ID: NIP-1

Contractor shall deliver an External Search Service independent from the NIP/EDMS/TT+ environment.

Verification Method: Test

Requirement ID: NIP-2

Consume External Search Service. NIP/EDMS/TT+ shall be able to be indexed by an external SharePoint based search service. While preserving the NIP access content permissions, so only users allowed to access NIP content can search it (not applicable to metadata).

Verification Method: Test

Requirement ID: NIP-3

The External Search Service shall use only the SharePoint permission trimming feature.

Verification Method: Test

Requirement ID: NIP-4

The Search service shall provide the following filters: Originating Office, Type of the item (list item, pdf, xls), Originating Command, Author, etc.

Verification Method: Test

Requirement ID: NIP-5

The Search Service shall provide a faceted search by Search Tabs: Search Tabs are search properties based on the content types. Example: a user will like to find all articles which are related to SHAPE IKM, the user will use the Articles tab and enter “SHAPE” in the search box.

Verification Method: Test

Requirement ID: NIP-6

The Search Service shall implement the following Tabs: Articles, Events, Individuals, Exercises, Countries, Visits, Operations and Everything.

Verification Method: Test

3.5.1.1.2.2 Content Types

Requirement ID: NIP-7

The NIP shall be able to reuse content types that are being provided through a SharePoint based content type hub.

Verification Method: Test

3.5.1.1.2.3 MS Office

Requirement ID: NIP-8

The NIP shall be fully compatible with Microsoft Office 2010 and 2013 .

Verification Method: Test

3.5.1.1.2.4 Compatibility

Requirement ID: NIP-9

The NIP shall be fully compatible with ITM provided SharePoint on premises.

Verification Method: Test

3.5.1.1.2.5 Time zone

Requirement ID: NIP-10

The NIP shall be able to reuse a configurable date and time format on Site Collection level to set/reset the date/time of the whole NIP instance.

Verification Method: Test

3.5.1.1.2.6 Infrastructure

Requirement ID: NIP-11

The NIP shall be able to be installed as a stand-alone application within a standard SharePoint (ITM SharePoint version) farm.

Verification Method: Test

Requirement ID: NIP-12

The NIP shall be compatible with being installed into an existing SharePoint environment as a dedicated Web Application.

Verification Method: Test

Requirement ID: NIP-13

The NIP shall be compatible with being installed into an existing SharePoint environment and shall be able to consume global metadata from an existing term store.

Verification Method: Test

3.5.1.1.3 High level design

Requirement ID: NIP-14

The NIP shall be designed in layers so that it is highly scalable horizontally and vertically in the different functions: web front end, dedicated services like Workflow and Search) and application and database layers, see current architecture as a sample in the following figure:

Verification Method: Test

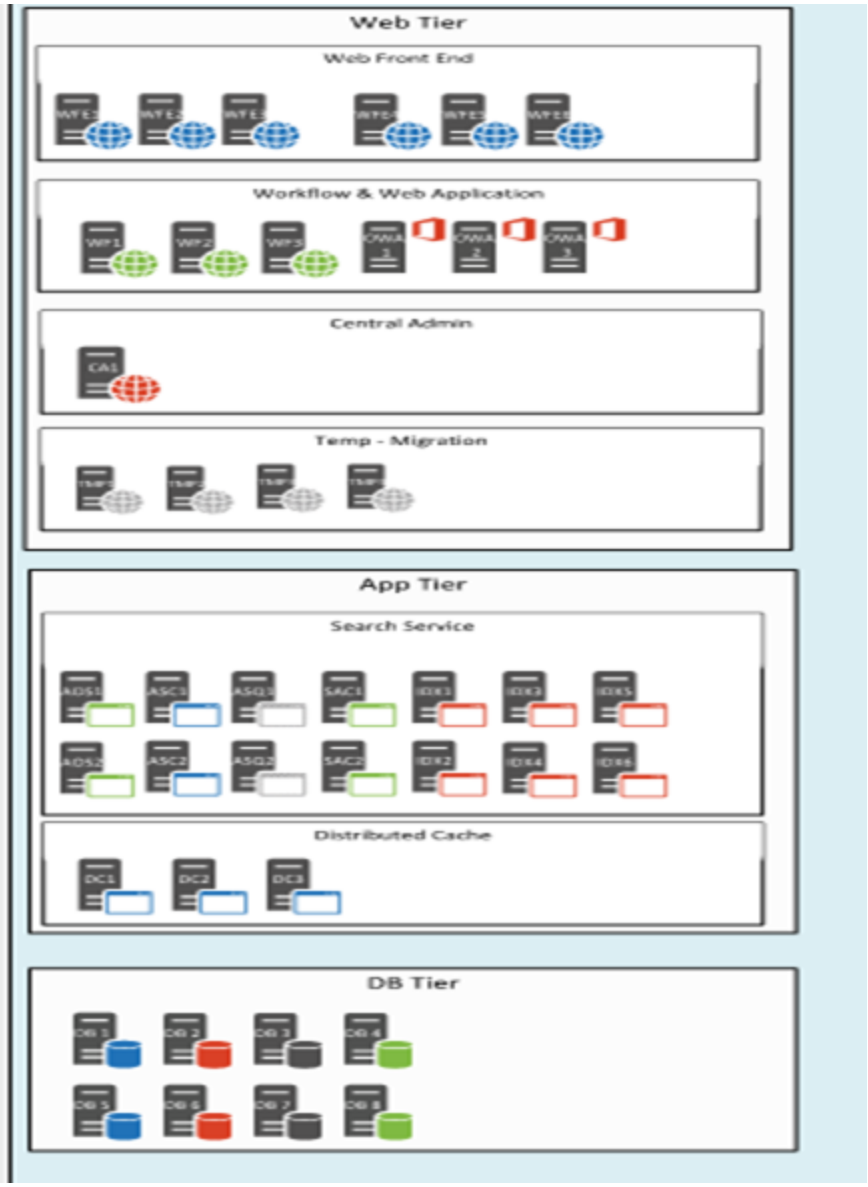


Figure 12 Farm Topology

Requirement ID: NIP-15

The NIP shall follow the accepted Industry standards when designing the system, specifically the Microsoft best practices.

Verification Method: Test

3.5.1.1.3.1 High Level Design

Requirement ID: NIP-16

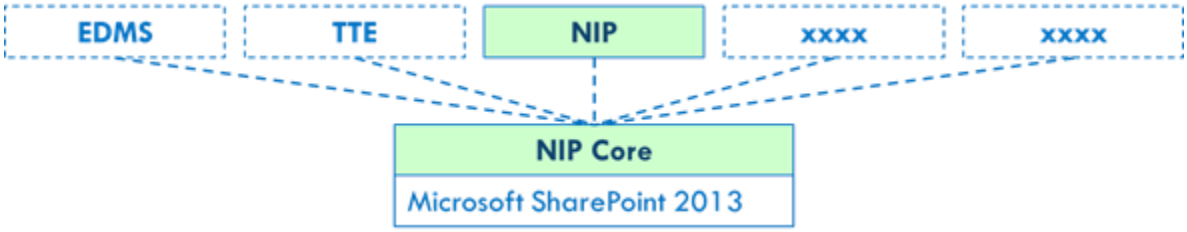
The NIP portal shall be modular to re-use existing services in the platform

Verification Method: Test

Requirement ID: NIP-17

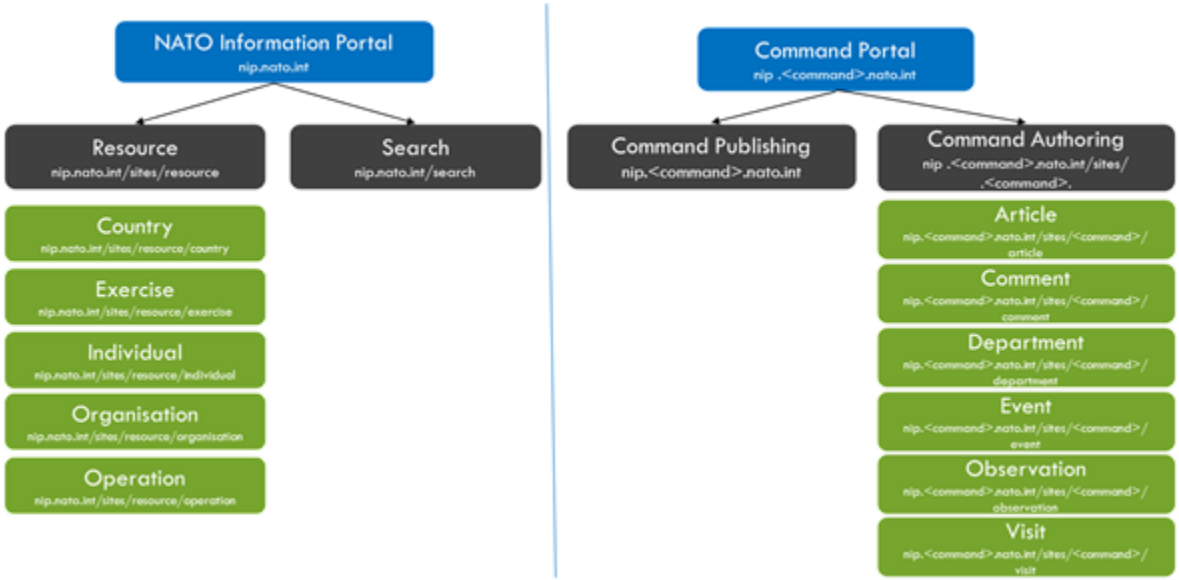
The NIP portal shall re-use the IKM Tools Framework (aka NIP Core) services for underlying dependant services like: metadata, content types, sync services and alike, see the following diagram:

Verification Method: Test



3.5.1.1.3.2 Architecture

NIP has two main areas: The resources subsite and the search subsite, see the following diagram:



Requirement ID: NIP-18

Under resources, administrators shall be able to manage Countries, Individuals, Exercises, Operations, and Organisations. Once created, these items are visible and can be used by anyone in the system.

Verification Method: Test

The Search subsite provides enterprise search functionality which allows users to find NIP content, as well as TT+ and EDMS data.

Requirement ID: NIP-19

Each Command portal shall consists of an authoring site and a publishing site.

Verification Method: Test

Users can create and edit content through the authoring site. Once the user saves the content, it is pushed over to the publishing site so that it can be visible to other users throughout the system

3.5.1.1.3.3 Top Level Sites

Requirement ID: NIP-20

The NATO Information Portal (NIP) shall host top level sites each requiring a DNS entry and a SSL certificate (or a wildcard certificate for the entire portal).

Verification Method: Test

Requirement ID: NIP-21

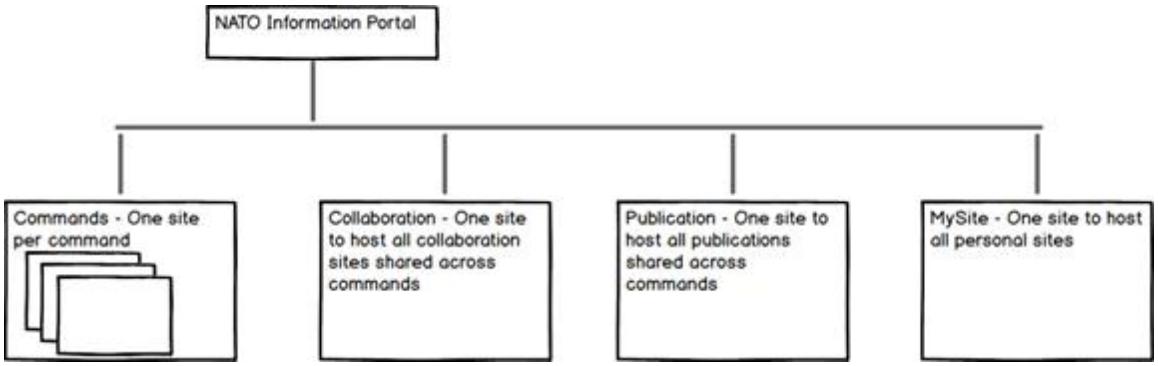
Top level sites shall be deployed as part of the portal implementation and cannot be deployed via the catalogue; additionally, all top level sites are prescriptive and under the technical administration of the NIP technical support team.

Verification Method: Test

Requirement ID: NIP-22

Each command shall have a top level site. Two additional top level sites are created to host publication and collaboration sites, deployed via the catalogue, and shared across the NATO organization. One final top level site will be used as the entry point for the users' personal sites. See the following diagram:

Verification Method: Test



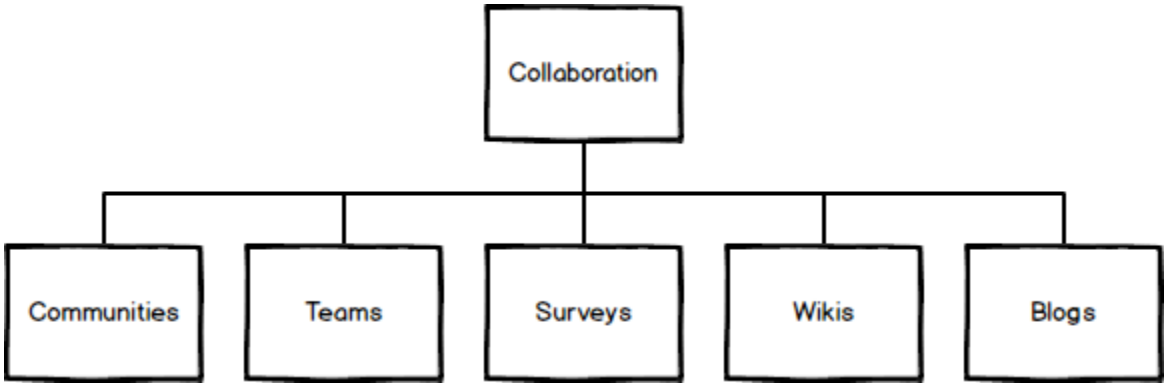
3.5.1.1.3.4 Prescriptive and Free-Form Sites

Requirement ID: NIP-23

Authorized users shall be able to automatically deploy, via the catalogue, additional enterprise level sites, both prescriptive and free form under the

corresponding top level portal site. Free-form collaboration sites are provisioned in a predefine location (path) depending on their type.

Verification Method: Test

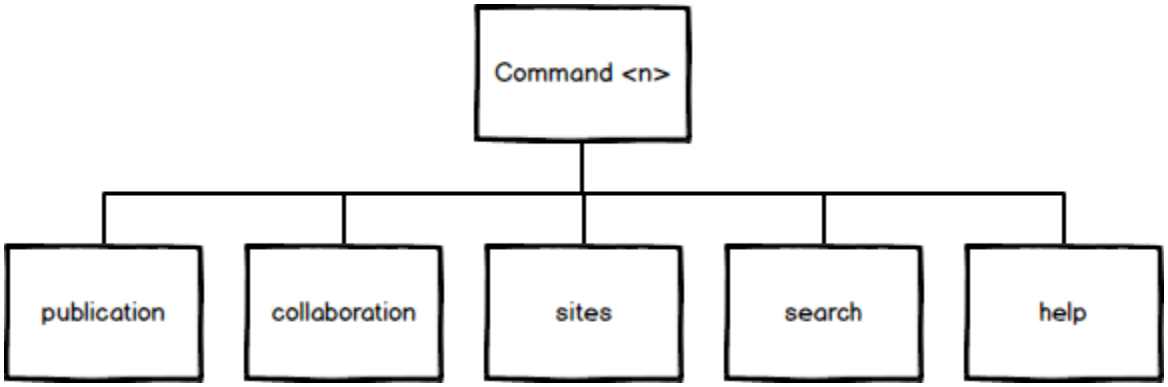


3.5.1.1.3.5Command Sites

Requirement ID: NIP-25

Authorized users shall also request sites (prescriptive or free-form) from the catalogue to be linked to their specific command. Those sites are provisioned in predefined paths (depending on their type) belonging to the specific command.

Verification Method: Test



3.5.1

Functionality

3.5.1.2.1 NIP Articles

NIP users can use this feature in order to inform other users about changes, new regulations, events or any kind of news which can be related to work. Inside of an article the author can add images, tables or text. The articles can be shared with other departments, displayed on the command page or to departments from other Commands.

Requirement ID: NIP-26

NIP shall be able to provide the option to create articles

Verification Method: Test

Requirement ID: NIP-27

NIP shall provide an Article WebPart which will display the latest 3 articles created for the selected Command/Department/Community/Nation page.

Verification Method: Test

Requirement ID: NIP-28

NIP shall provide a Shared Articles WebPart in which the users can view the latest 3 articles which are shared with the accessed location.

Verification Method: Test

3.5.1.2.2 Event Management

NIP Events offer the option for the user to promote an event of different categories across the Command or to departments of other Commands. The user has the option to book a conference room for the event, can invite other users to an event (they will receive invitations on their outlook accounts) or request details from the attendees by creating a registration list inside of the event.

Events in NIP are NATO Events, Daily duties which are completed by NATO employees. For example: official visits, Training, Briefings, Meetings, book a conference room for a specific event, invite users to an event.

Requirement ID: NIP-29

NIP users shall be able to create Events of different types, Booking System and Invite users to Events.

Verification Method: Test

Requirement ID: NIP-30

The Functional Admin shall be able to create and maintain Conference Rooms.

Verification Method: Test

Requirement ID: NIP-31

NIP Users shall be able to Book a conference room inside of an Event.

Verification Method: Test

Requirement ID: NIP-32

The Functional Administrators of the NIP Event Management feature shall be able to Approve/Reject/Edit a Booking Request.

Verification Method: Test

Requirement ID: NIP-33

NIP Users shall be able to Register/Invite other NIP users to a NIP Event

Verification Method: Test

Requirement ID: NIP-34

NIP Users shall be able to Register/Invite other NIP users to a conference room booking (inside of an event).

Verification Method: Test

Requirement ID: NIP-35

NIP users shall be able to create a Registration List inside of a NIP Event.

Verification Method: Test

Requirement ID: NIP-80

The NIP User shall have the option to choose from a default list of columns for the Registration List so that:

- The AO can export a registration template and import it in another event
- An AO is only able to select from a default list
- The default list should contain the following <predefined list> and new columns visible to add to the registration
- Users will not be able to see the full list of attendees, but only their own entries

Verification Method: Test

Requirement ID: NIP-36

The NIP User shall receive notifications regarding the status of the requested bookings (Approve/Rejected/Modified)

Verification Method: Test

Requirement ID: NIP-37

The Conference Room Functional Administrator shall receive notifications if two or more bookings requests will overlap.

Verification Method: Test

Requirement ID: NIP-38

The Functional Administrator of a conference room shall receive notifications when a booking request has been created, modified or cancelled.

Verification Method: Test

Requirement ID: NIP-39

NIP shall provide a Calendar in which all the booking request of a command will be displayed.

Verification Method: Test

Requirement ID: NIP-40

NIP shall display an Event Calendar on each Command/Department/Community/Nation page which will display the total of event for the current location

Verification Method: Test

Requirement ID: NIP-41

NIP shall have a Command Calendar in which the users can view all the events from the accessed Command.

Verification Method: Test

Requirement ID: NIP-42

NIP shall have Events WebPart in which NIP users can view the latest 3 events which are posted on the accessed location.

Verification Method: Test

Requirement ID: NIP-81

NIP shall be able to identify a Strategic Level Event from a user NIP department that includes the following metadata elements so that it can be associated with Countries, Individuals and Organizations:

- Related Place
- Related Individual
- Related Organization

This event can then be published and displayed on the NIP Bi-SC calendar as follows:

- The event created in NIP department can be promoted to become a Strategic Level Event
- When selected as a Strategic Level Event, the Display on Command Page (by default) is also selected
- The Event will appear in both the Command and the Bi-SC level calendars
- Related Place, Related Individual and Related Organization metadata elements can be associated to the event.

Verification Method: Test

Requirement ID: NIP-82

NIP shall be able to register a user attendance for an event in NIP. The user is able to choose the necessary fields for the registration from the list. Event Action Officers (AO) can create a Registration Form for an event. The same AO can create views from the list.

Verification Method: Test

Requirement ID: NIP-83

The user shall have the ability to attach a file (if necessary) to the registration. An example of this could be Security Clearance or Passport details.

Verification Method: Test

3.5.1.2.3 Approval Workflow

Requirement ID: NIP-43

NIP shall provide the option for the Content Functional Administrators to approve Events and Article on the Command page.

Verification Method: Test

Requirement ID: NIP-44

Email notifications shall be sent to the Content FA for any event or articles which is requested to be published on the Command Page.

Verification Method: Test

Requirement ID: NIP-45

The NIP approval notification shall contain a link which will redirect the Content FA to the Approval page.

Verification Method: Test

3.5.1.2.4 Alert State

Requirement ID: NIP-46

NIP shall provide the option to display the Alert State of the visited Command.

Verification Method: Test

Requirement ID: NIP-47

NIP shall provide a custom list on the Publishing site which will allow the Functional Administrator to set up the Alert State for the Command.

Verification Method: Test

3.5.1.2.5 Page Classification

Requirement ID: NIP-48

NIP shall display the classification of the visited page based on the keywords in the displayed content (NIP Security Classification metadata).

Verification Method: Test

3.5.1.2.6 Footer

Requirement ID: NIP-49

NIP shall provide the ability for the user to create Footer Links

Verification Method: Test

Requirement ID: NIP-50

NIP shall provide a custom list on the publishing site of each command which will allow the Functional Administrators to create Footer Links

Verification Method: Test

Requirement ID: NIP-51

NIP shall provide the ability for the user to create Footer Categories

Verification Method: Test

Requirement ID: NIP-52

NIP shall provide a custom list on the publishing site of each command which will allow the Functional Administrators to create Footer Categories

Verification Method: Test

3.5.1.2.7 Top Level Navigation

Requirement ID: NIP-53

NIP shall provide a Top-Level Navigation which will be available for NIP users in all pages (NIP Navigation shall follow the NIP Template which is approved by the NIP Governance Board).

Verification Method: Test

Requirement ID: NIP-54

The Top Level Navigation shall contain the following links:

Verification Method: Test

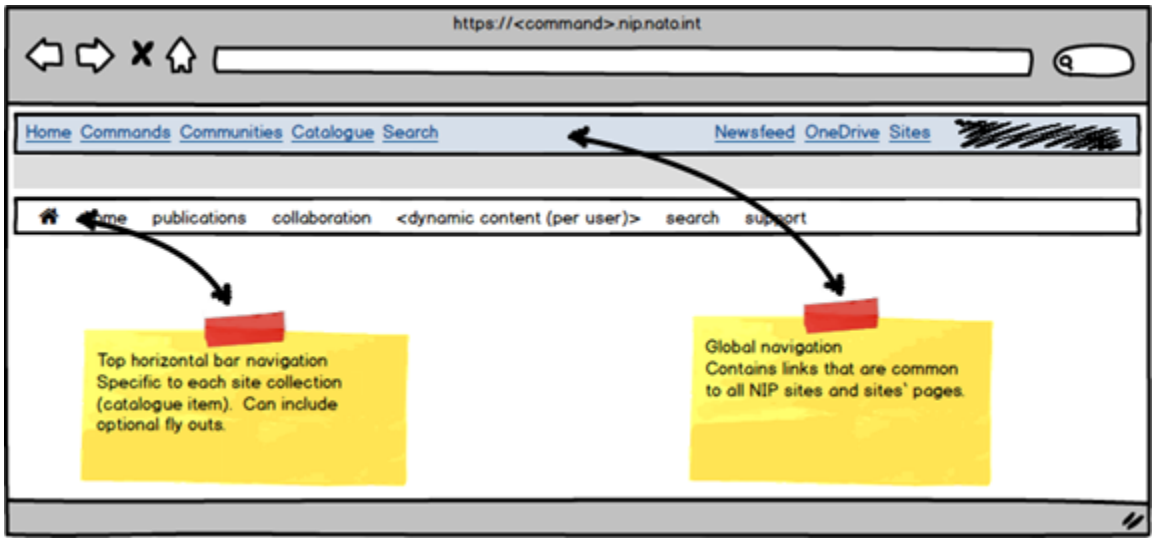
- NIP Commands
- Command Departments
- Command Communities
- Command Nations
- Booking Calendar

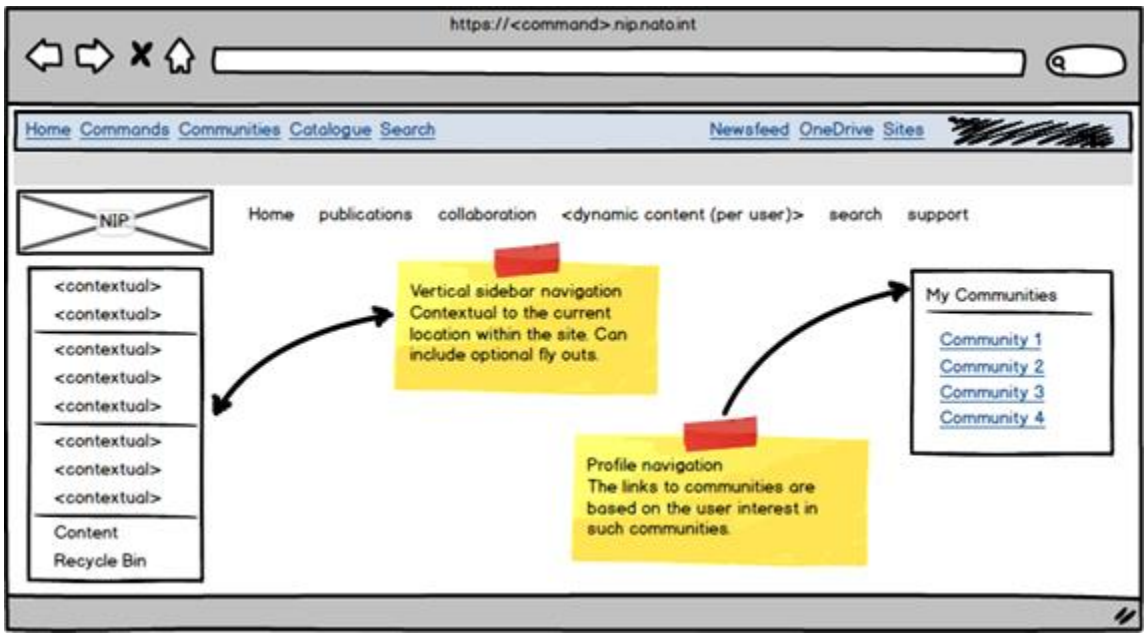
Requirement ID: NIP-55

NIP shall implement five types/levels of site/content navigation:

Verification Method: Test

- Global navigation: Located at the very top of the page for links that are common to all NIP sites and sites' pages.
- Top horizontal bar navigation: specific to each site collection (catalogue item). Can include optional fly outs.
- Vertical sidebar navigation: Available only in free-form sites it is contextual to the specific function performed or location within the site. Can include optional fly outs.
- Search navigation: Used to access content without knowing the exact location. Augmented with facets navigation based on metadata.
- Profile navigation: User is presented with navigation options based on his/her profile and/or browsing history.





3.5.1.2.7.1 Department Users

Requirement ID: NIP-56

NIP Users shall be able to create Departments users

Verification Method: Test

Requirement ID: NIP-57

NIP shall provide a WebPart which will display the department POC's

Verification Method: Test

Requirement ID: NIP-58

The POC WebPart shall have: Display Priority, Option "See more" in order to display more than 5 POC's

Verification Method: Test

3.5.1.2.7.2 Exercises

Requirement ID: NIP-59

NIP Users shall be able to create/edit Exercises.

Verification Method: Test

3.5.1.2.7.3 EDMS WebPart

Requirement ID: NIP-60

The NIP user shall view the latest 10 incoming documents from EDMS Command/Department/Community.

Verification Method: Test

Requirement ID: NIP-61

The WebPart Title shall be linkable and redirect users to the EDMS Workspace

Verification Method: Test

3.5.1.2.7.4 TT+ WebPart

Requirement ID: NIP-62

The NIP User shall view the latest 5 incoming taskers from TT+ Command/Department/Community.

Verification Method: Test

Requirement ID: NIP-63

The WebPart shall be linkable and redirect users to the TT+ page.

Verification Method: Test

3.5.1.2.8 Content Types

Requirement ID: NIP-64

NIP shall use content types whenever possible to ease data modularity and reuse

Verification Method: Test

3.5.1.2.8.1 Department Links

Requirement ID: NIP-65

NIP users shall have the option to create Department Links.

Verification Method: Test

Requirement ID: NIP-66

NIP shall provide a WebPart which will display the Departments Links.

Verification Method: Test

3.5.1.2.8.2 Observations

NIP Observation is the ability for NATO users to make observations related to daily activities and NIP functionalities.

Requirement ID: NIP-67

The user shall be able to add the following category of information: Remedial Notes, Observation Date, Observation details, Activity, Reference to NATO staff functions, Analysis Phase Notes, Function, Discussion, Recommendations and Conclusions.

Verification Method: Test

Requirement ID: NIP-68

NIP Users shall have the option to create Observations.

Verification Method: Test

3.5.1.2.8.3 Countries

Requirement ID: NIP-69

Specific NIP Users shall be able to create Countries and maintain the details (NMR's, NLR's and FA users).

Verification Method: Test

Requirement ID: NIP-70

NIP Country page shall have a WebPart for related individuals

Verification Method: Test

Requirement ID: NIP-71

NIP Country page shall have a Webpart which will display related events.

Verification Method: Test

3.5.1.2.8.4 Individuals

Requirement ID: NIP-72

Specific NIP Users shall be able to create Individuals (NMR's, NLR's and FA users).

Verification Method: Test

Requirement ID: NIP-73

NIP individual page shall have the following WebParts: Active Role, Former Role, Individual Details.

Verification Method: Test

3.5.1.2.9 Metadata

Requirement ID: NIP-74

NIP shall be able to use a combination of NCMS and NIP based metadata and extended NIP required metadata. Metadata fields shall be based on SharePoint columns.

Verification Method: Test

Requirement ID: NIP-75

NIP Metadata shall have at least metadata defined for (see the NIP Hub Metadata Extract for the full metadata taxonomy, content types and field types):

Verification Method: Test

- Documents
- Articles
- Links
- Events
- Countries
- Individuals
- Observations

Requirement ID: NIP-76

NIP shall allow extensions coherent with the NCMS guideline.

Verification Method: Test

Requirement ID: NIP-77

NIP shall allow users to make changes to NON-SYSTEM FILLED field to defaulted values.

Verification Method: Test

Requirement ID: NIP-78

NIP shall allow a user to assign a security classification to an activity.

Verification Method: Test

3.5.1.2.9.1 NIP Page Templates

Requirement ID: NIP-79

NIP shall provide a set of page templates with common base layout and with dedicated permissions per user groups:

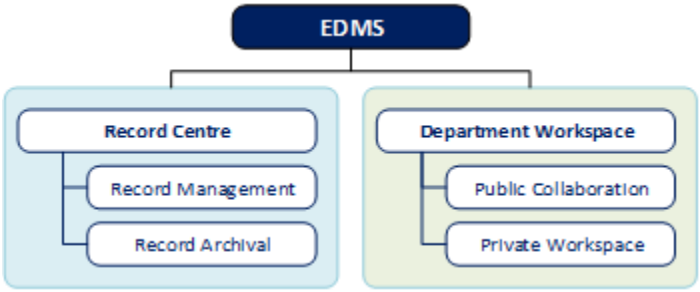
Verification Method: Test

- Command Template
- Department Template
- Community Template
- Nation Template
- Country Template

3.5.2 Enterprise Document Management System (EDMS)

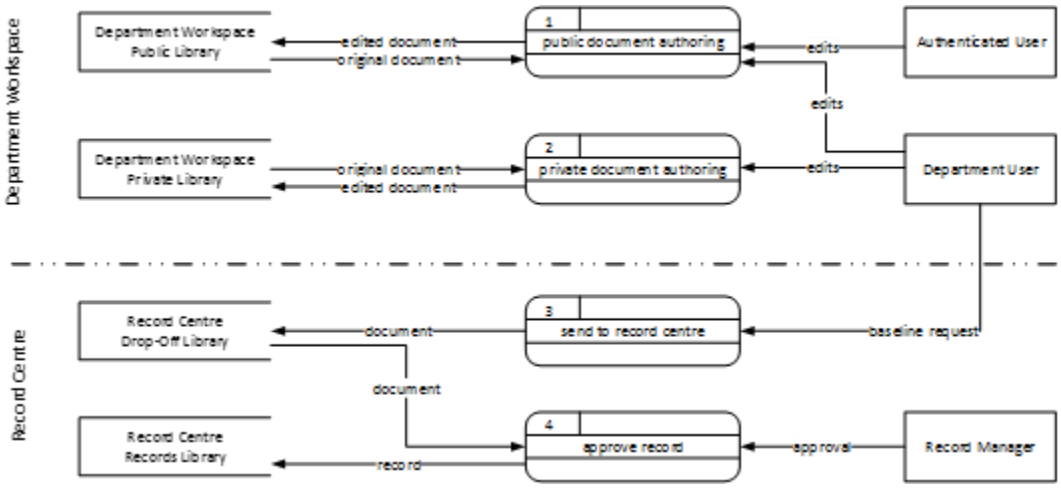
General

3.5.2.1 The EDMS application is comprised of two types of components (or services) – a Record Centre component and a Department Workspace component, see the following figure:



The Record Centre implements records management and archival capabilities, while the Department Workspace allows for document publication and collaboration, as well as private document storage and management.

During the initial rollout of EDMS, each Command will receive a single Record Centre component and multiple Department Workspace components. Department Workspace components will be implemented down to the level of “branch”. The relationship between Department Workspaces and the Record Centre is shown graphically in the Figure 3.



3.5.2.1.1 Introduction

The scope of the EDMS (Enterprise Document Management System) is the implementation of a collaborative workspace and a Document Management System (DMS) to serve as a common repository for Information Products.

The collaborative workspaces and content management repositories developed within the scope of this work package represent the implementation of the NATO Enterprise Document Management System (referred to as EDMS) leveraging Out of The Box (OOTB) technology provided by the NATO Information Portal (NIP) platform based on SharePoint 2013.

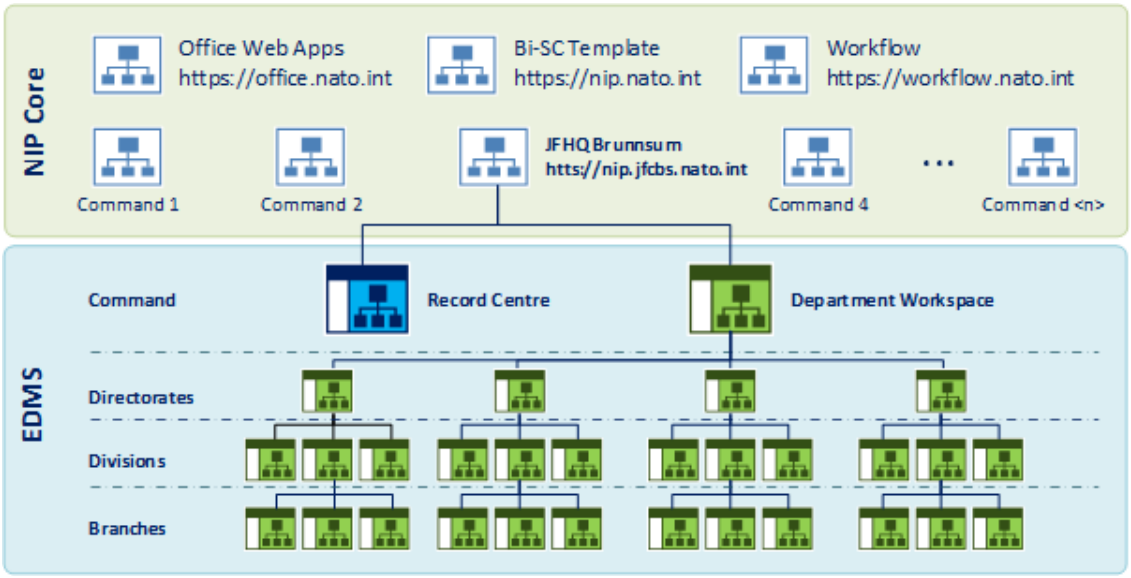
3.5.2.1.2 Integration

Requirement ID: EDMS-1

EDMS shall integrate with NIP Core

Verification Method: Test

EDMS is provisioned as a logical hierarchical structure that mimics that of the NATO Peacetime Establishment (PE) and aligns with the Organisational Units metadata element included in NIP Core. The elements of such structure from top to bottom are Command, Directorates, Divisions, and Branches; and they are collectively referred to as “departments.” See the following figure:



EDMS is currently implemented in the same web application as the NIP Portal but using different URLs and different site collections.

EDMS is deployed on a “per Command” basis. When deployed, the Command receives one Record Centre and one Department Workspace while each Directorate, Division, and Branch within the Command receives one Department Workspace each. Each Department Workspace within the Command is uniquely linked to the Record Centre for that Command (see Figure 3).

Requirement ID: EDMS-2

EDMS shall be instantiated per Command so that each Command can manage his own EDMS workspace.

Verification Method: Test

Record Centres and Department Workspaces are implemented as SharePoint 2013 host named site collections, in alignment with Microsoft best practices for SharePoint 2013 and SharePoint 2016, under the “edms” managed path, while leveraging the URL structure and root Command Template implemented in NIP Core.

3.5.2.1.3 High level design

Requirement ID: EDMS-3

EDMS shall be another component or application of the IKM Tools and it runs along the other tools leveraging their services. See figure below:

Verification Method: Test

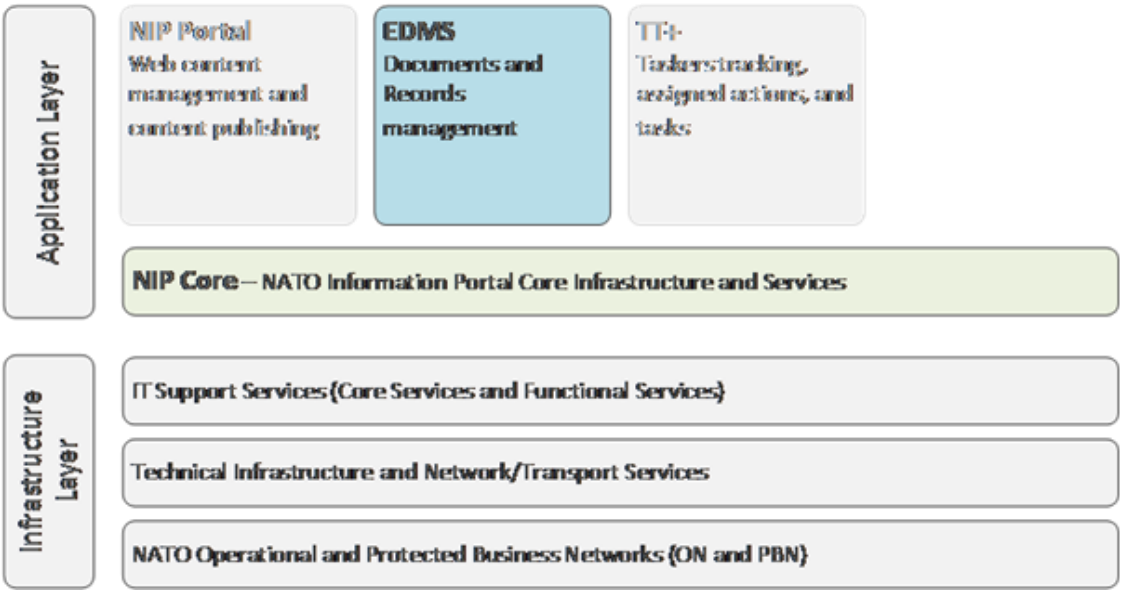


Figure 13 High Level Design

Search

3.5.2.2

Requirement ID: EDMS-4

EDMS shall leverage the Enterprise Search capabilities implemented in Run-Time NIP.

Verification Method: Test

Requirement ID: EDMS-5

EDMS shall allow the user to get search results categorized by at least these domains: Enterprise Domain Command Domain Office Document Type and TT+ domain (i.e.: via filters refiners or scope searches)

Verification Method: Test

Requirement ID: EDMS-6

3.5.2.3 EDMS shall extend the Search scope to the TT+ information source

Verification Method: Test

Functionality

3.5.2.3.1 Look & Feel

Requirement ID: EDMS-7

EDMS interface shall be similar to the current EDMS in operation look and feel.

Verification Method: Test

Requirement ID: EDMS-8

EDMS interface shall provide users with:

Verification Method: Test

Requirement ID: EDMS-9

EDMS shall have a navigation toolbar in the top area of the web pages

Verification Method: Test

Requirement ID: EDMS-60

EDMS shall allow the user to navigate to his departments NIP page from his EDMS workspace or library

Verification Method: Test

Requirement ID: EDMS-10

EDMS shall implement a search bar in the top area of the web pages

Verification Method: Test

Requirement ID: EDMS-11

EDMS shall have the contextual navigation showing site hierarchy and a site browser on the left side of the web pages

Verification Method: Test

Requirement ID: EDMS-12

EDMS shall present contextual content in the centre of the web page.

Verification Method: Test

Requirement ID: EDMS-13

The user shall be able to use the delivered EDMS GUI to browse Office/Branch files commensurate with the user's token (permissions):

Verification Method: Test

- Anonymous Users
- Authenticated Users
- Command Users
- Office/Branch/Department Users

Requirement ID: EDMS-14

EDMS site views shall be pre-defined and appropriate to each site configured using templates equivalent to those on the SHAPE EDMS, including but not limited to:

Verification Method: Test

Requirement ID: EDMS-15

EDMS shall include a Command Site template

Verification Method: Test

Requirement ID: EDMS-16

EDMS shall include a Branch/Office/Department Site Template

Verification Method: Test

Requirement ID: EDMS-17

EDMS shall include a Record Center site template

Verification Method: Test

Requirement ID: EDMS-18

EDMS Portal shall reside in the same Web Application as Run-time NIP.

