www.eu-spocs.eu

COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME

ICT Policy Support Programme (ICT PSP)

Preparing the implementation of the Services Directive

ICT PSP call identifier: ICT PSP-2008-2
ICT PSP main Theme identifier: CIP-ICT-PSP.2008.1.1

Project acronym: SPOCS

Project full title: Simple Procedures Online for Cross-border Services
Grant agreement no.: 238935

Specifications for interoperable access to eDelivery and eSafe systems

Functional Specification, Architecture and Trust Model

- Corrigenda 2.0 March 2012 based on Corrigenda 1.2 -

Deliverable Id :	D3.2
Deliverable Name :	Specifications for interoperable access to eDelivery and eSafe systems
Status :	Corrigenda 2.0
Dissemination Level :	SPOCS internal and EU Commission
Due date of deliverable :	30th September 2010
Actual submission date :	
Work Package :	WP3: Interoperable delivery, eSafe, secure and interoperable exchanges and acknowledgement of receipt
Organisation name of lead contractor for this deliverable :	**BVA**
Author(s):	Jörg Apitzsch (DE FHB), Olaf Rohstock (DE FHB), Lars Thölken (DE FHB), Luca Boldrin (InfoCert), Bernd Martin (DE Siemens), Stefanie Rieger (DE BVA), Michael Seeger (DE Siemens), Arne Tauber (SPOCS AT), Peter Worofka (DE Siemens)
Partner(s) contributing :	SPOCS.AT, GR MINT, IT InfoCamere, IT InfoCert, NL MINEZ, PL ILIM

Abstract: This is the main document of the second deliverable in work package 3 of the EU co-funded project SPOCS. It describes functional specifications, architecture and trust model for the interoperability layer to connect existing eDelivery and eSafe solutions based on a common eSecurity architecture. Based on the specifications open modules will be developed.

Revised version 1.1 and 1.2 respected perceptions of the implementation phase and of additional assessments provided by MS that joined SPOCS during the enlargement process.

This revision 2.0 is a formal adjustment of the sections related to eDelivery, respecting the fact that major concept details meanwhile are adopted by the latest version of the specification ETSI TS 102 640 on “Registered E-Mail”.

History

Version	*Date*	*Modification reason*	*Modified by*
0.1	19.04.10	Initial draft and deliverable structure	Jörg Apitzsch
0.02 - 0.8	21.04.10 - 26.05.10	Ongoing content production; version 0.8 ("General Architecture and Trust Model") was published for commenting through interested public	All authors and contributors
0.81 - 0.89	09.07.10 - 25.08.10	Chap. 2.1 "Trust Model" detailed Ongoing work	0.81
0.9	31.8.10	Complete detailing and re-editing for version to be provided to QA version.	Jörg Apitzsch Bernd Martin Stefanie Rieger Olaf Rohstock Michael Seeger Peter Worofka
0.91	09.09.10	Comments PTC added	Olaf Rohstock
092	14.09.10	Comments and rephrasing	Hardy Kloempges Michael Seeger Bernd Martin Arne Tauber Christian Schmitt Olaf Rohstock
0.93	15.09.10	Final Layouting	Lars Thölken
1.0	15.09.10	Versioning	Stefanie Rieger
1.0.x	16.09.10 - 09.02.11	Minor changes regarding improvements while creating the open modules	All authors and contributors
1.0 D3.3Rev	10.02.11	Versioning	Stefanie Rieger
1.1	10.03.11	Corrigenda according findings during implementation phase; Minor editing issues	Jörg Apitzsch
1.2	06.07.11	Further slight corrections according findings during implementation and project enlargement phase	Jörg Apitzsch Luca Boldrin Peter Worofka
2.0	26.03.12	eDelivery adjustments with respect to latest revision of ETSI TS 102 640	Jörg Apitzsch

Table of contents

List of figures

List of tables

List of abbreviations

Abbreviation	Explanation
CA	Competent Authority
DNS	Domain Name System
DTP	Document Transfer Package
EC	European Commission
eID	Electronic Identity
GW	(eDelivery) Gateway
IdP	Identity Provider
LSP	Large Scale Pilot
MD	(eDelivery) Management Domain
MS	Member State
OCD	Omnifarious Container for eDocuments
OCF	OEBPS Container Format
PEPPOL	Pan-European Public Procurement Online
PSC	Point of Single Contact
PTC	Technical Coordinator (of the SPOCS LSP, in this case)
REM	Registered E-Mail
SD	Services Directive
SP	Service Provider
SPOCS	Simple Procedures Online for Cross-border Services
SSL	Secure Socket Layer
STORK	Secure Identity Across Borders Linked
TLS	Transport Layer Security
TSL	Trust-service Status List
TSP	Trusted Service Provider
UDDI	Universal Description, Discovery and Integration
WP<n>	Work Package (number n)
WSDL	Web Service Description Language

Further abbreviations used in this document are explained on first occurrence.

Document structure of SPOCS D3.2

SPOCS deliverable D3.2 "Specifications for interoperable access to eDelivery and eSafe systems" consists of several documents.

The main part gives a complete description of the general context, functionality of solutions provided, their architectural details and covered security and trust establishment features.

Additional documents are provided for detailed technical specifications of the buildings blocks, considered security architecture modelling and development baselines and according operational policies.

Following documents are provided for SPOCS D3.2:

The main document:

- SPOCS D3.2 Functional Specification, Architecture and Trust Model

is accomplished by following appendix documents, provided as separate ones to improve readability of the whole deliverable:

- Appendix 1: Security Architecture Development Process
- Appendix 2: Trust-service Status List Profiling ("SPOCS TSL")
- Appendix 3: eDelivery Interconnect Protocol and Gateway Specification
- Appendix 4: eSafe – Operations in Detail
- Appendix 5: SPOCS TSL Accreditation and Operation Policy
- Appendix 6: Security Model.

Note: Together with the corrigendas 1.1 and 1.2 of the D3.2 main document, revisions where made for the appendix documents 2, 3 and 4. The current corrigenda 2.0 is accomplished by major revsions of the Appendix 3 document.

Used abbreviations, general document conventions as well as referenced XML namespaces are outlined in this main document, while the overall reference list is contained in every separate appendix document, too.

Structure of this document

In the Introduction (section 1), the present main document of D3.2 initially gives an overview on scope of objectives, relations to project-external and -internal activities, requirements and challenges to be considered. This is followed by a description of the D3.2 approach basics, methodology and SPOCS user scenarios relevant for WP3.

Section 2 describes the overall topology and solution architecture specified with this deliverable. Besides central aspects as security architecture and a model for trust establishment between the different systems to be interconnected, the basic concepts for eDelivery and eSafe are detailed with regard to identified systems entities and actors.

The general architecture is broken down into details for eDelivery and eSafe in the following sections 3 and 4, followed by concluding remarks in section 5.

For all building blocks to be implemented, additional detailed technical specifications and operational guidelines are given in the separate annex documents mentioned above.

Project management aspects are outlined in appendices bound to the present document, the same applies for a graphical representation of the processes identified for SPOCS.

Document conventions

Notation

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

This specification uses the following syntax to define normative outlines for messages:

- The syntax appears as an XML instance, but values in italics indicate data types instead of values.
- Characters are appended to elements and attributes to indicate cardinality:
  - "?" (0 or 1)
  - "*" (0 or more)
  - "+" (1 or more)
- The character "|" is used to indicate a choice between alternatives.
- The characters "(" and ")" are used to indicate that contained items are to be treated as a group with respect to cardinality or choice.
- An ellipsis (i.e. "...“) indicates a point of extensibility that allows other child or attributes content specified in this document. Additional children elements and/or attributes MAY be added at the indicated extension points but they MUST NOT contradict the semantics of the parent and/or owner, respectively. If an extension is not recognized it SHOULD be ignored.
- XML namespace prefixes are used to indicate the namespace of the element being defined.

Elements and attributes defined by this specification are referred to in the text of this document using XPath expressions. Extensibility points are referred to using an extended version of this syntax:

- An element extensibility point is referred to using {any} in place of the element name. This indicates that any element name can be used, from any namespace other than the namespaces referred to in the table below.
- An attribute extensibility point is referred to using `@{any}` in place of the attribute name. This indicates that any attribute name from any namespace can be used.

The terms “header" and “body" used in this document are used as abbreviation of “SOAP header" respective “SOAP body".

Referenced XML Namespaces

Prefix	XML Namespace	Reference
ds	http://www.w3.org/2000/09/xmldsig#	[3]
esaf	*Preliminary:* http://uri.eu-spocs.eu/esafe/v1	This specification
ids	http://busdox.org/transport/identifiers/1.0/	[11]
ocdm	http://www.eu-spocs.eu/ns/ocdmetadata	[49]
rem	http://uri.etsi.org/02640/v2#	[12]
remsoap	http://uri.etsi.org/02640/soapbinding/v1#	[46]
s12	http://www.w3.org/2003/05/soap-envelope	[14]
saml2	urn:oasis:names:tc:SAML:2.0:assertion	[15]
stsl	*Preliminary:* http://uri.eu-spocs.eu/tsl/v1#	This specification
tsl	http://uri.etsi.org/02231/v2#	[9]
wsa	http://www.w3.org/2005/08/addressing	[5]
wsse	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd	[6]
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd	[6]
xades	http://uri.etsi.org/01903/v1.3.2#	[27]
xenc	http://www.w3.org/2001/04/xmlenc#	[19]
xs	http://www.w3.org/2001/XMLSchema	[20]

Table 1: Referenced Namespaces

Executive summary

The Large Scale Pilot Simple Procedures Online for Cross-Border Services (SPOCS) deals with enhancing cross border communication between Service Providers (SP) and Points of Single Contact (PSC) by electronic means, to support the implementation of Article 8 of the Services Directive.

D3.2 provides basic technical specifications for functionality and interfaces of open software modules to (i) enable cross-border interoperability of different eDelivery and eSafe systems and (ii) to implement a secure interoperable connection for data and document exchange between the heterogeneous infrastructures of different countries; meeting the functional and security requirements of each solution in a fully interconnected way.

eDelivery focuses on the path of data transmission between two communication partners. Those can be human endpoints with their client computers or protected services like eSafes, eSafes are systems designed to archive, manage and share data as a source for “authenticated documents”. Both systems are required to fulfil maximum security requirements within its operation but also for the communication with others.

The proposed solution incorporates existing national infrastructures by combining their positive aspects, gaining benefits of results in similar fields/topics of sibling LSP projects such as STORK (Pan European Proxy Services - PEPS) and PEPPOL (Business Document Exchange Network- BusDox and e-Signature Validation Infrastructure).

When defining the technical specifications, a strong motivation was to support acceptance beyond the project borders and to ensure long term sustainability. In order to propose an integrated solution, suitable for SPOCS business cases, highly accepted and state-of-the-art standards have been incorporated wherever needed. These are specifications provided by ETSI (Registered E-Mail – REM, Trust-service Status List - TSL), as well as several others from W3C, IETF, WS-I, and OASIS.

Besides requirements for security, reliability and traceability, the following core constraints have been captured in the technical specifications described in D3.2:

- Avoid the need for additional user registration (e.g. as already given by the MS solutions; centralised directories may even be forbidden by national law and regulations).
- Usage of e-addresses in a common, intuitive format must be possible (e.g. known from e-mail).
- Existing solutions have the general purpose of exchanging or storing any kind of document and information, not bound to specific business scenarios, and as such are payload-agnostic, as it should be with SPOCS eDelivery.
- Development of a consistent security model according to a security architecture development approach, meeting the business requirements, architecture design principles and architecture constructs, with respect to the characteristics and security aspects of a service oriented architecture.
- Support the process of attaching documents to an application in a simple and comfortable but highly secure and trusted way for existing and future systems.
- Establishment of trust between interconnected solutions must be transparent and not affect the trust model in the MS. Trust establishment should be achieved by similar, currently emerging concepts, extending the term "trusted service" beyond issuing eIDs or qualified signature certificates.
- The solution must provide maximum ease in connecting to the MS solutions in a

standardised, controllable way.

- As a result, the following key solutions have been defined for every key aspect of the specifications contained in this deliverable:

eDelivery

As the basic concept for the eDelivery infrastructure, the gateway approach was chosen, where different solutions exchange messages via specific gateways, with protocol transformation functionality as one main service. The different gateways can uniformly communicate to each other using one common protocol.

Cross-solution addressing

As well known to the human actors and manageable by most eDelivery user agents, as default address format the email address is chosen. With each national solution assessed in D3.1 a commitment was made regarding the ability to use the format directly or – e.g. for SOAP-based solutions – the ability to map such e-addresses to the domestic address scheme in use. A "normalised" message format is defined, enabling the mapping of domestic message format to the SPOCS interconnect one and vice versa. To be extensible, other address schemes are foreseen – thus, it is possible to deal even with the format as defined for BusDox ("Participant Identifier").

Delivery traceability and provability

For transport evidences and receipts SPOCS incorporates the ETSI REM specification TS 102 640 (REM evidences). These are used on the interoperability layer and may be – like the whole normalized message - converted by gateways to the corresponding domestic receipt- or notification formats.

eSafe

For the cross border integration of eSafes into SPOCS scenarios a protocol was specified, based on well established standard protocols as HTTP or Web services and which was inspired on the proven eps e-payment standard [44] for integrating payment activities in foreign portals. The whole communication is secured by using SSL/TLS.

Document Transfer Payload

As for the document transfer payload the OCD container (WP2) is used, the integrity and authenticity is guaranteed by making use of digital signatures.

eSafe Interoperability

Two principles of eSafe integration are identified and specified in detail: PUSH and PULL method. In the first case, a SP delivers via his eSafe electronic documents to a PSC. With the PULL principle, the PSC in turn is able to retrieve these e-documents while it has the appropriated access information, an access token issued by the eSafe. The specification is open to be extended for multiple transport mechanisms, such as the eDelivery module.

Trust Model

Given the assumption that trust within each realm (i.e. national infrastructure) is granted by specific policies in the realm itself, one pillar of the establishment of end-to-end trust within SPOCS infrastructure is a gateway-to-gateway trust model, based on Trust-Service status List (TSL), as defined by ETSI TS 102 231 [9]. TSL are therefore used to establish trust in the communication between eDelivery gateways, PSCs and eSafes.

SPOCS interconnect protocol

For the interoperability layer, the Web services protocol stack has been chosen as the base technology set. For serving the particular SPOCS requirements appropriate protocols or protocol extensions have been designed, all based on SOAP, WS-

Addressing, WS-Security and underlying mechanisms for message security, SAML token [15][17] profiled according STORK for authentication, HTTP, SSL/TLS and other proven technologies.

In course of the SPOCS project, this protocol was standardized by ETSI as part of TS 102 640 Registred Electronic Mail, sub-part “REM-MD SOAP Binding Profile”[46].

Security mechanisms

To provide secure communication between the eDelivery gateways that constitute parts of the SPOCS interoperability framework WS-Security with underlying TLS signature and encryption as well as a specific mutual authentication mechanism will be used. Message signing will be performed using X509v3 certificates and the accompanying Web Services Security X.509 Certificate Token Profile 1.1.

The next activities of WP3 will focus on implementing and, if necessary, further detailing the specifications in D3.2. Furthermore, there will also be a focus on the integration and set-up of the open modules for eDelivery (SPOCS Interconnect Protocol) and eSafe (open modules for PSCs and eSafes) considering the security architecture results of this deliverable.

1 Introduction

1.1 Scope and objectives

This deliverable D3.2 describes, based on the results of D3.1, how to build the solution architecture and the respective open modules. These descriptions will provide a clear definition of the general structure of the open modules, their necessary functionality and interfaces. Based on this foundation, the technical instructions describe unambiguously how the implementation of open modules should be carried out.

Following these rules it is guaranteed that although open modules will be built from different organisations; they always offer the same functionalities and interfaces.

The general objective of these specifications is to achieve and describe, based on a common security architecture, the cross-border eDelivery approach and an eSafe concept for storing and sharing personal electronic data and documents.

In the case of “eDelivery”, the objective of the specifications is a precise description of how to develop open modules, which provide a secure interoperable connection for data exchange between the infrastructures of different countries (which support the Service Directive measures).

Regarding “eSafe”, the objective is to specify how to integrate secure storages of / access to archived documents for authorised parties.

As SPOCS has only the mandate to define solutions regarding technical interoperability issues, it has to be acknowledged that the defined specification/modules are to be seen under the condition of legal and organisational approval from the EU Member States.

To reach these objectives, the deliverable D3.2 in Part 1 focuses on the ideas of the construction (General Architecture) and secondly, based on this, in the annex documents defines precisely in a detailed description (clearly and unambiguously) the technical specifications. Additionally the sections cover the necessary security architecture and security mechanisms.Since

1.2 Relations to internal SPOCS environment

WP3 Deliverable 3.2 has strong dependencies with the results of the WP 1, 2, 4, and 5 following the “SPOCS Big Picture”:

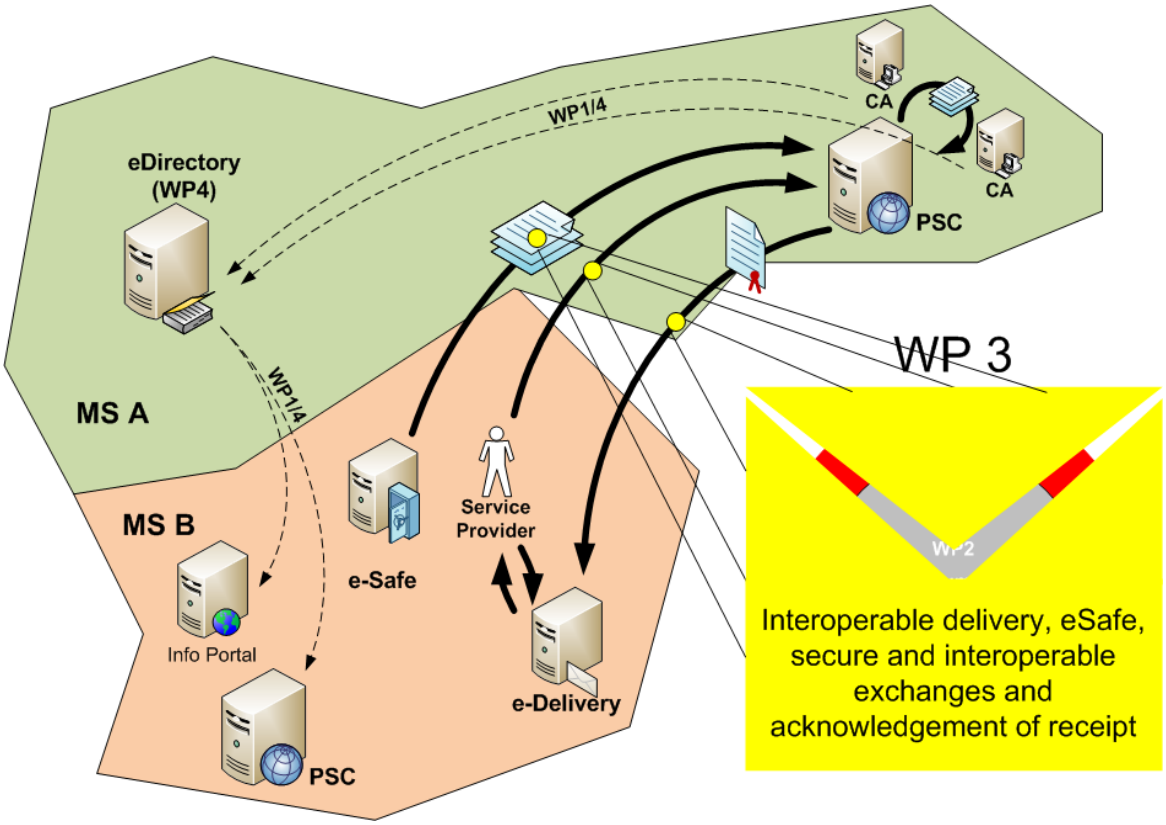

Figure 1 "Big Picture" focussed on WP3

Following the “SPOCS Big Picture”, the dependencies to the other work packages can be defined like this:

WP1 Contents Syndication, Multilingual Issues and Glossary

Contents Syndication requires distributed data access/exchange; an API is expected for the eDelivery interoperability framework, functionally fitting for these specific requirements.

WP2 eDocuments

WP2 will rely on the secure eDelivery interoperability framework to be provided by WP3; eSafe functionality for secure storage of and access to eDocuments is required in addition. Furthermore, eDocuments container from WP2 will be used as the payload data between the transfer fom eSafe to the PSC.

WP4 Interoperable eService Directories

WP4 will rely on the trust establishment mechanisms provided by WP3 ("SPOCS TSL") for delivering eDirectory contents to the identified recipient and secured access to eSafe instances.

WP5 Experimentation

Besides the WP3 infrastructures components needed by all WPs, WP5 expects to be provided with tests cases and support by WP3 for experimentation and validation.

Furthermore, there is a strong relationship between WP3 and all other SPOCS WPs in order to be able to provide a consistent security architecture for the SPOCS environment.

Although the core of security related activities is within WP3, security is a cross functional layer and is also relevant at least for WP2 and WP4.

The following security related aspects play an important role in developing a comprehensive security architecture:

- Integration with the business functionality for the SPOCS work packages 2-4
- Specification of the security architecture as a concern that may cut across the SPOCS work packages 2-4
- Realisation of the SPOCS security architecture covering WPs 2-4
- Evaluation of the security properties and interoperability of the SPOCS security architecture
- Formal design validation of key components of the security architecture. This task will either give formal approval of the validated components or will report specification errors back to the architecture/design group

This list does not claim to be complete. It should rather highlight the most important paths along which WP3 needs to be transferred or exchanged. Therefore, from a technical perspective, eDelivery and eSafe can be considered as being the "backbone" of other components or applications used in SPOCS.

These internal relations have been taken into account by choosing an appropriate methodology of work. In particular, all findings represented by this deliverable have been coordinated with all work package leaders

With perspective to the next Deliverable D3.3., the defined specifications will provide the solutions for interoperability of national approaches. D3.3 will implement the defined and required modules and/or gateway functionalities, which are still under conditional acceptance depending on the piloting scenarios.

1.3 Relations to external SPOCS environment

The WP3 concerns eSecurity and interoperable connectivity of eDelivery and eSafe services have been addressed by many efforts conducted in the last decade by IT industry, standardisation bodies and public institutions, especially for e-government and e-business.

First of all, an interoperability framework aiming to interconnect distributed different solutions in choice for WP3 must build on the relevant specifications and profilings of W3C, IETF, OASIS and the WS-I organisation. Thus, focus for the concept presented here is given on these standards.

On European level, relevant ETSI specifications apply; here, awareness must be taken to refinement respective reshaping of e-signature standards started in 2009. Efforts conducted by Member states and the EC for interoperability and convergence of national solutions are considered, fitting concepts or solutions are adopted. WP3 has strong awareness to

- Activities conducted under the auspices of the IDABC, especially recommendations, guidelines and architectural frameworks:
  - Studies on eID[1] , e-signature and (federated) validation services, update on MS country profiles assessed in 2009[2]

[1] For eID, see http://ec.europa.eu/information_society/newsroom/cf/itemdetail.cfm?item_id=4147 (last visited on Sep. 11, 2010)

  - European Interoperable Infrastructure Services (EIIS), Study on potential reuse of service modules and components (part of the European Interoperability Framework - EIF[3])
  - Study on electronic documents and electronic delivery for the purpose of the implementation of Art. 8 of the Services Directive[4]

- Handbook on the implementation of the Service Directive[5] , published by Directorate-General for Internal Market and Services
- Outcomes of sister Large Scale Pilot projects
  - PEPPOL: WP1 (EU-wide e-signature/e-Id validation – SPOCS will use these results 1:1[6]); WP 8 (BusDox specification and implementation for transport and directory infrastructure [11] – a matter of running alignments)
  - STORK: Cross-border authentication in general, interoperable usage of eIDs; WP 6.4 (eDelivery).
- Current ETSI standardisation activities on eDelivery
  - ETSI has established a Specialists Task Force (STF 402) working on "REM Interchange: e-mail Interchange between Registered E-Mail (REM) systems based on different transmission protocols".

As the concepts and solutions required from SPOCS WP3 must be indifferent to the concrete business scenarios served (content agnostic, at least), a generic approach has been chosen, feasible for e-government scenarios with comparable requirements. With the goal to obtain a broader, generic and broadly accepted approach towards an European eDelivery/e-Transport interoperability framework, ongoing collaboration with the mentioned Large Scale Pilot projects and ETSI STF 402 was established. Mid 2011, the work done by this STF was concluded and approved as new version of the ETSI TS 640 102 on “Registered Electronic Mail”, now covering also the concepts brought in by SPOCS WP3.

1.4 Methodology

To identify the requirements for all SPOCS work packages (WP), the entire application process as addressed by the Service Directive (SD) was firstly documented with a business process and relevant needed business activities. The agreed process was the basis for further atomic high level business use cases.[7] The main focus was on added

---

[2] For e-signature, see http://ec.europa.eu/idabc/en/document/6485/5938 (last visited on Sep. 11, 2010)

[3] Details see http://ec.europa.eu/idabc/en/document/7795 (last visited on Sep. 11, 2010)

[4] See http://ec.europa.eu/idabc/en/document/7667/5644 (last visited on Sep. 11, 2010)

[5] http://ec.europa.eu/internal_market/services/services-dir/proposal_en.htm (last visited on Sep. 11, 2010)

[6] relevant deliverables see http://www.peppol.eu/deliverables/wp-1 (last visited on Sep. 11, 2010)

[7] See E Appendix – Process Model for a visualisation of the SPOCS process model as a whole. Which WPs have to be concerned about which each respective process is represented by numbers attached to the processes.

value that would be provided by SPOCS, especially for the main application process beneficiary, the SP but secondly also to the Member States.

During a decision process embracing all WPs the process steps where classified in the categories

- Scope of the national implementations of the SD (as such, assumed to be operational already)
- Scope of SPOCS, supporting and benefitting the application process.

Depending on functionalities offered in the different Point of Single Contact (PSC) portals, possible variants of sub-process combinations and sequences where identified. SPOCS relevant sub-processes were further rated *mandatory* for all variants and *required* or *optional* for those that only provide additional comfort for one or more of the stakeholders, but are not fully necessary for the application process as a whole.

For all sub-processes and activities, relevant actors, targets and tasks as well as prerequisites and dependencies have been identified. In a last step, relevant activities and sub-processes have been assembled for WP3. Following that, WP3 relevant national eDelivery and eSafe solutions as well as the according security architectures of the SPOCS partner Member States have been rated according their functionalities. Thereby the focus was on scope and reusability of current concepts as well as integration and interoperability. The inputs were mostly the result of the previous assessment deliverable D3.1 [7] but also the requirements documents relevant for WP3 [8].

1.5 Challenges identified on base of requirements assessment

Based on the results of the requirement assessment, the future reference architecture should specifically address the following goals:

- Authentication and signature verification in cross border scenarios, supporting identity federation techniques.
- Strong authentication and signatures aligned with business requirements and legislation.
- Avoiding the need of special hard- or software that must be installed by the user in advance and may not be readily available locally.
- Avoiding cumbersome registration procedures, especially which require physical presence in remote locations.
- Intuitive interfaces as well as clear and understandable security policies.
- Minimisation of required trust in commercial providers or government authorities.

This list represents the most prominent goals. Among the fulfilment of the goals, the agreement on common (detailed) business requirements (see the above mentioned process model and use cases in order to get a complete requirement specification) is an absolute must, in order to develop and address the reference architecture properly.

