Smart Open Services for European Patients

Open eHealth initiative for a European large scale pilot of

Patient Summary and electronic Prescription

D3.9.1: epSOS Pilot System Components Specifications

WORK PACKAGE	**JWG 3.8/3.9**
DOCUMENT NAME	D3.9.1: epSOS Pilot System Components Specifications
SHORT NAME	**D3.9.1**
DOCUMENT VERSION	**1.0**
DATE	**01/10/2010**

COVER AND CONTROL PAGE OF DOCUMENT	
Document name:	**D3.9.1** - epSOS Pilot System Components Specifications
Document Short name:	**D3.9.1**
Distribution level	**PU**
Status	Final
Author(s): Organization:	B.Horvat, S.Evdokimov, R.Melgaard, P.Loubjerg, S.Bittins, J.Caumanns, A.Estelrich, P.Ruestchmann, A.Périé, G.De Bejarry, A.Hansen, M.Kovarova, G:Heider, P.Gross, T.Pass, G.Cangioli, S.Lotti, M.Melgara, E. Albertini, G.Orsi **JWG-Technical Core Team and WP3.9**

Dissemination level: PU = Public, PP = Restricted to other programme participants, RE = Restricted to a group specified by the consortium, CO = Confidential, only for members of the consortium.

ABSTRACT

"**D3.9.1 - epSOS Pilot System Components Specifications**" provides the information concerning the software component development activities performed by the Proof of Concept implementation workstream of WP3.9, in co-operation with WP3.8 in the Joint Working Group WP3.8 / 3.9.

Strategic activities like implementation strategy definition and implementation proposal selections were performed in co-operation with TPM task force and PMT.

This document is heavily based on the High Level Design document, epSOS Internal Deliverable: "Architecture of the National Contact Point in a Box (NCP-in-a-Transparent-Box)", developed by the Joint Activity Group (JWC) 3.8 / 3.9.

The document is addressed to the MS and to the vendors who wants to develop the NCP and the Front-end Portal. It is an umbrella document covering the Specifications of WP3.3 – 3.7, the Guidelines of WP3.8, the testing procedures defined in WP3.9 and applied in WP3.10, and the MS implementation of PD4.

After a summary of WP3.9 organisational and methodological approaches, the NCP implementation guidelines, provided by PEB/PSB are introduced.

Basic requirements for NCP implementation are analysed. High Level Design of the NCP-in-a-Box concept, both considered as gateway toward Country A and toward Country B, is provided. It contains general description of the NCP-in-a-Box architecture as well as details about individual software components and Country B

Front-end.

Software Component specifications are provided in the document, while Detailed Technical Specifications of some of them are reported as Appendixes.

A specific chapter is devoted to Semantic Group activity description, including the definition of Central Services for Terminology Management and processes to develop and maintain Master Value Sets and Translation/Transcoding Catalogues.

Implementation strategies and scenarios are analysed, describing the procurement procedures defined and actuated by epSOS.

The Proof of Concept NCP implementation, assigned Fraunhofer ISST / Elga Team (FET) consortium, including Vendors, are finally described. The implementation differs in some aspects from the overall design described in Chapter 3 - 7: this is not infringing the interoperability concepts and specification. Chapter 9 and Appendix A provide a view on differences and detailed technical specification of the implemented proof of concepts.

It was decided to keep both views in D3.9.1 to allow MSs and vendors willing to implement the NCP as defined in the High Level Design to get the appropriate specification and to provide the detailed information to the MSs willing to adopt FET implementation.

Change History

Version	Date	Status Changes	From	Details	Review
V0.0	24/05/2010	Draft	Lombardy	Initial Structure based on minutes by JWG 3.8/3.9 (TConf. 18/05/ 2010)	JWG-TC
V0.1	04/06/2010	Draft	Lombardy+ Gematik	Comments from Ben and Marcello. Revised 1.5 Scope of the Document	
V0.2	04/06/2010	Draft	WP3.9	Assignments during 4/6/10 TConf	
V0.3	23/06/2010	Draft	Gematik	Structure of the document reworked	

V0.4	24/06/2010	Draft	Lombardy	Comments on structure	
V0.41	28/06/2010	Draft	Gematik	Updated according to the outcome of 28/06/10 TConf	
V0.42	8/07/2010	Draft	Gematik	Updated according to the results of 06/07/10 meeting in Vienna	
V0.43	16/07/2010	Draft	Gematik	Added text for section 5.3 that was provided by CareCom	
V0.44	23/07/2010	Draft	Gematik	Updated glossary (1.6) and NCP Front-End Portal (A11) sections	
V0.45	23/07/2010	Draft	Lombardy	Included Appendix B1, provided by ASIP. Re-introduced Appendix B2 and B3 in accordance with what decided during today tconf	
V0.46	28/07/2010	Draft	Dig.Health	Added two contributions 5.1 & 5.2	
V0.47	02/08/2010	Draft	Dig.Health	Split Appendix A into separate doc. Added chapter 6 Updated chapter 7.4 & 7.6	
V0.48	06/08/2010	Draft	Dig.Health	Revised Chapter 1	
V0.49	10/08/2010	Draft	Lombardy	Revised Chapter 7.5 and 8 Released for Internal QA	
V0.50	29/08/2010	Draft	Lombardy	Revised Abstract, Chapter 1	

V0.51	03/09/2010	Draft	Gematik	Integrated results of the internal review	
V0.52	03/09/2010	Draft	Lombardy	Version for external review	
V0.53	23/09/2010	Draft	Gematik, Fraunhofer	Integrated results of the internal review; included new version of Chapter 9	
V0.90	24/09/2010	Draft	Lombardy	Version for PEB approval	
V1.0	01/10/2010	Final	Lombardy	Version approved by the PEB	

TABLE OF CONTENTS

1 Introduction 10
1.1 Background 10
1.2 WP3.9 Methodological approach 11
1.2.1 Proof of Concept implementation workstream 12
1.2.2 Semantic interoperability workstream 14
1.2.3 Test strategy definition and test tool development workstream 15
1.3 Specification Methodology followed 17
1.4 Scope, audience and purpose of this document 19
1.5 Glossary 20
1.5.1 Terms and definitions 20
1.5.2 Abbreviations 22
1.6 Requirements 25
1.6.1 Basis Requirements 25
1.6.2 Additional Requirements 26
2 NCP-in-a-Transparent-Box and Front-end 28
2.1 Overview 28
2.2 Common Components 29
2.3 Nation-Specific NCP Components 30
2.3.1 Building Blocks of the National Connector 30
2.4 NCP-B Internet Front-end 31
2.4.1 System Overview 32
2.4.2 Security 32
3 Components High Level Design 34
3.1 InboundProtocolTerminator 35
3.2 OutboundProtocolTerminator 36
3.3 WorkflowManager 37
3.4 Security Manager 39
3.5 TransformationManager 40

3.6 TerminologyServicesAccessManager 46
3.7 AuditTrailWriter 50
3.8 AuditRepository 51
3.9 RoutingManager 52
3.10 ConfigurationAndMonitoringManager 52
3.11 NationalConnector 53
3.12 Automatic Data Collection 55
4 NCP Components Interactions 56
4.1 epSOS Patient Identification Service 56
4.2 epSOS Patient Service 58
4.3 EpSOS Order Service 63
4.4 EpSOS Dispensation Notification Service 65
4.4.1 Initializing a Dispensation 65
4.4.2 Discarding a Dispensation 67
4.5 EpSOS Consent Notification Service 67
4.5.1 Providing a Consent 67
4.5.2 Revoking a Consent 69
5 Central Services 70
5.1 Central & shared service goal and responsibilities 70
5.1.1 Common data objects 70
5.1.2 Implementation overview 71
5.2 MS-List service 71
5.3 Configuration Repository Manager 71
5.3.1 Functionalities and responsibility 72
5.3.2 Architecture and interfaces 73
5.3.3 List of configuration data needed by NCPs 74
5.4 eCRTS 75
5.4.1 Creation of national translation / transcoding respoinsibilities and technical aspects 76

5.4.2 eCRTS architecture and interfaces 79

5.4.3 epSOS Terminology generation and management 83

6 Guidelines to define the Low Level Design 85

7 NCP-in-the-Box Reference Deployment Composition 86

7.1 Navigation Paths 87

7.1.1 NCP-B Role 87

7.1.2 NCP-A Role 88

7.1.3 Central Common Service Role 89

7.2 Deployment 89

7.3 Implementation Platform 90

7.4 NCP Front-end Portal 91

7.4.1 Country B Portal 91

7.4.2 Country A Portal 91

7.4.3 Functions 91

8 NCP Implementation Strategy 94

8.1 NCP implementation: Common Components and National Infrastructure Integration 94

8.2 Implementation scenarios (X, Y, Z) 96

8.3 epSOS / MS procurement procedures and Call for Invitation 102

8.4 Overall implementation planning 104

8.5 Industry team Role: Component Catalogue 106

9 Fraunhofer ISST / ELGA / Industry Reference Implementation 108

9.1.1 Baseline Assumption 108

9.2 Separation of Responsibilities 108

9.3 Commonly Developed Components Activity Provision 110

9.3.1 Consent Manager 110

9.3.2 Configuration Manager 111

9.3.3 Audit Manager 112

9.3.4 Transformation and Taxonomy Service Access Manager 113

9.3.5 Security Manager / TRC STS 115
9.4 Closed-Source Components / epSOS Backbone 116
9.5 Workshops 116
9.6 epSOS Liaison / epSOS Alignment 117
9.7 Change Proposal and HLDD-Delta in CCD 118
10 Conclusions; lessons learned 120
10.1 Open Issues 120
Appendix A FET Solution 123
Appendix B Semantics 124
B1 epSOS Semantic Implementation Guidelines 124
B2 MVC / MTC 124
B3 epSOS Central Reference Terminology Server (eCRTS) 124

1 Introduction

1.1 Background

The overarching goal of epSOS (Smart Open Services for European Patients) is "to develop a practical eHealth framework and ICT infrastructure that will enable secure access to patient health information, particularly with respect to a basic Patient Summary and ePrescription, between European healthcare systems." [1]

The infrastructure will demonstrate the principle of interoperability between two or more Electronic Health Record Systems, allowing the exchange of computer interpretable data and human understandable knowledge.

- epSOS has identified means of interoperability which will allow to connect services and architectures, potentially different in every Member State (MS), to provide Patient Summary (PS) and ePrescription (eP) cross-border services.
- epSOS has defined, and it is currently developing and testing services to allow a patient from country A while being in country B, to exploit eP and PS services available in country A.
- epSOS Large Scale Pilot (LSP) is based on legal, functional and technical pre-requisites that represent the pillar for the definition of the services and the architecture of epSOS.
- epSOS LSP will provide solutions and validate them with Pilots for Cross-border interoperability of eP and PS
- epSOS LSP will not develop European eP / PS services
- epSOS eP and PS Cross-border interoperability are based on already existing National eP and PS services,
- epSOS eP and PS Cross-border interoperability should not interfere with National eP and PS services or request their modification,
- epSOS eP and PS Cross-border interoperability must be based on a Legal and Regulatory framework which includes the signature of contractual agreements among the Member States to commit to their legal responsibilities and to assure the adequate level of trust.

Those basic pillars are representing the compulsory, binding boundary conditions for all the activities in WP3.9: "Proof of concept implementation".

[1] Annex I, Ch. B1.1.2.1

The objective of WP3.9 is to develop the proof of concept system for the pilot phase taking into account the whole set of specifications approved in the PD 3 WPs, the legal requirements, the information on selected pilots. This WP will ultimately allow the MSs to ensure the selection of Vendors, including SMEs, that will be in charge for the pilot phase implementation. This WP will entail activities at both MS and EU level as consequence of the common approach followed by the project.

In parallel to the proof of concept building process, the test plan and tools definition will be performed, starting with the identification and the selection of the methodology to be proposed and, when approved, testing tools will be defined for the main epSOS elements.

The set of specifications will be described and published for implementation purposes. More specifically it will be necessary to identify procedures, methods and instruments for making available the specifications at large. The specifications will be primarily used by companies, including SMEs, willing to participate in a “call for invitation” that is aimed at selecting organizations which are able to develop IT systems needed for carrying out the pilot phase at MS/local level.

1.2 **WP3.9 Methodological approach**

Within this chapter, a summary of the adopted methodological approaches adopted in WP3.9 will be summarised.

WP3.9 activities can be grouped under three main workstreams:

- Proof of Concept implementation
- Semantic interoperability
- Test strategy definition and test tool development

All the three workstreams have been addressed according to three phases, not necessarily fully sequential, but heavily overlapped:

- Strategy definition: performed by identified core-teams, composed by specific experts, other WP Leaders, the Technical Project Manager, the Project Co-ordinator, with the goal to build strategic approach, to identify pros/cons, to present it to the TPM, PEB, PSB. The approved strategy is actuated to allow the execution of the subsequent phases. The phase, in some cases, has included the definition of specific contracts with Suppliers or activity/funding assignment to beneficiaries.
- Specification: performed by enlarged team of experts, generally from most of the WP Beneficiaries. The goal is to provide MS developers and MS suppliers with the needed specifications for the implementation. The draft specifications are submitted to the whole WP for approval, following the consolidated epSOS methodologies and tools.

- Implementation: performed by either all or specific beneficiaries or by selected Industry Team Members. The goal is to make available data, components, tools for the epSOS Proof of Concept creation and testing.
- The implementation phase, in all the three workstreams, has included the definition and the technical monitoring of the activities performed either by specific beneficiaries or external suppliers, regulated by agreements or by contracts assigned by the PSB, ratified by the epSOS Co-ordinating Beneficiary.

A short summary of the major action performed per stream and per phase will be provided.

1.2.1 **Proof of Concept implementation workstream**

Strategy definition:

- Definition of the Pilot scope, identifying the service characteristics, among possible alternatives proposed in the functional and technical specifications of WP3.1 – WP3.7, making choices to allow the practical implementation of the Proof of Concept
- Definition of the Implementation strategy: this topic is multi-folded:
- Build and support the decision process to allow PEB / PSB to decide if the NCP has to be developed as a customized solution by every MS, or as a common development.
- PSB decision was to try to maximize the components in common, building the NCP-in-a-Transparent box, that could be used by MSs, both as a single box, or adopting specific components only
- Create the WP3.8 – 3.9 Joint Working Group to optimize the effort and the competences, including the Industry Team, to specify the NCP-in-a-Transparent-Box both at high level and low level detailed technical specification. All PD3 WP Leaders have been involved in the Joint Working Group.
- Define the implementation scenarios, to allow the MS to choose the most suitable one, according to its own pre-existing infrastructure, its eHealth strategy, its implementation policy and attitude, its procurement strategy
- Define and perform the call for invitation for implementers of the NCP solutions and implementation scenarios
- Define the criteria for selecting offers and perform the evaluation process to allow PEB / PSB to select the solutions to be adopted and funded.
- The Fraunhofer ELGA Team (FET), composed by Fraunhofer Institut, ELGA and five Vendors from the Industry Team, was assigned with the task of implementing the NCP-in-a-Transparent-Box by the PSB.

- Definition of the Liability and Accountability model: provide a methodological and technical support to WP2.1, Legal requirements, to identify the technical and medical legal responsibility in the implementation process and in the operation phase. The results have been incorporated into WP2.1 deliverables.
- Connected to this activity, the Central Service implementation strategy and legal impact (in conjunction with TPM and WP2.1) was performed.
- Define the Piloting Strategy: provide a methodological and technical support to TPM core team, composed by the WPLs of WP1.2 Evaluation, WP2.1 Legal Requirements, WP3.8 Guidelines, WP3.9 Implementation WP4.1 Pilot site selection, WP4.2-4.3 Pilot implementation, WP4.4-4.5 Pilot Operation, to define the Pilot implementation and operation strategy, optimizing the aforementioned activities and defining the Pilot implementation, deployment and operation flow.

Specification:

- The JWG core team, composed by WPLs and technical experts, in strict co-operation with TPM, has gathered, analysed and proposed solutions to the open issues, in order to come to the definition of the Pilot Scope. The leadership of the JWP was given to the WP3.8 Leader and to the WP3.9/3.10 Leader. Subsequently the WP3.8 - WG B Technical Specification Guidelines Leader took over WP3.8 WPL role.
- The JWG, starting from technical specifications of WP3.3 – WP3.7, has defined the epSOS High Level Design Document (HLDD), consolidating epSOS architecture, including NCP, Central Services and Front-end Portal, identifying and specifying the components that could be developed in common. The HLDD has represented the basis for the implementation offer from Industry Team members and for evaluating the received offers.
- The specification cycle was completed in conjunction with the Fraunhofer ELGA Team, defining the Detailed Technical Specifications. FET role was to provide the detailed specification of the implemented components. JWG experts have provided support during the specification and performed the validation against the HLDD requirements. HLDD and detailed technical specifications are gathered in this document.
- The return of information toward WP3.8, to finalise the Guidelines and WP4.2 to assure the knowledge diffusion among the MSs through the SPOCs (Single Point Of Contacts) for implementation purposes, was assured.

Implementation:

- In application of PSB decision, the Fraunhofer ELGA Team was appointed with the task of implementing, documenting, testing the NCP-in-a-Transparent-Box and the Front-end Portal for Country B. Every specified

Component in Common will be released by FET according to a timing that will allow MSs to develop, integrate and test the NCP in time to participate to the Projectathon, the interoperability testing event (see the Testing workstream).

- The MS who decided to adopt the NCP-in-a-Transparent-Box, will develop the so called National Connector, to interconnect the National Infrastructure and the NCP through the National Interface developed by FET.
- The MS who decided to adopt the Country B Front-end Portal, with NCP-in-a-Transparent-Box, will develop the so called National Connector, to interconnect the National Infrastructure and the NCP through the National Interface developed by FET.

1.2.2 **Semantic interoperability workstream**

The Semantic workstream represent the continuation of the WP3.5 Semantic Services. In order to assure the continuity, WP3.5 Leader has been appointed as Semantic Stream activities within WP3.9.

The WP3.5 approach to address in parallel syntax, terminology and semantic services was inherited by the WG Semantic. Within this organisation, WP3.5 WG Terminology Leader was confirmed as epSOS Terminology Responsible.

Strategy definition:

- Extend as much as possible the involvement of experts, developers and the MS representatives: the WG Semantic includes about 140 members, was gathering members of WP3.1, 3.2, 3.5, 3.8, 3.9, 4.2, experts of the new MS. In addition, a Google Expert Group was created including worldwide experts, external to epSOS, in order to validate the technical choices on epSOS CDA implementation.
- Define the approach to generate, manage and maintain Master Value Set Catalogue (MVC) and Master Translation/Transcoding Catalogue (MTC): centrally or by every MS.
- PEB/PSB decision was to manage MVC and MTC generation using a centralised tool. CareCom HealthTerm system was selected by PEB/PSB as centralised system to manage MVC/MTC. CareCom was appointed as technical responsible of the central tool.
- Contracts were defined and signed centrally between SALAR and CareCom, to get assistance and support from CareCom to create MVC. MS level Contracts at MS level with CareCom, were defined and signed, to get the support to handle MS level Code Systems for translation and transcoding.
- Proactively involve MS in the definition and validation of the Terminology used to transfer semantic meaning through ValueSet translation and transcoding. MS Terminology Responsible persons have been identified,

under the centralised monitoring of epSOS Terminology Responsible, in charge of managing the MTC for their MS.

- License Agreement definition to be allowed to use Code Systems in epSOS: technical support was provided to epSOS Management, to define and obtain preferential License Agreement from the Standardisation Bodies to be allowed to use/translate the selected ValueSets. **A Memorandum of Understanding for the free use of the Coding Systems for the duration of the pilot in epSOS was defined and signed by epSOS and the Standardisation bodies**
- The validation and maintenance procedures of MVC and MTC were defined. A core team of CDA experts was created to help develop the syntax specifications of the pivot documents (Patient Summary, ePrescription and eDispensation) into an implementation guide that can be used by the MS developers to create epSOS CDA pivot documents.

Specification:

- Starting from D3.5.2, Semantic Service, Appendix C: Pivot Document Specifications, the epSOS CDA Schema was generated. An implementation guideline was produced (see Appendix B1). Specifications in D3.9.1 Annex B1 superseeds D3.5.2 Annex C.
- A Quality Assurance process was put in place to validate MVC generated by WP3.5, with the involvement of all MSs. MVC 1.2 was released to MS for the creation of MTC 1.0. It is foreseen that, all requested fine tuning of MVC, derived from the MTC creation, will be consolidated, after Quality Assurance in MVC2.0, to be used for piloting purposes.
- The NCP components performing semantic tasks were specified in JWG, by the experts of the WG-Semantic.
- Guidelines and training material to use the Central Terminology Server based on HealthTerm system, were defined and shared with terminology experts. On line and Face-to-face training sessions were organised by CareCom

Implementation:

- Sample CDA documents were generated by MS implementers, applying the defined guidelines. These examples have the goal to understand epSOS CDA, in order to be able to build the National Connector to generate the document in the input format to the NCP. They are also used as reference for Test Document generation in WP3.10.
- MS level MTC are implemented by the MS terminology experts, using the Central Terminology Server or uploading into it pre-existing translations of the epSOS ValueSets and / or transcoding the coding systems in use in the MS. NCP Semantic components are developed by FET, with the support of the experts who defined their functions

1.2.3 **Test strategy definition and test tool development workstream**

The epSOS Annex I to the Contract: "Description of Work" has reorganised WP3.9 and WP3.10 Testing activities, in order to avoid potential duplications, reducing overhead and increasing synergies.

Test strategy and testing tool developed was consolidated in WP3.9 and documented in the deliverable 3.9.2

Test data generation and testing execution were assigned to WP3.10.

We provide here a summary, in order to give a full picture of WP3.9 related activities.

Strategy definition:

- Definition of the overall Testing Strategy: the goal of the activity is to agree on the global methodology to be applied to perform the interoperability testing of the NCP implemented by the MS. Alternatives submitted to PEB/PSB where every MS develops and perform its test, or test tools and data are generated by the Project. A further proposed decision was if epSOS Beneficiaries would have developed tests, or a specialised expert entity would have performed most of the activities. The core team defining the overall testing strategy was managed by Gottfried Heider, in strict co-operation with Charles Parisot.
    - PEB/PSB decision was in favour of developing test tools by the Project.
    - **IHE Europe, epSOS Beneficiary, was assigned the task to define, develop the testing tools and simulators**, since the majority of profiles to be tested are IHE profiles. IHE Europe activities for testing were assigned to WP3.9/3.10 for their technical management.
    - Within the definition of the overall Testing Strategy, the way in which interoperability testing was analysed:
- NCP Interoperability testing will be performed in specific Connectathon, organised and managed by IHE Europe, called Projectathon.
- Projectathon will not be limited to epSOS MS, but open to vendors, and other entities willing to qualify products and systems according to epSOS specifications
    - It is planned to organise two Projectathon; one in November 2010 in Slovakia and one in April 2011 (probably in Italy, in conjunction with theEupean Connectathon), to allow gradual inclusion of MSs and services in epSOS LSP.

Specification:

- Definition of testing methodology (also called “epSOS Testing Strategy”), identifying test phases, test targets, responsibilities, in line with the overall testing strategy. The testing methodology also includes guidelines to MS on how to perform integration, testing and post-deployment testing. The test strategy activity was managed by Steven Sampson.
- Definition of test tools and simulator: the activity aims to optimise the development of testing tools, by identifying the re-usability of already existing tools, developed by IHE. Gap analysis between standard IHE Profiles and epSOS defined profiles in WP3.3, 3.4, 3.5, 3.6 was performed, trying to minimise the delta. The activity was performed by IHE Europe and the aforementioned WP Leaders.
    - The decision of adopting Gazelle, the testing environment developed by IHE Europe was approved by PEB and PSB. It will be used also as remote reference testing environment.
- Definition of Test Plan: Test scenarios and profiles to be tested were described and inserted in the Gazelle system. IHE and ELGA were responsible for this activity

Implementation:

- Implementation is an ongoing activity. It is planned that all testing tools, simulators and workflow manager will be gradually released up to September 2010, according to a release plan agreed with FET and in line with MS development and testing plan. Activity is lead by IHE Europe, with the relevant contribution of epSOS experts.

1.3 Specification Methodology followed

Within this chapter, a zoom-in of the Specification methodology, followed in the Specification Stream described in 1.2.1, will be provided.

The epSOS specifications have been realized as a cooperation of different Work Packages (WP). The WP3.3 (Architecture) and WP3.4 (Common Components) are responsible for defining epSOS architecture and interoperability specifications. Requirements on security are defined by WP 3.7. The functional services performed by the epSOS architecture are defined in accordance to WP 3.1 and WP 3.2 Functional Requirements on ePrescription, eDispensation (eP/eD) and Patient Summary (PS), the Patient and HCP Identification processes (WP3.6) and the Semantic Interoperability Services (WP 3.5).

All the functional and technical requirements directly derive and must be conformant to the Legal and Regulatory Requirements and the defined and signed Agreements among the MS to set up the Circle of Trust.

WP 3.8 (MS Guidelines definition) and WP3.9 (Proof of Concept definition) have continuously and actively co-operated with all the WP leaders and the TPM,

adopting the following procedure for defining the High Level Software Design of the NCP Implementation, as described in epSOS Annex I and Deliverables:

- creation the Joint Working Group, composed by WP3.8 and WP3.9 members and the WP leaders
- definition of the Pilot scope
- identification of open issues related to implementation choices
- convergence among all the Specifications of each WP into the proposed High Level Design Document (HLDD)
- finalization of the first draft of HLDD, including the rough estimation of the implementation effort
- set up of sharing and approval process of HLDD among Beneficiaries
- creation and release of consolidated version of HLDD to be used for the definition of Implementation Guidelines (WP3.8), Detailed Technical Specification and Testing tools (WP 3.9)
- definition of the Detailed Technical Specifications for the implementation.

Set up of strict co-operation between the WP3.9 and the team in charge of the Common Component Development, to provide practical support on detailed specification interpretation and jointly fix open issues, and to validate defined and released Common Components.

The most important part in the epSOS use case is the National Contact Point gateway (further in the text referred to as NCP Implementation), the software components that are responsible for connecting national infrastructures. For the pilot phase, it was decided to identify “common software components” of the NCP gateway similar for all member states (MS) which can be developed in common.

Not all functionalities of the NCP Implementation can be fulfilled by the common software components. As a result, some of the functionalities require software components that are specific for each MS. The focus of this implementation guide is the description of the common software components.

For the pilot it was decided to build the “NCP-in-a-Transparent-Box” that is composed from the common software components that implement functionality that may be common for several MSs and are connected to nation-specific software components implementing nation-specific processes. The resulting NCP Implementations can be used by the MS to support their pilot sites.

A top-down approach was used for developing this document. An emphasis was made on understanding the general architecture, followed by a more detailed description of the common software elements (common components). This top

down approach will also be followed when describing the Detailed Technical Specifications. The document was built in several steps:

The common software components were defined

A core team of technical experts was identified and split into three functional groups:

- The architecture group defining the architecture of the "NCP-In-A–Transparent-Box".
- The semantic group
- The group providing detailed description of the common software components

Several countries have actively participated at these activities: Austria, Denmark, France, Germany, Lombardy, Slovakia, Spain, Sweden; UK provided consultancy support.

1.4 Scope, audience and purpose of this document

This document provides the information concerning the activities performed by the Proof of Concept implementation workstream of WP3.9, in co-operation with WP3.8 in the Joint Working Group WP3.8 / 3.9.

Strategic activities like implementation strategy definition and implementation proposal selections were performed in co-operation with TPM task force and PMT.

This document is heavily based on the High Level Design document, epSOS Internal Deliverable: "Architecture of the National Contact Point in a Box (NCP-in-a-Transparent-Box)", developed by the Joint Activity Group (JWC) 3.8 / 3.9.

The document is addressed to the MS and to the vendors who wants to develop the NCP, the Front-end Portal. It represent a bridge among the Specifications of WP3.3 – 3.7, the Guidelines of WP3.8, the testing procedures defined in WP3.9 and applied in WP3.10, and the MS implementation of PD4.

The primary audience of the document is project managers, system architects and developers involved in the building and deployment of the NCP implementation that is based on the NCP-in-a-Transparent-Box or reuses some of its components. The document explain the concepts of the NCP-in-a-Transparent-Box, Common Components and describes how MSs can use these for developing and deploying an NCP implementation If none of the parts of the NCP-in-a-Transparent-Box will be reused, this document can serve as a reference architecture aiding in designing an own implementation of the NCP.

After a summary of WP3.9 organisational and methodological approaches, the NCP implementation guidelines, provided by PEB/PSB are introduced.

Basic requirements for NCP implementation are analysed. High Level Design of the NCP-in-a-Box concept, both considered as gateway toward Country A and toward Country B, is provided. It contains general description of the NCP-in-a-Box architecture as well as details about individual components and Country B Front-end Portal.

This document defines the architectural view on the “NCP-in-a-Transparent-Box”, describing the functionalities which are common and the “common components” implementing these functionalities. It documents which functionality can be developed in common and which “common components” will be specified.

The “Box” is defined “Transparent” because every MS can decide if it is worth for it to adopt all or only part of the components defined as potentially common.

Furthermore, it describes on a high level the national part of the “NCP-in-a-Transparent-Box”, namely the “NationalConnector”.