Verification Method: Test

Requirement ID: EDMS-19

It shall have Site Collections underneath one per Command/Organizational Element.

Verification Method: Test

Requirement ID: EDMS-20

EDMS shall display the aggregated views for Command View

Verification Method: Test

Requirement ID: EDMS-21

EDMS shall display the aggregated views for Bi-SC View

Verification Method: Test

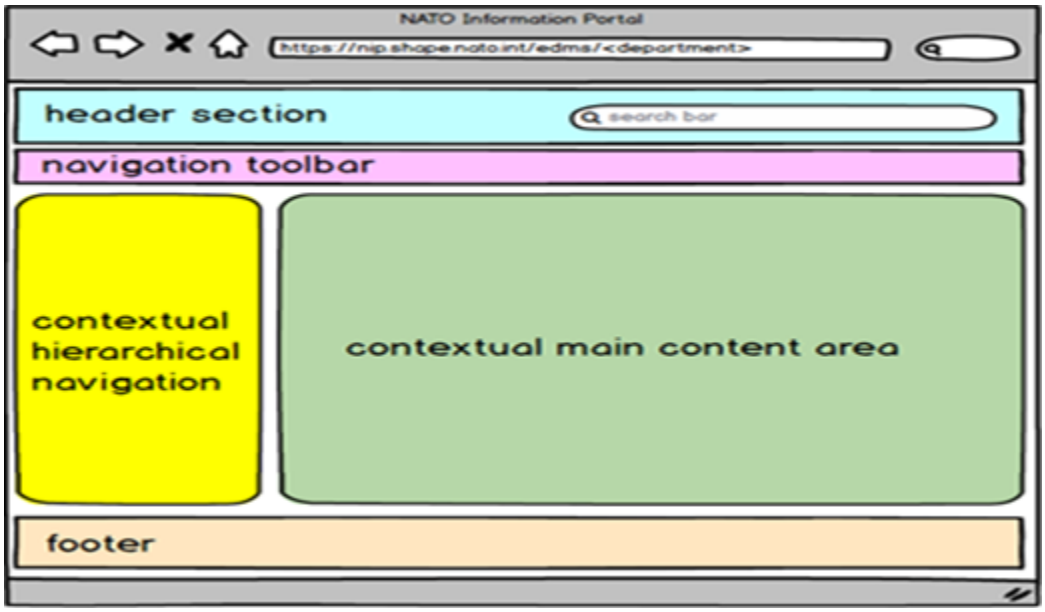


Figure 14 EDMS Page Layout

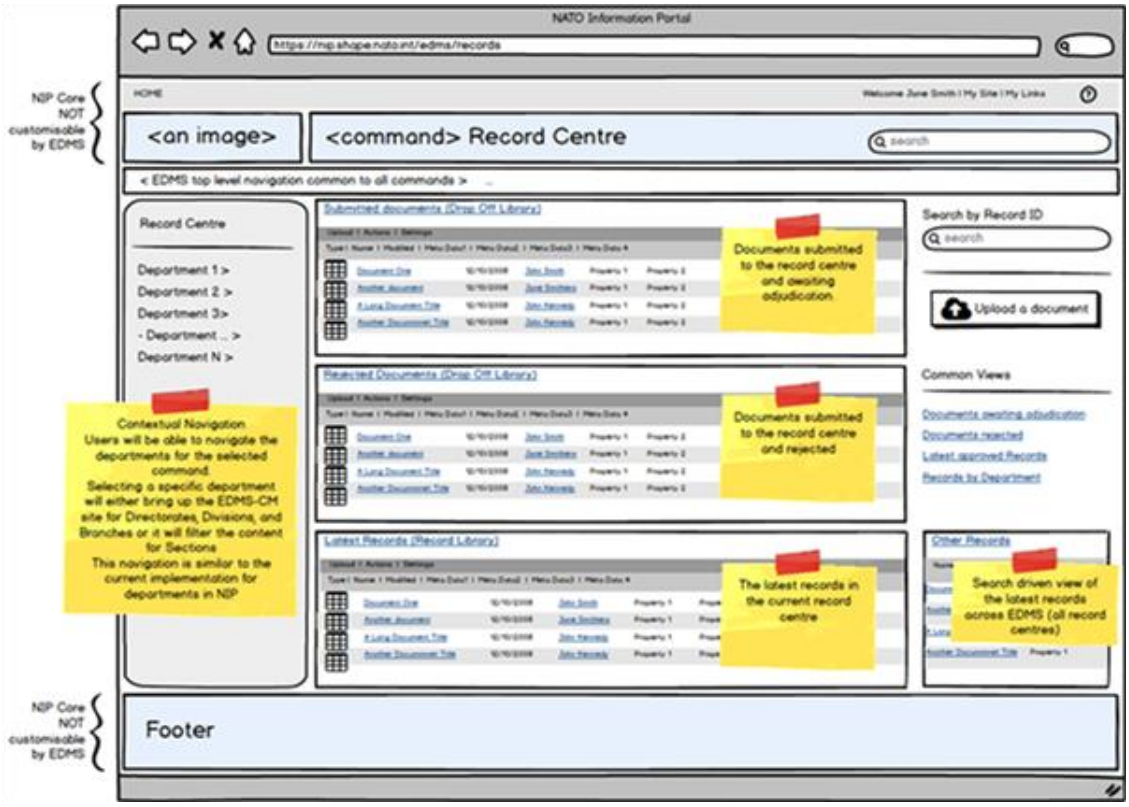


Figure 15 EDMS Landing Page

3.5.2.3.2 Presentation area elements

Requirement ID: EDMS-22

The home page of each branch/office shall display the content of the Record Centre

Verification Method: Test

Requirement ID: EDMS-23

The home page of each branch/office shall display the documents related to the branch/office

Verification Method: Test

Requirement ID: EDMS-24

Record Centre Documents

Verification Method: Test

Requirement ID: EDMS-25

The branch/office home page shall display documents rejected in the Record Centre

Verification Method: Test

Requirement ID: EDMS-26

The branch/office home page shall display documents awaiting approval

Verification Method: Test

3.5.2.3.3 Record Center

Requirement ID: EDMS-27

EDMS shall implement one Record Center per command using the SharePoint 2013 out-of-the-box functionality

Verification Method: Test

Requirement ID: EDMS-28

The Record Center template shall be enriched with NCMS metadata in line with the NIP (i.e: Date Disposition Keywords Publisher Owner)

Verification Method: Test

Requirement ID: EDMS-29

EDMS shall provide a ""'Record Centre Manager"" role with elevated privileges over the content of the record center.

Verification Method: Test

Requirement ID: EDMS-30

Workflow. EDMS shall implement a Workflow for Publishing Documents in the Record Center coherent with the existing EDMS workflow.

Verification Method: Test

Requirement ID: EDMS-31

EDMS shall allow users to send a request for storing a document in the record center commensurate with the requesting user's permissions in accordance with the prescribed workflow.

Verification Method: Test

Requirement ID: EDMS-32

A workflow shall manage the review and approval (or rejection) of documents sent to the record center

Verification Method: Test

Requirement ID: EDMS-33

EDMS shall provide the mechanism to archive documents in PDF format while preserving a copy of the original document format (i.e.: in a hidden store as EDMS)

Verification Method: Test

3.5.2.3.4 EDMS Managed Metadata

Requirement ID: EDMS-34

EDMS shall leverage the functionality for management of metadata implemented in Run-Time NIP, provided by NCMS, in accordance with NATO Core Metadata Specification (NCMS) Implementation Guidance, Version 1.0 Draft 0.1, 14th January 2016.

Verification Method: Test

Requirement ID: EDMS-35

EDMS metadata shall comply with the NCMS

Verification Method: Test

Requirement ID: EDMS-36

EDMS shall re-use the existing NIP metadata

Verification Method: Test

Requirement ID: EDMS-37

EDMS shall allow extensions coherent with the NCMS guideline

Verification Method: Test

Requirement ID: EDMS-38

EDMS shall pre-fill the maximum metadata default values for documents, allowing the user to verify and make final changes.

Verification Method: Test

The Communities of Interest metadata differs from the department in that the terms are reused from the sub term layer “Department>Community”

The Shared documents Web part on the Workspace does not function correctly as a result of this.

Requirement ID: EDMS-41

The same content types shall be available as departments'

Verification Method: Test

Requirement ID: EDMS-42

Permissions shall be implemented in the department workspaces for the different access: Private Documents only members of the current department have contributor permissions and for Public Libraries all Authenticated Users can contribute.

Verification Method: Test

3.5.2.3.5 User Information

Requirement ID: EDMS-43

EDMS shall leverage the User Profile synchronisation functionality implemented in Run-Time NIP.

Verification Method: Test

3.5.2.3.6 Content Types and templates

Requirement ID: EDMS-44

In respect of EDMS, Navigation NIP Terms are re-used from the NATO Domain Group , Specifically from the “Organisational entity” term set predominantly sub term layer “Department>Command” . The following EDMS Content types shall be available in the Content Hub for NIP

Verification Method: Test

- EDMS Document
- Directive
- Letter
- Point Paper
- Spreadsheet
- Presentation

Requirement ID: EDMS-45

The Communities of Interest metadata shall be similar to the department metadata except that the terms are reused from the sub term layer “Department>Community”

Verification Method: Test

- The Shared documents Web part on the Workspace does not function correctly as a result of this.
- The same content types are available as departments
- Permissions reflect department workspaces (e.g. Private Documents libraries have Anonymous access disabled)

Requirement ID: EDMS-46

EDMS shall provide with the content types and templates equivalent to the current EDMS document templates in SHAPE, including but not limited to:

Verification Method: Test

- EDMS shall include a content type for "Military Letter"
- EDMS shall include a content type for "Memo"
- EDMS shall include a content type for "Directive"
- EDMS shall include a content type for "Document"
- EDMS shall include a content type for "Personal Letter"

Requirement ID: EDMS-47

The templates shall retrieve metadata from the original uploaded document, including but not limited to:

Verification Method: Test

- EDMS shall automatically populate the creation date for newly created content
- EDMS shall automatically populate the author metadata with the identity of the user creating a new document
- EDMS shall automatically populate the pre-defined security classification metadata element when creating new content
- Mandatory fields, as described in the metadata mappings, shall be auto-populated, with zero user-interaction, unless required by the user.

3.5.2.3.7 Versioning and CMS

Requirement ID: EDMS-48

EDMS Document Libraries shall have major versioning enabled by default

Verification Method: Test

Requirement ID: EDMS-49

EDMS Document Libraries shall have minor versioning enabled by default

Verification Method: Test

Requirement ID: EDMS-50

EDMS Document Library shall be configured to retain a predefined number of major versions and a predefined number of minor versions by default

Verification Method: Test

Requirement ID: EDMS-51

EDMS system shall enable users to restore prior versions of a document

Verification Method: Test

Requirement ID: EDMS-52

EDMS shall display the version history for documents

Verification Method: Test

Requirement ID: EDMS-53

EDMS shall leverage SharePoint 2013 versioning capability to manage different versions of a single document.

Verification Method: Test

3.5.2.3.8 Naming Conventions (Users, Command, Branch)

Requirement ID: EDMS-54

The Command/Department home page title shall be the long name of such Command or Department, consistent with the NIP implementation.

Verification Method: Test

3.5.2.3.9 Accessibility & Permissions

Requirement ID: EDMS-55

Each Bi-SC, Command and Branch/Office shall have public spaces where to store information available to all Users. Only the members belonging to each space can contribute on that particular space

Verification Method: Test

Requirement ID: EDMS-56

Command areas shall have a protected space for Command internal working, where only Command Users can access and contribute

Verification Method: Test

Requirement ID: EDMS-57

Branch/Office areas shall have a protected space for Branch/Office internal working, where only Branch/Office Users can access and contribute

Verification Method: Test

Requirement ID: EDMS-58

EDMS shall implement Role Based Access Control (RBAC) permission model. Enabling similar permission control as the current EDMS.

Verification Method: Test

Requirement ID: EDMS-59

Each Office, Department or Branch Site Template shall implement a Public Library that is accessible by all users of the command, consistent with their permissions.

Verification Method: Test

3.5.3 Tasker Tracker Plus (TT+)

TT+ is an application built on top of SharePoint 2013. The application leverages and extends the capabilities of SharePoint by deploying several TT+ core packages and TT+ extension packages to the SharePoint farm. These packages were initially created at NSHQ in Mons, BE over the past several years (then called the Project Management Tool, “PMT”). For the purposes of the TT+ rollout, these packages and solutions are to be industrialized (read: prepared to work in a standardized NATO environment) and packaged for rollout within the NATO static command structure throughout Europe and the USA. There are some modifications that have been agreed upon and included in the project scope of work that will customize PMT to function more similarly to the already existing TTE. This new version of PMT – with the industrialization changes and “TTE” customizations is what comprises the “new” application TT+. The packages mentioned above are divided into “solutions”.

Functional Requirements

3.5.3.1.1 General

Tasker Tracker Plus (TT+) is a SharePoint application for managing the creation of documents in a collaborative environment with the support of a built-in workflow process.

3.5.3.1 TT+ offers the following services:

- Task/Process management for the collaborative creation of documents or collections of documents for a wide variety of purposes.
    - Workflows are based on pre-configured offices containing AD groups (populated with user accounts) for the division of work and permissions.
    - TT+ manages the process of document/task creation including and managing Task status, updates to the status, sends E-mail notifications to receivers of Task activities and provides archiving functionality.
- TT+ uses the standard SharePoint framework (i.e. Document Libraries and Lists) for the management of Activity / Tasker information and documentation (incl. versioning, auditing, metadata, etc.).
- TT+ extends the standard SharePoint functionality to encapsulate all of the necessary TT+ functionality into one place (i.e. Site Collection) and leverage the customization options within SharePoint to deliver the required functionality. Using pre-existing SharePoint Application Programing Interfaces (APIs), TT+ maintains the “feel” of SharePoint and reduces the overall amount of custom code required, ensuring fewer security risks and reduced maintenance effort.
- TT+ includes reporting, document archiving features that have been developed based on “best practices” gathered from other NATO applications (i.e. Tasker Tracker Enterprise (TTE)) and requirements from a wide range of users (i.e. NATO static command, NSHQ).
- TT+ leverages the SharePoint framework including client and server integration components to create an easily understandable and efficient (read: fast) application that can be customized further through very flexible configuration options using SharePoint standards (i.e. configuration mostly through items in SharePoint lists).

3.5.3.1.1.1 Integration

Requirement ID: TTP-1

The IKM Tools shall leverage the ITM provided Chat and Presence Services, being Microsoft Skype for Business or Microsoft Teams.

Verification Method: Test

Requirement ID: TTP-2

The IKM Tools shall integrate with the ITM Share Point delivered platform, including with the ITM subservices for:

Verification Method: Test

Requirement ID: TTP-3

The TT+ application shall be installable via a scriptable and automated process.

Verification Method: Test

Requirement ID: TTP-4

Parameters for the installation shall be managed through the scripted installation files.

Verification Method: Test

Requirement ID: TTP-5

TT+ shall be able to be integrated in the infrastructure monitoring.

Verification Method: Test

Requirement ID: TTP-6

Microsoft best practices shall be used for the development to allow further improvement and bug fixing.

Verification Method: Test

Requirement ID: TTP-8

TT+ shall be able to integrate with EDMS.

Verification Method: Test

Requirement ID: TTP-9

EDMS shall be able to transfer a document within a specified tasker.

Verification Method: Test

3.5.3.1.1.2 High level design

The logical infrastructure of TT+ is described as follows:

TT+ is installed on an existing SharePoint 2013 farm in a pre-existing “WebApp” containing a Site Collection, in which TT+ is created. This Site Collection consists of various Lists and Document Libraries connected with code. TT+ can be configured in Microsoft’s Internet Information Services as either Host Named or Path Based Site Collections, allowing more flexibility and deployment environment options.

Requirement ID: TTP-10

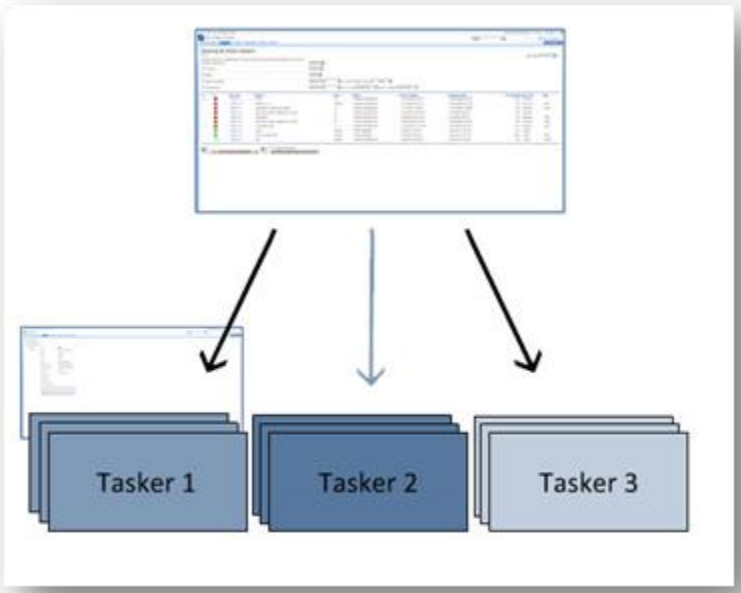
Each Command shall have a TT+ instance they can manage and operate.

Verification Method: Test

Requirement ID: TTP-11

Each TT+ instance shall be able to have multiple Taskers, see the following diagram:

Verification Method: Test



The hierarchy is as follows:

- Each TT+ instance is a Site Collection:
- There is a “Rootweb”, which contains:
    - Configuration lists and libraries
    - The User Interface (UI) for TT+ configuration
    - Configuration Webs, which contain:
        - The configuration lists
        - And are used for workflow setup per TT+ Activity Type
    - Activities (like Taskers) are stored in a central list
    - For each Activity, a workspace is created and connected, which contains:
        - Roles
        - Actions
        - Members

See the following diagram:

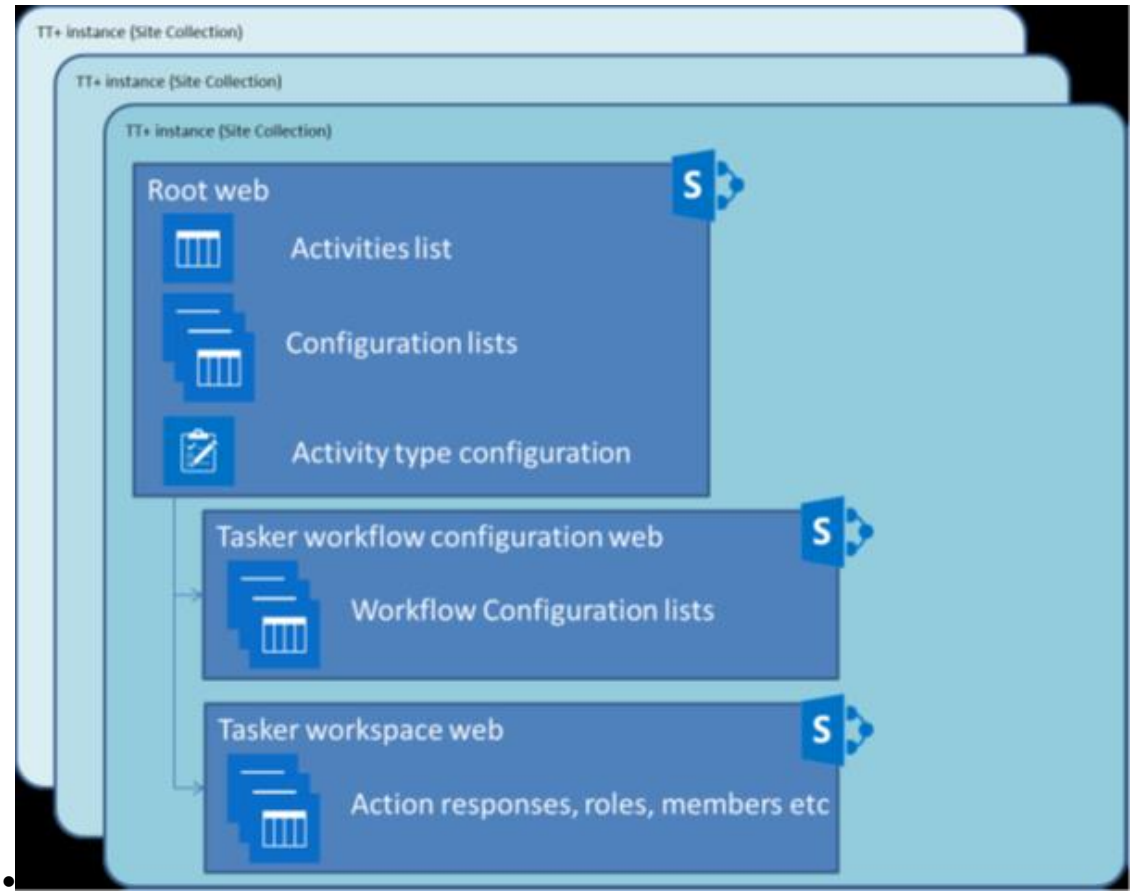


Figure 16 Farm Topology

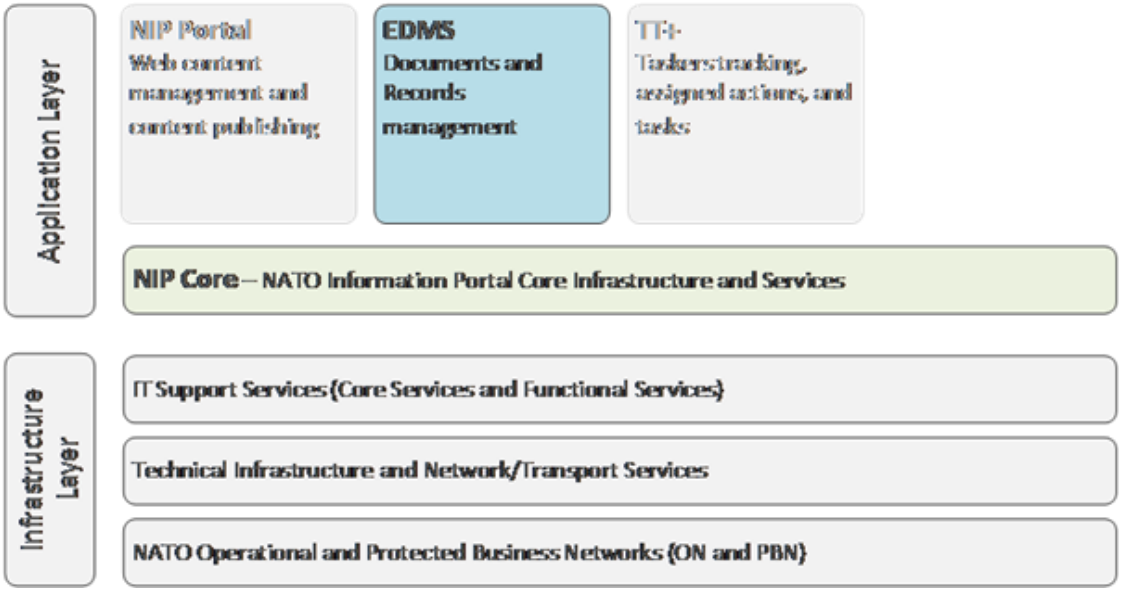


Figure 17 High Level Design

3.5.3.1.2 Tasker as collaborative workflow

Requirement ID: TTP-12

The status of the Tasker shall be updated and the various workflow steps are completed. The different Tasker status are defined based on the linear workflow for TT+, which result from the definition of the Roles within the Tasker, see the following Tasker workflow diagram:

Verification Method: Test

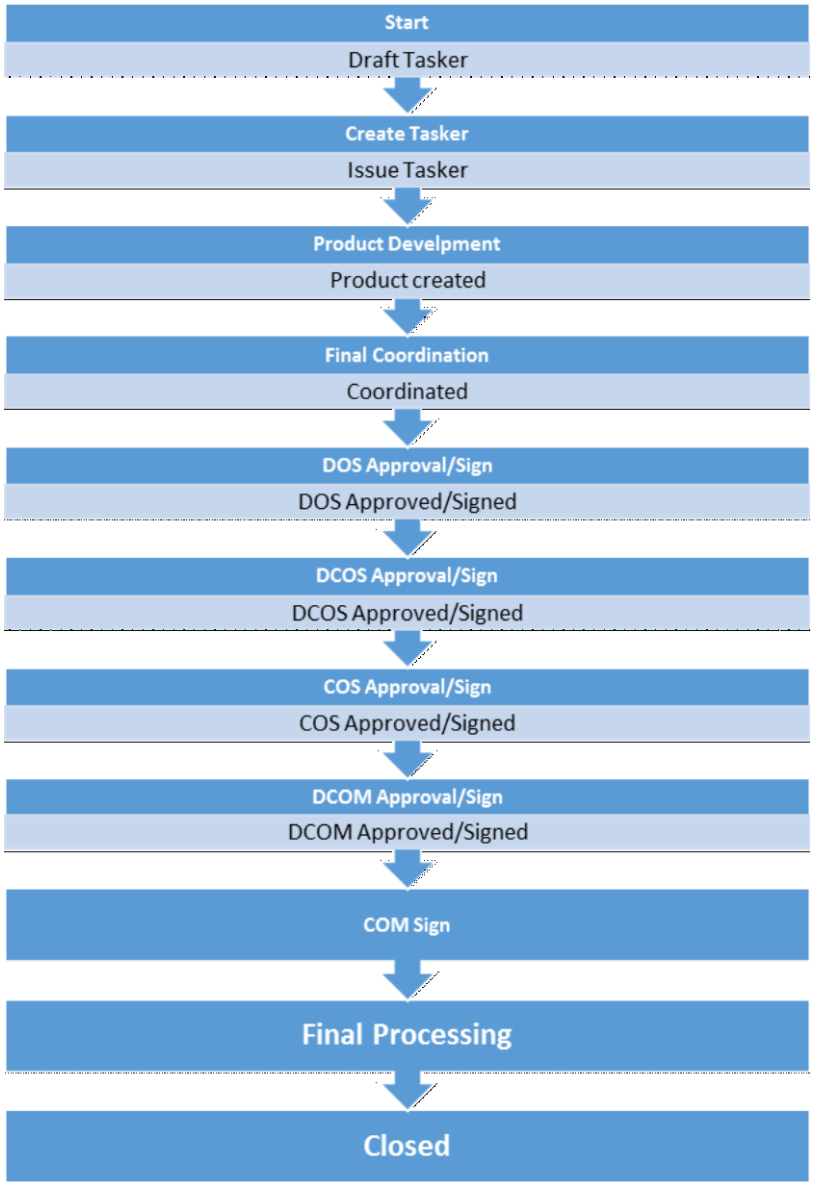


Figure 18 Tasker workflow steps

Requirement ID: TTP-13

A tasker workflow shall be configurable to include more steps, offices and status

Verification Method: Test

3.5.3.1.2.1 Organization structure

Requirement ID: TTP-14

The TT+ organizational structure shall include a set of:

Verification Method: Test

- Roles (grouped by different level of Offices)
- Assigned Actions to each role
- Offices and
- Display Panels (where the offices must be grouped and presented to users in the Tasker form)

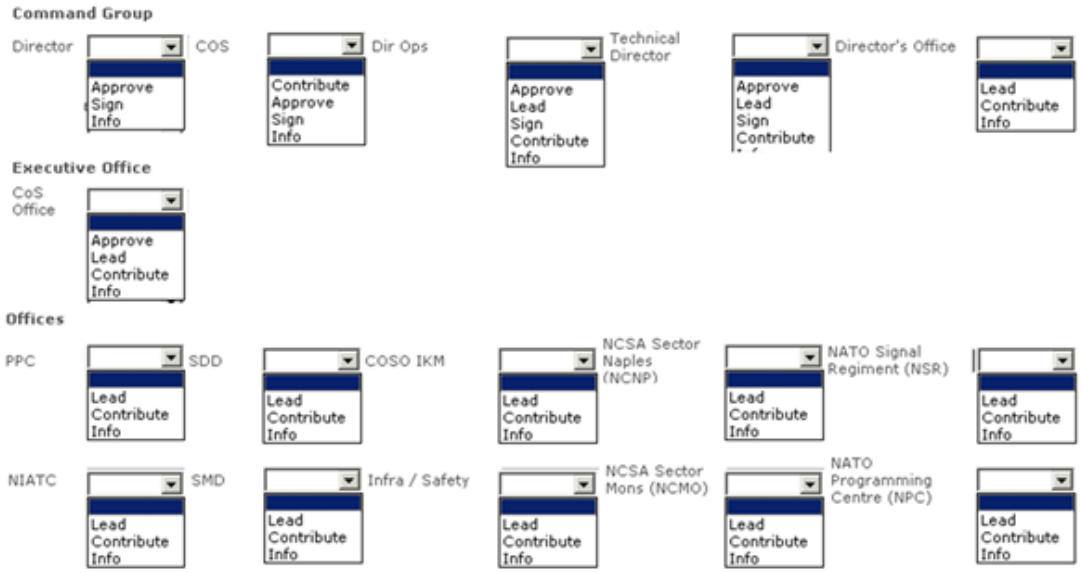


Figure 19 Tasker form panel Office Organization

Requirement ID: TTP-15

The organization structure shall be configurable.

Verification Method: Test

Requirement ID: TTP-16

The Roles shall be associated with a specific level of “Office Roles”, commonly known as “Office Level”, and each role can be associated with one Tasker status and one or more Action Responses

Verification Method: Test

Requirement ID: TTP-17

For display purposes in Tasker Form, panels shall group offices by their “Office Roles” level

Verification Method: Test

Requirement ID: TTP-18

Each office shall be assigned an “Office Roles” level and will be displayed in Tasker Form according to that level. An override mechanism may be used by selecting directly in the Office list the display panel where the Office should appear

Verification Method: Test

Requirement ID: TTP-19

The configuration of all Tasker workflow elements (Offices, Roles etc.) shall comply with their defined relationship, so that Offices have roles associated, roles have status associated and action response defined. See figure below:

Verification Method: Test

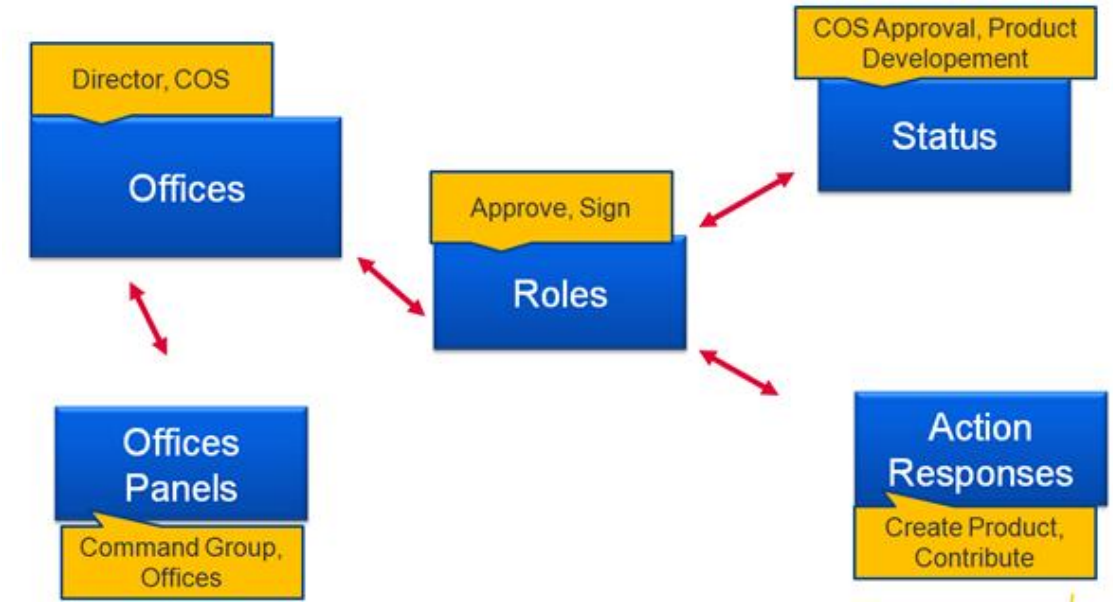


Figure 20 Relationship in Tasker elements

3.5.3.1.3 Search

Requirement ID: TTP-20

The IKM Tools shall leverage existing search services provided by NIP when present and also integrate with the SOA and IdM Platform Information Search Services.

Verification Method: Test

3.5.3.1.4 Functionality

TT+ is an application built on top of SharePoint 2013. The application leverages and extends the capabilities of SharePoint by deploying several TT+ core packages and TT+ extension packages to the SharePoint farm. These packages were initially created at NSHQ in Mons, BE over the past several years (then called the Project Management Tool, “PMT”). For the purposes of the TT+ rollout, these packages and solutions are to be industrialized (read: prepared to work in a standardized NATO environment) and packaged for rollout within the NATO static command structure throughout Europe and

the USA. There are some modifications that have been agreed upon and included in the project scope of work that will customize PMT to function more similarly to the already existing TTE. This new version of PMT – with the industrialization changes and “TTE” customizations is what comprises the “new” application TT+. The packages mentioned above are divided into “solutions”, which the final files are created to be installed on the SharePoint farm. The following sections contains the details of these solutions.

Requirement ID: TTP-24

Integration of External Web Services. TT+ shall be able to integrate with external web services through server or client integration

Verification Method: Test

Requirement ID: TTP-25

Integration of External Applications. TT+ shall allow connection of external applications through the SharePoint API and .NET client integration components

Verification Method: Test

Requirement ID: TTP-26

Set Date/Time Format Used For The Whole Instance. TT+ shall be able to reuse a configurable date and time format on Site Collection level to set/reset the date/time of the whole TT+ instance.

Verification Method: Test

Requirement ID: TTP-27

Infrastructure. TT+ shall be able to be installed as a stand-alone application within a standard ITM version SharePoint farm

Verification Method: Test

Requirement ID: TTP-28

Infrastructure. TT+ shall be compatible with being installed into an existing SharePoint environment as a dedicated Web Application

Verification Method: Test

Requirement ID: TTP-29

Infrastructure. TT+ shall be compatible with being installed into an existing SharePoint environment and shall be able to consume global metadata from an existing term store

Verification Method: Test

Requirement ID: TTP-30

Archive Function. TT+ shall be able to archive a closed activity out of the operational system by converting activity related metadata and documents in to a summary report document stored in a SharePoint based document archive.

Verification Method: Test

Requirement ID: TTP-31

Reporting. TT+ shall be able to generate document based reports on activities and provide a chart feature displaying overview of assigned roles per office.

Verification Method: Test

Requirement ID: TTP-32

Office Inbox. TT+ shall provide an Office Inbox feature to show users the assigned activities to a specific office.

Verification Method: Test

Requirement ID: TTP-33

Activity List. TT+ shall have a central location to display all the currently available activities in a commands TT+ instance and provide users the possibility to filter the displayed list content.

Verification Method: Test

Requirement ID: TTP-34

Error Handling. TT+ shall handle errors and log them in a centralised error log.

Verification Method: Test

Requirement ID: TTP-35

External Integration Component. TT+ shall provide the possibility to interact with external applications using SharePoint OOTB APIs and web service functionality.

Verification Method: Test

Requirement ID: TTP-36

Metadata. TT+ shall be able to use a combination of NCMS and NIP based metadata and extended TT+ required metadata. Metadata fields shall be based on SharePoint columns.

Verification Method: Test

Requirement ID: TTP-37

Search. TT+ shall be able to integrate with a SharePoint 2013 search engine.

Verification Method: Test

Requirement ID: TTP-38

Notification/Mail Functionality. TT+ shall send notification messages to members involved in an activity / when a subtask is assigned to a user / when an office has been assigned a role.

Verification Method: Test

Requirement ID: TTP-88

TT+ shall send notification when an action is completed in a Tasker to the next office in line, notifying that office that their action is now required, excluding "Info" action.

Verification Method: Test

Requirement ID: TTP-39

Security Classification. TT+ shall allow a user to assign a security classification to an activity.

Verification Method: Test

Requirement ID: TTP-40

Content Types. TT+ shall use a set of content types.

Verification Method: Test

Requirement ID: TTP-41

Search Results Security Trimming. TT+ search results shall be security trimmed.

Verification Method: Test

Requirement ID: TTP-42

Metadata. TT+ shall allow users to edit metadata of a draft or issued activity item trough a SharePoint user interface.

Verification Method: Test

Requirement ID: TTP-43

Sub-Tasking. TT+ shall allow a user to be able to delegate activities through creating a subtask within a tasker. The user shall be able to assign subtasks or create a new top-level subtask.

Verification Method: Test

Requirement ID: TTP-44

Edit Action Responses. TT+ shall allow users to select an action response for their related role

Verification Method: Test

Requirement ID: TTP-45

Activity Lifecycle / Change Status. TT+ shall proceed an activity through the activity lifecycle from draft till closed (and archived) if actions linked to the current status are finished. A tasker owner shall be able to manually cancel or suspend an activity.

Verification Method: Test

Requirement ID: TTP-46

Simultaneous Editing of Activity. TT+ shall allow multiple users to work simultaneously on different areas in an activity.

Verification Method: Test

Requirement ID: TTP-47

Documents. TT+ shall provide a SharePoint document library that allows only Tasker's users to upload and contribute to documents belonging to a specific activity

Verification Method: Test

Requirement ID: TTP-48

Reporting. TT+ shall be able to generate a configurable report of an activity.

Verification Method: Test

Requirement ID: TTP-49

Links. TT+ shall provide users the possibility to add links as part of an related to activities

Verification Method: Test

Requirement ID: TTP-50

Delete Activity. TT+ shall provide users the possibility to delete activities and their related workspace

Verification Method: Test

Requirement ID: TTP-51

Tasker History. TT+ shall include a history list as part of an activity to track role assigned roles, action response and workflow processing

Verification Method: Test

Requirement ID: TTP-52

Create Workspace. TT+ shall allow users to create a workspace for an activity type. Allowing the activity creation and approval similarly to the TTE so that there are user roles defined for issuing and approving an activity.

Verification Method: Test

Requirement ID: TTP-53

Draft Activity. TT+ shall allow users to create a draft version of an activity.

Verification Method: Test

Requirement ID: TTP-54

Issue Activity. TT+ shall allow users to issue an activity either directly from the creation form or from a draft activity.

Verification Method: Test

Requirement ID: TTP-55

Permission Control. TT+ shall allow users to assign and revoke permissions for each individual activity. Either by using pre-defined active directory permission groups tied to an assigned office or by manual interaction.

Verification Method: Test

Requirement ID: TTP-56

Instance. TT+ shall have configurable settings throughout a Command's TT+ instance. TT+ shall provide application level configuration: activity configuration, field access, list validation, and activity requests.

Verification Method: Test

Requirement ID: TTP-57

Activity Types. TT+ shall be able to configure different types of activities in order to manage multiple projects types

Verification Method: Test

Requirement ID: TTP-58

Instance of activity type (e.g. Tasker). For each activity type, TT+ shall be capable of configuring its own workflow configuration, activity related roles, offices, default activity access and pre-defined assigned actions.

Verification Method: Test

Requirement ID: TTP-59

Archiving. TT+ shall provide the possibility to configure an archive location as part of a commands instance configuration.

Verification Method: Test

Requirement ID: TTP-60

TT+ shall provide cross site tasking functionality as follows:

Verification Method: Test

- An authorized user in a Command creates an Office representing the remote Command to task (i.e. in ACT Tasker creates SHAPE Office). That office has a "sweeper" staff or representative of the Command.
- The authorized user then Tasks that office in the Tasker
- The sweeper receives the Task and then subtasks as necessary in his Command
- When Task is finished the sweeper updates the Task in the Original Command

Requirement ID: TTP-61

TT+ shall be able to handle Simultaneous editing of taskers

Verification Method: Test

Requirement ID: TTP-62

By default, all taskers' metadata shall be visible to everyone

Verification Method: Test

Requirement ID: TTP-63

TT+ shall provide a user the ability to create a "Close hold" Tasker, so that only member of the Close Hold group can see and modify the Tasker.

Verification Method: Test

Requirement ID: TTP-64

TT+ shall allow a tasker drafter to restrict a new Tasker to only key users of the offices involved.

Verification Method: Test

Requirement ID: TTP-65

TT+ shall provide a configurable Prioritisation indicator on Taskers

Verification Method: Test

Requirement ID: TTP-66

TT+ shall provide a configurable "Office inbox"

Verification Method: Test

Requirement ID: TTP-67

All visual references to PMT shall be replaced by "TT+"

Verification Method: Test

Requirement ID: TTP-68

TT+ shall allow a user with appropriate permissions to create a Tasker from a document stored in EDMS

Verification Method: Test

Requirement ID: TTP-69

TT+ shall allow external services to store the URL of the tasker initiation document. (Create field to foresee previous requirement)

Verification Method: Test

Requirement ID: TTP-70

TT+ shall provide a configurable "My tasks page"

Verification Method: Test

Requirement ID: TTP-71

TT+ shall provide every user the functionality to save Favourite users

Verification Method: Test

Requirement ID: TTP-72

TT+ sub tasking shall leverage Favourite users as follows:

- A user can select 1 or any of their favourite users
- A user can add a member to the list
- A user can remove a member from the list
- A confirmation screen pops up for any added member

Verification Method: Test

Requirement ID: TTP-73

TT+ 'adding a member' shall leverage Favourite users

Verification Method: Test

Requirement ID: TTP-74

TT+ shall provide a centralised Help Centre

Verification Method: Test

Requirement ID: TTP-75

TT+ shall allow a user to send a Tasker Product to the external documents landing zone

Verification Method: Test

Requirement ID: TTP-76

The selected task shall be highlighted on the sub tasks tree

Verification Method: Test

Requirement ID: TTP-77

TT+ shall provide a configurable Business Intelligence (BI) Reporting component

Verification Method: Test

Requirement ID: TTP-78

TT+ shall provide a configurable Time Left indicator on Taskers (red, yellow and green)

Verification Method: Test

Requirement ID: TTP-79

TT+ shall provide a robust 'Mail All Members' functionality for every Tasker

Verification Method: Test

Requirement ID: TTP-80

TT+ shall allow a user to clone another Tasker

Verification Method: Test

Requirement ID: TTP-81

TT+ shall provide a comprehensive history for every Tasker

Verification Method: Test

3.5.3.1.4.1 Core metadata for TT+

The definition of agreed metadata is vital for the finalisation of the TT+ configuration, installation routines, preparation for the NATO IV&V processes and data migration purposes among other areas of importance. The following list contains the expected metadata fields that TT+ will implement, which can be used to tagging purposes. This list has been communicated with the NCIA for several months and this is the current version as of the submission date of this document. Any changes will require further review and might require significant effort on the part of the Contractor. In such cases, an Engineering Change Proposal (ECP) or Change Request (CR) as defined by NATO standard processes could be required.