During the development of the security architecture, especially while identifying information security requirements for the message flows from the process descriptions (see E Appendix – Process Model), it became apparent that some potential attacks cannot be mitigated by technical or organisational measures from SPOCS alone. During the preparation phase from the reference process model the ground is prepared for a number of attacks that are especially important in cross-border scenarios. We think it is

important to outline these in the context of this document and propose solutions although those are explicitly out of the SPOCS scope.

If during the preparation phase a Service Provider (SP) searches for the competent PSC to apply for a business in a foreign country, he will typically do so by using some of the popular search engines. It cannot be assumed that even if official national directories of valid PSCs exist, those are known to SPs, neither in a domestic context nor especially in an international context.

As a consequence the SP has basically no way to decide if the PSC proposed by a directory or a search engine is in fact a valid, official government operated one or rather a malicious impersonation. Such a malicious impersonation can prepare a number of phishing attacks. eSafe credentials, eDelivery accounts, personal identities and strategic business information is at risk here.

As a possible solution we would like to propose the establishment of a reserved government domain either on a national or on an EU level. This could be realised as a special domain under the national or `.eu` top level domain like `government.eu`, `government.de` etc. However, it is important that the naming scheme is consistent within the EU. All national agencies, PSCs and other public authorities are than required to register under those dedicated domains.

This way, a citizen of a MS could easily verify, if a foreign site is in fact a valid governmental agency or potentially spurious.

If such a decision is favoured by the commission and/or national authorities, WP8 would be an opportunity to communicate this approach to the general public.

If one takes into account the problems caused by phishing attacks, today it seems not acceptable to recognize potential avenues for such attacks within the SPOCS context without at least trying to mitigate them in advance.

1.5.1 eDelivery

As already stated in the assessment deliverable D3.1 [7], SPOCS WP3 has to deal with eDelivery infrastructures, which are focussed on the requirements regarding the national and/or regional level only. Obviously the solutions are not designed for cross-border interoperability yet. Even most of the eDelivery solutions are more or less all-purpose systems for secure e-communication over broadly available open networks, provided functionalities differ in detail, and there is a high diversity of architectural/technical approaches. Apart from the aspects of cross-solution connectivity and -interoperability, the required base eDelivery functionalities, as identified by the SPOCS process model and respective requirements, are provided by all solutions assessed.[8]

To interconnect the different eDelivery solutions transparently, the following main challenges will be solved by this specification:

Mapping of different message formats and transport protocols

To bridge between SMTP-, SOAP-based approaches, as well their (partly) proprietary extensions, respective general message structures (envelope and payload), transport metadata container, -formats and -semantics, message attributes relevant for securing

[8] Obviously, this had to be expected – eDelivery is a general e-government base infrastructure, as such already in use for most of the national SD implementations.

the message, further on those extensions used for addressing and authentication, these approaches must be assessed in detail and mapped to abstract, generic constructs.

Cross-solution addressing and routing

A solution comprehensive addressing scheme and routing model must be designed, capable to match the addressing attributes and -mechanisms of each solution to an abstract addressing model.

Trust establishment

Provision of secure eDelivery must be seen as services to be trusted in by all respective stakeholders. It can be assumed, that trust relationship between the relevant service nodes/instances is established inside each solution, in each case realised by different, Member State (MS) specific means. This has to be extended to the whole network of interconnected eDelivery solutions on the basis of a generic, comprehensive approach.

Authorisation and authentication

All solutions assessed have provisions to authenticate communication partners, needs and mechanisms for authorisation, which are foreseen in very different manners. Mostly, message exchange is only possible for communication partners registered in the particular domain of an eDelivery system. A concept must be developed, that allows secure cross-solution authorisation and authentication. This implies a calibration of respective procedures and mechanisms for user registration with regard to strength of identification and, again, attributes per user needed in the eDelivery process.

1.5.2 eSafe

The results from the assessment phase indicate that there is a small number of eSafes productive already. So far, the solutions are focussed on the national context only. Furthermore, the solutions are mostly integrated into portal solutions and are highly coupled with business applications, which are also integrated into the portal. This also results in a common usage of authentication and authorisation components of applications within the portal.

As the results show, eSafes are in an emerging phase (only) regarding the integration into public business applications. Therefore integration processes, addressing schemes, protocols and mechanisms regarding the usage across borders have to be developed by considering the current *nationally isolated* solutions.

Within the applied methodology, first an analysis and agreed documentation of a common process model has taken place. On this basis a common view regarding the required procedures followed by making use of the provided business use cases and requirements documents were established. Finally two alternatives evolved - the PUSH and PULL principle. Both principles support the interoperability layer and enable the concerned components (PSC, eSafe and SP) to communicate with each other. Furthermore, the PUSH principle is open for the usage and integration of the eDelivery component as well.

1.6 Approach basics

The WP3 vision is an interoperability solution to interconnect eDelivery, eSafe and eSecurity infrastructures in place in the MS. The interoperability layer must be able to support the functional and security requirements of each solution in scenarios of cross-border respective cross-solution document and message exchange. To make this task manageable and extendible, the concept is based on extensive use of according open, well accepted standards, designed to soundly interconnect distributed systems, even if partially based on different technologies.

1.6.1 Overall design principals of the security-sensitive infrastructure

Whenever distributed IT systems process valuable resources and make them available in an automated fashion, they depend on a security subsystem that facilitates support for concerns such as authentication and authorisation. This is particularly true for distributed applications in the SPOCS e-government environment, which demand adequate security not only for legal or regulatory reasons (e.g. privacy protection), but also for user acceptance reasons.

Modern distributed systems are no longer monolithic designs but adhere to the Service oriented Architecture (SOA) approach.

Composition of existing security-sensitive services in a SOA-based infrastructure entails a wide range of trust and security issues. Solving them is difficult, since securing all service components separately is not sufficient: composing services leads to new subtle and dangerous vulnerabilities due to interference between component services and policies, the shared communication layer, and application functionality. Thus, the overall service architecture needs validation of both, the service components and their composition into a secure service.

To meet these requirements, security has to be considered from the beginning (architecture and design phase) and a structured Security Architecture Development Approach has to be followed.

The intention of the Security Architecture Development Approach is to combine business requirements, architecture design principles and architecture constructs in order to develop national security architecture

The following three factors play a decisive role in the Security Architecture Development Process[9].

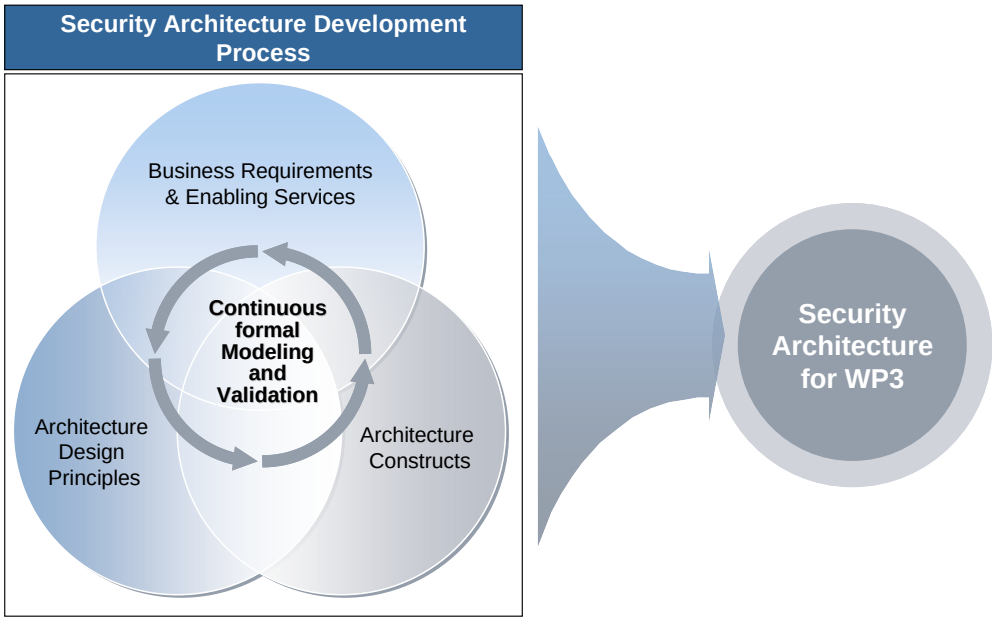

Figure 2 Security Architecture Development Process overview

[9] Following the general structure of NISTIR-7497 - Draft Security Architecture Design

Business Requirements and Enabling Services

As a general input for the security architecture, business requirements have to be considered. Business requirements include business processes and SPOCS use cases, and hence the functional requirements of the WP3 building blocks. Enabling services such as authentication, authorisation, data integrity and data confidentiality mechanisms serve as an input to create a notional architecture that achieves the security goals derived from the business requirements.

Architecture Design Principles

Architecture design principles are guiding principles identified from MS implementations or other best practices derived from large-scale information-sharing implementations. Design principles serve as the overall guidance for building security and privacy services for SPOCS. A design principle is e.g. the "least privilege" principle, "defence in depth" or to maintain a standards-based, technology-neutral, and vendor-neutral architecture.

The following architectural design principals were considered for the overall architecture:

- Conduct a risk assessment to determine appropriate assurance levels for shared information
- Create a "master" trust agreement describing requirements for a trust domain
- Separate credential management and privilege management
- Develop data protection capabilities as plug-and-play services
- Maintain a standards-based and vendor-neutral architecture
- Defence in depth
- Simplicity - easy to implement, easy to maintain
- Least privilege
- Need to know, need to access

Architecture Constructs

Architecture constructs originate from various industry standards and identify the basic security feature for a notional architecture. Important architecture constructs are e.g. Web services security standards for security management, identity management, message security, access control etc.

Continuous formal Modelling and Validation

In order to provide the best possible conditions for the completeness and correctness of the security architecture design in WP3, functions, structure and interrelationship of the security components of the security architecture are formally modelled and validated.

The main result of the formal analysis task is feedback on the completeness and consistency of the security architecture, as well as feedback on the effectiveness of the security mechanisms in its components and their interplay. Doing the formal analysis continuously in parallel with the development of the security architecture accelerates and maximizes its impact on the quality of the security architecture.

WP3 uses the results of the EU-project AVANTS[10]SAR: Automated VAlidatioN of Trust and Security of Service-oriented Architectures" (FP7-ICT-2007-1 Project no. 216471) for the formal specification and automated validation of trust and security of at least the most critical parts of the common specifications.

[10] see http://www.avantssar.eu/

1.6.2 Interconnection of eDelivery and eSafe systems

WP3 eDelivery cannot be build up "from scratch". Features, restrictions and requirements of existing solutions as stated in section 1.5.1 must be formally aligned to the design principals outlined above.

The high level design of interoperability layers follows the principles outlined above. In further functional concretion for the cross-solution message exchange scenarios, it must be able to serve as an "umbrella" satisfying the respective solution specific features, restrictions, requirements and message formats of existing solutions as stated in section 1.5.1. These details are related to generic, solution-neutral constructs. The latter build the base for the design of the interoperability layer general functionalities and message layout/constituents.

When connecting a concrete solution to the interoperability layer, besides a specification for mapping of functionality and formats a gap analysis is needed, identifying possible lacks completeness fulfilment as defined on the interoperability layer level. Identified lacks my lead to the need of providing according functionality in the gateway used to establish interconnectivity.

Basically the same applies for the integration of eSafes in WP3.

1.7 SPOCS framework regarding WP3

The scope of SPOCS is to propose a solution for the second generation of PSCs, which is enhancing interoperability features, facilitating the exchange of information and data in cross border scenarios. In this context, WP3 has a domain specific role to propose specifications for secure message exchange interconnecting the national eDelivery solutions of the Member States that participate in SPOCS. Moreover, WP3 is focusing in document exchange between PSCs and eSafe solutions from different Member States. Apart from that, WP3 has a horizontal role in proposing the mechanism that supports the creation of a Circle of Trust (CoT) among the entities that participate in the communication among the SP and the PSC. Such entities can be the PSCs, the Service Catalogues, eDelivery and eSafe solutions. From SPOCS pilot perspective, WP3 is supporting secure communication for specific scenarios and specific Member States.

WP3 incorporates existing national infrastructures by combining positive aspects of different solutions. WP3 will also use concepts and ideas from STORK (Pan European Proxy Services - PEPS), PEPPOL (Business Document Exchange Network- BusDox), ETSI (Registered E-Mail – REM), W3C, IETF, WS-I, and OASIS in order to propose an integrated solution suitable for SPOCS business case.

WP3's intention is to achieve the maximum possible security among the entities participating in the communication of a SP and a PSC. Every entity must be uniquely identified and conform to the security aspects of the WP3 proposed architecture.

All addressed security requirements such as confidentiality, integrity, authenticity, accountability and non repudiation will be satisfied for the entities involved, regardless of the external (uncontrollable) nodes that intervene in this communication.

WP3 will extend the message structures of the different national infrastructures, so that these messages can be interpreted by different systems in different countries. Furthermore, WP3 will propose methods and architecture for establishing circles of trust based on equivalent levels of trust level and thus ensuring a secure message exchanges and validation among different Member States. WP3 will also deal with other components such as message routing data, the message addressing information relating

to the delivery of a message and the message header, which are further described in the next sections.

1.8 SPOCS user scenarios regarding WP3

In this section main scenarios are described, in which at least one of the two WP3 components eDelivery and eSafe are actively involved. Firstly the overall SPOCS scenario is given; later the relevant scenarios regarding the WP3 components are further detailed.

1.8.1 General usage scenario

Initially the SP starts the scenario by applying for a service at the PSC. The PSC therefore offers information and finally an online application, in order to provide the best possible support for the SP according to the main intention of the service directive. Depending on the solution of the PSC, the SP needs to authenticate first.

The SP fills out the online form and has then to provide additional documents, if required by the competent agency.

Depending on the functionalities of the PSC application solution and the location in that the SP stores these documents, the SP may choose to

(1) submit them by directly via a file upload to the PSC portal (this case is not relevant for WP3, thus not further considered in this document, even if the SP previously downloaded the documents from the eSafe)

(2) use his eSafe to attach those documents (following either the PUSH or the PULL principle)

(3) use the eDelivery channel to submit the documents in an asynchronous manner.

After the application form is completed and all relevant documents are provided (which may take some time when submitting documents via the eDelivery channel), the PSC handles the processing of the application with the relevant CAs concerned.

(4) If further information is required, the SP is asked to provide them in order to finalise the request. The eDelivery channel may be chosen by the PSC or the CA to address this request to the SP; the SP then provides information and/or documents according to **(1), (2)** or **(3)**.

(5) Finally the CA creates the admittance attestation or notice and submits it to the PSC for delivery to the SP. Depending on the functionality offered by the PSC, the attestation may be transmitted to the SP using eDelivery or made available for download in the PSC's portal; in the latter case, at least a notice should be send to the SP (any communication channel available may be chosen here – eDelivery, phone call etc.).

The responsibilities of the PSC and CA might vary within the different national solutions. It might be possible that the PSC delivers a preliminary attestation on the basis of formal completeness of the required documents, following the same flow as above **(5)** but with the PSC as initiating party rather than the CA.

Most of electronic documents and other information to be supplied by the SP may require authentication by means of digital signatures. This at least applies for the admittance attestation (or even detailed information in case of non-admittance) to be delivered to the SP by electronic means. e-Signature application and validation is out of scope of the present specification, as assumed to already be constituted by the national

implementation of the SD; for cross-border e-Signature, respective eID validation, we refer to the solutions provided by PEPPOL and STORK.

Interaction between the different eDelivery and eSafe systems involved in the usage scenarios must be possible without lack of confidentially, reliability, authenticity, evidence traceability and other security requirements.

1.8.2 SP attaches documents to the PSC/CA by using his eSafe

*Relevant in scenario **(2)** and **(4).***

Once the SP needs and wants to attach documents, the SP's eSafe can be the source of the required authenticated documents. Therefore the PSC redirects the SP to his eSafe by using the eSafe's URI. After authentication, the SP then works directly in his eSafe and selects one or more documents to be attached to the application request at the PSC. Once selected, the eSafe offers the SP to initiate the document transfer. If confirmed the PSC receives the documents and attaches them to the previously started application request. The SP is redirected back to the PSC and proceeds with the application request.

1.8.3 SP delivers documents to the PSC by using eDelivery

*Relevant in scenario **(2)** and **(3).***

If the PSC is reachable through a domestic eDelivery system, the SP may also use the eDelivery channel for submitting the documents to the PSC's mailbox in order to attach them to the application request. Therefore the SP has to know the e-address (e.g. by publishing it on the PSC's portal).

In the current stage of SPOCS, using the eDelivery channel requires that the SP has the documents locally available, maybe previously retrieved from other sources (considering also his eSafe). Future (eSafe) solutions may also offer the delivery of documents via eDelivery directly.

Anyhow, once the documents arrive in the PSC's mailbox they are to be attached to the application request. Depending on the PSC's implementation this may either be handled fully automated or may require human interaction at the PSC side.

1.8.4 PSC/CA delivers the application attestation, requests for further information or notices to the SP by using eDelivery

*Relevant in scenario **(4)** and **(5).***

The eDelivery channel may be chosen by the PSC for further inquiries, as well as the delivery of the application process results to the SP. To facilitate this, the precondition is that the SP provided at least one e-address used in an eDelivery system he is registered in. The PSC/CA then uses their domestic eDelivery infrastructure to edit, address and dispatch the messages to the SP's eDelivery inbox.

1.8.5 SP provides further information/documents to the PSC by using eDelivery or eSafe

*Relevant in scenario **(4)** and respective **(1), (2)** or **(3).***

If the SP needs to provide further information and/or documents, the same scenarios are available as described in 1.8.2 and 1.8.3.

1.9 Pre-requisites to use the SPOCS open modules

1.9.1 Pre-requisites to use the SPOCS eDelivery open modules

This section provides an overview of the requirements national PSCs and eDelivery Gateway Providers should fulfil in order to use the SPOCS eDelivery open modules according to the specifications provided.

The remainder of this section is organized as follow: Section 1.9.1.1 contains a checklist of the minimal pre-requisites needed to use the SPOCS eDelivery open modules. Section 1.9.1.2 lists the module's functionalities (or features) that can be deployed provided the minimal pre-requisites are met.

1.9.1.1 Checklist minimal pre-requisites

In order to use the SPOCS eDelivery open modules, national PSCs and the eDelivery Gateway Provider should fulfil the following pre-requisites.

eDelivery Gateway Provider

- Runtime environment for the open modules
- Security features

  - Cryptography extensions to create and validate electronic signatures
  - TLS security mechanisms for secure messaging
- Registration as REM service within the SPOCS TSL according to the specifications of D3.2 Appendix 2.
- Internet connection to access the SPOCS TSL and other registered eDelivery Gateways according to the SPOCS Interconnect Protocol specifications of D3.2 Appendix 3.
- Interface to enable access for national PSCs, either by
  - Connecting the Gateway to the national eDelivery infrastructure or
  - Providing a proprietary interface

PSC

- Accessibility to the national eDelivery Gateway Provider by using the local eDelivery infrastructure, either by
  - directly connecting to the Gateway or
  - using the national eDelivery infrastructure
- Knowledge of recipient's electronic address

1.9.1.2 Enabled module's features for minimal pre-requisites

Based on the minimal pre-requisites, national PSCs will be able to use the following SPOCS eDelivery open modules features:

- Sending of arbitrary documents to recipients of a foreign REM system
- Receiving of arbitrary documents from senders of a foreign REM system

The interface between PSCs and the SPOCS eDelivery open modules are country specific and will be specified by each piloting country. The SPOCS Interconnect Protocol for inter-gateway communication is specified according to D3.2 Appendix 3.

1.9.2 Pre-requisites to use the SPOCS eSafe open modules

This section provides an overview of the requirements the national PSC or eSafe solutions should fulfil in order to use the SPOCS open modules for eSafe integration according the specification provided.

The remainder of this section is organised as follow: Section 1.9.2.1 contains a checklist of the *minimal pre-requisites* needed to use the SPOCS eSafe open modules. Section 1.9.2.2 lists the module's functionalities (or features) that can be deployed provided the PSC contains the minimal pre-requisites.

1.9.2.1 Checklist minimal pre-requisites

In order to use the SPOCS eSafe open modules, the national PSC or eSafe solution should fulfil the following pre-requisites:

eSafe and PSC

- Runtime environment for the modules as well as the integration via Java APIs (currently the development is planned to be based on JEE)

- Consideration of the security requirements, especially operating the website via SSL/TLS
- Registration for the service to be included in the SPOCS TSL according to the specification D3.2, Appendix 4
- Accessibility of services from the Internet
- Measures to support the UI-URLs according to the specification D3.2, Appendix 4
- Support of the minimal required cryptographic algorithms listed in the specification D3.2, Appendix 4 by a Java Security Provider

eSafe

- Portal functionality offered to the SP in order to select documents from his eSafe
- Measures to hand over documents from the eSafe to the modules for the transfer to the PSC according to the specification
- Provisioning of minimal meta information according to the specification D3.2, Appendix 4

PSC

- Portal functionality offered to the SP in order to attach documents to an application request via an eSafe according to the specification D3.2, Appendix 4
- Measures to bind an OCD container (according to the D2.2) received from the eSafe to the application request from the SP

1.9.2.2 Enabled module's features for minimal pre-requisites

The detailed features are described in the deliverable, D3.2 Appendix 4. As a minimum the modules will provide the whole set of messages required for supporting the PUSH principle.

2 Interoperability layers topology, base technology and architecture

2.1 Trust model and TSLs

Establishment of end-to-end trust is a critical factor in an eDelivery process as well as for reliable functionality of eSafe and other services. The proposed trust architecture is based on the hypothesis that trust within each eDelivery realm (context of one eDelivery solution) is established by specific policies in the realm itself. Realms will normally provide other different services as well (eSafe, PSC, Service Catalogue and Directories, etc.).

For what eDelivery is concerned, SPOCS is going to add a gateway-to-gateway trust. Generally explained, gateways are entry and exit points of one system which allow or control access to other systems. In this scenario gateways, besides being included in real specific trust circles, must also be part of the gateway trust circle, which is managed by SPOCS (at least in the piloting phase).

On the other hand, trust needs to be established between PSCs , eSafes and other services as well in order to prevent unintended document transfers.

2.1.1 Trust-service Status List (TSL) for eDelivery and eSafe services

The model is based on Trust-service status List, as defined by ETSI TS 102 231 v3.1.2 [9], which specifies the incorporation of status information for general trusted services. The model will reflect the trust model of the pilot solutions of our companion project PEPPOL, WP1 D1.1 part 4 [10], while extending managed entities to Trusted Service Providers (TSP) different from Certification Authorities. In SPOCS an own (SPOCS-) TSL will be managed for SPOCS managed services.

The TSL will list the related eDelivery gateways specifying both their current status and, optionally, their status history. Any gateway belonging to a different realm will access the TSL and verify its authenticity as provided for by the SPOCS policies. In this TSL, the signing public key of each gateway, or preferably its corresponding certificates, will be published so that any relying party will be able to use it to verify the signature of each eDelivery Envelope issued. Where applicable, the history of the gateway's status (when it was initially introduced, if and when such an approval was revoked and when it was subsequently reinstated) may also be found in the TSL. This allows users to ascertain the reliability of a specific envelope for the past as well.

eSafe instances on the other hand are included in order to establish a basic trust model within SPOCS providers beforehand. eSafes are listed as provided in the TSL on the PSC for selection of the SP when choosing an attachment of documents out of the SP's eSafe. Furthermore the certificates given in the TSL enable the communication partners to verify the trust relationships. This is also the reason why many different services are listed in the TSL as well with their certificates.

2.1.2 TSL content

According to TSL specifications, services included in the list must be characterised with respect to the services they offer. The mandatory field "Service type identifier" explicitly

accounts for this characterisation; for TSL type “Generic”, values are restricted to specific URIs as defined by ETSI TS 102 231[11].

Nevertheless, other URIs may be used for this field, provided they are registered and described by the scheme operator or another entity that is recognized by the intended user community. It is however forbidden to introduce new URIs related to “local” services. In case this field is left unspecified, it is still possible to characterise the service using the TSP "Service information extensions" element in the TSL.

We recommend the introduction of specific URIs for the purpose of the SPOCS project:

1. http://uri.spocs-eu.eu/Svctype/eDelivery
2. http://uri.spocs-eu.eu/Svctype/eSafe
3. http://uri.spocs-eu.eu/Svctype/PSC/v1
4. http://uri.spocs-eu.eu/Svctype/SC/v1
5. http://uri.spocs-eu.eu/Svctype/eSD/v1
6. http://uri.spocs-eu.eu/Svctype/searchModule/v1
7. http://uri.spocs-eu.eu/Svctype/syndicationModule/v1

In the longer term, it is envisaged that sustainability is achieved by ensuring that a central European authority takes in charge the management of the URIs above (see section 2.1.3.1, TSL operations and administration).

TSL can also be leveraged for distribution of routing information between different gateways (see section Cross eDelivery Management Domain/Realm addressing). In this case an appropriate "Service information extensions" field shall be used. Extensions are freely selectable by the scheme operator according to the meaning and information she wants to convey within the TSL.

The concrete TSL profile specified for SPOCS is defined in a dedicated document Appendix 2: Trust-service Status List Profiling ("SPOCS TSL"). Relevant TLS policy issues will be provided in subsection 2.1.3.

2.1.3 TSL policy issues

The entire policies accompanying the usage of TSL as detailed in the sections below should be regarded as work in progress as they need to be aligned with several other EU projects such as PEPPOL and BusDox. As of today these projects have not made policies regarding TSL available. So an alignment process was initiated but not yet finished.

2.1.3.1 TSL operations and administration

With reference to the different operational approach to TSL management, SPOCS follows the general trend promoted by the EC. The model is based on the standard ETSI TS 102 231, recently updated (v.3.3.1 October 2009) and defined by the European Commission, with Decision 16th Oct. 2009 n.767, the standard model to issue the National Trust Lists.

[11] As e.g. http://uri.etsi.org/TrstSvc/Svctype/CA/QC for a certification authority issuing public key certificates

On the basis of the above Decision, Member States should implement a TSL containing at least Certification Service Providers issuing Qualified Certificates.

The expectation is that other Trusted Services will be included as well in the same MS's TSL. SPOCS WP3 will contribute to this standardisation process, aligned with PEPPOL WP1 for including validation services in the TSL.

Currently a TSL may be implemented in two distinct formats – a human-readable PDF-format and a machine readable XML-format.

Note: SPOCS will only use the TSLs in XML-format, as TSL entry location and according attributes of Trusted Services addressed here must be machine processable at runtime[12].

Note: As defined in detail by ETSI TS 102 231, Annex B.6, TSLs have to be protected by an enveloped XML-Signature following ETSI TS 101 903 (XAdES) [27]. The certificate used for signing MUST be exposed by the TSL issuer.

2.1.3.2 Decentral TSL maintenance and publishing

This model is the standard one chosen by the EC for the Trusted Lists (TL) of qualifies CSPs. TL issuers are usually national supervision agencies, the EC only provides a central List of Lists[13] pointing to the TL instances provided by the MS.

[12] TSLs published in PDF-format could only be used for according configuration of the system components involved here, which is seen as additional operational burden and source of possible faults.