In order to provide a light integration solution for those MSs who, at this moment, do not plan a full integration of the NCP Country-B functionality with their national infrastructure, the document also describes a Country B Front-end solution.

Certain functionality required by NCPs (e.g., distribution of configuration-relevant data) is to be implemented as central services and referred to as Central Common Services. The specification for the Central Common Services is not included in this document. However, the way in which the NCP will be connected to them is defined, together with the NCP component to perform the interconnection.

Component specifications are provided in the document, while Detailed Technical Specifications of some of them are reported as Appendixes.

A specific chapter is devoted to Semantic Group activity description, including the definition of Central Services for Terminology Management and processes to develop and maintain Master Value Sets and Translation/Transcoding Catalogues.

Implementation strategies and scenarios are analysed, describing the procurement procedures defined and actuated by epSOS.

The NCP in a transparent box implementation, assigned Fraunhofer ISST / Elga and IT consortium are finally described.

The implementation differs in some aspects from the overall design described in Chapter 3 - 7: this is not infringing the interoperability concepts and specification. Chapter 9 and Appendix A provide a view on differences and detailed technical specification of the implemented proof of concepts.

It was decided to keep both views in D3.9.1 to allow MSs and vendors willing to implement the NCP as defined in the High Level Design to get the appropriate specification and to provide the detailed information to the MSs willing to adopt FET implementation.

1.5 Glossary

The epSOS Glossary will not be exhaustively explained in this document. The deliverables of the preceding work packages are seen as a basis for this document. The terms commonly used in epSOS (like HCP, NCP, …) are not included in this document. Definitions reported into the epSOS official Glossary applies.

1.5.1 Terms and definitions

Central Common Services	A set of central services jointly used by NCPs.
Circle of Trust	See epSOS Glossary.
Common Software Components	Software components of the NCP-in-a-Transparent-Box
Country A	See epSOS Glossary
Country B	See epSOS Glossary
eDispensing (eD)	See epSOS Glossary.
ePrescription (eP)	See epSOS Glossary.
epSOS Central Repository Terminology Server (eCRTS)	Web tool (based on the HealthTerm system developed and maintained by CareCom) to allow every MS to develop, validate, archive and download to the NCP the MVC and MTC.
epSOS Large Scale Pilot (LSP)	See epSOS Glossary.
epSOS Master Translation/Transcoding Catalogue (MTC)	This is the epSOS Master Value Sets Catalogue which contains, in addition to the original terms, their translation in different languages corresponding to the respective Member States and the possible cross-referencing (transcoding) with other code system that are used at the national level. The translation and the cross-referencing (transcoding) is a national responsibility. Providing the content of each country's contribution to the epSOS Member States is under epSOS' responsibility.
epSOS Master Value Sets Catalogue (MVC)	Collection of terms used within certain parts of the pivot documents (either parts describing the patient demographics or the clinical problems for example) based on known code systems such as ICD-10, SNOMED CT, ATC, EDQM, UCUM, etc.
Health Care Professional	See epSOS Glossary.

(HCP)	
Health Care Provider Organization (HCPO)	See epSOS Glossary.
High Level Design Document (HLDD)	Document describing the concept and high level architecture of NCP-in-a-Transparent-Box
Member State (MS)	See epSOS Glossary
National Contact Point (NCP)	See epSOS Glossary.
National infrastructure	It represents all entities where patient or HCP or Health Care records are managed in member states.
NationalConnector (NC)	Entity that encapsulates the Nation-Specific NCP Components. The NationalConnector is implemented as a black box having its subcomponents hidden from the NCP
Nation-Specific NCP Components	The components of the NCP that will not be developed in common but are still part of the NCP
NCP gateway	A gateway system under the control of the NCP that manages all epSOS transactions and which connects the National Infrastructure (NI) to the epSOS backbone. It is a point of entry/exit to/from the NCP, acting on behalf of a HCP (at a PoC) who requires access to a patient's medical data through epSOS, or acting as service broker of an epSOS data provider
NCP-in-a-Transparent-Box	A modular set of software components (Common Software Components) intended to facilitate a NCP implementation that can be used completely or partly by any MS to fulfil NCP obligations. This implementation is not mandatory.
Patient	See epSOS Glossary.
Patient Consent (PC)	See epSOS Glossary.
Patient Summary (PS)	See epSOS Glossary.
Pseudonymised data	Patient's credentials are substituted by anonymous data to avoid the infringement of privacy requirements
Point of Care (PoC)	See epSOS Glossary.
Requirements	Definition of all relevant needs (business, functional, non functional, technical and technological) for

	system specification and implementation
Software Component	Software that can be installed or replaced only as an entity to create a scalable implementation (e.g. component can provide directory services, data storage, provisioning, etc.)

1.5.2 **Abbreviations**

ADF	Access-Control-Decision-Facility
ADI	Access Decision Information
AEF	Access-Control-Enforcement – Facility
CA	Certificate Authority
CCD	(epSOS) Common Component Design
CDA	Clinical Document Architecture
CRL	Certificate Revocation List
DAO	Data Access Objects
DOM	Document Object Model
eCRTS	epSOS Central Repository Terminology Server
eD	eDispensation
EDQM	European Department for the Quality of Medicines
eP	ePrescription
ETSI	European Telecommunications Standards Institute
FET	Fraunhofer ISST / Elga and IT consortium
GUI	Graphical User Interface
HCP	Health Care Professional
HCPO	Health Care Provider Organization
HLDD	High Level Design Document

ID	Identifier
IHE	Integrating the Health Care Enterprise – Europe
IT	Information Technologies
IU	Interaction Use
JDBC	Java Database Connectivity
JWG	Joint Working Group of Work Packages 3.8 and 3.9
LSP	Large Scale Pilot
MS	Member State
MTC	epSOS Master Translation/Transcoding Catalogue
MVC	epSOS Master Value Sets Catalogue
NC	NationalConnector
NCP	National Contact Point
NCP-A	National Contact Point of Country A
NCP-B	National Contact Point of Country B
NI	National Interface
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PAP	Policy Administration Point
PC	Patient Consent
PD4	Project Domain 4
PDF	Portable Document Format
PDP	Policy Decision Point
PEB	Project Executive Board

PEP	Policy Enforcement Point
PID	Patient Identity
PMT	Project Management Team
PoC	Point of Care
POJO	Plain Old Java Object
PoU	Purpose of Use
PS	Patient Summary
PSB	Project Steering Board
SAML	Security Assertion Markup Language
SME	Subject Matter Expert
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSL	Secure Socket Layer
STS	Security Token Service
TPM	Technical Project Manager
TRC	Treatment Relationship Confirmation
TSAM	Terminology Services Access Manager
TSL	Trusted Service List
URL	Uniform Resource Locator (world wide web address)
WSDL	Web Services Description Language
WP	Work Package
WS	Web Service
XFRM	Transformer

XCA	Cross Community Access
XDR	Cross-entreprise Document Reliable Interchange
XML	eXtensible Markup Language
XUR	Cross-Enterprise User Authentication

1.6 **Requirements**

For all NCP implementations, the requirements of the current epSOS deliverables are mandatory. These are not repeated in this document. The deliverables of all technical working packages and the legal working package WP 2.1 are relevant for the NCP design and the definition of common components.

1.6.1 **Basis Requirements**

The requirements of this section form the basis for the work of the technical core team.

- epSOS basic pillars have to be respected and safeguarded (cfr: D5.2.1: Initial Scope)
    - epSOS LSP will provide solutions and validate them with Pilots for Cross-border interoperability of eP and PS
    - epSOS eP and PS Cross-border interoperability must be based on a Legal and Regulatory framework which includes the signature of contractual agreements among the Member States to state the legal responsibilities and assure the adequate level of trust.
    - epSOS eP and PS Cross-border interoperability must be based on already existing National eP and PS services,
    - epSOS eP and PS Cross-border interoperability should not interfere with National eP and PS services or request modification to them
    - epSOS eP and PS Cross-border interoperability services must not decrease the level of security provided by every MS to its Citizens
    - epSOS LSP will not develop European eP / PS services, but make existing national services accessible from all participating member states
- HLDD is based on the legal requirements of WP2.1, the functional requirements of WP3.1 and 3.2, the technical requirements of WP3.3, 3.4, 3.5, 3.6, 3.7, applied to the defined Pilot Scope, taking into account deliverable harmonization and open issue solutions defined by JWC (including PD3 WP Leaders and TPM).

- NCP in a transparent box means that every component has a defined interface and if needed can be easily substituted by a corresponding national implementation. It is agreed upon that the NCP must be provided as a very flexible framework.
- As many components as possible should be developed in common. However, this will not be possible for all functionalities of the NCP.
- The possible strategies that might be followed to develop in common the NCP components and the Country-B Front-end will be described in the HLDD, however it is the responsibility of Project Coordinator and PEB/PSB to choose among alternatives and adopt the selected one.

1.6.2 **Additional Requirements**

To enable the development of the “NCP-In-A-Transparent_Box” additional requirements have been defined by the technical core team. These additional requirements are listed below:

- The NCP Gateway is composed of a common (epSOS) part and an individual (nation specific) part. The components of the common part are specified in detail. The nation specific part is implemented in the “NationalConnector” that is not specified in detail but implements a stable interface that is defined by the joint working group 3.8/3.9 (JWG).
- The components in the common part might or might not be adopted by the different MSs, according to their National infrastructure and the MS defined implementation policies.
- The architecture and the epSOS transactions must be able to transport pseudonymised data that was sent by country A in a way that only pseudonymised data can be seen by NCP-B.
- The individual security requirements of each MS, are documented in D3.7.2. Organizational and technical means must be defined in the deliverable to make sure security requirements are fulfilled. If security requirements cannot be fulfilled, this must be documented.
- The authentication of the health care professionals (HCP) is a prerequisite for any epSOS transaction (usually Identification Handshake). The authentication of the HCP is done by the identity provider which belongs to the national infrastructure.
- The identification request is checked in the common part of the NCP A in the way that the XML containing the patient traits is schema validated; the semantics of the traits are checked in the national part.

- The security audit that forms the basis for administrative tasks and reporting must be separated from the functional logging.

2 NCP-in-a-Transparent-Box and Front-end

This section presents an overview of an "NCP-in-a-Transparent-Box". First it presents the components of the NCP. Then interactions between the components are presented.

2.1 Overview

Deliverable 3.3.2 describes in the Information System View the structure and the high level behaviour of the NCP gateway.

The NCP-in-a-Transparent-Box is one scenario of an NCP implementation that includes special requirements (see Chapter 1.6). Therefore, the composition of the NCP is slightly different from the architecture defined in Deliverable 3.3.2. The main difference is that the components are separated into "common components" and "nation-specific components" with the nation-specific components being encapsulated by the "NationalConnector".

In order to provide a stable basis for the common components, the interface of the NationalConnector has to be defined. The National Connector is implemented as a component integrated to the Common components via well defined interfaces. They are nevertheless part of the NCP because they implement core functionality of the NCP and fulfil functional and non-functional requirements.

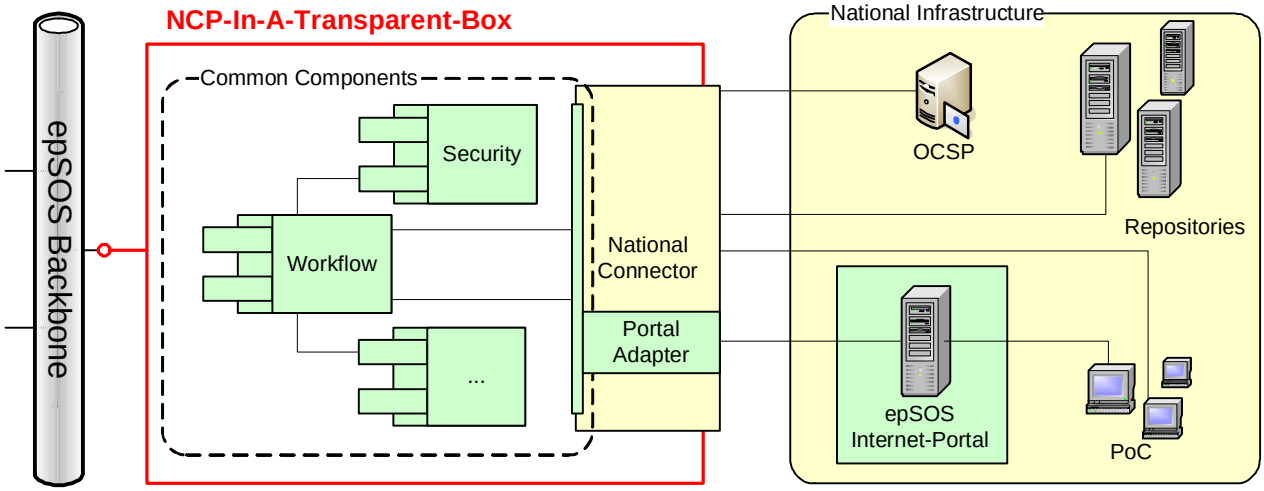


Figure 1 NCP-in-a-Transparent-Box

The NCP-in-a-Transparent box is meant to be a flexible framework of common components that may be used by MS to fulfil the requirements of their pilot sites and epSOS regarding the epSOS transactions. Figure 1 shows by color which

components MAY be developed in common (green) and which are specific for each member state (MS).

Each member state is responsible for its own and unique implementation of the NCP gateway. We assume that for reducing the costs, each MS will try to use as many common components as possible. Nevertheless a reasonable part of the NCP gateway remains nation specific and therefore cannot be specified nor jointly developed by epSOS. The focus of this chapter lays on the common components that are defined in section 2.2.

Common central services are not shown in that abstract figure but they will be connected to the epSOS Backbone.

The national connector encapsulates the nation-specific part of the NCP hence it is not a common component. Nevertheless it has an interface that is defined in common to achieve interoperability between the common components and the national part.

2.2 **Common Components**

The following components have been identified by the technical core team:

- InboundProtocolTerminator
- OutboundProtocolTerminator
- SecurityManager
- TransformationManager
- TerminologyServiceAccessManager
- WorkflowManager
- AuditTrailWriter
- RoutingManager
- ConfigurationAndMonitoringManager

The following components were discussed but have been discarded

- Time synchronisation is done by the platform on which the NCP is deployed, therefore there will be no dedicated component for that task
- There will be no dedicated LoggingManager component since the logging will be processed by a logging framework. The logging that is needed for reporting and administrative tasks has to be separated from the security audit.

- There will be a management console that is used for administrative tasks, for monitoring and for reporting. The management console will be able to read the logfiles and will be able to provide the configuration data.

The definition of the components is very close to the basic architecture presented in Deliverable 3.3.2. There are, however, a number of differences that is a consequence of new requirements stated by the JWG. On the one hand, the architecture of the NCP-in-a-Transparent-Box should allow having as much of commonly developed components as possible. On the other hand, the architecture should be flexible enough for supporting the variety of national solutions implemented by MSs.

2.3 Nation-Specific NCP Components

The term “Nation-Specific NCP Components” in this document is used to identify the components of the NCP that will not be developed in common but are still part of the NCP.

The concept of NCP-in-a-Transparent-Box bundles these components in the National Connector (NC). The NC connects the national infrastructure to the common components of the Workflow Manager of the NCP. It has to implement a defined and stable interface (Appendix A). While the interface must be common the implementation can vary according to the needs of the national systems.

The National Connector (NC) of NCP-B MUST guarantee the correctness of the HCP authentication claims that are assigned to the National Interface (NI)

The National Connector of NCP-B MUST guarantee that any CDA documents obtained from the National Interface are forwarded only to the HCP that was authenticated as issuer of the request.

2.3.1 Building Blocks of the National Connector

The following suggestions can be made to the building blocks of the National Connector. It is likely there will be the need of components that encapsulate functionality as described in Figure 2.

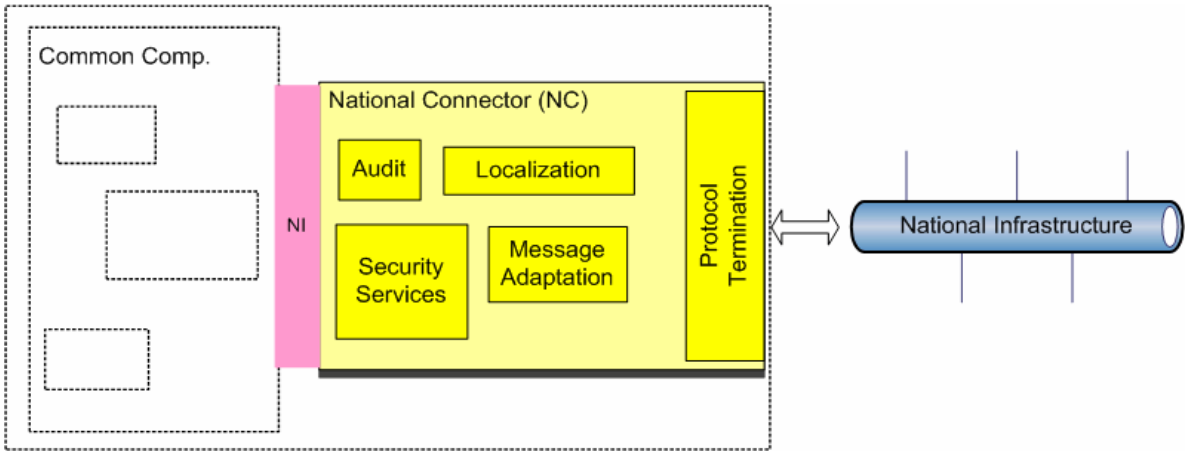


Figure 2 - National Connector building blocks

The shown building blocks are suggested to have the following meaning:

- Protocol Termination: The NC will have an interface to the national Infrastructure and needs a building block to terminate the protocol used in the national infrastructure.
- Security services: Assuming that certificates are used to verify the authenticity of holders of signature and encryption keys this component is for example responsible to validate certificates according to the national policies.
- Message Adaptation: If a tight integration with the national systems is planned it may occur that the message sequences on the national side do not directly map to the corresponding epSOS sequences. For example it may be that certain information in the national infrastructure is transmitted in split messages while in epSOS only one message is used. Another example is that information is distributed and must be consolidated. If that is the case a component is needed which manages the mapping of the message sequences.
- Localization of medical data: the NCP will have to support a nation-specific architecture that implements localization and storage of medical documents.
- Auditing: There will be the need to design and build a component which provides functionality to read audit logs according to the national requirements and legislation. In addition there may be special requirements on the national side which produce the need for an extended auditing.
- Access control (part of the security services): A structured approach for the component dedicated to access decision control has proved itself as beneficial. Thereby the component is composed of the following subcomponents:

- The PEP (Policy Enforcement Point) controls every data access and thereby makes sure that no access is possible without a former authorization check.
- The PDP (Policy Decision Point) is responsible to provide the decision whether access is granted or not. It makes a decision based on the PIP (Policy Information Point).

Dependant on the architecture and design of the national infrastructure there may be more building blocks needed but the denominated are the most likely.

2.4 NCP-B Internet Front-end

The NCP-B Front End will provide a simple interface for retrieving Patient Summaries (PS), ePrescriptions (eP) as well as for issuing eDispensations (eD). The existence of such a front-end can significantly reduce efforts related to piloting of country B epSOS use cases. For country A use cases no front end can be used since in this case the NCP-gateway has to be integrated with the national infrastructure.

HCPs acting as a PoC in a country B shall have the option to use the web portal to access epSOS services. Strictly speaking, this is part of the national infrastructure, but as that part of the NCP is useful to a number of MS, it will be provided as part of the NCP-in-a-Transparent-Box.

2.4.1 System Overview

The NCP-B Front-End SHOULD be based on the enterprise java (Java EE) referable reduced version containing the necessary part for Java Servlet Pages/Java Server Faces (JSP/JSF) execution (recommendation), but this can be change with justification. It MUST support the Services for Patient Identification (ID), Patient Consent (PC), patient Summary (PS), ePrescription (eP) and eDispensing (eD); consuming the first four Services and publishing/sending/pushing the latter.

The interface with/to the NCP MUST go through the NCP Portal Adapter, that decouples the country-B web portal and the core NCP (by this NCP and portal can be deployed in different security zones and multiple, user-group specific portals can be connected to a single NCP). Communication to and from the Front-End to NCP need the Portal Adapter, because the National Connector is unknown (national specific), and therefore to have a common/unique interface between the two systems, a portal specific adapter is needed.

The connectivity between the NCP Portal Adapter and the NCP-B Front-End can be java-to-java technology, Plain Old Java Object (POJO), but it MUST be so, that the Front-End and the Portal adapter can be separated by network zone and physical execution environment. The transportation of POJOs can easily be facilitated with the Spring Remoting framework or equivalent.

The System overview below is informative.

2.4.2 **Security**

Whenever application are used, the web application MUST use Server-Side session, which are invisible to the client. Furthermore the application SHOULD use standard web-server security measures.

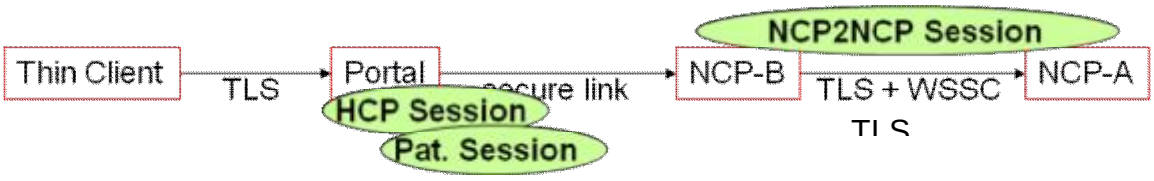


Figure 3 Security of Front-End solution (portal)

The user browser access to the portal MUST be only acceptable by HTTPS/TLS with at least 128bit encryption. This MUST be implemented with a MS specific certificate which is certifiable (can be checked by national certificate authorization (CA)).

The portal MUST use a secure link to national side of NCP, which MUST be terminated in the NCP Portal Adapter

The portal MUST guarantee that any medical document is only accessible by the HCP that was authenticated by the patient and by NCP A to request this document.

3 Components High Level Design

This section contains the individual description of the components and a further decomposition of the solution for the NCP.

The Detailed Level Technical Specifications of the components are contained in the Appendix A. References to corresponding sections of the appendix are within high level design descriptions.

An overview of the components is given in Figure 4 Common Components.

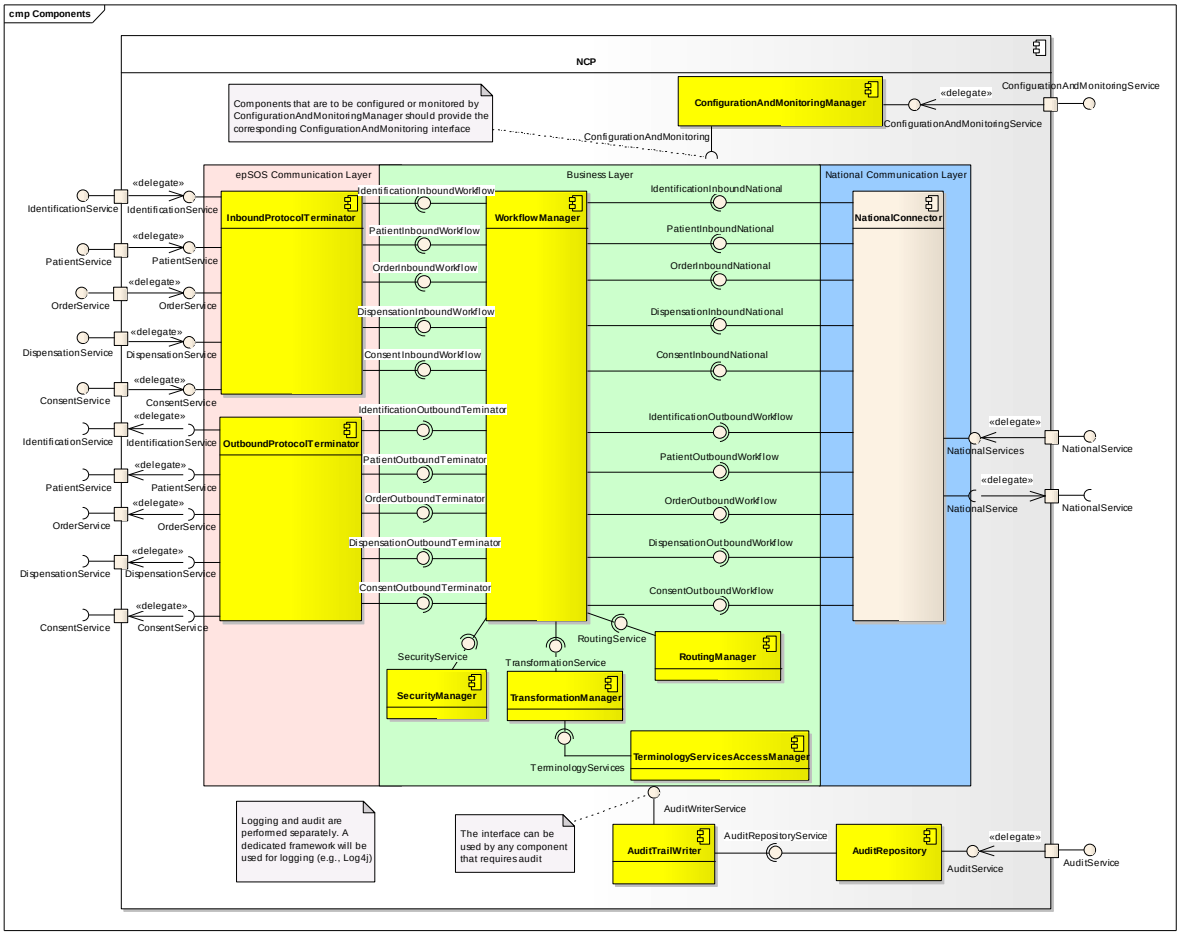


Figure 4 Common Components

Components that are not common but part of the NCP gateway are grouped in the NationalConnector.

The basic concepts of the architecture are derived by the logical sectioning of the NCP gateway into three sections (vertical layers). The workflow manager is the component which is responsible for managing communication between the layers.

The high-level architecture described in this section serves as a reference architecture and its purpose is to serve as a guideline for the actual NCP implementations. As a consequence, the actual implementations deployed by MSs can derive from the design proposed here. Regarding the jointly developed NCP-In-A-Box, its architecture SHOULD be conforming to the architecture presented here. Deviations are possible, but should be approved by the authors of the HLDD and this deliverable.

3.1 InboundProtocolTerminator

The *InboundProtocolTerminator* plays the role of epSOS web services provider. The service is defined with a set of epSOS WSDL and a stack of security protocol such as: SSL, SAML and others (see below). These protocols have to be terminated for any incoming message in the *InboundProtocolTerminator* component. The component implements the endpoint of the epSOS services.

The *InboundProtocolTerminator* checks signatures of SOAP requests initiated by the *OutboundProtocolTerminator* of another NCP, performs deserialization of these requests into Java objects and passes them to the *WorkflowManager* by calling an appropriate operation of the corresponding interface. Once the *WorkflowManager* returns the result of the call, the *InboundProtocolTerminator* serializes it into a SOAP response, signs the response and sends it to the NCP from which the SOAP request originated.

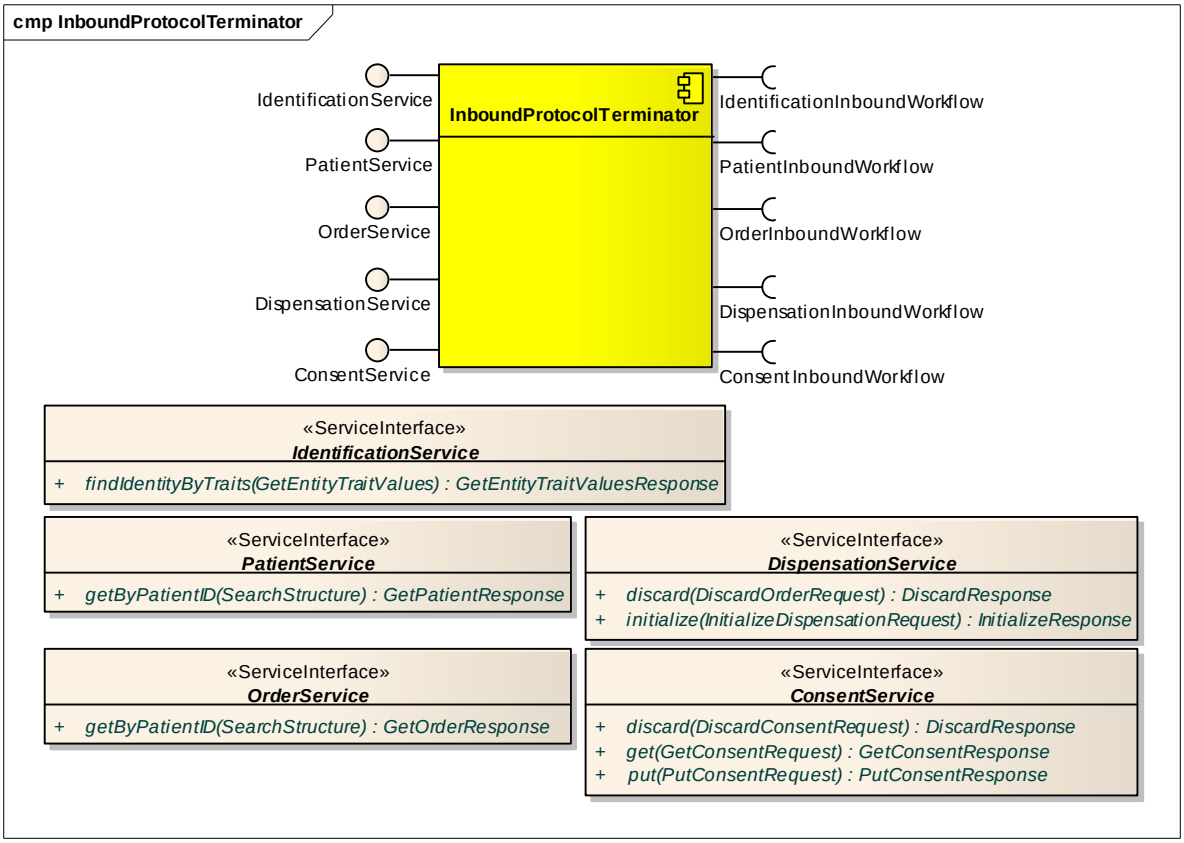


Figure 5 InboundProtocolTerminator component

The detailed specification of the epSOS services provided by this component is provided in D3.4.2.