TT+ Tagging Metadata list			
TT+ Metadata	**Mandatory**	**Internal Name**	**Field Type**
Background	FALSE	PMT_Background	Note
HQ Position	FALSE	PMT_HQPosition	Note

Instructions	FALSE	PMT_Instructions	Note
Funded	FALSE	PMT_Funded	Boolean
Needed By	FALSE	PMT_NeededByeDate	DateTime
Participants	FALSE	PMT_Participants	Note
Recommendations	FALSE	PMT_Recommendations	Note
Scope	FALSE	PMT_Scope	Note
Revision Date	FALSE	PMT_RevisionDate	DateTime
Contribution Due Date	FALSE	PMT_ContributionDueDate	DateTime
Coordination Due Date	FALSE	PMT_CoordinationDueDate	DateTime
Date Signed	FALSE	PMT_DateSigned	DateTime
Due Date	FALSE	PMT_DueDate	DateTime
End Date	FALSE	PMT_EndDate	DateTime
Start Date	FALSE	PMT_StartDate	DateTime
Suspense Date	FALSE	PMT_SuspenseDate	DateTime
Target Completion Date	FALSE	PMT_TargetCompletionDate	DateTime
Views of others	FALSE	PMT_ViewsOfOthers	Note
Priority	FALSE	PMT_Priority	TaxonomyFieldType
Security Classification	TRUE	PMT_SecurityClassification	TaxonomyFieldType
Action Officer	FALSE	PMT_ActionOfficer	User
Initiating Office	FALSE	PMT_InitiatingOffice	Lookup
Country Code	FALSE	PMT_ISO3_CountryCode	Text

Figure 21 TT+ Tagging Metadata List

Requirement ID: TTP-82

Consume External Managed Metadata Server. TT+ shall be able to integrate with an external SharePoint 2013 based managed metadata service.

Verification Method: Test

Requirement ID: TTP-83

Consume External Content Type Hub. TT+ shall be able to reuse content types that are being provided through a SharePoint 2013 based content type hub

Verification Method: Test

3.5.3.1.4.1.1 Dependencies

Requirement ID: TTP-84

The following list catalogues the baseline of TT+. Once installed, these features shall not be changed/updated without going through formal change procedures and including all testing and approval processes. Changing features for the various activities and activity request metadata and content types is very critical and not foreseen. Additionally, the various list definitions for activities and activity requests should not be changed once installed. This is specifically

mentioned as these points can be changed by system administrators and users with elevated access privileges (i.e. “FuncAdmins”).

Verification Method: Test

- The current baseline is based on the core functionalities of Tasker Tracker and PITT[2]:

[2] *Legacy software used before PMT or TTE.*

1. Core metadata for TT+
2. Core TT+ list and views
   - Activity request
   - Activities
   - Office levels
   - Offices
   - Notification Messages
   - Office templates
   - Dependencies
3. Core TT+ pages and web parts
   - Shared Documents
   - Activity Plan
   - Roles
   - Actions
   - Service
4. Core TT+ configuration lists
   - Approval Routing
   - Default activity access
   - Semantics
   - Configuration
   - Activity prefixes
   - Country Codes
5. Configuration sites: every activity type can have its own configuration site:
   - Activity states list
   - Roles list
   - Actions list
   - Office groups list
   - Fixed actions
6. Reporting
   - Document Data links
   - Report templates

7. Archiving: Archiving tool using the core reporting functionalities to archive activities

- Summary report to include core information from the top level activity list, as well as the roles and action list, the project plan list, history and document library overview
- This summary report will be stored together with the activity related document in a folder/document set
- The activity item will be linked to this report/folder, the site will be deleted

Note: Functional metadata i.e. required metadata for an Activity workflow will be part of the TT+ core components. Remaining fields provided by the NATO Core Metadata Standard (“NCMS”) will be integrated into the document metadata used in Document Libraries.

Note: The Archiving Feature can be implemented in several different scenarios, i.e. in a separate WebApp or Site Collection/Content Database. Archiving old and closed Activities allows the administrator to remove non-active Activities from the production area, reducing the impact of storage and system performance and also allows the administrator the option of moving the data to less expensive (i.e. slower) hardware and storage, as these Activities will not be accessed as often.

Requirement ID: TTP-85

Summary report shall include core information from the top level activity list, as well as the roles and action list, the project plan list, history and document library overview

Verification Method: Test

Requirement ID: TTP-86

This summary report shall be stored together with the activity related document in a folder/document set

Verification Method: Test

Requirement ID: TTP-87

The activity item shall be linked to this report/folder, the site will be deleted

Verification Method: Test

4 Non-functional Requirements

The Non-Functional Requirements (NFR) repository categorizes system/software product characteristics which cannot be defined as "functional".

For IKM Tools the Non-functional Requirements repository will contain the following requirements categories:

- Service Criticality (section 4.1)
- System quality (section 4.2)
- Compatibility - Interoperability (Section 4.3)

- Design constraints (section 4.4)
- Technical Documentation Requirements (section 4.5)
- Computer Resource Constraints (section 4.6)

4.1 Service Criticality

The IKM Tools services can be grouped into three categories based upon their level of criticality to the overall effectiveness of the IKM Tools:

- Level 1: IKM Tools cannot be considered to be functional without these services running (Critical)
- Level 2: The IKM Tools can operate without these services, but its effectiveness is degraded (Important)
- Level 3: These services provide value-add, but the IKM Tools can be used effectively without them (Standard)

The following table illustrates which IKM Tools services fall into each category. This categorization is used as a guideline for developing the Non-Functional Requirements and necessary service levels for the IKM Tools.

Service	Critical Service (L1)	Important Service (L2)	Standard Service (L3)
NIP	X		
EDMS	X		
TT+	X		
Workflow Application	X		
Workspace Application		X	
Search	X		
Analytics			X
Cross Domain			X
Distribution and Archiving			X

Table 1 Service Criticality Levels

4.2 System Quality

The System Quality section of Non Functional Requirements (NFR) categorizes system/software **product quality.**

The quality of a system is the degree to which the system satisfies the stated and implied needs of its various stakeholders, and thus provides value. These stated and implied needs are represented by quality models that categorize product quality into characteristics, which in some cases are further subdivided into sub-characteristics. (Some sub-characteristics are divided into sub-sub-characteristics.) This hierarchical decomposition provides a convenient breakdown of product quality.

Characteristic definitions in this section are based on ISO/IEC 25010:2011(E) - System and software quality models.

A tailoring of the standard have been made to assure that the Quality Characteristics necessary for this specific project are correctly selected and defined, in accordance to chapter 3.5 of the Standard.

The quality of a product can be seen in many different way, depending on the stakeholder point of view.

In this case, the stakeholders view to be used is depending on the **Primary and Secondary Users (administrators, maintainers, etc.)** point of view, therefore, in accordance with the ISO 25010 standard the quality model should be tailored on this stakeholder.

The following table shall be considered in reading this section:

Product Quality Characteristic	Influence on Quality (Primary Users)	Influence on Quality (Maintenance Tasks)	Influence on Quality (Other Stakeholders' Concerns)
Performance Efficiency	Yes		Yes
Reliability	Yes		Yes
Availability	Yes	Yes	
Fault Tolerance		Yes	
Recoverability	Yes	Yes	
Portability		Yes	
Adaptability		Yes	
Maintainability		Yes	Yes
Modularity		Yes	
Analysability		Yes	
Testability	Yes	Yes	

Table 2 Quality factors influence

4.2.1 Measuring the quality characteristics

Requirement ID: IKM-SRS-267

For monitoring the quality characteristics of the system some definition of when a system is not correctly performing shall be given.

Verification Method: Analysis

For the IKM Tools the following definition shall be applied:

***Error**: A design flaw or malfunction that causes a Failure of one or more Configuration Items. A mistake made by a person or a faulty Process that affects a CI is also an Error (human Error). For Platforms all the Human error shall not be taken into consideration in measuring the quality performances.*

***Fault:** see Error*

***Failure:** Loss of ability to Operate to Specification, or to deliver the required output. The term Failure may be used when referring to Services, Processes, Activities, Configuration Items, etc.*

***Problem:** A cause of one or more Incidents. The cause is not usually known at the time the incident happens.*

4.2.2 Performance Efficiency

Performance Efficiency is the performance relative to the amount of resources used under stated conditions. Resources can include other software products, the software and hardware configuration of the system, and materials (e.g., print paper, storage media)

Capacity

4.2. *Degree to which the maximum limits of a product or system parameter meet requirements.*

NOTE Parameters can include the number of items that can be stored, the number of concurrent users, the communication bandwidth, throughput of transactions, and size of database.

Requirement ID: IKM-SRS-268

The IKM Tools shall allow to operate to 30,000 users in the Data Center mode (authenticated and anonymous users) and 1,000 users in the Standalone mode

Verification Method: Test

Requirement ID: IKM-SRS-269

The IKM Tools shall allow the concurrent execution of ALL IKM Tools components by 6,000 users in DC mode and meet the performance objectives for these functions.

Verification Method: Test

Requirement ID: IKM-SRS-270

The IKM Tools shall support 20 % concurrency (of total users) considering a 50 request/hour workload

Verification Method: Test

Requirement ID: IKM-SRS-271

The IKM Tools shall allow at least 300 TB of storage capacity for Data Center mode and 2 TB of storage capacity for Standalone mode.

Verification Method: Analysis

Requirement ID: IKM-SRS-272

When the maximum number of allowed concurrent user is using the system, there shall not be any error/fault 99,5 % of the operational time.

Verification Method: Test

Resource Utilization

Degree to which the amounts and types of resources used by a product or system, when performing its functions, meet requirements.

4.2.2.2

Requirement ID: IKM-SRS-273

During normal IKM Tools utilization, the system shall not foresee any memory error/fault for 99% of the operational time.

Verification Method: Test

Requirement ID: IKM-SRS-274

The IKM Tools Search shall perform the re-indexing without any degradation of other IKM Tools.

Verification Method: Test

4.2.2.3

Time Behaviour

Degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements.

Requirement ID: IKM-SRS-275

The IKM Tools functionality: web pages load, button click reaction and general user interaction with the IKM Tools GUI shall be available for an authorised user within 1 second the 95% of the times and within 2 seconds the 100% of the times. The time will be calculated from the instant that the user hits the button to the complete rendering of the page on the client machine (in order to measure the “end-to-end user experience”), with at least 50 items showed in the page (documents, list items, events, etc.)

Verification Method: Test

Requirement ID: IKM-SRS-276

The IKM Tools shall execute the log-in and log out functions within 2 seconds (excluding dependant ADFS services execution time).

Verification Method: Test

Requirement ID: IKM-SRS-277

The IKM Tools web services shall respond to authorised users within 0.1 seconds of submitting the request the 95% of the times and within 0.2 seconds the 100% of the times. The time will be calculated from the instant that the request is arrived to the server to the instant that the servers sends the response.

Verification Method: Test

Requirement ID: IKM-SRS-278

The IKM Tools search services shall respond within 2 seconds of submitting the search query the 95% of the times and within 3 seconds the 100% of the times. The time will be calculated from the instant that the user hits the button to the complete rendering of the page on the client machine (in order to measure the "end-to-end user experience"), with at least 50 search results and at least 20% of the users performing concurrently different searches.

Verification Method: Test

4.2.3 Scalability

Scalability is defined as the capability of a system to increase (or decrease) total throughput under an increased load when resources (typically hardware) are added (or subtracted), so the scalability quality figures are defined accordingly

Requirement ID: IKM-SRS-279

The IKM Tools shall be able to support a throughput increase of 10% every year with a response time degradation of not more than 5%.

Verification Method: Analysis

Requirement ID: IKM-SRS-280

The IKM Tools shall be Horizontal Scalable by allowing the deployment of system components on different instances.

Verification Method: Analysis

Requirement ID: IKM-SRS-281

The IKM Tools shall be Vertical Scalable.

Verification Method: Analysis

4.2.4 Maintainability

Degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers.

Modifications can include corrections, improvements or adaptation of the software to changes in environment, and in requirements and functional specifications. Modifications

include those carried out by specialized support staff, and those carried out by business or operational staff, or end users.

Maintainability includes installation of updates and upgrades.

Maintainability can be interpreted as either an inherent capability of the product or system to facilitate maintenance activities, or the quality in use experienced by the maintainers for the goal of maintaining the product or system.

General

MTTR is the mean time for the system to be repaired after a failure.

MaxTTR is defined as the maximum time required to perform a maintenance action.

4.2.4.1

Service Type	MTTR	MaxTTR
Level 1	2 hours	4 hours
Level 2	4 hours	8 hours
Level 3	7 hours	24 hours

Table 3 Maintainability by Service Level

Requirement ID: IKM-SRS-282

The IKM Tools shall provide a Mean Time To Repair (MTTR) according to the Table 3 Maintainability by Service Level.

Verification Method: Test

Requirement ID: IKM-SRS-283

The MaxTTR for the IKM Tools shall not exceed the times stated in the Table 3 Maintainability by Service Level.

4.2.4.2

Verification Method: Analysis

Modularity

Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components

Requirement ID: IKM-SRS-284

IKM Tools shall be composed of discrete components such that a change to one component has minimal impact on other components, specifically: IKM Tools Framework, NIP, EDMS, TT+, Workspace and Workflow.

Verification Method: Analysis

Requirement ID: IKM-SRS-285

IKM Tools shall allow to update and deploy individual services and components without the need to re-install the full IKM Tools.

Verification Method: Test

Requirement ID: IKM-SRS-285b

IKM Tools shall allow to update and deploy individual services and components independently to each IKM tool, so that the update in one IKM Tool doesn't impact another IKM Tool.

Verification Method: Test

Requirement ID: IKM-SRS-286

The 99.9% of the time a maintenance action is required on a SW component of the IKM Tools, this action shall not cause any possible fault/error in other components of the system.

Verification Method: Analysis

Requirement ID: IKM-SRS-287

IKM Tools shall be modular enough so that the failure of one module or application (due to heavy processing or storage outage) doesn't affect the other modules.

Verification Method: Test

4.2.4.3

Analysability

Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.

Implementation can include providing mechanisms for the product or system to analyse its own faults and provide reports prior to a failure or other event.

Requirement ID: IKM-SRS-288

The system shall be effective and efficient in the possibility to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.

Verification Method: Analysis

Requirement ID: IKM-SRS-289

The system shall be able to detect the 99.5% of the possible fault/error which can occur, triggering the user with a message.

Verification Method: Analysis

Requirement ID: IKM-SRS-290

The system shall be able to diagnose the 95% of the possible fault/error which can occur, triggering the user with a message which identifies the happened failure/error.

Verification Method: Analysis

Requirement ID: IKM-SRS-291

The IKM Tools messages (e.g., error, warning, notification and informational messages) shall contain initiating module information, context sensitive help or directives on where to find answers and solutions.

Verification Method: Analysis

Requirement ID: IKM-SRS-292

The IKM Tools log messages shall contain:

Verification Method: Analysis

- initiating module information
- Date/Time(Z)
- system instance
- (log) message
- category/severity
- user (invoker of function)
- context information (like mission/session, service/function, parameters, and trace-log)

4.2.5 Reliability

4.2.5.1 *Degree to which a system, product or component performs specified functions under specified conditions for a specified period of time.*

General

MTBF (Mean Time Between Failures) is defined as the mean time between two consecutive faults which generate a failure.

MTBCF (Mean Time Between Critical Failures) is defined as the mean time between two consecutive faults which generate critical failures.

A critical failure is a failure which cause the complete system unavailability with consequent loss of the provided service/capability.

Requirement ID: IKM-SRS-293

IKM Tools shall exhibit a Mean-Time-Between-Failure (MTBF) as depicted in the following table:

Verification Method: Analysis

Service Type	MTBF
L1	365 days
L2	180 days
L3	30 days

Requirement ID: IKM-SRS-294

IKM Tools shall exhibit a Mean-Time-Between-Critical-Failures (MTBCF) of more than 730 consecutive days.

Verification Method: Analysis

Availability

4.2.

Degree to which a system, product or component is operational and accessible when required for use.

Here considered only the Inherent Availability (Intrinsic): assumes ideal support (i.e., unlimited spares, no delays, etc.), only design related failures are considered.

4.2.5.2.1 Inherent Availability

Inherent Availability (Intrinsic) assumes ideal support (i.e., unlimited spares, no delays, etc.); only design related *Failures* are considered.

Requirement ID: IKM-SRS-295

The IKM Tools shall have a System Inherent Availability at Data Centres and Standalone modes (static and deployed) as depicted in the table below:

Verification Method: Analysis

4.2.5.3

Service Type	Inherent Availability
L1	99.97 %
L2	99.9 %
L3	99 %

Fault Tolerance

Degree to which a system, product or component operates as intended despite the presence of hardware or software faults.

Requirement ID: IKM-SRS-296

For the 99% of the possible fault/error in one of the system node, the system shall be able to switch to another equivalent node in less than 5 seconds without loss of data.

Verification Method: Analysis

Requirement ID: IKM-SRS-297

For the 99% of the possible fault/error in the IKM Tools access function (availability of web pages), the system shall be able to switch to an alternative web site without loss of data in less than 10 seconds (e.g.: using alternate DNS/IP entry point)

Verification Method: Analysis

Requirement ID: IKM-SRS-298

For the 99% of the possible fault/error in the IKM Tools function (availability of web pages), the IKM Tools shall be able to operate independently from each other so that if one has a downtime it doesn't affect the rest (e.g. when NIP is unavailable, EDMS, TT+, Workspaces and Workflows are still available).

Verification Method: Test

Requirement ID: IKM-SRS-299

The IKM Tools shall notify the user for potential loss/deletion of information objects during modification of any information object (e.g. by cascading deletion). When prompted by a notification about the data that might be lost/deleted, the user shall be able to choose the action that shall be taken by the system (e.g. cancel, continue).

Verification Method: Analysis

Requirement ID: IKM-SRS-300

The IKM Tools shall automatically report errors and suggested corrective actions with respect to the creation, change, exchange and storage of data elements, objects and products.

Verification Method: Analysis

Requirement ID: IKM-SRS-301

The IKM Tools messages (e.g. error, warning, notification, and informative messages) shall contain initiating module information, and contain context sensitive help and directives on where to find answers and solutions. Technical or debugging error scripts are not acceptable (e.g. "Java object 01 not accessible").

Verification Method: Analysis

Requirement ID: IKM-SRS-302

The IKM Tools shall report errors in context (e.g. given within the same page where they are encountered). An error message shall be displayed or provided in a popup. Invalid entries shall be highlighted or marked so they can be quickly identified and corrected.

Verification Method: Analysis

Requirement ID: IKM-SRS-303

The IKM Tools shall display only one error report popup screen for the same error.

Verification Method: Analysis

Requirement ID: IKM-SRS-304

The IKM Tools shall not in any case permit loss of user-entered data due to receipt of an error or other message. User input shall never be lost, discarded or corrupted unless a user actually chooses to delete or reset the input.

Verification Method: Analysis

Requirement ID: IKM-SRS-305

The IKM Tools error message content shall allow the IKM Tools System Administrator to re-create the conditions when the problem occurred (e.g. the appropriate screen and information content that caused the problem).

Verification Method: Analysis

Requirement ID: IKM-SRS-306

The IKM Tools shall be able to queue requests to an unavailable Worklow Service and deliver them when the Service becomes available again. The IKM Tools shall change its Operational State accordingly.

Verification Method: Analysis

4.2.5.4 Recoverability

Degree to which, in the event of an interruption or a failure, a product or system can recover the data directly affected and re-establish the desired state of the system.

Following a failure, a computer system will sometimes be down for a period of time, the length of which is determined by its recoverability.

The RTO (Recovery Time Objective) is the target time set for the recovery of the IKM Tools after a disaster has struck.

The RPO (Recovery Point Objective) is the maximum data loss can be recovered after a disaster has occurred. It is highly related to the backups frequency to restore the system and the latest recorded status.

Requirement ID: IKM-SRS-307

The IKM Tools shall provide the following RTO and RPO metrics:

Verification Method: Analysis

Service Type	Recovery Time Objective (RTO)	Recovery Point Objective (RPO)
L1	10 sec	8 h
L2	4 h	24 h
L3	7 h	48 h

Requirement ID: IKM-SRS-308

The IKM Tools shall queue pending asynchronous ((i.e. do not need immediate feedback) requests to an unavailable Workflow Services (e.g.: User responded action) and Workspace Services (e.g.: workspace document edit change) and deliver them when the Service becomes available again.

Verification Method: Analysis

High latency is defined as latency exceeding 700 milliseconds (SATCOM Round-Trip-Time)

Requirement ID: IKM-SRS-309

The IKM Tools shall resume/retry the IKM Tools services in case of high latency/timeout/loss of network connectivity without loss of data (e.g.: using reliable UDP technology or TPC accelerators to overcome high latency interruptions)

Verification Method: Analysis

Limited bandwidth is defined as bandwidth of less than 12 Mbps.

Requirement ID: IKM-SRS-310

The IKM Tools shall ensure the IKM Tools availability in limited bandwidth context so that users do not experience interruption or degraded IKM Tools Services.

Verification Method: Test

4.2.6 Portability

Portability is the degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another.

Requirement ID: IKM-SRS-311

The IKM Tools shall support the ITM provided virtual platform so that it can be transferred to a different HW without major configuration work.

4.2.6.1

Verification Method: Test

Adaptability

Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.

Requirement ID: IKM-SRS-312

The IKM Tools web capability shall be able to be run at least in the AFPL web browsers and versions: Microsoft Internet Explorer, Mozilla Firefox, Google Chrome and Apple Safari. Especially for the ITM provided web browser.

Verification Method: Test

Requirement ID: IKM-SRS-313

The IKM Tools shall be able to operate in Standalone mode.

Verification Method: Test

Requirement ID: IKM-SRS-314

The IKM Tools shall be able to operate in Data Center mode.

Verification Method: Test

Installability

Degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.

4.2.6.2

Requirement ID: IKM-SRS-315

The IKM Tools shall be successfully installed and/or uninstalled in a specified environment effectively and efficiently.

Verification Method: Analysis

Requirement ID: IKM-SRS-315b

The IKM Tools shall be automatically installed via a Graphical User Interface (GUI).

Verification Method: Analysis

Requirement ID: IKM-SRS-316

The IKM Tools shall recognize automatically the environment in which they are deployed (development, test, Iv&V, production), in a way that the same “release package” (i.e. the same solution build) can be used for all the deployment environments, without the need to recompile the application each time

Verification Method: Test

Requirement ID: IKM-SRS-317

The IKM Tools components and/or patches shall be capable of being installed in an automated way, e.g: by a Windows Installer or similar service/product installation package.

Verification Method: Analysis

Requirement ID: IKM-SRS-318

The IKM Tools shall detect, during installation and uninstallation, if the user has sufficient privileges required for the action. The IKM Tools shall report the details of the access failure to the user before aborting the (un)installation.

Verification Method: Test

Requirement ID: IKM-SRS-319

The IKM Tools shall be equipped with an Installation Guide.

Verification Method: Analysis

Requirement ID: IKM-SRS-320

The IKM Tools Installation Guide shall explain all actions to take in order to install and configure the IKM Tools, including COTS components, and underlying needed services and hardware configurations (i.e. Share Point farm). Every action shall be followed by a description (text and/or screenshots) of the feedback which will be displayed.

Verification Method: Analysis

Requirement ID: IKM-SRS-321

The IKM Tools Installation Guide shall describe:

Verification Method: Analysis

- Prerequisites for installing the IKM Tools. (e.g., the necessary OS access right to be able to install the IKM Tools).
- The necessary software, drivers, etc. to install the IKM Tools.
- How to address integration in the 'environment' (node) - like configuration of monitoring and backup functions.
- The (environment specific) configuration changes necessary on the system and the environment.
- The required disc space.

Requirement ID: IKM-SRS-322

The IKM Tools Installation Guide shall describe how to configure the system backbone to be able to run the IKM Tools.

Verification Method: Analysis

Requirement ID: IKM-SRS-323

The IKM Tools Installation Guide shall describe how to configure the DBMS for the IKM Tools. This shall include both the data model and any access/replication mechanism.

Verification Method: Analysis

Requirement ID: IKM-SRS-324

The IKM Tools Installation Guide shall contain a description of all configuration files. The following points shall be described:

Verification Method: Analysis

- The location of the configuration file
- The content of the configuration file
- The available settings of the items in the configuration file and their meaning
- How to change the configuration file

Requirement ID: IKM-SRS-325

The IKM Tools shall provide a GUI automated capability to completely uninstall the IKM Tools application(s)/component(s).

Verification Method: Test

Requirement ID: IKM-SRS-326

The IKM Tools shall allow, as appropriate, "Complete" and "Stand alone" (un)installation options to perform complete (i.e., all components) and Standalone (i.e., reduced-selected components), respectively.

Verification Method: Analysis

Internationalisation

4.2.6.3

Requirement ID: IKM-SRS-327

The IKM Tools shall change effectively and efficiently to deal with additional international conventions in terms of language, time zones and alike localization functionalities.

Verification Method: Analysis

4.2.6.4

Replaceability

Degree to which a product can replace another specified software product for the same purpose in the same environment.

4.2.7 Usability

Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. Usability can either be specified or measured as a product quality characteristic in terms of its sub characteristics, or specified or measured directly by measures that are a subset of quality in use.

Requirement ID: IKM-SRS-328

The content and information within the system shall be presented to the user in a consistent, standardized manner.

Verification Method: Test

Requirement ID: IKM-SRS-329

Every input by a user shall consistently produce some perceptible response output from the computer.

Verification Method: Test

Requirement ID: IKM-SRS-330

Only data essential to the user's needs shall be displayed.

Verification Method: Inspection

Requirement ID: IKM-SRS-331

Given a simple Test Suite relative to the major functionalities of the IKM Tools (e.g. a search, the upload of a document, the completion of a workflow task), after an introductory training of no more than 1 hour, the 95% of a population of at least 50 users (randomly selected from the population of the real users that has never been exposed to the new KM Tools developed by the Contractor) will be able to complete the Test Suite in less than 5 minutes

Verification Method: Test

Requirement ID: IKM-SRS-332

The dedicated user roles for the IKM Tools shall adhere to the minimum user rights needed to operate the tools, so that no more than needed rights are assigned to a particular user role (i.e. : End User doesn't need to have System Administration permissions to simply operate the IKM Tools basic functions).

Verification Method: Test

4.2.8 Security

Security is defined as the capability of the software product to protect information and data so that unauthorised persons or systems cannot read or modify them and such that authorised persons or systems are not denied access to them. As well as data stored in or by a product or system, security also applies to data in transmission.

For purposes of this document, the following definitions are used:

Confidentiality: the property that information is not made available or disclosed to unauthorised individuals or entities.

Integrity: the property that information (including data, such as cipher text) has not been altered or destroyed in an unauthorised manner.

Non-repudiation: the measure of assurance to the recipient that shows that information was sent by a particular person or organisation and to the sender that shows that information has been received by the intended recipients.

Accountability: the degree to which actions of an entity can be traced uniquely to the entity.

Authenticity: the degree to which the identity of a subject or resource can be proved to be the one claimed.

The following CIS Security functionalities will be provided by the BI-SC AIS:

- **Confidentiality**. Military-grade NATO IP cryptographic equipment (NICE) will provide confidentiality to User data as well as cryptographic separation between security Domains (for example, NATO SECRET, NATO UNCLASSIFIED. MISSION SECRET). Information exchange between these security domains will be achieved through appropriate boundary protection services (BPS). As a minimum, NICE will be located at each boundary between the local area networks (LANs) and the NATO wide area network (WAN). This will ensure that all User data will be encrypted prior to transmission across the NATO WAN. Software application layer mechanisms will be used for Community-of-Interest (COI) separation.

- **Integrity**. Digital signatures and authentication services will be used by various protocols (e.g., SNMP, IPSEC) to provide integrity and strong authentication to User data and network configurations. The NATO Public Key Infrastructure (NPKI) will enable these specific security services.

Infrastructure security as provided by the Bi-SC AIS Infrastructure will be transparent to the IKM Tools.

Requirement ID: IKM-SRS-333

The IKM Tools shall be compliant with NATO document [C-M(2002)49] for the protection of NATO classified information, supporting systems services and resources in CIS, or other storing devices, processing and transmitting systems.

Verification Method: Analysis

Requirement ID: IKM-SRS-334

The IKM Tools components shall be configured with the latest security patches and updated with the latest security guidelines from the NATO Information Assurance Technical Centre (NIATC). This shall be enforced on all environments (e.g. operational environment, training environment, reference systems environment), while applying the appropriate CM process and procedures.

Verification Method: Analysis

Requirement ID: IKM-SRS-335

The IKM Tools shall be capable of operating within the NS and MS WAN environment (including servers, network, services and workstations) in the presence of the currently approved NATO Security Settings (target version to be provided by the Purchaser during the Design Stage). Any deviations from the approved security settings shall be identified by the Contractor prior to testing and shall be subject to approval of the Purchaser.

Verification Method: Test

Requirement ID: IKM-SRS-336

The IKM Tools shall adhere to the "CIS Security Technical and Implementation Directive for the Security of Web Applications" (NHQC3S(C3CAM)0014-2019 (INV), 26 Sept. 2019) for securing the web services and applications.

Verification Method: Analysis

Confidentiality

Requirement ID: IKM-SRS-337

4.2.8.1 The IKM Tools shall ensure that a confidentiality label (policy, classification, releasability) is automatically included into each the IKM Tools information <element or product>, showing the highest classification of information it contains.

Verification Method: Analysis

Requirement ID: IKM-SRS-338

In line with C-M(2002)49-COR12 , the Security Classification in the IKM Tools shall include:

Verification Method: Analysis

- Policy Identifier/ Information Ownership: e.g. NATO, NATO/EAPC (Euro-Atlantic Partnership Council), ISAF (International Security Assistance Force)
- Classification Marking: e.g. Unclassified, Restricted, Confidential, Secret
- Category/Caveats: e.g. Releasable to AUS/FIN, Releasable to RS, Releasable to Coalition

Requirement ID: IKM-SRS-339

The machine readable structure of the Security Label in the IKM Tools shall be in accordance with INFOSEC Technical & Implementation Guidance for Electronic Labelling of NATO Information [AC/322-D (2004)0021].

Verification Method: Analysis

Requirement ID: IKM-SRS-340

The IKM Tools shall provide visual confirmation to Users (on-screen) of the security classification including any releasability caveats (e.g., Releasable to RS) of the displayed data.

Verification Method: Inspection

Requirement ID: IKM-SRS-341

The IKM Tools shall include a configurable colour-based visual cue in addition to text to indicate security classification in screens, reports and prints.

Verification Method: Test

Requirement ID: IKM-SRS-342

The IKM Tools shall insert a Security Classification construct into headers/footers and metadata of generated, created or exported reports, MS Office files and PDF files. The user shall be prompted to be able to change the security classification.

Verification Method: Inspection

Requirement ID: IKM-SRS-343

The IKM Tools shall propose the highest classification level of the selected objects. If no classification is specified for the selected objects, then the repository classification level shall be proposed.

Verification Method: Test

Requirement ID: IKM-SRS-344

The IKM Tools shall allow the authorised user to override the proposed classification level by choosing another. Classification is mandatory and shall be composed of three fields: authority, classification, and releasability.

Verification Method: Test

Integrity

4.2.8.2

Requirement ID: IKM-SRS-345

The IKM Tools shall maintain referential integrity between entities across all services avoiding orphan entities (e.g.: deletion of a workflow results in deletion of all related entities: sub activities, sub-workspaces, documents and alike).

Verification Method: Analysis

Requirement ID: IKM-SRS-346

The IKM Tools shall implement a two-phase deletion process (i.e., a logical/soft delete with User-controlled permanent deletion/purging) for entities.

4.2.8.3

Verification Method: Analysis

Authenticity

4.2.8.3.1 General

Definitions:

User: refers to a person having access to the operating system (an OS User) and IKM Tools Services. Each User of the IKM Tools is assigned Access Rights based on its Role, the Permissions within that Role, and optionally the Organization of the User.

Role: Defined by a set of permissions (i.e., access to objects and functionality) to perform certain operations.

Basic Role: One of [NUMBER] primary roles in IKM Tools (i.e., User, Organisational Node Administrator, Enterprise Administrator, System Administrator).

The primary Basic Roles in the IKM Tools are:

- **The IKM Tools User:** a person having Access Rights for the IKM Tools User Functionality. This functionality includes viewing, creating, collaborating and maintaining [INFORMATION_OBJECTS].

- **The IKM Tools Organizational Node Administrator**: a person having Access Rights for the IKM Tools Organizational Node Administrator Functionality. This functionality includes managing the IKM Tools Accounts, Access Rights, defining the Application or Information Portal Structure, and defining Information Exchange Contracts.

- **The IKM Tools Enterprise Administrator:** a person having Access Rights for the IKM Tools Enterprise Administrator Functionality. This functionality includes maintaining the enterprise-wide configuration (e.g., domain values).

- **The IKM Tools System Administrator:** a person having Access Rights for the IKM Tools System Administrator Functionality. This functionality includes the functionality for the IKM Tools System Administration and the IKM Tools System Maintenance. The IKM Tools System Administration Functionality includes deploying, configuring and updating the IKM Tools.

List of basic roles:

Organizational Node Administrators are generally members of the staff responsible for User management, domain value management and system configuration for that particular the IKM Tools organizational node.

Organizational Node Administrators are also responsible for adapting and localising production workflow sequences to guide and control processes.

Organizational Node Administrators are able to assign User permissions on types of information objects (e.g., Overlay) and functions (e.g., Read, Create, Modify, Delete) on those objects for that particular organizational node. To simplify administration, a role may be specified from more basic roles and permission sets.

Organizational Node Administrators will have the capability to perform content management functions, including data cleansing and archiving.

Enterprise Administrators are responsible for overall management and administration of the system, including both technical and procedural aspects. In general, Enterprise Administrators are identified for each mission/domain.

Procedural and administrative responsibilities of the Enterprise Administrators include the creation, documentation and enforcement of operating policies and procedures associated with functional system configuration; domain management; User access and privilege management; data stewardship; workflow management; and identification and resolution of functional issues.

Enterprise Administrators are responsible for overseeing development and maintenance of Standard Operating Procedures (SOPs) and coordination with Organizational Node Administrators.

The technical responsibilities of Enterprise Administrators include enterprise domain management; collection of performance and accounting data; and ensuring security mechanisms are working.

Enterprise Administrators are also responsible for identifying standard production workflow sequences.

System Administrators are generally NCI Agency or other local CIS support personnel responsible for system and network technical issues, and for ensuring the proper configuration, network connectivity and recoverability of the system. Responsibilities of the System Administrators include network and domain management; back-up and recovery of file systems and databases; and administration of the IKM Tools applications and servers. System administrators are responsible for maintaining Windows User groups and adding new Users to the Windows domain, and (re)installing the system as required.

Requirement ID: IKM-SRS-347

The IKM Tools shall support access (if authorised) to all system functionalities up to the maximum number of Users identified for that site category (e.g., there shall be no licensing restrictions on the number of Users who can simultaneously access a particular functionality).

Verification Method: Analysis

4.2.8.3.2 Authentication Processing

Requirement ID: IKM-SRS-348

The IKM Tools shall uniquely Identify and Authenticate Users.

Verification Method: Analysis

Requirement ID: IKM-SRS-349

The IKM Tools shall allow an authorised user (i.e. The IKM Tools Admin) to manage (create, update, delete) the IKM Tools User Accounts, password details, and assign User Roles to User Account and manage general access privileges of individual User Accounts.

Verification Method: Analysis

Requirement ID: IKM-SRS-350

The IKM Tools shall apply password policy [AC/322-D/0048-REV3]. .

Verification Method: Analysis

The password policy will enforce individuals to select a password that is at least 9 characters long, comprising uppercase, lowercase and symbols. The password must be changed every 6 months or if the user suspects it has been compromised (for example, if the user thinks anyone has observed entering the password).

Requirement ID: IKM-SRS-351

The IKM Tools shall deny the re-use of [NUMBER] previous passwords.

Verification Method: Analysis

Requirement ID: IKM-SRS-352

The IKM Tools user accounts shall be locked after defined unsuccessful authentication attempts.

Verification Method: Analysis

Requirement ID: IKM-SRS-353

The IKM Tools passwords shall be stored in encrypted form.

Verification Method: Analysis

Requirement ID: IKM-SRS-354

The IKM Tools shall protect User credentials in transit.

Verification Method: Analysis

Requirement ID: IKM-SRS-355

The IKM Tools shall protect session IDs.

Verification Method: Analysis

Requirement ID: IKM-SRS-356

The IKM Tools session IDs shall never be included in any URL or sent in the referrer header to prevent caching by the browser. Session IDs shall be long, complicated random numbers which cannot be easily guessed.

Verification Method: Analysis

Requirement ID: IKM-SRS-357

The IKM Tools shall employ encryption of the entire login transaction using SSL or similar technologies. This requirements shall only be applicable for the option when no domain authentication is used.

Verification Method: Analysis

Requirement ID: IKM-SRS-358

The IKM Tools shall allow the system administrator to set a "timeout" period which shall automatically log-out any sessions which have been inactive for that period of time. The system administrator shall be able to disable this feature.

Verification Method: Analysis

Requirement ID: IKM-SRS-359

The IKM Tools shall allow the system administrator to prevent multiple concurrent authentications from the same user from different locations (IP addresses). The system administrator shall be able to disable this feature.

Verification Method: Analysis

Requirement ID: IKM-SRS-360

The IKM Tools shall protect the User's entire session via SSL to ensure that the session ID (e.g., session cookie) cannot be read off the network.

Verification Method: Analysis

Requirement ID: IKM-SRS-361

The IKM Tools shall allow the User (with the same User-id) to access the same information and functionality from any workstation on the NS WAN (i.e., 'roving User' functionality). This capability shall not depend on the availability of Active Directory.

Verification Method: Analysis

Requirement ID: IKM-SRS-362

Role-based access control shall be applied according to the following guidelines:

Verification Method: Analysis

- Users are associated with User Roles and also with Organizations.
- User Roles determine the functions and types of objects available to the User.
- Organizations determine the data available for use by the available functions.
- A User has permission on a particular data item only if the User has an authorised Role and is a member of that Organization.

Requirement ID: IKM-SRS-363

If an authorised the IKM Tools User is a member of more than one Organization (i.e., Organizational Node), the User shall be prompted to select the Organization to be used during that session.

Verification Method: Analysis

Requirement ID: IKM-SRS-364

The IKM Tools shall allow authenticated Users to manage their password and their User profile (e.g., e-mail address, unit) information.

Verification Method: Analysis

Requirement ID: IKM-SRS-365

The IKM Tools shall provide help texts to support the login process together with links to recover lost password and login details.

Verification Method: Analysis

Requirement ID: IKM-SRS-366

The IKM Tools shall limit the feedback of information during authentication to prevent Users gaining knowledge of the authentication process.

Verification Method: Analysis

Audit and Accountability

Requirement ID: IKM-SRS-367

4.2.8.4.

The IKM Tools shall generate audit records for auditable events, addressing at a minimum the following events:

Verification Method: Test

- system start-up (including re-starts) and shutdown
- log-on (including log-on attempts) and log-off of individual users
- changes to permissions and privileges of users and groups
- changes to security relevant system management information(including audit functions)
- start-up and shutdown of the audit function
- any access to security data
- deletion, creation or alteration of the security audit records;#
- changes to system date and time
- unsuccessful attempts to access system resources

Requirement ID: IKM-SRS-368

The IKM Tools shall establish access permissions to audit information.

Verification Method: Test

Requirement ID: IKM-SRS-369

The IKM Tools shall associate individual user identities to auditable events.

Verification Method: Test

Requirement ID: IKM-SRS-370

The IKM Tools shall action on failed attempts at log-on.

Verification Method: Test

Requirement ID: IKM-SRS-371

The IKM Tools shall create and maintain an archive of audit information.

Verification Method: Test

Requirement ID: IKM-SRS-372

The IKM Tools shall retain audit information for a configurable period of time by IKM Administrator.

Verification Method: Test

Requirement ID: IKM-SRS-373

The IKM Tools shall ensure log files are manageable so that the size doesn't affect the performance or functionality. (i.e. limiting the size to 10 Mb).

Verification Method: Analysis

4.2.8.4.1 User Audit Log

Requirement ID: IKM-SRS-374

The IKM Tools shall record in traceable logs all selected transactions, database activities, technical events (e.g., dataset synchronisation, directory replication) and accessing of data.

Verification Method: Test

Requirement ID: IKM-SRS-375

If so configured, the IKM Tools shall ensure operations at the business object level are recorded in traceable logs.

Verification Method: Test

Requirement ID: IKM-SRS-376

If so configured, the IKM Tools shall ensure User operations at system function level are recorded in traceable logs.