[13] The location of the EC List of Lists is https://ec.europa.eu/information_society/policy/esignature/trusted-list/tl-mp.xml

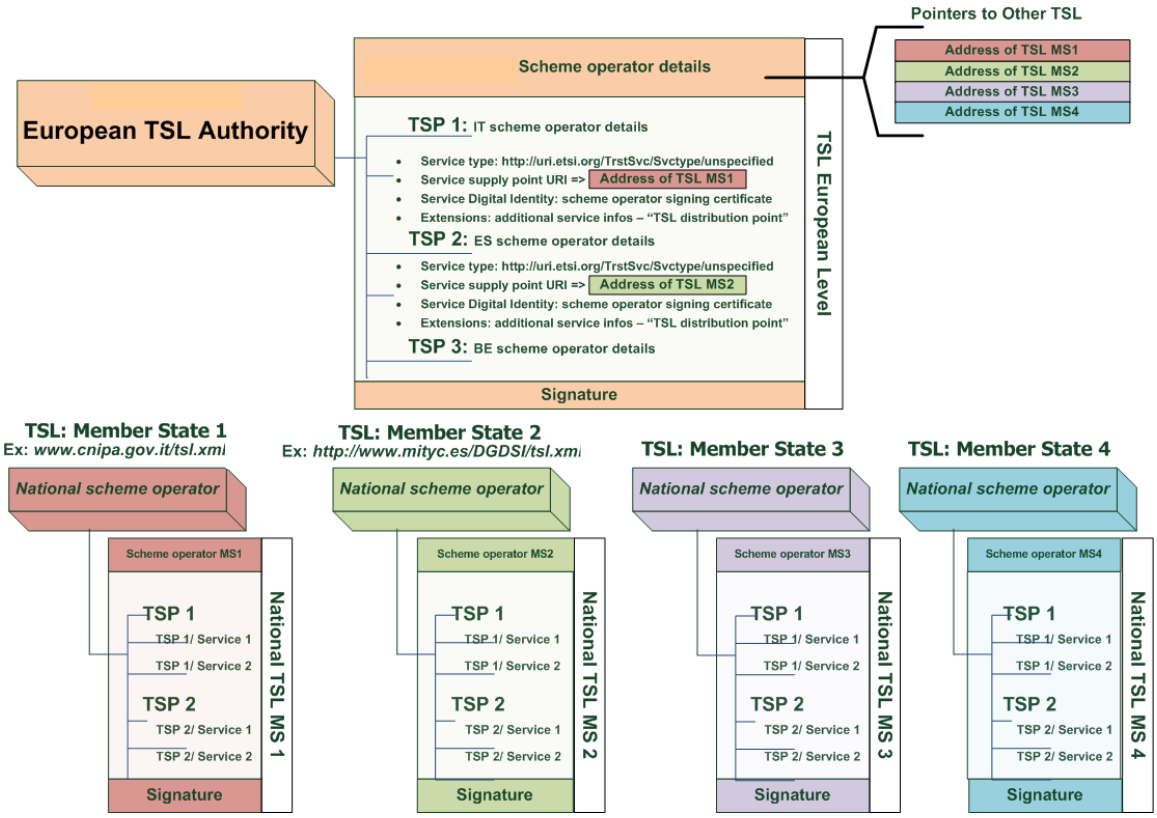

Figure 3 TSL - operational hierarchy

In this situation the management of TLs would be in charge of MS's. Unfortunately we cannot rely on this architecture in the time frame of the SPOCS pilots.

2.1.3.3 Central TSL maintenance and publishing

As extended TSL issuing operated by the MS will not be in place in the short term, SPOCS must take an active role in issuing TSLs for pilots. At least for the piltoting phase, SPOCS TSL will be maintained and issued by one of the consortium partners[14].

Note: SPOCS will provide a centrally managed TSL within the piloting phase.

WP3 will provide an according TSL editor, extending the standard functionality for those entries as defined in Appendix 2: Trust-service Status List Profiling ("SPOCS TSL").

The SPOCS TSL will be published under following unique, project-specific URL:

`http://www.eu-spocs.eu/TSL/currentTSL`

TSLs contain an element `tsl:NextUpdate` outlining the `tsl:dateTime` for the next time instant an update of the central SPOCS TSL must be expected.

2.1.3.4 Accreditation process of eDelivery and eSafe systems

The inclusion of an eSafe, an eDelivery provider or a gateway in a circle of Trusted Services represented by a SPOCS-TSL requires an accreditation process. For a service to be accredited, the service must explicitly consent to a "SPOCS-TSL accreditation and

[14] Details will be published on the SPOCS portal, as soon as agreed on the operational and governance details by the SPOCS EB

operation Policy". Please refer to the dedicated document Appendix 5: SPOCS TSL Accreditation and Operation Policy

2.2 Security Architecture

The intention of the Security Architecture Development Approach was to combine business requirements, architecture design principles and architecture constructs in order to develop a consistent security model. For a detailed description of the security architecture development process please refer to Appendix 1.

As already mentioned in section 1.6.1, three factors play a decisive role in the Security Architecture Development Process. Business requirements, enabling services and the resulting security targets, architectural design principles, and architectural constructs were formally modelled and validated. Due to the limited scope/coverage of SPOCS and the necessary interfaces to and dependencies on local solutions we had to make a number of assumptions and limitations. We intentionally limited the scope to cover only the SPOCS infrastructure components in detail. For all domestic systems and processes we made plausible but simplifying assumptions. We consider it important that such a systematically approach be taken to iteratively improve the overall security. It should be clearly stated that a number of security related issues cannot be solved (not even addressed consistently) on a technical base only, but require changes in business/administrative processes and procedures, political commitment, and changes in accompanying legislation.

The security concept of the SPOCS interoperability components is formed by the following building blocks

- Interoperable entity authentication methods and identity federation support (federate identities securely between systems)
- eDelivery gateways for secure and reliable document transport with non-repudiation for sender and receiver
- eSafes
- Trust relationships via SPOCS TSL and X509v3 certificates
- Interoperable document signature
- Trustworthy information for SP and third party websites tailored for cross border scenarios
- Secure operation guidelines for SPOCS interoperability components

In the following subsections provide a brief security context overview, where identified SPOCS roles and security objectives are described as well as some general security measures for authentication and communication are given.

2.2.1 SPOCS roles

The following roles and components are important for the overall understanding of the architecture:

- Service Provider (SP)
  A Service Provider is a citizen or a legal person from a member state who wants to inform herself or apply for starting a business in another member state. This SP must not be confounded with the commonly known IT service provider, who is called a resource provider in here. A typical SP is assumed to be only familiar with basic computer and internet technologies. Consequently each concept must take into account that installing additional software and hardware components as well as understanding of cryptographic methods, certificates, validation procedures, and lurking attacks cannot be assumed.

- Resource Provider
  A Resource Provider is typically a website that provides information and or services to end users, in our case to SPs.
- PSC
  Point of Single Contact for online applying for a business according to the EU service directive.
- Identity Provider (IdP)
  An Identity Provider stores information about a subject's digital identity and are able to authenticate those subjects. It may issue identifiers about those subjects to other entities and provide assertions about a subject and its authentication status.
- V-IDP
  A proxying instantiation of an IdP.
- Attribute Provider
  An Attribute Provider stores information about a subject's digital identity. It may issue information about attributes of a subject to other entities. Typically an attribute provider is either associated or incorporated with an IdP.
- PEPS
  As defined in STORK a Pan European Proxy Service is a proxying component that helps evaluating security assertions in complex, cross border federation scenarios.
- TSL maintainer
  An entity that maintains a TSL as defined by ETSI and detailed for SPOCS WP3 purposes in the present document.
- Local registration authority
  A local registration authority establishes the link between a physical person and her digital identity. A user registers at registration authority authenticating herself in a traditional way (e.g. password, national ID card) and is subsequently provided with a digital identity.
- eDelivery Gateways
  eDelivery Gateways are national “channels” or systems for data transmission between two communication partners. Those can be human endpoints with their client computers or protected services like eSafes (see next bullet). These Gateways are required to fulfil maximum security requirements within its operation but also for the communication with others.
- eSafe
  A source of authenticated documents where a SP can store any type of electronic document securely. If needed, the SP can share documents with externals.

2.2.2 Security objectives

The following list of primary security goals are given as the motivation for the security architectural decisions and security mechanisms described in sections 3 and 4. The acronyms used in the beginning of each bullet point will be referenced in Appendix 5

CON_BIZ: Confidentiality of strategic business planning information of SPs

- between SP and PSC (mandatory, this is in the SPOCS focus)
- between SP and CA (recommended, but not governed by SPOCS)

- **CON_PID**: Confidentiality of PID (personally identifiable data) of SPs
  - between SP and PSC (mandatory, this is in the SPOCS focus)
  - between SP and CA (recommended, but not governed by SPOCS)

- **INT_PID**: Integrity of personally identifiable data of SPs
  - between the SP and PSC (mandatory, this is in the SPOCS focus)
  - between SP and CA (recommended, but not governed by SPOCS)
- **AUT_PID**: Authenticity of PID, current and in long-term retrospective (e.g. for forensics)
- **AV_PID**: Availability of PID
- **CON_DOC**: Confidentiality of submitted documents of the SPs
  - between SP and PSC (mandatory, this is in the SPOCS focus)
  - between SP and CA (recommended, but not governed by SPOCS)
- **INT_DOC**: Integrity of submitted documents of the SPs
  - between the SP and PSC (mandatory, this is in the SPOCS focus)
  - between SP and CA (recommended, but not governed by SPOCS)
- **AUT_DOC**: Authenticity of submitted documents of the SPs
- **AV_DOC**: Availability of submitted documents of the SPs
- **NR_DOC**: Non-repudiation of submitted documents
- **TS_DOC**: Authentic time stamps on submitted documents
- **RET_DOC**: Compliance with Retention time legislation for submitted documents of the SPs (recommended, but not governed by SPOCS)
- **CON_DEC**: Confidentiality of official notifications on CA decisions
  - between PSC and SP (mandatory, this is in the SPOCS focus)
  - between CA and SP (recommended, but not governed by SPOCS)
- **INT_DEC**: Integrity of official notifications on CA decisions
  - between PSC and SP (mandatory, this is in the SPOCS focus)
  - between CA and SP (recommended, but not governed by SPOCS)
- **AUT_DEC**: Authenticity of official notifications on CA decisions
- **AV_DEC**: Availability of official notifications on CS decisions
- **NR_DEC**: Non-repudiation of official notifications on CA decisions
- **TS_DEC**: Authentic time stamps on official notifications on CA decisions
- **RET_DEC**: Compliance with Retention time legislation for official notifications on CA decisions (recommended, but not governed by SPOCS)

Due to the scope of SPOCS, which deliberately does not cover the communication between PSC and individual CAs, it must be explicitly stated that end-2-end Security (with regard to integrity and confidentiality) cannot be guaranteed. Especially end-2-end encryption will not be delivered by SPOCS because this would require that every national and regional CA has a valid encryption key. This is however subject to national governance. SPOCS will try to support end-2-end security as best-effort wherever possible.

2.2.3 General security measures for authentication and communication

Authentication credentials must be transferred only over secure communication channels. Therefore HTTPS (SSL/TLS) is the preferred standard and will be used for all connections between:

- SP and PSC
- SP and eSafe
- eSafe and PSC
- SP and IdP (not governed by SPOCS)

WS-Security with X509 profile and XML-Encryption will be used if SSL/TLS is not applicable especially for communication between:

- IdP and V-IDP (not governed by SPOCS)
- Resource provider and PEPS (not governed by SPOCS)
- PEPS and V-IDP (not governed by SPOCS)
- PEPS and PEPS (not governed by SPOCS)
- V-IdP and V-IdP (not governed by SPOCS)
- eDelivery gateways
- PSCs are expected to present at least a valid certificate from a generally recognised certification authority. To be more specific this means from a certification authority that is one the default list of trusted authorities present in widely-spread browsers like
- Microsoft Internet Explorer and
- Mozilla Firefox.

This is intended to provide the SP a basic server authentication.

- For the authentication of entities (humans, SP) each system will provide:
- At a minimum local registration with password based authentication
- Optionally local registration with Smart Card based authentication using X509 certificates
- Optionally registration at a remote identity provider and authentication towards the identity provider who will issue security assertions for evaluation by the requested resources.

For password based authentication

- a non-trivial password policy must be enforced
- passwords must not be stored in clear text
- passwords must not be transmitted in clear text
- passwords must not be visible during input.

If certificate based smart card authentication is supported, each component (be it an eSafe, a PSC or any other) must ensure that certificates from a multiple certification authorities are accepted.

Particularly all certificate authorities that are included in national TSL must be accepted. This can either be achieved by locally maintaining and evaluating against the current TSL or to make use of the STORK authentication framework for identity federation. This second approach is recommended. However for the piloting phase we could not depend on it as it is not yet rolled out.

For the handover of eDelivery Sender authentication information the originating eDelivery system is free to implement any authentication method. The authentication information is transported to foreign systems as a SAML sender-vouches. This means that the originating system uses the sender-vouches confirmation method to assert that it is acting on behalf of the subject of SAML statements attributed with a sender-vouches SubjectConfirmation element.

The authentication of systems will use:

- unilateral authentication using X509 certificates
- mutual authentication[15] using X509 certificates.

Valid certificate authorities must be obtained by evaluating a valid SPOCS-TSL.

The SPOCS-TSL will extend the service type see section 2.1 with entries for

- eDelivery gateway
- eSafe
- PSC
  The TSL will also need to be extended by certificate authorities that are “non qualified” but nationally approved within at least one MS

Each interoperability infrastructure component, in particular

- PEPS
- V-IdP
- eDelivery Gateway
  must be able to present a valid certificate from a certificate authority included in at least one current valid TSL.

2.3 eDelivery

For each eDelivery solution in place we can assume the "service provider model" – communication endpoints (message originators and recipients) exchange data using a server side, centralised or federated infrastructure for message dispatching, relaying and/or routing, registration and authentication.

Besides secure and reliable message delivery, the infrastructure must provide mechanisms for proper management of “evidences”. Actors should be provided with the possibility to prove successful "in time" communication / document delivery in case of disputes.

Section 7 of Appendix 3 contains an overview of the eDelivery solutions in the piloting countries.

2.3.1 Relevant entities and actors

eDelivery Management Domain (MD)

Each instance of such an eDelivery communication network is referred to as eDelivery Management Domain. A MD is an organisational instance operating – through one ore more servers - one or more trusted eDelivery domains or realms (see next paragraph) under given national regulation or comparable settlements. eDelivery functionality is

[15] eDelivery Gateways implement a specific authentication mechanisms to be able to validate a presented SSL certificate against the SPOCS TSL

provided for actors bound to a MD (e.g. senders/recipients registered here or an independent IdP instance trusted by the MD). MD is in charge of:

- Authentication of actors
- Reliable message routing/delivery (outbound case, when acting for the sender)
- Message acceptance and relaying (inbound case, secure message box provided for recipient)
- Authentication of nodes a message is targeted to (or: originates from)
- In most cases: provision of message transport evidences, enabling actors to trace (or even to prove) successful message delivery or the respective cause of unsuccessful message delivery.

eDelivery Realm

Some MS have interconnected MDs to a circuit or "Realm" of domains. In such a Realm, MDs are operated on base of one common protocol and functionality stack; trust relationship is established between respective eDelivery MD providers.

Dispatch message

A Dispatch is a bundle of data, originated by its sender and targeted to recipient(s) chosen by the sender. The original message as provided by the Sender's domestic solution SHOULD be included in the Dispatch. To facilitate mapping between different message and addressing formats in use by Sender's and Recipient's Realm, the original message MUST be converted to a "normalised format" as specified in this document.

In addition, a Dispatch MUST carry a security token stating the Senders authenticity as defined in section 3.4.

A Dispatch MAY be enriched by evidence information items on the message route, if those evidences should be targeted to nodes on the forward message route.

According to the specification ETSI TS 102 640 adopted, the technical term for this type of message is **REMDispatch**.

Evidence message

An Evidence is a message carrying information on evidences raised on the message route. Different evidences may be raised by all entities involved in the message transport and may be targeted to different consumers (transport nodes and/or Senders as well as Recipients). They serve as control, proof or notification of the Dispatch message flow. Evidence data is digitally signed and thus transferable, i.e. it can be used by each party to prove authenticity of the Evidence without the need to involve the issuing entity.

An Evidence message MAY be amended by the Dispatch message the Evidence is related to.

According to the specification ETSI TS 102 640 adopted, the technical term for this type of message is **REMMDMessage**.

Sender (SE)

The Sender is the originator of a Dispatch. For all eDelivery solutions which are considered here, the Sender must authenticate to his MD via his eDelivery provider, but the choice of authentication mechanisms is left to the specific eDelivery provider. The used respective policy or regulation is bound to the solution operated by the provider.

Recipient (RE)

The entity a Dispatch or Evidence is addressed to. Alike the Sender, the Recipient must authenticate to his eDelivery provider, using the MD specific mechanisms. Recipients can – by means of the specific implementation – retrieve messages from a personalised, secured message box, provided by the MD provider they are registered at.

eDelivery Gateway (GW)

A Gateway is a service used to facilitate message exchange between MDs or Realms based on different message formats and transport protocols. For the cross-MD/Realm message exchange, Gateways are in charge of

- the appropriate message conversion and reliable cross-MD/-Realm routing
- provision of message transport evidences
- mutual authentication of nodes involved in the cross-MD/-Realm message transfer (by using the SSL/TLS and the SPOCS TSL in the present concept).

Note on generic GW and local Realm GW parts:

This specification specifies the GW functionality and the message format used on the interoperability layer ("**Interconnect Protocol**"). The **generic GW** functionality will be provided by open modules implementing this specification, while the **specific local Realm GWs**, which are primarily in charge to perform the conversion from/to the Realm's domestic protocols on base of the API provided by the generic GW implementation. The specifications for these local Realm GWs are not part of the present document.

SPOCS Trust-service Status List (SPOCS TSL)

A TSL is used for cross-MD/Realm trust establishment and routing. This TSL carries an entry per instance of an eDelivery Gateway. For details, see Trust model, section 2.1.

2.3.2 Topology

The Gateway model is chosen to interconnect the MS' eDelivery MDs and/or Realms, essentially based on message exchange between MDs/Realms in a *1:n* manner using one common "Interconnect Protocol" for inter-MD/Realm message transfer. For each MD/Realm, Gateway-functionality has to be established for the conversion from the domestic protocol to the Interconnect Protocol, and vice versa, as well as for outbound routing and inbound message entry.

Compared to *n:n* or central hub topologies, the Gateway model provides following advantages:

- Minimized complexity; each specific Realm (or MD instance respectively) only has to establish the conversion from/to domestic versus Interconnect Protocol via a MD/realm specific Gateway implementation; finally it must be registered to the SPOCS-TSL.
- No centralised, EU-wide service instances needed in addition the SPOCS-TSL mentioned above; no point of single failure, if the SPOCS-TSL is replicated to local instances.
- "Local" maintainability: changes in the local infrastructure/protocol only affect the locally used Gateway.
- Scalability: Future integration of additional MD specific eDelivery solutions require an appropriate new Gateway, none or only minimal effects on the existing interconnect infrastructure and solutions already connected herewith.

Figure 4 illustrates of the landscape of different eDelivery solutions interconnected by gateways:

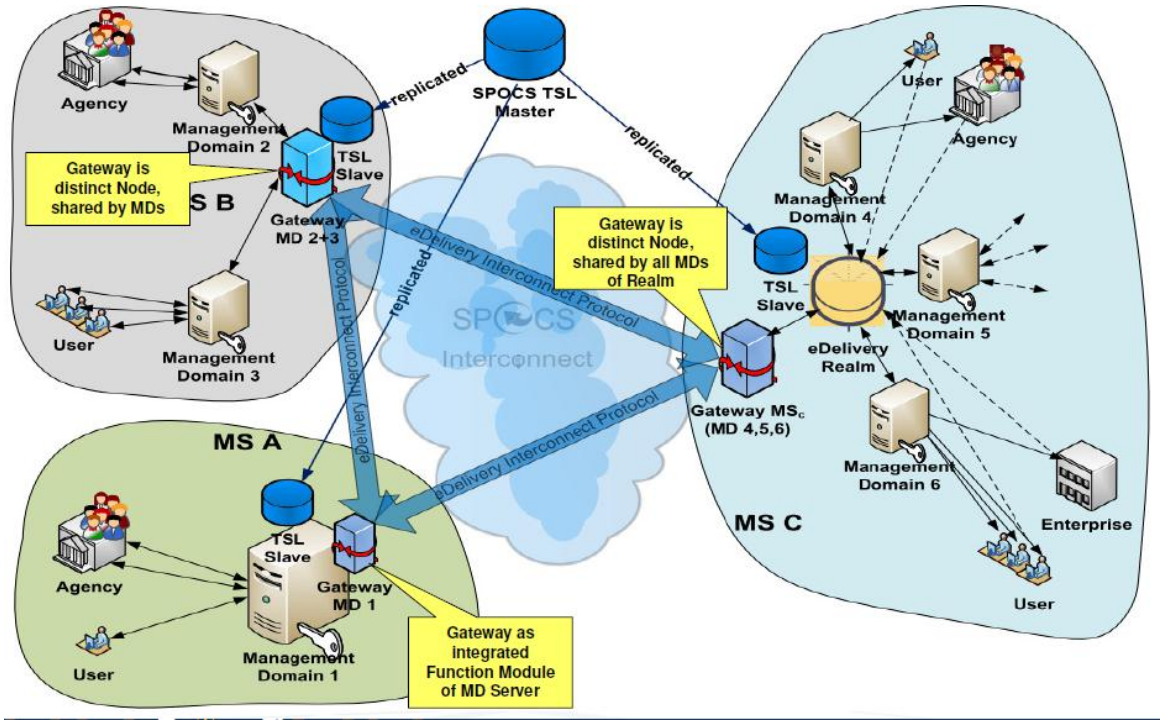

Figure 4 Interconnected eDelivery systems landscape

In general, originators/Senders of a message should be able to provide e-addresses as in use in the target MDs/Realms. Foreign MD/Realm e-addresses must be recognised by means of the respective MD solution itself or by using the according target MD entry of the SPOCS-TSL.

In the outbound case, this entry is used by the Gateways to derive the technical address of the Gateway the message must be targeted to after conversion to the Interconnect Protocol format.

The SPOCS-TSL should be held as copy locally available, but the central one provided by the SPOCS consortium may be used, too. The standard generic part of the Gateway implementation will assume a local copy.

Three variants are shown in Figure 4:

1. **MS A, Gateway bound directly to the MD system instance of solution provider**
   Outbound routing is generally handled via the MD system instance(s) of the Sender, which is extended by the Gateway functionality. More the one MD system instance of that type may be operational in this MS.

2. **MS B, MD instances share an instance of a Gateway**
   The use of the same domestic protocol by the different instances and mutual trust establishment is assumed. The Gateway is operated as separate instance, used by all domestic MDs connected. In the inbound case, this Gateway can route the message to the appropriate MD system instance on base of the e-address of the recipient. It is a matter of the concrete implementation, whether attributes of the according MD entry in the SPOCS-TSL must be reflected in this case.

3. **MS C, using a MS-central Gateway instance, positioned as additional member of the eDelivery realm in place**.

Here, for the outbound case, Senders use central Realm functionalities (e.g. registry lookup) initially before dispatching messages directly to the target MD instance determined in the initial step (for simplification, shown as one dashed line above). This will be the same for cross-border targets – the central Gateway is the "proxy" node the message has to be dispatched to, representing all external MD instances connected to the SPOCS eDelivery network. Gateway functionality for messages inbound to this circle is very much like outlined above for MS B. Depending on concrete needs, central Realm functionalities for routing must be involved in the Gateway implementation. Recipients connect directly to their MD provider system instance only for message inbox access.

2.3.3 Base protocol stack

Basically, SPOCS eDelivery has to deal with two different transport protocol stacks. Solutions are based on:

- The e-mail protocol stack, extended by metadata containers or x-headers to realize the requirements going beyond standard e-mail functionalities (trust establishment, other security, evidence tracking …). The approach of ETSI TS 102 640 "Registered E-Mail" (REM, [12]) must be seen as one of the straightforward one in this landscape.
- The Web services protocol stack, mostly somewhat profiled and/or extended according to concrete MD solution requirements.

For the SPOCS interconnect protocol, WS-* is chosen:

- WS-* is standardised on the international level, well adopted and established by the IT industry
- WS-* offers dedicated bricks for the different feature classes (addressing, security and trust, reliable delivery…), combinable according concrete requirements to be served
- Flexibility: standardised, machine consumable profiling mechanisms
- Many implementations available, many of them open-source-softwares
- Strong interoperability efforts driven by implementers (e.g. ws-i.org[16], WSIT[17])

Following functionalities and underlying specifications of the Web services stack are used or profiled by the WP3 eDelivery interoperability layer:

- SOAP 1.2 as enveloping message structure
- MTOM[18] for seamless attachment handling
- WS-Addressing, profiled to serve the particular requirements
- WS-Security and underlying XML Signature for message security and authentication of Gateways exchanging messages. Confidentiality and message integrity is handled on the transport layer (SSL/TLS)
- WS-ReliableMessaging [48] to ensure message delivery in an “exactly once” manner.

---

[16] http://www.ws-i.org (last visited Sep 11, 2010)

[17] Web Services Interoperability Technologies (WSIT); (previously known as Project Tango) https://wsit.dev.java.net/ (last visited Sep 11, 2010)

[18] SOAP Message transmission Optimization Mechanism, http://www.w3.org/TR/soap12-mtom/, (last visited Sep 11, 2010)

Bridging different eDelivery systems requires that evidences are properly managed and translated by the gateways. The management of Evidences at the gateways will make it possible for a sender in one eDelivery realm to receive proper confirmation that a message has been routed to a recipient in a different realm.

In order to avoid a non-scalable one-to-one translation of evidences, SPOCS' approach has been to define a “lingua franca” which provides a complete set of evidences to act as a common language between different systems. On the basis of the analysis performed in D3.1, REM-MD evidences as provided by ETSI TS 102 640, part 2 [12], providing a complete set of evidences where chosen.

This choice is also reinforced by the fact that UPU[19] , in its work-in-progress S.52 standard, is building on a REM set of evidences.

2.4 eSafe

There are several options on how to integrate eSafes according to the SPOCS scenario and how to achieve interoperability between the existing systems of the Member States. But since there are no solutions in place that are productive in an international context today a common gateway infrastructure is not needed. A brief description of the solutions of the piloting countries is provided in the Appendix – eSafe solutions to be integrated with SPOCS (Appendix F).

2.4.1 Interoperability building blocks

To provide interoperability between the different components and parties in exchanging documents (in the context of SPOCS) that are stored and genuinely managed inside an eSafe, the following architecture has been chosen:

- Establishment of trust based on the trust model described in section 2.1.
  For trusted interactions between PSCs and eSafe SPOCS related TSLs are used. Each PSC and eSafe SHOULD be listed (proper *service information extensions* are assumed) with at least
    - the SSL public key certificate
    - the URI pointing to SPOCS related service access points
    - the country to which the system is related with.
    - the supported document exchange principles (PUSH, PULL- detailed descriptions will be provided later).

  This information than will be used for
  - lookup of trusted eSafes, appropriate for the SP’s origin
  - lookup of the trusted eSafe’s services needed for document access
  - proof of authenticity when establishing a conversation between the involved systems.