3.2 OutboundProtocolTerminator

The *OutboundProtocolTerminator* plays the role of epSOS web services consumer. The component serializes Java objects passed as to it by the *WorkflowManager* into SOAP requests, signs the request and transmits them to the remote *InboundProtocolTerminator* of another NCP by calling an appropriate operation of one of its service interfaces. When the response arrives, it is the responsibility of the *OutboundProtocolTerminator* to validate signatures of the response, deserialize the response and pass the resulting Java object to the *WorkflowManager.*

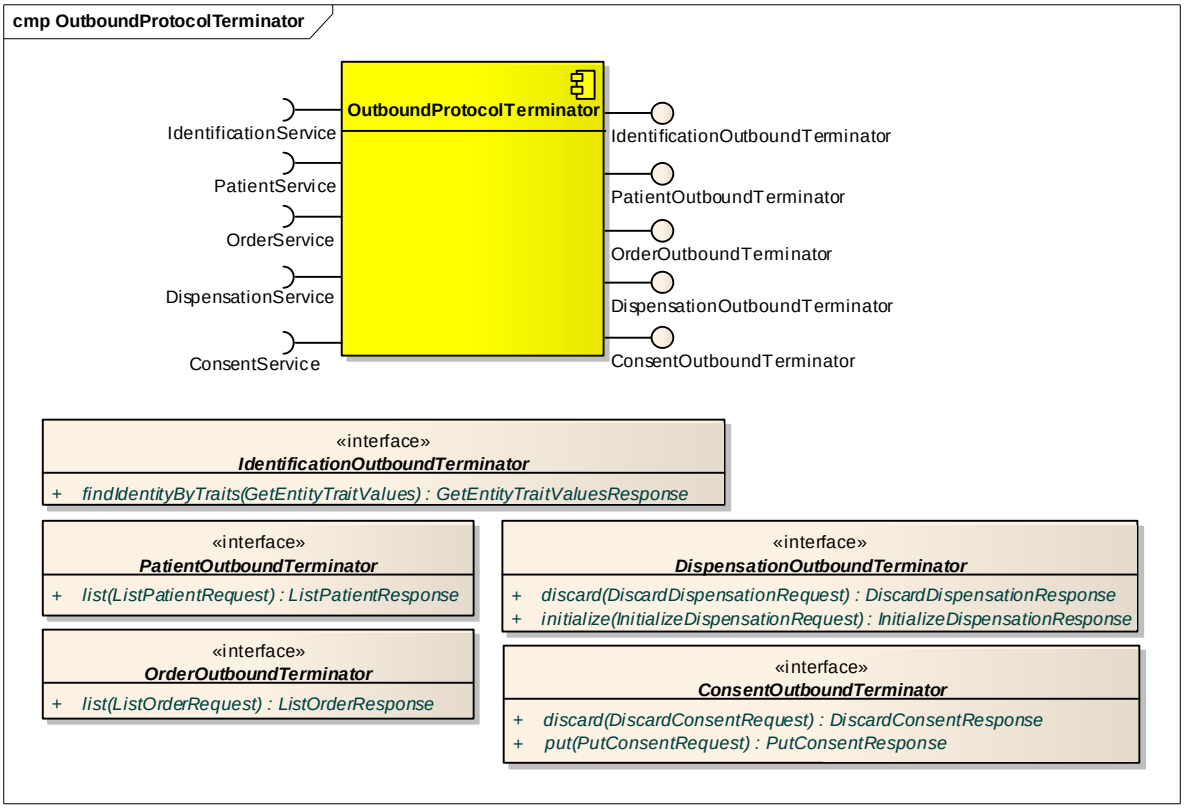


Figure 6 OutboundProtocolTerminator component

The detailed specification of the epSOS services consumed by this component is provided in D3.4.2.

3.3 **WorkflowManager**

The *WorkflowManager* is called from the *InboundProtocolTerminator* as well as from the *NationalConnector*. This component realizes a Process Manager pattern. It is the entry point into the business layer of the NCP. Therefore this component is the first to be called after a message is received and deserialized. The *WorkflowManager* acts as orchestrator and realize the chain of operations call. The operations are exposed by interfaces of others components of business layer and, at the end, the result will be passed to the *OutboundProtocolTerminator* or to the National Connector.

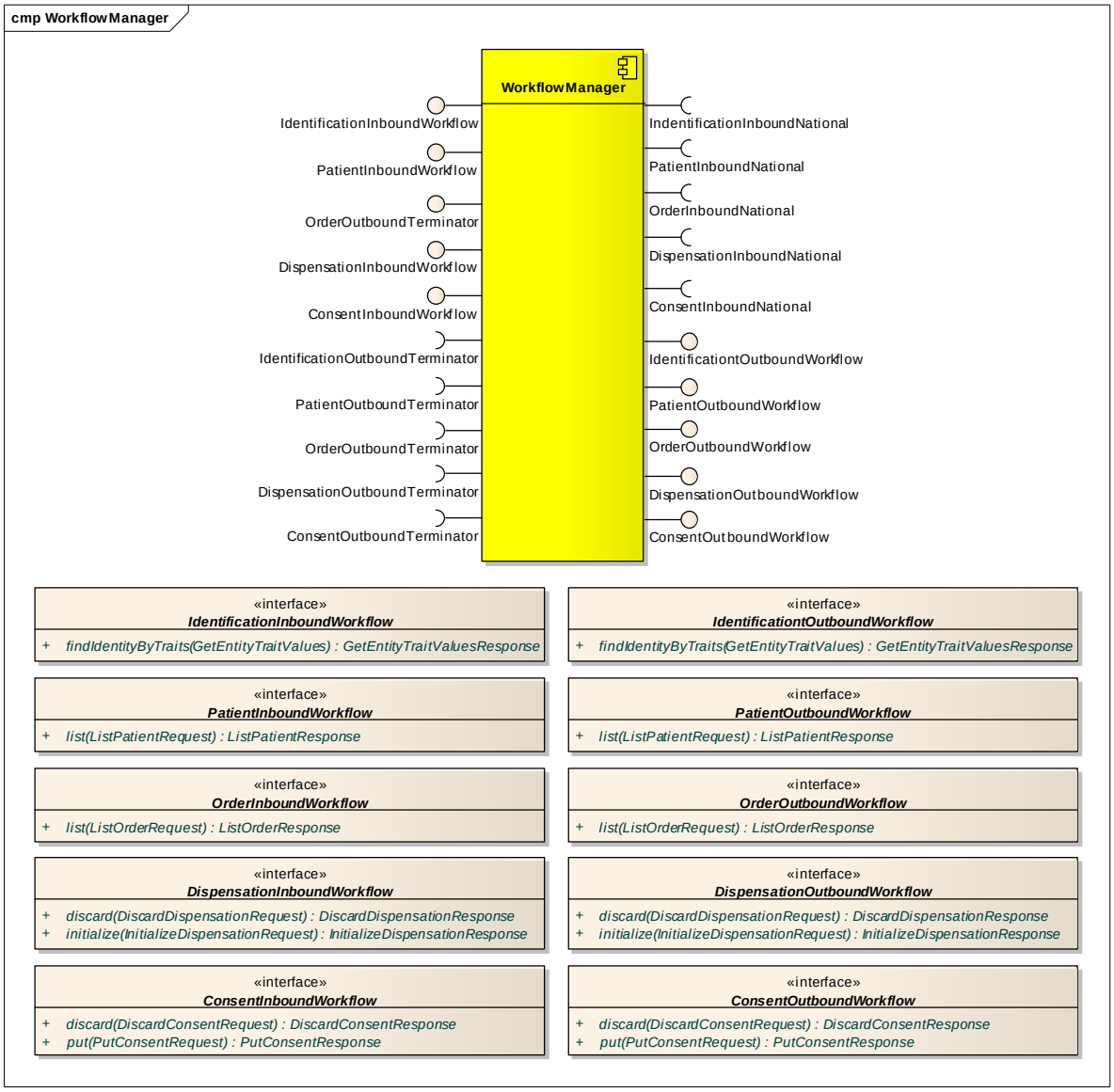


Figure 7 WorkflowManager component

The *WorkflowManager* has 2 possible modes:

a) Dispatch to configured functions. This option sees a sequencer controlled by a configuration table, which will call standard functions, passing the message object to business functions with generic interfaces.

b) Dispatch to registered functions. This approach allows components to register as handlers for specific message types, that itself will call business functions with specialized interfaces.

For the time being, approach b) was chosen as being more flexible and simple.

In the actual implementation this component will be provided as a closed-source part of the Tiani Product.

3.4 Security Manager

The *SecurityManager* is used for certificate validation and XML-Signature creation and validation. It is mandatory that the security manager has a list of all trusted certificates to check whether the given certificate is member of the circle of trust. The certificate validation includes the mathematical check, the check of the validity in time and the OCSP call.

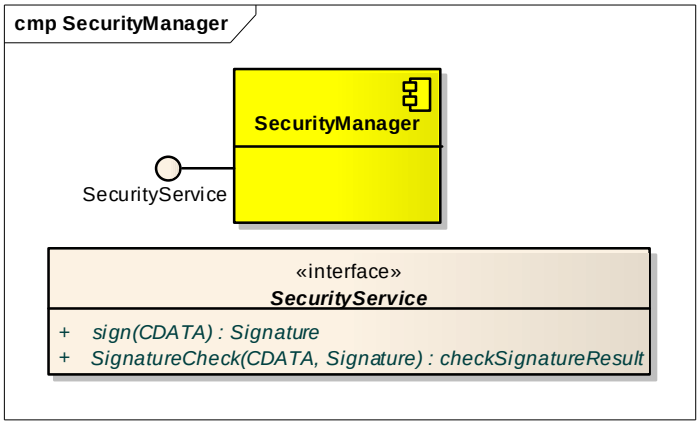


Figure 8 SecurityManager component

sign(CDATA) : Signature

Textual description of operation	
This operation processes a XML DSig signature for a given XML according to the XML DSig standard and the epSOS profilation. Only known documents are signed and before the signature processing a schema validation is done. The documents that are known are configurable.	
Input parameters	CDATA XML Document to sign
Output parameters	Signature
Notes	In case it is decided that it is enough to provide WS-signatures, this functionality can be discarded

CheckSignature(CDATA, Signature) : CheckSignatureResult

Textual description of operation	
The XML DSig signature of a XML document is validated including the validation of the certificate that confirmed the signature.	
Input parameters	CDATA XML containing Signature Digital signature
Output parameters	CheckSignatureResult Encoded in the status information are • Validity of signature • Validity of certificate
Notes	All the known and trusted certificates have to be registered by configuration beforehand. In case it is decided that it is enough to provide WS-signatures, this functionality can be discarded

The detailed description of the component can be found in Appendix A.

3.5 **TransformationManager**

This component will be called by the *WorkflowManager* in two different scenarios: either for data transformation from a national language to the epSOS Reference Terminology or for data transformation from the epSOS Reference Terminology to a national language).

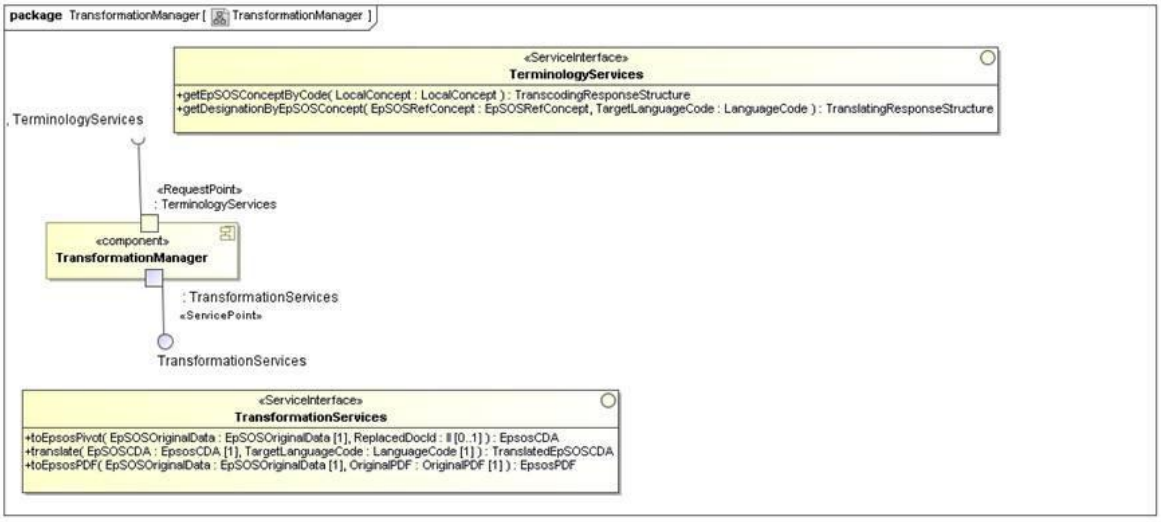


Figure 9 – Component : TransformationManager

Component responsibilities

- Translating and/or transcoding (if necessary) the original data compliant to epSOS CDA syntax from the national language and possibly from the national code system(s) in the document creator country (in most cases Country A) to the epSOS Reference Terminology. From a functional point of view the translation and transcoding are the same operation for the Transformation Manager from the point of view of the document creator country (in most cases Country A).
- Creating an epSOS unstructured CDA by embedding the original data from document creator presented in the pdf format. Note: the assumption is made that the original data is already presented in a pdf format. The document creator country may have other formats internally; however this data must be presented in a pdf format so that the document consumer country can read it. If, for whatever reasons, the original data needs to be sent, it can be processed in a similar way. However, for the pilot purposes only the pdf operation is defined.
- Translating the coded data elements from the epSOS Reference Terminology to the national language in document consumer country (in most cases Country B).

`ToEpSOSPivot(EpSOSOriginalData; ReplacedDocId) :EpsosCDA`

Textual description of operation
Transformation of national data to epSOS pivot format.

Input parameters	EpSOSOriginalData Medical document in its original data format as provided from the NationalConnector to this component. The provided document is compliant with the epSOS pivot CDA (see Appendix B1) unless the adoption of the element binding with the epSOS reference Value Sets. [Mandatory] **ReplacedDocId** is an instance identifier describing the document to be replaced.
Output parameters	**EpSOSCDA** structure Response structure including the epSOS pivot CDA and the response status structure. The response status structure provides information about the operation results, including possible errors and warning.
Component behaviour	After having received a *toEpSOSPivot*() request, this component takes the EpSOSOriginalData (already compliant to epSOS CDA syntax) and using the TSAM capabilities, accomplishes the eventual transcoding of the terms present in the epSOS value sets, while also keeping the original codes and display name. An epSOS pivot document with epSOS coded concepts is therefore produced. The epSOS pivot document shall to have a link to the EpSOSOriginalData. *Exceptions*: in case of processing error or warning, the responseStatusStructure will be used to convey this information to the calling component with an appropriated error and warning code. A detailed list of the managed exceptions will be provided in the Detail Design Specification document. Each exception condition occurred will be logged (standard and audit), reporting both the exception code and its English description.
Notes	

ToEpSOSPDF(EpSOSOriginalData; OriginalPDF) : EpSOSPDF

Textual description of operation	
Transformation of national data to epSOS pivot format.	
Input parameters	EpSOSOriginalData Medical document in its original data format as provided from the NationalConnector to this component. The provided document is compliant with the epSOS pivot CDA (see Appendix B1) unless the adoption of the element binding with the epSOS reference Value Sets. [Mandatory] OriginalPDF Printable representation (PDF/A) of the original national data as we expect have been seen by the originator HCP [Mandatory].
Output parameters	**EpSOSPDF** structure response structure including the epSOS unstructured CDA embedding the original pdf and the response status structure. The response status structure provides information about the operation results, including possible errors and warning.

Component behaviour	After having received the *ToEpSOSPDF()* request, this component takes the EpSOSOriginalData and the OriginalPDF and generates an unstructured CDA embedding the PDF using the information already present in the EpSOS original data. As a final result the PDF embedded CDA is returned to the requesting component. The embedded PDF CDA shall to have a link to the EpSOSOriginalData. *Exceptions*: in case of processing error or warning, the responseStatusStructure will be used to convey this information to the calling component with an appropriated error and warning code. A detailed list of the managed exceptions will be provided in the Detail Design Specification document. Each exception condition occurred will be logged (standard and audit), reporting both the exception code and its English description.
Notes	

`Translate (EpsosCDA; TargetLanguageCode) : TranslatedEpSOSCDA`

Textual description of operation	
Translation from epSOS pivot data to consumer country language.	
Input parameters	**EpSOSosCDA** Document in epSOS pivot format (with epSOS codes) **TargetLanguageCode**. Identifier (code) of the target language.
Output parameters	**TranslatedEpSOSCDA** epSOS pivot CDA with translated epSOS codes into the consumer country language.

Component behaviour	After having received a *translate*() request, this component starts to process the received *EpSOSosCDA* in order extract the epSOS coded concepts. Subsequently, for each coded concept found, it makes use of the TSAM capabilities to obtain the representation of that concept in the target TargetLanguageCode identifier. This information is therefore used by this component to update the displayName attribute of that coded entry. After the completion of this translation phase, an epSOS pivot document with "translated" concepts is obtained. This document is therefore returned to the requesting party. No changes are applied to the document identifiers. *Exceptions*: in case of processing error or warning, the responseStatusStructure will be used to convey this information to the calling component with an appropriated error and warning code. A detailed list of the managed exceptions will be provided in the Detail Design Specification document. Each exception condition occurred will be logged (standard and audit), reporting both the exception code and its English description.
Notes	

Linking documents

The relationship between the epSOS pivot document and the embedded PDF CDA is always inferred via the XFRM relationship with their parent document EpSOSOriginalData. Optionally, a direct relationship between the epSOS pivot document and the embedded PDF CDA could be created on request as RPLC relationship.

Component behaviour (called operations)

`getEpSOSConceptByCode()`

This component issues a getEpSOSConceptByCode_() request in order to know the best matching epSOS Concept, according to the information provided.

Exceptions: in case of returned errors this component shall appropriately use NullFlavors for valorizing the “main” coded concept; nevertheless the original code shall be provided through the <translation> element.

`getDesignationByEpSOSConcept()`

This component issues getDesignationByEpSOSConcept_() request in order to know the target language epSOS Designation, according to the information provided.

Exceptions: in case of returned errors no actions are required for displayName translation.

This component provides capabilities for coded concepts translation and transcoding if necessary.

The detailed description of the component can be found in Appendix A.

3.6 **TerminologyServicesAccessManager**

This component is called by *TransformationManager*. The component is responsible for translating a given concept designation into the requested target language as well as transcoding a given “local” coded concept into the appropriate epSOS coded concept using the information present in the Terminology Repository. The Terminology Repository is a database that is a part of the NCP and represents the epSOS Reference Terminology. The content of the Terminology Repository is specific for each MS. It is the responsibility of the MS to maintain and update the content of the Terminology Repository.

This component provides capabilities for coded concepts translation and transcoding if necessary.

The first term (translation) stands for the capability of associating to an epSOS coded concept the localized concept description or display name: i.e. the translation into the target language of the “concept” conveyed (e.g. code “30001000” EDQM can have the display names “Φύσιγγα”, “Ampulka” or “Ampoule”, depending on where it is used.)

With the second term (transcoding) we mean the capability of getting the epSOS quasi-synonymous[2] associated to a “local” coded concept.

In both cases, the content needed by the Terminology Services Access Manager is the Terminology Repository. The Terminology Repository is a database representation of the epSOS Reference Terminology.

It must be noted that the epSOS Reference Terminology has as a starting point the epSOS MVC, which in turn is the basis for the epSOS MTC (see Appendix B2 for

[2] i.e. a coded concept derived from the appropriate epSOS Value Set semantically equivalent to a given coded concept.

further details). The mapping activity from the “local” coded concept to the epSOS Value Sets present in the epSOS MVC is out of scope of epSOS and it is the responsibility of the National Linguistic Competence Centers from each Member State . This mapping is compiled in the epSOS Master Value Sets Translation/Transcoding Catalogue (epSOS MTC). The maintenance of the MTC and its relationship with the MVC are out of scope of the common components design.

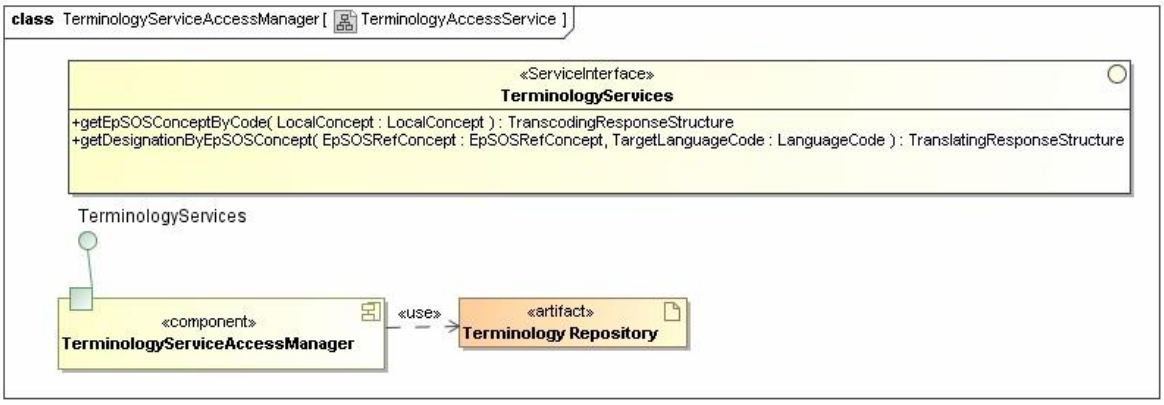


Figure 10 – Component : TerminologyServiceAccessManager

Component responsibilities

1. Translating a given concept designation into the requested target language using the information present in the Terminology Repository.
2. Transcoding a given “local” coded concept into the appropriate epSOS coded concept using the information present in the Terminology Repository.

`getEpSOSConceptByCode(LocalConcept) : TranscodingResponseStructure`

Textual description of operation
Transcoding a given “local” coded concept into the appropriate epSOS coded concept using the information present in the Terminology Repository.

Input parameters	LocalConcept; structure used to convey the concept derived from the epSOS Original Data. It shall include at least the concept code and the concept code system. Code System Version, Country Code and value set OID - if available - should be provided.
Output parameters	TranscodingResponseStructure Response structure including: 1. the epSOS Reference Concept: this means the Concept Code, the English designation, the concept code system (OID), Code System Version, Value Set OID; Value Set Version, 2. The responseStatusStructure, providing information about operation result, including possible errors and warning.
Component behaviour	When this component receives a getEpSOSConceptByCode_() request, it uses all the data extracted from the LocalConcept structure in order to search within the Terminology Repository for the best matching epSOS Concept, according to the local information provided (e.g., if no code system version is indicated, the latest version will be provided). All information retrieved is finally returned to the requesting component. *Exceptions*: if there is no transcoding or a processing error occurs, the responseStatusStructure will be used to convey this information to the calling component with an appropriated error and warning code. A detailed list of the managed exceptions will be provided in the Detail Design Specification document. Each exception condition occurred will be logged (standard and audit), reporting both the exception code and its English description.
Notes	Makes use of Terminology Repository

getDesignationByEpSOSConcept(EpSOSRefConcept; TargetLanguageCode) : TranslatingResponseStructure

Textual description of operation	
Translating a given concept designation into the requested target language using the information present in the Terminology Repository	
input parameters	EpSOSRefConcept. Structure used to convey the concept derived from the epSOS pivot CDA. It shall include at least the concept code and the concept code system. Code System Version, Country Code and value set OID - if available - should be provided. TargetLanguageCode identifier (code) of the target language.
Output parameters	translatingResponseStructure Response structure including: the target language concept designation; the responseStatusStructure providing information about operation result, including possible errors and warning.

Component behaviour	When this component receives a getDesignationByEpSOSConcept_() request, it uses all the data extracted from the EpSOSRefConcept structure in order to search within the Terminology Repository for the target language epSOS Designation, according to the local information provided (e.g., if no code system version is indicated, the latest version will be provided). All information retrieved are finally returned to the requesting component. *Exceptions*: if there is no translation or a processing error occurs, the responseStatusStructure will be used to convey this information to the calling component with an appropriated error and warning code. A detailed list of the managed exceptions will be provided in the Detail Design Specification document. Each exception condition occurred will be logged (standard and audit), reporting both the exception code and its English description.
Notes	Makes use of Terminology Repository

The detailed description of the component can be found in Appendix A.

3.7 **AuditTrailWriter**

As required by WP 3.3 WP 3.4, WP 3.6 and WP 3.7 every transaction in epSOS must be audited with very limited information. National requirements for an extended auditing must be realized in the *NationalConnector* or the national infrastructure. The *AuditTrailWriter* component is responsible for formatting an EventLog message in an Audit Trail and Node Authentication (ATNA) - compatible way and passing it securely to the audit repository.

In future versions it may be an option to start the auditing in an asynchronous manner. For the current version this option will be omitted due to simplicity.

Note, that it should be differentiated between logging and audit. Audit captures business-level events (e.g., request for PS) while logging occurs on a lower level and is responsible for capturing application-level events.

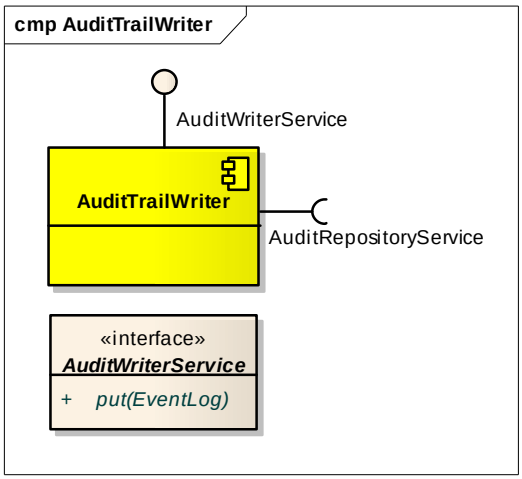


Figure 11 AuditTrailWriter component

`put(EventLog)`

Textual description of operation	
This operation stores an audit event in the audit trail.	
Input parameters	EventLog Audit entry to record in a format which is yet to define
Output parameters	-
Notes	No identifiable medical information should be contained in EventLog

The detailed description of the component can be found in Appendix A.

3.8 AuditRepository

This component is responsible for storing audit trail that is captured by *AuditTrailWriter* component. Additionally this component exposes an interface that is used by the national infrastructure to selectively querying fragments of the audit trail. The content of the audit repository will be analyzed in the national infrastructure. Therefore, presented here the *AuditService* interface has a very general structure that can be further adjusted by member states according to their specific needs.

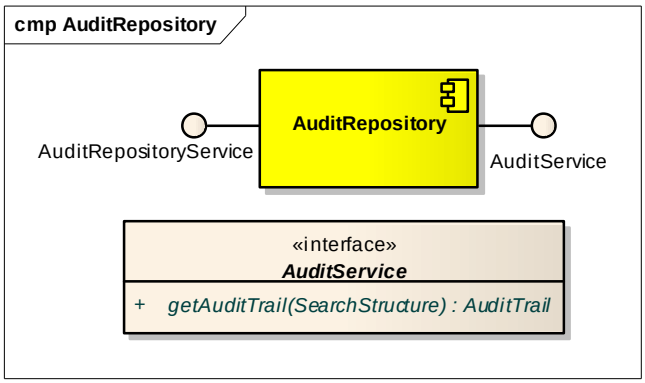


Figure 12 AuditRepository component

`getAuditTrail(SearchStructure): AuditTrail`

Textual description of operation	
The operation allows to selectively extract records from the audit trail	
Input parameters	SearchStructure The structure contains search parameters that define which records of the audit trail should be extracted
Output parameters	AuditTrail Records of the audit trail that correspond to the parameters defined in SearchStructure
Notes	No identifiable medical information should be contained in the audit repository

The detailed description of the component can be found in Appendix A.