Verification Method: Test

Requirement ID: IKM-SRS-377

If so configured, the IKM Tools shall support audit trailing to all User and the IKM Tools system actions and messages on sending, deleting and viewing to log all User activities.

Verification Method: Test

Requirement ID: IKM-SRS-378

If so configured, the IKM Tools shall log all configurations changes with the trace to persons or systems.

Verification Method: Test

4.2.8.4.2 System Audit Log

Requirement ID: IKM-SRS-379

The IKM Tools shall generate and maintain an Audit Log for each of the following auditable events, shall associate individual User identities to those events, and shall include date and time of the event, type of event, User identity, and the outcome (success or failure) of the event:

Verification Method: Test

- System start-up and shutdown
- the start/end time of usage of system applications (system components) by individual Users
- Changes to permissions and privileges of Users and groups
- Changes to security relevant system management function
- Configuration changes
- Any access to audit log
- Deletion, creation or alteration of the security audit records
- All privileged operations
- All updates of the IKM Tools access rights
- All attempts to delete, write or append the Audit files

4.2.8.5 Web Security

4.2.8.5.1 Authentication

Requirement ID: IKM-SRS-380

Authentication shall be through an authorized identity provider. Where appropriate, other authentication requirements shall be enforced by the authorized identity provider.

Verification Method: Analysis

Requirement ID: IKM-SRS-381

Identity Information related to authentication (such as credentials) managed by the website shall not traverse public networks unencrypted.

Verification Method: Analysis

Requirement ID: IKM-SRS-382

Forgot password functionality and other recovery paths shall do not send the existing or new passwords in clear text to the user.

Verification Method: Analysis

Requirement ID: IKM-SRS-383

No default passwords shall be used, for the website or any components.

Verification Method: Analysis

Requirement ID: IKM-SRS-384

Passwords shall never be hard-coded in any source code. Not even in an encrypted/hashed form.

Verification Method: Analysis

Requirement ID: IKM-SRS-385

All authentication controls shall be enforced on the server side.

Verification Method: Analysis

Requirement ID: IKM-SRS-386

Account passwords shall be stored encrypted or hashed in such a way it should not be possible to identify identical passwords.

Verification Method: Analysis

Requirement ID: IKM-SRS-387

Users shall be able to safely change their credentials using a mechanism that is at least as resistant to attack as the primary authentication mechanism.

Verification Method: Analysis

Requirement ID: IKM-SRS-388

Authentication credentials shall be configured to expire after an administratively configurable period.

Verification Method: Test

Requirement ID: IKM-SRS-389

All authentication decisions shall be logged, including increased delays between successive unsuccessful logging attempts (linear back-offs) and temporal account locks (soft-locks).

Verification Method: Test

Requirement ID: IKM-SRS-390

Forgotten password and other recovery paths shall send a time-limited activation token or use two factor proofs

Verification Method: Test

Requirement ID: IKM-SRS-391

The IKM Tools implementation shall be compatible with FMN Spiral 4 Service Instructions for Web Authentication.

Verification Method: Analysis

4.2.8.5.2 Session Management

Requirement ID: IKM-SRS-392

A session management mechanism shall be used after a successful authentication, and the related security context shall be maintained until the session expires. Any change in the security context shall require re-authentication.

Verification Method: Test

Requirement ID: IKM-SRS-393

Sessions shall be invalidated when the user logs out.

Verification Method: Test

Requirement ID: IKM-SRS-394

Sessions shall timeout after a specified period of inactivity.

Verification Method: Test

Requirement ID: IKM-SRS-395

Only non-persistent cookies shall be used for session management purposes, so that the session ID does not remain on the web client cache for long periods of time.

Verification Method: Test

Requirement ID: IKM-SRS-396

Session ID shall be changed or cleared on logout or when the session expires.

Verification Method: Test

Requirement ID: IKM-SRS-397

At least one mechanism shall be used to prevent cookie theft and session hijacking (e.g. HttpOnly and Secure attributes or usage of TLS during the entire session)

Verification Method: Inspection

Requirement ID: IKM-SRS-398

The Session ID shall be changed on re-authentication.

Verification Method: Test

Requirement ID: IKM-SRS-399

Session IDs shall never be cached, applications must use restrictive cache directives for all the web traffic exchanged through HTTP and HTTPS, such as the "Cache-Control: no-cache, no-store" and "Pragma: no-cache" HTTP headers, and/or equivalent META tags on all or (at least) sensitive web pages.

Verification Method: Test

Requirement ID: IKM-SRS-400

It shall not be possible to determine a session ID knowing the previously generated ID(s). For that reason, session IDs shall be generated using a cryptographically secure (pseudo)random number generator and they shall be at least 128 bits long.

Verification Method: Test

Requirement ID: IKM-SRS-401

Only Session IDs generated by the application framework shall be recognized as valid by the application, unless for a business requirement such as single sign on.

Verification Method: Test

Requirement ID: IKM-SRS-402

Session IDs using cookies shall have their path set to an appropriately restrictive value for that site. The domain cookie attribute restriction should not be set unless for a business requirement, such as single sign on.

Verification Method: Test

Requirement ID: IKM-SRS-403

The application shall not permit duplicate concurrent user sessions, originating from different machines.

Verification Method: Test

4.2.8.5.3 Access Control

Requirement ID: IKM-SRS-404

Access controls shall be enforced on the server side.

Verification Method: Analysis

Requirement ID: IKM-SRS-405

User and data attributes and policy information used by access controls shall not be manipulated by end users unless specifically authorized.

Verification Method: Analysis

4.2.8.5.4 Input validation

Requirement ID: IKM-SRS-406

The runtime environment shall not susceptible to buffer overflows, or there shall be security controls preventing buffer overflows.

Verification Method: Test

Requirement ID: IKM-SRS-407

The runtime environment shall not susceptible to SQL Injection, or there shall be security controls preventing SQL Injection.

Verification Method: Test

Requirement ID: IKM-SRS-408

The runtime environment shall not susceptible to Cross Site Scripting (XSS), or there shall be security controls preventing XSS Injection.

Verification Method: Test

Requirement ID: IKM-SRS-409

The runtime environment shall not susceptible to LDAP Injection, or there shall be security controls preventing LDAP Injection.

Verification Method: Test

Requirement ID: IKM-SRS-410

The runtime environment shall not susceptible to OS Command Injection, or there shall be security controls preventing OS Command Injection.

Verification Method: Test

Requirement ID: IKM-SRS-411

The runtime environment or parser shall not be susceptible to XML and XPath injection or there shall be security controls preventing XML and XPath injection.

Verification Method: Test

Requirement ID: IKM-SRS-412

All input validation failures shall result in input rejection or input sanitization in accordance with the NCIRC guidance.

Verification Method: Analysis

Requirement ID: IKM-SRS-413

Input validation or encoding routines shall be performed and enforced on the server side in accordance with the NCIRC guidance.

Verification Method: Analysis

Requirement ID: IKM-SRS-414

If input validation controls are enforced by the presentation layer on the client side (e.g. size or format constraints, input type, etc.) the same controls shall be enforced on the server side.

Verification Method: Analysis

Requirement ID: IKM-SRS-415

Parameters shall be canonicalized, input validated, and output encoded to prevent both local and remote file inclusion attacks, particularly where input could be executed, such as header, source, or template inclusion. Parameters shall never be used to manipulate filenames, pathnames or any file system object without first being canonicalized and input validated to prevent local file inclusion attacks.

Verification Method: Analysis

4.2.8.5.5 Cryptography at rest

Requirement ID: IKM-SRS-416

All cryptographic functions shall be implemented on the server side unless purposely designed in another manner.

Verification Method: Analysis

Requirement ID: IKM-SRS-417

Data-at-rest decryption keys shall be protected from unauthorized access.

Verification Method: Analysis

Requirement ID: IKM-SRS-418

Cryptographic keys shall be managed (e.g., generated, distributed, revoked, expired) using approved NATO policies [AC/322-D/0047, 2009].

Verification Method: Analysis

Requirement ID: IKM-SRS-419

Cryptographic algorithms used by the application shall be selected from the NATO Type B algorithm suite.

Verification Method: Analysis

4.2.8.5.6 Error handling and Logging

Requirement ID: IKM-SRS-420

The application shall not output error messages or stack traces containing sensitive data that could assist an attacker, including Session ID and personal information.

Verification Method: Test

Requirement ID: IKM-SRS-421

Logs events shall include at a minimum:

Verification Method: Test

- Time stamp from a reliable source
- Severity level of the event
- An indication that this is a security relevant event (if mixed with other logs)
- The identity of the user that caused the event (if there is a user associated with the event)
- The source IP address of the request associated with the event
- Status of the event (e.g. succeeded or failed)
- A description of the event

Requirement ID: IKM-SRS-422

Logging controls shall be implemented on the server.

Verification Method: Analysis

Requirement ID: IKM-SRS-423

Security logging controls shall have the ability to log both success and failure events that are identified as security-relevant.

Verification Method: Test

Requirement ID: IKM-SRS-424

Security logs shall be protected from unauthorized access and modification.

Verification Method: Analysis

Requirement ID: IKM-SRS-425

A log analysis tool shall be available to allow the analyst to search for log events based on combinations of search criteria across all fields in the log record format supported by this system.

Verification Method: Test

4.2.8.5.7 Data Protection

Requirement ID: IKM-SRS-426

Forms containing sensitive information shall have disabled client side caching, including autocomplete features.

Verification Method: Test

Requirement ID: IKM-SRS-427

Sensitive data shall be sent to the server in the HTTP message body (i.e., URL parameters are never used to send sensitive data).

Verification Method: Test

Requirement ID: IKM-SRS-428

Cached or temporary copies of sensitive data sent to the client or stored in the server shall be protected from unauthorized access or purged/invalidated after the authorized user accesses the sensitive data (e.g., the proper no-cache and no-store Cache-Control headers are set).

Verification Method: Test

Requirement ID: IKM-SRS-429

The list of sensitive data processed by the site shall be identified, and that there shall be an explicit policy for how access to this data must be controlled, and when this data must be encrypted (both at rest and in transit).

Verification Method: Test

Requirement ID: IKM-SRS-430

There shall be a method to remove each type of sensitive data from the application at the end of its required retention period.

Verification Method: Test

Requirement ID: IKM-SRS-431

The integrity of interpreted code, libraries, executables, and configuration files shall be verified using checksums or hashes.

Verification Method: Test

Requirement ID: IKM-SRS-432

Sensitive data shall be rapidly sanitized from memory as soon as it is no longer needed.

Verification Method: Test

Requirement ID: IKM-SRS-433

The application should have the ability to detect and alert on abnormal numbers of requests to prevent screen scraping, automated use of web service extraction or data loss prevention.

Verification Method: Test

4.2.8.5.8 Communications Security

Requirement ID: IKM-SRS-434

IKM Tools certificates shall have a path built from a trusted CA to each Transport Layer Security (TLS) server certificate, and each server certificate shall match the Fully Qualified Domain Name of the server and be valid.

Verification Method: Inspection

Requirement ID: IKM-SRS-435

TLS shall be used for all connections, internal (e.g. backend) or external, that involve sensitive data or functions.

Verification Method: Test

Requirement ID: IKM-SRS-436

Backend TLS connection failures shall be logged.

Verification Method: Test

Requirement ID: IKM-SRS-437

Connections to external systems that involve sensitive information or functions shall be authenticated.

Verification Method: Test

Requirement ID: IKM-SRS-438

Connections to/from external systems shall use accounts configured to have the minimum privileges necessary for the application to function properly.

Verification Method: Test

Requirement ID: IKM-SRS-439

Failed TLS connections shall not fall back to an insecure connection.

Verification Method: Test

Requirement ID: IKM-SRS-440

Certificate paths shall be built for all client certificates using configured trust anchors and revocation information.

Verification Method: Test

Requirement ID: IKM-SRS-441

The application shall use a single standard TLS implementation that is configured to operate in an approved mode of operation

Verification Method: Test

Requirement ID: IKM-SRS-442

Specific character encodings shall be defined for all connections (e.g., UTF-8).

Verification Method: Test

4.2.8.5.9 HTTP Security

Requirement ID: IKM-SRS-443

The application shall accept only a defined set of HTTP request methods, such as GET and POST, unused methods shall be explicitly blocked.

Verification Method: Test

Requirement ID: IKM-SRS-444

Every HTTP response shall contain a single content type header specifying a safe character set (e.g., UTF-8).

Verification Method: Test

Requirement ID: IKM-SRS-445

HTTP headers, in both requests and responses, and URIs shall contain only printable ASCII characters.

Verification Method: Test

Requirement ID: IKM-SRS-446

Web sites shall never switch a given session from HTTP to HTTPS, or vice versa, as this can disclose the session ID in the clear through the network.

Verification Method: Test

Requirement ID: IKM-SRS-447

If HTTPS is required, the web application shall make use of "HTTP Strict Transport Security (HSTS)" (previously called STS) to enforce HTTPS connections.

Verification Method: Test

Requirement ID: IKM-SRS-448

When feasible, web applications should not mix encrypted and unencrypted contents (HTML pages, images, CSS, JavaScript files, etc.) on the same host (or even domain - see the "domain" cookie attribute), as the request of any web object over an unencrypted channel might disclose the session ID.

Verification Method: Test

Requirement ID: IKM-SRS-449

When feasible, web applications, should not offer public unencrypted contents and private encrypted contents from the same host. It is recommended to instead use two different hosts, such as www.example.com over HTTP (unencrypted) for the public contents, and secure.example.com over HTTPS (encrypted) for the private and sensitive contents. The former host only has port TCP/80 open, while the later only has port TCP/443 open

Verification Method: Test

4.2.8.5.10 Files and resources

Requirement ID: IKM-SRS-450

Files, other than static pages and dynamic content (CGI scripts), shall be stored outside the Webroot.

Verification Method: Analysis

Requirement ID: IKM-SRS-451

The web or application server shall be configured by default to deny access to remote resources or systems outside the web or application server.

Verification Method: Analysis

Requirement ID: IKM-SRS-452

The application code shall not execute uploaded data.

Verification Method: Analysis

Requirement ID: IKM-SRS-453

Rich internet applications cross domain resource sharing configuration shall be configured to prevent unauthenticated or unauthorized remote access.

Verification Method: Analysis

Requirement ID: IKM-SRS-454

Remote IFRAMEs and HTML 5 cross-domain resource sharing shall not allow inclusion of arbitrary remote content.

Verification Method: Analysis

4.2.8.5.11 Miscellaneous (HTML5/JavaScript/ActiveX)

Requirement ID: IKM-SRS-455

Sensitive data shall not be stored in storage area of a thick client, including to system or application logs.

Verification Method: Analysis

Requirement ID: IKM-SRS-456

3rd-party JavaScript libraries or thick-client shall be hosted within the web application/web server and not hot-linked from external untrusted sources

Verification Method: Analysis

Requirement ID: IKM-SRS-457

Thick clients shall not request more permissions or access to resources than those strictly required for its correct operation

Verification Method: Analysis

Requirement ID: IKM-SRS-458

3rd-party JavaScript libraries in use shall be up to date and contain no known vulnerabilities.

Verification Method: Analysis

Requirement ID: IKM-SRS-459

Thick-client code shall be signed.

Verification Method: Test

Requirement ID: IKM-SRS-460

Thick-client shall be configured to run a in a restricted sandbox, with no access to OS resources, such as file system or native libraries.

Verification Method: Test

Requirement ID: IKM-SRS-461

The thick-client binary shall be obfuscated.

Verification Method: Test

Requirement ID: IKM-SRS-462

The thick client shall implement certificate pinning to prevent the proxying of app traffic.

Verification Method: Test

Requirement ID: IKM-SRS-463

The query string shall not be used for sensitive data. Instead, a POST request via SSL should be used with a CSRF token.

Verification Method: Test

4.3 Compatibility-Interoperability

4.3.1 Interface Requirements

Interoperability is defined in ISO 25010 as the degree to which two or more systems, products or components can exchange information and use the information that has been exchanged.

Within NATO, interoperability is defined as, the ability to act together coherently, effectively and efficiently to achieve Allied tactical, operational and strategic objectives.

Requirement ID: IKM-SRS-464

The IKM Tools software code and components shall comply with the latest version of the NATO Interoperability Standards and Profiles (NISP). Any deviation is to be justified and reviewed by the Technical Project Board.

Verification Method: Analysis

Requirement ID: IKM-SRS-465

If new interoperability profiles are to be developed (e.g.: new templates) they shall be compliant and included into the NISP volumes. The interfaces will use the IKM Tools Data Model and be compliant to NATO Core Metadata Specification.

Verification Method: Analysis

4.3.1.1 Interface Control Document

Requirement ID: IKM-SRS-466

For each specified interface (i.e., inputs and outputs to the IKM Tools and between the IKM Tools and another systems), the IKM Tools shall be equipped with an Interface Control Document (ICD) describing the interface. The ICD shall be in a format proposed by the Contractor and accepted by the Purchaser. The content shall include, where applicable, the following information:

Verification Method: Analysis

- A list of the applicable technical standards
- A catalogue of the services and interfaces exposed by the IKM Tools
- A detailed description of the interfaces, including diagrams, data elements, data formats, performance values, communication protocols, security settings, etc.
- Descriptions of data elements
- units of measure required for the data element, such as seconds, meters, kilohertz, etc.
- limit/range of values required for the data element (for constants provide the actual value)
- precision or resolution required for the data element in terms of significant digits,

- frequency at which the data element is calculated or refreshed, such as 10 KHz or 50 msec
- legality checks performed on the data element
- data type, such as integer, ASCII, fixed, real, enumerated, etc.
- data representation/format
- priority of the data element
- Service Descriptors, identifying the services endpoints, a detailed description of the service operations and service parameters
- All related artefacts such as WSDL, schema files and descriptors
- Message descriptions
- Interface priority
- Communications protocol

Interface Mechanisms

4.3.1.2

Requirement ID: IKM-SRS-467

The following information exchange mechanisms can be used in the IKM Tools:

Verification Method: Analysis

- Web services
- File exchange
- Direct database and file access (justification needed)
- API (justification needed)

4.3.1.2.1 Web Services

Web services provide a standard means of interoperability between different software applications, running on a variety of platforms and/or frameworks.

The architecture style defining a SOA (Service Oriented Architecture) describes a set of patterns and guidelines for creating loosely coupled systems that enable a clear separation between the service provider and service consumer. The service provider is the one, who publishes a service description and provides the implementation of the service; whereas the service consumer is the one who invokes the service without knowing any implementation details about the service. This approach not only enables a loosely coupling integration between systems but also simplifies the integration by hiding the unnecessary implementation details.

Web Services are intended to provide self-describing, self-contained, modular units of software application logic that provide defined business functionality. Web Services are consumable software services that typically include some combination of business logic and data. Web Services can be aggregated to establish a larger workflow or business transactions. Inherently, the architectural components of web services support messaging, services description, registries and loosely coupled interoperability.

Requirement ID: IKM-SRS-468

[See Bi-SC Core Enterprise Services Requirements for IKM Tools requirements and Bi SC AIS Platform -Interface Requirements module for a complete set of Web Services requirements via the traceability link].

Verification Method: Analysis

4.3.1.2.2 File Exchange

Requirement ID: IKM-SRS-469

File exchange will require in some instances, to enable the exchange of information between systems for reasons of legacy, security, connectivity, capability or efficiency. Bespoke file formats, where possible, shall use XML as the primary mechanism for file-level information exchange.

Verification Method: Analysis

Requirement ID: IKM-SRS-470

The IKM Tools shall provide adequate documentation for the content and meaning of the file formats it produces or accepts. An adequate definition is one that enables a programmer or user to understand the meaning of the data and determine whether it is suitable for its intended use. The IKM Tools shall supply a definition for every element, attribute, and enumeration value defined in the file format.

Verification Method: Analysis

4.3.1.2.3 Direct Database and File Access

In limited instances, direct database access in the IKM Tools may be required to enable information exchange or visualisation.

Requirement ID: IKM-SRS-471

As a design rule, direct database access in the IKM Tools should be avoided.

Verification Method: Analysis

4.3.1.2.4 Application Programming Interfaces (APIs)

Requirement ID: IKM-SRS-472

The IKM Tools may need to use Application Programming Interface (API) in some special cases only when the use of Web Services is not possible (needs justification). An adequate definition for an API is one that enables a software architect or developer to understand the meaning of the interface and determine whether it is suitable for its intended use. For each API component, the IKM Tools will fully document the interface, including:

Verification Method: Analysis

- Mechanisms for securely invoking the API
- Available methods and functionality
- Available information elements, including attributes and enumeration values
- Error handling.

Interface Security

Requirement ID: IKM-SRS-473

4.3.1.3 The IKM Tools shall use Generic Security Services Application Program Interface (GSS API), as possible, as the application programming interface for accessing security services.

Verification Method: Analysis

Requirement ID: IKM-SRS-474

The IKM Tools primary security services (access control, confidentiality, integrity, authentication, and non-repudiation) shall be supported by X.509.

Verification Method: Analysis

Requirement ID: IKM-SRS-475

The IKM Tools X.509 support to primary security services shall be compliant with NPKI.

Verification Method: Analysis

4.3.2 External Interface Requirements

The IKM Tools has interfaces with NATO external systems and services in order to be able to exchange information. These interfaces can be placed in different networks and 4.3.2.1 domains (NS, NR, NU and Mission Secret).

NATO Bi-SC AIS Core Services

Core Information Services provide the common foundation and standard interfaces to support inter-domain *Interoperability* within NATO and with NATO partner nations. They reflect the minimum requirement to support the command and control of all military functions.

4.3.2.1.1 Unified Communication and Collaboration Services (UCC)

UCC is intended to provide interaction and collaboration services (e.g. text chat, voice over IP, VTC over IP) to support the command and control of all military functions.

Requirement ID: IKM-SRS-476

The IKM Tools shall be integrated with the Bi-SC AIS Unified Communication and Collaboration Services.

Verification Method: Analysis

Requirement ID: IKM-SRS-477

The IKM Tools shall be compatible with Bi-SC AIS collaborative services and protocols (e.g., SIP).

Verification Method: Analysis

4.3.2.1.1.1 Instant Messaging

Instant messaging supports instantaneous synchronous communication and includes the capability to discover people/contacts including their real-time presence information, ad hoc collaboration and participation in a number of concurrent sessions.

Requirement ID: IKM-SRS-478

The IKM Tools shall support interoperability with instant messaging based on the Extensible Messaging and Presence Protocol (XMPP).

Verification Method: Analysis

Requirement ID: IKM-SRS-479

The IKM Tools shall conform to the fundamental features and security mechanisms of instant messaging as described in the Service Interface Profile for Basic Collaboration Services (AI 06.02.12).

Verification Method: Analysis

4.3.2.1.2 Information Exchange Services

The IKM Tools services will perform in multiple security domains (NATO UNCLASSIFIED, NATO SECRET) where rules, regulations and category of personnel are different.

When the IKM Tools servers will exchange and synchronise information between (i.e. across) the NS and MS domains, multiple procedures and devices will be available in support of these exchanges. The IKM Tools will provide the appropriate means to support these cross-domain exchanges.

Cross-domain support service is necessary when the IKM Tools is used in a distributed environment for creating and feeding the Information Products from various sources spread across different Security Domains

The NATO IEG supports different scenarios in order to meet operational needs for Cross-domain information exchange as described in AC/322-D0030-Rev5.

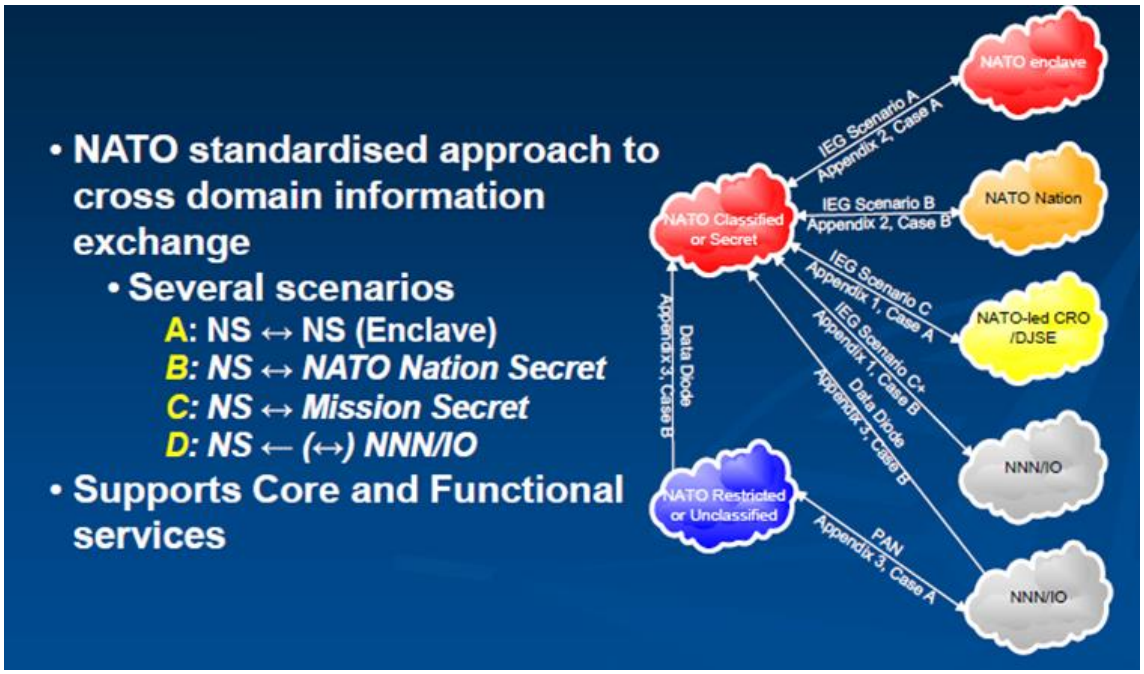


Figure 22 IEG Scenarios

IEG-C operates as two chained HTTP proxies, one for High to Low information exchanges and another one for Low to High information exchanges. Between the proxies there is a filtering engine that sanitizes the information flows in each direction, checking both the header and the body of the HTTP transport messages. The IEG-C targets the content of the HTTP body, which must be an XML document.

IEG-C can filter both SOAP and REST Web Services, or any custom XML protocol that uses HTTP as transport. The IEG-C expects a specific structure for the labelling of the data objects, and the signatures to bind the security labels metadata to the data objects.

The current IEG solution cannot handle HTTPS endpoints.

Requirement ID: IKM-SRS-480

The IKM Tools shall create as required (if not existing from the source), include and maintain the appropriate Security Classification (including Policy (e.g. NATO, NATO/EAPC), Classification (e.g. SECRET) and Category (e.g. Releaseable to ISAF)) for each piece of information across the IKM Tools Services through the use of NATO XML labelling standards.

Verification Method: Analysis

Requirement ID: IKM-SRS-481

The IKM Tools shall assign Confidentiality Labels as defined in [ADatP-4774] and [ADatP-4778] to all information products, information objects, messages, files, bounded data streams that are subject to be transferred over IEG or cross-domain.

Verification Method: Analysis

Requirement ID: IKM-SRS-482

The IKM Tools shall be able to send information products to the IEG-C for NATO Secret to Mission Secret information distribution.

Verification Method: Analysis

Requirement ID: IKM-SRS-483

The IKM Tools shall maintain releasability information for each piece of information across the IKM Tools Services.

Verification Method: Test

Requirement ID: IKM-SRS-484

The IKM Tools shall support Security Labelling elements upon information entity entry and create the corresponding Security Classification using approved NATO labelling standards for each Information Element. 'Entry' refers to direct creation in the IKM Tools; entities imported from an external system/file shall be required to have security labels.

Verification Method: Test

4.3.2.1.3 Windows Domain Services

Windows Domain Services provide Security and Directory Services to the Bi-SC AIS Domain.

Requirement ID: IKM-SRS-485

The IKM Tools shall integrate with the Bi-SC AIS Directory Services Active Directory.

Verification Method: Analysis

Requirement ID: IKM-SRS-486

If the IKM Tools requires a schema change, these schema extensions shall be documented. Any changes to the schema shall be submitted for approval to the Purchaser during the Design Stage.

Verification Method: Test

Requirement ID: IKM-SRS-487

The IKM Tools shall be compatible with Active Directory services and protocols.

Verification Method: Test

Requirement ID: IKM-SRS-488

The IKM Tools shall support integration with Windows File and Print Services (including publishing and lookup through Active Directory).

Verification Method: Inspection

Requirement ID: IKM-SRS-489

The IKM Tools shall support integration with Windows built-in services (e.g., Domain Name System (DNS), Internet Information Services, RUP, Terminal Server).

Verification Method: Inspection

Requirement ID: IKM-SRS-490

The IKM Tools shall support integration with Windows Security Services.

Verification Method: Inspection

Requirement ID: IKM-SRS-491

The IKM Tools shall support integration with Active Directory-supported security access control (e.g., ACL, security groups).

Verification Method: Inspection

Requirement ID: IKM-SRS-492

The IKM Tools shall be able to operate with the latest security settings from the NCIRC without change.

Verification Method: Inspection

4.3.2.1.3.1 Malware Detection Services

Requirement ID: IKM-SRS-493

The IKM Tools will be able to run with the available Malware Detection Services and anti-virus software.

Verification Method: Test

Requirement ID: IKM-SRS-494

The IKM Tools shall coexist (i.e. work correctly and not adversely impact other applications) with Bi-SC AIS standard Anti-Virus software during installation and operation.

Verification Method: Test

Requirement ID: IKM-SRS-495

The IKM Tools shall be equipped with security software that can detect malicious software contained in files of the IKM Tools-delivered workstations and servers. The software shall have the ability to scan any file or directory to detect any malicious software. The supplied software shall be compatible with the NATO Anti-Virus management centre and approved by the Purchaser.

Verification Method: Test

NATO Bi-SC AIS Deployable CIS

4.3.2.2.1 Introduction

NATO DCIS provides a deployable Communications and Information Systems (CIS) capability for deployed forces. DCIS provides CIS capabilities in-theatre and extends services from the fixed infrastructure to deployed users.

4.3.2.2 DCIS will support the full range of required CIS services at the deployed HQ. These CIS services include communication and the Bi-Strategic Command Automated Information Services (Bi-SC AIS) core and functional services. The communication services provide telephony, video conferencing and fax. The core services provide standard office automation including email and data transfer. The Functional services provide services specific military functions. To support deployed operations which may involve C2 participants from Nations which cannot be cleared for access to NATO SECRET (NS) information, it is necessary for the DCIS C2 nodes to support a MISSION SECRET (MS) domain. A NS domain is still required to support NATO activities and a NATO UNCLAFFIFED (NU) domain is required to support sharing of information in an unclassified environment.

NATO will provide the operational command structure of NATO lead expeditionary operations:

- Multinational Force, rotational
- Mission usually involves forces from non-NATO Nations
- Liaison with local governments, NGO, etc.
- 3 information security domains (plus national domains), namely:
    - Mission Secret (default secret mission execution domain, comparable to ISAF Secret),
    - Mission Unclassified
    - NATO Secret

NATO expeditionary operations have the following characteristics:

- Rapidly deployable
- NTM from 2 to 30 days
- Roll-on Roll-off to tactical airlift
- Deployment of forces and HQ structure mission tailored:
- Small mobile teams
- Deployable HQ from 50 to 500 users
- Disaster relief to Corps size operation
- Every mission is different, and never as planned. For this reason, Deployable CIS (DCIS) have to maximize modularity, scalability and flexibility.
- Sustainable with not host nation support

4.3.2.2.2 Context

Once a mission is started, all relevant pre-mission information is transferred into the MS Mission Anchor Point. Operational users from the static environment continue mission preparation in the MS environment, while the deployable nodes that will support the mission are selected and prepared. (note that for missions engaging the NRF-in-standby this nodal selection and preparation is already done during the NRF preparation phase).

Phase 1: Mission secret domain used for mission specific planning

Once the deployable nodes that support the deployable C2 are selected and prepared, all mission applications required for the deployable C2 Entity (JFT HQ and JSLG in the example) and the related databases will be installed on the deployable nodes, and while still in garrison, all preparation data will be synchronised to those nodes.

Phase 2: DCMs are assigned to support NRF C2 entities

When the node deploys into theatre, the applications and databases are not reachable, but planning and preparation by the operational users, still in their peacetime HQ, continues using the MAF.

Once the deployable node becomes operational in theatre, the data in the deployed node is synchronised with the MAF. Operational users in the JTF HQ forward will use the local FS installations for their work, while users supporting the mission from the rear will work on the installations in the MAF, and only the resulting database synchronisation will be executed over the long delay and narrow SATCOM links.

Phase 3: DCMs deploy forward, deployed node contains primary COI database

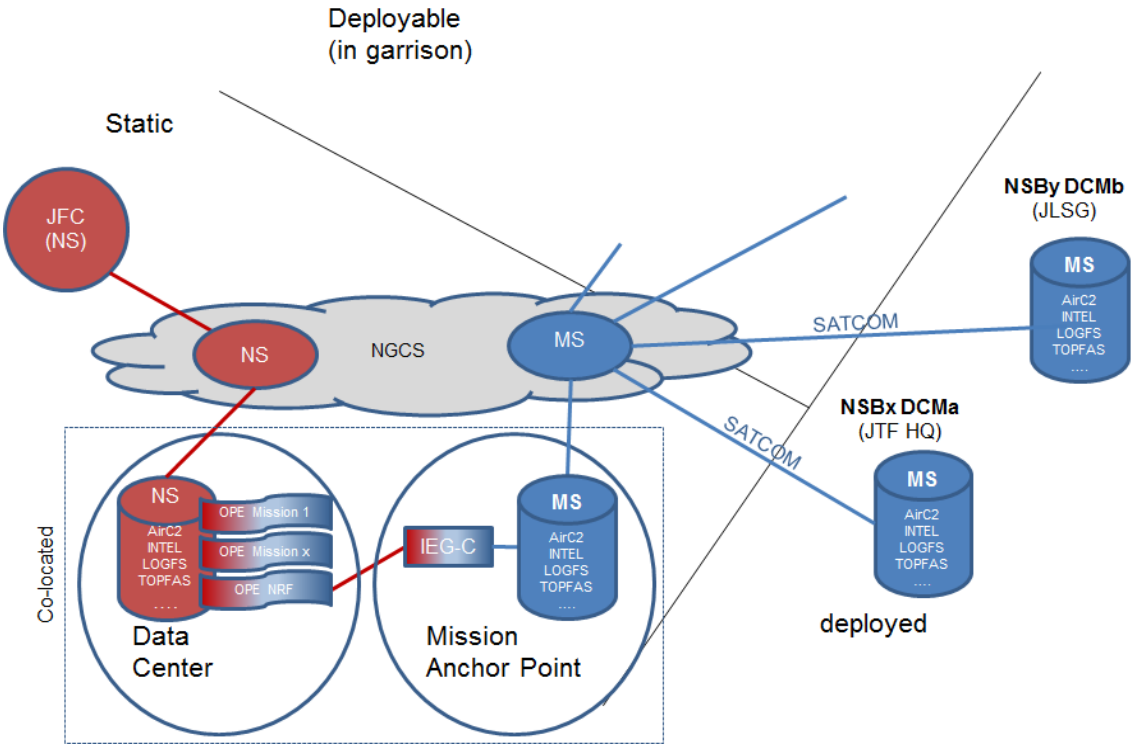


Figure 23 Phase 3: DCMs deploy forward, deployed node contains primary COI database

FAS capabilities will be made available to the users through two options; a full-feature web-based application running on the existing NATO Security Domains (NS, MS and PAN, when applicable) and an equal full-feature client application (for those situations where higher responsiveness is needed, where network latency is too high or where reach-back connectivity is not available).

Any contractor involved in the development of a FAS capability that is meant to be deployable will define and develop test programs, plans, and procedures and conduct testing, oriented specifically to test the deployability of the system, as well as evaluate and document the results. The hardware, software, testing equipment, supplies, facilities, and personnel will be available and in place to conduct or support each test. For acceptance and operational testing, the test data will also be included, and mimic the anticipated operational quantities and sizes of information objects that will be identified in the NATO FAS NFRs. These tests will be performed in an existing or new DCIS Reference environment that accurately simulates the latency and bandwidth network conditions of the deployable DCIS infrastructure.

4.3.2.2.3 IKM Tools Requirements

The services that are actually provisioned on a DCIS node will depend on the nature of the mission involved. It is anticipated that a single DCIS node will vary between requiring the full suite of IKM Tools services (comparable to a Data Centre), selected IKM Tools Services (comparable to an Enhanced Node) or none (a Standard Node).

Requirement ID: IKM-SRS-496

The IKM Tools shall be able to run on the DCIS platform.

Verification Method: Test

Requirement ID: IKM-SRS-497

The IKM Tools shall be deployed depending on the type of DCIS node. A single DCIS node shall vary between:

Verification Method: Analysis

- full suite of IKM Tools services (comparable to a Data Centre),
- selected IKM Tools Services (comparable to an Enhanced Node) or
- none (a Standard Node).

4.3.2.2.3.1 Infrastructure Requirements

Requirement ID: IKM-SRS-498

DCIS must provide equipment to host the NATO Bi-Strategic Command Automated Information Services (Bi-SC AIS) Functional Services, along with any requisite supporting infrastructure. DCIS must also both host and provide Bi-SC AIS Core Services. (The AIS Functional Services applications themselves are provided from outside the DCIS programme).The AIS services must be able to have access to the communications network to allow interaction between elements of the AIS services and to enable the services to be

accessed by users. This equipment is grouped into the µISM subsystem. Each security domain is served by a separate µISM module.

Verification Method: Analysis

Requirement ID: IKM-SRS-499

The backend deployment for NATO locations shall be done using the NATO Infrastructure (Processing, Storage, Networking) Services.

Verification Method: Analysis

Requirement ID: IKM-SRS-500

The IKM Tools shall be deployable in a virtualised server MS Hyper-V and VM Ware virtualised environments

Verification Method: Test

4.3.2.2.4 Quality Requirements

4.3.2.2.4.1 Portability

Requirement ID: IKM-SRS-501

Any instance or application of The IKM Tools shall be able to be centrally configured to use a specific remote or local authorisation and authentication node.

Verification Method: Test

4.3.2.2.4.2 Usage scope and limitations

Requirement ID: IKM-SRS-502

The IKM Tools shall not bear additional licences and charges for deployment of the IKM Tools Product if used in a NATO context (exercise, mission, static and deployable commands, NRF).

Verification Method: Inspection

4.3.2.2.4.3 Availability

In the deployed IKM Tools, five levels of network degradation will be considered due to outages and/or jamming:

- Normal: 90%-100% of the throughput available
- Degraded: 70%-90% of the throughput available
- Severely Degraded: 10%-70% of the throughput available
- Minimum: >0%-10% of the throughput available
- Off-line: 0% of the throughput available

A **degraded network mode of operation** is when the mission-specific WAN or mission-specific LAN is providing a reduced level of service that may impact one or more of the

IKM Tools services (or Application and Interface Products). Reduction in service may be due to bandwidth limitations or a communication degradation affecting some part of the LAN and/or WAN.

Requirement ID: IKM-SRS-503

If the IKM Tools deployed kit cannot connect to the WAN due to communication system failure, bad weather or operational restrictions (e.g. EMCON), the IKM Tools will change the mode of operation to Standalone and continue its operation with limited functionality.

Verification Method: Test

Requirement ID: IKM-SRS-504

A local data storage shall be available in each deployable IKM Tools entity, so that service use and mission execution can continue even if the node is disconnected from the Wide Area Network.

Verification Method: Test

Requirement ID: IKM-SRS-505

The IKM Tools shall queue pending asynchronous ((i.e. do not need immediate feedback) requests to an unavailable service and deliver them when the Service becomes available again (restoring normal functioning).

Verification Method: Test

4.3.2.2.4.4 Maintainability

Requirement ID: IKM-SRS-506

Each deployable IKM Tools entity shall have the capability to be operated by local administration and management, so that if a node is isolated from the central support, local administration and management can be executed.

Verification Method: Test

Requirement ID: IKM-SRS-507

The IKM Tools preparation for deployment and movement shall not take more than 5 days from receiving the Notice to Move until the equipment is packed up ready to move.

Verification Method: Test

4.3.2.2.4.5 Interface Requirements

Once the deployable nodes that support the deployable C2, here IKM Tools, are selected and prepared, all mission applications required for the deployable C2 Entity and the related databases will be installed on the deployable nodes, and while still in garrison, all preparation data will be synchronised to those nodes. It shall be required to switch the roles of the databases. In the early phases, the main database is the in the Mission Anchor Function, and the secondary is the one in the node in garrison. When a node in garrison is deployed, these roles may change.

Requirement ID: IKM-SRS-508

The IKM Tools data resynchronisation in deployment after long periods off-net shall resolve consistency conflicts.

Verification Method: Test

Requirement ID: IKM-SRS-509

The IKM Tools shall implement data compression that guarantees an efficient use of network bandwidth.

Verification Method: Test

Requirement ID: IKM-SRS-510

The IKM Tools shall implement incremental database synchronization communication protocol that guarantees an efficient use of network bandwidth.

Verification Method: Analysis

Requirement ID: IKM-SRS-511

The IKM Tools shall not encrypt WAN data exchange.

Verification Method: Test

4.3.3 Co-existence Requirements

Co-existence is the degree to which a product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product.

Requirement ID: IKM-SRS-512

The IKM Tools shall operate with other Bi-SC AIS Functional Services in the same environment without causing any error condition in itself or in other systems.

Verification Method: Test

4.4 Design Constraints

4.4.1.1

4.4.1 Architectural Constraints

General

Requirement ID: IKM-SRS-513

The IKM Tools shall be compliant with the standards given in the SoW section Applicable Documents. Any proposed deviation shall be approved by the Purchaser.