    For this specification it is regardless whether a central TSL or interlinked national TSLs are used. Due to the manageable number of known and expected PSCs and eSafes, this should be feasible.
- Definition of a common interaction model (business process) as described in detail in section 4 “Protocol for retrieving documents from an eSafe” including the following

[19] Universal Postal Union, http://www.upu.int/ (last visited Sep 11, 2010)

actions:

- Definition of an interactive session
  - starting with the SP's request for attaching documents that are stored in an eSafe.
  - spawned over the PSC and the eSafe.
  - ending with the PSC's feedback of a successfully completed or failed data transfer.
- Definition of the UI / control flow spawned over the two systems
- Definition of the data flow spawned over the two systems
- Responsibilities of the PSC
- Responsibilities of the eSafe

- Ensuring data exchangeability by means of the SPOCS specification D2.2 "Standard Document and Validation Common Specifications".
  For interoperability with the documents to be exchanged on the PSC side (and likely also on the side of CA, which is however out of scope of this specification) the documents will be transferred to the PSC as an OCD Container (see SPOCS specification D2.2 "Standard Document and Validation Common Specifications").
- Ensuring secure data access and data transfer based on the security model described in section 2.2 "Security Architecture" and on the SPOCS specification D2.2 "Standard Document and Validation Common Specifications".
  The data transfer SHOULD be ensured in a secured and trusted way, based on well known algorithms and best practices. Additionally, the data's integrity and confidentially CAN be ensured based on the OCD signing and encryption functionalities. Depending on the PSCs capabilities the documents can be encrypted either for the target PSC or for the target CA (in order to gain end-to-end security).
- Ensuring ease of integration by providing open modules.
  In order to make integration easy, WP3 will deliver open modules for the piloting phase. The modules will support managing the control flow, cover the creation of document transfer packages in the form of OCD containers and cover the service oriented communication between the PSC and eSafe.

2.4.2 Interaction model basics (business process)

As noted above, one important element of the eSafe interoperability architecture is a common interaction model between the PSC and the eSafe.

Note: While the whole usage scenario described in subsection 1.8 covers the full range of scenarios where the SP, PSC, eSafe, eDelivery and also CA are involved, this common interaction model for retrieving documents from an eSafe focuses only on the selection of documents in an eSafe and transporting them to the PSC in such a way that the PSC bind them to an application request and forward them to a CA.

A general prerequisite is that the SP has documents available in an eSafe. So they need to be uploaded once before the transaction takes place (**`UploadDocument`**, see Figure 5). The various options about who is granted what in the eSafe depend on the national implementation. However, the eSafe MUST guarantee that the producer is authenticated and authorised in order to upload documents. The communication channel used MUST be secured according the generals security requirements (e.g. via SSL/TLS). The relevant parameters for storing data in the eSafe are the document itself and its metadata. A set of metadata required by SPOCS is described in subsection 4.3.3 "Document Metadata".

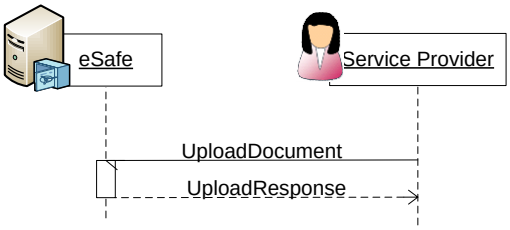

Figure 5: Upload Document to eSafe

Figure 6 shows the general process relevant with an eSafe integration which is assumed and based within SPOCS. The specification relevant interaction model starts with the SP applying at the PSC. To apply online for a service, the SP has to start the (online) application at the PSC. There is a web communication between these communication partners including the exchange of information and finally filling out an application request and - if needed - attaching required documents.

As mentioned above depending on the implementation and the choice of the SP the attachments are either provided by direct transfer of by using eSafe functionality.

If the SP decides to attach documents from an eSafe, the PSC will provide an appropriate list of eSafes (taken from the TSL, probably filtered by some criteria like “country”, “signature supported”) of which the SP can choose from (**`eSafe Lookup`**).

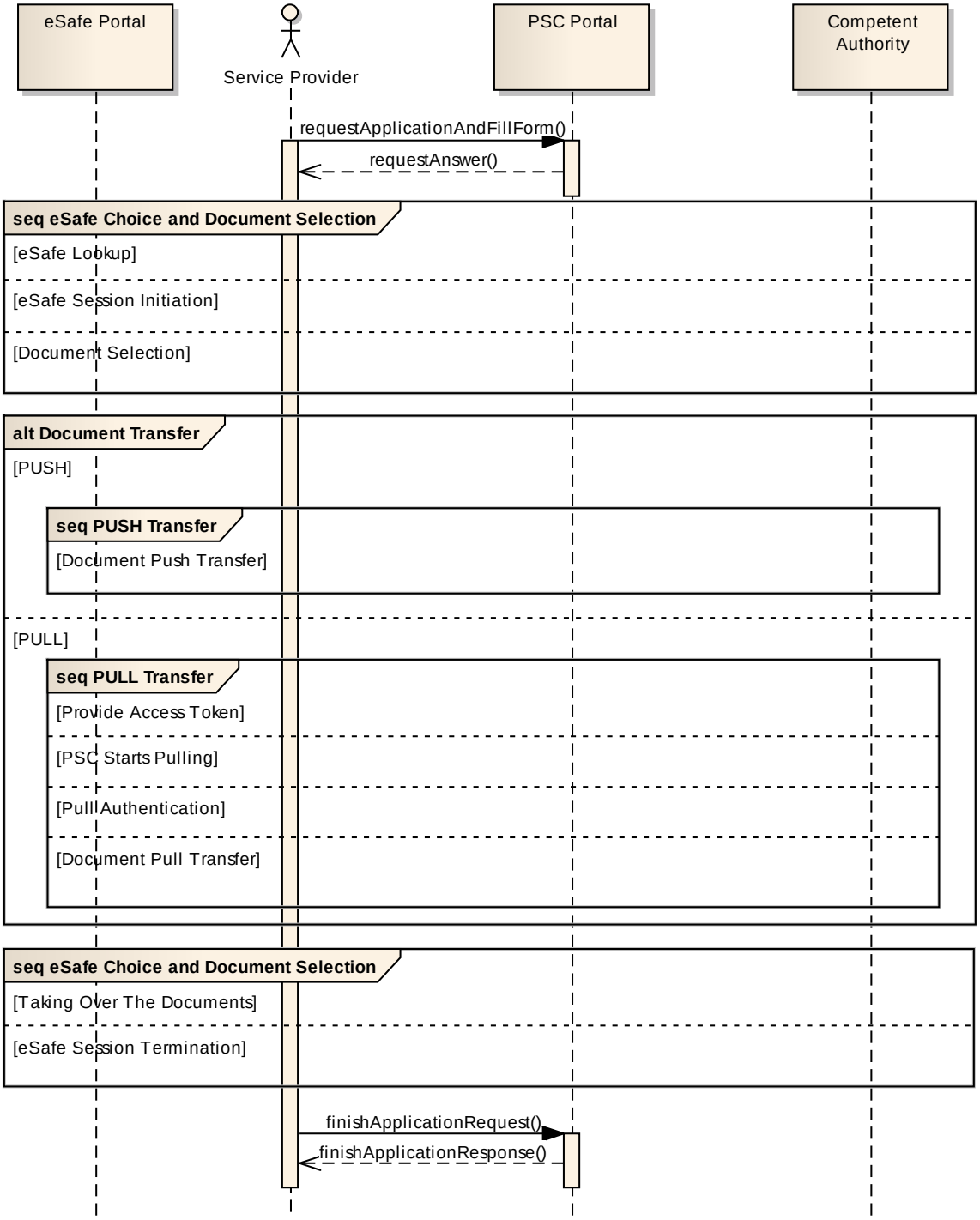

Figure 6: Attaching documents to an application request (overview)

The SP picks the one of interest. As a result, the PSC and the chosen eSafe establish a *document exchange session* via Web service calls. They implement a handshake mechanism at session initiation (**`eSafe Session Initiation`**) where they

- mutually authenticate each other,
- select various document transfer options based on each other's capabilities (e.g. transfer principle to be followed, algorithms for encryptions and signing) and
- exchange a dictionary of URLs implementing specific process steps.

After that the SP is redirected to the eSafe. The target URL with its required parameters (e.g. indicating the session) has been provided by the eSafe with the initial handshake at session initiation.

The SP needs to logon at the eSafe. After that the SP selects the relevant documents for the transfer (**`Document Selection`**). Once this is finished, the eSafe is requested to initiate the document transfer.

This protocol considers two operation modes (principles):

- PUSH principle (RECOMMENDED)
  Following the PUSH principle the eSafe pushes the selected documents directly to the PSC using a set of Web services[20] described later in this document in detail (**`Document Push Transfer`**). Anyhow, each service call refers to the document exchange session that was initiated at the beginning. So the PSC is able to assign the documents to the right application request.

  As long as the transfer is in progress the SP remains at the eSafe side. The eSafe is responsible for visualising the ongoing document transfer, which can take some time if the documents are of respectable size.

  Having transmitted all documents successfully, the eSafe redirects the SP back to the PSC. The target URL with its required parameters has been provided by the PSC with the initial handshake.

  See subsection 4.5.2 “Retrieving documents following the PUSH principle” for further details.

  Note: The PUSH principle is the recommended principle in S<POCS since it is easier, less complex and allows a better and more controllable management. Furthermore the security risks are lower and decrease the potential damage to a single document since no external entity is accessing someone’s eSafe area.

- PULL principle
  Following the PULL principle the eSafe allows the PSC to pull the selected documents from the eSafe. During the document selection a secure access token is issued. Then, this access token, needed for unlocking the selected documents, is encrypted for the target PSC and either provided to the PSC directly or to the Service Provider through an independent channel (**`Provide Access Token`**). In fact, the following strategies are applicable:
  1. Provide access token via service (RECOMMENDED)
     The encrypted access token is sent to the PSC by a Web service.
  2. Provide access token via redirect
     The encrypted access token is provided as a parameter of the subsequent redirection.

[20] In future eDelivery mechanisms may be used as an alternative.

3. Provide access token manually
   The SP will need to type in the encrypted access token. The token may be transported by email or using another transport channel, or maybe simply displayed in a popup page (to copy and paste), depending on the national implementation.

In any case, the control is then returned to the PSC, who is further responsible for the residual control flow and tries to retrieve the documents (**`PSC Starts Pulling`**). The target URL with its required parameters was provided by the PSC during the initial handshake. If the encrypted access token is not yet known by the PSC (see PULL, strategy 3, “Provide access token manually”), the SP is asked to enter it.

Once the token is available, it will now be decrypted first. Then the eSafe processes the authentication of the PSC with the decrypted token and unlocks the document selection (**`Pull Authentication`**).

After that the PSC pulls the documents using a set of Web services (which are described later in this document in detail, see **`Document Pull Transfer`**). Anyhow, each service call refers to the *document exchange session* that has been initiated at the beginning.

As long as the transfer is in progress the PSC is responsible for visualising the ongoing document transfer, which can take some time if the documents are of respectable size. See subsection 4.5.3 “Retrieving documents following the PULL principle” for more details.

Note: If the PULL principle is implemented, the eSafe must implement fine grained and auditable access control and authentication mechanisms as well as additional auditing functionality. If the implementation is deficient the content of an user or even the whole eSafe could be compromised. Overall, the PUSH principle provides a more robust specification here.

When the document transfer is finished, the PSC checks the document transfer’s success and the documents’ transfer package integrity (if the document transfer package was secured with a digital signature). If everything is ok the PSC attaches the documents to the application request. Finally the PSC provides an appropriate feedback to the SP (**`Taking Over the Documents`**). Thereby the SP might also have the chance to verify the documents at the PSC.

Once, everything is done, the PSC closes the session (**`eSafe Session Termination`**).

Note: For a more detailed description see subsection 4.5 “Operations for retrieving documents from an eSafe”.

From the SP’s point of view the two principles do not differ much. At the end of the document transfer the PSC tells the SP that the documents have been attached to the application request and proceeds with the process started with the initial application request.

The protocol is open for extensions in future versions. One extension option is an additional operation mode in that the eSafe transports the selected documents using the eDelivery channel (therefore eDelivery must be available for the PSC in order to receive documents and for the SP to send documents). However, this scenario is not part of the current specification.

The communication between the SP and the PSC as well as the communication between the SP and the eSafe MUST be realised in the national infrastructure. The

communication between the PSC and the eSafe following this specification CAN be completely realised by the involved systems themselves. Alternatively, the realisation can be done based on open modules that will be provided as part of the SPOCS project's deliverables (see section 4.7).

2.4.3 Base protocol stack

The common interaction model for retrieving documents from an eSafe is based on the following protocol stacks:

- Web protocols for the UI and control flow (mainly HTTP and SSL/TLS)
- Web service (WS-*) protocols (mainly SOAP 1.2, WSDL) including the underlying Web Standards (mainly HTTP and SSL/TLS) and XML-Standards (XML, XML Schema, XML Encryption etc.) for the data flow

All HTTP-based communication between SP, PSC and eSafe SHOULD take place over communication channels that are secured by SSL/TLS.

As already described for eDelivery

- Web- and WS-Standards are standardised on the international level, well adopted and established by the IT industry
- WS-* offers dedicated parts for the different feature classes (security and trust, reliable delivery etc.), combinable according concrete requirements to be served
- The chosen protocol stacks are flexible and allow standardised, machine consumable profiling mechanisms
- Many implementations of these standards are available, many of them are open-source-software
- Strong interoperability efforts are driven by implementers (e.g. ws-i.org[21], WSIT/ project Tango[22]).

The base protocol stack and standards according the OCD Container are listed in the respective specification parts. Trust-service Status Lists are covered in section 2.1 and in Appendix 2: Trust-service Status List Profiling ("SPOCS TSL").

[21] http://www.ws-i.org (last visited on 22/08/2010)

[22] https://wsit.dev.java.net/ (last visited on 22/08/2010)

3 eDelivery functionality and architecture

The principal Gateway-approach outlined in section 2.3 is detailed on functional level in this section for the Dispatch and Evidence Message flow (section 3.1), rough message format and security mechanisms (sections 3.2 and 3.4) as well as the cross-MD/Realm addressing concept (section 3.3).

3.1 Cross border/solution eDelivery message flow

To make the more general eDelivery architecture outlined in Figure 4 more concrete, the sequence diagrams showing the message exchanges between PSC and SP are provided in Figure 7 below.

Both PSC and SP can be in the role of a Sender or Recipient; for simplification, only one Dispatch direction in shown, which in the SPOCS scenario could be a delivery of additional documents from SP to PSC as well as the delivery of the application process result from PSC to SP. Even CAs may act in the Sender/Recipient roles - a scenario which is out of scope for SPOCS, anyway principally covered by the solution presented here.

REM specifications provide a list of Evidence types, to be generated on occurrence of delivery events according to the model described in TS 102 640, part 1.

List of Evidences to be supported between SPOCS eDelivery Gateways (GW):

	REM Evidence	**Shortname in** Figure 7
1	SubmissionAcceptanceRejection	SubmissionAcRe
2	RelayREMMDAcceptanceRejection	RelayAcRe
3	RelayREMMDFailure[23]	RelayFailure
4	DeliveryNonDeliveryToRecipient	(Non)Delivery
5	RetrievalNonRetrievalByRecipient[24]	(Non)Retrieval
6	AcceptanceRejectionByRecipeint[24]	AcReByRecipient
7	ReceivedByNonREMSystem[23]	RecByNonREM

Table 2; Supported REM Evidences

In general, details of generation and validation of Evidences the specification ETSI TS 102 640 applies. Evidence generation and delivery in the general message flow is shown in the sequence diagrams below.

While Evidences generated by and exchanged between Gateways are of format and content as defined by ETSI TS 102 640, Part 2, the evidences generated or consumed

[23] Not yet considered in this version

[24] Optional – only to be provided, if domestic MD is able to signal this type of Evidence

by the Sender's respective Recipient's MD in general are of domestic format of respective MD, thus to be seen as "logical" ones in the figures below.

Those Evidences which may not or only conditionally be provided by certain MDs are marked by dotted flow arrows.

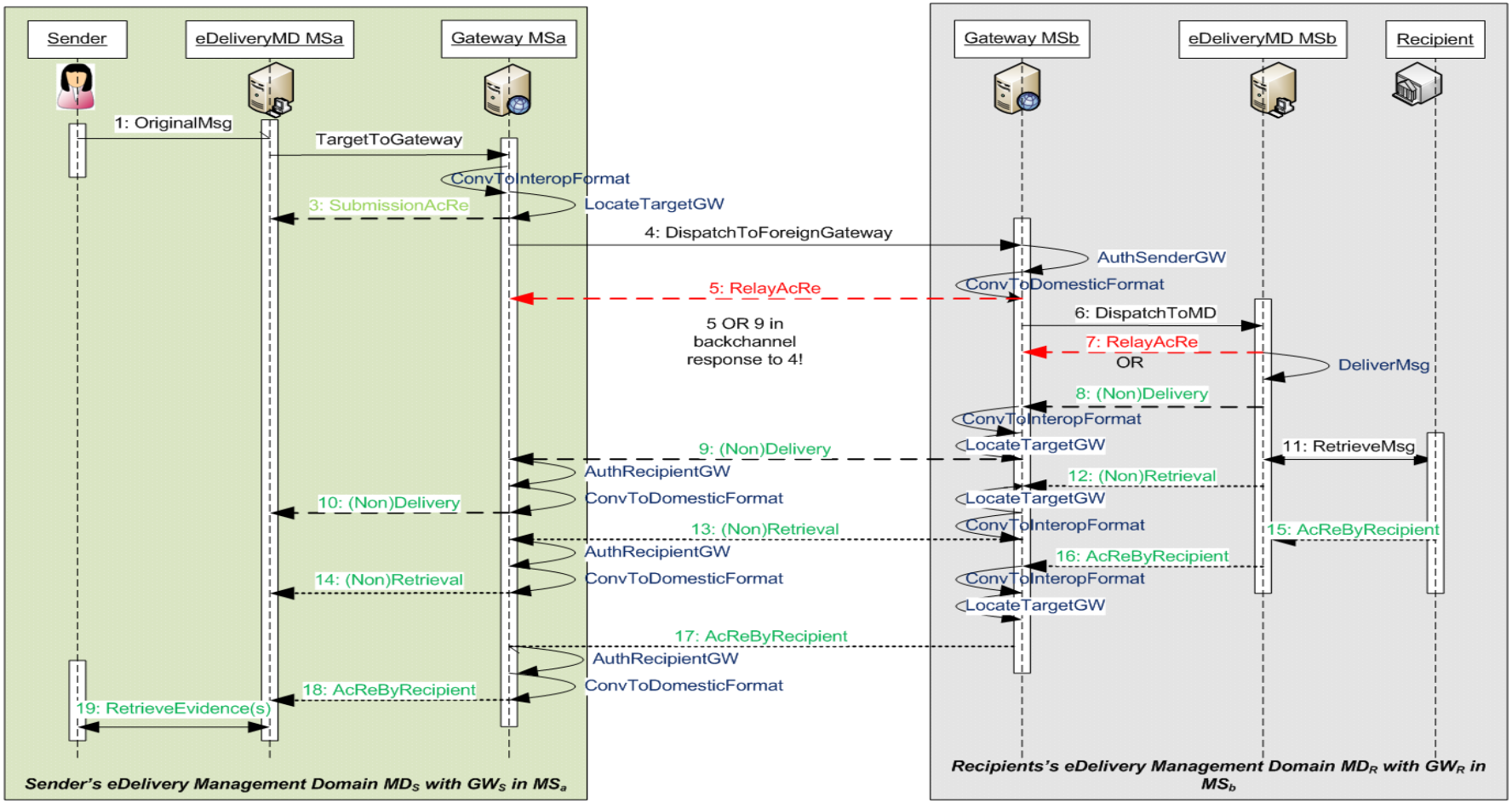

Figure 7: eDelivery message flow overview

Description of the sequence diagram in Figure 7:

(1) **`OriginalMsg`**

Sender initiates message to the intended Recipient, which is targeted to MD_{SE}. **`OriginalMsg`** requires forgoing authentication of SE according to domestic functionality. Assumption: Functionality is given by existing solution.

MD_{SE} must be able to identify, if recipients e-address is outside home MD/Realm. This functionality must be provided by MD solution!

To attest Dispatch submission details for the Recipient, MD_{SE} may add an according Evidence to the Dispatch (for detailed description see Appendix 3: eDelivery Interconnect Protocol and Gateway Specification).

(2) **`TargetToGateway`**

If **(1)** results to external target, message is given to GW in MSa where MD_{SE} is related to. This functionality must be provided by the MD solution! Authentication of MD_{SE} at GW is a matter of this MD's /Realms policy; but may even be resolved by means of the SPOCS TSL.

`ConvToInteropFormat`: The GW MSa performs outbound protocol format conversion. This functionality must be provided by specific GW implementation, by using open modules to be provided by WP3 as specified in present document!

`LocateTargetGW`: GW MSa locates target GW MSb appropriate for recipients MD_{RE}, using the according TSL entry. Further Dispatch completion is done according TSL entry attributes; in addition, Sender's authentication attestation (SAML token) is included in the Dispatch. Functionality will be provided by WP3 open modules. For the SAML token, a specific stub is provided for mapping the domestic authentication mechanisms or token to the interoperability format one.

(3) **`SubmissionAcRe`**

GW MSa delivers an Evidence of success (or fault) to MD MSa. Evidences always are generated as REM-Evidences according to ETSI TS 102 640 and must be converted to the MD's domestic format, if REM-format not understood by this MD.

(4) **`DispatchToForeignGateway`**

The Evidence generated in **(3)** in added to the Dispatch; finally, GW MSa digitally signs the converted Dispatch before targeting it to GW MSb. SSL/TLS is used to ensure confidentiality and mutual GW authentication.

On the destination GW MSb side, the inbound Dispatch is checked, including signature validation. The source GW must have a valid entry / digital identity in according TSL entry (**`AuthSenderGW`**). **`ConvToDomesticFormat:`** GW MSb performs inbound protocol format conversion, including the optional submission Evidence provided by GW MSa. This functionality must be provided by specific GW implementation, by using stubs of the generic GW implementation.

(5) **`RelayAcRe`**

Evidences (5) and (7) are generated only in the case GW MSb delivers the incoming Dispatch in an asynchronous manner to MD_{RE}; if this delivery can be processed synchronously, Evidence (9), generated on base of (8), is given in the network backchannel of (4).

GW MSb delivers an Evidence (REM-formatted) of success (or fault) to GW MSa. **`AuthRecipientGW:`** Authentication of the Evidence at GW MSa is analog as for **`AuthSenderGW`** above. It's left to the concrete GW implementation, how to deal with this Evidence[25].

(6) **`DispatchToMD`**

The GW MSb targets Dispatch – converted to in MD_{RE}'s domestic format - to Recipient's MD_{RE}. This functionality must be provided by specific GW implementation, using the according local functionality. Authentication of sending GW MSb at MD_{RE} is a matter of the MD's/Realm's policy; but may even be resolved by means of the SPOCS TSL.

(7) **`RelayAcRe`**

If Evidence (8) can not be given synchronously instead, MD_{RE} may deliver a (non)acceptance Evidence (in domestic format) to GW MSb, to be used for message flow control by the local GW implementation part. **`DeliverMsg`**: MD_{SE} stores Dispatch in message box of RE. Assumption: Functionality is given by existing solution.

(8) **`(Non)Delivery`**

MD_{RE} may deliver an Evidence of delivery (in domestic format) to GW MSb, depending on functionality provided here. On base of **(7)** and/or **(8)**, GW MSb must produce an Evidence of successful/unsuccessful delivery in REM-format (**`ConvToInteropFormat)`**, to be send back to the GW MSa (**`LocateTargetGW`**) the Sender's MD_{SE} it relates to.

(9) **`(Non)Delivery`**

At GW MSa, the Evidence of delivery MUST be authenticated (**`AuthRecipientGW)`** and converted to the MD_{SE}'s domestic format (**`ConvToDomesticFormat)`**, if REM-format not understood by this MD.

(10) **`(Non)Delivery`**

GW MSa targets the converted Evidence to MD_{SE}, to be delivered to the Sender's message box.

[25] At least, the Sender MUST be informed about the occurrence of a fault Evidence!

(11) **`RetrieveMsg`**

The Recipient is accessing the Dispatch or pulling it out of his message box. **`RetrieveMsg`** requires foregoing authentication of Recipient according to domestic functionality. Assumption: Functionality is given by existing solution.

(12) **to (14):** **`(Non)Retrieval`**

Analogous as for (8) through (10) for Evidence on delivery: MD_{SE} may generate an Evidence on retrieval of the Dispatch through the Recipient, to be delivered back to the Sender's message box.

(15) **to (18):** **`AcReByRecipient`**

Analogous as for (8) through (10) for Evidence on delivery: MD_{SE} may generate an Evidence on retrieval of the Dispatch through the Recipient, to be delivered back to the Sender's message box.

(19) **`RetrieveEvidence(s)`**

The Sender is accessing the Evidence(s) or pulling it (them) out of his message box. **`RetrieveEvidence(s)`** requires foregoing authentication of Recipient according to domestic functionality. Assumption: Functionality is given by existing solution.

3.2 Interoperability layer message structure

As mentioned in section 2.3.3, the message format used between Gateways is defined on base of SOAP 1.2 and WS-* SOAP header components. A specific SOAP body structure is defined to cover the specific needs of this interconnectivity framework.

Data needed for message routing, -authentication and -security between Gateways is placed in SOAP header blocks.

Message payload – either Dispatch or Evidence - is placed in the SOAP body. Gateways are seen as Web Service endpoints generating or consuming the SOAP body. Dispatches targeted to a Gateway from the source MD in the MD's domestic format must be converted to a normalized format as defined in this specification, to be available at the destination Gateway for re-conversion to domestic format used in this eDelivery realm. This applies for Evidences as well.

In general, an eDelivery infrastructure should be payload-agnostic, but when crossing protocol boundaries the principle of an "opaque body" in most cases would result in lack of interoperability.

A generic payload structuring in presented here, addressing the more or less unbound communication, like commonly used for exchange of (probably somewhat qualified) e-documents (usually as message attachments), accomplished by covering notes.

The use of this format has following prerequisites and/or impacts:

- Dispatches encrypted as a whole for the Recipient cannot be structured according the format below; they can only be carried in binary format "as is".
- Dispatches signed as a whole, but not encrypted, can be restructured - by loss of the signature validity. In this case, the original message SHOULD be carried as additional copy in a binary SOAP body container.
- To avoid complications arising when mapping cryptographically secured data,

message parts SHOULD be secured (signed, encrypted) separately. But, this still has the impact applications used by the recipient MUST deal at least both with the CMS and XML Signature/-Encryption worlds[26].

- Message metadata mapping requires Gateway access to these message parts. The according contents SHOULD be somewhat insignificant regarding data protection and confidentiality issues.

Dispatches in the source MD domestic format MUST be included in the SOAP body in addition to the normalized format, too, if intended (or even required) to be made available to the recipient's MD.

Figure 8 illustrates the general message layout; for brevity, WS ReliableMessaging headers and the case of transferring a SOAP fault in the network backchannel is not covered.

[26] This challenge is not in addressed by WP3 tasks. At least, e-signature and e-Id validation should e solvable by using the PEPPOL WP1 and STORK outcomes.