3.9 **RoutingManager**

Before any message can be sent by the NCP (in the role of country B), information for the correct routing to the corresponding NCP (in the role of country A) must be resolved. The outcome of the *RoutingManager* is a URL of the corresponding NCP.

The address lookup table is a XML-document that can be stored in the NCP's local file system or be fetched (and cached) from a URL of a central service.

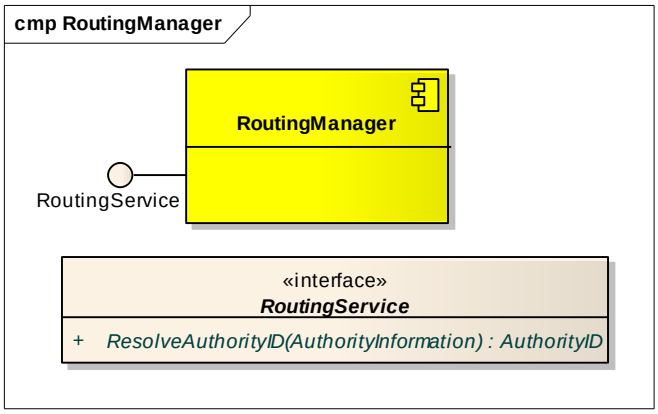


Figure 13 RoutingManager component

`ResolveAuthorityID(AuthorityInformation) : AuthorityID`

Textual description of operation	
The result of this operation is the OID of the inbound gateway that has to be requested. It is not specified yet in which ways this OID can be resolved. Therefore the input parameters are not stable.	
Input parameters	AuthorityInformation Not specified in which way authority information is coded.
Output parameters	AuthorityID OID which clearly references the inbound gateway that will be requested.

In the implementation, however, this component will be developed as a part of the ConfigurationManager.

3.10 ConfigurationAndMonitoringManager

The *ConfigurationAndMonitoringManager* provides a monitoring console that is used for managing configurations of the nation-specific components as well as monitoring their status.

As one of its subcomponents, ConfigurationAndMonitoringManager should include an Abuse Detection System. This subcomponent should be responsible for analyzing the audit trail and, based on a configurable rule set, prevent possible abuses (such as excessive requests issued from a HCP or a patient is queried from more than one country at a time).

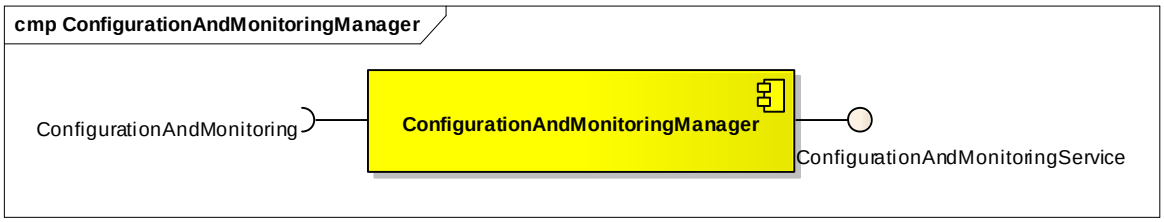


Figure 14 ConfigurationAndMonitoringManager component

The detailed description of the component can be found in Appendix A.

3.11 NationalConnector

The *NationalConnector* is not a common component, but a collection of adapters to the national protocols and data formats. The *NationalConnector* makes use of a common exposed API as required by the *WorkflowManager* as well as exposes a common API to the *WorkflowManager* itself. In order to ensure that the commonly developed components can interoperate with the NCP-in-the-box, interfaces the NationalConnectors exposes to the WorkflowManager should be kept consistent across all individual implementations. Interfaces of the NationalConnector exposed to the national infrastructure are country-specific with no restrictions imposed on it and therefore are not described in the document.

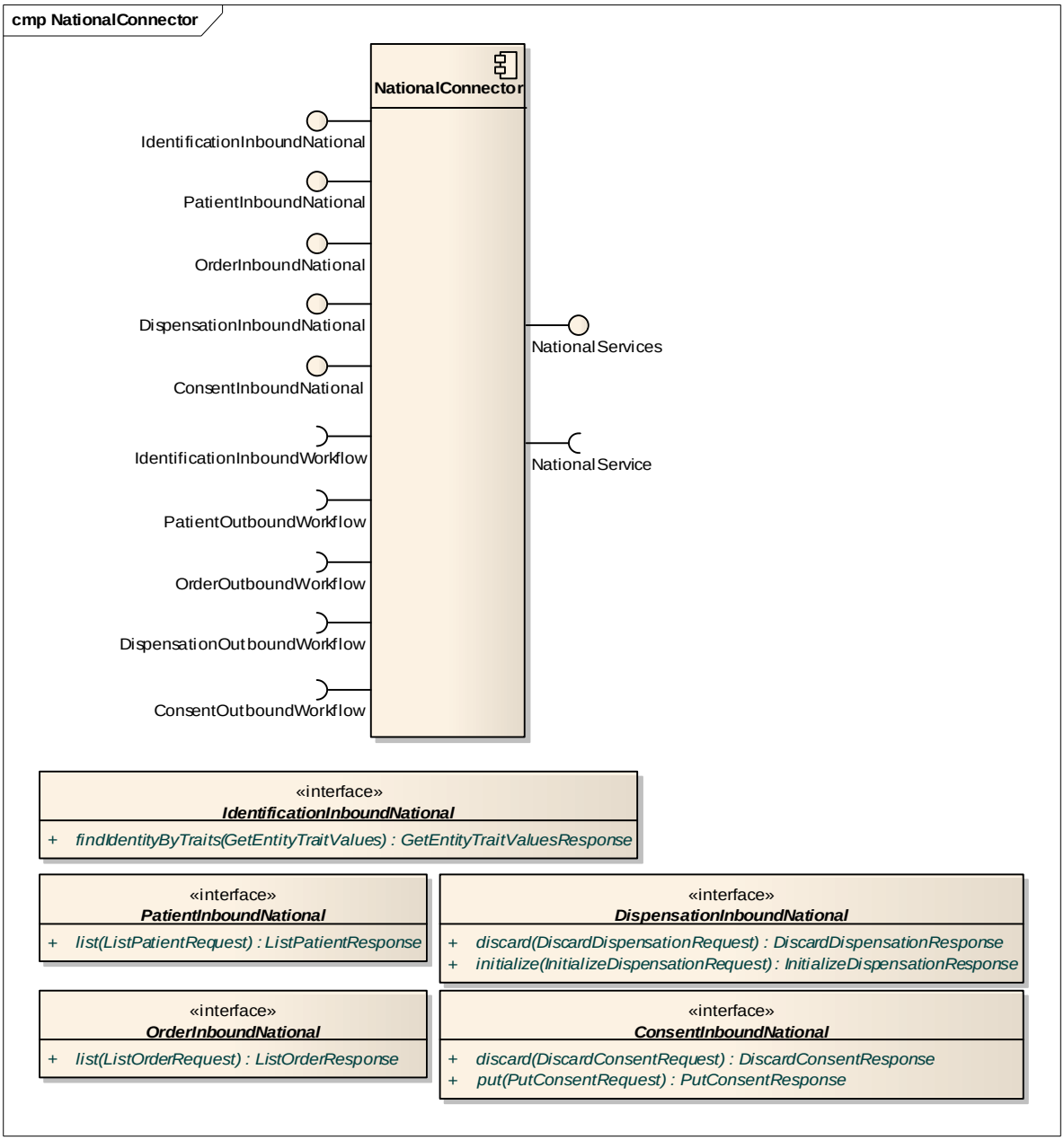


Figure 15 NationalConnector component

The NationalConnector is a component that is developed by MSs and is responsible for connecting the NCP to the national infrastructure of the MS.

The detailed description of the interfaces that are offered and consumed by the component can be found in Appendix A.

3.12 Automatic Data Collection

Automatic data collection is a feature requested to the NCP to provide information to evaluate the epSOS interoperability system performance and to collect statistics on the population using epSOS services.

WP 1.2, Project Evaluation, has provided a list of data and parameters to be anonymously collected.

The component in charge of the automatic data collection is the Audit Manger.

The ConfigurationAndMonitoringManager will be used to define the rules for collecting Automatic Data, and to extract the requested anonymous reports.

See Appendix A for details on the implementation of the Automatic Data Collection.

4 NCP Components Interactions

This section describes the end2end interactions between the components of the NCP that correspond to invocations of the epSOS front-end services and that are defined in D3.3.2:

- epSOS Patient Identification Service
- epSOS Patient Service
- epSOS Order Service
- epSOS Dispensation Service
- epSOS Consent Service

The sequence diagrams provided below shows in which order and by which method calls the NCP components are invoked when one of the front-end services is called.

4.1 epSOS Patient Identification Service

There are different alternatives to discover a Patient ID and therefore different attributes to be used by calling NCP-B. But all of them result in providing NCP-A with a PID (Patient Identity).

The transaction is visualized by sequence diagrams displayed at Figure 16 and Figure 17:

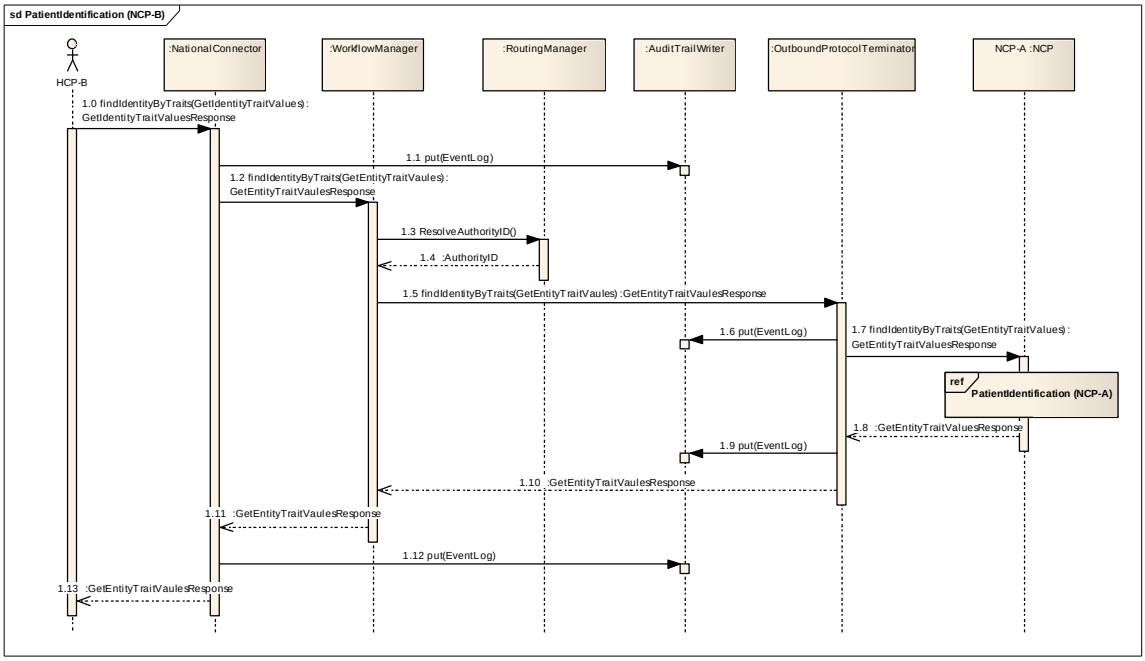


Figure 16 Patient Identification - NCP-B

- [1.0] The HCP in MS B issues a request to the NCP-B by calling an appropriate method of the *NationalConnector* and providing corresponding identity traits.
- [1.1]-[1.2] The *NationalConnector* calls the *AuditTrailWriter* to log the patient identification request in the audit trail and forwards the request to the *WorkflowManager*.
- [1.3]-[1.4] The *WorkflowManager* contacts the *ServiceRoutingManager* in order to obtain the URL of the corresponding NCP-A. The URL of the NCP-A is returned.
- [1.5] The request is forwarded to the *OutboundProtocolTerminator*.
- [1.6]-[1.7] The *OutboundProtocolTerminator* logs the patient identification request in the audit trail, wraps the request in a SOAP envelope and issues the SOAP request by performing a remote call of the NCP-A's *IdentityService*::*findIdentityByTraits()* operation.
- [1.8]-[1.13] As the response it expects a SOAP message containing G*etEntityTraitValuesResponse* with a *PatientID* or status information in case the identification was not successful. Upon a successful identification, the *PatientID* is returned to the HCP with corresponding records made in the audit trail.

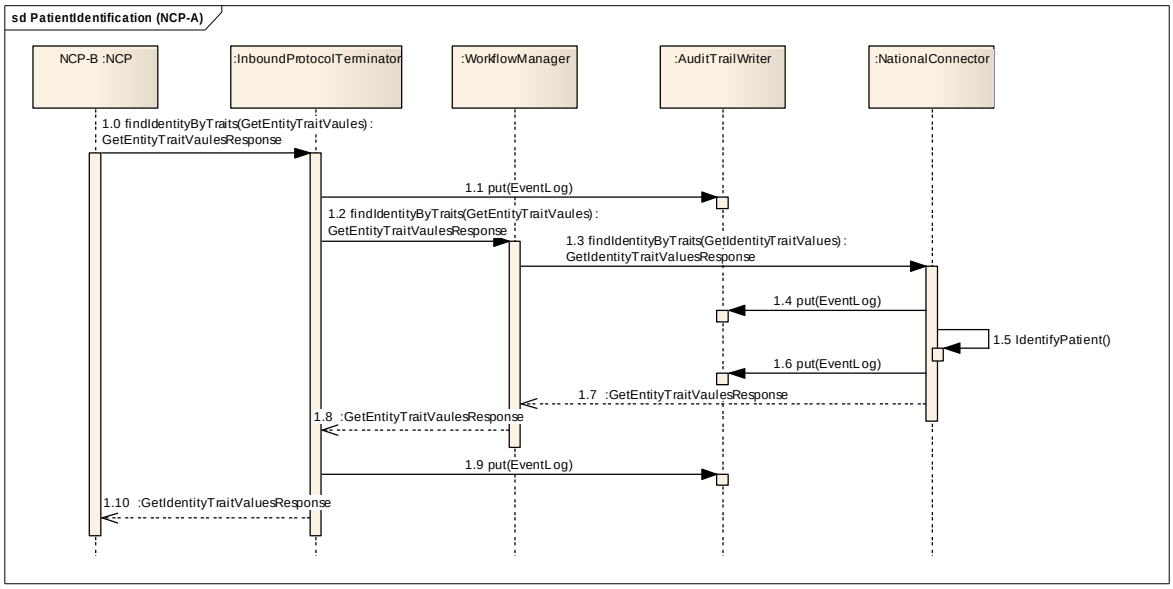


Figure 17 Patient Identification - NCP-A

- [1.0]-[1.1] Upon the arrival of an *IdentityService*::*findIdentityByTraits()* SOAP request, the *InboundProtocolTerminator* calls the *AuditTrailWriter* to log the patient identification request in the audit trail.
- [1.2] The *InboundProtocolTerminator* forwards the trait values to the *WorkflowManager*.
- [1.3] The *WorkflowManager* forwards the identity traits to the *NationalController*.
- [1.4]-[1.5] The *NationalConnector* makes a corresponding record in the audit trail and issues a request to the national infrastructure. The response should contain either the ID of the patient (*PatientID*) or status information in case the identification was not successful.
- [1.6]-[1.10] If the identification was successful, the *PatientID* is returned to the *InboundProtocolTerminator* with corresponding records being made in the audit trail. The *InboundProtocolTerminator* wraps the *PID* into a SOAP envelope and sends it to NCP-B as a SOAP response.

4.2 epSOS Patient Service

The transaction is visualized by sequence diagrams displayed at Figure 18 and Figure 19:

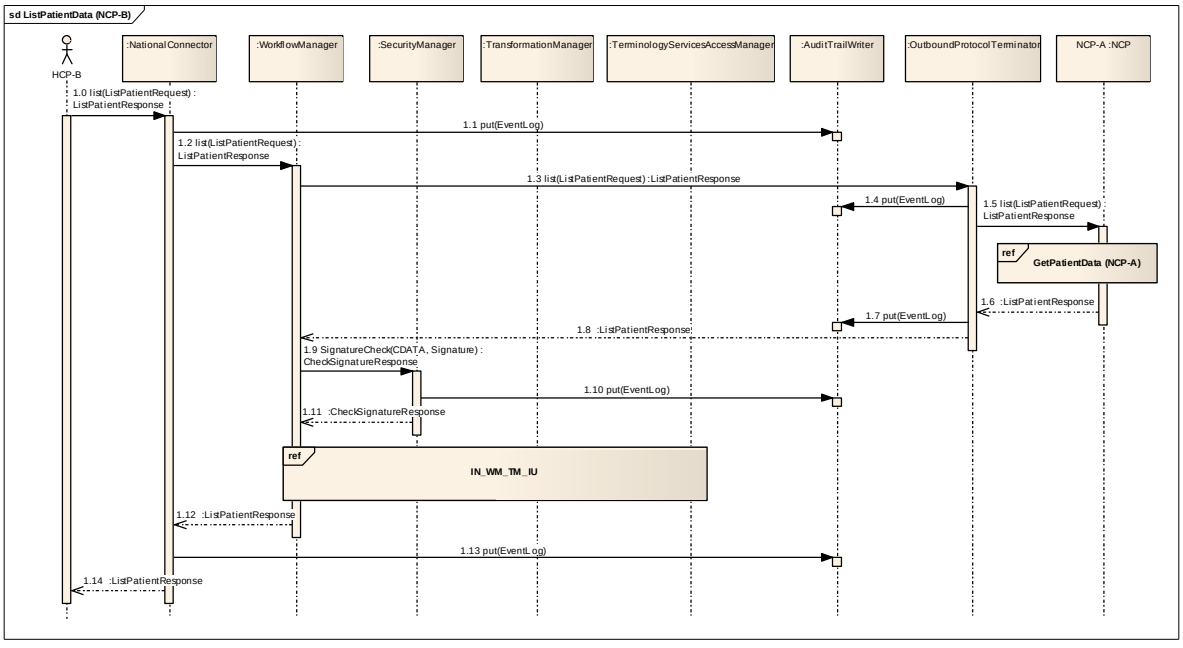


Figure 18 List patient data NCP-B

- [1.0] The HCP in MS B issues a request for retrieving a PS to the NCP-B by calling a corresponding method of the *NationalConnector* and providing ID of the corresponding patient.
- [1.1]-[1.2] The *NationalConnector* calls the *AuditTrailWriter* to log the patient data request in the audit trail and forwards the request to *WorkflowManager*.
- [1.3] The WorkflowManager forwards the request to the OutboundProtocolTerminator.
- [1.4]-[1.5] The *OutboundProtocolTerminator* wraps the request in a SOAP envelope, logs the request for patient data in the audit trail and performs a remote call of *NCP-A's PatientService::list()* operation by issuing a SOAP request. As the response it expects a SOAP message that contains the patient summary in the epSOS pivot format.
- [1.6]-[1.8] After the PS is found, the *NationalConnector* makes the corresponding record in the audit trail and returns the result to the *WorkflowManager*.
- [1.9]-[1.11] The *WorkflowManager* calls *SecurityService::SignatureCheck ()* to verify the signature of the PS. The corresponding record is made in the audit trail.
- [IN_WM_TM_IU] The PS is forwarded to the *TransformationManager* where it is transformed and translated into MS B format.
- [1.12]-[1.14] The translated and signed patient summary is returned to HCP with corresponding records being made in the audit trail.

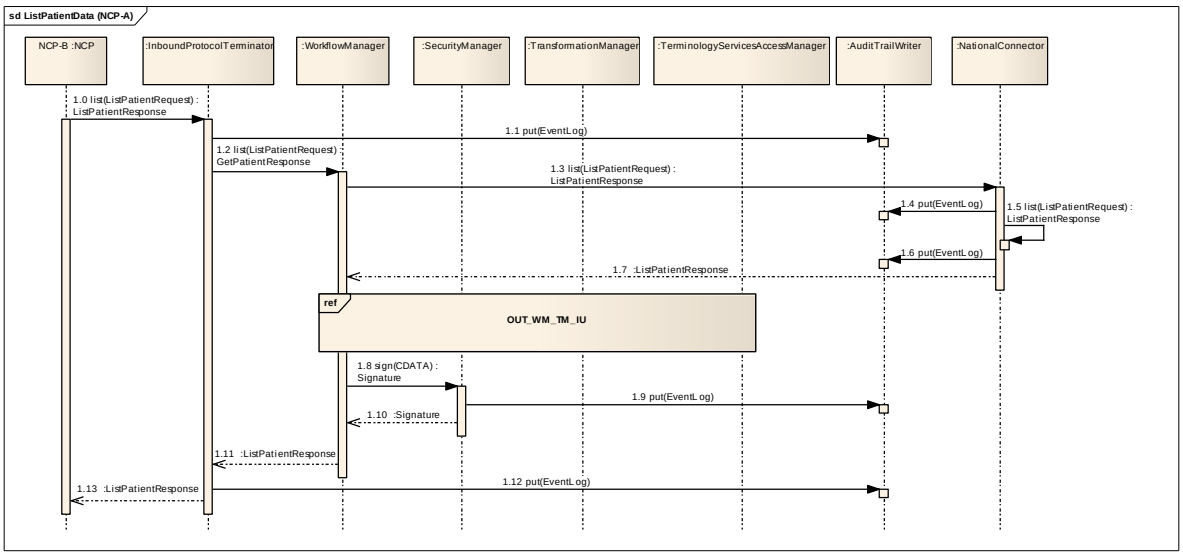


Figure 19 List patient data NCP-A

- [1.0]-[1.2] Upon the arrival of a *PatientService::list()* SOAP request, the *InboundProtocolTerminator* calls the *AuditTrailWriter* to log the patient data request in the audit trail, and forwards the resulting reqest to the *WorkflowManager*.
- [1.3] The WorkflowManager calls OrderInboundNational::list() method of the NationalConnector.
- [1.4]-[1.5] The *NationalConnector* makes an appropriate record in the audit trail and issues a corresponding request to the national infrastructure. The response should contain the requested PS in the national format of MS A.
- [1.6]-[1.7] After the requested PS is found, the *NationalConnector* makes an appropriate record in the audit trail and returns the result to the *WorkflowManager*.
- [OUT_WM_TM_IU] The successfully verified PS (in MS A format) is forwarded to the *TransformationManager* where it is transformed into epSOS pivot format.
- [1.9]-[1.10] The *WorkflowManager* calls *SecurityService::Sign()* to sign the transformed PS with the NCP-A signature. The corresponding record is made in the audit trail.
- [1.11]-[1.13] The signed PS in epSOS pivot format is returned to the *InboundProtocolTerminator* with corresponding records being made in the audit trail. *InboundProtocolTerminator* wraps the PS into a SOAP envelope and sends it to NCP-B as a SOAP response.

Nested diagrams In_WM_TM_IU and Out_WM_TM_IU correspond to the “interaction uses” (IU) in which the Semantic Components are involved.

The Document Consumer Workflow Manager Transformation Manager Interaction Use (In_WM_TM_IU) is described on Figure 20. This IU is performed at a certain time of the workflow by the Document Consumer NCP (usually NCP-B) to create a translated version of the epSOS pivot CDA into the target language.

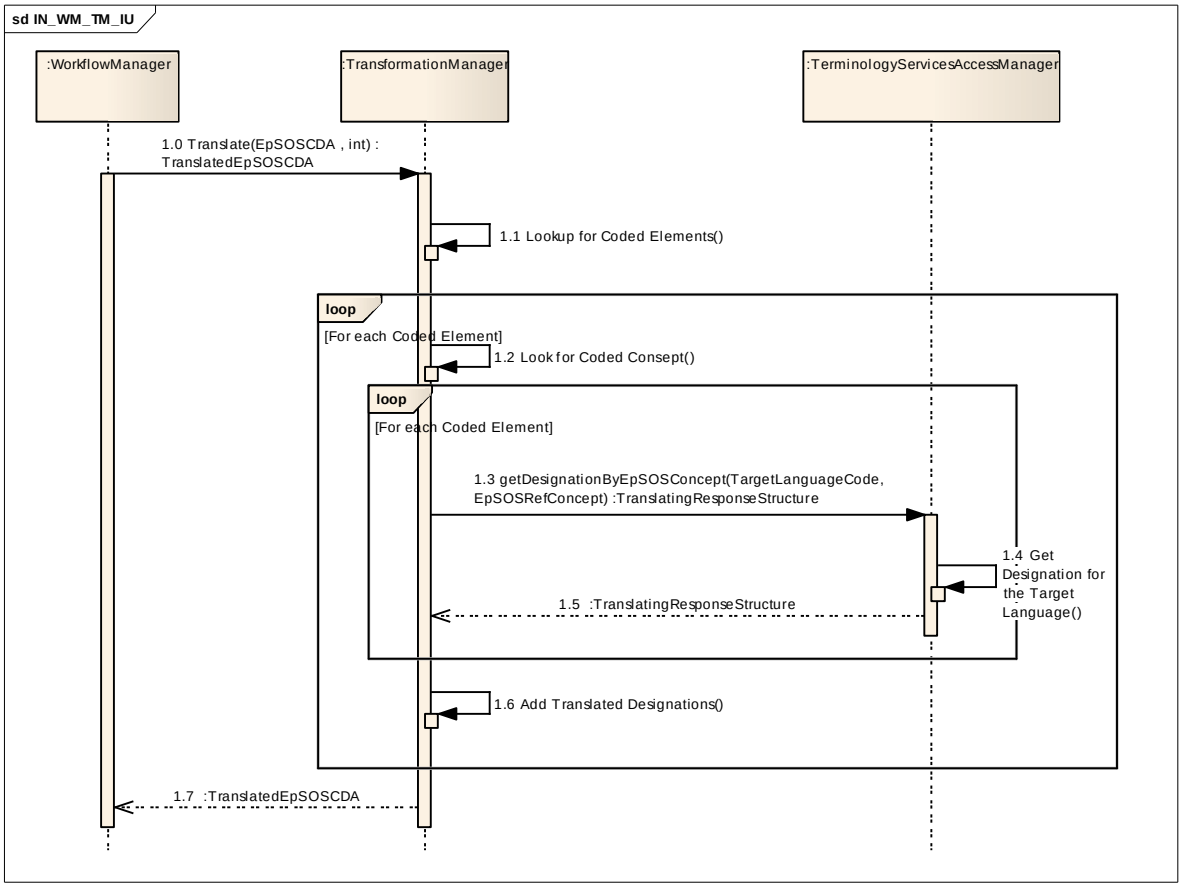


Figure 20 Document Consumer WorkflowManager-TransformationManager Interaction Use (IN_WM_TM_IU)

The Document Provider Workflow Manager Transformation Manager Interaction Use (Out_WM_TM_IU) is depicted on Figure 21. This IU is performed at a certain time of the workflow by the Document Provider NCP (usually NCP-A) to create the epSOS pivot CDA and the CDA with the original PDF embedded.