Verification Method: Analysis

Requirement ID: IKM-SRS-513b

The IKM Tools shall be compliant with the Microsoft supportability limitations so that the recommended limitation are not breached, i.e. maximum number of Sites per Site Collection or maximum Content Database size) (see Ref. [Software boundaries and limits for SharePoint Servers 2016 and 2019]). Any proposed deviation shall be approved by the Purchaser.

Verification Method: Analysis

Requirement ID: IKM-SRS-514

The IKM Tools shall be designed and implemented with economy of bandwidth (e.g., caching, light-weight previewing, metadata exchange, lazy loading).

Verification Method: Analysis

Requirement ID: IKM-SRS-515

The proposed software architecture, development environment, middleware system and the separation of components (Human Machine Interface, Business and Data) for the IKM Tools shall be documented and explained in detail.

Verification Method: Analysis

Requirement ID: IKM-SRS-516

The IKM Tools shall expose selected functionality as Services, as components of a service-oriented architecture to encourage reuse and interoperability with other applications in a distributed way. The criteria used for selection of functionality shall be documented.

Verification Method: Analysis

Requirement ID: IKM-SRS-516b

The IKM Tools Search service shall be implemented as autonomous and separated from the IKM Tools, that is, as separated Share Point farm following the Microsoft best practices.

Verification Method: Analysis

Requirement ID: IKM-SRS-516c

The IKM Tools Search service in ON shall be implemented as a Medium Search Farm following the Microsoft Guidelines (ref: [Plan enterprise search architecture in SharePoint Server]).

Verification Method: Analysis

Requirement ID: IKM-SRS-517

The IKM Tools services shall comply with the C3 Classification Taxonomy and applicable Service Interface Profiles.

Verification Method: Analysis

Requirement ID: IKM-SRS-518

The IKM Tools shall, where applicable, make use of REST architecture to make resources available over a URL in promotion of interoperability.

Verification Method: Analysis

Requirement ID: IKM-SRS-519

The IKM Tools design process shall balance design implementation with cost for implementation and support to minimise life cycle cost. The IKM Tools design shall take into account the technical, support and cost impacts for NATO.

Verification Method: Analysis

Browser-based Functionality

4.4.1.2

Requirement ID: IKM-SRS-520

The IKM Tools user functionality shall be browser-based, except as specifically waived by the Purchaser.

Verification Method: Inspection

Requirement ID: IKM-SRS-521

The IKM Tools user functionality shall require only an ordinary browser and should not require the installation of additional software, components or plug-ins on the user workstation, except as specifically waived by the Purchaser.

4.4.1.3

Verification Method: Test

COTS selection and integration

Requirement ID: IKM-SRS-522

The IKM Tools shall be based on COTS in its architecture in place of dedicated solutions when the functionalities of a COTS matches the requirements for a Service with no or minimal adaptation.

Verification Method: Analysis

Requirement ID: IKM-SRS-523

When no or minimal adaptation is necessary for the COTS integration, the IKM Tools shall use COTS with no additional layer to masquerade nor shield the COTS and the complete functionalities of the COTS shall be available to the IKM Tools Services.

Verification Method: Analysis

Requirement ID: IKM-SRS-524

The IKM Tools adaptations shall be delivered as additional Services that complement the COTS native functionalities.

Verification Method: Analysis

4.4.2 Data Management

General

Requirement ID: IKM-SRS-525

4.4.2.1 The IKM Tools shall utilise a Database Management System (DBMS) to manage all internal storage (some configuration files can be excluded).

Verification Method: Analysis

Requirement ID: IKM-SRS-526

The IKM Tools DBMS shall be compatible with NATO Infrastructure.

Verification Method: Analysis

Requirement ID: IKM-SRS-527

The IKM Tools database shall support backup and archiving. The backup and archive shall be full, incremental backups and archives of the IKM Tools data. The backup and archive shall be performed both automatically at a configurable frequency and manually by the authorised User.

Verification Method: Test

Requirement ID: IKM-SRS-528

The IKM Tools shall support recovery facilities of the database from backup and archive data.

Verification Method: Test

Requirement ID: IKM-SRS-529

4.4.2.2 The IKM Tools shall provide ADO.NET and ODBC database interfaces.

Verification Method: Analysis

Data Modelling

Requirement ID: IKM-SRS-530

The IKM Tools shall have a documented Conceptual, Logical and Physical data model

Verification Method: Analysis

Requirement ID: IKM-SRS-531

The IKM Tools shall use a standard naming convention for the database design.

Verification Method: Analysis

Data Replication

The basic function of the data replication capability is to make the same planning data or operational data available on both the source and destination databases on different servers separated by a wide area network. Replication can also simply have redundancy purposes.

4.4.2.3

Requirement ID: IKM-SRS-532

The IKM Tools shall provide the necessary mechanisms to ensure complete, accurate, timely, confidential and consistent data coherence between the different distributed Organizational Nodes of the system.

Verification Method: Analysis

Requirement ID: IKM-SRS-533

The IKM Tools shall use the ITM replication mechanism (aka: Metalogix, or iOra) to replicate the data from SHAPE Data Center to the Lago Patria Data Center so that the IKM Tools availability and Fault Tolerance KPIs are met.

Verification Method: Test

Requirement ID: IKM-SRS-534

The IKM Tools data replication shall be:

Verification Method: Analysis

- Synchronous: the IKM Tools shall automatically replicate information held in a source database only when it is created/changed/deleted, and that is recognised by the replication mechanism. It shall replicate to all co-operating systems/nodes that have been configured to exchange this information.
- Asynchronous: the IKM Tools shall allow an authorised User to specify intervals of replication.

Requirement ID: IKM-SRS-535

The IKM Tools shall allow an authorised User to selectively identify the data to be replicated.

Verification Method: Test

Requirement ID: IKM-SRS-536

The IKM Tools shall allow the IKM Tools System Administrators to configure authoritative sources of data, and the destination databases (i.e., what systems/nodes) to which data is to be replicated.

Verification Method: Test

4.4.3 Graphical User Interface (GUI)

NCIA and NATO guidelines

Bi-SC AIS applications are developed as projects within NCIA to be used by NATO users. Both NCIA and NATO have their own standards and guidelines that will influence or directly affect Bi-SC AIS applications' visual design. Although Bi-SC AIS applications can have their own identity, any new application needs to feel like other products NCIA or NATO have previously created and share the same organizational values.

4.4.3.1

Requirement ID: IKM-SRS-537

The IKM Tools visual design shall follow the recommendations and guidelines stated in the following documents:

- NATO Visual Identity Guidelines
- NCIA Visual Identity Guidelines

Verification Method: Inspection

Requirement ID: IKM-SRS-538

The IKM Tools shall follow the recommendations and guidelines of the HMI Style Guide for C4ISR Rich Applications regarding to windows and layouts, user interactions, user support and feedback, common user interface components design, visual design and text use.

Verification Method: Inspection

4.4.3.2

ISO standards

Requirement ID: IKM-SRS-539

The IKM Tools icons included in the designed solution shall be compliant with the ISO 18152 standard series.

Verification Method: Analysis

Requirement ID: IKM-SRS-540

The IKM Tools shall be compliant with the ISO 9241 standard series. In particular:

Verification Method: Analysis

Requirement ID: IKM-SRS-541

The IKM Tools shall be compliant to ISO 9241-125 for the presentation of information.

Verification Method: Analysis

Requirement ID: IKM-SRS-542

The IKM Tools shall be compliant to ISO 9241-13 for user guidance.

Verification Method: Analysis

Requirement ID: IKM-SRS-543

The IKM Tools shall be compliant to ISO 9241-14 for menu dialogues.

Verification Method: Analysis

Requirement ID: IKM-SRS-544

The IKM Tools shall be compliant to ISO 9241-143 for form filling dialogues

Verification Method: Analysis

Requirement ID: IKM-SRS-545

The IKM Tools shall be compliant to ISO 9241-171 for accessibility.

Verification Method: Analysis

Requirement ID: IKM-SRS-546

The IKM Tools shall follow the dialogue principles stated in ISO 9241-110 or ISO/DIS 9241-110 (if available)

Verification Method: Analysis

4.4.3.3 Log-on procedures

Requirement ID: IKM-SRS-547

In applications where users must log-on to the system, log-on shall be a separate procedure that must be completed before a user is required to select among any operational options.

Verification Method: Analysis

Requirement ID: IKM-SRS-548

Appropriate prompts for log-on should be automatically displayed on the user's terminal when accessing the application.

Verification Method: Analysis

Requirement ID: IKM-SRS-549

User identification procedures shall be as simple as possible, consistent with adequate data protection.

Verification Method: Analysis

Requirement ID: IKM-SRS-550

When required, the password shall not be echoed on the display. An asterisk (*) or similar symbol will be displayed for each character when inputting secure passwords during log-on.

Verification Method: Analysis

Requirement ID: IKM-SRS-551

When passwords are required, users shall be allowed to choose their own passwords since a password chosen by a user will generally be easier for that individual to remember. Guidelines for password selection shall be given so that users will not choose easily guessable ones.

Verification Method: Analysis

Requirement ID: IKM-SRS-552

Users should be allowed to change passwords whenever they choose; all passwords should be changed at periodic intervals (not to exceed a configurable number of months).

Verification Method: Analysis

Requirement ID: IKM-SRS-553

Users shall be provided feedback relevant to the log-on procedure that indicates the status of the inputs.

Verification Method: Analysis

Requirement ID: IKM-SRS-554

If a user cannot log-on to a system, a prompt should be provided to explain the reason for this inability. Log-on processes should require minimum input from the user consistent with the requirements prohibiting illegal entry.

Verification Method: Analysis

4.4.3.4 Log-off procedures

Requirement ID: IKM-SRS-555

When a user signals for system log-off, or application exit or shut-down, the system should check pending transactions to determine if data loss seems probable. If so, the computer should prompt for confirmation before the log-off command is executed.

Verification Method: Test

4.4.3.5 Data entry

Requirement ID: IKM-SRS-556

Data entry functions shall be designed to establish consistency of data entry transactions, minimize input actions and memory load on the user, ensure compatibility of data entry with data display, and provide flexibility of user control of data entry.

Verification Method: Inspection

Requirement ID: IKM-SRS-557

Data entries should be validated by the system for correct format, legal value, or range of values. Where repetitive entry of data sets is required, data validation for each set should be completed before another transaction can begin.

Verification Method: Test

Requirement ID: IKM-SRS-558

The user shall not be required to enter data already available to the software.

Verification Method: Inspection

Requirement ID: IKM-SRS-559

The IKM Tools shall support copying and pasting from other applications via the Clipboard (including text and graphics).

Verification Method: Test

Requirement ID: IKM-SRS-560

The IKM Tools shall allow users to edit Metadata in bulk so that user can see and modify at least 100 list items (i.e. via window scrolling).

Verification Method: Test

4.4.3.6 Data and content display

Requirement ID: IKM-SRS-561

In the IKM Tools, user-accessed information shall be filtered according to the user's role and organization, so that Users can only see what they are allowed to view and/or change (including both data and functionality).

Verification Method: Inspection

Requirement ID: IKM-SRS-562

Data and content displayed shall follow the Windows and Layout recommendations of the HMI Style Guide for Rich C4ISR Applications

Verification Method: Analysis

Requirement ID: IKM-SRS-563

The IKM Tools shall provide cursor control capability via a Cursor Control Device (mouse or similar). The response of a cursor to control movements shall be consistent, predictable, and compatible with the user's expectations.

Verification Method: Inspection

Requirement ID: IKM-SRS-564

Data deletion or cancellation shall require an explicit action, such as depressing a DELETE key. Permanent deletion (in absence of an "undo" function) of more

than one character shall not be allowed without an affirmative response to an "are you sure?" type of query.

Verification Method: Inspection

Default values

Requirement ID: IKM-SRS-565

4.4.3.7 Default values shall be used to reduce user workload. Currently defined default values should be displayed automatically in their appropriate data fields with the initiation of a data entry transaction and the user shall indicate acceptance of the default.

Verification Method: Test

Requirement ID: IKM-SRS-566

The IKM Tools shall provide auto-completion feature with session-available information to be filled automatically by the IKM Tools (i.e., date/time, name and other profile information of the User, classification). This shall apply to "entry fields" (including text-fields, drop down lists, radio-buttons).

Verification Method: Inspection

Requirement ID: IKM-SRS-567

The user should have the option of generating default values based on operational experience if the systems designer cannot predefine appropriate values.

Verification Method: Inspection

Requirement ID: IKM-SRS-568

The user shall be able to replace any default value during a given transaction without changing the default definition.

Verification Method: Test

4.4.3.7.1 Error management and data protection

Requirement ID: IKM-SRS-569

Where users are required to make entries into a system, an easy means shall be provided for correcting erroneous entries. The system shall permit correction of individual errors without requiring re-entry of correctly entered commands or data elements.

Verification Method: Test

Requirement ID: IKM-SRS-570

A capability should be provided to facilitate detection and correction of errors after keying in, but before entering into the system. While errors should be detected early, error checking should occur at logical data entry breaks, e.g., at

the end of data fields rather than character-by-character, in order to avoid disrupting the user.

Verification Method: Test

Requirement ID: IKM-SRS-571

Provision shall be made to prevent accidental actuation of potentially destructive control actions, such as accidental erasure or memory dump.

Verification Method: Test

4.4.3.7.2 Individualization

Requirement ID: IKM-SRS-572

Mechanisms shall be provided to allow the characteristics of the IKM Tools to be modified by the user to take account of the diversity of user characteristics, where such needs typically occur.

Verification Method: Analysis

Requirement ID: IKM-SRS-573

The IKM Tools shall allow Users to quickly revisit recent functions and features and to save 'favourites' of features and functions that are often used.

Verification Method: Analysis

Requirement ID: IKM-SRS-574

The IKM Tools shall allow the User to set up personal preferences for layout and content.

Verification Method: Test

Requirement ID: IKM-SRS-575

In the IKM Tools, the feature to change the user interface appearance shall be visible and accessible on every screen.

Verification Method: Test

Requirement ID: IKM-SRS-576

The IKM Tools shall provide a list of the 'last accessed items' or recently used items/functions. This list shall always be selectable for display.

Verification Method: Test

Requirement ID: IKM-SRS-577

The IKM Tools shall persist user customization settings.

Verification Method: Test

4.4.4 Software Design

General

Requirement ID: IKM-SRS-578

The IKM Tools shall be designed to work in the ITM ON and PBN environments as consumers and providers of other Functional Services and Core Services.

4.4.4.1 *Verification Method:* Test

Requirement ID: IKM-SRS-579

The IKM Tools shall comply with latest versions of the Microsoft Windows operating system available and supported by the ITM servers and workstations. This should include all versions of the operating systems planned to become available for the ITM prior to the Integration Tests.

Verification Method: Test

Programming Languages and Technologies

4.4.4.2 *Requirement ID:* IKM-SRS-580

The IKM Tools shall comply with the language specifications and versions compatible with ITM and described below, and with the AFPL. Any variations from the languages or specifications shall be agreed with the Purchaser:

Verification Method: Inspection

- .NET
- C++ [ISO/IEC 14882]
- C# [ISO/IEC 23270:2003]
- Common Language Infrastructure (CLI) [ISO/IEC 23271:2003 and ISO/IEC 23272:2003]
- Java [JSR 379]

4.4.4.3

- JavaScript [ECMA 262]
- HTML [ISO/IEC 15445, 2000]

Coding Standards

Requirement ID: IKM-SRS-581

Whilst the specifics of coding syntax are not specified, a convention shall be adopted and applied consistently across all code artefacts for each programming language employed.

Verification Method: Inspection

Requirement ID: IKM-SRS-582

Source code artefacts delivered for the IKM Tools shall be written using Standard English (e.g. for Classes, Methods, Variables etc.). Industry coding best practices shall be used.

Verification Method: Inspection

Requirement ID: IKM-SRS-583

The IKM Tools .Net components shall be developed in compliance of the following Microsoft Visual Studio Code Analysis Tool rule sets:

Verification Method: Inspection

- Microsoft Basic Correctness Rules
- Microsoft Basic Design Guideline Rules
- Microsoft Extended Correctness Rules
- Microsoft Extended Design Guideline Rules
- Microsoft Minimum Recommended Rules
- Microsoft Security Rules

Requirement ID: IKM-SRS-584

Valid exceptions to those rules shall be created for each applicable occurrence (e.g. global exclusion is not allowed unless explicitly approved by the Purchaser)

Verification Method: Inspection

Requirement ID: IKM-SRS-585

All custom code methods shall have a Unit Test defined and included in the delivered package to the Purchaser.

Verification Method: Analysis

4.4.4.4 Code Documentation

Requirement ID: IKM-SRS-586

Source code delivered for the IKM Tools shall be documented with in-line comments using Standard English. Commercial best practices shall be used in the level of commenting.

Verification Method: Inspection

Requirement ID: IKM-SRS-587

Comments for the source code of the IKM Tools shall be used to clarify intent of the code and shall be provided for:

Verification Method: Inspection

- Each class definition explaining the purpose of the class
- Each member function explaining what the function does, its inputs and outputs
- Each member variable explaining what the variable means
- Each type definition (enums) explaining what the type represents

Requirement ID: IKM-SRS-588

Comments for the source code of the IKM Tools shall be able to be automatically extracted and formatted to augment technical documentation.

Verification Method: Inspection

Requirement ID: IKM-SRS-588b

Every method of every class shall contain at minimum the following comments on the header of the method, using the following template:

Verification Method: Inspection

```
    <summary>
/// short description mentioning the purpose of this method
/// </summary>
/// <param name="FieldInternalName">the value</param>
/// <param name="anotherInternalName">the valueparam>
/// <returns>value returned</returns>
```

Requirement ID: IKM-SRS-588c

The source code provided by the Contractor shall conform to the coding conventions stated with Annex XXX “C# Coding Conventions”

Verification Method: Inspection

Requirement ID: IKM-SRS-588d

The Contractor shall provide an API reference conforming to the “General API Reference Code Comments”

4.4.4.5

Verification Method: Inspection

Registry Settings

Requirement ID: IKM-SRS-589

All usage of the Windows Registry by the IKM Tools applications shall be fully documented, and requires approval by the *Purchaser* not later than the Design Stage.

Verification Method: Inspection

4.4.5 Free and Open Source software (FOSS)

Requirement ID: IKM-SRS-590

FOSS components in the IKM Tools shall comply with the NATO strategy on the use of Open Source Software in NATO systems.

Verification Method: Analysis

Requirement ID: IKM-SRS-591

Any IKM Tools components based on free and open source software shall be provided with the source code for the FOSS. The source code shall correspond to the delivered component (i.e., same version), and the component shall be capable of being built from the delivered source code.

Verification Method: Analysis

Requirement ID: IKM-SRS-592

Use of a FOSS component shall not limit the deployment or use of the IKM Tools in any way and shall not require the release of code developed for the IKM Tools.

Verification Method: Analysis

4.5 Technical Documentation Requirements

4.5.1 General

The requirements describing which technical documentation shall be developed and how the technical documentation shall be managed and taken under configuration control are in the SoW. This section of the SRS will cover the requirements which are applicable to the online technical documentation.

Requirement ID: IKM-SRS-593

The general requirements for technical documentation developed in the SoW shall also apply to the on line documentation.

Verification Method: Analysis

Requirement ID: IKM-SRS-594

The IKM Tools on-line User documentation and help system shall be compliant with standards identified under section "Applicable Standards".

Verification Method: Analysis

4.5.2 Technical Documentation

On-line Help

4.5.2.1.1 General

The IKM Tools will be used by organizations in various time zones throughout NATO territories and other areas of NATO operations. During crisis use of the IKM Tools will be high and over extended working hours. Full on-line help capability will be required to supplement the IKM Tools help-desks.

4.5.2.1

Requirement ID: IKM-SRS-595

The IKM Tools shall support on-line help describing all functionality of the IKM Tools capability (i.e. publishing the delivered User and Administration Manuals).

Verification Method: Inspection

Requirement ID: IKM-SRS-596

The IKM Tools on-line help shall translate every use case and usage scenario into a browsing sequence. Every browsing sequence shall be structured according to the User workflow.

Verification Method: Inspection

Requirement ID: IKM-SRS-597

The IKM Tools on-line help shall describe each the IKM Tools function, the interrelationships between and the logical sequence of functions.

Verification Method: Inspection

Requirement ID: IKM-SRS-598

The IKM Tools on-line help shall explain all menu items, dialog windows, data entry and query fields implemented in the IKM Tools Product Baseline.

Verification Method: Inspection

Requirement ID: IKM-SRS-599

The IKM Tools on-line help shall include a glossary providing definitions of all terms and acronyms implemented in the IKM Tools Product Baseline.

Verification Method: Inspection

Requirement ID: IKM-SRS-600

All definitions in the IKM Tools glossary shall be available in roll-over, pop-up windows linked to every appearance in on-line help of the corresponding term or acronym.

Verification Method: Inspection

Requirement ID: IKM-SRS-601

In the IKM Tools, each dialogue, menu item, toolbar item, function, field or button (each item on the screen) shall have an on-line help option. This shall be clearly visible, but not intrusive.

Verification Method: Inspection

Requirement ID: IKM-SRS-602

The IKM Tools on-line help shall be concise, compact and clear to the User.

Verification Method: Inspection

Requirement ID: IKM-SRS-603

The on-line help shall include snapshots of the IKM Tools screens, windows, and dialogue boxes. The snapshots shall be provided in a suitable lightweight format (e.g., GIF, PNG) approved by the Purchaser.

Verification Method: Inspection

Requirement ID: IKM-SRS-604

Pictures in the IKM Tools on-line help showing more than five GUI elements/controls shall have a clickable image map describing each element.

Verification Method: Inspection

Requirement ID: IKM-SRS-605

If the IKM Tools on-line help topic requires a large picture that does not fit on a normal page, a reduced copy shall be additionally included on the Help page that will expand to its full size on User request.

Verification Method: Inspection

Requirement ID: IKM-SRS-606

The IKM Tools on-line help shall be context-sensitive (i.e., based on a specific point in the state of the software and providing help for the situation that is associated with that state on action being performed).

Verification Method: Inspection

Requirement ID: IKM-SRS-607

In the IKM Tools, all source code elements shall be configured to link the GUI elements to their context-sensitive topics.

Verification Method: Inspection

Requirement ID: IKM-SRS-608

The IKM Tools shall contain help functions that provide access to interactive training sessions to guide Users through procedures and functions.

Verification Method: Inspection

Requirement ID: IKM-SRS-609

The IKM Tools on-line help shall be given by a small pop-up screen or info tip screen. This screen shall appear quickly and be very easy to hide, for instance clicking anywhere within it.

Verification Method: Inspection

Requirement ID: IKM-SRS-610

The IKM Tools shall be able to display search query results for finding help items in the online help in a list. The IKM Tools shall display the help item when the User selects a query result in this list.

Verification Method: Test

Frequently Asked Questions (FAQ)

4.5.2.2

Requirement ID: IKM-SRS-611

The IKM Tools shall provide a list of Frequently Asked Questions.

Verification Method: Inspection

Requirement ID: IKM-SRS-612

The IKM Tools shall allow the User to search the IKM Tools FAQ

Verification Method: Test

4.6 Computer Resource Constraints

4.6.1 Hardware and Software Components ITM

Requirement ID: IKM-SRS-613

The IKM Tools shall be able to run on NATO-provided infrastructure including virtual servers and operational workstations.

Verification Method: Analysis

Requirement ID: IKM-SRS-614

The IKM Tools shall be compatible with the x86-64 architecture (32-64 bit applications).

Verification Method: Analysis

Requirement ID: IKM-SRS-615

The IKM Tools shall support the following environment configuration for the server components:

Verification Method: Analysis

- Operating System: Microsoft Windows Server.
- Database Server: Microsoft SQL Server
- Web Server: Internet Information Services (IIS)

Requirement ID: IKM-SRS-616

The IKM Tools shall use virtualized infrastructure components (processing, storage, local area networking, monitoring, management and security) with a resource definition equivalent to the requirements defined in NATO Bi-SC AIS Minimum Hardware Procurement Specifications (MHWPS), when applicable. The allocation of virtualized infrastructure components shall be approved by the purchaser.

Verification Method: Analysis

Requirement ID: IKM-SRS-617

The infrastructure requirements of a service or application shall not be designed and deployed with a "per service, per application" approach but shall be covered through a harmonized "infrastructure services" approach applicable to all services and applications making use of the infrastructure services.

Verification Method: Analysis

Requirement ID: IKM-SRS-618

The IKM Tools software shall run on NATO Server Baseline (see Ref [MHWPS], [BAPPL12] and [AFPL]).

Verification Method: Inspection

Requirement ID: IKM-SRS-619

The IKM Tools software shall not have any direct dependency to the physical parameters of the storage environment (such as disk type, connection type, SAN topology, SAN protocol).

Verification Method: Inspection

Requirement ID: IKM-SRS-620

The IKM Tools software shall not have any hard coded UNC, File Path, Drive Letter or similar storage location settings.

Verification Method: Inspection

Requirement ID: IKM-SRS-621

All UNC, File Path, Drive Letter or similar storage location settings shall be parametric, configurable, and possible to automate for unattended installation, backup, recovery.

Verification Method: Inspection

Requirement ID: IKM-SRS-622

The IKM Tools software shall have no hard coded URL, DNS or IP Address settings. All URL, DNS, IP Addressing and similar network settings shall be parametric, configurable, and possible to automate for unattended installation, backup, recovery.

Verification Method: Inspection

Requirement ID: IKM-SRS-623

The server deployment package (virtual appliance or installation package) shall be tested against the following hypervisors, configured in accordance with NATO Security Settings:

Verification Method: Analysis

- VMWare ESX
- Microsoft Hyper-V

Requirement ID: IKM-SRS-624

The IKM Tools Client functionality shall be compliant with the NATO Desktop Baseline. Including:

Verification Method: Test

- MS Windows Operating system
- MS Office
- MS Edge browser and MS Internet Explorer
- Adobe Acrobat Reader
- Java Virtual Machine
- McAfee Anti-Virus and Data Loss Prevention (DLP) agent
- NCIRC desktop Host-based Intrusion Detection System (HIDS) and Forensics analysis based agents
- VPN client for PBN mobile client devices; and
- Disk encryption for PBN mobile client devices

Requirement ID: IKM-SRS-625

The IKM Tools SMC integration shall be compliant with the NATO EMS.

Verification Method: Analysis

Requirement ID: IKM-SRS-626

The IKM Tools Servers shall be able to be deployed in the virtual environments running on:

Verification Method: Analysis

- Data Centres provided by the NATO IT Modernisation (ITM) project, in the static environment, and
- Micro Information Service Module (æISM) provided by NATO Response Force (NRF) Deployable Communications and Information Systems (DCIS) project, in the deployable configuration.

Requirement ID: IKM-SRS-627

The IKM Tools shall support replication and backup services provided by the Data Centres deployed by the NATO IT Modernisation (ITM) project.

Verification Method: Test

Requirement ID: IKM-SRS-628

The IKM Tools shall support multiple browsers, including the ITM provided browsers: Internet Explorer, Edge and Firefox

NATO Communications and Information Agency
Agence OTAN d'information et de communication

NCI Agency Core Enterprise Services (CES) - Infrastructure as a Service (IaaS)

Hosting Guidelines for New Applications' Management Standard Operating Procedure (SOP)

Hosting Guidelines for New Applications' Management SOP

Version 0.51

Table of Contents

1 Introduction

This section contains a description of the context, vision, objectives, policy, strategy, assumptions and references of the Hosting Guidelines document.

1.1 Vision, scope and objectives

This Hosting Guidelines document has a vision to serve as the de-facto hosting guidelines for application owners/developers who develop new applications that they want to host on the ON and/or PBN environments, and as such should be included in the Statement of Work requirements for software projects.

The specific scope and objectives of this Hosting Guidelines document are to:

- Provide guidance to application owners/developers so that applications are developed / adapted (including possibly redesigned, if required. However, please note: it is strongly recommended that application owners/developers do ***not*** wait that long before applying these guidelines set out in this document, so as to have to redesign the application as a result, but that they apply them at the very beginning of the development effort to avoid any redesign and rebuilding efforts) so that they can be:
  - Hosted within the ON and/or PBN Infrastructure as a Service (IaaS) environments
  - Provided to end users via the ITM application provisioning mechanism and technology
  - Managed by the ITM SMC toolset
- Provide, through the Hosting Guidelines:
  - An overview of the ITM scope and purpose
  - ON and PBN environment hosting information
  - Information required by NCI Agency from the applications to be hosted (e.g. CPU, RAM, and storage capacities).

1.2 Target Audience

This document is intended for Application Owners/Application Developers who wish to integrate and host their newly-developed/to-be-developed applications on the NCIA ON and/or PBN environments. The Application Owners/Application Developers shall use this document for the following purposes:

- Provide to NCIA the necessary input information in terms of infrastructural requirements in order to host their application on the ON and/or PBN environments (chapter 3).
- Use it to define project requirements for new applications to be capable to be hosted and operated on the ON and/or PBN environments (chapter 4).

1.3 Policy

The Hosting Guidelines specified in this document are in compliance with, and a complement to, the NCI Agency Directive AD 06.00.07 “NCI Agency Information and Communication Technology (ICT) & Architecture Principles”, effective date: 15 September 2015.

1.4 Strategy

The Hosting Guidelines specified in this document can be applied by the Application Owners/Developers at any point in time throughout the development lifecycle of their application, but it is advisable that they are read and adhered to at the start of the development effort, to minimise the risks and costs to this effort.

1.5 Assumptions

The Hosting Guidelines laid out in this document are based on the following assumptions:

1. The Hosting Guidelines cover new applications only (i.e. not currently existing within the NATO estate) to be hosted on the ON and/or PBN environment.
2. Existing NATO applications to be migrated to the ON and/or PBN environments are out of scope of the Hosting Guidelines. Migration and infrastructure Hosting Guidelines for these applications will be provided in ITM site-specific Design and Implementation Plans.
3. The Hosting Guidelines will be focused on applications joining the ON and/or PBN environments as the target scenario.
4. The Hosting Guidelines laid out in this document are not Release and Deployment (as defined by ITIL v3) guidelines. The latter will be specified by NCIA and must be complied with separately, by all applications wishing to be hosted on the ON and/or PBN environments.

1.6 Structure of the document

This document consists of the following chapters:

- Chapter 1 – Introduction
- Chapter 2 – Overview of NCIA's IT Modernisation project
- Chapter 3 – Application requirements for NCIA IaaS hosting (to be provided by Application Owners/Developers to NCIA)
- Chapter 4 – Hosting Guidelines for hosting NATO cloud-capable applications on the NCIA ITM infrastructure (provided by NCIA to Application Owners/Developers)

1.7 Contributors

The following people have contributed to the development of this document:

Name	**NCI Agency Business Unit**	**Document areas contributed to**
Lothar Meinel (document owner from 2017 onwards)	Core Enterprise Services	All
Lukasz Sokolowski (document owner from 2016-2017)	Ex-ITM Project Manager	All
Annie Nicholson-Jones (Asparouhova) – author and Gartner point of contact	External (Gartner)	All
Martin Diepstraten	Core Enterprise Services	All
Steve Fouquaert	Network Services and IT Infrastructure	All
Torsten Graeber	Service Strategy	All
Aneta Kubicka	Cyber Security	Security aspects across all
Dave Stanley	Cyber Security	Security aspects across all
Yves Van Nimmen	Cyber Security	Security aspects across all
Erdem Ayvaz	Service Management and Control	SMC aspects across all

Ioannis Dimou	Service Management and Control	SMC aspects across all
Jose Luis Pascal Herrero	Service Strategy	Chapter 3
Andy Stefanescu	Core Enterprise Services	Mobile baseline in Annex C
Pieter Jansen	Acquisition	Chapter 3.11
Sarah Green	Acquisition	Chapter 3.11
Bart Van Miert	Service Strategy	ITM Service Catalogue in Annex B

1.8 References

The references used in compiling these Hosting Guidelines are:

- ITM Work Package 1 – Bidder Instructions
- ITM Work Package 5 – Task Order 2 deliverables (ITM Service catalogue and Cost models)
- NCI Agency Customer Services Catalogue v2.2 10 July 2017

Gartner best-practices:

[1] Gartner report G00270845 “A Guidance Framework for Architecting Portable Cloud Applications”
[2] Gartner report G00270697 “Assessing Micro-service Architecture for Scalable Application Delivery”
[3] Gartner report G00162312 "Composite Applications Help Create Value by Blending SOA, Web 2.0, Cloud and Legacy Applications
[4] Gartner report G00226919 “Cloud Characteristics, Principles and Design Patterns”
[5] Gartner report G00308407 “A Guidance Framework for Architecting Highly Available Cloud-Native Applications”
[6] Gartner report G00296114 “How to Architect and Design Cloud-Native Applications”
[7] Gartner report G00314366 “Best Practices for Backup and Recovery of Large Virtualized Environments”
[8] Gartner report G00319911 “Technology Insight for Cloud Infrastructure as a Service”
[9] Gartner report G00326583 “How to Assess Your Application to Adopt Cloud-Native Architecture”

1.9 Glossary

Term	Definition
ACT	Alliance Command Transformation based in Norfolk, USA
ACO	Alliance Command Operations based in Mons, Belgium
Application Owner	Owner of a software application
CSRS	Community Security Requirement Statement
FOC	Full Operational Capability

Instrumentation	A software application is well instrumented when its performance can be monitored and measured and errors can be diagnosed, using a management tool like SolarWinds for example.
ITM	IT Modernisation – Modernisation of NCIA's IT infrastructure
IV&V	Independent Verification and Validation. In NATO, IV&V is sometimes referred to, what is called in Industry – Quality Assurance.
NCIRC	NATO Computer Incident Response Capability
NOS	NATO Office of Security based in NATO HQ, Brussels, Belgium
NSAA	NATO Security Accreditation Authority
NSAB	NATO Security Accreditation Board
NR	NATO Restricted information
NU	NATO Unclassified information
ON	Operational Network (for information up to NS)
PBN	Public Business Network (for information up to NR)
QA	Quality Assurance – term used in Industry to refer to validation and verification that a certain piece of work/product/service conforms to pre-agreed requirements and standards.
SAA	Security Accreditation Authority
SAP	Security Accreditation Plan
SecOps	Security Operating Procedures
SISRS	System Interconnection Security Requirement Statement
SRA	Security Risk Assessment
SSRS	System Security Requirement Statement

2 NCIA ITM overview

This chapter contains a brief overview of the ITM scope, services and technical requirements.

2.1 ITM scope and objectives

The ITM project will transform the way IT services are provided to users across the NATO enterprise, including the NATO Command Structure (NCS), the NATO Headquarters (NHQ), elements of the NATO Force Structure and NATO agencies. This will be achieved by modernizing, consolidating, and centralising the infrastructure and service management, and by pooling and abstracting resources.

IT Modernization will deliver **a private, on premises cloud, providing Infrastructure as a Service (IaaS) for the ON and PBN environments**. These services will be used to support all of NATO's business needs for static elements. Two, resilient, logically integrated but geographically dispersed, infrastructures will host all of NATO's applications supporting static locations, negating the need for individual projects to provide hardware to support their capabilities.

The Functional Scope of the ITM Project consists of the implementation of four ITM services:

1. **Infrastructure as a Service** that offers the backend infrastructure services to provide the user applications at the required service levels based on an efficient service delivery method. It consists of five elements:
    - Datacentre Node: key centralised IaaS location where the bulk of computing will take place.
    - Standard Node: Location with limited amount of computing and storage in support of local user services access.
    - Enhanced Node: Location with enhanced computing and storage capabilities in order to support applications that are not deemed to be centralise-able for technical or other reasons, to provide a level of graceful degradation should communications be interrupted, or to provide a higher level of user experience. In all cases the delivered capability shall still be provisioned and managed centrally, with local support being limited to first level.
    - Service Operations Centre: The NCI Agency will operate two Service Operations Centres from which it will administer, and operate all of the NCI Agency provisioned C4ISR services. The two Service Operations Centres will operate in an active-active mode, such that between them they can operate all services on a 24/7 basis, and when needed either can subsume the complete role, administering and operating all NCI Agency provisioned services.
    - Remote Nodes: In some user locations, with a limited number of users (<10), ITM will provision client devices, but all other services will be centrally provisioned.
2. **Client Provisioning** that will allow NATO applications to be accessed from any approved client appliance form factor from authorised locations by authorised users. This includes the support for user mobility in NATO locations and beyond.
3. **Service Management and Control** that will manage and control the provision and operation of services identified above.
    - This includes the implementation of the ITIL processes that have been tailored for the Purchaser, and the monitoring to allow the tracking of the achievement of Service Level Targets.
    - The SMC capability consists of three layers, Enterprise SMC, Domain SMC and Element Manager.
4. **Enterprise Core Services** for mail, IM and portal.

These four services will be made available to NATO Enterprise users through the ITM Service Portfolio and Catalogue, available here.

For more detailed information about the ITM project and each of the above four services, please refer to the ITM Frequently Asked Questions, available here.

3 Application requirements for hosting and costing

This chapter contains the template that the Application Owner/Developer must complete and return to NCIA (fields in yellow only), specifying the requirements of the application in order for it to be hosted on the ON and/or PBN environment. The information provided allows NCIA to ensure that appropriate capacity is allocated within the private cloud infrastructure to leverage full benefits of agility and resiliency within required service levels.

The text marked in *[green]* specifies the link to the Additional Capacity Form fields that need to be completed by the Service Owner of projects to develop new applications that will need to be hosted on ON and/or PBN. The TBCE costing requirements and Additional Capacity Form are described in section 3.11.

3.1 General information

In this section the Application Owner/Developer shall provide general information in terms of:

- Application general details
- Application dependencies
- Application components and software stack

3.1.1 Application general details

Please specify in the following table the application general information regarding application demographic and main characteristics.

Application General Details	
Application Name	*<Specify application name>*
Application Code	*<Specify the official code of the application>*
Application Publisher	*<Organization that provides the application (i.e. NATO)>*
Application current version	*<If the application is already in production specify the current version>*
Application release	*<Release number to bring in production >*
Release date	*<Forecasted installation date for the release, if known>*
Application Status	*What is the current lifecycle stage of the application?* *(In development, testing, early adoption, production, declining, sunsetting)*
Application type	*<Specify application type: web-app; mobile app; portal; mail; desktop app; batch/massive; analytics app; integration.* *It could be also a combination of the above>*
User access type	*<Specify in which of the following way the application will be accessed, and what user authentication it requires:* - *a thick client, installed on the fixed client device (back end in the case of VDI clients)* - *a thick client on the PBN mobile client device in support of off-line use;* - *Browser based access to web enabled applications;*

	- *Virtualized access through application streaming, remote desktop or Server based computing>*
Business criticality	*<Rank the business criticality of the application* *(1=Mission Critical, 2=Important, 3=Not Critical)>*
Working period	*<Specify the range of hours/days in which the application is in use (e.g.8-18 Mon-Fri; 8-20 Mon-Sat; 24X7)>*
Peak period	*<Specify peak period during the day/weak (e.g. 15-17 Mon-Fri)>*
Application security level	*<PBN or ON>*
Data security classification	*<Classify the security classification of data managed by the application: cosmic top secret, NATO secret, NATO confidential, NATO restricted, , NATO unclassified, public>* *[this is the Classification/Network type, in the Additional Capacity Form]*
Type of users	*<Specify which are the type of users :* - *Fixed-Desktop clients - ON network* - *Fixed-Desktop clients PBN network* - *Mobile clients - PBN network*
N. of users	*< Specify total number of users that will access the application and their location inside NATO, and which users from which domain-security zone.>*
Application description	*<Provide a brief description of the application>*
Application functionalities	*<List the primary business function provided by the application. (e.g. sales management and reporting for division A)>*
Business application owner	*<Person responsible for application from Business>*
IT application owner	*<Person responsible for application from IT>*
Business Entities supported	*<List the business/units serviced by the application>*
Multi-division	*<Is the application used in multiple Business Divisions / Lines of Business? (Y/N)>*
Multi-country	*<Is the application used in multiple countries/geographies? (Y/N)>*
Multi-language	*<The application supports many languages? (Y/N)>*
Standalone application	*<Is the application stand-alone or does it communicate with other systems/applications? (Y/N)>*
Interoperability with other systems	*<Is the application dependent on another application for major functionality? If Yes, pls fill out next table>.*
N. of Interfaces	*<Estimate the number of interfaces. (e.g. 50-75)>*

Communication Interfaces	*<List the type of communication interfaces: API, WS, Events, file transfer, ETL, DB link, NFS>*

Table 1 - Application general details

3.1.2 Application dependencies

Please specify in the following table the application dependencies with other applications.

Application dependencies	
Application name	**Dependency Relationship**
<Specify the name of the dependent application>	*<Specify the dependency details:* - *type of interface (online, batch)* - *mode of interaction (sync, asynch, scheduled)* - *exchange format (REST/Jason, SOAP/XML, FTP/file, etc)* - *protocols (HTTP/S,SSL, FTP,etc.)* - *frequency (n messages/sec for online, scheduled time for batch)* - *volumes (message size for online, file size for batch interface)* - *location (DC in which reside the application)*

Table 2 - Application dependencies

3.1.3 Application components

Specify the list of application components that can be deployed separately in a virtual or physical machine.

Application Components (VM)		
Application component name	**Description**	***Virtual/Physical***
<Specify the name of the application component>	<Provide a brief description of the component>	<Specify if the component can be deployed on a virtual or physical machine>

Table 3- Application components

3.1.4 Application component deployment model

Specify for each application components the allocation of the instances on each node of the network how many instances will be deployed on which nodes and in which modality (active or passive instance).

Application Components (VM)			
Application component	**Node name (DC,EN)**	**Active instances**	**Passive (Standby) instances**
<Specify the name of the application component>	<Specify the type (DC, EN) and name of the Node on which it will be deployed>		

Table 4 - Application deployment

All application components should be deployed in the NCIA data centres. NCIA will allow some stand-by instances to be deployed to an enhanced node (no applications are allowed to be hosted on standard nodes). Enhanced node instances will be activated only if there is a loss of connectivity or for recovery from a DC fail-over.