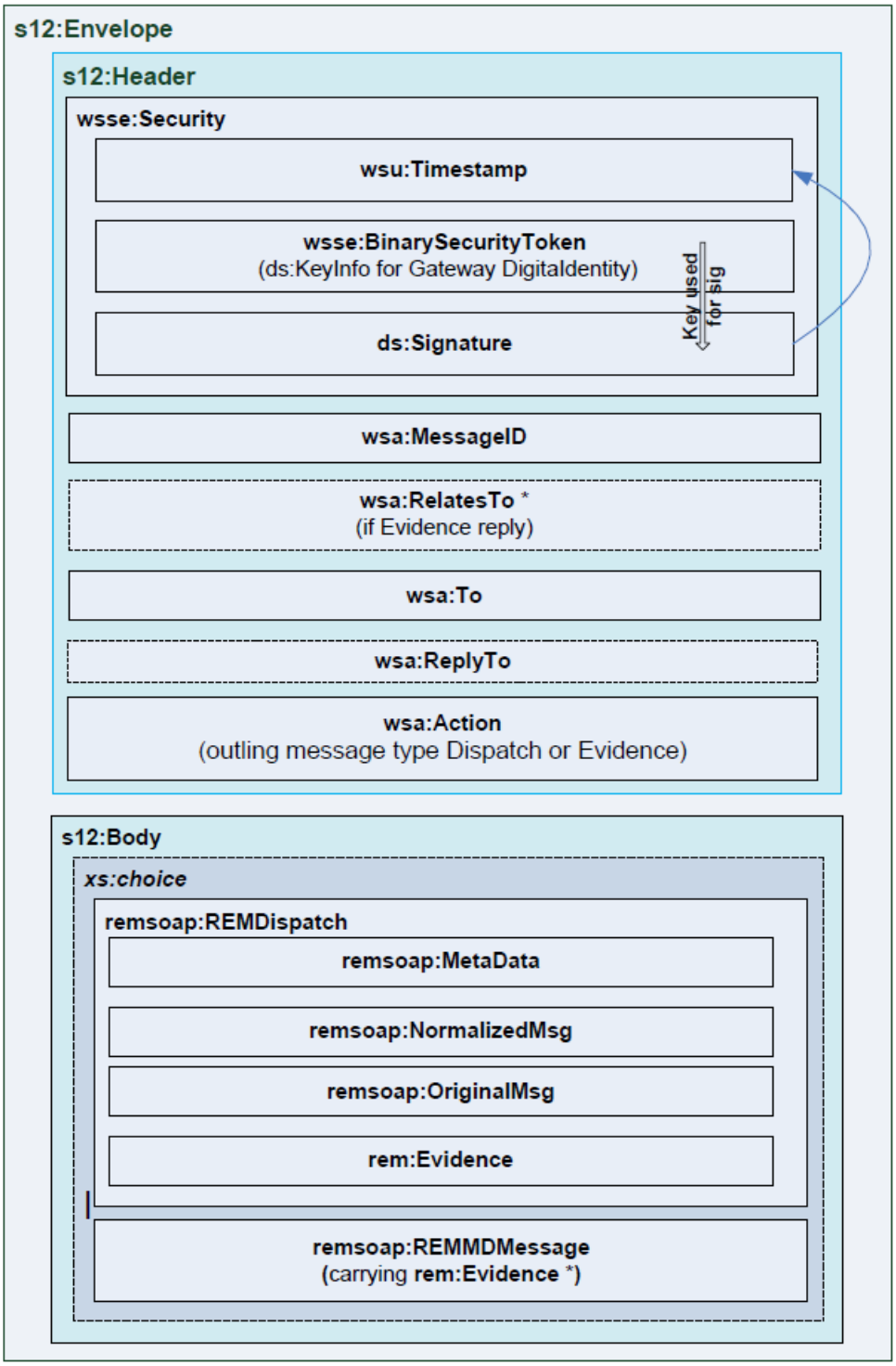

Figure 8 High level message structure

Attachments in any format can be part of a Dispatch. On the wire, a SOAP message is serialized as MIME Multipart/Related message. For transparent handling and optimized transport of attachments the W3C specifications "SOAP Message Transmission Optimisation Mechanism (MTOM)" [30] and "XML-binary Optimized Packaging (XOP)" [31] are incorporated without restrictions. These mechanisms are provided by commonly available SOAP implementations and thus can transparently be used.

High level description of the SOAP message blocks shown in Figure 8[27]:

WS-Security header block[28]

This header block carries a security timestamp **`wsu:Timestamp`**, a reference **`wsse:SecurityTokenReference`** to the signing certificate of the message source GW pointing to according **`tsl:DigitalIdentity`**, which is used for the digital signature over the **`remsoap:REMDispatch`** or **`remsoap:REMMDMessage`** held in the SOAP body, a such serving for the SOAP message source GW authentication, too.

WS-Addressing header blocks

This header blocks are only used for addressing, routing and message correlation between GWs. They MUST NOT be confused with the according ones of the original message payload carried in the body (see below). **`wsa:Action`** denotes, whether the message is a Dispatch or Evidence (the latter one MUST NOT be receipted by an Evidence again).

SOAP body

These elements are built up by the Realm's outbound GWs on base the according domestic source message, and, by the destination realm GW, converted the destination Realm's domestic format. According to the **`wsa:Action`** value, the body carries either a **`remsoap:REMDispatch`** or a **`remsoap:REMMDMessage`** complex element.

The detailed definition is given by TS 102 640, part REM-MD SOAP Binding [46], sketched here to give on overview:

`/remsoap:REMDispatch`

> Container carrying the Dispatch in the original format (optional) and in a "normalized" format, used to facilitate cross-conversion between different formats.

`/remsoap:REMDispatch/remsoap:MetaData`

> Contains metadata related to transport outside the Gateway-to-Gateway communication line. Besides attributes needed for transport functionalities, most of the eDelivery solutions provide some meta-information about the message payload[29] . The sub-elements are built up by the Realm's outbound GWs on base the according domestic source message metadata and, by the destination Realm GW, converted to metadata format defined here.

`/remsoap:REMDispatch/remsoap:NormalizedMsg`

> Container carrying the "non-meta" Dispatch items like message body, attachments and either describing elements, converted to an XML-format called "normalized".

---

[27] For detailed specification, see Appendix 3: eDelivery Interconnect Protocol and Gateway Specification. In Section 7 of this Appendix an overview of the eDelivery solutions in place in the Piloting Countries is given.

[28] See section 3.4 for further details.

[29] This may not be the case for all solutions, as seen as subject to data protection issues and thus only contained in the message payload, which might be encrypted for the recipient.

/remsoap:REMDispatch/remsoap:OrginalMsg ?

Element to include the Dispatch in the untouched, original format; encoded base64Binary

/remsoap:REMDispatch/rem:SubmissionAcceptanceRejection

A **rem:SubmissionAcceptanceRejection** Evidence MUST be included in the REMDispatchIt attests the time the source GW took charge of this Dispatch as well as the Sender's authenticity. If the Dispatch initially was received form a non-REM conformant MD (like e.g. regular e-mail), a **rem:ReceivedByNonREMSystem** Evidence may be included, too.

A REMDispatch MUST carry a SAML [15][17] assertion for the authenticity attestation of the Sender of this message. As the source Gateway is in charge to issue this token, the method used MUST be "sender vouches". The SAML assertion always MUST be included in the element **rem:SenderAuthenticationDetails** of the evidence **rem:SubmissionAcceptanceRejection.**

Note on support of non-REM conformant MDs:

It is strongly advised not the support such scenarios, as this will produce wholes in the eDelivery trust chain and delivery control flow. Gateways SHOULD not accept such Dispatches (but, at least, matter of decision of concrete Realm).

/remsoap:REMMDmessage

Container carrying a list of **rem:Evidence** type elements according to ETSI TS 102 640. Those elements are signed separately, again using the issuing GW's signing certificate mentioned above.

Following list of evidences will be supported between Gateways:

- SubmissionAcceptanceRejection
- RelayToREMMDAcceptanceRejection
- RelayToREMMDFailure
- DeliveryNonDeliveryToRecipient
- RetrievalNonRetrievalByRecipient[30]
- AcceptanceRejectionByRecipient[30]
- ReceivedByNonREMSystem.

Above evidences use **tsl:ElectronicAddress** as type for e-address elements. As this is not sufficient to cover the SPOCS eDelivery cross MD/Realm requirements (see section 3.3), in addition an element **rem:ReplyToAddress** is included in a **rem:EvidenceType by extension**, which conforms to the extended e-address structure used here.

A **remsoap:REMMDMessage** MAY include the **remsoap:REMDispatch** message the Evidence list is related to.

SPOCS Interconnect defines few restrictions to the scheme specified by TS 102 640 REM-MD SOAP-Binding:

- [30] Only if Event can be provided by domestic solution

- A **`remsoap:REMMDMessage`** SHALL always contain only one Evidence, and the types of possible Evidences are restricted.
- A **`remsoap:REMDispatch`** SHALL always contain an Evidence **`rem:SubmissionsAcceptanceRejection`**, as the inner SAML-Token attesting Sender's authenticity SHALL always be provided to the target Gateway.

3.3 Cross eDelivery Management Domain/Realm addressing

The SPOCS cross-MD/realm addressing and routing concept tries to solve following challenges:

(1) Regardless of the technical addressing format used in the recipient's realm, a sender should be able to use an intuitive common format for addressing the intended recipient.

(2) The addressing scheme used must be able to support different formats and must be open for future extensions.

(3) Zero knowledge is required from the end entities sender/recipient about the infrastructure used for message delivery. Correct routing from the sender's realm to the one of the recipient must be able on base of the recipients e-address entered by the sender.

Thus, for each e-address format used for SPOCS eDelivery, it is REQUIRED that the information about the ultimate point a message must be routed to ("destination MD") can be extracted from the e-address (e.g. domain-part of an e-address).

Source MDs MUST be enabled to decide on base of the destination MD, whether a message can be delivered directly (e.g. destination MD is in same realm) or if the message must be passed to a GW for delivery to foreign realms including required protocol conversion.

In the outbound case, the GW performs a SPOCS TSL lookup using the destination MD information. In case of success, the result will be the TSL entry(ies) of the GW(s) in the destination MD it relates to. After conversion to the interconnect format and application of message security related procedure/information, the message is targeted to (one of) the address(es) outlined in the **`tsl:ServiceSupplyPoint`** element of the destination GW(s) TSL entry(ies).

End entity e-addresses contained in the initial message are carried in according elements in **`remsoap:Metadata`** described in [46]. As for the whole message, the destination GW converts (if required) the e-addresses to the corresponding fields and formats of the domestic protocol, before targeting the message to the ultimate MD as pointed to by the recipients e-address.

If a Dispatch is addressed to several recipients (either in the "To:" or "Cc:" role), one copy of the message MUST be build up for every recipient. This splitting functionality is not defined for the generic parts of the Gateway, only one recipient is supported for a message (see [46]). "Bcc:" roles are not supported in this version of the Interconnect Protocol.

To be able to support different formats of e-addresses, TS 102 640 Part 2 [9] defines a complex **`remsoap:AttributedElectronicAddressType`**, able to carry the e-address value itself and an attribute denoting the e-address scheme used. Optionally, an attribute **`@DisplayName`** can be provided, semantics as known from the e-mail world.

GWs MUST express in their TSL-entry the supported address-scheme used on the interoperability layer. A source GW must decide, if the e-address conversion implemented here can deal with the address-scheme supported by the target GW.

It is left to the concrete GW implementation, if and how conversion is done to the domestic e-address format. Knowledge of the e-address schemes used in foreign Realms is not needed.

Note: Handshake of Gateways concerning e-address scheme

Source Gateways MUST verify whether one of the e-address schemes exposed in the according TSL entry of the destination Gateway is understood and according e-address conversion can be performed; otherwise an Evidence of non acceptance MUST be returned to the source MD.

3.3.1.1 Default scheme for e-adresses

The default end entity e-address scheme for SPOCS communication is the e-mail address format, as specified by RFC 5322 [32], section 3.4. "Address Specification".

- E-mail addresses (used for example in the Italian PEC[31]) can be used without any modification and traditional e-mail clients can even be used for SPOCS communication;
- Web-service addresses (as used e.g. in the German OSCI infrastructure [25]) can be easily translated in an e-mail address using the same domain-name and using the file part as local part (changing, of course, the separator); so an address like **`http://osciprovider.bremen.de/recipient-id`** results to **`recipient-id@osciprovider.bremen.de`**;
- Other addressing information sets (like the Austrian ones) can be converted to an e-mail address concatenating the single elements; for example to address a person using given-name, family-name and date of birth, the whole information can be coded in **`given.family.ddmmyyyy@spocs.austria.at`**.

The value for scheme of this type of e-address is defined as "mailto" (see [9]) **.**

Requirements for existing eDelivery solutions:

- If the address format compliant to RFC 5322 is not supported by eDelivery user agent (client) applications in use, according supporting features must be provided.
- End entities (Service Provider and Point of Single Contact) must expose their e-address in the mentioned format.

Note for Gateways supporting this e-address format:

If e-addresses in this format differ from the one used inside the MD/realm of their GW, according conversion MUST be done for all domestic e-address fields in the outbound and inbound case.

3.3.1.2 Other e-address schemes

Other schemes MAY be introduced by concrete GW implementations, which is sketched here for a GW implementation to a BusDox Access Points (AP).

[31] Posta elettronica certificata, www.pec.it

To be able to exchange messages with participants of the BusDox network in establishment by PEPPOL WP8, e-addresses as defined here by the "Universal Participant Identifier Scheme" (see BusDox [11], "Common Definitions" MAY be supported, too, by according GW implementations to BusDox Access Points (AP).

The XML format for a **`/ids:ParticipantidentifierType`** is defined as attributed xs:string, where the attribute **`@ids:scheme`** (format: **`xs:string`**) denotes the identifier scheme as "busdox-actorid-upis"; **`ids:ParticipantidentifierType`** carries two values separated by a colon in the form **`<IdentifierType>:<Identifier>`**[32]. This matches to the according **`remsoap:AttributedElectronicAddress/@scheme`** attribute and its non-empty **`xs:anyURI`** element value.

The value for a scheme of this type of e-address is defined as "busdox-actoris-upis".

3.4 Security mechanisms and related message elements

To provide secure communication between the eDelivery Gateways that constitute parts of the SPOCS interoperability framework WS-Security with underlying SSL/TLS message signature and encryption MUST be used. In particular, all messages between Gateways must obey to the OASIS SOAP Message Security 1.1 as known as WS-Security 2004 [6].

Additional message signing will be performed using X509v3 certificates and the accompanying Web Services Security X.509 Certificate Token Profile 1.1. This signature and used certificates serves for the authentication of the message origin Gateway. This requires that all nodes implement the SPOCS-TSL in order to evaluate trusted certificate authorities and trusted service nodes within the SPOCS interoperability framework. Please not that this signature is only used for integrity protection between peers of the SPOCS interoperability framework. It is deliberately not intended to provide end-2-end security.

All SOAP message body parts exchanged between Gateways must be signed using **`ds:Signature`** as enveloped signature.

To enforce a consistent security policy among all nodes **`/wsse:Security@S12:mustUnderstand`** must be set to “true” or 1.

Timestamps must be signed on application level. In order to prevent replay attacks nodes must set **`/wsu:Timestamp/wsu:Created`** and **`/wsu:Timestamp/wsu:Expires`**. The interval between Created and Expires shall not exceed 300s. Nodes MUST implement a message Id that is unique at least within the period between **`/wsu:Created`** and **`/wsu:Expires`**. Nodes SHOULD cache message Ids for a minimum of 300s and generate errors messages if a replay is detected.

A Gateway may produce signature confirmation using **`/wsse:SignatureConfirmation`**.

A source GW uses the sender-vouches confirmation method to assert that it is acting on behalf of a subject using SAML statements attributed with a sender-vouches **`SubjectConfirmation`** element. This makes sure that a remsoap:REMDispatch delivered via the SPOCS interoperability framework originates from an authenticated local subject. As stated above, this token MUST be inserted in the **`s12:Body`** inside the **`rem:Evidence`** type structures.

---

[32] Possible values for Identifier type see BusDox specification [11]

3.5 eDelivery requirements on open modules

The WP3 eDelivery solution must support all the necessary mechanisms in the interaction between the sender, receiver and nodes involved in the Dispatch and Evidence transport as described in this specification.

The present concept relies on the Gateway approach; Gateways are the core modules to establish interoperability and security between the existing eDelivery solutions; this concept tries to minimize the need of supporting enhancements for those solutions.

Trust establishment between system instances of eDelivery MDs/Realms as well as Gateways will be done by the concept of SPOCS-TSL. WP3 will provide the according modules for SPOCS-TSL maintenance and distribution.

Obviously, enabling existing eDelivery solutions for the interconnection framework main efforts lie in the concrete Gateway specification and –implementation. WP3 will provide

(1) all those function modules, which must be part of every solution specific gateway implementation

   a. for routing functionally, this is the e-address and TSL handler

   b. inter-Gateway Evidence generator/controller

   c. SAML sender-vouches token builder and validator

   d. Gateway authenticator (validating X509v3 certificate used for transport signatures against according SPOCS-TSL-entry (digital identity of Gateway instance)

(2) the interface implementation for the Gateway-to-Gateway transport way connection, based on the a standard OSS Java implementation of the Web services protocol stack; this includes the message security functionality, based on WS-Security

(3) API for the connection to domestic MD/Realm systems, providing

   a. in core an assembly of getter/setter methods for the constituents of the interoperability layer message parts

   b. access to results of security token validation done by the generic Gateway part

   c. a set of interfaces for exchange of complete messages of type Dispatch and Evidence in the interoperability layer format

A WSDL and the schema files will be provided as zip-files.

4 Protocol for retrieving documents from an eSafe

4.1 eSafe concept and solutions

The eSafe concept is being defined as the architecture to archive, manage and share documents and files. In general an eSafe has to follow some concepts and architectural characteristics such as rules for legal topics and authorised accessibility. Authentication, integrity and confidentiality are primary satisfactory architectural requirements for an eSafe as far it concerns the transactions and privacy of eSafe's users.

An eSafe solution is software that implements the eSafe concept. It may be a standalone software system, offering upload, download and some workflow mechanisms, or it may be part of bigger software system, integrating the eSafe functionality in workflows of overall business functionality.

Service Providers may have been provided with the ability of managing their documents through eSafe interfaces and systems and with the ability of finally sharing them with others in a secure way (e.g. by signing documents, granting access to specific users etc.). Alternatively somebody else (e.g. some public authority or profession in the SP's home country) uses an eSafe system as a source of authenticated documents and the Service Provider is granted access rights to view/retrieve that set of documents related with him or her.

In general, the security requirements may be high or very high. Documents in the eSafe may be stored encrypted. When retrieving documents from the eSafe the document access might be allowed only through secure connections and the documents themselves may be encrypted for the target receiver for providing a higher confidentiality level. Accessing the documents should follow secure authentication and access mechanisms and should be logged. Finally, eSafe users need to go under an appropriate registration (through simple username/password techniques and/or through eID techniques) which is guaranteeing undisputed entrance.

Normally a number of metadata is stored in the eSafe for each document. For instance, in some solutions the metadata could be used for search methods. Storage periods of the documents within the eSafe may also be bound to legal requirements.

4.2 Mapping of eSafe functionality to the interoperability layer

Data exchange, under a secure interoperability network of all the Member States, is one of the main aspects to be considered for SPOCS. The process of a simple transaction between the main actors of an eSafe is introducing two major roles: the document's producer and the document's consumer.

The producer archives the electronic documents and the authorised consumer receives or retrieves these documents by electronic means. A producer of course can play the role of a consumer and vice-versa while the appropriated access and authorisation has been given to each user-role. The actors concerning the eSafe within the SPOCS scenario are mainly the SP and the PSC.

Inside an eSafe system, authentication and authorisation aspects are provided in such a way that the availability, integrity and confidentiality of the stored electronic documents are guaranteed. Furthermore, the eSafe may sign documents electronically. By doing that the eSafe guarantees that the document received is exactly the document stored within the eSafe (authenticity).

The workflow scenarios inside an eSafe system (where all the necessary functionalities like archiving and retrieving electronic documents are implemented under a *secure and trusted* membership of the actors) should offer the following:

1. Store document
2. Select document
3. Transfer document or provide access to document

As already mentioned, two principles of eSafe integration were identified, the PUSH and PULL principle. In the first case, a SP *delivers (pushes)* a selection of documents via his eSafe to a PSC (using a set of Web services or in future maybe using eDelivery functionality). Following the PULL principle, a SP grants a PSC to access a selection of documents inside an eSafe system. The PSC is then able to retrieve these documents when providing the appropriated access information (within this specification the access information is realised with an access token).

4.3 Prerequisites, preconditions and assumptions

Within this section the major general conditions are summarised, which are necessary to enable an interoperable SPOCS scenario with integrated eSafe components.

4.3.1 Trust-service Status Lists

For trusted interactions between PSCs and eSafe SPOCS related TSLs are assumed. Each PSC and eSafe SHOULD be listed with the following attributes:

- Service type identifier
  Attribut in the TSL: Service Information/Service type identifier
- SSL public key certificate
  Attribute in the TSL: Service Information/Service digital identity
- URI pointing to the SPOCS related Document Exchange Info Web service.
  Attribute in the TSL: Service Information/Service supply points
- Country where the system belongs to
  Attribute in the TSL: Scheme extension/Service country code
- Supported document exchange principles (PUSH, PULL)
  Attribute in the TSL: **`Scheme extension/Service working mode`**

By having the systems listed in the TSL

- the PSC is able of searching for trusted eSafes in a specific country
- the eSafe is able to accept requests from trusted PSCs only and
- the PSC is able to accept documents from trusted eSafes only as well.

The general usage is regardless of

- a totally centrally hosed TSL,
- a centrally hosted TSL pointing to child TSLs of each member state or
- distributed TSLs.

Further details on TSLs see section 2.1 as well as Appendix 2: Trust-service Status List Profiling ("SPOCS TSL").

The SPOCS protocol for retrieving documents from an eSafe uses the TSL to establish the initial trust between the communication partners. An additional handshake mechanism at session initiation allows the partners to exchange further information like

additional communication endpoints, additional certificates for document encryption and signing and further document transfer options based on each other's capabilities.

Note: For testing purposes PSCs and eSafes are encouraged to provide SPOCS compliant test Web services (implementing the same protocol but with test URIs and test data only) and either publish them in a test TSL or provide the data in another way for using them in a non-TSL test mode.

4.3.2 Trusted Communication

As a general rule, all communication between the SP, the PSC and the eSafe SHOULD be done via channels secured by SSL/TLS encryption. This includes:

- Web browser based interactions of a SP with a PSC
- Web browser based interactions of a SP with an eSafe
- Any communication between PSC and eSafe

4.3.3 Document Metadata as provided by the eSafe

It is assumed that the eSafe stores some metadata together with the documents saved. The following list shows REQUIRED and RECOMMENDED metadata which will be transported to the PSC together with the documents:

Attribute name	MUST or OPTIONAL	Description	OCD Attribute Mapping[33]
id	MUST	Unique identifier of a document within the eSafe; to make it unique outside the eSafe, some eSafe ID SHOULD be added. e.g. **<eSafeId>-<docId>**	**ocdm:DocumentReference/ cbc:ID**
fileName	MUST	Name of the document (without PATH information)	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:FileName**
mimeCode	MUST	Mime-Type of the document according the RFC for Internet Media Type [2]	**ocdm:DocumentReference/ cbc:MimeCode**
documentVersion	OPTIONAL	Version of the document	**ocdm:DocumentReference/oc dm:DocumentVersion**
description	OPTIONAL	Short description of the document	**ocdm:DocumentReference/oc dm:Description**
formatCode	OPTIONAL	Format of the document, e.g. PDF/A	**ocdm:DocumentReference/ cac:ExternalReference/ cbc:formatCode**
characterSetCode	OPTIONAL	character set of the document e.g. UNICODE-	**ocdm:DocumentReference/ cac:Attachment/**

[33] Documents will be transferred as an OCD Container (see SPOCS specification D2.2 "Standard Document and Validation Common Specifications"). The document's meta data will be provided within the OCD metadata XML file. This column shows which XML type will be used for that.

Attribute name	MUST or OPTIONAL	Description	OCD Attribute Mapping[33]
		16, ANSI	**cac:ExternalReference/ cbc:characterSetCode**
encodingCode	OPTIONAL	character encoding of the document, e.g. UTF-8	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:encodingCode**
issueDate	OPTIONAL	Date of issue of the document	**ocdm:DocumentReference/ cbc:IssueDate**
issueTime	OPTIONAL	Date of issue of the document	**ocdm:DocumentReference/ cbc:IssueTime**
languageId	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:ID**
languageName	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:Name**
localeCode	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:LocaleCode**

Table 3: Document related metadata to be provided by the eSafe

4.4 Main Principles

This protocol considers two operation modes: *PUSH principle* and *PULL principle*, which were introduced in section 2.4.2 already.

4.4.1 PUSH Principle

Following the PUSH principle the document transfer is totally driven by the eSafe. After having selected and confirmed a set of documents to transfer, the eSafe immediately pushes the documents to the PSC using a set of Web services.

This operation mode is RECOMMENDED, due to the following reasons:

- Easier and less complex flow of activities (see section 4.5 “Operations for retrieving documents from an eSafe” and section 4.5.2 “Retrieving documents following the PUSH principle”)
- Supporting a better and more controllable management
- Lower security risks and potential damage to a single document since no external entity is accessing someone’s eSafe area.

When initiating the document transfer the eSafe knows exactly how to address the target receiver, since PSC and eSafe share a common document exchange session throughout the whole conversation.

The PUSH principle is open for extensions in future versions. One extension could be of adding an additional operation mode, where the eSafe pushes the selected documents using the eDelivery channel.

4.4.2 PULL Principle

Following the PULL principle the eSafe, the document transfer is partially driven by the eSafe and partially driven by the PSC. After having selected and confirmed a set of documents to transfer, the eSafe hands over control to the PSC to pull the documents, again using a set of Web services.

This operation mode is second choice, due to the following reasons:

- More complex flow of activities and additional protocol messages (see section 4.5 “Operations for retrieving documents from an eSafe” and section 4.5.3 “Retrieving documents following the PULL principle”)
- Need to provide an extra authentication mechanism to get access to the selected documents in combination with a fine grained access control mechanism with auditing functionality
- Higher security risks and potential damage (even of the whole eSafe system) due to external access to documents

The usage of access tokens requires a strategy for transporting the access token to the target PSC, which is specified below.

Access token transfer strategy

After having selected and confirmed the set of documents to transfer the access token is created and then encrypted for the PSC. Following the PULL principle the following strategies for transporting the access token to the PSC are applicable.

- Provide access token by service (recommended)
  Before handing over control to the PSC (by redirection of the browser to the PSC portal) the encrypted access token is sent to the PSC via a Web service.
- This strategy allows a secure transport to the PSC without involving the SP. The

access token is not visible to the SP's browser.

- Provide access token with redirect
  Control is handed over to the PSC by redirecting the SP to the PSC portal and include the encrypted access token as a parameter of the redirection request.
- This strategy involves the SP's browser in the transport to the PSC, which has a higher security risk than the recommended strategy. However, the transport of the token itself is secured by SSL/TLS and the access token is furthermore encrypted for and therefore only useable by the target PSC.
- Provide access token manually
  Within this strategy the eSafe uses an alternative transport channel to send the access token to the SP (e.g. displaying the token in a pop up page to copy and paste or submission via email, eDelivery or SMS) with further information to the SP of how to proceed34. Then the control is handed over to the PSC by redirecting the SP to the PSC portal, where the encrypted access token has to be entered by the SP.
- This strategy's security risk depends mostly on the chosen transport channel for the access token. However, the access token is also encrypted for and therefore only useable by the target PSC.