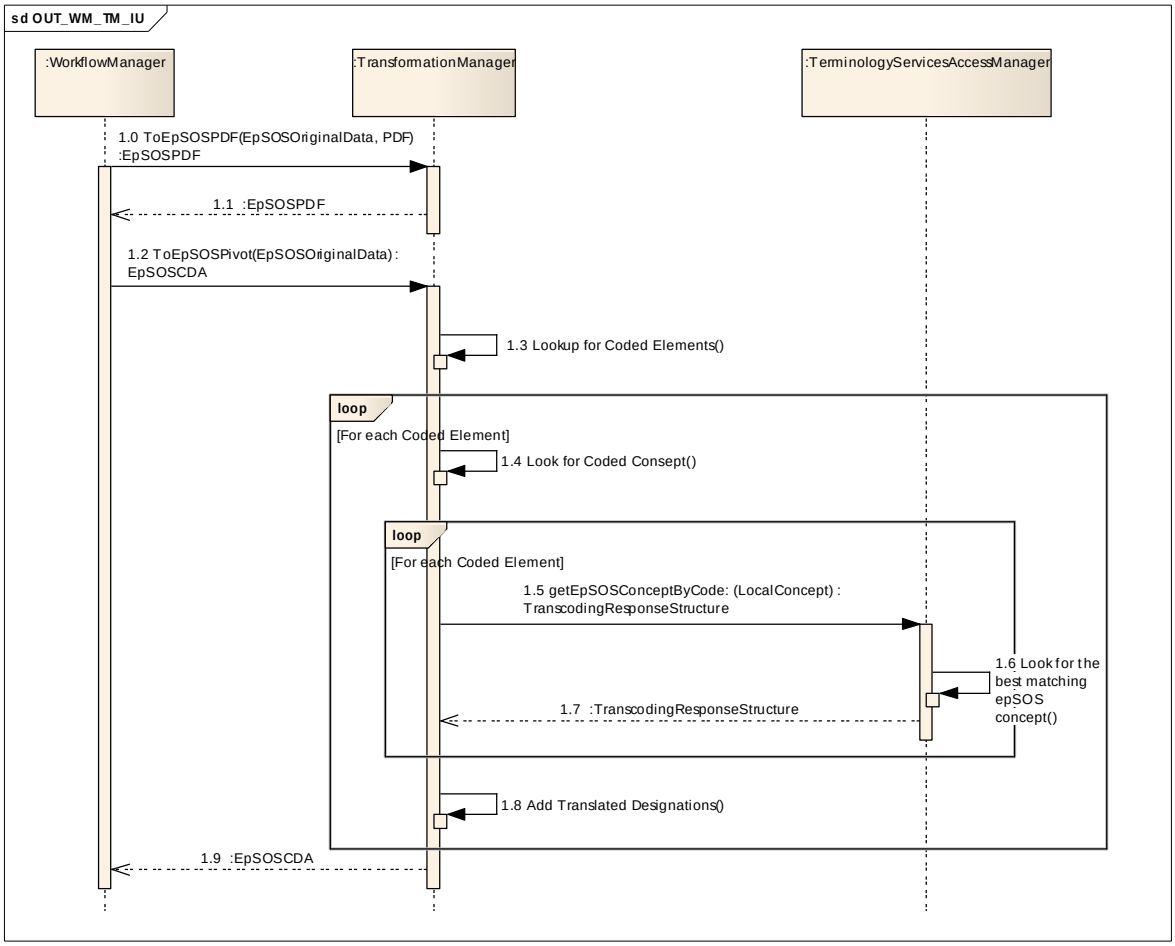


Figure 21 Document Provider WorkflowManager-TransformationManager Interaction Use (OUT_WM_TM_IU)

4.3 **EpSOS Order Service**

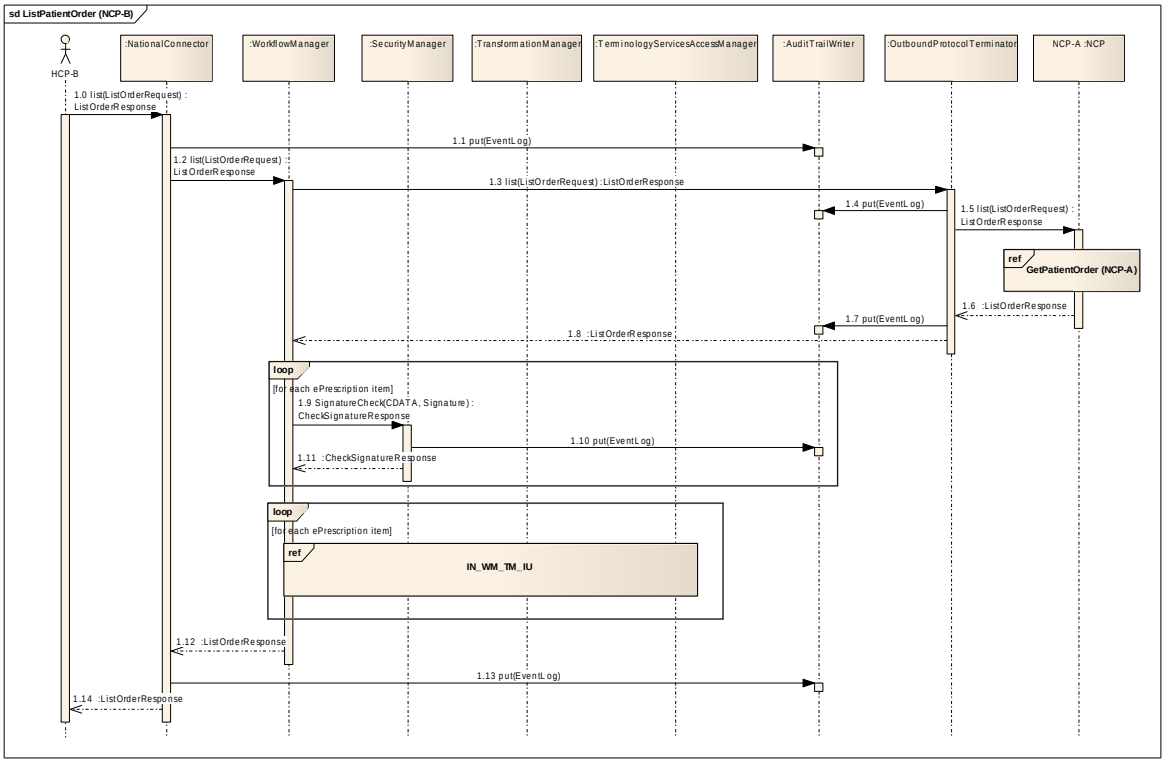


Figure 22 List patient prescriptions NCP-B

- [1.0] The HCP in MS B issues a request for patient prescriptions to the NCP-B by calling a corresponding method of the *NationalConnector* and providing ID of the patient.
- [1.1]-[1.2] The *NationalConnector* calls the *AuditTrailWriter* to log the request for patient prescriptions in the audit trail and forwards the request to the *WorkflowManager*.
- [1.3] The WorkflowManager forwards the request to the OutboundProtocolTerminator.
- [1.4]-[1.5] The *OutboundProtocolTerminator* wraps the request in a SOAP envelope, logs the request for patient prescriptions in the audit trail and performs a remote call of *NCP-A's OrderService::list()* operation by issuing a SOAP request. As the response it expects a SOAP message that includes the prescriptions in the epSOS pivot format.
- [1.6]-[1.8] Upon the arrival of the SOAP response, the *OutboundProtocolTerminator* makes a corresponding record in the audit trail, extracts *ListOrderResponse* object from the SOAP response and forwards it to the *WorkflowManager*.

- [1.9]-[1.11] By consequently calling *SecurityService::SignatureCheck(),* the signatures of the arrived prescriptions are verified. The corresponding records are made in the audit trail.
- [IN_WM_TM_IU] (see below) The received prescriptions in epSOS pivot format are consequently forwarded to the *TransformationManager* where they are transformed and translated into MS B format.
- [1.12]-[1.14] The translated and signed prescriptions are returned to HCP with corresponding records being made in the audit trail.

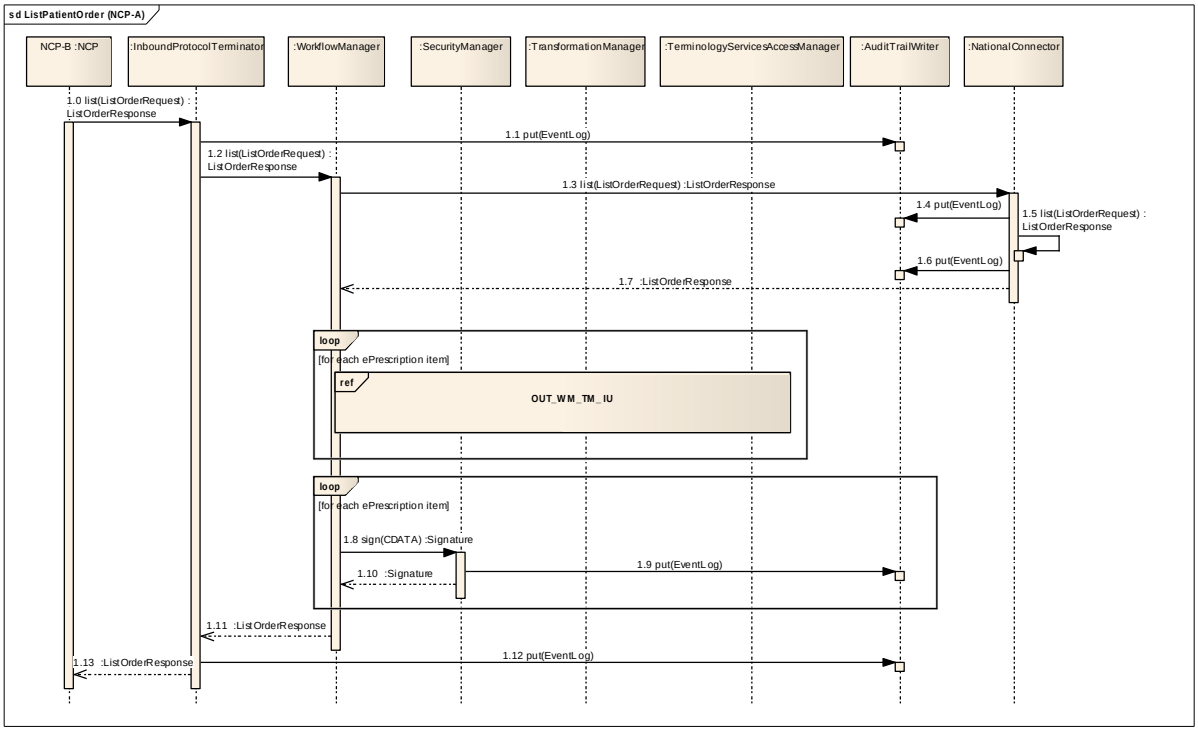


Figure 23 List patient prescriptions NCP-A

- [1.0]-[1.2] Upon the arrival of a *OrderService::list()* SOAP request, the *InboundProtocolTerminator* calls the *AuditTrailWriter* to log the request for the patient prescriptions in the audit trail, and forwards the request to *WorkflowManager*.
- [1.3] The WorkflowManager calls PatientInboundNational::list() method of the NationalConnector.
- [1.4]-[1.5] The *NationalConnector* makes an appropriate record in the audit trail and forwards the request to the national infrastructure. The response should contain requested prescriptions in the national format of MS A.

- [1.6]-[1.7] After the corresponding Prescriptions are found, the *NationalConnector* makes an appropriate record in the audit trail and returns the result to the *WorkflowManager*.
- [OUT_WM_TM_IU] The successfully verified prescriptions (in MS A format) are consequently forwarded to *TransformationManager* where they are transformed into epSOS pivot format.
- [1.8]-[1.10] By consequently calling *SecurityService::Sign()*, the transformed prescriptions are signed with the NCP-A signature. The corresponding records are made in the audit trail.
- [1.11]-[1.13] The signed prescriptions in epSOS pivot are returned to the *InboundProtocolTerminator* with corresponding records being made in the audit trail. The *InboundProtocolTerminator* wraps the prescriptions into a SOAP envelope and sends it to NCP-B as a SOAP response.

4.4 EpSOS Dispensation Notification Service

EpSOS should enable an HCP to dispense a medication or revoke an already provided dispensation. Here sequence diagrams reflecting corresponding *initialize()* and *discard()* calls of the NCP's Dispensation Service are provided.

4.4.1 Initializing a Dispensation

The transaction is visualized by sequence diagrams displayed at Figure 24 and Figure 25:

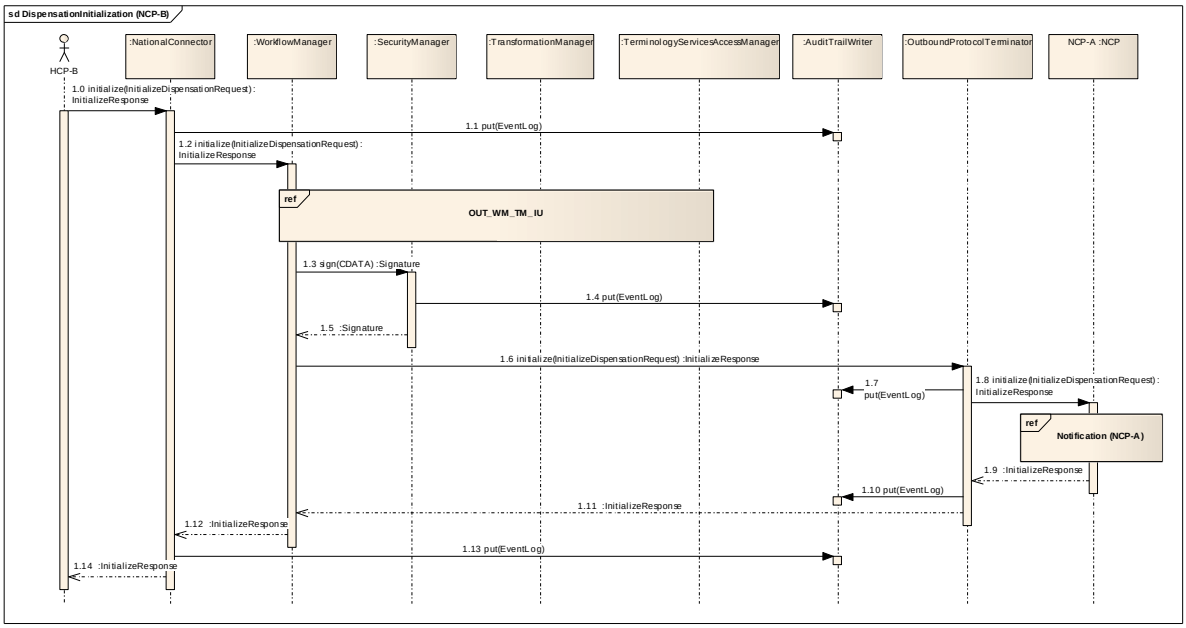


Figure 24 Dispensation initialization NCP-B

- [1.0] The HCP in MS B issues a dispensation notification by calling a corresponding method of the *NationalConnector*.
- [1.1]-[1.2] The *NationalConnector* calls the *AuditTrailWriter* to log the dispensation request in the audit trail and forwards the request to the *WorkflowManager*.
- [OUT_WM_TM_IU] The successfully verified dispensation (in MS B format) is forwarded to *TransformationManager* where it is transformed into epSOS pivot format.
- [1.3]-[1.5] By calling *SecurityService::Sign()*, the transformed dispensation is signed with the NCP-B signature. The corresponding record is made in the audit trail.
- [1.6] The *WorkflowManager* forwards the transformed in epSOS pivot format and signed dispensation to the *OutboundProtocolTerminator*.
- [1.7]-[1.8] The *OutboundProtocolTerminator* wraps the dispensation in a SOAP envelope, logs the request in the audit trail and performs a remote call of NCP-A's *DispensationService::initialize()* operation by issuing a SOAP request. As a response it expects a SOAP message containing the *InitializeDispensationResponse* that contains the identifier of a defined notification acknowledgement.
- [1.9]-[1.14] The *InitializeResponse* is returned to HCP with corresponding records being made in the audit trail.

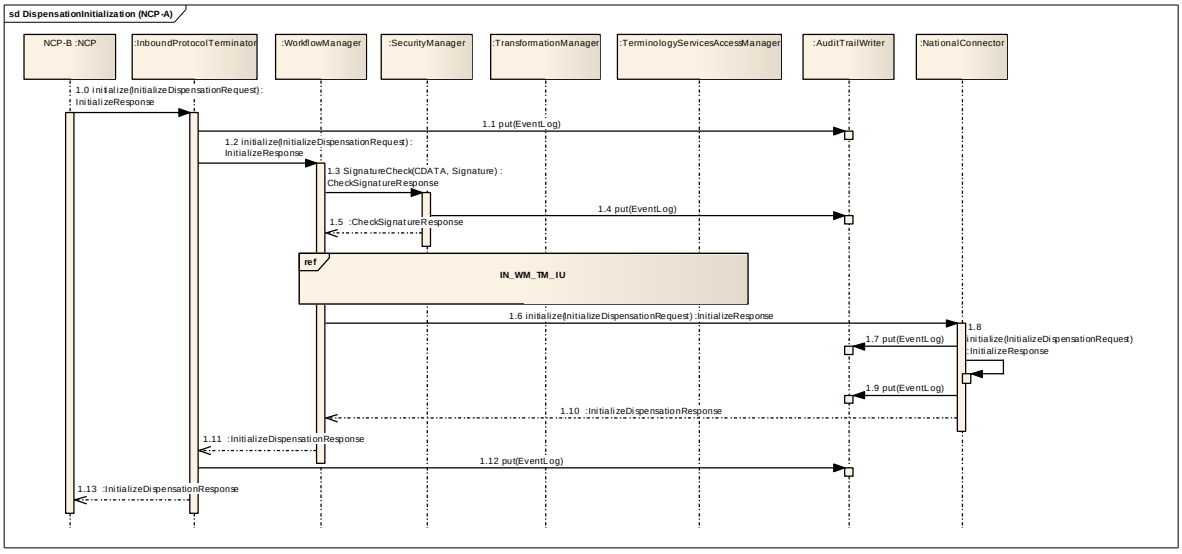


Figure 25 Dispensation initialization NCP-A

- [1.0]-[1.2] Upon the arrival of a *PatientService::initialize()* SOAP request, *InboundProtocolTerminator* calls the *AuditTrailWriter* to log the dispensation

request in the audit trail, and forwards the resulting object to the *WorkflowManager*.

- [1.3]-[1.5] By calling *SecurityService::SignatureCheck(),* the signature of the arrived dispensation is checked. The corresponding record is made in the audit trail.
- [IN_WM_TM_IU] The successfully verified dispensation in epSOS pivot format is forwarded to the *TransformationManager* where it is transformed and translated into MS A format.
- [1.6] The WorkflowManager calls DispensationInboundNational::initialize() method of the NationalConnector.
- [1.7]-[1.8] The *NationalConnector* makes an appropriate record in the audit trail and issues a corresponding request to the national infrastructure. The response *InitializeResponse* should contain the identifier of a defined notification acknowledgement.
- [1.9]-[1.13] *InitializeDispensationResponse* is returned to the *InboundProtocolTerminator* with corresponding records being made in the audit trail. The *InboundProtocolTerminator* wraps the response into a SOAP envelope and sends it to NCP-B as a SOAP response.

4.4.2 Discarding a Dispensation

The sequences for discarding a dispensation are identical to the sequences presented on Figure 24 and Figure 25 with *initialize(InitializeDispensationRequest):InitializationResponse* calls substituted by *discard(DiscardDispensationRequest):DiscardResponse* calls.

4.5 EpSOS Consent Notification Service

EpSOS should enable a patient to provide or revoke consent for participation in EpSOS use cases while being outside of her country of residence. Here sequence diagrams reflecting corresponding *initialize()* and *discard()* calls of the NCP's Consent Service are provided.

4.5.1 Providing a Consent

The transaction is visualized by sequence diagrams displayed at Figure 26 and Figure 27:

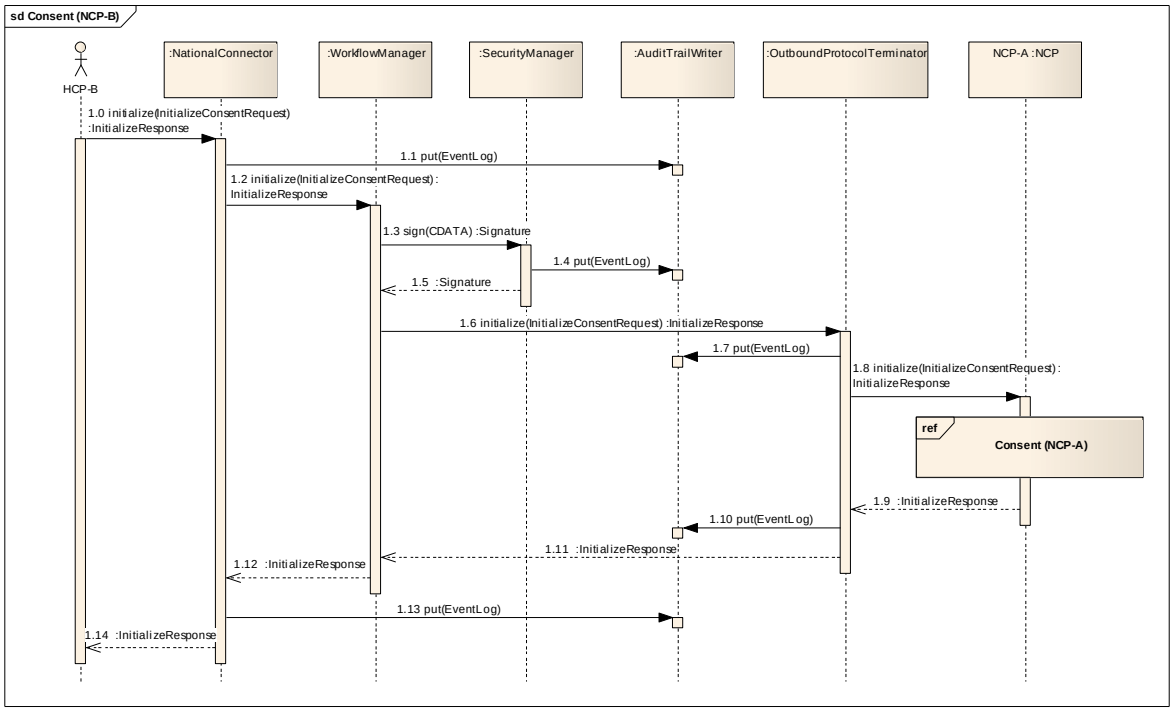


Figure 26 Consent initialization NCP-B

- [1.0] The patient uses the HCP in MS B as an intermediary for providing a consent by calling a corresponding method of the *NationalConnector*.
- [1.1]-[1.2] The *NationalConnector* calls the *AuditTrailWriter* to log the consent initialization in the audit trail and forwards the request to the *WorkflowManager*.
- [1.3]-[1.5] By calling *SecurityService::Sign()*, the consent is signed with the NCP-B signature. The corresponding record is made in the audit trail.
- [1.6] The *WorkflowManager* forwards the signed consent to the *OutboundProtocolTerminator*.
- [1.7]-[1.8] The *OutboundProtocolTerminator* wraps the consent in a SOAP envelope, logs the request in the audit trail and performs a remote call of NCP-A's *ConsentService::initialize()* operation by issuing a SOAP request. As a response it expects a SOAP message containing the *InitializeResponse* that contains the identifier of a defined notification acknowledgement.
- [1.9]-[1.14] The *InitializeResponse* is returned to HCP with corresponding records being made in the audit trail.

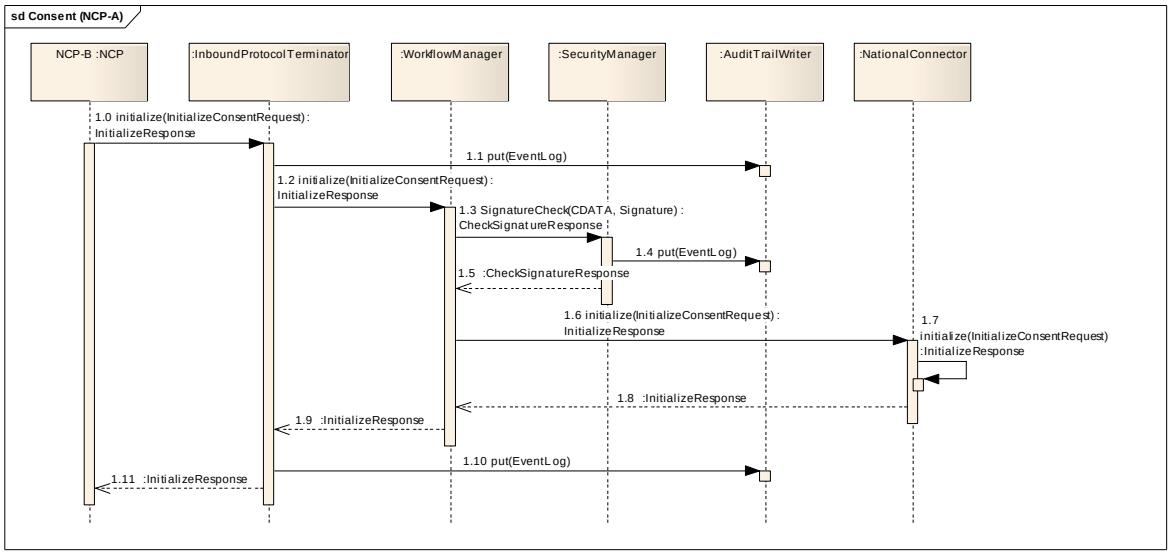


Figure 27 Consent initialization (NCP-A)

- [1.0]-[1.2] Upon the arrival of a *PatientService::initialize()* SOAP request, *InboundProtocolTerminator* calls the *AuditTrailWriter* to log the consent initialization request in the audit trail, and forwards the resulting object to the *WorkflowManager*.
- [1.3]-[1.5] By calling *SecurityService::SignatureCheck()*, the signature of the consent initialization is verified. The corresponding record is made in the audit trail.
- [1.6] The WorkflowManager calls DispensationInboundNational::initialize() method of the NationalConnector.
- [1.7]-[1.11] The *NationalConnector* makes an appropriate record in the audit trail and issues a corresponding request to the national infrastructure. The response *InitializeResponse* should contain the identifier of a defined notification acknowledgement.
- [1.12]-[1.16] *InitializeResponse* is returned to the *InboundProtocolTerminator* with corresponding records being made in the audit trail. The *InboundProtocolTerminator* wraps the response into a SOAP envelope and sends it to NCP-B as a SOAP response.

4.5.2 **Revoking a Consent**

The sequences for revoking a consent are identical to the sequences presented on Figure 26 and Figure 27 with: *initialize(InitializeConsentRequest):InitializationResponse* calls substituted by *discard(DiscardConsentRequest):DiscardResponse* calls.

5 Central Services

5.1 Central & shared service goal and responsibilities

The central & shared services components deal with those common data that need to be maintained for and distributed to each NCP to facilitate the operation of specified epSOS business transactions. The objective of central services is to create, maintain and share common data in a secure and efficient manner. Each service must map to a responsibility of a real-world entity that fits the epSOS organizational structure.

5.1.1 Common data objects

Common data in the context of epSOS technical infrastructure has following distribution types:

(1) identical for all MS participating in epSOS

(2) specific for each NCP but derived from a common source

(3) specific for each NCP and public to all other NCPs

Common data consists of following classes[3]:

Data class	Distribution type
a) Master Value Catalog (MVC)	(1)
b) List of participating MS (“MS-List”)	(1)
c) each member state's Master Translation Catalog (MTC)	(2)
d) Trust anchors	(3)
e) End point addresses, VPN, NCP, IdP and HCP certificates (in TSL)	(3)
f) MS policies & documents	
1. (Patient) information paper (D2.1)	(3)
2. Confirmation requirement policy (D3.6)	(3)
3. HCP allowed roles (D3.6)	(3)
4. NCP PS eP data hiding disclaimer (D3.4.1)	(3)

3This list is intended to be complete.

Data objects used outside the scope are email communication, documents on project place, evaluation and audit reports communicated out-of-band that are either not operational or no configuration data for an NCP.

5.1.2 Implementation overview

There will be two types of services: Central services with a single distribution point and shared services with a distribution point per NCP.

a) Central services

provide MS-List, MVC and MTC.

The Management of the MVC and MTC will be provided by the epSOS Central Repository Terminology Server (eCRTS). eCRTS is based on CareCom's HealthTerm.

The Responsibility for this component is to provide to each member state a web based user interface to develop and validate their MTC. and also a webservice that can be called from the NCP (TSAM) to download the MVC and the appropriate MVC and MTC to the NCP. For a more detailed description see Chapter 5.3.

The requirements for hosting including Service Levels will be described in D3.8.x

b) Shared services

Shared services, formerly called “virtual central services”, keep all data objects from distribution class 3. A list of common configuration data needed by the NCP is described in Chapter 5.3

5.2 MS-List service

The MS-list is a document signed by TPM and uploaded via Webform or WebDAV using a plain authentication over TLS. It is published as HTTP-service to the NCPs. Usage of TLS is optional, as the document signature is good enough to protect the data. NCPs download the file using HTTP-tools (e.g. curl) to update the list periodically, like every night.

It should be deployed on the eCRTS-server to group similar services.

5.3 Configuration Repository Manager

The configuration manager and configuration management support tools will be part of the F.E.T common components.

The configuration manager implementation & support tools are currently under development.

Planned are the following tools

TSL Editor (Editor for handling TSL configurations)

NCP TSL Synchronizer (batch tool for copying TSL configuration files)

Planned are the following programmatic interfaces

NCP Routing Manager (Access to the core NCP routing functionality)

NCP Configuration Manager (Access to the NCP configuration information)

F.E.T will provide detailed/low-level design on ProjectPlace: **https://service.projectplace.com/pp/pp.cgi/0/499890943**

5.3.1 **Functionalities and responsibility**

Item	Responsibility	Description
Information Structure	NCP organisation	Versioning: two visual versions: Drop-zone (TO-BE) & Running (AS-IS), named: `drop` & `running` Underlying the information should be placed on a SubVersioned structure for full traceability (not shown in diagram). Structure for suggested central information for organization contact data, NCP TSLs & semantic MTC & MVC <version>/ <version>/organisation/ <version>/organisation/contacts.pdf <version>/NSL/*.TSL (D3.4.2) <version>/MTC/… (Semantic Expertise Group) <version>/MVC/… (Semantic Expertise Group) The information structure should be accessed in a REST way, as resources via HTTP protocol. Therefore the structure can be virtual in itself. Root of central services is `centralservices`
DNS	NCP	epSOS owns the 'epsos.eu' domain and is

Item	Responsibility	Description
name standard	organization (epSOS & EU-hostmaster)	responsible for ordering the requested DNS name for the MS NCP. The name standard follow the pattern: <country-code>.epsos.eu/centralservices/<version>/<hierarchy>…/<document> **EX**: Getting the current organization contacts for Denmark https://dk.epsos.eu/centralservices/running/organization/contects.pdf
eCRTS Management tool	epSOS / SALAR	The carecom tool for editing and managing the epSOS and MS MVC & MTC. The tool guarantees that the information inputted by epSOS and MS is storage and transferred to the NCP – drop-zone, version `drop` safely and in intact form. To ensure safe transfer, file-trasfer over HTTPS is used with NCP certificates. Alternatively safe transfer could be on of the following: SFTP Webservice over HTTPS

5.3.2 **Architecture and interfaces**

This document is based on the technical note on central services by Rainer Hörbe and discussion hereof. Please see this note for a full discussion of all proposed solutions. Introduced here is a solution based on one of the central services proposals mentioned in the technical note:

> *Virtual service based on DNS and HTTP Distribution: HTTP on distributed servers. Each country would maintain a HTTP-service on its NCP, that provides the documents in its*

responsibility. Distributed addressing is based on the convention that NCPs follow a certain naming scheme, like https://country-code.epsos.eu/centralservice/docname

5.3.2.1Overview

The solution addresses information structure & versioning, service addressing, network & transfer security aspect and more importantly service responsibility.