Please state the impact such a loss would have on your application, if any, and how your application will recover from this.

3.1.5 Application software stack

In the following table specifies for each application component the corresponding software stack and the required hosting options:

- **Platform as a service (PaaS)**: the software is provided as a service by NCIA
- **Managed service (MS)**: the software is provided by the application owner but is fully managed by NCIA (DIS and DAS)
- **Unmanaged service (US)**: the software is provided and managed by the application owner but hosted on NCIA infrastructure. NCIA provides only hardware, OS instance and storage

Application Software stack			
Application component name	**Type of software**	**Description**	***Required service model***
<Insert application component name>	***Server OS***	*<Specify server operating systems and specific version supported: Windows, RH Linux, Solaris>*	*<Specify the service model applicable for this component: PaaS, MS, US)*
<Insert application component name>	***DBMS***	*<Specify the DBMS supported and the specific version: MS SQL Server, Oracle >*	*<Specify the service model applicable for this component: PaaS, MS, US)*

<Insert application component name>	***Middleware***	*<Specify the middleware required for running the applications: applications server, web server or integration software. Examples are iPlanet, Novell, Active Directory, WebSphere/MQSeries, Domino, TIBCO, Informatica, Apache Tomcat; web proxies, gateways - Brio, HTTP Server, Visualizer, web servers >*	*<Specify the service model applicable for this component: PaaS, MS, US)*
<Insert application component name>	***Other required SW, COTS or libraries - server side***	*<Specify any additional runtime software libraries or COTS needed to run the application server side>*	*<Specify the service model applicable for this component: PaaS, MS, US)*
<Insert application component name>	***Web browser***	*<For web based application specify the Browser supported and the specific version: IE11, Edge, Chrome, Safari>*	*<Specify the service model applicable for this component: PaaS, MS, US)*
<Insert application component name>	***Client OS***	*<Specify client operating systems and specific version supported: Windows 7,8,10>*	*<Specify the service model applicable for this component: PaaS, MS, US)*
<Insert application component name>	***Other required SW or libraries – client side***	*<Specify any additional runtime software libraries needed to run the application client side>*	*<Specify the service model applicable for this component: PaaS, MS, US)*

Table 5 - Application SW stack

3.2 ITM Services required by the application

In this section the Application Owner/Developer shall specify the list of ITM services required by the application in order for it to run. For more information about each service, please see Annex B.

3.2.1 Workplace services

If workplace services are needed please provide the following information:

Workplaces services	
Managed Device	*<Specify the number of devices for PBN and ON users (#desktop,# laptop, # tablet, # thin client, #mobile/smartphone), if any>*
User Access	*<Specify the number of users that require user access, if any>*

Print/Scan/Copy	*< Specify: # devices required from one of the following, for both ON and PBN, if any:* ▪ *Large MFP (Printer/ Scanner/Copier) Badge controlled* ▪ *Printer* ▪ *Customer Owned Printer* ▪ *Scanner* ▪ *Large Format printer (Plotter)* >
E-mail	*<Specify the number of user mailboxes required, if any>*
Instant Messaging Collaboration	*<Specify the number of Skype for Business accounts to be activated, if any>*
Enterprise User License	*<Specify the number of physical users required for one of the following, if any:* ▪ *NATO Light Profile is allowed to use the licensed Products on up to 2 (two) Qualified Devices whereas at least one of those Qualified Devices is connected to the internet.* ▪ *NATO Standard Profile is allowed to use the licensed Products on up to 5 (five) Qualified Devices.* >

Table 6 Workplace services

3.2.2 Platform services

If platform services are needed please provide the following information:

Platform services	
Database Platform	*<Specify the number of instances and size required by your application of one of the following database platform services, if any:* ▪ *Oracle shared high-availability database platform.* ▪ *Oracle dedicated Single Instance database platform* ▪ *Oracle dedicated high-availability database platform* ▪ *SQL Server shared high-availability database platform* ▪ *SQL Server dedicated Single Instance database platform* ▪ *SQL Server dedicated high-availability database platform* >
Web Information Publishing and Portal	*<Specify if your application needs a:*

	▪ *NATO Information Portal (NIP): SharePoint NATO SECRET Site Collection with 10 GB (expandable) of storage;* ○ *<Specify how much additional storage you require, if any>* ▪ *Or a NATO Standard collaboration platform: Sharepoint Site Collection with 10GB (expandable) of storage;* ○ *<Specify how much additional storage you require, if any>* ▪ *Or a NATO NU extranet web publishing and portal (up to and including NATO Unclassified)* ○ *<Specify how much additional storage you require, if any>* *>*
Web Hosting	*<Specify the number of instances your application requires of one of the following services, if any:* ▪ *Shared High-availability web hosting platform (load balancing with multiple web-servers)* ▪ *Dedicated Standard web hosting platform* ▪ *Dedicated High-availability web hosting platform (load balancing with multiple web-servers)* *>* *<Specify which web hosting platform you require for your chosen web hosting service above:* ▪ Microsoft Internet Information Services (IIS) ▪ Apache ▪ SharePoint >

Table 7 - Platform services

3.2.3 Infrastructure services

Please provide the following information for required infrastructure services:

Infrastructure services	
Infrastructure Storage	*<Specify the number of gigabytes (GBs) required for each of the following, if any:* ▪ *Storage Area Network (SAN)* ▪ *Network Attached Storage (NAS)* *> <Specify, for each of the above, if high availability is required>*

Infrastructure Backup/Archive	*<Specify the number of gigabytes (GBs) required for backup/archiving, if any>*
LAN	*<Specify the number of 24 ports required for one of the following, if any:* - *Wired* - *Wireless*
Enterprise Internet Access	*<Specify the number of users requiring one of the following, if any:* ▪ *Fixed Internet Access: a fixed connection to the end user, by means of a cable, through which the user can access onto the internet.* ▪ *Wireless Internet Access: provision of wireless connectivity from the user's device to the internet.* *>*
Infrastructure Hosting	*<Specify the number of instances required of one of the following, if any:* ▪ *Physical General Purpose Server: A physical server comprising of a dual x86 CPU, 16GB RAM and 80GB 1 Disk storage.* ▪ *Physical High Performance Server: A physical server comprising of a dual x86, 64GB RAM, 80GB 1 Disk Storage, dual 10GB network interface and SAN HBA.* ▪ *Virtual Large Server: A virtual server supporting 8 vCPU's, 64GB RAM and 80GB disk storage, shared 10 gigabit network access.* ▪ *Virtual medium Server: A virtual server supporting 4 vCPU's, 32GB RAM and 80GB disk storage, shared 10 gigabit network access.* ▪ *Virtual small Server: A virtual server supporting 2 vCPU's, 8GB RAM and 80GB disk storage, shared 1 Gigabit network access.* *>* *<Specify if the above-chosen service is to be provided with Windows or Linux Operating System>* *<Specify if the above-chosen service requires additional options:* ▪ *Redundancy* ▪ *Load balancing for selected applications and protocols (e.g. HTTP)* ▪ *Additional SAN/NAS storage* ▪ *Backup/archiving*

	>
Infrastructure Network	*<Specify the number of purchasing customers requiring an infrastructure network service>* *<Specify if you require any of the following additional options:* ▪ *Load balancing for selected applications and protocols (e.g. HTTP)* ▪ *Network acceleration for specific applications* >

Table 8 - Infrastructure services

3.3 Capacity allocation requirements for infrastructure services

In this section the application provider shall specify the application requirement in terms of computing capacity and storage capacity.

3.3.1 Processing capacity

Please provide for each component (virtual or physical) and for each environment the # of instances required and the capacity required as multiples of CU unit (1 CU unit = 1vCPU, 2 GB RAM, 64 GB DASD)

Computing capacity					
Environment	**Application component name**	**VM/ Server**	**# instances**	**Processing Capacity**	**Type of configuration**
<Specify the type of environment: - DEV - TEST - IV&V - STAGING - PROD>	<Specify the name of the application component>	<Specify if it will be installed on a VM or a physical server>	<Specify the # of instances to be installed >	<Express the capacity required considering foreseen average workload. For VM specify capacity as a multiple of CU units (1vCPU, 2 GB RAM, 64 GB DASD) For dedicated server specify the exact size in terms of #core/CPU, #GB RAM, # GB HD) >	<Describe type of configuration: Active-Active (load balancing) Active-Passive (fail over) Active-active-passive (load balancing + fail over) >
<Specify the integration environment>	<Specify the name of the application component>	<Specify if it will be installed	<Specify the # of instances to	<Express the capacity required considering	<Describe type of configuration:

		on a VM or a physical server>	be installed >	foreseen average workload. For VM specify capacity as a multiple of CU units (1vCPU, 2 GB RAM, 64 GB DASD) For dedicated server specify the exact size in terms of #core/CPU, #GB RAM, # GB HD) >	Active-Active (load balancing) Active-Passive (fail over) Active-active-passive (load balancing + fail over) >

Table 9 - Computing capacity allocation

3.3.2 Storage capacity

Please provide in the following table for each environment the storage capacity for Enterprise or Midrange SAN including VM storage

Storage capacity			
Environment	**Type of Storage**	**Node type**	**Storage capacity**
<Specify the type of environment: - DEV - TEST - IV&V - STAGING - PROD>	<SAN Enterprise, SAN Midrange, NAS>	<Indicate the type of node where the storage must be provided>	<Indicate storage capacity foreseen as #GB > *[this is the Total GB of raw, useable storage required, in the Additional Capacity Form]*
<Specify the integration environment>	<SAN Enterprise, SAN Midrange, NAS>	<Indicate the type of node where the storage must be provided>	<Indicate storage capacity foreseen as #GB >

Table 10 - Storage capacity allocation

Please provide in the following the storage capacity required for data archiving:

Archiving storage capacity						
Type of Storage	**Node type**	**Type of archive**	**Archiving duration**	**Frequency**	**Retention period**	**Capacity**

<SAN Enterprise, SAN Midrange, NAS	<DC,EN>	<Online/ offline>	<Specify for how many months the data is maintained in the archive >	<Specify how often and/or which are the rules of the archiving process>	<Specify the number of months the archive data must be maintained on the storage before deleting or moving to a low-cost type of storage)	<Indicate required storage capacity as #GB > *[this is the Total GB of raw, useable storage required, in the Additional Capacity Form]*

Table 11 - Back up capacity allocation

Please provide in the following table for each environment the storage capacity required for online and offline backup:

Back up storage capacity							
Environment	**Type of Storage**	**Node type**	**Type of Backup**	**Capture**	**Frequency**	**Retention period**	**Capacity**
<Specify the type of environment: - DEV - TEST - IV&V - STAGING - PROD>	<SAN Enterprise, SAN Midrange, NAS	<DC,EN>	<Online/ offline>	<Specify if the backup should be full or differential>	<Specify the frequency of the backup>	<Specify the number of months the backup data must be preserved)	<Indicate required storage capacity as #GB > *[this is the Total GB of raw, useable storage required, in the Additional Capacity Form]*
<Specify the integration environment>	<SAN Enterprise, SAN Midrange, NAS	<DC,EN>	<Online/ offline>	<Specify if the backup should be full or differential>	<Specify the frequency of the backup>	<Specify the number of months the backup data must	<Indicate required storage capacity as #GB >

						be preserved)	

Table 12 - Back up capacity allocation

3.3.3 Network capacity

In the following tables the application provider shall specify the networking capacity for PBN and ON users for each consumer site.

PBN LAN-WAN segment network capacity- Users		
Site Name *<specify the site or domain name from which users will access the application>*	# named static users	<number of users that are authorized to access the application from the site on a wired-fixed network>
	# concurrent static users (average)	<number of users that access concurrently at the application from the site during normal operation>
	# bandwidth per static user (average)	<Kbps consumed by each static user using the application >
	# named mobile users	<number of mobile users that are authorized to access the application from the site on a wireless network>
	# concurrent mobile users (average)	<number of mobile users that access concurrently at the application from the site during normal operation>
	# bandwidth per mobile user (average)	<Kbps consumed by each mobile user using the application >
	# named admin users	<number of admin users that are authorized to access the application from the site during normal operation>
	# concurrent admin users (average)	<number of admin users that access concurrently at the application from the site during normal operation>
	# bandwidth per admin user (average)	<Kbps consumed by each admn user using the application >
	Total user bandwidth x site (average)	*<Kbps required totally by the site (average)>*
	Traffic details	*<specify QoS , aggregated traffic classes, traffic flows, MTU sizes, packet sizes, packet size distribution, and multicast details, where known>*

Table 13- PBN Network users

ON LAN-WAN Network capacity - Users		
Site Name	# named static users	<number of users that are authorized to access the application from the site on a wired-fixed network>

<specify the site or domain name from which users will access the application>	# concurrent static users (average)	<number of users that access concurrently at the application from the site during normal operation>
	# bandwidth per static user (average)	<Kbps consumed by each static user using the application >
	# named admin users	<number of admin users that are authorized to access the application from the site during normal operation>
	# concurrent admin users (average)	<number of admin users that access concurrently at the application from the site during normal operation>
	# bandwidth per admin user (average)	<Kbps consumed by each admin user using the application >
	Total user bandwidth x site (average)	<Kbps required totally by the site (average)>
	Traffic details	*<specify QoS , aggregated traffic classes, traffic flows, MTU sizes, packet sizes, packet size distribution, and multicast details, where known>*

Table 14 - ON Network users

In the following table please specify the Network requirements at DC segment level.

Data Centre - LAN-WAN Network capacity		
DC/ Node Name *<specify the Data centre or Node name, if known >*	*End Users generated traffic on node*	*<volume of traffic on average generated on DC by end users (kbps)>*
	Admin users generated traffic on node	*<volume of traffic on average generated on DC by admin users (kbps) >*
	Traffic from other application residing on different DC/EN/Domain	*<volume of traffic on average generated on DC/EN by integrated applications residing on different DC/EN>*
	Traffic details	*<specify QoS , aggregated traffic classes, traffic flows, MTU sizes, packet sizes, packet size distribution, and multicast details, where known>*

Table 15 - DC network capacity

In the following table please specify the Data centre to Data centre replication requirements:

Data Centre - Data replication		
	Target DC	*<specify the target node of the data replication>*

DC/ Node Name *<specify the source data centre or Node name, if known >*	*Volume*	*<volume of data on average moved by a replication process (kBps)) >*
	Type	*<Sync/ asynch>*
	Time to replicate	*<Max latency for data synchronization>*

Table 16 - Data replication

In the following table please specify the Load balancing requirements for each node:

Data Centre - Load balancing		
DC/ Node Name *<specify the source data centre or Node name, if known >*	*Protocol*	*<Specify the used protocol (e.g. TLS)>*
	Trans/sec	*<Specify the number of transactions per sec>*
	Dispatching scheme	*<Specify the load balancing dispatching scheme>*

Table 17 - Load balancing

3.4 Operational level agreements

In this section please state which Operational Level Agreement (OLA) targets your application requires so that you can meet your SLA with your customer.

Operational Level Agreement		
Environment	**OLA item**	**Description**
<Specify the type of environment for which the OLA is required: - DEV - TEST - IV&V - STAGING - PROD>	Operation hours	<Specify operation time. This is the agreed time period when a particular IT service should be available. E.g. "Monday to Friday 08:00 to 17:00 except public holidays" >
	Support	1st level run support (e.g. self-service: Mo-Fr 8-18 CET).
		2nd level support (e.g. on escalation from level 1: Mo-Fr 8-18 CET).
	Deployment window	< Specify the required availability window for deployment on the ON and PBN environment>
	Maintenance Windows	< This agreed time is reserved to the provider to update the hardware, the operation system or the application provided. During this time a service is potentially not available. The times of maintenance windows will be excluded in the intrinsic availability calculation. (e.g.) 1 x quarterly Saturday 4 pm – Sunday 12 am CET/CEST >
	Intrinsic Availability Level	< Specify the expected level of availability (e.g. 98%, 99%, 99, 8%, and 99,99%)> Intrinsic availability is the theoretical availability of the system. This is based on the MTBF and the organizational MTTR only. Intrinsic availability is expressed as a

		percentage per year, using the following formula: 100 x (MTBF/(MTBF+MTTR))> *[this is the Level of Availability, in the Additional Capacity Form]*
	Operational availability	<Specify the expected operational availability. This includes all components that are part of the end-to-end service, even those that are not under the responsibility of the ON and PBN infrastructure such as the WAN equipment, application availability and the operational support services. Operational Availability is expressed in a percentage per month, using the following formula: 100 x (uptime – downtime)/uptime.>
	Reliability	<Specify Mean Time Between Service Incidents (MTBSI) measured at Service Desk. Reliability = Available time (Hours)/ Number of breaks - for a service>
	Backup interval	< Specify the expected backup interval (e.g. backed-up every 24 hours), capture (full/differential) and retention period (e.g. backup and restore is available for 90 days)>
	Disaster Recovery Level (RTO and RPO)	<Specify the required level for Disaster recovery in terms of: RTO: This is the threshold for how quickly the applications and information is restored RPO: This is the threshold of how much data can be afforded to be lost since the last backup. RPO is measured in hours of data loss >
	Incident response time (Is the time from opening an incident by 1st Level Support to its first assignment to a person in the resolver group)	< Specify the expected response time of incidents based on their severity: Critical tickets: 95% <= 30min Important tickets: 95% <= 1h Major tickets: 80% <= 4h Minor tickets: 80% <= 8h >
	Incident resolution time Is the time from opening an incident by 1st Level Support to its resolution	< Specify the expected resolution time of incidents based on their severity: Critical tickets: 95% <= 4h Important tickets: 95% <= 8h Major tickets: 80% <= 3BD Minor tickets: 80% <= 5BD >

Table 18 - SLA requirements

3.5 Scaling requirements

Specify year-on-year (YoY) scaling requirements for computing processing.

Computing scaling requirements							
Application component name	**VM/ Server**	**# instances**	**Y1 (percentage increase)**	**Y2 (percentage increase)**	**Y3 (percentage increase)**	**Y4 (percentage increase)**	**Y5 (percentage increase)**
<Specify the name of the application component>	<Specify if it will be installed on a VM or a physical server>	<Specify the # of instances to be installed >	<Express the expected increase in processing capacity after 1 year in production >	<Express the expected increase in processing capacity after 2 years in production >	<Express the expected increase in processing capacity after 3 years in production >	<Express the expected increase in processing capacity after 4 years in production >	<Express the expected increase in processing capacity after 5 years in production >

Table 19 - Computing scaling capacity YoY

Specify normal operation time and peak period for online processing

Peak period			
Application component	**Normal operation time**	**Peak period**	**Peak increase**
<Application component Name>	<Specify the operation time (e.g. 8-18 Mon-Fri)>	<Specify the peak period inside normal operation time (e.g. 9-11 Mon-Fri)>	>Specify the % increase respect to normal operation in number of transaction related to that component>

Table 20 – Peak period

Specify batch volumes and time windows for massive processing.

Batch volumes and windows			
Application component	**Frequency/Schedule**	**Batch window**	**Batch volume**
<Application component Name>	<Specify the frequency of the batch (e.g daily, weekly, biweekly, monthly>	<Specify the time windows for the batch (e.g. 12.pm - 2.p.m)	<Specify the average volume of data elaborated by the batch>

Table 21 – Batch windows

Specify YoY scaling requirements for Storage capacity (max capacity expected). ON/PBN storage capacity will be able to scale up instantly by 25% over and above the stated yearly capacity and year-on-year growth requirements specified for your application.

Storage scaling capacity						
Type of storage	**Type of Node**	**Y1 (percentage increase)**	**Y2 (percentage increase)**	**Y3 (percentage increase)**	**Y4 (percentage increase)**	**Y5 (percentage increase)**
<SAN Enterprise, SAN Midrange, NAS >	<DC, EN>	<Express the expected increase in storage capacity after 1 year in production >	<Express the expected increase in storage capacity after 2 years in production >	<Express the expected increase in processing storage after 3 years in production >	<Express the expected increase in processing storage after 4 years in production >	<Express the expected increase in processing storage after 5 years in production >

Table 22 - Storage scaling capacity YoY

Specify YoY scaling requirement for PBN network users. ITM LAN capacity will be able to scale up instantly by 25% over and above the stated yearly capacity and year-on-year growth requirements specified for your application.

PBN network scaling capacity						
Node	**Type of users**	**Y1 (percentage increase)**	**Y2 (percentage increase)**	**Y3 (percentage increase)**	**Y4 (percentage increase)**	**Y5 (percentage increase)**

<Site Name>	<Static, mobile, admin>	<Express the expected increase in concurrent users after 1 year in production >	<Express the expected increase in concurrent users after 2 years in production >	<Express the expected increase in concurrent users after 3 years in production >	<Express the expected increase in concurrent users after 4 years in production >	<Express the expected increase in concurrent users after 5 years in production >

Table 23 - PBN Network scaling capacity YoY

Specify YoY scaling requirement for ON network users. ON/PBN LAN capacity will be able to scale up instantly by 25% over and above the stated yearly capacity and year-on-year growth requirements specified for your application.

ON network scaling capacity						
Node	**Type of users**	**Y1 (percentage increase)**	**Y2 (percentage increase)**	**Y3 (percentage increase)**	**Y4 (percentage increase)**	**Y5 (percentage increase)**
<Site Name>	<Static, mobile, admin>	<Express the expected increase in concurrent users after 1 year in production >	<Express the expected increase in concurrent users after 2 years in production >	<Express the expected increase in concurrent users after 3 years in production >	<Express the expected increase in concurrent users after 4 years in production >	<Express the expected increase in concurrent users after 5 years in production >

Table 24 - ON Network scaling capacity YoY

3.6 Security requirements

Please specify in the following table which security mechanisms your application is dependent on and any particular requirements or constraints of this application. The security mechanisms provided by the ON/PBN environment are described in Annex A.

Security mechanisms			
SM nr.	**SM Name**	**In Scope (Y/N)**	**Requirements and constraints**
SM01a	Malware Protection for Server (e.g. AV for servers)		
SM01b	Malware Protection for Application Server (e.g. SharePoint)		
SM01c	Malware Protection for Multifunction Printing (MFP) device		
SM01d	Malware Protection for Database Server		
SM02a	Malware Protection for Client (e.g. AV for clients)		
SM02b	Second but different Malware Protection for Client (e.g. second but different AV for clients)		
SM03	Malware Protection for handheld devices (e.g. smartphones)		
SM04	Malware Protection for Email server (e.g. AV for email services)		
SM05	Malware Protection for Web server (e.g. AV for Web Services)		
SM06	Messaging Content Filtering (MCF)		
SM07a	Web Content Filtering (WCF) - Categorization		
SM07b	Web Content Filtering (WCF) - Content Inspection		
SM07c	Web Content Filtering (WCF) - SSL Intercept		
SM08	HTTP AV Proxy		
SM09	FTP AV Proxy		
SM10	Integrity Checker		
SM11	SSL (TLS) and SSH		
SM12	Strong Authentication (User Token)		
SM13	Enterprise Single Sign-On (ESSO)		
SM14	Firewall (FW) for Outer Perimeter/ Border Protection		
SM15	Firewall (FW) for Inner Perimeter		
SM16N	Intrusion Detection and Prevention System (IDS/IPS) - Network (N) Based		
SM16H	Intrusion Detection and Prevention System (IDS/IPS) - Host (H) Based		
SM17	Network Access Control (NAC) / Network Access Protection (NAP) and Network Access Quarantine (NAQ)		
SM18	IP Filtering & Management		
SM19	Network Address Translation (NAT/PAT)		
SM20	(Web) Application Firewall and other proxy/reverse proxy		

SM21a	System and Security Logging & Auditing - Infrastructure and Servers		
SM21b	System and Security Logging & Auditing - Applications		
SM22	Hardening of Networked Devices		
SM23	Voice over Internet Protocol (VoIP) protection		
SM24	Wireless Network Protection (and Jamming)		
SM25	Storage Compartmented Security Mode (SCSM)		
SM26	Backup, Recovery and Contingency planning		
SM27	Network Management System (NMS) /Systems Management System (SMS)		
SM28	IT Forensic and Incident Handling		
SM29	Cryptographic security (e.g. encryption / Decryption)		
SM30a	NPKI - User certificate		
SM30b	NPKI - Device certificate		
SM31	Logical Security Zones		
SM32	Policies, directives, guidance and procedures		
SM33	Load Balancing (LB) / Fail over (FO)		
SM34	Information Protection Control (IPC) – Classification/Marking		
SM35	Information Protection Control (IPC) – Data Loss/Leak Prevention (DLP)		
SM36	Vulnerability Scanning & Compliancy Check		
SM37	OS security settings		
SM38	Quality of Service (QoS)		
SM39	Development/Test/Pre-production Environments		
SM40	Configuration Management		
SM41	CAPTCHA (and its alternatives)		
SM42	Identity & Access Management (IAM or IdM)		
SM43	Management of the Security Mechanisms		
SM44	Time Synchronisation (e.g. NTP)		
SM45	Storage Areas security		
SM46	Data Scramble		
SM47	Data Safeguarding (e.g. DAM, FSAM)		
SM48	Password Management		
SM49	Identity & Authentication, Access Control (IAAC)		
SM50	Protection against (Distributed) Denial-of-Service or (D)DoS		
SM51	Physical security		
SM52	Personnel security		
SM53	Environmental security		
SM54	Emission security		
SM55	Data processing		
SM56	Data Diode		

Table 25 – Security requirements

3.7 Monitoring and service management requirements

Specify for each application component any monitoring requirements.

Monitoring and service management	
Application component name	**Monitoring requirements**
<Specify application component name>	<State application monitoring requirements (e.g. logging mechanisms, SMTP traps, custom monitoring console and reporting interface, Patching and Software Update Services)>

Table 26 – Monitoring requirements

3.8 Manpower requirements

Specify indicative manpower required for transition period and during operations to manage, operate and monitor the application.

Manpower requirements for application operation	
Specify type of activity	**# FTE**
<Specify type of activity name (e.g. Transition period, Service desk; technical management; It operations management; application management)>	<Average number of FTE required to perform the activity>

Table 27 – Manpower requirements

Specify other requirements or prerequisites, like training, in order to enable take-over of the application.

Prerequisites for application take over	
Type of prerequisites	**Requirements**
< Specify type of prerequisites (e.g. training)>	<State which are the training requirements for people who have to operate the application>

Table 28 – Prerequisites for application take over

3.9 Decommissioning requirements

Please state what applications and/or related hardware/software assets need to be decommissioned as a result of introducing this new application

Decommissioning applications	
Application name	**Decommissioned items**

<Specify application to be decommissioned>	<Specify the list of hardware/software related to the decommissioned application>

Table 29 – Decommissioning applications

3.10 Data migration requirements

Please state your data migration requirements as a result of introducing this new application.

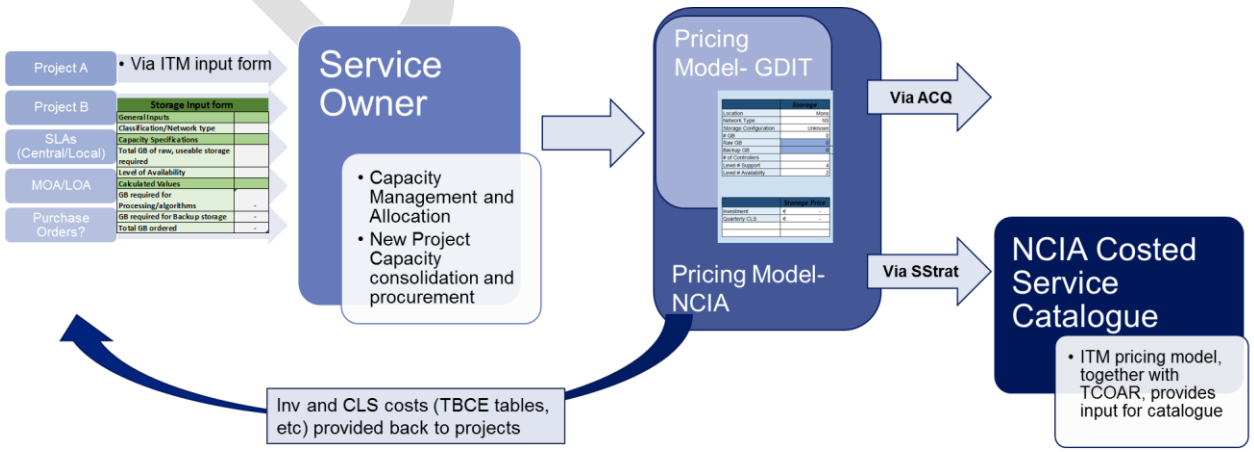
If data shall be physically moved from one site to another, the media used shall be compliant with NATO regulation (EG Encryption requirements for sensitive information).

Data migration					
Data source	**Data target**	**Volumes**	**Type of migration**	**Other requirements**	
<Specify the source of data to be migrated>	<Specify the target of data e.g (node, application component, type of storage>	<Specify the volume of the data to be migrated>	<Specify the modality used for migration (e.g. FTP, ETL)>	<Specify any other requirements)>	

Table 30 – Data migration

3.11 TBCE Costing requirements for additional capacity

Service Owners who require funding for their project to develop a new application which will be hosted on ON and/or PBN, need to complete the **Additional Capacity form**, **located here**. This form is the first step in acquiring funding for their project, as illustrated by the overall Process to Procure Additional Capacity in the figure below. As the cost tables in the TBCE need to be delivered by the Cost Estimating & Analysis Group, part of Acquisition, it is important for the Service Owner to cooperate with them when filling in the Additional Capacity form. The Point of Contact is Mr. Pieter Jansen: pieter.jansen@ncia.nato.int. Please notify him when you start filling in the form.



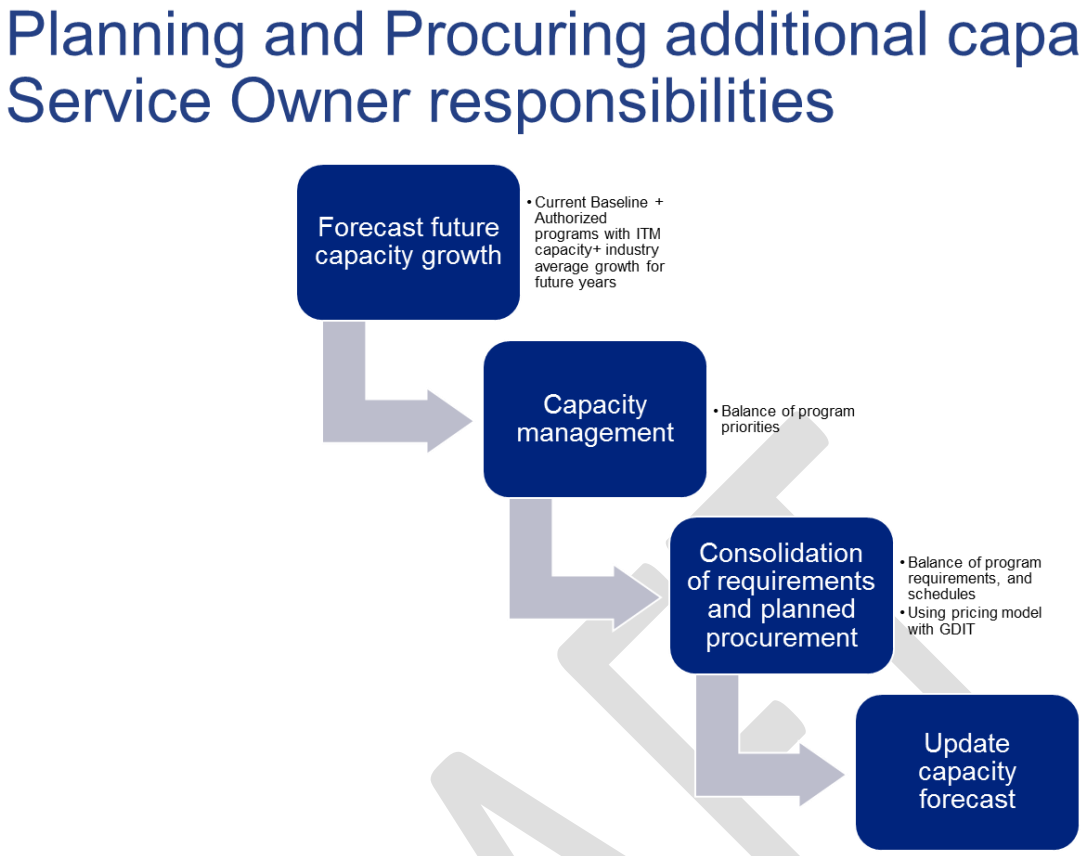


Figure 1 – Process to Procure Additional Capacity and Service Owner Responsibilities

The exact fields that the Service Owner will need to complete in this Additional Capacity Form are highlighted in yellow in the figures below, and include general; processing; and storage input fields. Please note, these fields should only be completed through the link above.

	Input Fields
General Inputs	
Program Name	
Program Manager/POC	
Required Delivery by (date)	
Location of capability (end user)	

Figure 2 – Additional Capacity Form fields under General Inputs tab

Processing Input form #1	
General Inputs	
Classification/Network type	
Location of physical servers	
Capacity Specifications	
Server Configuration	
Enter capacity in only 1 of 2 ways:	
# Physical CPU Cores	
# Virtual CPU Cores	
Level of Availability	
Calculated Values	
Physical Servers	-
Capacity of RAM (GB)	-
# O/S Instances	-

Figure 3 – Additional Capacity Form fields under IaaS Processing tab

Storage Input form	
General Inputs	
Classification/Network type	
Location of physical storage	
Capacity Specifications	
Total GB of raw, useable storage required	
Level of Availability	
Calculated Values	
GB required for Processing/algorithms	-
GB required for Backup storage	-
Total GB ordered	-

Figure 4 – Additional Capacity Form fields under IaaS-Storage tab

4 ON and PBN Hosting Guidelines for new NATO cloud-capable applications

This section contains the Hosting Guidelines (HGs) for new cloud-capable applications to be hosted and supported in the ON and/or PBN networked environments.

4.1 Application Hosting Guidelines

The following section contains the Hosting Guidelines (in the red cells) that each application needs to abide by, in order to be hosted successfully on the ON and/or PBN networked environments.

Section 8 provides further best-practice guidelines in support of these HGs.

4.1.1 Hardware and Software related HGs

The following table illustrates the hardware and software related Hosting Guidelines (HG) an application must abide by in order to be hosted and operated successfully on the ITM infrastructure. Note that the following HGs are for new applications, while for existing applications to be migrated on ITM infrastructure some exceptions could be acceptable.

Hardware and software related HGs	
Hardware Architecture	■ **HG1:** The application shall be compatible with x86-64 architecture (32 bit applications are supported). Other architectures like ARM, SPARC, MIPS and PowerPC are not supported.
Server Operating System	■ **HG2:** The server-side of the application shall be able to run on one of the operating systems specified in the Baseline Configurations table in Annex C.
Client Operating system	■ **HG3:** The client-side of the application shall be compatible with the client operating systems specified in the Baseline Configurations table in Annex C.
Client and Mobile Platform Baseline compatibility	■ **HG4:** The application shall be tested against the NATO client and mobile platforms and shall be compatible with these baselines. The NATO client and mobile platform baselines include all the software specified in the Baseline Configurations table in Annex C. If the application requires any add-ons/plugins/java VMs or other platform enhancements, these shall be separately identified as Software Items in the NCI Agency's A&T catalogue and the dependency to these Items shall be modelled.
Administrator privileges	■ **HG5:** The application shall be able to be run on a hardened and highly controlled environment without requiring elevated enterprise level administrator privileges.
Mobile operating system	■ **HG6:** The application shall support at least Android, Apple iOS and Windows 10 Mobile operating systems. The application shall be written independently from the specific operating system. It shall be developed to adapt its user interface to devices like smartphones and tablets with small screen size and different resolution types. Mobile applications shall be written to leverage touchscreen device capabilities.
IP protocol	■ **HG7:** The application shall support the IPv6 protocol. The application should only use the IPv4 protocol where it needs to access an existing application. IPv4 IPv6 in a dual stack configuration is only allowed for existing applications.

Database	■ **HG8:** The application shall support MS SQL Server 2012R2 or later, if it requires database management services.
Application Server	■ **HG9:** The application shall support Orion IPAM IPX, MS IIS, Tomcat and any other AS compatible with the ITM environment. [P94 guidance placeholder].
Middleware	■ **HG10:** The application shall support middleware compatible with the ITM environment, as stated in [add DAS guidelines/policy doc].
Hypervisor	■ **HG11:** The application's server deployment package (virtual appliance or installation package) shall be compatible with the hypervisor specified in the Baseline Configurations table in Annex C.
SW installation package	■ **HG12:** The software installation package of the application shall be compatible with the packages specified in the Baseline Configurations table in Annex C, and shall be installed only by authorised users.
Configurable application	■ **HG13:**The application software shall not have: - Any hard coded URL, DNS or IP Address settings. - Any hard coded UNC, File Path, Drive Letter or similar storage location settings. All URL, DNS, IP Addressing and similar network and storage settings of the application shall be parametric, configurable, and able to be automated for unattended installation, backup, and recovery.
Storage independence	■ **HG14:** The application shall not have any direct dependency on the physical parameters of the storage environment (such as disk type, connection type, SAN topology, SAN protocol).
ITM Infrastructure requirements	■ **HG15:** The Application Owner shall specify the infrastructure requirements of the application (e.g. computing, network, storage, backup up, archiving, scalability, availability, hosting) in terms of the ITM infrastructure services usage, as per the instructions in the 'Application requirements for NCIA IaaS hosting' section of this document.
Performance	■ **HG16:** The Application Owner shall leverage the ITM distributed and virtualised architecture to design a scalable application both vertically and horizontally, and shall test the performance of the application in order to manage the forecasted increase in users and data.
Service Management and Control	■ **HG17:** The application shall integrate with the ITM SMC service supporting at least the following capabilities: - Logging events exportable to Microsoft SCOM, to enable tracking and monitoring of: user access, application performance, availability of services and component, errors, integrity checks - Remote start and stop of services and components - Migrate application to a different node - Exclude logged users from the application - Send meaningful error and notification messages to the users

	- Health check procedures and dashboards based on MS SCOM (System Centre Operation Manager).
Software provisioning	■ **HG18:** The application's provisioning capability shall be compatible with the ITM software provisioning service and provides the following capabilities: - Proper versioning and naming of software components - Software deployment SMC mechanisms as stated in Annex D to install software packages and patches in unattended mode - Scripts to fully uninstall software packages in unattended mode.
Software versioning	■ **HG19:** The application shall adopt a coherent naming and versioning mechanism as per Technical instruction of NCIA 06.03.01 "Identification of Software Assets" [1], in particular Annex A of that document.

4.1.2 Application Design related HGs

The following table illustrates the application design related Hosting Guidelines an application must abide by in order to be hosted and operated successfully on the ITM infrastructure.

Application design related HGs	
Web enabled application	■ **HG20:** The application shall be web-enabled in order to minimise the footprint on the client device, and shall avoid the use of plug-ins and runtime environments (e.g. Flash plug-in, Silverlight). The use of HTML5 and AJAX is strongly recommended.
Multi-browser support	■ **HG21:** The application shall support multiple browsers (as a minimum Internet Explorer (latest approved version at time of deployment), and FireFox (latest approved version at time of deployment).
Responsive design UI	■ **HG22:** The application shall be accessible from different mobile devices (running the Android, iOS or Windows Mobile OS), with a screen width between 320 and 1024 pixels without the need to refactor its User Interface.
Multilanguage	■ **HG23:** The application shall have multilingual capabilities and be UNICODE compliant.
Web oriented architecture	■ **HG24:** The Application shall act as a provider and consumer of lightweight web services. The Application Owner should thus consider partitioning the application and its capabilities into more granular components that can be implemented, tested, and scaled separately.
Highly decoupled	■ **HG25:** The application shall be able to interact with other ON/PBN-hosted applications and services via published and versioned APIs. APIs are the building blocks and leverage points of ON/PBN-hosted cloud applications. Deconstructing the application into separate web service APIs enables easy re-use and publishing of the application's capabilities for integration into other cloud apps, while also

[1] Available here: https://reccen.nr.ncia/Registry/NCIARECCEN-4-111498.pdf

	enabling appropriate scaling of each individual service. Additionally, well-designed APIs will shield the application from underlying technology implementations as well as vendor-specific implementations.
Configurability	■ **HG26:** The application shall provide a clear distinction between configuration components and execution components by putting all configuration files in a folder called “Configuration”. The overall software design should be based on the principle of minimum customisation and maximum configuration. There should not be any hard-coding in any aspect of the development and release lifecycle of the proposed application. This will allow the application to scale-up in run-time to handle more simultaneous service requests, while at the same time allow the administrators to manage configurations for multiple tenants. NCIA should be able to, for example, configure the same application for NATO and for a Nation by just changing the configuration files.
Multi-tenancy	■ **HG27:** The application shall be configurable to allow multiple tenants to share the same instance of the application with all data and user accounts separated from each other.
Abstracted user authentication	■ **HG28:** To allow for Single Sign On and Federation, the application shall rely on Claims-Based Authentication, specifically the application shall support Security Assertion Markup Language (SAML) v1.1 or v2.0 (preferred), and shall implement strong authentication when required.

4.1.3 Architecture related HGs

The following table illustrates the architecture related Hosting Guidelines an application must abide by in order to be hosted and operated successfully on the ITM infrastructure.

Architecture related HGs	
Latency-aware	■ **HG29:** Latency refers to the round-trip time for a request, specifically, the time for the initial packet to reach its destination, for the destination machine to reply, and for that reply to reach the requestor. The application shall be capable of dealing with variable extended communications latencies ensuring service continuity to the users. Latency is defined as follows: - RTT (Round-trip time) < 250 ms = normal communications latency - RTT (Round-trip time) > 250 ms = extended communications latency ITM hardware (and possibly other layers of the infrastructure stack) is shared and virtual. This impacts response times for all its tenants.