4.4.3 UI entry points

The interaction model includes several redirections, either from the PSC to the eSafe or from the eSafe back to the PSC. Since every PSC and every eSafe may implement a different logic in building the pages' URLs, a flexible and system independent solution for defining target URLs is needed.

This protocol considers that the PSC and the eSafe exchange a set of URL templates at the beginning of their conversation (at session initiation). These templates include a set of variables which will be replaced by their current values when used. E.g. the URL

```
https://.../spocs/download/documentsTransferred?session=${pscSessionId}
```

may be transferred to

```
https://.../spocs/download/documentsTransferred?session=ZGFUZFFFFUZ87
```

The following table shows the URLs to be exchanged. All URLs SHOULD be SSL/TLS protected and therefore using the protocol HTTPS.

Owner	UI entry point symbolic name	Description
eSafe	**START_ESAFE_DOCUMENT_SELECTION**	Redirect to this page to let the SP start the document selection.
PSC	**ESAFE_DOCUMENT_SELECTION_CANCELLED**	Redirect to this page, if the SP has cancelled the document selection.
PSC	**ESAFE_DOCUMENTS_READY_FOR_TRANSFER**	Redirect to this page, if the SP has finished the document selection, and the transfer can start (relevant only for the PULL principle).
PSC	**ESAFE_DOCUMENTS_TRANSFERRED**	Redirect to this page, if the documents have successfully been transported to

[34] Not supported by the open modules in the initial version.

Owner	UI entry point symbolic name	Description
		the PSC (relevant only for the PUSH principle).
PSC	**ESAFE_DOCUMENT_TRANSFER_FAILED**	Redirect to this page in case of a non recoverable error.

Table 4: Keys for looking up specific UIs on the partner's side

Variables that can be used in the in the UI entry points' URL templates are:

Variable	Description
${pscSessionId}	The session Id is used by the PSC, MANDATORY for all URLs on the PSC side, otherwise OPTIONAL.
${eSafeSessionId}	The session id used by the eSafe, MANDATORY for all URLs on the eSafe side, otherwise OPTIONAL.
${accessToken}	Relevant only for the PULL principle with the access token strategy PROVIDE_WITH_REDIRECT when redirecting to the ESAFE_DOCUMENTS_READY_FOR_TRANSFER page: The variable has then to be replaced by the access token encrypted for the PSC.

Table 5: Variables for URL templates

4.4.4 Document transfer packages

Regardless of the document transfer principle (PUSH or PULL) the PSC will always receive a document transfer package. On the logical level a document transfer package looks like shown in Figure 9:

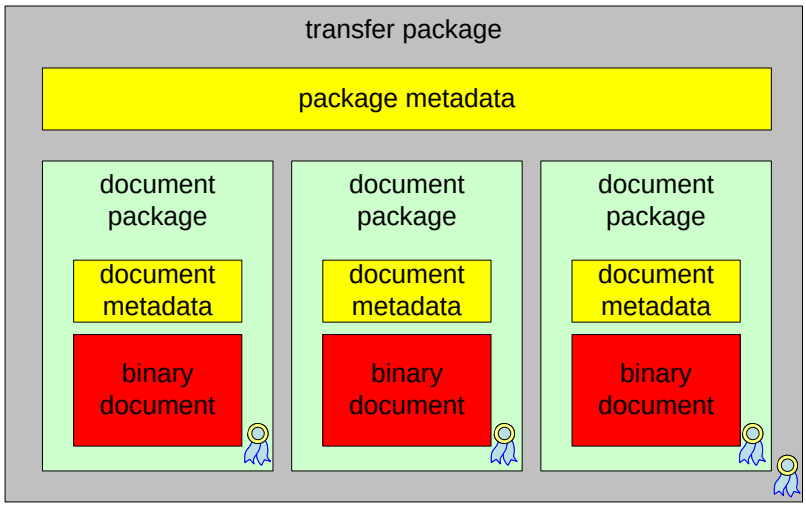

Figure 9: Example of a package with three documents to transfer

A document transfer package contains one or more document packages, each consisting of the binary document and the document's metadata as defined in this specification.

Each document may be digitally signed for ensuring the data's integrity by creating a digest value of the binary document and adding it to the document's metadata. Then the

metadata is signed with the eSafe's signature certificate. This signature is created, regardless if any electronic signature is embedded in the binary document or not.

Having received the document transfer package, the PSC can then use the eSafe's public key (of the certificate) for verifying the document's integrity as well as the signer's authenticity. The value of the signature guarantees that the document provided is exactly the same as it is stored in the eSafe. Signing the documents is therefore RECOMMENDED, especially if the documents are not signed with any other electronic signature (which may not be enforced or checked by the individual eSafe). However, the usage of digital signatures as well as the algorithms to be used for creating the digest values and the signatures depends on the PSC's and the eSafe's capabilities. This specification only defines the minimal set of supported algorithms (see subsections 4.4.9 "Signature and digest algorithms") which must be adapted in case of newly identified vulnerabilities or key aging.

Each document may also be digitally encrypted. The PSC can choose among various certificates to be used for encryption. The default certificate to be used for encryption is the PSC's SSL public key certificate. However, the PSC can also provide a public key certificate at session initiation. This can either be another certificate owned by the PSC or it may even be a certificate owned by the target CA (to gain end-to-end security). The decision is up to the PSC's national implementation.

Since the document transfer between eSafe and PSC is currently based on Web services over HTTP via SSL/TLS, the encryption of the documents is OPTIONAL. This specification only defines the minimal set of supported algorithms (see subsection 4.4.8 "Encryption algorithms") which must be adapted in case of newly identified vulnerabilities or key aging.

As with the documents the document transfer package itself contains a set of (top level) metadata attributes, like a subject or shipping data.

Furthermore the whole document transfer package may be signed as well. In this case the signature considers all the document packages and the document transfer package's top level metadata.

4.4.5 Document transfer package metadata

The top level metadata of the whole document transfer package are listed in the Table 6[35]:

Attribute name	MUST or OPTIONAL	Description	OCD attribute mapping[36]
version	MUST	Version of the OCD metadata structure, currently "1"	**ocdm:Version**

[35] The sort order of the attributes is not reflecting the order of the attributes in the OCD container file, which is specified by the OCD metadata XML schema definition.

[36] The documents will be transferred to the PSC as an OCD Container (see SPOCS specification D2.2 "Standard Document and Validation Common Specifications"). The metadata will be provided as attributes within the OCD metadata XML file. This column shows which XML type will be used for that.

Attribute name	MUST or OPTIONAL	Description	OCD attribute mapping[36]
subject	MUST	"eSafe DTP (Document Transfer Package)"	**ocdm:Subject**
creationDate	MUST	Creation date	**ocdm:CreationDate**
annotation	OPTIONAL	reserved	**ocdm:Annotation**
catchWord	OPTIONAL	reserved	**ocdm:CatchWord**
senderId	MUST	eSafe Id	**ocdm:SenderParty/ cbc:EndPointID**
senderName	OPTIONAL	eSafe name	**ocdm:SenderParty/ cac:PartyName/cbc:Name**
receiverId	MUST	PSC Id	**ocdm:ReceiverParty/ cbc:EndPointID**
receiverName	OPTIONAL	PSC name	**ocdm:ReceiverParty/ cac:PartyName/cbc:Name**
originCountryId	OPTIONAL	eSafe country	**ocdm:OriginCountry/ cbc:IdentificationCode**
originCountryName	OPTIONAL	eSafe country	**ocdm:OriginCountry/ cbc:Name**
destinationCountryId	OPTIONAL	PSC country	**ocdm:DestinationCountry/ cbc:IdentificationCode**
destinationCountryName	OPTIONAL	PSC country	**ocdm:DestinationCountry/ cbc:Name**

Table 6: Top level metadata as included in the document transfer package

4.4.6 Document metadata as included in the document transfer package

The document's metadata as included in the document transfer package are listed in the Table 7[37]:

Attribute name	MUST or OPTIONAL	Description	OCD attribute mapping[38]
id	MUST	Unique identifier of a document within the eSafe; make it unique	**ocdm:DocumentReference/ cbc:ID**

[37] The sort order of the attributes is not reflecting the order of the attributes in the OCD container file, which is specified by the OCD metadata XML schema definition.

[38] Documents will be transferred as an OCD Container (see SPOCS specification D2.2 "Standard Document and Validation Common Specifications"). The document's metadata will be provided within the OCD metadata XML file. This column shows which XML type will be used for that.

Attribute name	MUST or OPTIONAL	Description	OCD attribute mapping[38]
		outside the eSafe, some eSafe ID SHOULD be added. e.g.**<eSafeId>-<docId>**	
fileName	MUST	Name of the document (without any path information)	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:FileName**
documentVersion	OPTIONAL	Version of the document	**ocdm:DocumentReference/oc dm:DocumentVersion**
documentHash	OPTIONAL	Document fingerprint. In case of document integrity assertion by signatures the metadata including the fingerprint will be signed.	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:DocumentHash**
description	OPTIONAL	Short description of the document	**ocdm:DocumentReference/oc dm:Description**
mimeCode	MUST	Mime-Type of the document according the RFC for Internet Media Type [2]	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:MimeCode**
formatCode	OPTIONAL	Format of the document, e.g. PDF/A	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:formatCode**
characterSetCode	OPTIONAL	character set of the document e.g. UNICODE-16, ANSI	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:characterSetCode**
encodingCode	OPTIONAL	character encoding of the document, e.g. UTF-8	**ocdm:DocumentReference/ cac:Attachment/ cac:ExternalReference/ cbc:encodingCode**
issueDate	OPTIONAL	Date of issue of the document	**ocdm:DocumentReference/ cbc:IssueDate**
issueTime	OPTIONAL	Date of issue of the document	**ocdm:DocumentReference/ cbc:IssueTime**
languageId	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:ID**
languageName	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:Name**
localeCode	OPTIONAL	Language of the document	**ocdm:DocumentReference/ cac:Language/ cbc:LocaleCode**

Table 7: Document related metadata as included in the document transfer package

4.4.7 OCD Container

For interoperability purposes for the documents exchanged between an eSafe and a PSC (and likely also between PSC and CA, which is however out of scope of this

specification) the document transfer packages will be transferred to the PSC as an OCD Container as mentioned before already (see SPOCS deliverable D2.2).

As OCD is designed to be an open standard, various usage scenarios are possible. For retrieving documents from an eSafe in the context of SPOCS, a specific profile named "eSafe DTP profile" (DTP - Document Transfer Package) has been created in D2.2, with the following characteristics:

- `ZIP` is used as the container file format
- Definition of the minimal and optional meta data for the document transfer package (the whole container)
- Definition of the minimal and optional meta data for each document of the document transfer package
- Definition of the supported algorithms regarding the creation of digest values, electronic signatures and for encryption

4.4.8 Encryption algorithms

The SPOCS protocol for retrieving documents from an eSafe allows the documents to be encrypted for the target PSC or for a target CA.

Furthermore, in case of the PULL principle an access token is required, which has to be transported from the eSafe to the PSC (see subsection 4.4.2 "PULL Principle" for further details). This access token MUST be encrypted for the target PSC.

Therefore, the support of a minimal set of encryption algorithms is required in the piloting phase in order to be compliant to this specification[39]. Further algorithms are optional.

Short name	Identifier	MUST or OPTIONAL	Description
`RSA`	http://www.w3.org/2001/04/xmlenc#rsa-1_5	OPTIONAL	Used for the encryption of the access token and the encryption of the sym-metric key, optionally for encrypting the data. RSA as used in this specification refers to the RSASSA-PKCS1-v1_5 algorithm [42]
`AES128`	http://www.w3.org/2001/04/xmlenc#aes128-cbc	OPTIONAL	Data encryption algorithm [43]

Table 8: Encryption algorithms for retrieving documents from an eSafe

4.4.9 Signature and digest algorithms

The SPOCS protocol for retrieving documents from an eSafe allows the documents (more specifically the document's fingerprint / hash value) to be digitally signed.

[39] The identifiers listed are as defined in „Algorithms" in the „XML Encryption Syntax and Processing" [19] „XML Signature Syntax and Processing (Second Edition)" [3], and „Additional XML Security Uniform Resource Identifiers (URIs)" [40].

Therefore, the support of a minimal set of cryptographic hash and signature algorithms is required in the piloting phase in order to be compliant to this specification[39]. Further algorithms are optional.

Short name	Identifier	MUST or OPTIONAL	Description
SHA-256	http://www.w3.org/2001/04/xmlenc#sha256	MUST	Secure Hash Standard FIPS PUB 180-2 [39]
SHA-512	http://www.w3.org/2001/04/xmlenc#sha512	OPTIONAL	Secure Hash Standard FIPS PUB 180-2 [39]
RSA-SHA256	http://www.w3.org/2001/04/xmldsig-more#rsa-sha256	MUST	RSA as used in this specification refers to the RSASSA-PKCS1-v1_5 algorithm [42]
RSA-SHA512	http://www.w3.org/2001/04/xmldsig-more#rsa-sha512	OPTIONAL	RSA as used in this specification refers to the RSASSA-PKCS1-v1_5 algorithm [42]

Table 9: Signature and digest algorithms for retrieving documents from an eSafe

4.4.10 Session Handling

Another principle is that the Web service based communication between PSC and eSafe shall be open for asynchronous processing, which allows better utilisation of the systems' resources, since blocking of communication facilities (e.g. TCP/IP connections, SOAP handler objects etc.) is avoided. Each message exchanged between PSC and eSafe and in any direction has therefore to address the right conversation (see Figure 10).

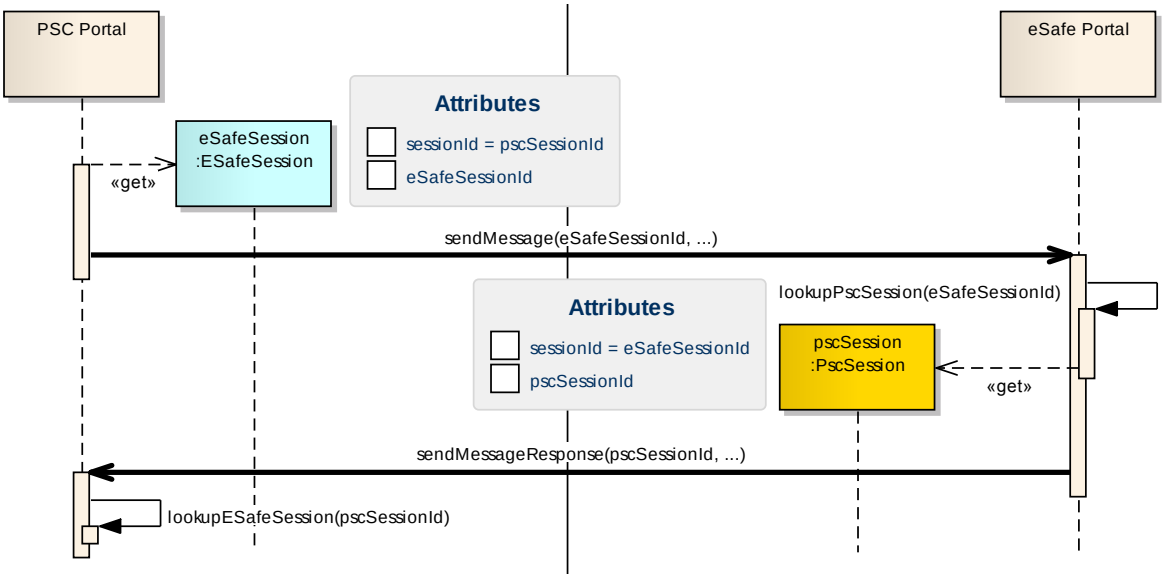

Figure 10: Addressing the right document exchange session

Therefore, as stated above in subsection 2.4.2 "Interaction model basics", the PSC and eSafe start their conversation by establishing a common session. A handshake mechanism is used for creating a session Id and for defining the effective transfer options (document transfer following the PUSH or PULL principle, access token transfer strategy in case of pull transfer etc.). In fact, both communication partners create an own

session Id and share it with each other as one of the session parameters. In the following communication messages always the partner's system Id for addressing the target session is used.

Two individual session Ids for addressing one conversation was chosen in order to avoid potential duplicates in situations, where one PSC handles conversations with several eSafes concurrently while some of the eSafes concurrently handle session with other PSCs as well. As each system has full control of its own session Ids, no session Id SHOULD be used twice.

The session is valid until either one communication partner closes the session or until a session timeout occurs. The specification does not propose a specific timeout value. However, in case of a session timeout, the partner system SHOULD be notified by submitting a `closeSession` message.

4.4.11 Secure communication

As already mentioned, all (HTTP- and Web service based) communication between the SP, PSC and eSafe SHOULD be done using communication channels secured by SSL/TLS. This is to ensure overall communication confidentially and integrity.

For each communication an own session key will be established and used for the whole conversion. Every national eSafe and PSC solution is strongly encouraged to allow secure cipher suites only[40].

For the communication between PSC and eSafe it is assumed that both communication partners are registered with their SSL certificate in the Service digital identity attribute (Service Information entry) within the TSL. The verification of the communication partner's certificate MUST be done during the session initiation (see subsection 4.5 Operations for retrieving documents from an eSafe).

4.4.12 Testing functionality

It is RECOMMENDED that the PSC and eSafe will provide some testing functionality in order to test the integration. The testing functionality SHOULD provide UI parts and the Web services, all implementing the full SPOCS protocol for retrieving documents from an eSafe, but not using any production user credentials or data.

This is also applicable for the TSL, where either each solution should provide a non-TSL testing possibility or a test-TSL is usable.

4.5 Operations for retrieving documents from an eSafe

The following diagrams provide an overview of the protocol specified for integrating an eSafe in the SPOCS scenario.

Messages marked in black colour are independent of the chosen document exchange principle. Messages in red colour do mark principle specific messages (PUSH or PULL). As this description is focussed to provide an overview no specific security information is provided and the provided parameters are simplified as well.

[40] Further Information for FIPS SSL CipherSuites can be found here: http://www.mozilla.org/-projects/security/pki/nss/ssl/fips-ssl-ciphersuites.html (last visited on 27 August 2010)

4.5.1 Attaching documents to an application request

This diagram describes the context where the SPOCS protocol for retrieving documents from an eSafe is embedded.

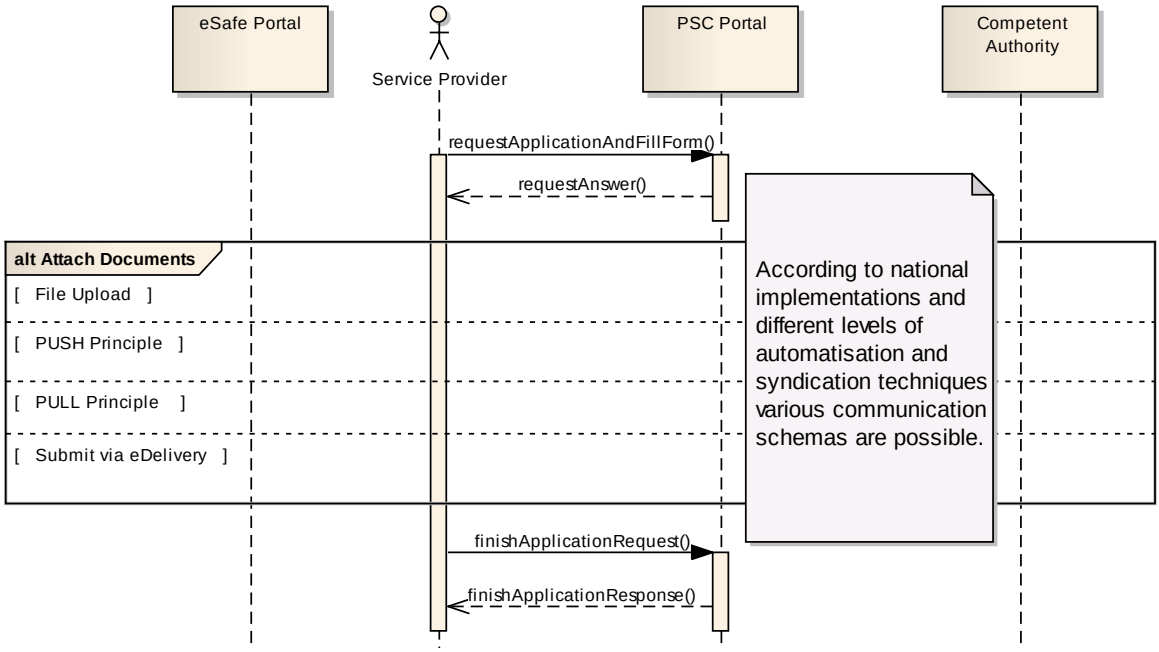

Figure 11: Attaching documents to an application request

Initially the SP starts the scenario by applying for a service at the PSC. The PSC therefore offers information and finally an online application, in order to provide the best possible support for the SP according to the main intention of the service directive.

Depending on the solution of the PSC, the SP needs to authenticate first. Following that, the SP fills out the online form and is asked to provide additional documents, if required by the competent authority.

Depending on the functionalities of the PSC solution and the location where the SP stores these documents, the SP may choose to

- submit them by directly via a file upload to the PSC portal (this case is not relevant for WP3 and therefore not further considered in this document, even if the SP previously downloads the documents from the eSafe).
- use his eSafe to attach those documents (following either the PUSH or the PULL principle)
- use the eDelivery channel to submit the documents in an asynchronous manner (which is not relevant now and therefore not specified further in the context of the eSafe specification).

After the application form is completed and all relevant documents are provided, the PSC attaches the received documents to the application request of the SP. After that the PSC handles the application over for further processing from the relevant authority. The following table describes the message flow shown in the diagram.

Messages Attaching documents to an application request

Message	eSafe_OV_Attach_[41]0 - requestApplicationAndFillForm	
From:	Service Provider	
To:	PSC Portal	
Message:	`requestApplicationAndFillForm()`	
Notes:	The SP requests an application form and enters the required fields.	
Message	eSafe_OV_Attach_1 - requestAnswer	
From:	PSC Portal	
To:	Service Provider	
Message:	`requestAnswer()`	
Notes:	Depending of the type of the application request, the PSC requests some documents to be attached and offers various alternatives to do so.	
Alternatives:	eSafe_OV_Attach_2 - Attach Documents	
Alternative 1:	File Upload	Submission of the documents via file upload to the PSC portal (not relevant for WP3, thus not further considered in this document)
Alternative 2:	PUSH principle	See section 4.5.2 “Retrieving documents following the PUSH principle”.
Alternative 3:	PULL principle	See section 4.5.3 “Retrieving documents following the PULL principle”.
Alternative 4:	Submit via eDelivery	Submission of the documents using the eDelivery channel (not relevant at this time, thus not further considered in this document)
Message	eSafe_OV_Attach_3 - finishApplicationRequest	
From:	Service Provider	
To:	PSC Portal	
Message:	`finishApplicationRequest()`	
Notes:	The SP confirms the attached documents.	
Message	eSafe_OV_Attach_4 - finishApplicationResponse	
From:	PSC Portal	
To:	Service Provider	
Message:	`finishApplicationResponse()`	
Notes:	The PSC continues with the processing and displays the appropriate UI.	

Table 10: Attaching documents to an application request (messages)

Note: `requestApplicationAndFillForm` and `requestAnswer` and `finishApplicationRequest` and `finishApplicationResponse`

These communication messages are out of scope regarding the eSafe communication. Therefore no further description is provided here except that the general security requirements for the communication SHOULD be implemented (e.g. via SSL/TLS).

[41] For gaining unique message names throughout the specification the message name is prefixed with “eSafe_O(ver)V(iew)_<diagram short name>_<message number in flow>”.

4.5.2 Retrieving documents following the PUSH principle

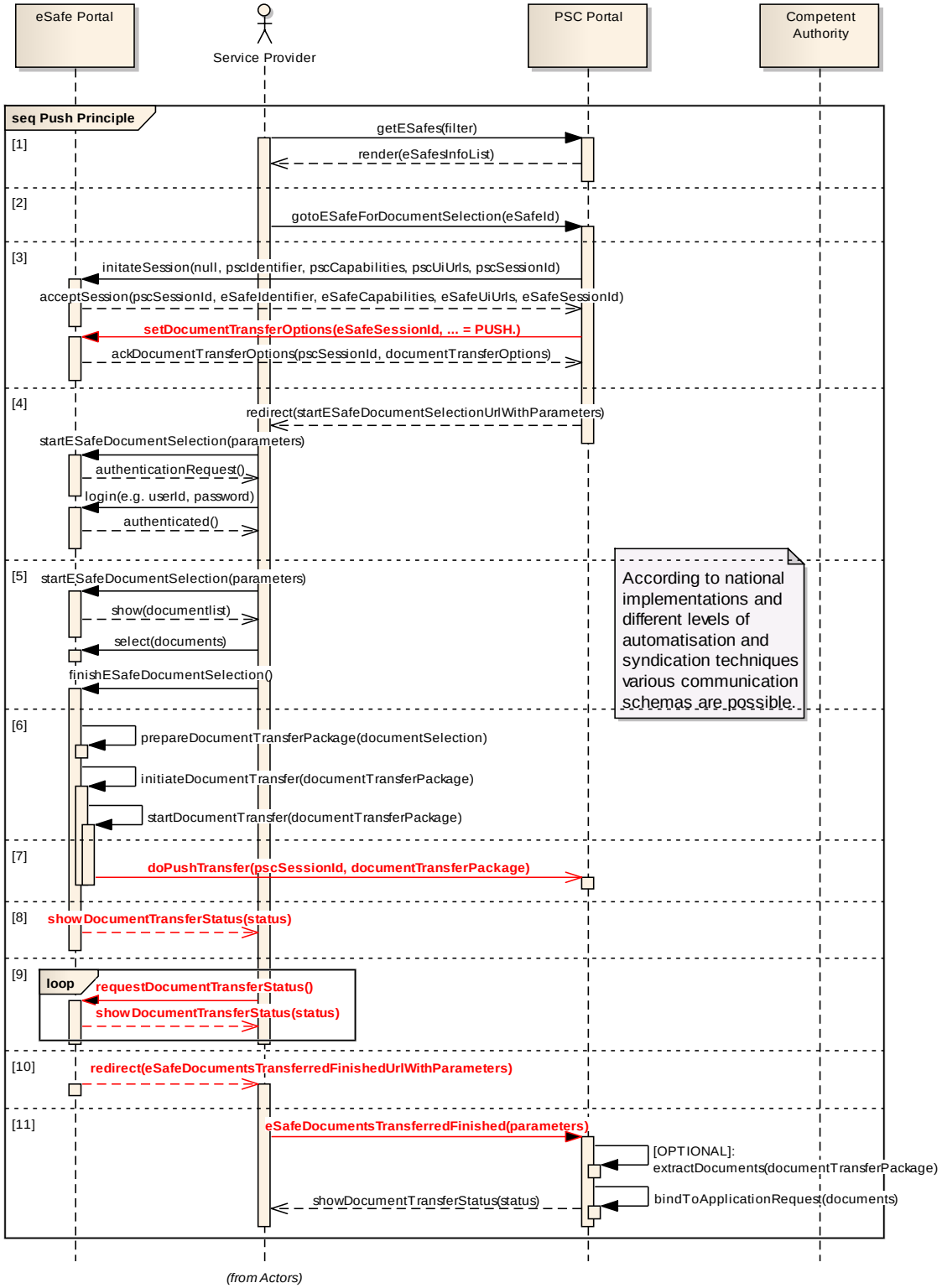

Figure 12: Retrieving documents from an eSafe - PUSH principle messages

Description

Initially the SP starts the scenario by applying for a service at the PSC and filling out an application request for which documents have to be attached. As the SP decides to attach documents from an eSafe, a list of appropriate eSafes is offered to choose from **[1]**.