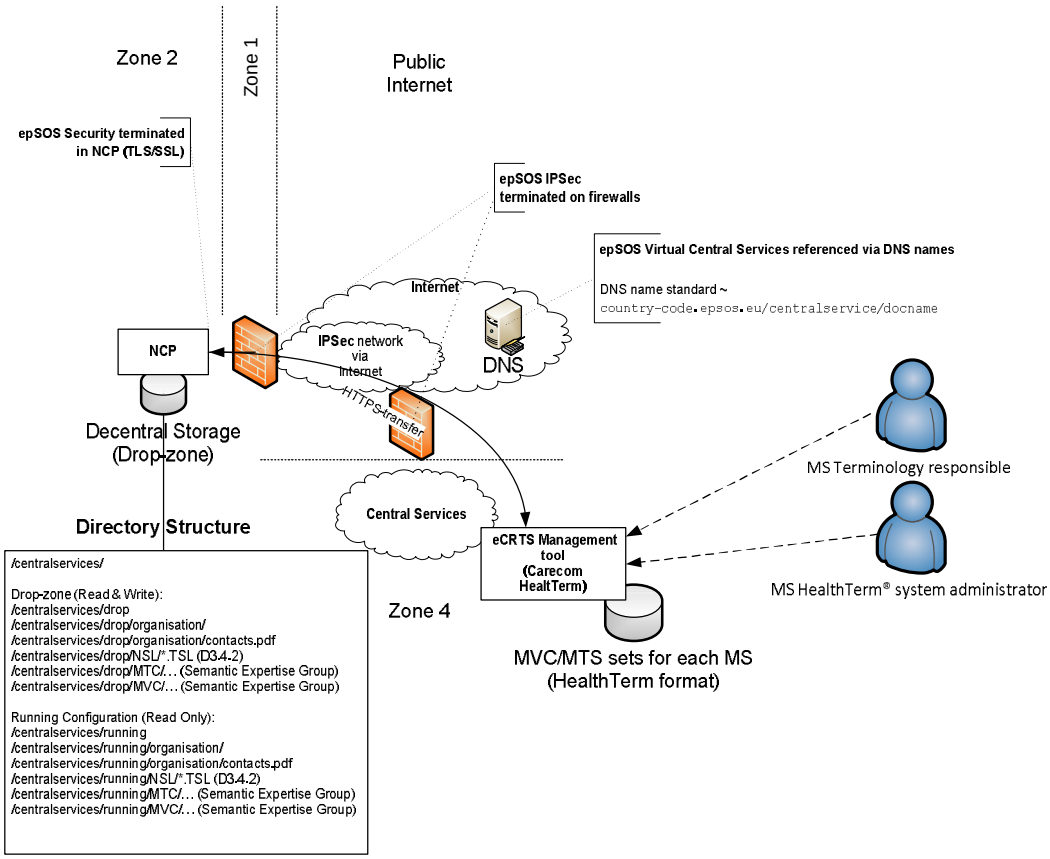


Figure 28 Virtual Central Service - proposed solution

5.3.3 List of configuration data needed by NCPs

Key (Configuration item id)	Description of Configuration Value
Local.CentralServices	Path where local copies of configuration da stored. the default value is "%epsos%/centrals
Local.NslDropZone	Path were the NCP's own NSL is maintained is "%Local.CentralServices%/drop/NSL"
Local.NslRunningZone	Path were the current version of the NCP's N

Key (Configuration item id)		Description of Configuration Value
		access by other NCPs. The de "%Local.CentralServices%/running/NSL".
CoT.CountryCodes		Semicolon-separated list of all countries that a same circle of trust as the NCP. Example: "dk
Local.LanguageCodes		NCP-supported LanguageCodes (ISO 639-1 PoC Registration Data for multi-lingual MS
Configuration for each connected NCP / country **ZZ** is short country-code	ZZ.nsl.URL	URL of the Austrian, German, Danish, etc. N "https://countrycode.epsos.eu/centralservices
	ZZ.OrderService.WSE ZZ.PatientService.WSE ZZ.DispensationService.WSE ZZ.ConsentServce.WSE	Webservice endpoints of the national epS WSE are taken from the countries' NS Synchronizer and written into the configuratio entries MUST NOT be managed manually.

5.4 **eCRTS**

Within the usage of the epSOS Central Reference Terminology Server one can distinguish a series of informational scenarios.

The main use case of the service is the transactional translation scenario of each (coded) concept included in the original Patient Summary, ePrescription and eDispensation documents.

The transactions involving the Patient Summary, the ePrescription and the eDispensation will be sent from originating MS in his language and received by another MS in his language. The switch between the two languages takes place by a lookup in the epSOS Master Translation/Transcoding Catalogue (epSOS MTC) through a standard interface.

Users of the epSOS Central Reference Terminology Server (HCPs, NCP employees, epSOS terminology curators) will further submit concepts or terms applying for integration into the epSOS MTC with the help of the epSOS Central Reference Terminology Server.

Submitted artifacts may be carefully analyzed by medical terminology experts and eventually included in future releases of the epSOS MTC.

The NCPs will be notified of the existence of a new terminology release and will eventually update their locally used epSOS repository if they are using the distributed terminology access services model.

Transcoding of the concepts involves a complex decisional scenario in finding semantic equivalents between two code systems. Since commonly used official terminologies or controlled medical vocabularies are most often hierarchically organized, granularity issues are quite common in transcoding projects. This often results in 1-n or n-1 type mappings.

Language translation of display names or surface forms is also a common scenario. An initial translation of epSOS Value Sets into local languages and subsequent translations/retranslations when new releases of Value Sets are published.

5.4.1 Creation of national translation / transcoding respoinsibilities and technical aspects

The epSOS project has through inputs from a team of Semantic Experts created the epSOS Master Value Set Catalogue (MVC).

The MVC (Master Value Set Catalogue) is a data structure containing all the value sets created and selected by the epSOS project.

The value sets should represent the codes needed for a Member State (MS) to be able to send the information required in the three pivot documents: ePrescription, eDispensation and Patient Summary.

First the MVC was created in an Excel file. The excel file contains numerous sheets where each sheet represents a value set. Every value set has a name, an OID, a column with the codes and an adjacent column with the English display names representing the codes.

The value sets in the MVC can be entire coding system and sometimes part of a coding system.

Therefore, if an MS should have all the information needed to be able to make high quality translations and transcodings of all of the selected coding systems the full coding systems is needed and is available for the MS in the epSOS Central Reference Terminology Server.

Experience in translation projects all over the world has demonstrated the importance of having the full code systems available in a browser in order to give the translators a good understanding of the context in which the concepts' definitions and the relationships between the concepts in a given code system shall be interpreted.

All of the epSOS code systems are available in the browser of the epSOS Central Reference Terminology Server and all the MVC value sets is created, saved and logged with a timestamp and a version ID to ensure traceability in the maintenance of the MVC in the future.

The value sets can now easily be copied by an MS and saved with the information of the epSOS MS namespace identifier. Hereby an MS can begin to translate the value sets in their own national language (dialect).

The list of Coding Systems and of Value Sets is reported in Appendix B2.

MTC creation tasks

There are 1-3 tasks for each MS to perform to be able to make their national translations and transcodings in accordance with the requirement to deliver the data needed in the epSOS pivot documents.

The first task is mandatory for all MSs. The second and third tasks are only required if an MS uses another code systems than the ones selected in the epSOS Reference Terminology to represent the data in a value set and if they wish to see the data in their own systems and not only wrapped in an additional pdf file.

The 3 tasks are illustrated in the next figure and described below:

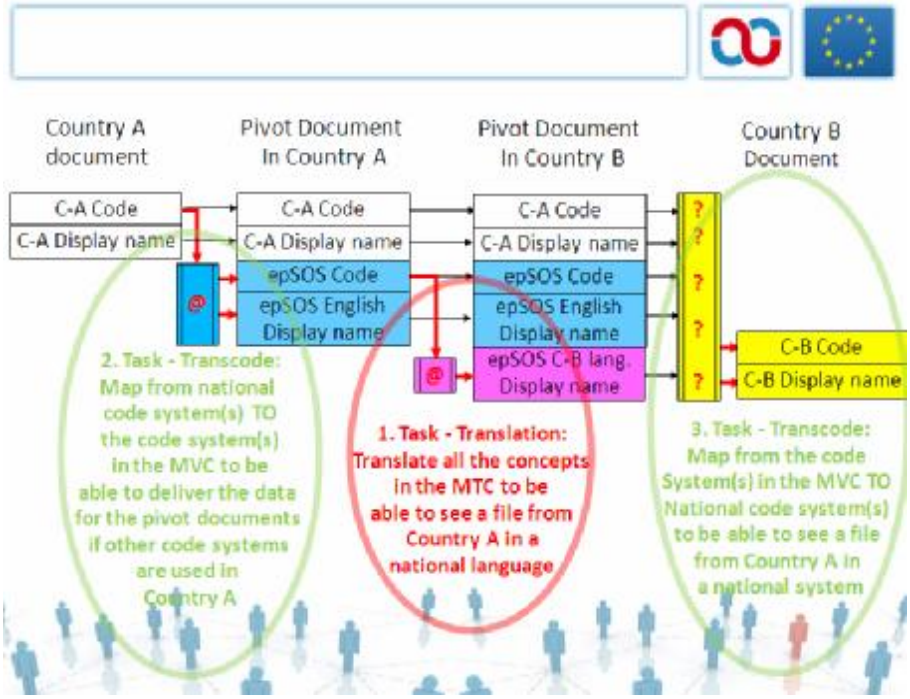


Figure 29

1. Task – translation of the MVC (Mandatory):

Each MS in the epSOS project has to complete a translation of all the concepts in the MVC to be able to read a file from Country A in the MSs national language.

The MSs can use the functionalities in the epSOS Central Reference Terminology Server to access the MVC value sets and prepare them for translation (for more

details see the section about translation functionalities). The MSs can thereby translate the value set to their national language using the epSOS Central Reference Terminology Server's translation module.

The translations are hereby saved and logged in the epSOS Central Reference Terminology Server and can be used for the transactional translation scenarios.

2. Task – transcode FROM national code system TO the MVC (semi-optional):

If an MS uses another code system nationally than the one selected to represent the data in the value sets the MS must transcode/map from the national code systems to the code systems in the MVC to be able to deliver the data required in the pivot documents.

The MSs can use the functionalities in the epSOS Central Reference Terminology Server and the already created MVC value sets and prepare them for a transcoding/mapping (for more details see the section about transcoding/mapping functionalities). The MSs can thereby transcode/map their national code systems to the code systems in the MVC using the epSOS Central Reference Terminology Server's transcoding/mapping module.

The transcodings/mappings are saved and logged in the epSOS Central Reference Terminology Server and can be used for the transactional transcoding scenarios.

3. Task – transcode FROM the code systems in the MVC TO National code systems (optional):

If an MS wishes to see a file from Country A in the MSs national systems and uses other code systems than the ones in the MVC then the MS must transcode/map from the code systems in the MVC to national code systems to be able to see a file from Country A in the MSs national systems.

The epSOS Central Reference Terminology Server provides this additional opportunity for the MSs by providing the transcoding/mapping module in the epSOS Central Reference Terminology Server. It is not in scope of epSOS 1, but up to the MS to decide if this task is needed in their own country.

Creation of the epSOS MTC

When the epSOS project and the MSs have fulfilled the tasks described below then the MTC is created and the epSOS Central Reference Terminology Server is the epSOS MTC.

Responsibilities from epSOS

None other than make the epSOS Central Reference Terminology Server available for the MSs

Responsibilities from Member States

If an MS already has translated some of the value sets in the MVC, then these translations can be imported in the epSOS Central Reference Terminology Server by CareCom (Supplier of the eCRTS).

The translation can be given a status either in the epSOS Central Reference Terminology Server as 'translation finished and Quality Assured' or the translation can be processed into a certain stage of the translation workflow for further QA using the translation module.

The value sets being part of the MVC in the epSOS Central Reference Terminology Server that still needs to be translated must be placed in a translation workflow. Using the value set module an MS can save the value set including the epSOS MS namespace identifier. Hereby an MS can begin to translate the value sets into their own national language (dialect).

When the MS translates using the translation module, the translation is saved with an ID also containing the epSOS MS namespace identifier. This secures the traceability of which MS has made which translations in the epSOS Central Reference Terminology Server.

When the MS has finished translating and QA'ing, then the MS's mandatory tasks regarding the MTC have been fulfilled.

The next task may be to transcode/map using the mapping module in the epSOS Central Reference Terminology Server. If an MS uses another code system than the ones selected in the MVC to represent a certain value set then the MS must transcode/map their national code systems to the code systems selected in the MVC.

Before a MS can begin to setup a transcoding/mapping project in the epSOS Central Reference Terminology Server the MS must get CareCom to import their national code systems into the epSOS Central Reference Terminology Server.

When the national code systems have been imported, an MS can use the mapping module in the epSOS Central Reference Terminology Server to setup a mapping project (including assigning roles to persons in the different stages of the mapping workflow) and begin to transcode/map their national coding systems to the epSOS value sets.

When the MS has finished transcoding/mapping, then all the MS's tasks regarding the MTC have been fulfilled.

5.4.2 **eCRTS architecture and interfaces**

The eCRTS is a web based application which contains several modules for handling the terminology task which have been identified within epSOS. Some of the modules are described underneath:

User administration

User administration is done in several layers:

- Namespace administration
- Usergroup administration
- User administration

Namespaces:

Each MS is given an epSOS namespace identifier. The epSOS namespace identifier is a 7 digit identifier which the MSs national code systems is linked to. When an MS creates value sets, translation, transcodings etc. The epSOS namespace identifier is contained in the identifiers of the various components. Rights administration can be controlled using the epSOS namespace identifier.

User Groups:

An epSOS Terminology Administrator having been given the rights to create new users and user groups can create User Groups.

A user group is assigned a number of privileges. A User Group may contain a User Group, hence a hierarchical structure controls User Group placement.

The epSOS Terminology Administrator selects from the list of available privileges the ones that are relevant for this group. Amongst these privileges it is worth mentioning

- Namespace rights: Browsing rights, Editing rights, Upload rights, Requestor rights, Distributor rights, Value Set Access: From the list of available Value Sets the administrator selects the ones which are relevant for the user groups
- System Accessrights: user administrator rights, organizing rights etc.
- Content rights - Roles: The epSOS Terminology Administrator selects from the available list of roles the roles relevant for this user group (Translator, Reviewer1, Reviewer2, SME, Mapper1, Mapper2 etc.)

Rights:

Privileges of a user group may be inherited from a parent user group

User administration:

Users can be stated as active or inactive. The individual rights are given to the users by assigning them to one or more user groups.

Individual users have access to set a number of personal privileges:

Setting different filters to be applied to the many search facilities in the ECRTS, selecting the language to be displayed in the ECRTS user interface.

Browsing:

The ECRTS Browser module allows you to browse the code systems an ECRTS user have been granted access rights to.

The Browser window should allow the user to see the hierarchical structure of the classification as a tree structure. The browser window should also allow the user to select a concept and have its characteristics displayed.

The detailed view of a concept should contain information about:

- Any parents and children
- Fully specified name, concept ID/codes, Preferred Term, any synonyms,
- Relationships and inverse relationships and properties (dependant on the nature of the classification) including the relationships' statuses, such as current or retired.

Value set creation and maintenance:

The Value set module shall have a functionality that allows you to

- Display the hierarchy of created value sets
- Select a Value Set and display its IDs, version, language and status
- Given the right privileges you should be able to create new Value Sets. A large number of operations may be executed on Value Sets:Add members from Query by selecting single concepts, subhierarchies, other subsets either by adding or subtracting portions of concepts or by selecting the intersection one two sets
- Add members via import of external file(s)
- Replace inactive members: When new releases of classifications are published the NCPs may use the tool to identify the number of Value Sets that a given, changed concept is a member of and from this knowledge the NCP can decide whether to exchange the present concept or retain the concept in its old form

Translation:

The eCRTS comprises many functionality specific modules. The Translation module is one of the large, comprehensive modules that governs the complex workflow that all ontology related translation projects must go through in order to uphold high quality translations. This module should also offer many support features, aiding the translators and reviewers in the translation process.

Translation work is optimally based on extracting small subdivisions (subsets) of the larger classifications. These subsets are most often created as a collection of concepts that will be used in a common context. This context may be a clinical

specialty, it may be a certain eDocument, such as the ePrescription etc. The epSOS Value Sets already created may be regarded as such subsets.

When new situations require new or changed Value Sets, the ECRTS Value Set module (Subset module) described above may ease the workload of constructing these considerably.

Assignment of rights:

A Subset Administrator appointed by an NCP can create subsets within the namespace(s) he has access rights to

A Concept Manager can subdivide a Value Set into smaller batches selected from criteria relevant to the experience of the translators. Further the Concept Manager can assign batches to individual translators thereby entering the batches into the controlled workflow

MSs may choose different models for building their translation team. In some instances the MS will choose to employ an external translation bureau and supplement this with a number of internal resources with the sufficient clinical skills to review the translations done by the bureau. In other instances the translation process may be internal only to the NCP. Depending on the translation project setup different stages may enter the workflow. The ECRTS Translation module handles a dynamic construction of workflows.

Translation workflow:

A Translation workflow should comprise a number of stages that a concept translation passes through. These should as a minimum allow for these stages: Translation (external or internal), Review 1 (external), Review 2 (internal), SME (Subject Matter Expert, internal), Editorial Board (internal)

In a translation workflow it should be possible to go back and forth between different stages. Example: A reviewer is dissatisfied with a particular translation. He should be able to comment on this and revert the translated concept to the translation stage. This process may take place several times. Hence, the Translation module should be capable of registering these events in a history log.

As a quality assurance measure the translation module should be governed by user access rights, e.g. the system should block a translator from reviewing his own work.

Concordance Search

During a translation process it should be possible to view already performed translations of concepts which are semantically similar to the concept about to be translated.

An automated concordance search should contain the following selection criteria

- Language and
- Classification/terminology

Having a concordance search available facilitates two major issues in translation:

- Inspiration to the translation at hand, as the translator can see translations of similar concepts done by his colleagues (or himself)
- Quality measure, as it ensures a greater linguistic consistency of the translations

Statistics:

Comprehensive statistics are necessary for controlling the translation process. Hence the ECRTS should have a Statistics module that can generate statistics for each individual participant in the translation workflow. It is of vital importance to be able to gain an overview of the overall progress of one or more batches/subsets to be able to report to project management

References:

The ECRTS should allow many types of references be attached to the concepts being translated. This also applies to each stage in the workflow. References are most often based on uploaded documents accessible with the proper user rights. The documents may be medical encyclopaedias, text books, or linguistic guidelines. These references should be used to describe, .for example, why a certain translation has been used. References may also be made to external sites, such as web pages.

All types of references should be direct, i.e. clicking on a reference made for a translation should “open” the document on the right page, in the right line, and if more than one occurrence exists, all occurrences should be highlighted.

Transcoding/mapping:

The transcoding/mapping module has functionalities that allows ECRTS user to

- Create mapsets which are the folder for each transcoding/mapping project
- Create and select textual rule sets which can be used in the individual transcoding/mappings
- Create and select category set for helping the user link contextual categories to concepts which are declared unmappable
- Select how big an amount of concepts should be in a transcoding/mapping project and granting the rights for the roles in the mapping workflow
- Provide mappers and reviewers with:
    - search, browsing functionalities while doing the transcoding/mapping

- help finding the equivalent match for a transcoding/mapping in automatically finding semantically equivalent/similar concepts in the other code system
- information of the progress in the mapping workflow

Export/distribution

The eCRTS server contains an export distribution module were export of published data in the eCRTS can be exported to a secure ftp site, but it is more likely that all data transfers from the eCRTS to the MS will be performed using web services in the common component.

5.4.3 epSOS Terminology generation and management

Maintenance of the MTC:

epSOS responsibility:

There are several tasks involved in the maintenance of the MTC but with different frequencies:

1. If an MS has requested an error correction in the MVC or identified an omission, then the epSOS project's epSOS Terminology Administrator has a responsibility for obtaining an agreement of the amendments requested. The changes can be edited in the value set modules in the epSOS Central Reference Terminology Server. Or the epSOS Terminology Administrator can notify the change requestor by sending an email to either an MS or all MSs and delegate the editorial changes needed.
2. If one of the code systems has been updated in the MVC due to a new release, then the epSOS Terminology Administrator notify the MSs by email telling which value set(s) have been updated and if additional translation is required in the MTC.
3. If one of the code systems has been updated in the MVC due to a new release, then the epSOS Terminology Administrator notify the MSs by email telling which value set(s) have been updated and additional transcodings/mappings are required in the MTC.

Member States responsibility

There are several tasks regarding maintenance of the MTC but with different frequencies:

1. If an MS has discovered an error in the MSs responsibility in the MTC, then the changes can be edited in either of the maintenance modules in the epSOS Central Reference Terminology Server:
    a. Value set

b. Editing

c. Translation

d. Mapping

2. If the epSOS Terminology Administrator has delegated editorial changes, the MS must support the epSOS Terminology Administrator in completing the editorial change.
3. If one of the code systems has been updated in the MVC due to a new release, then an MS will have received a notification and the MS must translate the additional concept(s) which require a translation for the MTC.
4. If one of the code systems has been updated in the MVC due to a new release, then an MS will have received a notification and the MS must transcode/map if additional concepts require a transcoding/mapping for the MTC.

6 Guidelines to define the Low Level Design

Guidelines and processes for quality assurance have been provided by F.E.T, detailing the expected formats, delivery contents, testing and review processes. Two documents have been delivered (**https://service.projectplace.com/pp/pp.cgi/0/499311482**):

- epSOS CCD - Quality Assurance Process Description and Test Plan
- epSOS CCD - Documentation and Coding Guidelines

Example of documentation following the FET guidelines is the common component Audit Manager. The Documentation found at project place: https://service.projectplace.com/pp/pp.cgi/0/499312679

7 NCP-in-the-Box Reference Deployment Composition

Since all major actors or components inside the NCP-in-a-Transparent-Box are introduced in previous chapters, now a closer look how they might be deployed is given.

Introductory, the figure below depicts the placement of both NCPs. A Thin Client (Browser-based epSOS Portal or Front-end) enforces an authentication via a trusted Authentication Service, i.e. Identity Provider. The protocol used (either SAML Protocol or WS-Trust Protocol) is out of scope for the NCP implementation. Only the outcome of the Identity Provider – the (signed) SAML assertion (Identity Assertion) - issued by a trusted party located in the national infrastructure - is relevant. The epSOS Portal invokes the NCP-B providing all specified epSOS services. The Workflow Manager in NCP-B then sends by using the Outbound Protocol Terminator a cross-border SOAP request to the NCP-A of the patient's home country. This SOAP request is fully compliant to the epSOS D3.4.2 normative request message specification. Then, the NCP-A processes this request by using the common components bundled with the NCP-A. It furthermore uses the services that are located in the national infrastructure. In any case, the NCP-A responds with a message according to the specified D3.4.2 epSOS message format.

Figure 30 shows the proxy semantic of a NCP-B. That is, a generic Java-based invocation message is transformed into the epSOS format and forwarded to another NCP-A, and finally mapped on national specific service requests. For simplicity, only relevant components are shown, those which have an impact on the communication with involved services or other components. The flow of control with regard to the common components of the NCP is given in the next section.

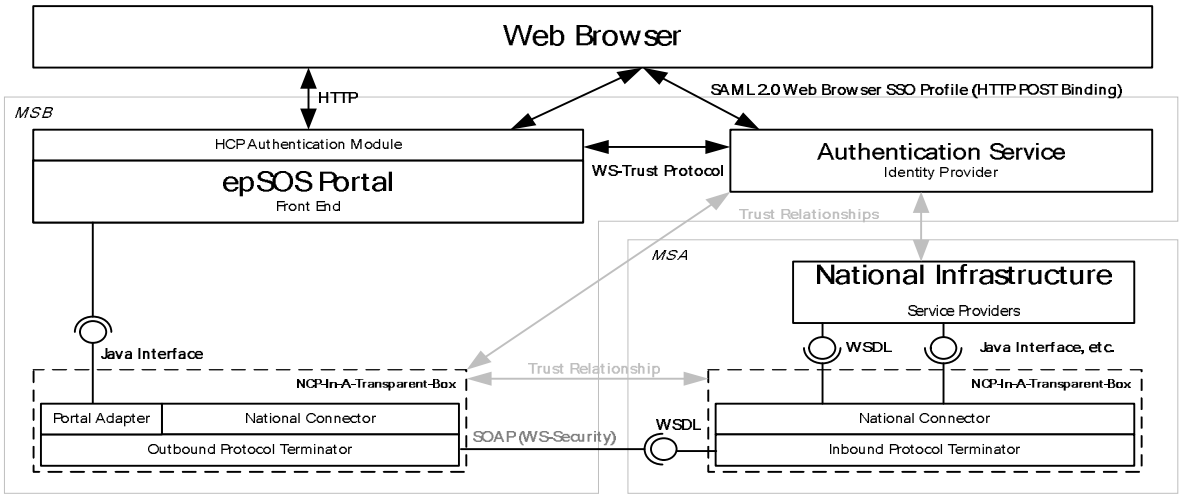


Figure 30 Logical View for both NCPs[4]

7.1 Navigation Paths

Since the NCP takes different roles in the Circle of Trust, the navigation paths for the message flow are exemplary shown below. Therefore a NCP provides three interfaces which are also described. These ones are the (1) NCP-B Interface, the (2) NCP-A Interface, and the Central Common Service Interface.

7.1.1 NCP-B Role

The epSOS portal invokes a method provided by the portal adapter interface of the National Connector. The portal adapter forwards the request to the Workflow Manager, that makes use of various common components in order to process the request:

- the provided assertions need to be validated, e.g. it has to be checked whether the epSOS profile is used.

- routing information needs to be obtained from the epSOS NCP configuration

- audit trail entries need to be written

- for eDispensations a mapping of the national format and taxonomies to the epSOS format has to be done. For ePrescriptions and patient summaries a mapping from the common epSOS format to the national format and vocabularies has to be done.

Finally, a SOAP request message is created and sent to the cross-border NCP-A. The response is mapped on the interface of the National Connector so that the epSOS Portal is able to render it accordingly.

[4] This figure shows a sample deployment. E.g. instead of WS Trust the SAML protocol could be used as well for obtaining the HCP identity assertion.

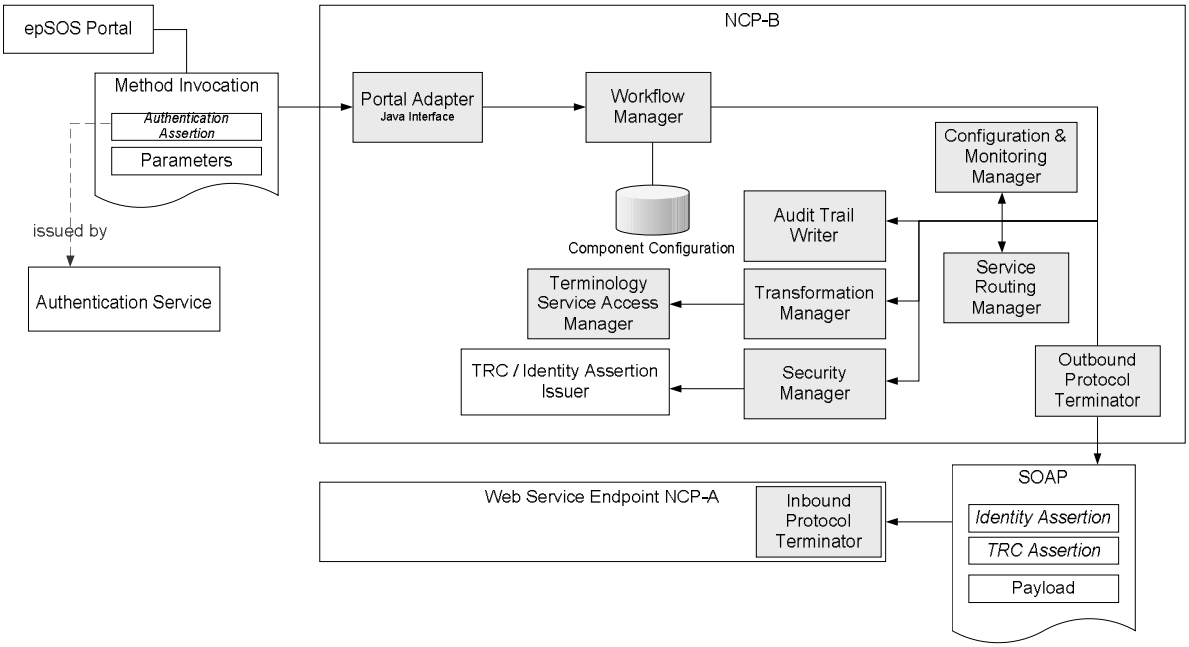


Figure 31 Overview for NCP-in-a-Transparent-Box in NCP-B Role

The epSOS Front-end is logically separated from the NCP-in-a-Transparent-Box. However, it is tightly bound to it, because the Front-End provides the user interface for the epSOS services which the NCP delivers or supplies. What is solely taken into account when regarding the NCP-B role is the communication between the Front-end and the National Connector that the NCP-B provides. Until now, the interface to be specified is an ordinary Java one, which accepts POJOs (Plain Old Java Object) as parameter values. Further information on the technology that implements the serialization and de-serialization of them is given in section 3.6.1.