	The application shall make use of caching mechanisms, lazy loading and light-weight previewing to improve user performance and minimise bandwidth utilisation. The application shall minimize the service chattiness to avoid communication overhead. The application must tolerate the effects of deduplication, acceleration and other network optimization features. The application shall use data compression to reduce the bandwidth usage.

Instrumentation	■ **HG30:** The application shall be easy to maintain, operate and troubleshoot. For this reason it shall be properly instrumented[1] so that it can log and analyse at least the following data:
	
	- Operational and runtime events that occur as part of the normal operation of the application, together with useful information about that event, such as the location or data store used and the response time for access to the data store. The event should not include any sensitive information such as credentials, or any other data that might enable an attacker to obtain the logs to compromise the system.
	
	- Auditable events associated with individual user identities including date and time of the event, type of event, user identity, and the outcome (success or failure) of the event:
	  - system start-up and shutdown
	  - log-on and log-off of individual Users
	  - changes to permissions and privileges of users and groups
	  - changes to security relevant system management function
	  - all updates of access rights
	
	- Specific data about errors that occur at runtime, such as the customer ID and other values associated with an order update operation that failed. Typically these are warning or error events and will contain one or more system-generated error messages.
	
	- Data from performance counters that measures specific values related to the operation of the application. These might be built-in system counters, such as those that measure processor load and network usage, or they might be custom performance counters that measure the number of orders placed or the average response time of a specific component.
	
	The application shall not attempt to write to or manage log files. Instead, each running process shall write its event stream. During local development, the developer will view this stream in the foreground of their terminal to observe the application's behaviour. In staging or production deploys, each process stream will be captured by the ITM execution environment (Service Management and Control service), collated together

[1] For more information about instrumentation, please refer to section 8.1.7.1.

	with all other streams from the application, and routed to one or more final destinations for viewing and long-term archival.
Failure-resilient and graceful degradation	■ **HG31:** The application shall be able to survive failures (hardware, software, network and human, such as WAN out, 1 datacentre down, 2 datacentres down) in the ITM distributed environment as per the attributes specified in Sections 4.1.4 and 4.1.5. When services become unavailable, the focus shall be on the user experience by granting a graceful degradation and recovery. In order to survive failure the application shall adopt the following resiliency mechanisms: - It shall notify the users with appropriate error-messages - It shall not contain single-point of failure - It shall provide a Mean Time to Repair (MTTR) of 4 hours or less. - It shall use mechanisms to recover immediately as soon as the failed service will return available - In case of node or instance failure it shall continue to function within and between the remaining active nodes rerouting the user traffic to the remaining active nodes or instances and if necessary it shall be capable of switching to a fail-over server (see HG39 and HG40). - It shall support fail-over in its architecture. Active clients shall be able to survive failure of application server without crashing or affecting the stability of the system - In case of LAN failure it shall continue to perform all non-networked functions - In case of WAN data centre failure it shall fail over the active clients on the enhanced node instances - It shall not cause data corruption or loss of data integrity due to connection failure or other hardware/software failures - It shall gracefully degrade in the condition where the infrastructure Services (e.g. core services, network services, directory services) are not available. - It shall queue all requests to an unavailable Core Service and deliver them when the service becomes available again. A graceful degradation means all application functions unrelated to the unavailable services shall continue to work. The application shall not raise exceptions, or even crash, but shall elegantly notify the users and if possible shall continue to provide some functions using local cache.
Event Driven enabled	■ **HG32:** The application shall be ready to react and properly behave to asynchronous events/warnings coming from the ON/PBN infrastructure: - technical failures (i.e. crashing of system components) - over-consumption of resources (e.g. RAM, CPU) - resource capacity limitation - out of disk/table space - system shutdown and reboot

	- low network bandwidth - core service unavailability - replication failure - security events and violations
Parallelizable (horizontally scalable)	■ **HG33:** The application shall be able to scale horizontally – that is, parallelize data and processing within the application, in order to best leverage the distributed, elastic and scalable ON/PBN infrastructure within the datacentres. The application shall be designed to leverage the following parallel processing mechanisms: - simultaneous execution of the same task on multiple processors (multithreading or multiprocessing) - workload decomposition and parallel distribution of different component tasks across multiple instances or nodes [1] (multi-instances and multi-node) - parallel distribution of data across multiple instances or nodes (multi-instances and multi-node). Each instance/node works on a subset of the data in parallel. Partitioning a database horizontally is known as "sharding." Shards can be distributed so the system maintains performance as load increases, and extra nodes can be added, maintaining scalability.
Automated and virtualisation-friendly	■ **HG34:** The application shall be resilient to transient infrastructure and shall be virtualisation-friendly: - The application shall be able to adapt immediately to changes in resource capacity due to changing priorities (e.g. shrinking RAM) using the IIS mechanism. - The application shall be able to add automatically a new instance without shutting down. - All infrastructure states for an application environment must be re-creatable automatically from templates that describe how the server instances need to be configured and updated with data.
Resource-consumption aware	■ **HG35:** The application shall be consumption aware to minimise consumption of the CPU, memory, network input/output (I/O) and storage I/O, thus enabling more users, transactions or requests to be packed into a dynamic, shared and virtual infrastructure. The footprint of each session of the application shall be predictable.
Portable	■ **HG36a:** The application shall minimise the dependencies with specific hardware and software components ensuring: - Portability at least within the same operating system environment

[1] This is not to be confused with 'ITM node'. Here we refer to a 'node' in relation to parallel processing only.

	- Ability to run on diverse hardware platforms (maintaining the same OS type)
Backing Services Independence	■ **HG36b:** The application shall minimise the dependencies on specific DBMS products, messaging services, caching API, using an abstract data access service layer to decouple the application business logic from the data persistence layer.
Interoperable	■ **HG37:** The application shall expose functionalities through clearly defined API interfaces, as defined in [placeholder for P94 guidance] and must be able to publish its API onto the tool specified in the Baseline Configurations table in Annex C.
Monitorable	■ **HG38:** The application shall enable the ON/PBN infrastructure to run scheduled or on-demand health check procedures to understand the status of the application. For this purpose the application shall provide a file with KPIs, metrics and thresholds (e.g. time responses, latency, service unavailability) for each component to understand how it behaves once in production and if corrective actions are required.
Automatically deployable	■ **HG39:** If the application is client-based, it shall be automatically deployable. For backed applications, they should employ as much automation as possible for their deployment. - Application deployment must be automated (for example, include no manual steps and be wrapped in a template) so that the services can be moved and quickly restarted on alternative servers. - Patches and upgrades to running software packages must be deployed and installed without service interruption.

4.1.4 Load balancing and high availability related HGs

The following table illustrates the load balancing and high availability related Hosting Guidelines an application must abide by in order to be hosted and operated successfully on the ITM infrastructure.

Load balancing and high availability related HGs	
Cloud multiple node deployment	■ **HG40:** The application shall allow distribution of its instances across the ITM nodes. - A minimal configuration for high availability should consider two instances deployed on separate Data centres and one instance deployed on the Enhanced node. For fail over configuration refer to HG41.
Load balancing (workload distribution)	■ **HG41:** The ITM infrastructure utilises load balancing to distribute workloads across multiple servers, network links, data storage and sites. This is done to achieve optimal utilisation of processing, storage and networking resources, while maximising throughput and minimising response time. The Load-balancing component provides: - Local load-balancing – between resources in the Datacentre; - Global load-balancing – between Datacentres (for high availability and load sharing purposes)

	The application shall use ITM global load balancing services, in order to scale and support increasing volumes. The application shall expose a unique URL to the users/consumers and the ITM global load balancer will take care to dispatch traffic to available active instances distributed on different nodes. The application shall specify load balancing requirements in terms of: instances/nodes, protocols, transactions per second, and dispatching scheme.
Fail over (active/active and active/passive design)	■ **HG42:** The application shall be able to survive a node or network failure. - The application shall support active/active design local to a datacentre and across datacentre nodes (with replication of session and states for stateful components). In this case both locations are running simultaneously, handling different users and ready to fail over to each other should it become necessary. - The application shall put enhanced node instances only in passive mode and activate them in case of a network outage with the datacentres. For stateful components a replication mechanism of application sessions and states must be put in place. - Active instances in enhanced nodes are potentially possible, and must be evaluated case by case by NCIA, only for read-only massive data applications, like core GEO services, in order to reduce network occupation. Maps loading must occur anyway in the data centre and then be replicated on the enhanced node.

4.1.5 Backup and recovery related HGs

The following table illustrates the backup and recovery related Hosting Guidelines an application must abide by in order to be hosted and operated successfully on the ITM infrastructure.

Backup and recovery related HGs	
Back-up and recovery	■ **HG43:** The application shall have a coherent and appropriate backup strategy to restore data in an automated way in case of data corruption events due to unintended user actions or application failures. The application shall provide the ability to perform full and incremental backups (i.e. snapshots) of data and software without impacting system availability. The application shall allow backup of both the complete repository and selected information element (e.g. objects, records, documents). The application shall allow backup of full or selected data to occur automatically at a configurable frequency. The back-up design shall focus both on required RTO and RPO. The application shall minimise any file system scanning during VM backup.

	The application shall use agentless image-based backups with incremental-forever using changed block tracking (CBT). The application shall use offline indexing of image backups for those systems that need to be indexed at the file level. For Virtual machine RPOs of 4 hours to 12 hours, the application shall use VM image software-based snapshots. For VM RPOs of 1 to 4 hours, the application shall use hardware storage array snapshots as a complement to software-based VM backups, not as a replacement for them. The application shall use offload-host VM backup for backup jobs that create larger I/O load and can impact the correct functioning of the production system for more than 5 minutes. At the time of deployment the application shall segregate (using virtual disks) the virtual machine file into operating system file, application file and data files to eliminate unnecessary data from the image backup and to improve system recovery. An image of each VM should be secured and tested at regular intervals and each time a change is made to the VM's guest OS or applications (for example, after patch or new software installation or other configuration changes). The application developer shall define in advance: - Storage capacity for back up - Type of storage to use - Back up frequency - Type of back up (full or incremental) - Level of information to back up The application shall provide mechanisms to restore software and data lost since last back-up. The application shall allow to select the backup files and the elements to restore. Application deployed on their own hardware shall provide a file for backup purposes to ITM infrastructure on a shared backup directory. Data recovery will be executed starting from the same file provided on request by ITM on a specified recovery directory.
Archiving	■ **HG44:** The application shall support NATO policy-driven functionalities to remove and store data that is no longer required for use from the system, thus reducing the size of the data store and improving performance. The application shall allow archiving of both the complete repository and selected information element (e.g. objects, records, documents). The application data shall possess the minimum level of metadata to support the archiving process. Archived data shall be searchable/readable and the application shall provide mechanisms for restoring it to a specified repository as required.

	Recovery of archives shall allow data to be overwritten or appended as required by the user. The application shall allow parts of archives to be selected and recovered to avoid rewriting a complete dataset.
Disaster-recovery	■ **HG45:** If the application is business critical, it shall have a clear plan for disaster recovery scenarios and exercises, providing RTO and RPO objectives. The plan shall define requirements for: - Type of replication - Volume - Time to replicate. To enable disaster recovery, data replications across multi datacentres is required. In the ITM infrastructure some links between datacentres support both synchronous and asynchronous replication scenarios (latency <= 7ms), other links support only asynchronous scenarios (latency > 40ms), implying some loss of data in case of a disaster.

4.1.6 Application Security related HGs

The following table illustrates the application security related Hosting Guidelines an application must abide by in order to be hosted and operated successfully on the ITM infrastructure.

Application Security related HGs	
Application security hardening	■ **HG46:** The application shall be able to operate within a hardened operating environment. The application shall be secure by design and developers should expect NATO to test and verify this in some detail. The ITM environment itself comes with a number (50+) of security mechanisms (see Annex A for details). These together with NATO's NCIA Security Configuration Catalogue (Repository on NR: https://nsap.nr.nato/SecSet/default.aspx) will inform developers of the likely hardening measures within the ITM environment. The application shall expect appropriate NCIA scrutiny per network. The NCIA Cyber Security Service Line is responsible for checking application security hardening.
Application security accreditation	**HG47:** NATO SAA (NSAB for NATO-wide applications) shall decide whether security accreditation is required for the application or whether following the Request for Change (RFC) process is sufficient. NATO standards on secure application development must be followed in detail. Depending on the categorization assigned by NATO SAA[1], the "application" will need to undergo one of the following security accreditation procedures:

[1] Possible types are: service, system or application. Definitions of each are available in the NSAB Reference List of Systems, Applications and Services assignment document (NR) - Extract from the NSAB Decision Sheet DS(2010)0001, 8 March 2010.

	- "system" - Formal security accreditation process with full set of documents (e.g. CIS description, SAP, SRA, SSRS, SecOPs, STVP, STVR); - "sub-system" - Simplified security accreditation process with simplified documents (e.g. SRA and "delta" to ITM CSRS, annexes to ITM SecOPs); - "application" - Standard RFC process Application developers who are familiar with NATO and this process will notice some changes. For those who have not worked within NATO before, or who are developing complex capability may wish to consider the appointment of a Point of Contact (PoC) responsible for dealing with security accreditation, or NATO-approved Security Assurance Coordinators (SAC) to facilitate their passage through the accreditation process.
Secure coding	■ **HG48:** The application shall be coded securely. The application shall support security dependencies upgrades (i.e. Java, .net, SQL Express, etc.) following IT security best practice. The adequacy of application security measures will be verified by NATO. In order to reduce the risk of significant software re-writes, developers are advised to incorporate best practices from the public domain into their software design lifecycles. Publically available resources that may be of use include: - The Open Web Application Security Project (OWASP) o https://www.owasp.org/index.php/Main_Page - The SANS Institute o https://software-security.sans.org/resources o https://www.sans.org/reading-room o https://www.sans.org - Software Assurance forum and Excellence in Code (SAFECode) o http://www.safecode.org/ o http://www.safecode.org/publication/SAFECode_Dev_Practices0211.pdf In addition to the above, developers are also advised to consult the following NATO Service Interface Profiles (SIPs): o INSTR TECH 06.02.01 SIP for Security Services, available here (NU). o INSTR TECH 06.02.03 SIP Security Token Services, available here (NU).
ITM security mechanisms	■ **HG49:** The application shall meet the ITM security requirements mentioned in Table A below. Some security mechanisms are applicable to certain applications only. For an explanation of each security mechanism, please refer to Annex A.

Table A – ITM Security mechanisms

SM nr.	SM Name	Security requirements
SM01a	Malware Protection for Server (e.g. AV for servers)	

SM01b	Malware Protection for Application Server (e.g. SharePoint)	The application shall be compatible with multiple AV instances across the ON/PBN environment.
SM01c	Malware Protection for Multifunction Printing (MFP) device	
SM01d	Malware Protection for Server Database	
SM02a	Malware Protection for Client (e.g. AV for clients)	
SM02b	Second but different Malware Protection for Client (e.g. AV for clients)	
SM03	Malware Protection for handheld devices (e.g. smartphones)	
SM04	Malware Protection for Email server (e.g. AV for email services)	
SM05	Malware Protection for Web server (e.g. AV for Web Services)	
SM06	Messaging Content Filtering (MCF)	The application shall be compatible with and provide rulesets for MCF where applicable.
SM07a	Web Content Filtering (WCF) - Categorization	The application shall be compatible with and provide datasets for WCF where applicable.
SM07b	Web Content Filtering (WCF) - Content Inspection	
SM07c	Web Content Filtering (WCF) - SSL Intercept	
SM08	HTTP AV Proxy	The application shall be compatible with AV proxies across the ON/PBN environment.
SM09	FTP AV Proxy	
SM10	Integrity Checker	The application shall be compatible with integrity checks.
SM11	SSL (TLS) and SSH	The application shall be compatible with SSL/TLS and VPN.
SM12	Strong Authentication (User Token)	The application shall be compatible with and make use of strong authentication.
SM13	Enterprise Single Sign-On (ESSO)	The application shall be compatible with SSO.
SM14	Firewall (FW) for Outer Perimeter/ Border Protection	The application shall work within NATO perimeter protection constraints
SM15	Firewall (FW) for Inner Perimeter	The application should expect multiple firewalls across NATO systems and shall work with their constraints.
SM16N	Intrusion Detection and Prevention System (IDS/IPS) - Network (N) Based	The application shall work within NATO IDS IPS NIPS NIDS HIPS HIDS constraints. If the application crosses network segments, it shall not introduce loopholes to enable attackers to traverse segments within the application layer.
SM16H	Intrusion Detection and Prevention System (IDS/IPS) - Host (H) Based	
SM17	Network Access Control (NAC) / Network Access Protection (NAP) and Network Access Quarantine (NAQ)	The application, particularly if it is a thick client application, shall incorporate patching that is compatible with NATO NAC/NAP/NAQ such that these

		technologies can remediate out of date instances. The application shall test and implement security patches, and be able to easily support patch and upgrade of all dependencies.
SM18	IP Filtering & Management	The application shall register fixed and dynamic IP requirements with NATO.
SM19	Network/Port Address Translation (NAT/PAT)	*Not applicable*
SM20	(Web) Application Firewall and other proxy/reverse proxy	The application shall comply with OWASP guidelines for the Top 10 security risks (https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project) and does not introduce unmanaged vulnerabilities.
SM21a	System and Security Logging & Auditing - Infrastructure and Servers	The application shall provide extensive logging options to enable automated smart analysis. Developers shall work with NATO to build a picture of what 'normal' looks like to facilitate smart analysis.
SM21b	System and Security Logging & Auditing - Applications	Security guidance and support to be provided directly from NCIRC.
SM22	Hardening of Networked Devices	The application shall be capable of operating within a hardened environment.
SM23	Voice over Internet Protocol (VoIP) protection	If the application utilises or interfaces with VoIP, it shall follow the NATO VoIP security recommendations.
SM24	Wireless Network Protection (and Jamming)	*Not applicable*
SM25	Storage Compartmented Security Mode (SCSM)	If the application processes, creates or stores multiple levels of protectively marked information, it shall comply with the requirements and be capable of operating SCSM.
SM26	Backup, Recovery and Contingency planning	*Not applicable*
SM27	Network Management System (NMS) /Systems Management System (SMS)	The application shall be compatible with application monitoring solutions.
SM28	IT Forensic and Incident Handling	The application shall be compatible with NCIRC provided forensic agents, including full packet capture.
SM29	Cryptographic security	The application shall ensure that data at in storage, in use (particularly passwords) and in transit (particularly passwords) complies with NIAPC requirements based on classification level.
SM30a	NPKI - User certificate	*Not applicable*
SM30b	NPKI - Device certificate	

SM31	Logical Security Zones	The application shall not introduce application layer vulnerabilities to allow unmonitored / uncontrolled lateral movement across NATO security zones.
SM32	Policies, directives, guidance and procedures	The application shall comply with NATO policies, directives, guidance and procedures.
SM33	Load Balancing (LB)/ Failover (FO)	*Not applicable*
SM34	Information Protection Control (IPC)-Classification/Marking	The application shall support meta data labelling in line with NATO labelling and binding standards.
SM35	Information Protection Control (IPC) – Data Loss/Leak Prevention (DLP)	The application shall be compatible with NATO IPC solutions (see above).
SM36	Vulnerability Scanning & Compliancy Check	Application developers shall supply NATO with a verifiable 'good' image of all current and previous production builds. Developers must remediate vulnerabilities detected in production builds within agreed timescales.
SM37	OS and application security settings	The application shall be able to operate with least privilege in this hardened environment.
SM38	Quality of Service (QoS)	The application shall be able to operate effectively within the QOS band into which they fall.
SM39	Development/Test/Pre-production Environments	Application developers shall follow NATO procedures in this regard.
SM40	Configuration Management	*Not applicable*
SM41	CAPTCHA (and its alternatives)	*Not applicable*
SM42	Identity & Access Management (IAM or IdM)	The application shall support NATO IAM requirements.
SM43	Management of the Security Mechanisms	Includes application security
SM44	Time Synchronisation (e.g. NTP)	*Not applicable*
SM45	Storage Areas security	*Not applicable*
SM46	Data Scramble	If the application touches critical data sets, it shall support data scramble.
SM47	Data Safeguarding (e.g. DAM, FSAM)	If the application touches critical data sets, it shall support DAM and FSAM.
SM48	Password Management	If the application does not follow ESSO/SSO, it shall comply with the strong passwords policy within the primary directive on CIS security.
SM49	Identity & Authentication, Access Control (IAAC)	The application shall be compatible with NATO IAAC.
SM50	Protection against (D)DoS	*Not applicable*
SM51	Physical security	*Not applicable*
SM52	Personnel security	*Not applicable*
SM53	Environmental security	*Not applicable*
SM54	Emission security	*Not applicable*
SM55	Data processing	The application shall comply with rule sets as established by the business except where such business objectives would introduce additional risk.

SM56	Data Diode	The application shall operate within this constraint.

5 ANNEX A – ITM Security Mechanisms

The following table explains the security mechanisms that shall be provided by ITM for the ON and PBN environments. All ITM security mechanisms shall be integrated with the NCIRC FOC infrastructure wherever possible.

SM nr.	SM Name	Security Mechanism (SM) Definition
SM01a	Malware Protection for Server (e.g. AV for servers)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM01b	Malware Protection for Application Server (e.g. SharePoint)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. with a particular attention given to the application-related weakness. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM01c	Malware Protection for Multifunction Printing (MFP) device	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. with a particular attention given to the MFP capability and weakness. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM01d	Malware Protection for Server Database	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. with a particular attention given to the database capability and weakness. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM02a	Malware Protection for Client (e.g. AV for clients)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM02b	Second but different Malware Protection for Client (e.g. AV for clients)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM03	Malware Protection for handheld devices (e.g. smartphones)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM04	Malware Protection for Email server (e.g. AV for email services)	Protection against malicious software should include anti-virus (AV), anti-spyware, anti-adware, anti-phishing, etc. In addition, there is a need to implement whitelists, heuristic and Bayesian statistical analysis to compute the probability of message spam, and antivirus filtering to rid of unsolicited email. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.

SM05	Malware Protection for Web server (e.g. AV for Web Services)	Protection against malicious software should include, anti-virus (AV), anti-spyware, anti-adware, anti-spam, anti-phishing, etc. It should be noted that best practices recommend a product differentiation between the malware protection running in the DMZ (i.e. gateways) and the one running in the internal network.
SM06	Messaging Content Filtering (MCF)	Messaging Content Filtering (MCF) is required with enforcing policies to block or allow content based on analysis. The solution is needed to monitor, encrypt, filter, and block content contained in email, instant messaging, peer-to-peer file transfer, web postings, and other types of traffic. This includes the SMTP AV Proxy.
SM07a	Web Content Filtering (WCF) - Categorization	Web Content Filtering (WCF) is required with enforcing policies to block or allow content based on analysis. The solution is needed to monitor, intercept SSL, filter, and block content contained in web-mail, instant messaging, peer-to-peer file transfer, web postings, and other types of web traffic.
SM07b	Web Content Filtering (WCF) - Content Inspection	
SM07c	Web Content Filtering (WCF) - SSL Intercept	
SM08	HTTP AV Proxy	There is a need to scan all incoming traffic (dynamic content, password protected pages, etc.) and data while downloading before this is processed by the applying system.
SM09	FTP AV Proxy	There is a need for controlling and examining bulk transfer of files thus to detect malware before transferring infection into or from system.
SM10	Integrity Checker	Files are to be checked for change and to ensure that applications have not been tampered (i.e. automated checksum). This also refers to ensure that required Logs have been archived and protected against modification (i.e. signed). Furthermore, data at rest including electronic information published on publicly available systems requires data integrity protection. Equally data in transit also needs data integrity protection especially that, if altered, would put at risk the CIS and its management, or cause serious damage to business on-line transactions.
SM11	SSL (TLS) and SSH	This is needed for managing the secure communication, data and message transmission within and with other NATO bodies, Remote Offices and Mobile Users (i.e. using a security protocol found in all standard Web browsers). Furthermore also Secure Shell (SSH) i.e. used for the management of the ICT environment (e.g. routers, switches, linux servers, etc.) is covered by this SM. This security mechanism it employs also an approved (or authorized) public-and-private key for encryption and authentication system.
SM12	Strong Authentication (User Token)	Strong Authentication: The Strong authentication mechanism provides a layered authentication approach relying on two or more authenticators to establish the identity of an originator or receiver of information. In addition, you may have what a user is (biometrics) such as a fingerprint, a retina pattern, a voice pattern, or a behaviour pattern such as that exhibited in using a keyboard (Characteristic). Source: AC/322-D(2005)0044 Any request for accessing to network resources needs to go through and satisfy the Authentication, Authorization, Accounting (AAA) and Auditing requirement. In addition, it needs to provide

		identity and traceability of users and systems. This includes the Remote Authentication Dial-In User Service (RADIUS), which is the current standard by which devices or application communicate with an AAA system. In fact, all requests for accessing the network have to be audited. This security mechanism may also be used to protect passwords in storage as well as in transit, in accordance with security policies and best practices, and in combination with other security mechanisms (e.g. SM13) when required. Following are the two well-known strong authentication mechanisms: One-Time Password (OTP), sometimes also called Medium Authentication: Traditionally static passwords can more easily be accessed by an unauthorized intruder given enough attempts and time. By constantly altering the password, as is done with a one-time password, this risk can be greatly reduced. Two-Factor Authentication (T-FA): What a user possesses such as a token. Examples of tokens might include swipe cards, smart cards, mechanical and electronic keys (Possession). An example of T-FA is a USB Token, a Debt-Card, etc.
SM13	Enterprise Single Sign-On (ESSO)	This is to be used to manage user credentials, authentication and auditing in conjunction with medium or strong authentication devices (e.g. combined with strong authentication methods like smart cards) and identity provisioning systems to achieve provisioning and easier identity mapping.
SM14	Firewall (FW) for Outer Perimeter/ Border Protection	Firewall may be hardware and/or software and is needed to enforce security policies, secure the network, shielding it from unauthorized users. In addition to preventing unrestricted access into the network, firewall needs to also restrict data from flowing out of the network. This SM often provides sandbox-based analysis to look over malicious behaviours.
SM15	Firewall (FW) for Inner Perimeter	Firewall may be hardware and/or software and is needed to enforce security policies, secure the network, shielding it from unauthorized users. In addition to preventing unrestricted access into the network, firewall needs to also restrict data from flowing out of the network. The so-called "personal firewall" most of the time deployed on clients (e.g. on REACH laptop) is also part of the Inner Perimeter.
SM16N	Intrusion Detection and Prevention System (IDS/IPS) - Network (N) Based	It may be software and/or hardware designed to detect/prevent unwanted attempts at accessing, manipulating, and/or disabling computer systems and network devices. It needs to detect/prevent attacks against vulnerable services, data driven attacks and applications, denial of service (DoS).
SM16H	Intrusion Detection and Prevention System (IDS/IPS) - Host (H) Based	It is software designed to detect/prevent unwanted attempts at accessing, manipulating, and/or disabling computer systems and server and clients. It needs to detect/prevent attacks against vulnerable services, data driven attacks and applications, denial of service (DoS), host based attacks such as privilege escalation, unauthorized logins and access to sensitive files.

SM17	Network Access Control (NAC) / Network Access Protection (NAP) and Network Access Quarantine (NAQ)	This is needed to bolstering the security of the network by restricting the accessibility and availability of network resources to endpoint devices, whether on the campus or outside it, that comply with a defined security policy. Users and/or endpoint devices must authenticate themselves before getting access to the network, even if they are roaming. The authentication methods may vary depending on the device and connectivity types (i.e. hard-coded address, challenge-response, cryptographic methods and others. The network can then make further policy decisions based upon user and device identity characteristics. Before gaining network access, endpoint devices must be checked for vulnerabilities, security software configuration parameters (e.g. whether antivirus signatures are current), and malicious code signatures. Further network decisions are based upon the results of this examination. For instance, a device that is not compliant with a defined security policy has to be isolated/quarantined and undergo through a remediation and sanitization process. Only when successful, the device may be granted access to the network and in accordance with its identity characteristics. In fact, the solution has to be centrally managed and configured to limit a device to specific network assets or tasks in accordance with policies and business requirements. For example, an IP phone may be restricted to a particular network VLAN and IP telephony gateway.
SM18	IP Filtering & Management	IP filtering is needed to filter traffic based and deep packet inspection pattern matching. This mechanism is needed to decide which type of IP datagram will be processed normally and which will be discarded. By discarded is intended that datagram is deleted and completely ignored, as it had never been received.
SM19	Network/Port Address Translation (NAT/PAT)	It needs to be used NAT/PAT to connect multiple devices to Internet by only using one public IP address. NAT/PAT traversal should be considered because of Voice over IP (VoIP) protocol enabled devices.
SM20	(Web) Application Firewall and other proxy/reverse proxy	(Web) Application firewall is required to further secure applications during the delivery process. It needs to limit the access which software applications have to the operating system services, and consequently to the internal hardware resources found in a computer. In addition, a dedicated Web Application Firewall (WAF) is needed to apply a set of rules to an HTTP conversation. This is often used to also provide additional proxy and reverse proxy for the other services (e.g. DNS), SSL Termination, Web Reverse Proxy, etc.
SM21a	System and Security Logging & Auditing - Infrastructure and Servers	Logging and auditing requirements are to be satisfied on each network component and event messages and alerts transmitted over the network, and centrally managed and stored. The standardization of the event log format and the aggregation and correlation of logs is recommended as to enable an “automated smart analysis” of the logging also when combining logs coming from multiple systems. A full control over the administrative channels is required as to enforce global regulations, irrespectively of the User Types (or

		Categories/Roles), as to monitor, log and audit the whole network infrastructure in a transparent and integral way. This will allow collecting reliable information concerning the security monitoring, logging and auditing of any activity concerning ICT services, systems and their components.
SM21b	System and Security Logging & Auditing - Applications	Logging and auditing requirements are to be satisfied application and event messages, to be centrally managed and stored. The standardization of the event log format and the correlation of logs is recommended as to enable an “automated smart analysis” of the logging also when combining logs coming from multiple systems.
SM22	Hardening of Networked Devices	Network devices (i.e. routers, switches) can be considered the very first line of defence and also the first line of attack. It provides packet routing and can also be configured to block or filter the forwarding of packet types that are known to be vulnerable or used maliciously. Switches are used to link physical segments of a network together but to also separate networks segments with Virtual LAN (VLAN), thereby reducing the spread of broadcast traffic and applying access control based on security rules. Using access controls between VLANs restrict the flow traffic between different segments of the network and provides a level of protection by blocking internal intrusion. In addition to firewall filtering, this raises the layer of security. Though more expensive, Switch Layer 3 is strongly recommended for several reasons: increased scalability, avoid slow convergence problem with Spanning Tree Protocol, enable load-balancing across multiple paths, enhance redundancy/resiliency, enhance security functions, etc.
SM23	Voice over Internet Protocol (VoIP) protection	VoIP attacks are becoming malicious as much as what we are now witnessing with other IP-based communications. Internal communication: VoIP network that runs in a converged environment needs to be separated at an abstract level (e.g. using VPN or VLAN); External communication: Ensure level of authentication and encryption to maintain integrity of communications (e.g. using VPN). Make sure the network and security devices such as IPS/IDS, firewall, etc. use technology that is “VoIP aware”. Have an IPS deployed between VoIP gateways and near the call manager. Should the IP signalling (e.g. VoIP) be widely used, this was deemed to be augmented by a BPD like a Session Border Controller (SBC), thus to exert great influence over the data flows of sessions. Additional reference: NIATC, NCIRC-TC security recommendation on IP Telephony systems architecture, deployment and configuration, dated 27 September 2010.
SM24	Wireless Network Protection (and Jamming)	The use of Wireless needs to also take additional security precautions. First of all, IEEE 802.1X, 802.11i and 802.11w standards is to be applied. Secure communications, which entails both encryption and data authentication, has to be mutually established between client and network. Finally,

		pertaining Security Policy and Supporting Directives are to be followed in accordance with the security classification level agreed with the business. Depending on the type of traffic supported over the wireless infrastructure (i.e. NATO business), wireless Intrusion Detection/Prevention and Air Monitor should be considered, inn addition of Jammers to inhibit or halt the transmission of signals, when appropriate and in line with the business requirements. Additional reference: INFOSEC Technical and Implementation Guidance on Wireless LAN Security, AC/322-D(2005)0049.
SM25	Storage Compartmented Security Mode (SCSM)	The purpose of SCSM is to split the information up into separate virtual compartments or security levels. Storage will use secure compartmented data access as information of different classification level is to be stored using compartmentalized access and access control over trusted and untrusted network. To create secure compartments, storage area will use trusted virtualization, application-level firewalls and other security devices that may be used for this purpose.
SM26	Backup, Recovery and Contingency planning	Proven, cost-effective and robust backup, replication and recovery solution is needed. The solution has to consider the centralized management of DBRR of the entire network and remote office(s) securely, easily and in timely manner, according to predefined business requirement and criticality. On-site and off-site backup/replication storage needs to be properly implemented and handled, including data encryption and dedicated/redundant bandwidth. As far as the Business Continuity Planning (BCP) and Contingency Planning are concerned, they need to be distinguished between "Local" (i.e. within the site) and "Non local" (i.e. across sites).
SM27	Network Management System (NMS) /Systems Management System (SMS)	A Network Management System, as a combination of hardware and software, is needed to provide activity monitoring, network utilization, controlling, auditing, assessing performance of devices, managing, configuring and administering portions or the entire ICT infrastructure. It needs to provide discovery capabilities, inventory, configuration and mapping dependency – mainly up to an including layer 3 of the OSI Reference Model. A Systems Management System (SMS) is required to provide systems integration as to orchestrate all systems providing services to the business, to coordinate the execution of multiple technical services and systems as to make them appearing as a “single” entity to service request. It needs to provide application monitoring, capacity monitoring, storage management, security management, auditing, assessing performance of systems and applications, managing, configuring and administering portions or all systems. Additionally, it shall provide discovery capabilities, inventory, configuration and mapping dependency – mainly from layer 4 and up to an including layer 7 of the OSI Reference Model. The SMS needs to integrate and/or interoperate with the NMS.
SM28	IT Forensic and Incident Handling	The IT Forensic is pertaining to legal evidence found in computers and digital artefact. Digital artefact includes computer system, storage media, electronic document, sequence of

		computer packets, etc. This also helps with achieving regulatory compliance and response capabilities to IT incidents. The IT Forensic security mechanism is needed to assure that the evidence are accurately collected and that there is a clear chain of custody from the scene of the crime to the investigator, and ultimately to the concerning Authority. This SM includes the Incident Handling and Response in accordance with AC/322-D(2008)0042-NR-Cyber Defence Management Authority (CDMA) Concept of Operations (CONOPS).
SM29	Cryptographic security	This is needed to cipher-text data-info that cannot be understood by unauthorized people, and so protect data confidentiality and integrity. This will be used to encrypt data at rest (e.g. for mobile user), in combination with SSL/TSL and VPN tunnelling. Depending of the classification level a different type of encryption/decryption mechanisms may be required (reference AC/322-D/0047 "INFOSEC Technical and Implementation Directive on Cryptographic Security and Cryptographic Mechanisms"). For this security mechanism, only products selected through the directive on the “Selection and Procurement of NATO Cryptographic Systems, Products and Mechanisms” (reference AC/322-D(2006)0041) and “NATO Off-Line Cryptographic Equipment (NOLCE)” (reference AC/322(SC/4)N(2008)0006) may be used , unless differently stated in other documents or because of specific weaver received from the competent authorities. The maintenance of crypto equipment is part of this SM. This security mechanism may also be used to protect passwords in storage as well as in transit, in accordance with security policies and best practices, and in combination with other security mechanisms (e.g. SM13) when required.
SM30a	NPKI - User certificate	There is a need to enable users, i.e. unsecured public network such as the Internet; to securely and privately exchange data through the use of a public and private cryptographic key pair issued by a NATO accredited Certification Authority (CA). Depending on the business requirements, this SM may also needs to provide the "Non-repudiation “as proof for: - Origin of data, - Genuine authentication with high assurance. Public Key Infrastructure (PKI). NPKI Users, in conjunction with SM12.
SM30b	NPKI - Device certificate	There is a need to enable devices i.e. unsecured public network such as the Internet; to securely and privately exchange data (both in transit and at rest) through the use of a public and private cryptographic key pair issued by a NATO accredited Certification Authority (CA). Depending on the business requirements, this SM may also needs to provide the "Non-repudiation “as proof for: - Origin of data, - Genuine authentication with high assurance.

		Public Key Infrastructure (PKI). NPKI Devices, in conjunction with SM17.
SM31	Logical Security Zones	The segregation in Security Zones (e.g. Public Security Zone, NATO UNCLASSIFIED Security Zone, NATO RESTRICTED Security Zone etc.) enhances security as whole. This can be achieved at one or more network layers (mainly at Layer 2 and 3 according to the OSI Reference Model), and using one or a combination of different types of technology as follows, but not limited to: • Virtual Local Area Network (VLAN) • Private Virtual Local Area Network (PVLAN) • Access Control List (ACL) • VLAN Access Control List (VACL) • Multiprotocol Label Switching (MPLS) • Virtual Private Network (VPN) • Virtual Private LAN Service (VPLS) • Virtual Device Context (VDC) • Firewall and IDS/IPS capabilities distinct at each layer • Distinct Firewall leg and port • Distinct Virtual Switch (vSwitch) The implementation of the "Security Zones" is called "Security Zoning" and may also include an air gap between CIS.
SM32	Policies, directives, guidance and procedures	Policies, directives, guidance and procedures are to be **developed** where not existing, or **regularly updated** and published where existing: - Training - Certified products - Hardening guides - Security settings. This includes the CSSL hardening guides, excluding security-related setting for OS and routers/switches since they are covered respectively with SM37 and SM22.
SM33	Load Balancing (LB)/ Failover (FO)	A backup operation that automatically switches to a standby database, server, device or network if the primary system fails or it is temporarily unavailable (e.g. for servicing) is needed for all the core and/or critical components of the IT infrastructure. Depending on the type of service, this should include a distributing processing and communications activity evenly across a computer network so that no single device is overwhelmed.
SM34	Information Protection Control (IPC)- Classification/Marking	This SM tightly linked to so-called Information Protection Control (IPC). This security mechanism is needed to electronically label all IT resources thus to indicate the security classification level that the IT resource is permitted to handle. In addition, all data has to be marked with the classification level in accordance with the need-to-know policy. This SM needs to generate/share classification metadata –via a single or bulk processing- that can be reused by other applications as well as by other security mechanisms, and that can be based on user attributes and roles.

		Depending on the business requirements, this SM may also need to cover the genuineness of data or "Data authenticity".
SM35	Information Protection Control (IPC) – Data Loss/Leak Prevention (DLP)	The Information Protection and Control (IPC) solutions, of which Data Loss/Leak Prevention (DLP) is a key component, is needed as solutions that monitor, encrypt, filter, and block content contained in email, instant messaging, peer-to-peer file transfer, web postings, and other types of traffic and to help preventing unauthorized access to sensitive information – anytime, anywhere. This SM make use also of metadata from other solution/security mechanisms as required (e.g. from classification metadata also generated by other SMs (i.e. SM34), thus to exhaustively preventing data loss/leakage including internal document slipping out. In fact, for effective DLP is intended a tight integration or interoperability of different DLP elements as to work together in order to provide a solid data defence. These elements can be categorized as follows: • Data discovery, classification and fingerprinting • Encryption • Gateway detection and blocking • E-mail integration (e.g. SM34) • Device and media management.
SM36	Vulnerability Scanning & Compliancy Check	A systematic examination of the AIS components and products is needed to determine the adequacy of the security measures, identify security deficiencies, provide data from which to predict the effectiveness of proposed security measures, and confirm the adequacy of such measures after implementation.
SM37	OS and application security settings	A collection of settings that define what a system will look like and how it will behave for a defined group of users. This should be centrally managed and auditable. Policies should allow registry-based changes, access control permissions, security options, software installation and maintenance, script options and folder redirection options. This SM includes the Group Policy Object (GPO) for Windows and non-Windows environments. Source: nww.ncirc.nato.int (over the NS WAN).
SM38	Quality of Service (QoS)	This is mainly an expectation or requirements of service regarding a system. More precisely QoS needs to provide the ability of implementing, monitoring and managing QoS of different priorities to different applications, users or data flows in a network. Most of the time, this SM makes use of the categories defined as follows: • Voice • Video • Important_Data • Routine_Data • Best_Effort_Data Services need to be assigned to the category/rule in accordance to requirements.