This list SHOULD be created from the TSL and CAN be pre-filtered by only showing eSafes of the SP's home country (the SP MUST have the option to delete the filter). It MAY also be filtered for only showing the eSafes satisfying preferred capabilities like supporting the PUSH principle or supporting digital signatures.

Alternatively, the SP may directly input the eSafe's name and address or provide this information before in a registration process, to store the SP's eSafes in his user profile.

Once the SP selects a specific eSafe **[2]** the PSC initiates a document exchange session with that eSafe, implementing a handshake mechanism **[3]** where they

- mutually authenticate each other,
- exchange session Ids for addressing the common document exchange session at the target system (**`eSafeSessionId`**: id created by the eSafe, **`pscSessionId`**: Id created by the PSC)
- select various document transfer options based on each other's capabilities (e.g. transfer principle to be followed, algorithms for encryption and signing) and
- exchange a dictionary of URLs implementing specific process steps.

After that the SP is redirected to the eSafe **[4]** by using the template which was provided by the eSafe with the handshake at session initiation (**`START_ESAFE_DOCUMENT_SELECTION`**). The target URL (template) is created by substituting the template's variable parts (e.g. **`${session_id}`**) by their current values.

The SP needs to logon at the eSafe. Then he selects the relevant documents for the transfer **[5]**. Once this is finished, the eSafe is requested to initiate the transfer.

Following the PUSH principle the eSafe creates the document transfer package and starts the document transfer **[6]**.

The eSafe pushes the document transfer package directly to the PSC. This is done asynchronously **[7]** using a set of Web services described further in the detailed specification (in future eDelivery MAY be used as an alternative transport channel).

As long as the transfer is in progress the SP remains at the eSafe side. The eSafe is responsible for visualising the ongoing document transfer, which can take some time if the documents are of respectable size **[8]**, **[9]**. The document transfer package is split into several frames and transmitted frame by frame in a loop. This enables the sender to react to transmission problems efficiently by retrying the frame in question only instead of sending the whole document transfer package again.

Having transmitted the document transfer package including all documents successfully, the eSafe redirects the SP back to the PSC **[10]** by using the template which was provided by the PSC with the initial handshake (**`ESAFE_DOCUMENTS_TRANSFERRED`**). The target URL is created by substituting the template's variable parts (e.g. **`${session_id}`**) by their current values. Back at the PSC **[11]**, the PSC finally

- checks the document transfer's success,
- checks the documents' integrity by verifying the electronic signature (if available),
- optionally extracts the documents from the document transfer package,
- attaches the whole document transfer package or the individual documents to the application request and
- provides an appropriate feedback to the SP. Thereby the SP MAY also have the chance to verify the documents at the PSC.

Message flow as shown in the diagram

Messages **Retrieving documents from an eSafe following the PUSH principle**

Message	**eSafe_OV_PUSH_0 - getESafes**
From:	Service Provider
To:	PSC Portal
Message:	**`getESafes(filter)`**
Notes:	The Service Provider requests a list of eSafes, optionally satisfying some filter criteria. Normally, the filter criteria should contain the Service Provider's country, but maybe additional criteria may apply. The filter must be able to be deleted by the SP.
Message	**eSafe_OV_PUSH_1 - render**
From:	PSC Portal
To:	Service Provider
Message:	**`render(eSafesInfoList)`**
Notes:	The PSC returns a list of eSafes satisfying the given filter criteria. For each eSafe a selection of identity attributes and capabilities (e.g. encryption capabilities) are displayed. The minimum requirement is to have the list distinguishable for the SP.
Message	**eSafe_OV_PUSH_2 - gotoESafeForDocumentSelection**
From:	Service Provider
To:	PSC Portal
Message:	**`gotoESafeForDocumentSelection(eSafeId)`**
Notes:	The Service Provider selects one of the provided eSafes and requests the document selection from the selected *source of authenticated documents*.
Message	**eSafe_OV_PUSH_3 - initateSession**
From:	PSC Portal
To:	eSafe Portal
Message:	**`initateSession(null, pscIdentifier, pscCapabilities, pscUiUrls, pscSessionId)`**
Notes:	The PSC initiates a session with the eSafe, sending his identification attributes, communication capabilities and a set of URLs on the PSC side needed for the UI communication flow according the SPOCS protocol specified. Furthermore, the PSC sends the eSafe the PSC-side session id. Whenever the eSafe sends a message to the PSC concerning this session this id has to be included as a parameter.
Message	**eSafe_OV_PUSH_4 - acceptSession**
From:	eSafe Portal
To:	PSC Portal
Message:	**`acceptSession(pscSessionId, eSafeIdentifier, eSafeCapabilities, eSafeUiUrls, eSafeSessionId)`**
Notes:	The eSafe accepts the session, returning the eSafe's identification attributes, communication capabilities and a set of URLs on the eSafe side needed for the communication flow according the SPOCS protocol specified. Furthermore, the eSafe sends the PSC the eSafe-side session id. Whenever the PSC sends a message to the eSafe concerning this session this id has to be included as a

Messages	Retrieving documents from an eSafe following the PUSH principle
	parameter.
Message	**eSafe_OV_PUSH_5 - setDocumentTransferOptions**
From:	PSC Portal
To:	eSafe Portal
Message:	**`setDocumentTransferOptions(eSafeSessionId, ... = PUSH.)`**
Notes:	After having established a session, the PSC sends the eSafe the document transfer options that have to be obeyed for this session. These options are computed by the PSC based on his own and the eSafe's communication capabilities, which have been exchanged during session initiation.
Message	**eSafe_OV_PUSH_6 - ackDocumentTransferOptions**
From:	eSafe Portal
To:	PSC Portal
Message:	**`ackDocumentTransferOptions(pscSessionId, documentTransferOptions)`**
Notes:	The eSafe accepts the document transfer options as requested by the PSC.
Message	**eSafe_OV_PUSH_7 - redirect**
From:	PSC Portal
To:	Service Provider
Message:	**`redirect(startESafeDocumentSelectionUrlWithParameters)`**
Notes:	The Service Provider is now redirected to the eSafe, using an URL that was provided to the PSC at session initiation merged with parameters defining the session.
Message	**eSafe_OV_PUSH_8 - startESafeDocumentSelection**
From:	Service Provider
To:	eSafe Portal
Message:	**`startESafeDocumentSelection(parameters)`**
Notes:	The eSafe is requested to provide the UI for the document selection (within the session defined through the parameters). At first, normally the Service Provider has to authenticate.
Message	**eSafe_OV_PUSH_9 - authenticationRequest**
From:	eSafe Portal
To:	Service Provider
Message:	**`authenticationRequest()`**
Notes:	When the SP is not authenticated yet, the eSafe requires the SP to authenticate first. The original HTTP request is stored for redirection after a successful login.
Message	**eSafe_OV_PUSH_10 - login**
From:	Service Provider
To:	eSafe Portal
Message:	**`login(e.g. userId, password)`**
Notes:	The login procedure is up to the portal's individual implementation.

Messages	Retrieving documents from an eSafe following the PUSH principle
Message	**eSafe_OV_PUSH_11 - authenticated**
From:	eSafe Portal
To:	Service Provider
Message:	`authenticated()`
Notes:	The eSafe accepts the Service Provider's login credentials and replies with a redirect request to the original requested URL.
Message	**eSafe_OV_PUSH_12 - startESafeDocumentSelection**
From:	Service Provider
To:	eSafe Portal
Message:	`startESafeDocumentSelection(parameters)`
Notes:	The Service Provider's browser executes the redirect request received from the eSafe for the UI for the document selection.
Message	**eSafe_OV_PUSH_13 - show**
From:	eSafe Portal
To:	Service Provider
Message:	`show(documentlist)`
Notes:	The eSafe provides the UI for the document selection.
Message	**eSafe_OV_PUSH_14 - select**
From:	Service Provider
To:	eSafe Portal
Message:	`select(documents)`
Notes:	The Service Provider selects the documents to be transferred to the PSC (this might be a complex interaction sequence in the responsibility of the national eSafe).
Message	**eSafe_OV_PUSH_15 - finishESafeDocumentSelection**
From:	Service Provider
To:	eSafe Portal
Message:	`finishESafeDocumentSelection()`
Notes:	The Service Provider finishes the selection of the documents to be transferred to the PSC. In this diagram the selection is assumed not to be empty.
Message	**eSafe_OV_PUSH_16 - prepareDocumentTransferPackage**
From:	eSafe Portal
To:	eSafe Portal
Message:	`prepareDocumentTransferPackage(documentSelection)`
Notes:	Given a collection of documents to be transferred to the PSC, the eSafe creates the document transfer package.
Message	**eSafe_OV_PUSH_17 - initiateDocumentTransfer**
From:	eSafe Portal
To:	eSafe Portal
Message:	`initiateDocumentTransfer(documentTransferPackage)`

Messages	Retrieving documents from an eSafe following the PUSH principle
Notes:	The prepared document transfer package is now ready to be sent to the PSC.
Message	**eSafe_OV_PUSH_18 - startDocumentTransfer**
From:	eSafe Portal
To:	eSafe Portal
Message:	`startDocumentTransfer(documentTransferPackage)`
Notes:	Following the PUSH principle the `initiateDocumentTransfer()` method ends with calling the eSafe's `startDocumentTransfer()` method. This method then triggers the asynchronous submission of the document transfer package, pushing it from the eSafe to the PSC.
Message	**eSafe_OV_PUSH_19 - doPushTransfer**
From:	eSafe Portal
To:	PSC Portal
Message:	`doPushTransfer(pscSessionId, documentTransferPackage)`
Notes:	The document transfer package is actually pushed to the PSC. This message symbolises a more complex asynchronous protocol part, since the document transfer is split into frames.
Message	**eSafe_OV_PUSH_20 - showDocumentTransferStatus**
From:	eSafe Portal
To:	Service Provider
Message:	`showDocumentTransferStatus(status)`
Notes:	The eSafe has just started to push the document transfer package asynchronously, the Service Provider is shown a page displaying the current transfer status.
Message	**eSafe_OV_PUSH_21 - requestDocumentTransferStatus**
From:	Service Provider
To:	eSafe Portal
Message:	`requestDocumentTransferStatus()`
Notes:	The status page is refreshed after a few seconds until the transfer finishes. This message gets the current transfer status.
Message	**eSafe_OV_PUSH_22 - showDocumentTransferStatus**
From:	eSafe Portal
To:	Service Provider
Message:	`showDocumentTransferStatus(status)`
Notes:	While the eSafe started to push the document transfer package asynchronously, the Service Provider is shown a page displaying the current transfer status.
Message	**eSafe_OV_PUSH_23 - redirect**
From:	eSafe Portal
To:	Service Provider
Message:	`redirect(eSafeDocumentsTransferredFinishedUrlWithParameters)`

Messages	Retrieving documents from an eSafe following the PUSH principle
Notes:	After the document transfer package is successfully pushed to the PSC the Service Provider is redirected back to the PSC, using an URL that was provided to the eSafe at session initiation.
Message	**eSafe_OV_PUSH_24 - eSafeDocumentsTransferredFinished**
From:	Service Provider
To:	PSC Portal
Message:	**`eSafeDocumentsTransferredFinished(parameters)`**
Notes:	After the document transfer package is successfully pushed to the PSC the Service Provider is redirected back to the PSC, using an URL that was provided to the eSafe at session initiation. The received documents are checked for integrity.
Message	**eSafe_OV_PUSH_25 - extractDocuments(documentTransferPackage)**
From:	PSC Portal
To:	PSC Portal
Message:	**`extractDocuments(documentTransferPackage)`**
Condition:	OPTIONAL
Notes:	The documents are extracted from the document transfer package. This includes the decryption of the documents (only possible if encryption was requested at session initiation and the private key is available - which won't be the case if the decryption key belongs to the target CA) and the verification of the signatures.
Message	**eSafe_OV_PUSH_26 - bindToApplicationRequest**
From:	PSC Portal
To:	PSC Portal
Message:	**`bindToApplicationRequest(documents)`**
Notes:	The documents are bound to the application request.
Message	**eSafe_OV_PUSH_27 - showDocumentTransferStatus**
From:	PSC Portal
To:	Service Provider
Message:	**`showDocumentTransferStatus(status)`**
Notes:	Finally, the PSC informs the Service Provider that he has received the documents and attached them to the application.

Table 11: Retrieving documents from an eSafe - PUSH principle messages

Alternative scenarios (not shown in the diagram)

If the SP cancels the document selection activity while he is connected to the eSafe the SP is redirected back to the PSC's (URL: **`ESAFE_DOCUMENT_SELECTION_CANCELLED`**).

In case of a transmission error (at any point in time) the SP is redirected back to the PSC as well (URL: **`ESAFE_DOCUMENT_TRANSFER_FAILED`**).

4.5.3 Retrieving documents following the PULL principle

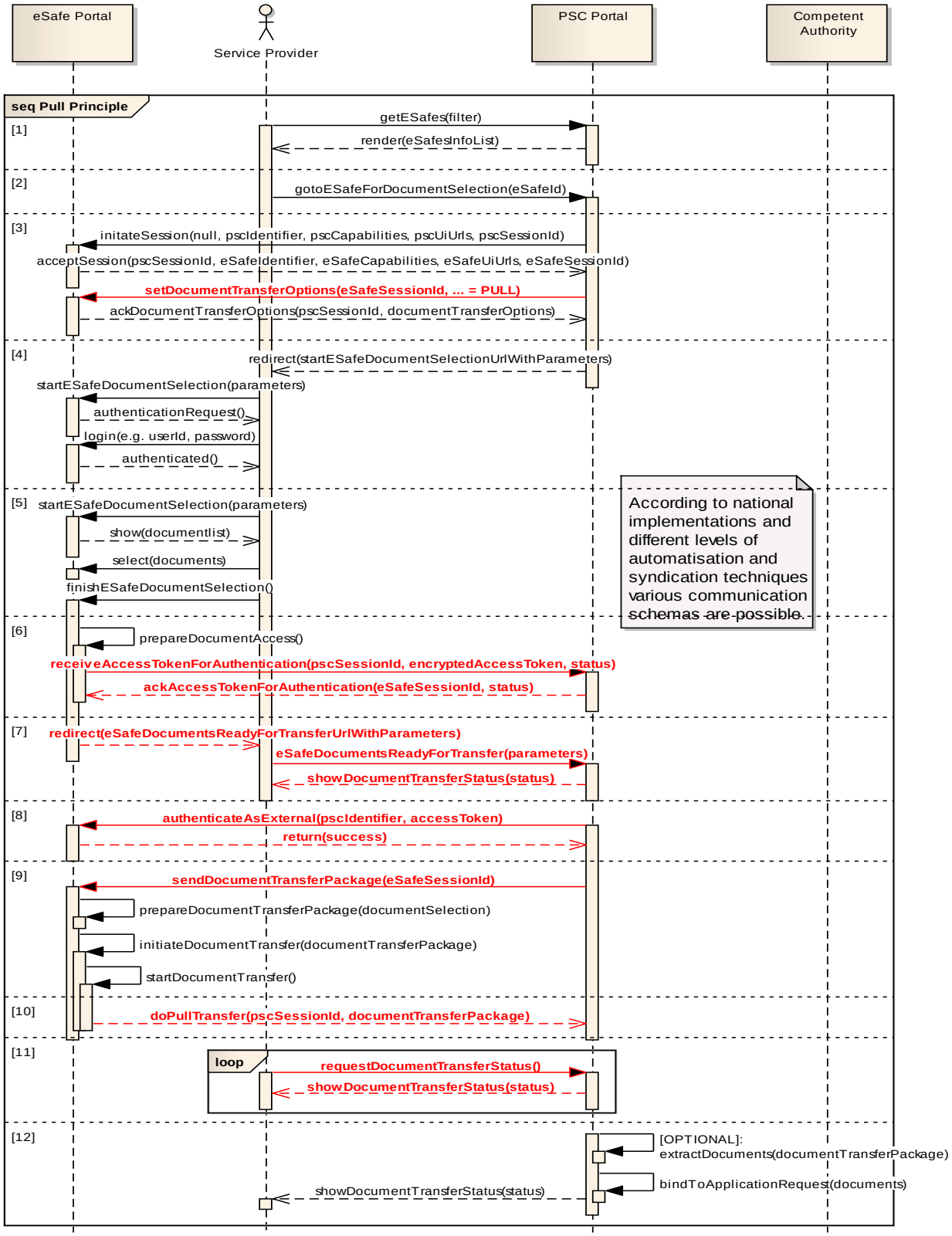

Figure 13: Retrieving documents from an eSafe - PULL principle messages

Description

The messages **[1]** to **[5]** are equivalent to the description in 4.5.2.

Following the PULL principle the eSafe allows the PSC to pull the selected documents from the eSafe. For that purpose the document retrieval is secured with an access token.

This access token has high security requirements. It MUST be generated in a secure environment. Furthermore, we strongly RECOMMEND that the access token is valid for a defined validity period only. Moreover, the access token SHOULD be usable only once (e.g. like a one time passwords).

The access token is encrypted for the target PSC and either provided to the PSC directly **[6]** or provided to the SP through an independent channel. As mentioned above, the following strategies are applicable:

- Provide access token by service (recommended)
  The encrypted access token is sent to the PSC via a Web service, which is shown in the diagram above.
- Provide access token with redirect
  The encrypted access token is provided as a parameter of the subsequent redirection (ESAFE_DOCUMENTS_READY_TO_TRANSFER needs to include an appropriate variable). By following this strategy, the last two messages of [6] have to be skipped.
- Provide access token manually
  The SP will need to type in the encrypted access token at the PSC portal. The token may be transported by email or using another transport channel, or maybe simply displayed in a popup page (for copy/paste). Again, the last two messages of **[6]** have to be skipped.

After that the control is returned to the PSC **[7]** via redirection and using the template which was provided by the PSC with the initial handshake. Again, the target URL is created by substituting the template's variable parts (e.g. `${session_id}`, optionally `${access_token}`) by their current values.

The PSC is then responsible for the residual control flow. If the encrypted access token is not yet known by the PSC, the SP has to type it in.

Regardless of the access token's provision strategy, this access token will now be decrypted. Then the PSC authenticates with the decrypted token at the eSafe und unlocks the document selection **[8]**.

After that the PSC starts pulling the documents. Having sent the initial transfer command to the eSafe, the eSafe checks the accessibility of the selected documents, creates the document transfer package and finally starts the transfer **[9]**.

The transfer itself is done asynchronously **[10]** using a set of Web services described further in the detailed specification.

The PSC is responsible for visualising the ongoing document transfer, which can take some time if the documents are of respectable size **[11]**. As within the PUSH principle, the document transfer package is split into several frames for the transfer.

Having transmitted all the documents successfully **[12]**, the PSC finally executes the same steps as within the PUSH principle.

Message flow as shown in the diagram

Within this description only those messages are shown, which are different to the messages described for the PUSH principle already (see section 4.5.2).

Messages **Retrieving documents from an eSafe following the PULL principle**

Messages Retrieving documents from an eSafe following the PULL principle

Message	eSafe_OV_PULL_5 - setDocumentTransferOptions
From:	PSC Portal
To:	eSafe Portal
Message:	`setDocumentTransferOptions(eSafeSessionId, ... = PULL)`
Notes:	After having established a session, the eSafe is sent the document transfer options that are to be obeyed for this session. These options are computed by the PSC based on his own and the eSafe's communication capabilities, which have been exchanged during session initiation.
Message	**eSafe_OV_PULL_16 - prepareDocumentAccess**
From:	eSafe Portal
To:	eSafe Portal
Message:	`prepareDocumentAccess()`
Notes:	Creating and encrypting the access token for external access to the documents selected.
Message	**eSafe_OV_PULL_17 - receiveAccessTokenForAuthentication**
From:	eSafe Portal
To:	PSC Portal
Message:	`receiveAccessTokenForAuthentication(pscSessionId, encryptedAccessToken, status)`
Notes:	The eSafe sends the PSC the encrypted access token. Later this token will be decrypted by the PSC and the PSC will authenticate at the eSafe for unlocking the document selection. Alternatively the encrypted access token can be provided with the redirection back to the PSC or typed in manually in the PSC's UI (both are not displayed in the diagram above).
Message	**eSafe_OV_PULL_18 - ackAccessTokenForAuthentication**
From:	PSC Portal
To:	eSafe Portal
Message:	`ackAccessTokenForAuthentication(eSafeSessionId, status)`
Notes:	The PSC returns the acknowledgement of receipt of the access token by sending a status message.
Message	**eSafe_OV_PULL_19 - redirect**
From:	eSafe Portal
To:	Service Provider
Message:	`redirect(eSafeDocumentsReadyForTransferUrlWithParameters)`
Notes:	Once the document access preparation have successfully finished the Service provider is redirected back to the PSC, using an URL that was provided to the eSafe at session initiation.
Message	**eSafe_OV_PULL_20 - eSafeDocumentsReadyForTransfer**
From:	Service Provider
To:	PSC Portal
Message:	`eSafeDocumentsReadyForTransfer(parameters)`

Messages Retrieving documents from an eSafe following the PULL principle

Notes:	Once the document access preparation is successfully finished the Service Provider is redirected back to the PSC, using an URL that was provided to the eSafe at session initiation.
Message	**eSafe_OV_PULL_21 - showDocumentTransferStatus**
From:	PSC Portal
To:	Service Provider
Message:	`showDocumentTransferStatus(status)`
Notes:	While the PSC started to pull the document transfer packages asynchronously, the Service Provider is shown a page displaying the current transfer status. The status page is refreshed after a few seconds until the transfer finishes.
Message	**eSafe_OV_PULL_22 - authenticateAsExternal**
From:	PSC Portal
To:	eSafe Portal
Message:	`authenticateAsExternal(pscIdentifier, accessToken)`
Notes:	To start pulling the documents the PSC authenticates himself with the access token that was provided to the PSC.
Message	**eSafe_OV_PULL_23 - return**
From:	eSafe Portal
To:	PSC Portal
Message:	`return(success)`
Notes:	The eSafe accepts the access token and grants access to the selected documents.
Message	**eSafe_OV_PULL_24 - sendDocumentTransferPackage**
From:	PSC Portal
To:	eSafe Portal
Message:	`sendDocumentTransferPackage(eSafeSessionId)`
Notes:	The document transfer package is requested from the eSafe.
Message	**eSafe_OV_PULL_25 - prepareDocumentTransferPackage**
From:	eSafe Portal
To:	eSafe Portal
Message:	`prepareDocumentTransferPackage(documentSelection)`
Notes:	Given a collection of documents to be transfered to the PSC the eSafe creates the document transfer package.
Message	**eSafe_OV_PULL_26 - initiateDocumentTransfer**
From:	eSafe Portal
To:	eSafe Portal
Message:	`initiateDocumentTransfer(documentTransferPackage)`
Notes:	The prepared document transfer package is now ready to be fetched by the PSC (in details: by the corresponding session at the PSC). Following the PULL principle an access token (for authentication purposes) needs to be transported to the PSC. The transfer initiation therefore immediately triggers the token

Messages	Retrieving documents from an eSafe following the PULL principle
	submission.
Message	**eSafe_OV_PULL_27 - startDocumentTransfer**
From:	eSafe Portal
To:	eSafe Portal
Message:	**`startDocumentTransfer()`**
Notes:	Following the PULL principle the **`initiateDocumentTransfer()`** method ends with calling the eSafe's **`startDocumentTransfer()`** method. This method then triggers the asynchronous submission of the document transfer package, letting it be pulled from the eSafe to the PSC.
Message	**eSafe_OV_PULL_28 - doPullTransfer**
From:	eSafe Portal
To:	PSC Portal
Message:	**`doPullTransfer(pscSessionId, documentTransferPackage)`**
Notes:	The document transfer package is actually pulled from the PSC. This message symbolises a more complex asynchronous protocol part, since the document transfer is split into frames.
Message	**eSafe_OV_PULL_29 - requestDocumentTransferStatus**
From:	Service Provider
To:	PSC Portal
Message:	**`requestDocumentTransferStatus()`**
Notes:	The status page is refreshed after a few seconds until the transfer finishes. This message gets the current transfer status.
Message	**eSafe_OV_PULL_30 - showDocumentTransferStatus**
From:	PSC Portal
To:	Service Provider
Message:	**`showDocumentTransferStatus(status)`**
Notes:	The PSC has started pulling the document transfer packages asynchronously. In parallel the Service Provider is shown a page displaying the current transfer status.

Table 12: Retrieving documents from an eSafe - PULL principle messages

Alternative scenarios (not shown in the diagram)

See PUSH principle in section 4.5.2.

4.6 Operations in detail

Due to reasons of readability the detailed description of the operations is not part of this document but is published separately.

4.6.1 Appendix eSafe operations in detail

The main part of the eSafe operations' detailed descriptions is published in the document Appendix 4: eSafe – Operations in Detail.

The following elements of the SPOCS protocol for retrieving documents from an eSafe are described:

- Scope, objectives and status of the document
- Domain Model
- Actors (as far as concerned by the specification)
- Attributes
- Methods
- Data types (as far as needed to be exchanged via the network)
- Simple types
- Complex types
- Data elements
- Document transfer options
- UI references
- Status and error codes
- Document metadata
- Metadata as required by the eSafe
- Document transfer package metadata
- Document metadata as included in the document transfer package
- Use Case Model
- Sequence Diagrams (as far as concerned by the specification)
- Interaction with SP via the web browser
- Messages to be exchanged via the network
- PSC and eSafe internal procedures (only as far as needed for understanding how to glue things together)
- Supported Algorithms
- Encryption
- Creation of digest values
- Digital Signature

4.6.2 Appendix eSafe WSDL specifications

Machine readable specifications for publishing Web services that are compliant to the SPOCS protocol for retrieving documents from an eSafe (as described in this specification) are provided for the PSC and the eSafe by appropriate XSD- and WSDL files, bundled as a ZIP file.

4.6.3 Appendix eSafe document transfer package examples

Note: The original appendix with an example of a document transfer packages that is compliant to OCD containers has been removed since the OCD format slightly changed. Please refer to the OCD specifications D2.2 “Standard Document and Validation Common Specifications“ and follow up documents..

4.7 Open modules for eSafe access

In order to support the integration of PSCs and eSafes within the SPOCS scenario most effective, WP3 will deliver open modules for the piloting phase. The modules will

- support managing the control flow and the service oriented communication (between PSC and eSafe)
- cover the creation of document transfer packages (OCD containers).

Currently, it is planned to create two separated packages, one for the PSC and on for the eSafe. Common functionality (e.g. for accessing the TSL, dealing with OCD container) may be developed as separate modules but integrated in these packages. If feasible, the packages will be provided as libraries that can be imported by PSC and eSafe and be accessed trough native API calls. However, this is not yet decided but task of the forthcoming specifications.