7.1.2 **NCP-A Role**

The incoming SOAP messages are processed by off the shelf Web service frameworks (cp. section 5.21). A registered Web service endpoint listener is called by the framework. Previously, the SOAP message was validated against the policy according to the WSDL document. For instance, this includes the token mapping (i.e. are SAML assertion(s) and signatures present), or the validation of timestamps stated in the message header. SOAP message handlers provide the possibility to access the entire SOAP envelope and might pre-process relevant data. The Endpoint Listener issues synchronous (blocking) requests to the Workflow Manager component. Then, this component uses registered components in order to process the request. In the end, the services located in the national infrastructure are invoked by means of the National Connector in order to create the SOAP response for NCP-B. The following figure sketches the role of a NCP when it is called by another NCP-B.

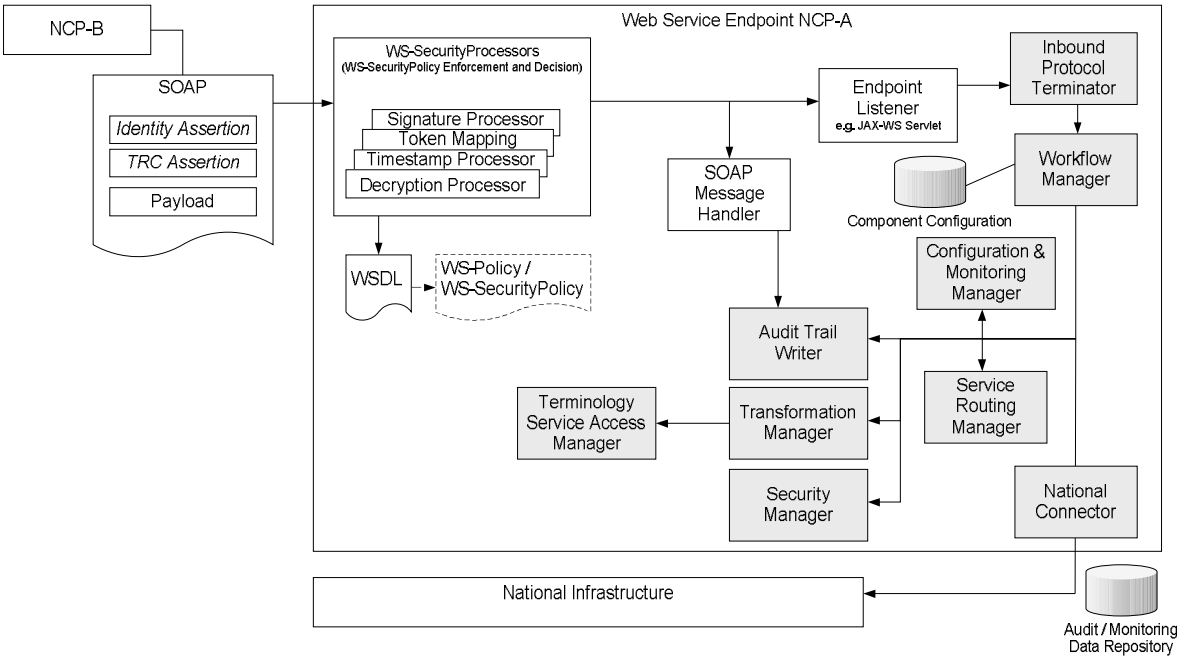


Figure 32 Overview for NCP-in-a-Transparent-Box in NCP-A Role

7.1.3 Central Common Service Role

There exist several external services that are consumed by a NCP implementation in order to process a request. In the TPM meeting of February 17(th), 2010 it was decided that the configuration data should be processed using a static document that is shared among MS. Regardless of the origin of the configuration (e.g. document or dedicated services), each NCP has to be initialized and put into operation in a normalized manner. This necessitates that a common interface is available for relying components inside the NCP. Section 3.2.10 introduced the Configuration and Monitoring Manager which encapsulates the source of the configuration. The access to the configuration data needs to be defined subsequently.

7.2 Deployment

Comments to the deployment diagram:

- Segregation of duties requires that the audit repository and NCP service MUST run in separate administrative domains. This does not require separate physical servers, but safeguards, that the NCP administrative staff does not have access to the audit repository. (D3.7.2 does not propose an epSOS-centralized audit service; the audit repository is still in control of the member state).
- Transport layer security configuration is dependent on whether a web service proxy must be installed. (Opinion: Using a web service proxy makes

certificate management more cumbersome, for a very small security advantage.)

There are no specific hooks for monitoring services at the application level; (Except for real-time abuse detection, but that is yet unspecified). The following figure sketches the exemplary deployment of the common components in a NCP and its related services used.

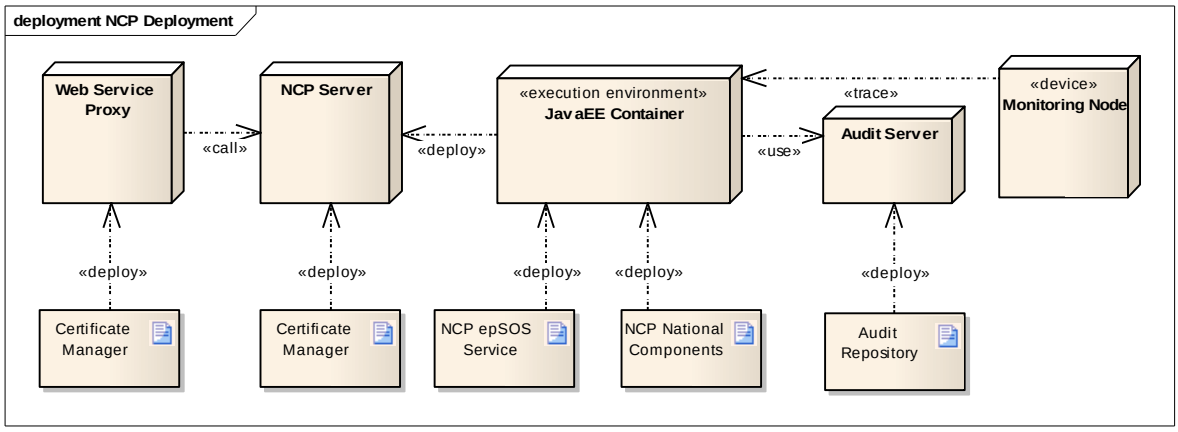


Figure 33 – Exemplary Deployment of the NCP Components

7.3 Implementation Platform

epSOS Common software will adopt Open Source frameworks and applications as much as possible.

Reference implementations will be made available on a single platform. To open vendors a wide range of options, the only requirements to have Linux as an operationing system

Vendors will have to specify the dependency of their solution on languages, frameworks, APIs and applications; the detailed specification will contain a sample list. All choices of frameworks, libraries application environment must be compatible as specified from source and by vendor test.

The goal of the following table is to provide a first list of software languages, frameworks and applications for epSOS software commonly developed components and for epSOS commonly developed Front-end.

Element	Required	Options / choices / Recommendation
Languages	Java EE	Choice between version 5 or 6
Operating System		RedHat Enterprise Linux (64bit) Choice between version 4 or 5

Element	Required	Options / choices / Recommendation
Application Server		Recommended choice between JBoss v.4 or 5 Glassfish v.3 Tomcat v.5.x / 6.x
GUI		Recommended build upon JSP/JSF
Security SAML Framework		Recommended OpenSAML library
Security WS-* Framework		Choice between SUN METRO AXIS2
Audit Framework		Recommended using log4j for audit logging

7.4 NCP Front-end Portal

epSOS will deliver a Portal for Country B and for Country A two functions and additional the necessary Web Services that the MS can develop their own Portal A. In this chapter the functionality of the portal solution will be described.

7.4.1 Country B Portal

Process and workflow is described in WP 3.6 D3.6.2.

Required functions:

- HCP Authentication
- Country Selection
- Patient Identification
- Patient Consent
- Confirmation
- View of Patient Summary
- View of ePrescriptions
- eDispensation

7.4.2 Country A Portal

Process and workflow is described in WP 3.6 D3.6.2.

Required functions:

- HCP Authentication
- Patient Consent – pre Consent

7.4.3 Functions

HCP Authentication

MS has to provide a HCP/HCPO Database or a LDAP to authenticate the HCP as a medical doctor or as a pharmacist. The Portal defines a web service (Portal Adapter) to receive from the National Infrastructure the HCP, HCPO, Role, ….

Country Selection

Country A of the patient has to be identified. This can be done either by a combo box or selecting the flag.

Patient Identification

According to the selected country a different search form will be displayed. Each country can have different identifiers to search for patients (Identifier definition see

WP 3.6 D3.6.2). Some also permit to search by demographics (name, address, birth date, gender etc).

Patient Consent

Next step is to check if a valid consent exists for this patient for Country B. The consent documents are always stored in Country A.

If exists the process can proceed,

if not, a new consent in country B for country B can be created. This consent will be send then afterwards to Country A. Additional to the Consent Document an Information Paper in the language of Country B and Country A will be printed and has to be signed from the Patient.

Consent Document: Each MS will have his own consent document defined by the Basic Patient Privacy Consent (BPPC) IHE profile and the metadata of this document contents the country code.

Consent can be given for a period of time.

Patient Confirmation (Treatment Relationship Confirmation (TRC))

According to a configuration parameter of NCP (MS likes to have confirmation or not) patient confirmation will be checked. This confirmation includes the patientid, the organization of doctor and a time limit (when to be invalidated). This confirmation will be stored at portal database. The confirmation is only valid within the same HCP organization or for a defined time (e.g. 1 month)

Patient Summary

The portal requests the patient summary documents (max two documents) in CDA Level 3 and PDF format. The portal renders the CDA document and downloads the PDF. The user can select which document he likes to see.

As for the CDA document, for safety reason, the HPC should be able to select if the document is displayed in Country B language, in the English version, in Country A languages with the original codes and if the narrative parts (not translated in epSOS) are shown.

ePrescription

After requesting open prescription a list of prescriptions (CDA Level 3 and PDF documents) will be received. Each document must have a meaningful header to select the right one.

The user selects one prescription and the items will be displayed.

As for the patient summary, The portal renders the CDA document and downloads the PDF. The user can select which document he likes to see.

For safety reason, the HPC should be able to select if the CDA document is displayed in Country B language, in the English version, in Country A languages with the original codes and if the narrative parts (not translated in epSOS) are shown.

eDispensation

Dispensed item will be copied from the prescription and the pharmacist can change manually the dispensation, if necessary.

The dispensation document will be created as a CDA document and delivered to Country A

Consent in Country A for different Country B's

Consent in Country A for different Country B's can only be done by an HCP/HCPO

Prerequisite functions before Consent can be changed (given, revoke, change of time):

- HCP Authentication
- Patient Confirmation (if necessary)

HCP requests the wanted country Consent document and can change this afterwards (Give/Revoke Consent, change of time parameters).

An Information Paper, which will be signed from the Patient, will be printed.

8 NCP Implementation Strategy

This chapter provides the description of the decision process that has brought to the identification of implementation scenarios alternatives and the assignment to Fraunhofer ELGA Team (FET) the task of developing the NCP-In-A-Transparent_Box solution.

It has to be read as decision building process, rather than as decision support at epSOS level. However it is believed it could be useful for new MSs joining the epSOS LSP.

8.1 NCP implementation: Common Components and National Infrastructure Integration

The NCP High Level Design has provided the specification to implement the NCP as a Transparent Box, composed by Components in Common.

The choice of every MS to adopt or not the possibility of exploiting common components and, as a consequence of the common agreement, the development in common of such modules, should be also based on advantages and disadvantages of adopting an NCP-in-a-Transparent-Box.

The main impact of the concept of the common development is that the technical interoperability of the NCP-to-NCP communication is almost granted. This advantage is accompanied by the drawback that the integration of the NCP into the national infrastructure contains some constraints and limitations, e.g. the technology the NCP is based on is determined.

A good part of the integration work and complexity is drawn into the member states and not anymore between the member states. This is a benefit for epSOS interoperability implementation and testing, but it might be a draw-back for MS.

Further **Advantages** might be:

- Development done (almost) only once
- Reduction of the overall development costs for the Member States
- Members of the Industry Team may candidate to develop single components
- Easier to guarantee compliance with security, legal requirements and FWA on a European level
- Simplified integration among the NCP
- Simplified testing procedures (always in line with Overall Testing Strategy and IHE)
- Simplified deployment procedures
- Simplified evaluation procedure

On the other side, possible **Disadvantages** might be:

- Lower possibility to compare different technical solution
- Less customization to MS specific needs/ Customization and integration to the national infrastructure will create considerable effort after the development in common is done.
- Need to manage a European procurement process (other solutions…)
- Need to manage quite a complex development & testing process
- Need to synchronize Common part and MS part development

If it is decided not to develop any NCP components in common, leaving the full implementation to MS, avoiding need for synchronization, the possible pros and Cons should be considered:

- Pros:
    - Apparently shorter development time
    - More straightforward development
- Cons:
    - Higher risk to be mitigated by specifically careful testing
    - Higher implementation and testing costs
    - No optimization and reuse

The Implementation of the NCP and/or the development of the Components in common could be done by the Beneficiaries, or by Suppliers selected with an Open Call for Tender.

MSs, PEB and PSB, while defining the procedures to be followed for implementing the epSOS NCP, were choosing among the alternatives in which the development is done either by Beneficiaries or by Suppliers.

The pros and cons of these alternatives might be summarized as follows:

Developments by Beneficiaries

- Pros:
    - Exploit allocated resources
    - Exploit existing experience
    - Exploit existing solutions
    - Easier management of Change Request
    - No problems with licences of existing products
- Cons:
    - Need to assure industrial level development
    - Need to check availability of skilled resources
    - Existing solutions must be made fully available to other beneficiaries
    - No real contractual leverage to get deadlines respected
    - Maintenance is not assured

Developments by Suppliers:

- Pros:
  - Exploitation of existing commercial solutions
  - Contractual customer/suppliers relationship with clauses to be respected
  - Industrial/professional level of solutions
  - Possibility to negotiate price
  - Maintenance has to be inserted in the contract
- Cons:
  - Suppliers will always use their existing products as a basis for a solution. This contains the following disadvantages:
  - The NCP will be bound to a certain product/ company
  - Extensibility is limited
  - Necessity to adapt epSOS requirements to exploit existing products
  - Products should be made available also to new beneficiaries
  - The option of “Open Sources” might be rejected by Vendors
  - Change request might need contract review

All the considerations, the alternatives, the advantages and disadvantages listed in this chapter, together with planning timing impacts, have been taken into account by TPM, PEB, PSB to define the Implementation strategy to be as opted for epSOS Pilot implementation. In the following chapter three implementation scenarios, built upon these considerations, are presented.

8.2 **Implementation scenarios (X, Y, Z)**

Starting from previous considerations, it was commonly accepted that a perfect solution to address a wide range of risks in an extremely varied MS environment does not exist.

Hence it was decided to identify a small number (3) of scenarios:

- that can be selected or mixed,
- that minimizes some of the most significant risks,
- that offer at a reasonable cost enough flexibility to epSOS, MS and IT
- that achieves effective interoperability.
- that allows MS, for their epSOS solution, to either:
- directly engage with the vendors of their choice
- leverage as much as possible vendors components in an epSOS integrated implementation solution
- Use one or more vendor epSOS solution (NCP/Portal) sponsored by epSOS

The software implementation identified objectives to be achieved, in order of priority were:

- To meet the epSOS schedule for the start of the Pilot on January 2011; that implies that the Projectathon should be performed in November 2010;
- To enable the MS to build their NCP consistent with their national infrastructure and epSOS interoperable (technical, semantic and legal) in a cost effective way and with limited risks;
- To perform epSOS interfacing in a way that is sustainable for future development and operation.

Building/using Common Software Components is not an objective, just a means among others.

Three possible non-mutually exclusive scenarios are described which are:

X:It is the ordinary approach in which a MS develops its own NCP. Industry team provides catalog of SW Components/Solutions for reuse and adaptation. MS develop their epSOS interfacing by selecting within this catalog or beyond. No JARF funds are being used.

Y:It is based on FHGISST/ELGA proposal, improved by reuse of existing industrial components. Development by a JARF funded team to (1) reuse/adapt from vendor provided catalog of SW Components, (2) develop SW Components if needed, and integrate them as an NCP-in-a-box. This team is led by an epSOS beneficiary and includes vendor contributors. MS develop their interfacing to this NCP.

Z:It is based on Vendor offering of complete NCP/Portal solutions. epSOS selects one or more existing vendor solutions that deliver either a NCP-A and/or NCP-B, and/or NCP-B portal. Use JARF fund to customize to the exact epSOS requirements. MS develop their interfacing to this NCP

The 3 scenarios allow covering MS / epSOS / IT requirements and expectations; in general, all mentioned scenarios:

- may be in line with the epSOS timing (for those countries not having to do a public call for tender)
- not mutually exclusives / not conflicting,
- coherent with HLDD and D3.1.2 – D3.7.2
- in line with epSOS testing strategy
- allowing real interoperability demonstration and evaluation (WP1.2)
- showing different MSs / epSOS / Vendors liability
- showing alternatives for Sustainability / Maintainability / Scalability
- allowing flexible choices of every MS, according to eHealth political strategies, attitude and level of eP / PS service implementation
- exploiting Vendors experience / existing products
- fostering the creation of the European eHealth market

The stakeholders responsibilities, connected to each scenario, are summarised in the following table.

	X	Y	Z
Cost	• Higher for MS, • no shared costs	• 300-600 K€ • Lower for MS • Medium for epSOS	• 200-500 K€ • Lower for MS Lower for epSOS
Achievability	• Higher if close to epSOS, lower if not. • MS System integration	• In line epSOS plan Higher interoperability • Reuse of full NCP • Testing by epSOS	• Best for timing • Higher Interoperab. • Reuse of full NCP • Testing by epSOS
Risk	• MS complexity Testing for epSOS compliance for MS	• Risk shared by epSOS / vendors • Higher SW complexity • Development delay	• Development on Vendor • MS at NI Connector • Non fully compliant
Strategic fit Sustainability	• Consistent with MS infrastructure • High maintenance	• Challenging for Eu r&d maintenance condition Sustainability	• High Maintenance • Sustainability conditioned by post epSOS agreement
Impact on epSOS	• MS Liability • Better MS control • Increase maturity eHealth market	• MS use epSOS as SW supplier • Liability share by several actors	• MS classical supplier relation • Increase maturity eHealth market

The following evaluation criteria were introduced, to help PEB / PSB and the single MS to express a position with respect to the three scenarios.

- Cost
- Achievability (e.g. development effort required)
- Risk (development risk, implementation risk and management risk)
- Strategic fit and Sustainability
- Impact on epSOS members, e.g. MS, vendor, legal, procurement

X is assessed only to provide a reference to MSs. since no decision on X has to be taken by PEB /PSB.

The following table summarises the evaluation results.

Actors	MS	epSOS	Industry Team
Scenario **X**	• Assume leadership • Customise epSOS implementation to MS reality • Choose implementation strategy, either: – Procure full system – Select/ procure components and perform system integration – Partnership with vendors	• Consolidate specs and train MS • Perform careful interoperability testing at EU interface	• Provide Catalog • Specific vendors support specific MS
Scenario **Y**	• Develop integration between MS system and provided NCPs • Provide MS specific elements • Perform MS level integration testing	• Assume leadership, liability • Create experienced team • Choose best of breed components • Assign JARF to the team • Create NCP.A/B, Portal • Perform system and component level testing • Maintenance/sustainability	• Provide Catalog • Participate to selection process to be part of the team • IPR remains with vendors: "adaptation" = customisation
Scenario **Z**	• Develop integration between MS system and provided NCPs • Provide MS specific elements • Perform MS level integration testing	• Consolidate specs • Select vendor(s) • Assign JARF to vendor(s) • Perform a careful interoperability testing at EU / MS interfaces	• Vendor(s) assumes leadership, liability • Vendor(s) customise, deliver and maintain SW for NCP.A/B, Portal

More in detail, these are the consideration for every scenario.

Scenario X:

- Cost:
  - Higher because, solution/component customization for epSOS need to be covered by MS (100%) rather than JARF (50%).
  - Savings due to directly tailored solution
- Achievability:
  - Less complex for MS whose National systems close to epSOS
  - More complex for MS whose National systems are far from epSOS specs
  - MS needs to provide system integrators leadership and experience
- Risk:
  - MS understanding of level of complexity, on epSOS specs and need for specs customisation to MS needs
  - all risks assumed by MS
  - Testing tools must demonstrate interoperability of all NCPs
  - Procurement process may induce delay. Will it be compensated by faster execution ?
- Strategic fit and sustainability:
  - Drive effective clarification of epSOS interoperability specs
  - Consistent with National infrastructure
  - Single responsibility, Maintenance done by vendor partner

- Opportunity to compare interoperability solutions
- Impact on epSOS members:
  - MS: better control and simpler organization, more likely to require a procurement
  - Vendors: Increase maturity of eHealth Market
- Legal: liability is clear, not diffused in epSOS.

Scenario Y:

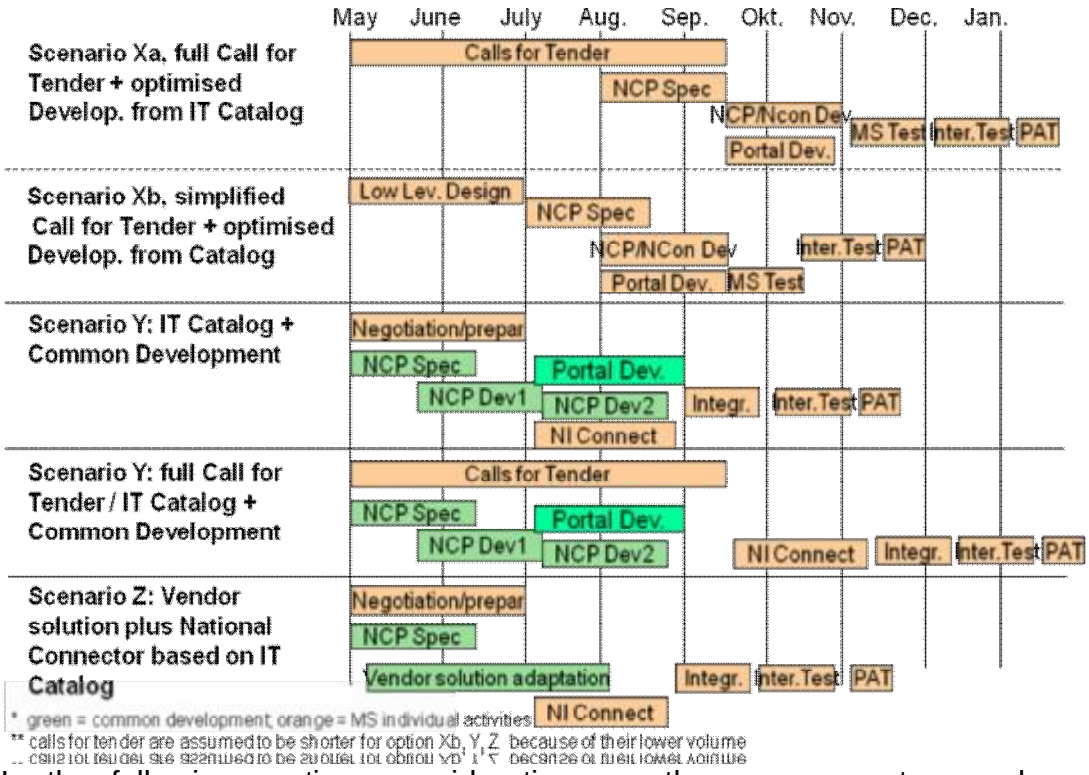
- Cost:
  - Likely to require 300-600K euros, depending on number of components reused from vendor/open source and level of epSOS customization
  - Shares costs for NCP, but induces higher costs for national "connection" as compared to X.
- Achievability:
  - Less MS driven effort (No need for full system procurement, development, integration)
  - MS needs less system integrators leadership and experience than X
  - In line with epSOS timing requirements
  - Higher probability to achieve interoperability in Piloting phases
  - Significant part of interoperability testing performed by epSOS development team.
- Risk:
  - Risks shared between n MS development teams and epSOS development team (N+1)
  - Reuse entire NCP, but distributed responsibility, time to fix issues.
  - epSOS faced with software liability
  - In some cases Increase overall SW size and debugging (encapsulation layers, connectors, workflow mgrs, etc;
  - Schedule slippage due to complex coordination among partners of development team
  - Delays or failures in developing common components will affect all MSs relying on them
- Strategic fit and sustainability:
  - European projects not suited for software maintenance
  - One size fits all may prove difficult to support all National infrastructures
  - Sustainability may be higher due to lower initial cost, but lower due to minor SW maintainability
- Impact on epSOS members:
  - MS: use a European entity as software supplier
  - Vendors: Slows down maturity of eHealth Market
- Legal: liability is unclear, diffused between vendors, epSOS and MS.

Scenario Z:

- Cost:
  - Likely to require 200-500K euros, depending on number of solutions selected from vendors and level of epSOS customization
  - Shares costs for NCP, but may induce additional cost for national "connection" than X.
- Achievability:
  - Less MS driven effort (No need for full system procurement, development, integration)
  - Increase overall SW size, but reduces debugging as existing solution reused.
  - MS needs less system integrators leadership and experience than X
  - In line with epSOS timing requirements
  - Higher probability to achieve interoperability in Piloting phases
  - Significant part of interoperability testing performed by vendor.
- Risk:
  - Risks shared between n MS development teams and epSOS development team (N+1)
  - Reuse entire NCP, but clearer vendor responsibility to fix issues.
  - No software liability for epSOS
  - MS Liability at National Connector and NCP Translation/Transcoding level
  - Divergence between what is implemented and epSOS interoperability specs
  - Compliance of Scenario Z with epSOS specs has to be verified before assigning JARF
- Strategic fit and sustainability:
  - Vendor suited for software maintenance
  - One size fits all may prove difficult to support all National infrastructures
  - Confuses issues related to epSOS interoperability specifications and poor implementation
- Impact on epSOS members:
  - MS: classical relationship with software supplier
  - Vendors: Increases maturity of eHealth Market

As far as the implementation timing for the various scenarios, a further variable is the fact that a public call for tender or a simplified procurement procedure are performed.

In the following drawing, a qualitative timing is shown. Only some possible cases are reported, just to provide a feeling of possible implications.



In the following section, considerations on the procurement procedure will be performed.

8.3 epSOS / MS procurement procedures and Call for Invitation

The Implementation of the NCP and or the development of the Components in common could be done by the Beneficiaries, or by Suppliers selected with an Open Call for Tender or a Negotiated Call for Tender.

Call for Invitation / Call for Tender should fulfill the rules stated by:

- DIRECTIVE 2004/18/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 31 March 2004 on the coordination of procedures for the award of public works contracts, public supply contracts and public service contracts

However, considering the nature of epSOS Pilot as a demonstrator of a R&D Project, not a standard service, the Procurement Departments might evaluate the applicability of the Article 31:

Cases justifying use of the negotiated procedure without publication of a contract notice:

- *Contracting authorities may award public contracts by a negotiated procedure without prior publication of a contract notice in the following cases:*

…

(2) for public supply contracts

(a) when the products involved are manufactured purely for the purpose of research, experimentation, study or development; this provision does not extend to quantity production to establish commercial viability or to recover research and development costs;

…

If Article 31 is applied, the possibility of performing Negotiated Procedures with Members of the epSOS Industry Team could be taken in consideration.

The respect of the transparency is assured by the fact that epSOS Contract was assigned after a Public Tender.

The existence of the Industry Team is publicly communicated on the Portal.

The existence of rules and procedures according to which ANY interested Entity can present an application to enter into the epSOS Industry Team is publicly available.

WP3.9 cannot give a closed indications to MSs, because each one has to apply its own procurement policies, to follow the procurement procedures regulated by National / Regional laws.

However, such rules and procedures have been applied to establish a Call for Invitation to epSOS Beneficiaries and to Members of the epSOS Industry Team to submit proposal for providing offers, compliant to epSOS D3.1.2 – D3.7.2 and HighLevel Design specifications for:

- Components to be used by MS to implement Scenario X
- Solutions to implement Scenarios Y
- Solutions implementing Scenario Z.