SM39	Development/Test/Pre-production Environments	This environment is needed to ensure that any application, system, etc. doesn't go into production prior to having been thoroughly tested, passed the Quality Assurance and successfully evaluated into pre-production. In other words, the pre-production environment is used for the functional final testing before the actual deployment into production, irrespective of whether hardware or software modifications. Complex enterprise application may require specific tests performed within a test environment, either physically or logically separated, from the production environment. However, while the Pre-production Environment is a requirement, the implementation of a Test Environment may be implemented only if the subsequent operation, support & management allow so.
SM40	Configuration Management	Configuration Management, which also includes CMDB(s) as well as the Change Management, it contains all relevant information about the components of the information system in use, which needs to be automatically populated in accordance with NATO policies and processes. This has to include the reference to the "authorized software & hardware lists", to be maintained by all entities and sites subject to the security accreditation. This SM also includes the Configuration Management (CM) process required to implement changes under control, and to build and maintain the asset inventory.
SM41	CAPTCHA (and its alternatives)	This challenge-response test is needed to ensure that response is not generated by a computer. This process usually requires a user to complete a simple test which the computer is able to generate and grade. This ensures that any user entering a correct solution is presumed to be human. This process is sometimes also described as reverse tuning test. In line with the evolution of such SM and in line with the business impact analysis, its alternatives can also be used : - Check box - Math question - Interactive games - Others
SM42	Identity & Access Management (IAM or IdM)	This security mechanism is an administrative process coupled with a technological solution. This is needed to identify, validate and provide unambiguous, assured identity credentials for all enterprise human and non-human entities e.g. services, processes, and devices –through authentication and/or strong authentication mechanisms- and guarantee to different type of users of devices the appropriate authorization to access different security zones. To this extent, this SM is very much linked and needs to interoperate with SM12 (Strong Authentication) regardless of using one or multiple identity services as follows: • Password Authentication; • One-Time Password (OTP); • Two-Factor Authentication (T-FA); • Chip Authentication; • Biometric Authentication;

		• Single Sign-On/Off (SSO); • Enterprise Single Sign-On/Off (ESSO). The security countermeasures associated with this SM include Role based Access Control (RBAC) and/or Attribute Based Access Control (ABAC). This service has many interdependencies with other services/security mechanisms. Major components of the IAM are as follows: User Provisioning, Access Management, Delegated Administration, Self-registration and Self-service, Workflow, Auditing and Logging and Reporting, Authentication, Integration.
SM43	Management of the Security Mechanisms	The implementation of one solution that would centrally manage all Security Mechanisms is rather unlikely, due to various and heterogeneous requirements of technology landscapes and processes; while an overall orchestration of the Security Mechanisms is to be envisaged by adopting a holistic approach. For instance, Security Information and Event Management (SIEM) solution is recommended to centrally manage the logging, reporting, searching and alerting of most –if not all- network devices, systems and applications, which should also include the integration with other solutions such as configuration management, problem management, incident management, forensic, etc. Depending on the Business Impact Analysis and the Risk Assessment, unified solutions such as Unified Communications, Unified Access Management, Unified Threat management and Extensible Threat Management, Unified Data Management, Unified Applications, etc. are to be privileged throughout the acquisition process. Last but not least, adequate resources are to be planned for a successful and effective management of the Security Mechanisms, which consequently has a positive reverberation on the business's mission and goals. All ITM security mechanisms shall be integrated with the NCIRC FOC infrastructure wherever possible.
SM44	Time Synchronisation (e.g. NTP)	This is needed for synchronizing the clocks of computer systems over packet-switched, variable-latency data networks. NTP uses a hierarchical, layered system of levels of clock sources. NTP timestamp is important for several reasons: maintenance, troubleshooting, forensic analysis, resolving technical or commercial disputes for time-sensitive transaction, etc. but time is inherently important to the function of routers and networks as it provides the only frame of reference between all devices of the ICT infrastructure.
SM45	Storage Areas security	This SM includes any storage areas such as Logical Unit Number (LUN) Management for Storage Area Network (SAN) security, and others. Software management tool for LUN is essential to enabled efficient LUN creation, management, reporting and auditing. Some of the tools may be vendor-specific or generic, the most important is that the selected tool supports the whole storage provisioning process, including the mapping to specific array-

		ports, masking specific host bus adapters (HBAs), reporting, logging and auditing functions and reclamation of storage that is no longer being used. Despite the term SAN is often used, this is not limited to that but rather pervasive to any storage service that require similar attention and precaution.
SM46	Data Scramble	Data Activity Monitoring (DAM) using the Data Scramble mechanism. This security mechanism uses an application intuitive scrambling technology that masks critical data sets and so ensuring that the protection and referential integrity of sensitive data in production, non-production and end-point environments remains usable and intact whether protecting structured or unstructured data. The security mechanism needs to maintain data integrity without going directly against the database when updating and concealing the sensitive information. This security mechanism is also known as Data Masking, Data Scrubbing and Data Obfuscation.
SM47	Data Safeguarding (e.g. DAM, FSAM)	Data Activity Monitoring (DAM) and File Server Activity Monitoring (FSAM) using the Data Safeguarding mechanism. This security mechanism needs to look after critical data infrastructure to ensure that: • Discover & Classify • Assess & Harden • Monitor & Enforce • Audit & Report On corporate data, whether hosted in a database or available through a network share, is efficiently and securely managed. In addition, it should also provide a change control solution, vulnerability assessment data leak prevention (the latter in can be implemented throughout or in conjunction with SM35 (DLP) regardless the user type accessing data (database administrator, system administrator, application administrator, or user).
SM48	Password Management	This security mechanism is required to create and/or enforce password policy, thus to ensure that users are choosing strong (difficult to crack) passwords. It needs to check every new password for compliance with the policy. Passwords that are not compliant are to be rejected together with a password policy message that helps users to choose compliant passwords. This SM is also supporting the SM37.
SM49	Identity & Authentication, Access Control (IAAC)	This is considered as the technical concept that gathers together all the technical mechanisms supporting the Security Mechanism 42 (SM42), Identity & Access Management (IAM). Although identities, or part of them, are sometimes managed within solutions due to technical constraints or other reasons, they still need to be consistent throughout the security domain and beyond that, when required. In addition, convergence of the management of identities span across physical and logical

		environments, whereas this is possible and in line with risk assessment.
SM50	Protection against (D)DoS	The Protection against (Distributed) Denial-of-Service (DoS) addresses the attempt to make a machine, or a network resource or an entire service unavailable to its intended users and/or services.
SM51	Physical security	As a minimum, this SM covers the physical security measures implemented at GSE and LSE level in accordance with [AC-35-D/2001].
SM52	Personnel security	As a minimum, this SM includes personnel security clearances, security awareness and the other measures listed in [AC/35-D/2000].
SM53	Environmental security	As a minimum, this SM includes fire protection, water protection, power protection, air conditioning, humidity control etc. in accordance with Supporting document on the Availability Aspects of Security (Ref. AC/322-D(2007)0043, section VII). To this respect and where NATO documentation is lacking, the TIA-942 "Telecommunications Infrastructure Standards for Data Centers" should be used.
SM54	Emission security	As a minimum, this SM includes TEMPEST Zoning and the other security measures listed in: - AC/322-D(2007)0036 - SDIP-27 "NATO Tempest Requirements and Evaluation Procedures" - SDIP-28 "NATO Zoning Procedures" - SDIP-29 "Facility Design Criteria and Installation of Equipment for the Processing of Classified Information"
SM55	Data processing	The security of data processing (aka data manipulation) is required when data entry and processing, as performed by the applications, requires an active and automated control that such a data (or code) is handled/processed in accordance with the rule sets as established by the business, and therefore consistent with the business objectives.
SM56	Data Diode	This refers to Boundary Protection Component (or BPC) that ensures information flow only one-way across the interconnection. Typically this is required for data transfer from low to high security domain (when there is a difference in sensitivity or classification between the two networks (e.g., from NATO RESTRICTED to NATO SECRET)), and/or when there is a greater threat from the other CIS (e.g. from public network to NATO SECRET). Implementation needs to adhere to the Data-diode security accreditation package, which also contains the configuration guidance, as endorsed by the NSAB.

6 ANNEX B – ITM Services

The ITM services can be found in the relevant sections of the document below.

NCIAgency
Customer Services C

7 ANNEX C – Baseline configurations

The following table illustrates the relevant software titles and version numbers applicable to the Hosting Guidelines (HGs) in this document.

HG #	HG Name	Software title and version applicable
HG2	Server Operating System	▪ Microsoft Windows Server 2012R2 64 bit or higher; ▪ Red Hat Enterprise Linux 6 or higher; ▪ Solaris 10 or higher.
HG3	Client Operating System	Windows 10 Enterprise or Mobile operating system.
HG4	Client and Mobile Platform Compatibility	ON ▪ MS Windows 10 Operating system; ▪ MS Office Professional Plus 2016; ▪ MS Internet Explorer (latest version at time of deployment); ▪ Adobe Acrobat Reader (latest version at time of deployment); ▪ McAfee Agent (latest version at time of deployment); ▪ McAfee Anti-Virus (latest version at time of deployment); ▪ McAfee DLP (latest version at time of deployment); ▪ Titus Email Security Classification; ▪ NCIRC HIDS and Forensics analysis agent. PBN ▪ MS Windows 10 Operating system; ▪ MS Office Professional Plus 2016; ▪ MS Internet Explorer (latest version at time of deployment); ▪ Adobe Acrobat Reader (latest version at time of deployment); ▪ McAfee Agent (latest version at time of deployment); ▪ McAfee Anti-Virus (latest version at time of deployment); ▪ McAfee DLP (latest version at time of deployment); ▪ McAfee Disk encryption (only applies to laptops and desktops); ▪ Titus Email Security Classification; ▪ NCIRC HIDS and Forensics analysis agent ▪ Cisco VPN Client (only applies to mobile client devices); ▪ McAfee HIPS (only applies to mobile client devices). Mobile (iOS devices) ▪ AirWatch ▪ Cryptify

		▪ Skype for Business ▪ VMWare Boxer ▪ EBA (tbc) ▪ Taleo ▪ HireView ▪ NA2B ▪ HeliOPS Mobile (Windows devices) ▪ AirWatch
HG11	Hypervisor	VMWare ESXi 6.5.
HG12	SW Installation Package	Windows Installer program, Redhat RPM, or Solaris Image Packaging System (IPS).
HG36	Interoperable	Microsoft BizTalk

8 ANNEX D – Best practice for developing cloud-ready applications

This section provides best practice guidance for developing applications that can be hosted on cloud infrastructure.

The following table summarises the relationship between an application's cloud characteristics, architectural principles and the best practice guidance. The sections below describe this guidance in more detail.

Application characteristic	Architectural principle	Best practice guidance
Application portability	Automated	- Prioritise Microservice architecture - Separate Business Logic from Infrastructure Concerns - Separate configuration from code - Treat backing services as attached resources - Isolate dependencies - Export services via port binding - IP protocol independence, dual protocol application (Ipv4 vs Ipv6)
Application interoperability	Automated	- Use interoperable API interfaces - Favour use of Rest[1]/JSON interface - Rely on asynchronous integration - Service registration/discovery
Application resiliency and availability	Failure-aware	- Event driven architecture
	Latency aware	- Data Caching

[1] As per INSTR TECH 06.02.02 SIP FOR REST SECURITY SERVICES, and INSTR TECH 06.02.07 SIP FOR REST MESSAGING

	Event Driven	- Redundancy - Data replication - Stateless Services - Restartable process - Design for idempotency - Automation
Application scalability	Parallelizable	- Scale out not up
	Event Driven	- Decompose application and workload
	Automated	- Favour finer grain service design - Avoid monolithic deployment - Design for statelessness - Design for data partitioning - Favour shared nothing architecture - Design for eventual consistency
Application efficiency	Computing and Storage - consumption aware	- Right sizing
	Parallelizable	- Share infrastructure with multi-tenancy - Avoid application clustering overhead - Adopt Throttling to limit resource usage - Balance finer grain service design and service chattiness
Application security	Secure	- Ensure compliancy with ITM security mechanisms - Apply secure coding practices - Ensure security accreditation is achieved
Application deployment	Automated	- Automate deployment - Externalised configuration - Template-driven deployment - Avoid monolithic deployment
Application monitorability	Instrumented	- Instrument your application
	Automated	- Log everything through event streams

8.1.1 Application Portability

The following best practices should be kept in considerations in order to build applications that are platform independent (infrastructure, operating system and topology independent).

Application developers should develop applications capable of running on highly virtualised and shared infrastructure. For this reason applications must be designed as much as possible independently from their surrounding physical environment.

8.1.1.1 Prioritise micro-service architecture

Microservices are independently deployable and manageable and this makes them more portable than traditional applications and services. Because a microservice bundles all of its capabilities into a lightweight unit, it inherently satisfies the constraints of contextual independence. Microservices should implement a single functional responsibility with the goal of reducing development and maintenance complexity for each individual service. This makes microservices more fine-grained than their "traditional SOA" predecessors.

Micro service architecture (MSA) takes the following principles and advocates applying them with renewed vigour to meet the goals of agility and scalability:

- Service orientation should be followed at a granular level and is combined with the single responsibility principle from object-oriented design.
- Separation of concerns should be followed to isolate services "all the way down" including service development, data management and runtime isolation.
- Loose coupling should be applied through well-designed APIs and interfaces. REST and event-driven patterns are commonly used to help maintain decoupling.

8.1.1.2 Separate Business Logic from Infrastructure Concerns

Clean separation of infrastructure from business logic allows for the decoupling of application platforms from the underlying infrastructure. This separation enables application hosting on standard infrastructure. For many modern applications, this is trivial. For example, any applications that run in managed environments, such as JVMs or .NET Common Language Runtimes (CLRs) are already separated somewhat cleanly from their infrastructure by their virtual machine. However, most Java EE applications rely on security and transaction infrastructure implemented in the Web container – you must recode the application to move it to a different type of container or to get it to rely on an external security service (for example, OAuth).

A container-agnostic framework, such as Spring, provides a layer of abstraction between an application and various containers. Applications compiled to native code for example, execute directly on an OS and are more challenging to move from one OS to another environment.

Unmanaged code, such as C or C++ that is written specifically for an OS version or C standard library, or GNU C Library version can be more challenging to migrate to a cloud and should be avoided.

8.1.1.3 Separate configuration from code

Application configuration here is everything that is likely to vary between deploys (e.g. staging, production, developer environments, etc.). This includes:

- Resource handles to the database, cache, and other backing services
- Credentials to external services
- Per-deploy values such as the canonical hostname for the deploy

Applications sometimes store configuration as constants in the code. This should be avoided. Cloud requires strict separation of configuration from code. Configuration varies substantially across deploys, code does not.

Best practice is to store configuration in environment variables that are easy to change between deploys without changing any code.

Anything configured within an application that directly connects the code to the part of the infrastructure that the application runs on is in contrast with virtualisation. Something as easy as hard-coding a host name or IP address into source code creates inadvertent hardware affinity. One design goal for cloud capable applications is to completely separate variable concerns (such as IP addresses, DNS names or file system volume names) from the other common concerns — the application configuration and code — so that the code can be hosted on any hardware platform.

8.1.1.4 Treat backing services as attached resources

Backing services, such as caching, analytics and messaging, should be treated as attached resources and consumed identically across all environments.

Each distinct backing service is a resource. For example, a MySQL database is a resource; two MySQL databases (used for sharing at the application layer) qualify as two distinct

resources. The application treats these databases as attached resources, which indicates their loose coupling to the deploy they are attached to.

Resources can be attached and detached to deploys at will. For example, if the app's database is misbehaving due to a hardware issue, the app's administrator might spin up a new database server restored from a recent backup. The current production database could be detached, and the new database attached – all without any code changes.

8.1.1.5 Isolate dependencies

Applications must explicitly declare and isolates dependencies via appropriate tooling (e.g., Maven, Bundler, NPM) rather than depending on implicitly realised dependencies in their deployment environments.

8.1.1.6 Export services via port binding

This means an application is self-contained and exports any/all services via port binding (including HTTP).

A cloud-capable application should be completely self-contained and not rely on runtime injection of a webserver into the execution environment to create a web-facing service. The web app exports HTTP as a service by binding to a port, and listening to requests coming in on that port.

This is typically implemented by using dependency declaration to add a webserver library to the app, such as Tornado for Python, Thin for Ruby, or Jetty for Java and other JVM-based languages. This happens entirely within the app's code. The contract with the execution environment is binding to a port to serve requests.

8.1.1.7 IP protocol independence, dual protocol application (Ipv4 vs Ipv6)

Application code must be written in a protocol-independent way, which will allow applications to work with both IPv4 and IPv6 (ITM infrastructure supports dual stack mode). The following list summarises the more common IP version dependencies that can be found and managed in source code:

- Presentation format for an IP address: An ASCII string that represents the IP address, a dotted-decimal string for IPv4 and a hexadecimal string for IPv6.
- Transport layer API: Functions to establish communications and to exchange information.
- Name and address resolution: Conversion functions between hostnames and IP addresses.
- Specific IP dependencies: More specific IP version dependencies, such as IP address selection, application framing, and storage of IP addresses.
- Multicast applications: One must find the IPv6 equivalents to the IPv4 multicast addresses and use the right socket configuration options.

8.1.2 Application Interoperability

The following best practices should be followed to design cloud applications that can be easily integrated with other components internal or external to the cloud environment.

8.1.2.1 Use interoperable API layers

Applications built with services need to interact with and be consumed by other applications, whether hosted on cloud infrastructure or deployed externally. In addition, bear in mind that APIs are the primary pattern for exposing services and microservices to consumers. Application components must assume they are cooperating in a heterogeneous system of resources to form a complete solution. Interoperability across cloud provider-consumer APIs

with explicit contracts minimises provisioning friction and reduces the need for prior agreement. Standard formats and protocols reduce technical challenges.

The only loosely coupled way to expose capabilities in cloud distributed solutions is to offer functionality to consumers as Web service APIs, rather than tightly coupled frameworks embedded in libraries. Support for a variety of Web service interface models (typically RESTful HTTP and SOAP/WS-*) boosts interoperability by supporting different consumer demands.

APIs should be designed with the following principles:

- Clear inputs and outputs
- Ability to perform business validations
- Clearly defined error codes
- Support asynchronous and synchronous models of integration
- Used policy mechanism to secure access

8.1.2.2 Rely on asynchronous integration

Rather than relying on blocking, synchronous interactions, using asynchronous integration within cloud applications decouples message producers' and consumers' work rates and life cycles. Work queues are an internal integration mechanism for exchanging work between application components.

8.1.2.3 Service registration/discovery

Applications that are deployed as a set of smaller services will depend on the ability of those services to interact with each other. Because each of those services could be running across multiple compute resources there needs to be a way for each service to be addressed. For example, in a traditional infrastructure if your front end web service needed to connect with your back end web service, you could hardcode the IP address of the compute resource where this service was running. Although this approach can still work on cloud computing, if those services are meant to be loosely coupled, they should be able to be consumed without prior knowledge of their network topology details. Apart from hiding complexity, this also allows infrastructure details to change at any time. Loose coupling is a crucial element if you want to take advantage of the elasticity of cloud computing where new resources can be launched or terminated at any point in time. In order to achieve that, applications should implement a service discovery mechanism.

The service registry provides a listing of all services and makes them available to frontend services through a client library which performs load balancing and routing to backend services.

One way to implement service registry/discovery is to adopt the API gateway pattern. API gateways are simply a special class of services that provide client applications with a single entry point to the backend.
They access tens (or hundreds) of micro-services concurrently with each request, aggregating the responses and transforming them to meet the client application’s needs. They also perform protocol translation (e.g., HTTP to AMQP) when necessary.

8.1.3 Application Resiliency and Availability

The following best practices should be followed to design applications failure-aware and resilient to different types of failure that can occur in the cloud environment: hardware, software, network and human.

8.1.3.1 Event driven architecture

Rather than being written as a sequence of actions, event-based programs are written as a set of event handlers that respond to different events. When an application component needs to communicate with another application component, it should "fire and forget" a message,

without waiting for an acknowledgment, thus assuming the infrastructure will deliver the message to the recipient. Event-driven architecture based on pub/sub messaging, is useful in increasing resiliency (the messaging intermediary [e.g., broker] stores messages if the receiver is down. If an instance is lost the damage is minimal. The most that is lost is a single transaction. The remaining information in the queue continues to be distributed to active instances

8.1.3.2 Data Caching

Caches aim to provide fast access to frequently accessed or rarely changing data. Distributed caches make any data item in an entire server cluster available quickly to any node that needs it. Keeping a copy of as much data as possible in the aggregated cache improves performance and scalability, avoiding costly disk reads. In the case of a data update, cache coherence protocols make sure all copies of that data are synchronised according to policies.

Caching can improve data availability by creating an entire redundant copy of every piece of data in the cache, split across all participants in the cluster.

Caching can also be used to move static content closer to users in order to reduced perceived latency. For examples to serve static files (for example, images, video, JavaScript, CSS and other media) or streaming media latency can be reduced by caching them in multiple nodes/locations, ideally in the same geographic region as each user group.

8.1.3.3 Redundancy

Redundancy can work at multiple levels to mitigate single-node failure (such as a server or disk failure) and certain network failures (such as partitioning, in which segments of the network are isolated by failed connections). One example is spreading application instances across multiple geographically distributed zones. Multiple instances should be managed with an active-active mode (instead of active-passive) with a global load balancers to distribute the load among nodes. The advantage versus active/passive is improved usage of cloud resources and a better ability to provide service availability closer to users through multiples node locations. Applications must be written for active/active design particularly in regards to handling conflict resolution.

8.1.3.4 Data replication

Related to redundancy is also the mechanism of data replication. Replication refers to creating a redundant set of data so that, if one set of instances fails, another copy of data is available for other instances to work with. Mirroring critical data to another datacentre provides a higher level of resilience and availability. For example, an organisation can utilise a third-party cloud brokerage to broker IaaS providers and autoprovision and spread data (and equally, server workloads) across several providers to increase IT service continuity availability and reduce risks associated with a single provider outage.

8.1.3.5 Stateless Services

Factoring components into stateless services – that is, services that do not maintain application processing state (such as client session data) – makes the services parallelizable because any service instance can process any request and also increases resiliency because, if a server instance fails, requests can be routed to another server instance, processing can be restarted, and if necessary, the infrastructure can automatically create a new instance to replace it. All this can and should be achieved without the client becoming aware.

8.1.3.6 Restartable

Software components must be capable of recovering the processing or task state externally in the event of a failure or a decision to tear down a processing instance for operations management reasons. Software designs must be resilient to reboot and re-launch.

8.1.3.7 Idempotency

Idempotency is a pattern for ensuring transactional integrity at the application semantics level. Idempotent operations can be called repeatedly and produce the same results. HTTP is not a reliable protocol; it has no mechanisms built into the protocol that allow HTTP request messages to be reliably delivered using once-and-only-once or retry-until-success semantics. Nor is there a mechanism in the protocol allowing the client and server to reach agreement on the success or failure of a message exchange. Though HTTP itself is unreliable, developers can implement reliable application semantics because idempotent methods can be called repeatedly until the client gets a response with no danger of unwanted side effects on the server

8.1.3.8 Automation

Human error can significantly decrease the availability of your application because a common issue for updating servers and releasing new versions of your application is the configuration drift from pre-production environments into production environments. Automating the resources that your application utilises ensures that you have a repeatable, predictable process when you deploy, remove, provision and upgrade resources. This additionally enables you to revert back to last known good versions of your application in the event of failures. Automation eliminates the human errors associated with doing updates manually.

Application developer should adopt automation capabilities in their application life cycle using products like Chef, Puppet and Ansible, to name a few.

8.1.4 Application Scaling out

The following best practices should be followed to design applications capable to scale horizontally in a cloud environment.

8.1.4.1 Scale out not up

Scaling up always has a break point. If developers scale up a point is reached where an application runs out of resources. Scaling out (horizontally) ensures nearly infinite scalability. As long application bottlenecks are anticipated and dealt with in application design, applications designed to scale out can readily take advantage of the cloud's massive scalability and elasticity.

8.1.4.2 Decompose application and workload

Decomposing applications into different components allows more close control of the number of instances to deploy to ensure that the application can meet peaks in demand. You can scale out as demand increases, and scale in when the application is not busy or you can configure automatic scaling. Applications can be decomposed into multiple components. For example, you might choose to decompose a complex application into separate logical compute instances that implement the website UI, API, administration site, background processing, caches, and more.

When considering decomposing applications, the primary design decision is to define the boundary between the separate parts of the application. Many applications have natural boundaries. For example, it is common to separate the UI from the background processing tasks and offload work to these tasks to maintain performance and responsiveness of the UI. Where an application contains distinct and separate UI sections, such as public and restricted areas, these may also be candidates for decomposition. Even a simple website UI can be decomposed into multiple components, for example by separating the pages that require high throughput from the remainder of the site.

If an application uses services that expose an API, these services are likely to be implemented as separate components or roles in order to manage their scale independently from the

website UI. It may be appropriate to separate the tasks in these components or roles as part of the decomposition process.

You should also take into account the workload of each separate part of the application. Workload decomposition refers to decomposing an application into parts based on functional workloads that may have different scale, security and management requirements. This may help you define the decomposition boundaries of the application.

Designing for complete separation of compute and persistence provides great deployment and scaling flexibility. If we make sure our cloud applications use web services for ALL data storage (including things we normally might not consider, such as log files and debugging streams), we enable the infrastructure around them to be managed and invested in separately, shield the application from underlying changes in storage technology, and decrease deployment overhead. This means that our compute instances can be deployed, destroyed, or relocated as needed.

Decisions about how the components of an application are grouped into a single or multiple compute hosts must be based on the requirements of the individual logical components. Components that have similar requirements can be grouped into the same partition. However, the requirements of the application as a whole must be considered. When allocating components to compute resources, consider:

- **Management and Maintenance**. The cost and effort of managing, monitoring, and maintaining applications (and the services, components, and tasks each one requires) depend to some extent on the range of different items that are deployed. Decomposing applications will increase management, monitoring, and maintenance overhead; although this is not a linear relationship because you will typically be able to extend existing tools and systems to include the additional deployments.
- **Runtime cost**. You are billed for every hosted compute instance you deploy to the cloud environment, and so decomposing applications is likely to increase runtime costs. However, implementing auto scaling can minimise the runtime cost for items that are subject to variable demand or load, while maintaining availability.
- **Dependencies**. Some components may have dependencies that prevent them from being separated. It may also be advantageous to minimise the requirements for inter-process communication between components by hosting them in the same compute instance, for example, to minimise latency or reduce deployment complexity.
- **Inter-process Communication**. Tasks may need to communicate with components in other compute instances, perhaps by using shared memory, private HTTP or TCP endpoints, asynchronous messaging, named pipes, data stores, or a global cache. When this is the case, consider how it will affect the design. Extremely chatty components, or components that are heavily dependent upon each other, could be hosted in the same instance to reduce the communication overhead.

8.1.4.3 Favour finer grain service design

Business logic must be decomposed into autonomous **finer grained services**. Services are autonomous when they can achieve much of their processing with minimum communications with peer processes, making them good candidates for parallel processing. Autonomous services can fail independently without impacting an entire system catastrophically.

Granularity impacts design and requires solution architecture to think more about composite instead of monolithic design.

Well-designed services are simple, autonomous and single purpose. More complex tasks get done by composing simple services. Modularity makes overall system design testable and comprehensible.

Service granularity must follow the two following principles:

- Tightly scoped: Microservices should implement a single functional responsibility with the goal of reducing development and maintenance complexity for each individual service.
- Highly cohesive: The capabilities provided by a single microservice should be closely related and should share a common purpose and common data on which they act. The goal of this cohesion is to prevent ripple effects as changes are made to a system. Service boundaries should be defined to minimise the chances of any given feature change requiring coordination between service implementation teams. This contrasts with coarse-grained services that aggregate a number of loosely related or disparate functions and actions on multiple data entities.

Finer grained component design also gives you the ability to run different types of workloads in a finite shared resource set.

8.1.4.4 Design for statelessness

Stateless services — that is, services that do not maintain application processing state (such as client session data) — make the services parallelizable and ready to scale because any service instance can process any request. To build stateless services two distinct approaches can be followed:

- Simple/Scaled — this service has no data persistence and might perform a compute intensive calculation or other workload based on parameters that are suitable for passing in an API payload. Note that this model could be extended to use an asynchronous interface if the processing is long-running.
- Complex/Scaled — this service has its own encapsulated data persistence. When that persisted data changes, it emits events to notify other services. Note that the persistence mechanism is scoped within the service and is shared by service instances to allow the service implementation to be stateless.

8.1.4.5 Design for data partitioning

Partitioning refers to dividing a large dataset into smaller databases. Smaller databases are easier to maintain, can complete operations such as indexing and queries faster, and most importantly, can be distributed across multiple, independent disks or servers.

Partitioning can be done in a number of ways, including by index or by data domain. Partitioning by index is typically done horizontally (separating rows by key or data range — such as "northern region" and "southern region"), but can be done vertically in some systems (placing different columns on separate partitions). Partitioning a database horizontally is known as "sharding." Shards can be distributed so the system maintains performance as load increases, and extra nodes can be added, maintaining scalability. Each node works on a subset of the data in parallel.

8.1.4.6 Favour shared nothing architecture

The goal of shared-nothing architecture (also known as "isolated tenancy") is a distributed system of independent components that does not share access to a latency-bound resource, such as memory or file storage. Thus, no single point of contention exists. When tasks run without being dependent upon one another, state coordination and communication is not required among tasks, which reduces latency. Data and processing should be partitioned to exhibit a high locality of reference so that all related data and code required to complete a task is local. More nodes can be added without affecting each other to achieve linear scalability.

8.1.4.7 Design for eventual consistency

In a cloud distributed system, where application semantics demand updating multiple data sources, guaranteeing ACID transactions using distributed two-phase commit (2PC) is challenging and should not be relied on. This scenario should be avoided by changing the

application design. Eventual consistency is one approach for a cloud application provider to govern visibility of updates across members of a distributed environment. Applications must assume reads of just-written data may be out-of-date. This condition requires a very different set of application assumptions than those available in an environment with ACID transactions.

The CAP Theorem states that — in a distributed system where C is consistency, A is availability and P is network partitioning tolerance — only two of the three properties can be guaranteed. Networks will fail, so solutions must be partition-tolerant (the P). For some cloud application use cases, availability (A) is more important than consistency (C).

8.1.5 Application Efficiency

The following best practice should be adopted to design efficient applications that are consumption-aware and minimise usage of the CPU, memory, network input/output (I/O) and storage I/O, thus saving costs and enabling more users, transactions or requests to be packed into a shared and virtual infrastructure.

8.1.5.1 Right sizing

Application developers should avoid traditional capacity provisioning practices that lead to sizing the underlying infrastructure for the worst-case resource consumption scenario, which usually implies a large, monolithic footprint. This "eager loading" pattern reduces the ability to tightly pack applications into finite physical resources.

Application developers should select the cheapest type of infrastructure that suits the workload's requirements. In other cases, using fewer instances than a larger instance type might result in lower total cost or better performance. Application developers should benchmark and select the right instance type depending on how your workload utilises CPU, RAM, network, storage size, and I/O.

Similarly, costs can be reduced by selecting the right storage solution for application needs. ITM infrastructure provides several type of storage infrastructure: SAN enterprise. SAN Midrange, NAS.

8.1.5.2 Balance finer grain service design and service chattiness

Optimising software components for virtualised and containerised environments requires fine-grained components. Finer-grained component design also gives you the ability to run different types of workloads in a finite shared resource set. Not only can you run more small pebbles, but dissimilar colours and shapes of pebbles will find capacity in a shared virtual and dynamic environment.

Smaller components (or microservices) can be created and destroyed more quickly, which enables rapid workload migration to leverage available infrastructure. Finely grained components deployed in a modular way minimise the total deployment footprint. Deploying the minimum size of components you need is beneficial — not because you want to be kind to the provider, but in a metered consumption environment, you only get charged for the footprint you deploy.

Fine-grained components allow solution providers to charge for features at different granularities.

This notion of granularity impacts design and requires solution architectures to think more about composite instead of monolithic design. In composite design, you need more of the different types of pebbles to create an application. Well-designed services are simple, autonomous and single purpose. More complex tasks get done by composing simple services. Modularity makes overall system design testable and comprehensible. A module implementer can focus on the implementation, optimisation and verification of the module with minimum concern about the rest of the design.

The flip side of this coin is that communication traffic between finely grained, distributed components can be expensive in a metered consumption computing platform. If the footprint

size relates to high cost and bandwidth and internal communications do not, finely grained components are a good approach. In the reverse case, they are not.

Extremely chatty components, or components that are heavily dependent upon each other, must be hosted in the same instance to reduce the communication overhead.

In some cases, it may be appropriate to consolidate multiple tasks or operations into a single computational unit to increase compute resource utilisation, and reduce the costs and management overhead associated with performing compute processing in cloud-hosted applications.

8.1.5.3 Share infrastructure with multi-tenancy

Applications in a cloud environment should be designed as multi-tenant so that multiple tenants can share the same instance of the application with all data and user accounts separated from each other.

The opposite traditional single-tenancy model, serves only one entity at a time from an application instance and limits the ability to share infrastructure. The overhead of creating and maintaining application data structures, such as in-memory objects and database tables for each client, is inefficient and also limits productivity of automating software updates to all users.

8.1.5.4 Avoid application clustering overhead

Scaling out in a limited way with an application server model typically implies a proprietary protocol for managing session state and server affinities. Deployment location and topology are fairly static in application server clusters, inhibiting flexible workload migration. Two anti-patterns exist here: the first is that the database is a shared resource for session state that becomes a bottleneck, secondly, using an app server cluster, you are paying two, or four, or sixteen times for core-based application server software licenses for a level of demand that may be variable. Licensing software by virtual cores helps but doesn't solve this problem.

8.1.5.5 Adopt throttling to limit resource usage

Throttling allows to limit the consumption of resources used by an instance of an application, an individual tenant, or an entire service. So that the overall system can still perform under pressure from increased demand. This pattern is useful for the following:

- To ensure that a system continues to meet service level agreements.
- To prevent a single tenant from monopolising the resources provided by an application.
- To handle bursts in activity.
- To protect against brute-force denial-of-service attacks.
- To help cost-optimise a system by limiting the maximum resource levels needed to keep it functioning.

Under load, an application in a cloud environment can scale, but the architect and administrator need to plan for the limits of allowable scaling. At this limit, which is determined by instrumentation, in order to keep the system responsive to some extent, requests from specific clients, or every client, need to be throttled (limited). Implementing this pattern will require interaction between the ITM monitoring system (SMC), the auto-scaling configuration and a mechanism for returning limit failures or levelling load.

Throttling can be implemented following several strategies, including:

- Rejecting requests from an individual user who has already accessed system APIs more than *n* times per second over a given period of time. This requires that the system meters the use of resources for each tenant or user running an application.
- Disabling or degrading the functionality of selected nonessential services so that essential services can run unimpeded with sufficient resources. For example, if the application is streaming video output, it could switch to a lower resolution.

- Using load levelling to smooth the volume of activity. In a multitenant environment, this approach will reduce the performance for every tenant. If the system must support a mix of tenants with different SLAs, the work for high-value tenants might be performed immediately. Requests for other tenants can be held back, and handled when the backlog has eased.
- Deferring operations being performed on behalf of lower priority applications or tenants. These operations can be suspended or curtailed, with an exception generated to inform the tenant that the system is busy and that the operation should be retried later.

8.1.6 Application deployment

This section contains guidelines to build applications which are able to deploy themselves easily onto a variety of platforms, identifying and simplifying the number of environment dependencies.

8.1.6.1 Automate deployment

In the ITM cloud environment, the application deployment process must be automated (for example, include no manual steps) so that services can be moved and quickly restarted on alternative servers. ITM infrastructure services are deployed on dynamic, shared and virtual infrastructure, the servers are transient, the data is volatile, and you can lose a server or disk at any time. So, you need to build something that's resilient to that transience. All infrastructure states for an application environment should be re-creatable from templates and recipes that describe how the server instances need to be configured and updated with data.
To provision quickly, the template must have clean separation of concerns between application parts that are variable and parts that are common – for example, factoring out endpoints, connection strings, UI settings, user skins, IP addresses and so on.

8.1.6.2 Externalised configuration - Template-driven deployment

All provisioning, instantiation and configuration of an application should be scripted and preferably encapsulated in an automatically unpacked and deployable template. Automating deployment also means encapsulating application dependencies for deployment as a single package, or a recipe. Configuration values in scope here, are anything that changes per deployment (for example, development, test, prod environments; hostnames; and ports), and this configuration must be stored in the environment and injected into the application at deploy or runtime as better explained hereunder.

The majority of application runtime environments include configuration information that is held in files deployed with the application, located within the application folders. In some cases it is possible to edit these files to change the behaviour of the application after it has been deployed. However, in many cases, changes to the configuration require the application to be redeployed, resulting in unacceptable downtime and additional administrative overhead.

Local configuration files also limit the configuration to a single application, whereas in some scenarios it would be useful to share configuration settings across multiple applications. Examples include database connection strings, UI theme information, or the URLs of queues and storage used by a related set of applications.

Managing changes to local configurations across multiple running instances of the application, especially in a cloud-hosted scenario, may also be challenging. It may result in instances using different configuration settings while the update is being deployed.

In addition, updates to applications and components may require changes to configuration schemas. Many configuration systems do not support different versions of configuration information.

The best practice is then to store the configuration information in external storage, and provide an interface that can be used to quickly and efficiently read and update configuration settings.

The type of external store depends on the application runtime but typically could be an ITM-based storage service or could be a hosted database.

The backing store chosen for configuration information should be fronted by a suitable interface that provides consistent and easy to use access in a controlled way that enables reuse. Ideally, it should expose the information in a correctly typed and structured format. The implementation may also need to authorise users' access in order to protect configuration data, and be flexible enough to allow multiple versions of the configuration (such as development, staging, or production, and multiple release versions of each one) to be stored.

8.1.6.3 Avoid monolithic deployment

At time of deployment, do not put all the components and all their dependencies into one monolithic archive file for deployment into an application server. Small changes, even if isolated to specific areas of an application, require packaging and redeploying the entire application (and sometimes a restart of the virtual machine instance hosting the application). This is the stuff of maintenance and operations nightmares. A monolithic approach to deployment will not support the agility you need to move and change those applications dynamically around the different topology options that the cloud gives you.

8.1.7 Application monitoring and control

In a cloud environment is it mandatory to develop instrumented applications that produce and log data about their execution, resource consumption and performance. Instrumentation is needed in a cloud environment for any system to support monitorability, elasticity and scalability.
The following subsections describe best practices to monitor the application state, adopting specific mechanisms and patterns.

8.1.7.1 Instrument your application

Applications should include diagnostics features that generate custom monitoring and debugging information, especially when an error occurs. This is referred to as *instrumentation*, and is usually implemented by adding event and error handling code to the application (following logging frameworks such as Log4net/Log4j is advisable). The process of gathering remote information that is collected by instrumentation is usually referred to as *telemetry.* Telemetry is used by the ITM infrastructure.

Instrumentation allows to capture vital information about the operation of your application. This information should include:

- Details of operational events that occur as part of the normal operation of the application, together with useful information about that event.
- Details of runtime events that occur, and useful information about that event such as the location or data store used and the response time for access to the data store. These are also informational events that can provide additional insight into the normal operation of the application. The event should not include any sensitive information such as credentials, or any other data that might enable an attacker to obtain the logs to compromise the system.
- Specific data about errors that occur at runtime, such as the customer ID and other values associated with an order update operation that failed. Typically these are warning or error events and will contain one or more system-generated error messages.
- Data from performance counters that measure specific values related to the operation of the application. These might be built-in system counters, such as those that measure processor load and network usage, or they might be custom performance counters that measure the number of orders placed or the average response time of a specific component.

A common implementation of instrumentation is an ability to change the level of detail that is collected on demand, usually by editing the configuration of the application. Under normal conditions the data from some informational events may not be required, thus reducing the cost of storage and the transactions required to collect it. When there is an issue with the application, you update the application configuration so that the diagnostics and instrumentation systems collect informational event data, as well as error and warning messages, to assist in isolating and fixing faults. It may be necessary to run the application in this extended reporting mode for some time if the problem appears only intermittently.

The fundamental approach to designing the instrumentation for any application can be defined by considering the requirements in terms of the basic stages for isolating and fixing errors:

1. **Detect performance issues and errors quickly**. Performance counters and event handlers can indicate problems in specific areas of the application due to component and service failures, overloading, and other issues; sometimes before end users are affected. This requires a constant or scheduled mechanism that monitors key thresholds and triggers the appropriate alerts. Detailed information from the instrumentation allows you to drill down into the execution and trace faults.
2. **Classify the issue to understand its nature**. In cloud-hosted applications that run in a shared multitenant environment, such as the ITM infrastructure, some issues such as connecting to a database may be transient and will resolve themselves. Other issues may be systemic, such as a coding error or an incorrect configuration setting, and will require intervention to resolve the problem. If an incident could be considered a security incident, it shall be reported to NCIRC FOC.
3. **Recover from an incident and return the application to full operation**. Use the information you collect, perhaps after turning on additional logging settings, to fix the problem and return the application to full service. This is especially important if it is a commercially valuable application for the organisation, such as an e-commerce site, or has SLAs that you must meet to avoid financial penalties and customer dissatisfaction.
4. **Diagnose the root cause of the problem and prevent it reoccurring**. Carry out root cause analysis to determine the original cause and the underlying nature of the problem, and make changes to prevent reoccurrence where this is possible. The instrumentation data collected over time will help you identify recurring patterns and trends that led up to the incident, such as overloading of a specific component or invalid data being accepted by the application.

To be able to perform these steps you will need to collect information from all levels of the application and infrastructure. For example, you should consider collecting data about the infrastructure such as CPU load, I/O load, and memory usage; data about the application such as database response times, exceptions, and custom performance counters; and data about business activities and KPIs such as the number of each type of business transaction per hour and the response time of each service the application uses.

8.1.7.2 Log everything through event stream

Logs are usually the way to create visibility into the behaviour of a running application. In server-based environments they are commonly written to a file on disk (a “log file”); but this is only an output format. Logs are the stream of aggregated, time-ordered events collected from the output streams of all running processes and backing services. Logs in their raw form are typically a text format with one event per line (though backtraces from exceptions may span multiple lines). Logs have no fixed beginning or end, but flow continuously as long as the application is operating.

Applications in the ITM environment should not attempt to write to or manage log files. Instead, each running process should write its event stream. During local development, the developer will view this stream in the foreground of their terminal to observe the application’s behaviour. In staging or production deploys, each process stream will be captured by the ITM execution environment (Service management and control service), collated together with all other streams from the app, and routed to one or more final destinations (such as SPLUNK and

TrueSight) for viewing and long-term archival. These archival destinations are not visible to or configurable by the application, and instead are completely managed by the ITM execution environment.

9 ANNEX E – Software deployment SMC mechanisms

[SMC team to add mechanisms]