The following offers were received:

1. Fraunhofer Institut / ELGA / Tiani-SPIRIT Scenario Y proposal
2. **Tiani SPIRIT/CISCO** Scenario Z proposal
3. **Gnomon, CompuGroup, Carecom and Oracle consortium**, Scenario Z proposal
4. **Microsoft** Implementation platform
5. **Gnomon** Component proposal
6. **PosAm** Component proposal
7. **CareCom** Component proposal
8. **Apollo** Component proposal
9. **CompuGroup** Component proposal
10. **Oracle** Component proposal
11. **Steria** Component proposal

Proposal 1, 2, 3, offering scenario Y and Z solutions, were evaluated by an epSOS evaluation team, while all the proposals have been gathered in the Industry Team Component / Solution Catalogue (see 8.5)

The evaluated elements of the proposals:

1. Fraunhofer Institut / ELGA / Tiani-SPIRIT
2. Tiani SPIRIT/CISCO
3. Gnomon, CompuGroup, Carecom and Oracle consortium

were:

- Overall technical feasibility:
  - offers to include a commitment to all requirements in D3.1.x-3.7.x and HLDD.
  - offers to include reference to the HLDD in terms of components, interfaces between them and to the national infrastructure.
- The design spec for the solution is to be provided by the implementing consortium but checked and verified by epSOS for compliance to epSOS requirements (to be completed by WP3.9), epSOS will be assisting and consulting the writing of the design spec.
- Organisational aspects
- Cost and payments
- Planning
- Maintenance during the pilot phase
- Sustainability
- availability of the source code
- maintenance after the end of epSOS
- further development of the product

The PEB and the PSB, on the basis of the mentioned evaluation criteria results have decided to assign to Fraunhofer / ELGA / Tiani consortium the development of the NCP-in-a-Transparent-Box demonstrator implementation, with the strong indication to involve other Members of the Industry Team in the developments.

PosAm, CareCom, Gnomon, NetSmart were added to the Consortium. CISCO system was adopted as preferential infrastructure.

The Consortium is referred to as F.E.T.: Fraunhofer ELGA Team.

8.4 **Overall implementation planning**

All MS who are planning to pilot within epSOS, have expressed the will of adopting Scenario Y provided by F.E.T.

In the sequel, the best overall planning for the major stakeholders is provided.

It has to be continuously monitored and revised, according to the real implementation status. It is however reported to provide the general timing framework to achieve the goal of start piloting on January 1st, 2011.

F.E.T. Implementation Plan:

- JAVA API of NI 25.06.2010;
- revised version: 15.07.2010
- Functional specification, Interfaces: CC Architecture, 30.06.2010; revised: 15.07.2010
- Developers FET Workshop for 14-15.07.10
- CC Skeletons 15.07.2010
- Specifications/Documentation for D3.9.1 F.E.T. 21.07.2010
- CC Skeleton QA'ed 30.07.2010
- (need IHE simul,+test data CC Funct. Finals 10.08.2010
- installation guide NCP-in-a-Box + 30.09.2010

WP3.9 / WP3.10 Implementation Plan:

- release for comments Test Strategy V0.4 30.06.2010
- for developers F.E.T. Workshop 14-15.07.2010
- integration in Gazelle Test cases 31.07.2010
- invitation to Industry Team for PAT 1 Send formal 20.08.2010
- (XML, not CDA) for F.E.T. testing First test data 03.08.2010
- FET specifications D3.9.1 release with 26.08.2010
- release for testing MTC 0.x: First 28.08.2010
- releases: starting from July 15 to Testing tool 15.09.2010

- Test Tools D3.9.2 release with 15.10.2010
- Projectathon Test data for 15.10.2010
- Training Gazelle Lab Usage 15.09.2010
- /Transcoding: MTC 1.0 MVC Translation 15.10.2010
- Projectathon Usage Training Gazelle 30.10.2010

MS challenging planning:

- Connector design Start National 25.06.2010
- integration test with Skeleton Start partial 15.07.2010
- Connector development Complete National 30.09.2010
- participate to PAT 1 Communication to 11.10.2010
- translation transcoding MVC: MTC 1.0 Complete MVC 15.10.2010
- integration test Perform full system 31.10.2010
- preparation tests for Projectathon Complete 15.11.2010
- Projectathon 1 22.11.2010
- Deploy and test 31.12.2010
- Field Test) Operation (LSP 01.01.2011
- participate to PAT 2 Communication to 07.01.2011
- be confirmed) Projectathon 2 (to 11.04.2011

8.5 **Industry team Role: Component Catalogue**

Industry Team has represented a relevant contributor to all the Work-Packages.

In particular in WP3.9 Industry Team members have participated to the definition of the implementation and testing strategies.

As for the participation to the Call for Invitation, several proposal were submitted.

It was jointly decided that, although the epSOS is not performing any direct qualification on solutions and components proposed by Industry Team Members, it is of high value for the MS while deciding how to implement the National solutions.

Industry Team Members products will be organized in a Catalogue, kept updated by the Industry Team Manager.

Industry Team Members are warmly invited to participate to the Projectathon, to qualify their proposed solutions according to the epSOS specifications implementing IHE interoperability profiles.

9 Fraunhofer ISST / ELGA / Industry Reference Implementation

9.1.1 Baseline Assumption

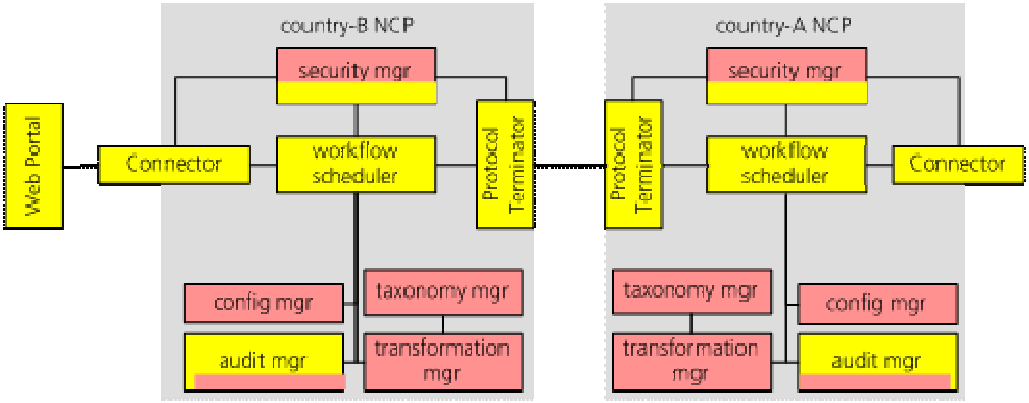
FHGISST, ELGA GmbH, and Tiani Spirit propose to use model Y as the envelope for the NCP setup and incorporate platforms from the Industry Team Catalogue and individual components developed on epSOS specification. The core principles of the FHGISST/ELGA/TIANI offer are:

- epSOS specific components are developed by epSOS beneficiaries and industry team (IT) members. Following model Y all of these components are provided as open source under an unrestrictive license.
- highly standardized components that only require minimum extensions to existing IHE X* and ATNA products SHOULD be provided off-the-shelf. Tiani Spirit will provide these components. Tiani Spirit will take responsibility for the integration of the commonly developed components with their product.

Nevertheless, MS are free to substitute the Tiani NCP „backbone“ by any other vendor’s product(s).

Coordination and quality assurance will be performed by the epSOS beneficiaries FHGISST and ELGA GmbH.

The Figure below shows the commonly developed components (pink) and the off-the-shelf „backbone“ provided by Tiani Spirit (yellow) (closed code).



9.2 **Separation of Responsibilities**

Fraunhofer ISST is leading sub-project I with a task assignment as follows:

- design of the interfaces for the common components of Configuration Manager, Transformation Manager, Taxonomy Service Access Manager, and Audit Manager
- subcontracting of the respective Industry partners
- coordination and Management of the development of the common components of Configuration Manager, Transformation Manager, Taxonomy Service Access Manager and Audit Manager. Industry partners that assume responsibility for any one of these components subcontract with FHGISST
- Quality Control of the source code and documentation of the common components of Policy Manager and TRC STS
- Quality Assurance of the backbone components and integration delivered by sub-project II (ELGA GmbH)
- Setup of the development environment and compilation of the coding / documentation guidelines
- Implementation of workshop I
- performing of the tasks assigned to the epSOS liason / alignment team
- Coordination with sub-project II

The sub-project II is coordinated and contracted by ELGA GmbH and contains the following tasks:

- design of the interfaces for the common components of Policy Manager and TRC STS.
- Management of the development of the common components of Policy Manager and TRC STS. Industry partners which take responsibilities for these components subcontract with ELGA GmbH
- Quality Assurance of the source code and documentation of the common components of Configuration Manager, Transformation Manager, Taxonomy Service Access Manager and Audit Manager.
- Quality Control and review of the adaptation work performed by Tiani Spirit for the NCP „backbone“ components.
- Quality Control and review of the components integration work performed by Tiani Spirit

- Implementation of workshop II
- Setup of a VM for the easy distribution of the full NCPs
- Coordination with sub-project I

9.3 Commonly Developed Components Activity Provision

This section sketches the functionality of the commonly developed components and the work plan for their implementation.

9.3.1 Consent Manager

This common component deals primarily with the creation and transmission of patient consents within epSOS for any connected client system, such as a national portal solution (country-b functionalities) or an existing integrated software solution at a point of care (PoC).

The consent manager is designed to act as a service provider for software components and software systems that have little or no knowledge on how an epSOS patient consent is created, compiled, and communicated. In order to provide this functionality, the consent manager featuring two sets of distinctive external interfaces:

- a D3.4.2-compliant epSOS-sided interface for NCP2NCP communication
- a WS-Interface for client connectivity and parameter exchange

The epSOS-sided interface is ensuring epSOS-compliance and a fully interoperable patient consent exchange between all participating Member States. The client-interface is a generic web-service interface that facilitates common client access for creating and revoking patient consents documents within epSOS: Due to the highly heterogeneous client capabilities within epSOS, this service relieves less complex clients, such as portals, from the burden of creating BPPC-compliant exchange documents and the subsequent wrapping of such documents into the D3.4.2-compliante communication and processing methods.

This component also facilitates the seamless integration of already existing client systems that are not fully capable of providing the required patient consents as of yet.

In summary, the consent manager is:

- accepting the required metadata and parameters of client system through a generic web-service interface
- creating and compiling a BPPC-compliant epSOS patient consent message, as specified in D3.4.2

- enriching the message by epSOS-specific routing, type, and addressing information
- preparing a BPPC-style message for transmission within the epSOS network
- providing D3.4.2-compliant service interfaces and methods (put/revoke) for the NCP2NCP communication

9.3.2 **Configuration Manager**

The common implementation of the epSOS Configuration Manager will provide the following functionalities to other NCP components:

- providing information on epSOS central services (centrally managed resources);
- management of an NCP's own configuration (addresses and certificates);
- management of trusted NCP's addresses and certificates.

The NCP-B routing manager will be implemented together with the configuration manager.

The epSOS Configuration Manager consists of 4 parts:

- **NCP Configuration Manager**: library for encapsulating access to the epSOS NCP configuration (path names, URLs, etc.).
- **Routing Manager**: library for runtime access of routing information.
- **NCP TSL Synchronizer**: batch tool for making local copies of all NCPs TSLs. Certificate data will be stored in a format that can easily be copied into gateway and web server configuration files. Routing information will be written to the NCP configuration.
- **TSL Editor**: GUI tool for editing the NCPs own TSL. Required functionalities include certificate import, editing of all elements, signature application and upload.

The following table lists the tasks that have to be performed for the implementation of the epSOS Configuration Manager.

Configuration Manager and Routing Manager	
Task	Description
Access to epSOS common configuration	providing access to information from epSOS common configuration (e.g. paths to schemas and taxonomy services)
Mgmt. of service WSEs and certificates	parsing and managing epSOS NCP config (acc D3.4.2)
configuration mgmt.	loading updates and managing releases
Configuration Editor	simple graphical editor for the management of an NCP's own configuration
security mgr. interaction	verification of signatures on configuration data
Routing Manager	NCP-B routing manager
Coordination	Coordination of Component development team
Testing	testing of the configuration mgr component
Documentation	incl. guideline for NCP configuration

9.3.3 **Audit Manager**

The epSOS audit manager consists of an audit repository and an audit writer. The audit repository holds audit trail data, while the audit writer is responsible for transforming activity information into the defined audit trail entry format. Acc. to IHE ATNA the audit writer will be part of the NCP „backbone“. The audit repository offers a BSD SYSLOG interface acc. to the IHE ATNA specification and will provide the following functionalities:

- Secure storing audit trail entries;
- adding audit trail entries to the repository;
- exporting audit trails;
- anonymizing audit trails for use regarding the epSOS reporting activities.

The epSOS Audit Manager consists of 5 parts:

- **Audit Writer**: library for populating audit trail entries
- **Syslog Client**: library to send the audit trail data to repository
- **Audit Repository**: persistent storage facility for audit entries
- **Anonymization**: stored procedure for anonymising the audit trail
- **Monitoring Manager**: GUI for analyzing the anonymised data

The following table lists the tasks that have to be performed for the implementation of the epSOS audit manager.

Audit Manager Task	Description
Audit Repository	design and implementation of an audit repository
Syslog interface	implementation of the operation for accepting, signing and storing audit trail entries
Export operation	implementation of audit data export according to D3.4.2
Anonymization	Anonymization of an audit trail
Advisory Console	Simple GUI to run preconfigured XSLT queries against an exported audit log
Coordination	Coordination of the component development team
Testing	testing of the audit manager component
Documentation	

9.3.4 **Transformation and Taxonomy Service Access Manager**

The NCP transformation manager and the taxonomy service access manager implement the NCP internal functionality of the epSOS semantic services. It is the task of these two components to transcode patient summary, ePrescription, and eDispensation documents from a national encoding to the epSOS pivot encoding and vice versa.

It must be noted that this transformation does not affect the structure of the transcoded documents. Schema mapping must be performed within the national infrastructures and is out of scope of common component development.

9.3.4.1 epSOS Transformation Manager

The **epSOS transformation manager** provides the transcoding/translation of controlled vocabulary elements within epSOS documents:

- ePrescription,
- eDispensation, and
- Patient Summary.

The epSOS Transformation Manager consists of 2 parts:

- **epSOS Transcoder**: Java library for the transcoding of documents that follow the epSOS schema (national to epSOS, epSOS to national)

The epSOS Transformation Manager is part of the epSOS semantic service. It includes all functionality that is related to rendering of the epSOS documents:

Component responsibilities:

- Translating and/or transcoding (if necessary) the original data compliant to epSOS CDA syntax from the national language and possibly from the national code system(s) in the document creator country (in most cases Country A) to the epSOS Reference Terminology. From a functional point of view the translation and transcoding are the same operation for the Transformation Manager from the point of view of the document creator country (in most cases Country A).

9.3.4.2epSOS Terminology Service Access Manager

The **epSOS terminology service access manager** encapsulates the local management and access to epSOS controlled vocabularies and mapping tables.

It is assumed that all epSOS terminologies/taxonomies will be centrally managed. MS are responsible for providing the mapping of the epSOS value sets onto their respective national encodings. Local excerpts of the central tables are kept within NCPs in order to provide efficient means for the mutual mapping of national codes and epSOS codes.

The common implementation of the Terminology Service Access Manager will provide the following functionalities to other NCP components:

- local management of a taxonomy excerpt
- single value mapping (both directions)
- value set download (using IHE SVS)
- verification of the authenticity of all taxonomies

The epSOS Terminology Service Access Manager consists of 4 parts:

- **Database Schema**: Schema for the local term mapping table.
- **ValueSet Retriever**: JAVA library for synchronizing a single value set by copying the centrally managed table into the local database
- **TSAM Synchronizer**: Batch job for synchronizing all value sets' local copies at once.

- **Taxonomy Mapping Service**: JAVA library for switching a single term

The following table lists the tasks that have to be performed for the implementation of the epSOS Terminology Service Access Manager.

Terminology Service Access Manager	
Task	Description
Framework Setup	setup of an open source DB for the local terminology repository
Taxonomy Repository	design and implementation of a local taxonomy repository
configuration mgmt.	loading updates and managing releases (based on IHE SVS)
security mgr. interaction	verification of terminology authenticity
Coordination	Component development team internal coordination
Testing	design of test cases and component testing
Documentation	

9.3.5 **Security Manager / TRC STS**

The epSOS security manager is responsible for the application of security mechanisms and their verification. It contains two security token services for issuing HCP identity assertions and TRC assertions as specified in D3.4.2. While the HCP identity assertion's STS is based on IHE XUA and therefore part of the NCP "backbone"; the STS for the TRC assertion is part of the common development. It will be provided as a WS Trust STS for issuing TRC assertions acc. to D3.4.2.

The epSOS Security Manager consists of 4 parts:

- **NCP Signature Manager**: operations for applying and verifying digital signatures.
- **NCP Certificate Validator**: operation for verifying the validity of a certificate or a key
- **TRC Assertion Issuer**: operation for issuing TRC assertions acc. D3.4.2.
- **TRC STS**: WS Trust RST/RSTR wrapper for the TRC assertion issuer.

The following table lists the tasks for the implementation of a WS Trust based TRC STS.

Security Manager	
Task	Description
Framework Setup	Setup of an epSOS security framework to integrate the open source libraries
TRC Provider	issuance of the epSOS TRC assertion
Testing	testing of the security mgr component
Documentation	

9.4 Closed-Source Components / epSOS Backbone

The reference implementation of the epSOS NCP provided by Fraunhofer and ELGA features a number of products as closed-source components. These components perform functions that are highly standardised and not epSOS-specific. The Industry partner that assumes the provision of the closed-source components is Tiani.

All closed-source components from Tiani Spirit are derived from a real-world product and subsequently comply with the required set of applicable international standards (such as IHE-X* or ATNA). Tiani Spirit will assume the responsibility for the appropriate integration of the commonly developed components with their product.

The closed-source components by Tiani Spirit will be provided as a set of components and functionalities that are:

- fully compliant to the applicable epSOS requirements,
- accessible at the specified interfaces of the epSOS specifications and HLDD,
- fully documented and provided with the required interfaces disclosed,
- and allow for potential Member State-specific substitutions.

9.5 Workshops

In order to facilitate the deployment and operation of the epSOS services, the CCD team implements a total of two workshops. The targeted audience are the experts appointed by each piloting Member State. The primary goal is to provide first-hands experience on the common components and the epSOS backbone, as well as the opportunity to observe the interaction within a reference NCP, the opportunity to speaking directly to the developers for potential questions, and a compilation of best practices, respectively lessons-learned, towards NCP integration and operation. All other epSOS beneficiaries are also welcome to participate.

Workshop I	
Company	Activities
ELGA	Planning and implementation of a workshop for the introduction and detailed explanation of the National Connector, enabling the interested Member States experts to gain first-hand experience on how to connect their National Infrastructure to the epSOS system.
Fraunhofer ISST	Provision of assistance with regards to the affected common components and their integration into the TIANI backbone.

Workshop II	
Company	Activities
Fraunhofer ISST	Planning and implementation of a workshop for the deployment and operation of the NCP and its full components, enabling the interested Member States experts to gain first-hand experience on the operation of an epSOS NCP.
ELGA	Provision of assistance with regards to integration of the TIANI backbone.

9.6 **epSOS Liaison / epSOS Alignment**

In order to secure the epSOS specification and the subsequently full alignment of the activities performed by the common components development team, a task force pursuing that goal has been established.

This task force will jointly:

- safeguard the full compliance of the common components and the backbone to the current epSOS specifications and their requirements

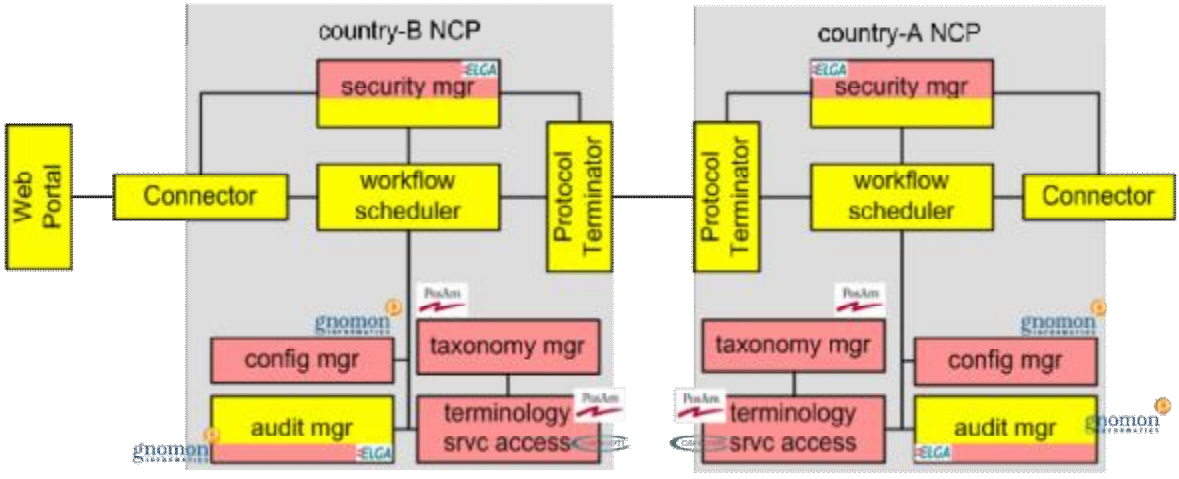
- assert the compliance of every common component
- identify and address potential gaps in the epSOS specifications and their resolution in the common component development
- document major decisions taken during the development process
- propose and compile the contents of low-level design document (LLDD)
- coordinate and adjust the development and testing schedule
- facilitate communication between the epSOS and development teams
- appoint and make available single points of contact (SPOC) regarding the development process

9.7 **Change Proposal and HLDD-Delta in CCD**

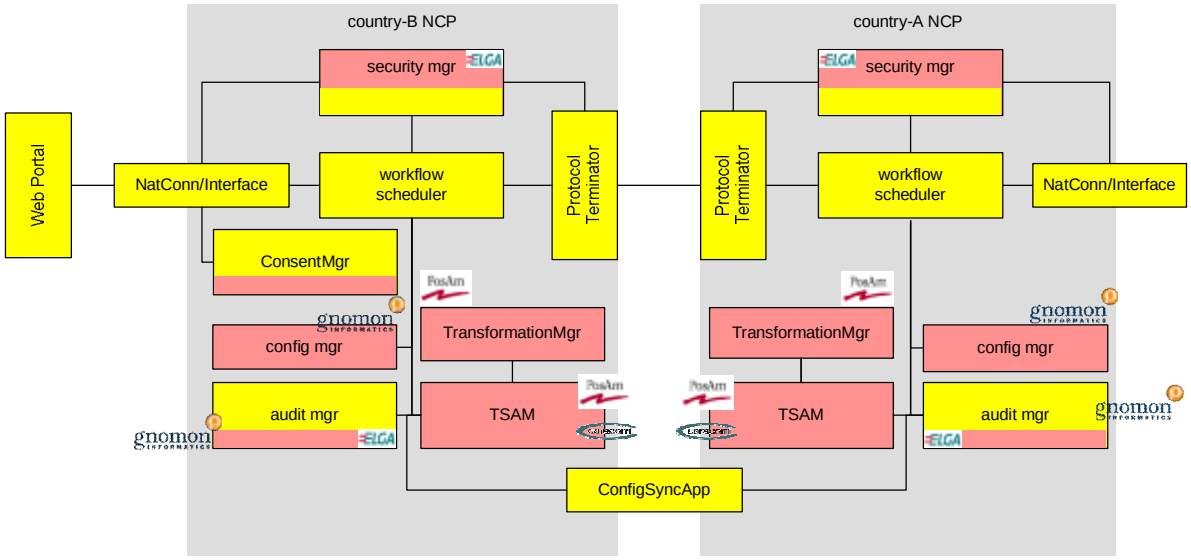
During the design and specification phase of epSOS, most of the work packages in the PD3 domain derived and assessed rather individual non-functional requirements specific to their assigned application domain. Although the effort performed by TPM and WPs to identify and fix inconsistencies, gaps arose from the chosen setup.

While those gaps are particularly difficult to spot in a specification environment, they are highlighted in an implementation situation. As already anticipated, WP 3.9 and the CCD team identified various technology gaps and inconsistencies during the implementation process.

The original NCP design as outlined in the HLDD is:



The NCP-in-a-Box as implemented by the CCD team is scoped as such:



Most of the differences are the consequence of the agreed change proposal and the features of the TIANI-SPIRIT Back-bone Close Source.

The list of the change proposals and the description of the major differences between HLDD and the implemented Common Components is provided in Appendix A.

The procedure set up to handle foresees that a potential change proposal is to be documented by the beneficiary, task force, or working group that is requesting the change. Change proposals MUST feature:

- a clear problem statement justifying the change
- an impact analysis identifying affected operations, transaction, and work flows
- the anticipated impact on other epSOS services and/or components
- the proposed solution clearly stating the implementation effort

Accepted and quality-assured change proposals are gradually uploaded onto ProjectPlace and provided to all epSOS beneficiaries in the folder:

“epSOS Common Components Development -> Change Proposal”

https://service.projectplace.com/pp/pp.cgi/0/518757384

10 Conclusions; lessons learned

WP3.9 Proof of Concept development, has represented the link between the functional and technical specification performed in WP3.1 – 3.7 and the implementation at epSOS and MS level.

The activities carried out according to the three described workstreams (proof of concept implementation, semantic interoperability, testing strategy and test tool development) have allowed building the expected bridge allowing:

- Industry Team to provide the expected contribution to epSOS implementation
- WP2.1 to fine-tune legal requirements on implementation liability and accountability issues
- WP3.8 to finalise the Guidelines to implement epSOS interoperability
- WP3.10 to develop the test sequences and perform the testing
- WP4.2 to define the implementation planning at epSOS and MS level
- More recently, WP4.3 to start the MS level implementation and deployment.

The impact and the demand for cross-border semantic interoperability is once more demonstrated by the very high number of members of the Semantic Working Group, gathering about 140 members, and the relationship with international interoperability standardisation bodies like HL7 International, IHE International and Coding System Standardisation bodies like WHO, IHTSDO, EDQM, UCUM, ILO.

10.1 Open Issues

The current version of D3.9.1 leaves open some issues, some related to not fully compliant FET implementations to epSOS specifications, other related to the partial lack of knowledge about the possibility of MS to derive information from their eHealth systems to build epSOS CDA.

The exhaustive list of issues is included in D3.9.1 Appendix A, for those related to F.E.T. implementation and in D3.9.1 Appendix B1, for those related to Semantic Interoperability implementation.

In the sequel a partial list of the still open issues is provided.

- **IHE XCA adaptation not ready in time**

D3.4.2 foresaw the XCA profile required a single transaction. An adaptation of the IHE profile was agreed. However, such adaptation (described in D3.4.2 version 1.0 in section 3.1.2) will not be ready before mid-2011.

Complete information on this issue was provided to MS for their National Connector implementation, and to IHE Europe, to be sure the related test tool is developed in line with IHE XCA or epSOS XCA Profile.

- **MS capability to extract all info for the generating an unambiguous eP**

Appendix B1 CDA Implementation Guide provides all the data structures needed to describes any kind of medications. However it is not known if the data contained either in the prescriptions or in the MS database allow to univocally describe medications to allow unambiguous dispensation.

MSs piloting eP have created an expert team to study available data and reach an agreement on how to exchange needed information through the defined eP CDA.

The field return during the LSP testing, performed in line with D3.9.2 testing strategy will provide a large amount of data to fine tune the specifications and implementations.

- **MS capability to generate unambiguous transcoding in the MTC**

The process of identifying and defining the ValueSets started by WP3.5 and completed by WP3.9 may have left codes that represents ambiguities for transcoding. A typical example is the relation many to many between ICD-9 4 character and ICD-10 3 character. Other similar issues may derive from coding systems defined by MS.

As for eP, a careful analysis of such issues will be followed first by an agreement among MS piloting the service using such codes, then by a careful LSP testing and data collection.

- **Presentation format of Documents in Country B**

WP 3.1 eP functional specification and WP3.2 PS functional specification have not provided stringent requirement on how documents have to be displayed, apart the request to avoid cumbersome presentations.

The CDA Implementation Guide gives generic indication on the use of stylesheet. The strong recommendation is that, for Patient safety reasons, the HCP should be able to select the language in which the document is displayed. This implies that the CDA display tools should be "*epSOS aware*", and able to interpret all translations of coded elements.

However, no epSOS style-sheet was specified, nor rigid specs on the text elements structure were provided.

More knowledge on the CDA formatting from every MS will be obtained with the development of Representative Test Data (activity from WP3.10).

An expert team was created to study the document presentation, taking into account usability for HCP and Patient Safety. The careful analysis of the

Representative Test Data will provide relevant info to the expert group to possibly introduce an epSOS style-sheet or further CDA spefications related to preferential formatting information.

Appendix A FET Solution

See Document D3.9.1: Appendix A FET Solution

Appendix B Semantics

B1 epSOS Semantic Implementation Guidelines

See Document: D3.9.1: Appendix B1: epSOS Semantic Implementation Guidelines

B2 MVC / MTC

See Document: D3.9.1: Appendix B2: MVC / MTC

B3 epSOS Central Reference Terminology Server (eCRTS)

See Document: D3.9.1: Appendix B3: B3 epSOS Central Reference Terminology Server (eCRTS)