The provisioning of appropriate information for the SP (in scope of SPOCS and beyond) is the PSC's unique responsibility. This is also the case for providing application request forms as well as for all the processing of the application requests, before as well as after taking over the document transfer packages pushed by or pulled from the eSafe.

The storage and management of personal electronic files and documents will still be carried out within the eSafes. The national solutions have to ensure the requirements for secure storage and access to trusted parties. They shall provide reliable and secure archival and retrieval mechanisms of the information.

All components, PSC, eSafe and open modules are REQUIRED to follow the security and reliability architecture as described in section 2.2.

5 Conclusion

The deliverable gives the audience an overview of existing specifications relevant for establishing interoperable data exchange, acknowledgements of receipt, access and functionalities around eDelivery and eSafe solutions and a general impression of the chosen security architecture.

The deliverable explained the necessary components and dependencies based on the defined Processes, Use Cases and Scenarios. Key objective was to enrich the existing Point of Single Contact (PSC) solutions with transactional accesses to foreign Member States eDelivery and eSafe Infrastructures, so SPs have no need to use foreign applications in the future.

As worked out in Sections 1 and 2 we elaborated for both core communication scenarios (eDelivery and eSafe) a common security Infrastructure, based on

(i) International Standards and

(ii) already developed specifications and modules from sibling LSPs and ETSI.

Certificate based communication and the challenge of verifying different nationally issued certificates was solved by adopting the TSL concept, bringing that forward with the PEPS infrastructure and implemented a “circle of Trust” all over Europe (SPOCS TSL). All addressed security requirements such as confidentiality, integrity, authenticity, accountability and non repudiation will be satisfied for the entities involved, regardless of the external (uncontrollable) nodes that intervene in this communication.

In Sections 3 and 4 we have specified the challenges of making national solutions already in place as interconnected as possible via an “SPOCS Interconnect protocol”. Therefore the needs for the open modules are described to bring the different national eDelivery Infrastructures and the national eSafe solutions together.

The next steps will focus on

(i) developing, based on the specifications in this deliverable, the eDelivery and eSafe open modules and

(ii) successfully deploying these modules in the SPOCS piloting countries, when the ability to solve the addressed interoperability issues will be shown in practice.

After all, the proof of the pudding is in the eating!

References

[1] RFC 2119: Key words for use in RFCs to Indicate Requirement Levels; http://tools.ietf.org/html/rfc2119 (last visited on 08th May 2010)

[2] RFC 2046: Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types; http://tools.ietf.org/html/rfc2046 (last visited on 08th May 2010)

[3] XML Signature Syntax and Processing (Second Edition): http://www.w3.org/TR/xmldsig-core/ (last visited on 08th May 2010)

[4] RFC 1951: DEFLATE Compressed Data Format Specification version 1.3; http://tools.ietf.org/html/rfc1951 (last visited on 08th May 2010)

[5] W3C Recommendation Web Services Addressing 1.0; http://www.w3.org/TR/ws-addr-core/ (last visited on 08th May 2010)

[6] OASIS Standard WS-Security; http://www.oasis-open.org/specs/index.php#wssv1.1 (last visited on 08th May 2010)

[7] SPOCS Project, D3.1 "Assessment of eDelivery systems and specifications required for interoperability", http://www.eu-spocs.eu/index.php?option=com_processes&task=showDocument&did=198&id=18&Itemid=1 (last visited on 20th May 2010)

[8] SPOCS Project, D5.2 parts "Functional requirements for WP3 eDelivery" and "Functional requirements for WP3 eSafe", not yet published

[9] ETSI TS 102 231, v3.1.2, Electronic Signatures and Infrastructures (ESI); Provision of harmonized Trust-service status information; http://www.etsi.org/deliver/etsi_ts/102200_102299/102231/03.01.02_60/ts_102231v030102p.pdf (last visited on 20th May 2010)

[10] PEPPOL Deliverable D1.1 Requirements for Use of Signatures in Public Procurement Processes, Part 4: Architecture and Trust Models; http://www.peppol.eu/work_in_progress/wp-1-esignature/results/d1-1-part-4-architecture-and-trust-models (last visited on 20th May 2010)

[11] PEPPOL BusDox v. 1.0 specifications, http://www.peppol.eu/work_in_progress/wp8-Solutions%20architecture %2C%20design%20and%20validation/specifications/v1-0-specifications (last visited on 21th May 2010)

[12] ETSI TS 102 640-2 V2.2.1, Electronic Signatures and Infrastructures (ESI); Registered Electronic Mail (REM); Part 2: Data Requirements and Formats for Signed Evidences for REM; http://www.etsi.org/deliver/etsi_ts/102600_102699/10264002/02.02.01_60/ts_10264002v020201p.pdf (last visited on 26th March 2012)

[13] SOAP Message Transmission Optimization Mechanism, W3C Recommendation 25 January 2005, http://www.w3.org/TR/soap12-mtom/

[14] SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), W3C Recommendation 27 April 2007, http://www.w3.org/TR/soap12-part1/

[15] Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.2, OASIS Standard, 2 September 2003, http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf, (last visited on 21th May 2010)

[16] Web Services Addressing 1.0 – SOAP Binding, W3C Recommendation 9 May 2006, http://www.w3.org/TR/2006/REC-ws-addr-soap-20060509/, (last visited on 21th May 2010)

[17] Web Services Security SAML Token Profile 1.1, OASIS Standard Specification incorporating Approved Errata, 1 November 2006, http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-errata-os-SAMLTokenProfile.pdf, (last visited on 21th May 2010)

[18] Web Services Security X.509 Certificate Token Profile 1.1, OASIS Standard Specification, incorporating Approved Errata, 1 November 2006, http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-errata-os-x509TokenProfile.pdf, (last visited on 21th May 2010)

[19] World Wide Web Consortium. XML Encryption Syntax and Processing, W3C Recommendation, 10.12.2002; http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/, (last visited on 21th May 2010)

[20] World Wide Web Consortium. Extensible Markup Language (XML) 1.0 (Fourth Edition), T. Bray, J. Paoli, C. M. Sperberg-McQueen, and E. Maler, Editors. 10 February 1998, revised 16 August 2006; http://www.w3.org/TR/2006/REC-xml-20060816/, (last visited on 21th May 2010)

[21] RFC 2368, The mailto URL scheme; http://www.rfc-editor.org/rfc/rfc2368.txt

[22] RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1; http://www.rfc-editor.org/rfc/rfc2616.txt, (last visited on 21th May 2010)

[23] RFC 2817, Upgrading to TLS Within HTTP/1.1; http://tools.ietf.org/html/rfc2817, (last visited on 21th May 2010)

[24] RFC 4122, A Universally Unique Identifier (UUID) URN Namespace, The Internet Engeneering Task Force July 2005, http://www.ietf.org/rfc/rfc4122.txt, (last visited on 21th May 2010)

[25] OSCI-Transport - Version 2.0, Edition 3 - Web Services Profiling and Extensions Specification, OSCI Steering Office 2010, http://www.osci.eu/transport/osci20/20100427/OSCI20_WS-ProfilingAndExtensionSpecification_Edition3.pdf, (last visited on 21th May 2010)

[26] RFC 822, Standard for the format of ARPA Internet text messages, The Internet Engeneering Task Force August 1982, http://www.ietf.org/rfc/rfc0822.txt, (last visited on 21th May 2010)

[27] ETSI TS 101 903: XML Advanced Electronic Signatures, V1.4.1 2009-06; http://pda.etsi.org/exchangefolder/ts_101903v010401p.pdf (last visited on 9th July 2010)

[28] RFC 822, Standard for the format of ARPA Internet text messages, The Internet Engineering Task Force August 1982, http://www.ietf.org/rfc/rfc0822.txt, (last visited on 21th May 2010)

[29] WS-I Basic Profile 2.0, Working Group Draft, 2007-10-25, Web Services Interoperability Organization, http://www.ws-i.org/Profiles/BasicProfile-2_0(WGD).html, (last visited on 19th July 2010)

[30] SOAP Message Transmission Optimization Mechanism, W3C Recommendation 25 January 2005, http://www.w3.org/TR/soap12-mtom/ (last visited on 27th July 2010)

[31] XML Binary Optimized Packaging, W3C Recommendation 25 January 2005, http://www.w3.org/TR/xop10/ (last visited on 27th July 2010)

[32] RFC 5322: Internet Message Format; http://tools.ietf.org/html/rfc5322 (last visited on 28th July 2010)

[33] Describing Media Content of Binary Data in XML, W3C Working Group Note, 5 May 2005, http://www.w3.org/TR/xml-media-types/ (last visited on 29th July 2010)

[34] RFC 2396: Uniform Resource Identifiers (URI): Generic Syntax, http://www.ietf.org/rfc/rfc2396.txt, (last visited on 11th August 2010)

[35] STORK D5.1.8.b - Interface Specification, 31/7/2009, https://www.eid-stork.eu/index.php?option=com_processes&Itemid=&act=streamDocument&did=960 (last visited on 08th August 2010)

[36] STORK D6.4.1 - eDelivery Functional Specification, 08/11/2009, https://www.eid-stork.eu/index.php?option=com_processes&Itemid=&act=streamDocument&did=971 (last visited on 08th August 2010)

[37] ISO 3166-1 Country Codes, Version2006, lat update 2009-10-23, http://www.tm-xml-wiki.org/wiki/TM-XML_ISO_3166_Country_Code_XSD (last visited on 18th August 2010)

[38] STORK D2.3 - STORK Quality authenticator scheme, 2009-03-03, https://www.eid-stork.eu/index.php?option=com_processes&Itemid=&act=streamDocument&did=577 (last visited on 18th August 2010)

[39] Secure Hash Standard, Federal Information Processing Standards Publication 180-2 (extended to include SHA-384, SHA-256, and SHA-512), 2002 August 1, http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf (last visited on 27th August 2010)

[40] RFC 4051, Additional XML Security Uniform Resource Identifiers, D. Eastlake 3rd, April 2005, http://www.ietf.org/rfc/rfc4051.txt (last visited on 27th August 2010)

[41] XML Signature Syntax and Processing Version 1.1, W3C Working Draft 04 February 2010, D. Eastlake 3rd et. al., http://www.w3.org/TR/2010/WD-xmldsig-core1-20100204/ (last visited on 27th August 2010)

[42] RFC3447, Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1, J. Jonsson et.al., http://www.ietf.org/rfc/rfc3447.txt (last visited on 27th August 2010)

[43] Specification for the Advanced Encryption Standard AES), 26 November 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf (last visited on 27th August 2010)

[44] eps e-payment standard Pflichtenheft V2.3, Joachim Geisler, Christian Matschi, March 2009, http://www.stuzza.at/1111_DE.6488C3D06b0d1db8f99c75d5082dbde7c4e3fa71 (last visited on 27th August 2010)

[45] ETSI TS 102 640-4 V2.1.2, Electronic Signatures and Infrastructures (ESI); Registered Electronic Mail (REM); Part 4: REM_MD Conformance Profiles; http://www.etsi.org/deliver/etsi_ts/102600_102699/10264004/02.01.02_60/ts_10264004v020102p.pdf (last visited on 26th March 2012)

[46] ETSI TS 102 640-6-3 V1.1.1, Electronic Signatures and Infrastructures (ESI); Registered Electronic Mail (REM); Part 6: Interoperability Profile: Sub-part 3: REM-MD SOAP Binding Profile; http://www.etsi.org/deliver/etsi_ts/102600_102699/1026400603/01.01.01_60/ts_1026400603v010101p.pdf (last visited on 26th March 2012)

[47] ETSI TS 102 640-2 V2.2.1, Electronic Signatures and Infrastructures (ESI); Registered Electronic Mail (REM); Part 1: Architecture; http://www.etsi.org/deliver/etsi_ts/102600_102699/10264001/02.02.01_60/ts_10264001v020201p.pdf (last visited on 26th March 2012)

[48] Web Services Reliable Messaging (WS-ReliableMessaging) Version 1.2, OASIS Standard, 2 February 2009, http://docs.oasis-open.org/ws-rx/wsrm/200702/wsrm-1.2-spec-os.html (last visited on 11th Feruary 2010)

All following SPOCS documents mentioned here are available at: http://www.eu-spocs.eu/index.php?option=com_processes&task=showProcess&id=18&Itemid=61

[49] SPOCS D2.2 Standard Document and Validation Common Specifications"

[50] SPOCS D3.2 Functional Specification, Architecture and Trust Model

[51] Appendix 1: Security Architecture Development Process

[52] Appendix 2: Trust-service Status List Profiling ("SPOCS TSL")

[53] Appendix 3: eDelivery Interconnect Protocol and Gateway Specification

[54] Appendix 4: eSafe – Operations in Detail

[55] Appendix 5: SPOCS TSL Accreditation and Operation Policy

[56] Appendix 6: Security Model

A. Appendix – Acceptance criteria

No.	Acceptance criteria	Norm	Process	Priority
1.	Conform to SPOCS template	• Template issued by Gov2u (9-2009)	Check against template by WP6	High
2.	Language & Spelling	• English (UK)	Review by an English language specialist by PD	Medium
3.	Readability			
	○ related aim groups			
	▪ Executive Summary (e.g. for members of the Commission or reference groups, who expect a rapid overview of the results)	• comprehensible English • Clear structure & interesting presentation • Highlight of outcomes and advantages	Review by PD	High
	▪ Main Part and Conclusion	• comprehensible for the technical experts	Review by IT Architects of SIEMENS and BOS using a cross-check method	High
	○ layout	• Structured paragraphs especially a hierarchical structure	Review by author of the deliverable	High
	○ Clearness of the sentences	• no complicated structure, no nested sentences, - in this context- no unusual vocabulary	Review by WP3 internal Quality Assurance	High
4.	Consistency with description in DoW	• Alignment between this deliverable and the DoW regarding the effected account/statements about objectives and the description of the deliverable 3.2	Examine this deliverable and the DoW and compare the affected account/statements. Review by the BOS Quality Assurance	High
5.	Consistency within the document	• Unity of assessment process regarding the building blocks • No contradictions and unnecessary repetitiveness	• To check towards assessment criteria. • To check that there are no contradictions and needless repetitiveness Review by Author D3.2 in collaboration with relevant partners (subtask leaders) e.g. the responsible people for eDelivery , eSafe, Security Architecture	High

No.	Acceptance criteria	Norm	Process	Priority
6.	Content is suitable for purpose	The deliverable fulfils the objectives of the DoW (See lists of the goals). The deliverable provides the necessary facts to inform the target groups and forms the basis of the decision about the detailed work in the next Task 3.4 (Development of appropriated specifications, see next item no 7 and see the lists of results)	To check towards the result list and usability. Reviewed by WPL in collaboration with relevant partners if needed.(e.g. author D3.1 in collaboration with the responsible people for eDelivery, eSafe, Security Architecture)	High
7.	Content is fit for use	Based on this deliverable the detailed work of task 3.3 can be planned and implemented. In detail is needed: • eDelivery specifications towards interoperability which interconnect the different eDelivery solutions with transparency and security • eSafe a standard specification for eSafe • Security and Architecture based on existing eSafe- and eDelivery-solutions and standards, a list of requirements (from D3.1) permit a standard specification for a security architecture framework. Al these results are aligned with, or based on, the solutions of the sibling LSP, as far es suitable to fulfil the objectives of SPOCS	To check towards this result list and usability. Reviewed by WPL in collaboration with relevant partners. (e.g. authors D3.2)	High

No.	Acceptance criteria	Norm	Process	Priority
		• Recommendations published by IDABC,ETSI TSL (Trust-service Status Lists) and ETSI (STF 402) "Rem Interchange..." • The dependencies of WP3 to the other WPs (especially to, WP2.) are considered.(see TA p 71 ff)		
8.	Commitment within WP	• Partners of WP are aware of this acceptance list • Commitment by every WP3-partner to the assessment statements and deliverable input.	The experts of the partners are involved in QA-Process. Review by e-mailing	High

Table 13: Acceptance Criteria

Legend:

- Deliverable - a description of the deliverable for which the acceptance criteria must be met.
- Acceptance criterion – a description acceptance criterion
- Norm – a description of the norm that is applied to measure conformance
- Process – a description of the process that is used to test conformance
- Priority – the priority to meet a acceptance criterion (Low = nice to conform to, Medium = important to conform to, High =necessary to conform to).

B. Appendix – Risk list

Threat	Consequence(s)	Measure(s)	Chance	Impact	Risk
Risks relating mainly to the developing of deliverables					
Provided contributions do not have the sufficient quality and quantity	Time and resources for revision are necessary Running behind the schedule Deliverable will not be submitted on time	The acceptance criteria list was defined before the start and agreed within the WP. This list contains e.g. the use of the SPOCS template internal quality checks of experts on important points during delivery (e.g. usability of the result) Early involvement of the Executive Board quality reviews (to save time) Controlling time line (agreed by the partners in the WP3 meeting) and reminding the partners to meet the deadlines. WP leader monitors the delivery Co-operation with PD	M	H	H
Contributions of partners/federal states contact persons are not delivered on time			H	H	H
Deliverable is not accepted by PTC, EQM, PD			M	H	H

Threat	Consequence(s)	Measure(s)	Chance	Impact	Risk
Risks relating mainly to the scope of the work package					
Not respecting deadlines	Relevant information or contribution could not be taken into account (potentially a loss of relevant content or quality aspects) Falling behind schedule Delay in progress in WP	Contact the partners and remind them Look for compromises Co-operation with PD Escalation within project	M	H	M
Poor cooperation from other stakeholders/national organisations	One sided results Potentially a loss of quality/quantity Partially missing the objectives of the WP	Further dialogue with organisations Preparing concise documents for easy response Escalation, possibly via the EU	M	M	L
Analysis is not detailed enough (we might miss some specification from non-consortium countries)		Specification task will take into account specialised assessment about further technical specifications and protocols Further countries need to be involved	M	H	L
Gateways and specifications become too complicated to implement	No real interoperability Objective missing	Development of simple architectures on time	M	H	M
Having too many systems differences in national specifications to deal with (complicated mappings and adjustment to national solutions)	Having no clear structure and not finding suitable solutions	Creation of system / specification overview, usage of electronic systems and complexity methods	H	M	M

Threat	Consequence(s)	Measure(s)	Chance	Impact	Risk
Risks relating mainly to the project in general					
No common agreement on architecture	As specifications should be based on a common architecture, there could be problems in achieving the objectives of the project	Strong current dialogue between partners, further discussion and decisions based on Athens needed	M	M	H
Different levels or different views on the implementation of the SD	Potentially the implementation of cross-over connections are not possible The SPOCS objectives could be missed	Negotiating and trying to achieve an agreement based on the common architecture solutions Raising awareness	H	M	M
Missing project acceptance by European Member States	Ultimately we would miss the objectives of the project	Public relations actions as well as clarifying actions of WP6 Ask for feedback from the Member States' view Integration of new piloting countries (extension), integration of feedback by industry, other member states, other public groups	M	M	L
Too few SPs want to use the pilot solutions					

Table 14 Risk List

Legend:

- Threat - Description of a potential danger towards the project.
- Consequence - Description of the negative effect the threat can have towards the project.
- Measure - Description of the measures that can be taken to prevent a threat from happening or to reduce negative the effects.
- Chance - Defines the likelihood of a threat to happen.
- Impact - Measure of the negative effect on the project.
- Risk - Chance * Impact, representing the priority.

C. Appendix – Risk Management

Suitable Risk Management assists in achieving the objectives on time and with sufficient quality. Again, in principle the dedicated risk management defined is applied for Deliverable D3.1, which naturally observes the WP7 Quality and Risk plan.

The types of the probable risks influence our methodology and there is a strong connection between risk management and quality actions which are typically utilised to avoid risks. Especially constructive quality actions are measures to avoid risks (complexity of gateways could not be managed and therefore interoperability is not reachable, narrow time span).

Coordinated with the subtask responsible people, the work package leader presents in each Executive Board an adjusted current risk list. Then the challenges and issues among all work packages can be managed together.

The risk list was regularly updated by WP-Leader and input of the partners was also considered.

The risks regarding the deliverable D3.2, and the risks concerning the whole WP3, had to be constantly under the work package leader's control

D. Appendix – Quality management

To make sure that the deliverable D3.2 is usable and fulfil its purpose, we follow in principle the dedicated quality process implemented for Deliverable D3.1, which of course observes the WP7 Quality and Risk plan. Some special constructive and analytical quality actions follow.

Constructive quality actions

- For usability of the later developed approaches of the different Work Packages (in particular their seamless connection) WP3 leader and partners promoted and encouraged intensively the definition of embracing processes and a general foundation of architecture with applicability to all technical work packages (elaborated by the Extended Framework Group of SPOCS).
- Collaboration meetings with other Large Scale Pilots (LSP) and programmes to ensure usefulness and capability of possible solutions
- The timeline was adapted to the necessity of the overall SPOCS schedule and was taken into account an adequate time span for SPOCS internal quality assurance.
- During the elaboration phase of D3.2, weekly conference calls ensured timeline and utility of the contributions for the deliverable D3.2.
- In the final phase the core team was responsible for completing the provided parts and ensuring consistency.

Analytical quality actions

- Each updated version of the deliverable was provided to the WP3 partners for content.approval (eDelivery, eSafe and Security Architecture) together with all WP3 partners and the Program Technical Coordinator (PTC).
- The QA was carried out internally by WP3:
- the core team revised and aligned the deliverable and completed the links and layout.
- The work package leader always monitored the process.and carried out the final, formal check of D3.2. This deliverable then passed through a multi-stage quality process, in the month before its submission to the EC.

E. Appendix – Process Model

This appendix gives a graphical overview to the SPOCS Process Model.

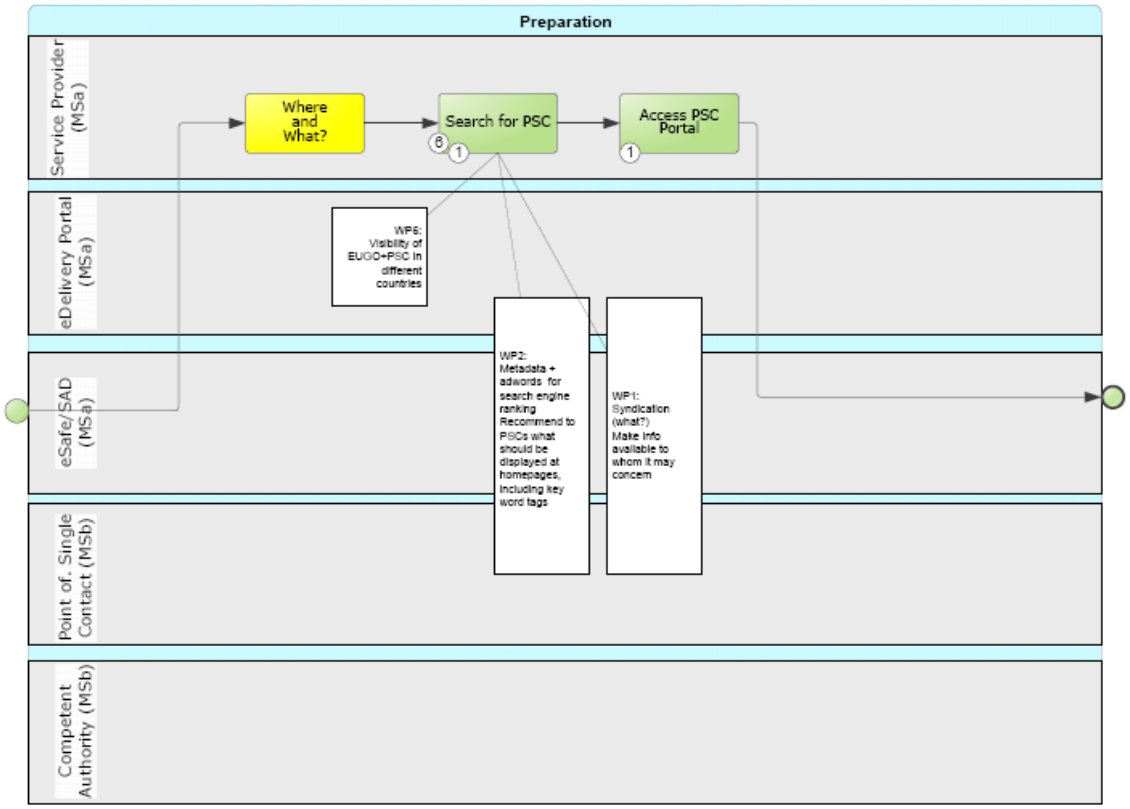

Figure 14: "Preparation" subprocess overview

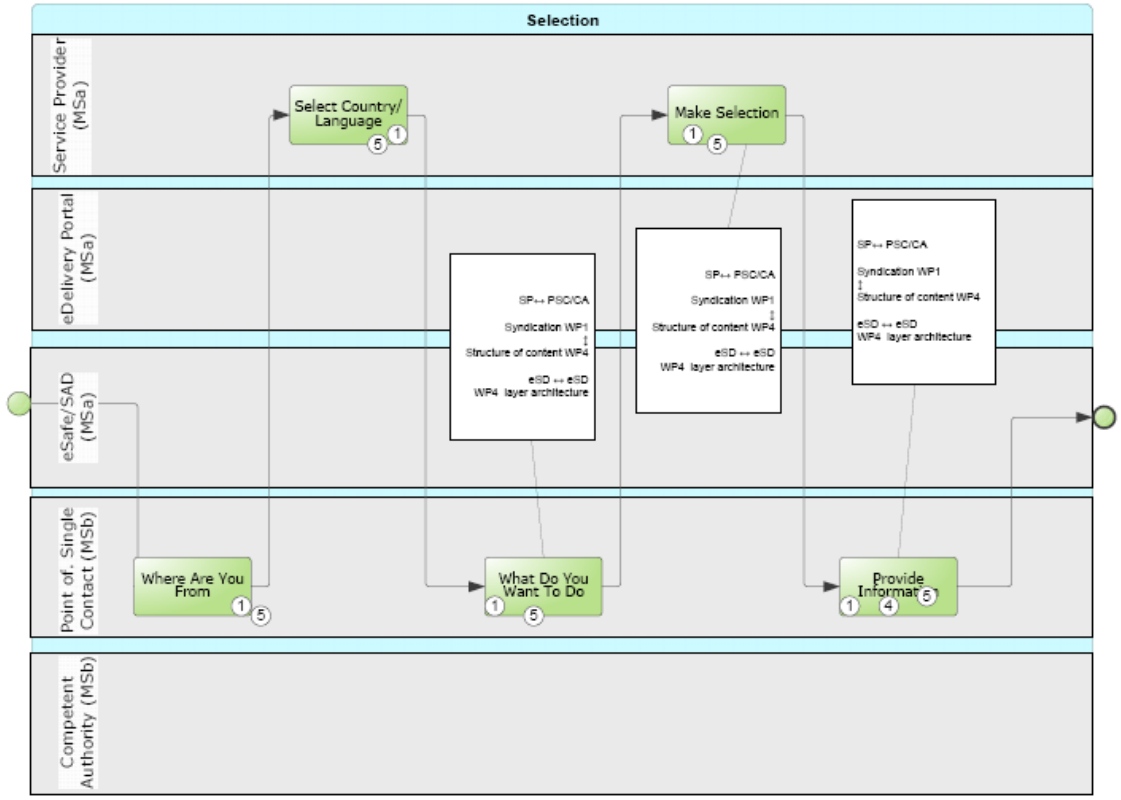

Figure 15: "Selection" subprocess overview

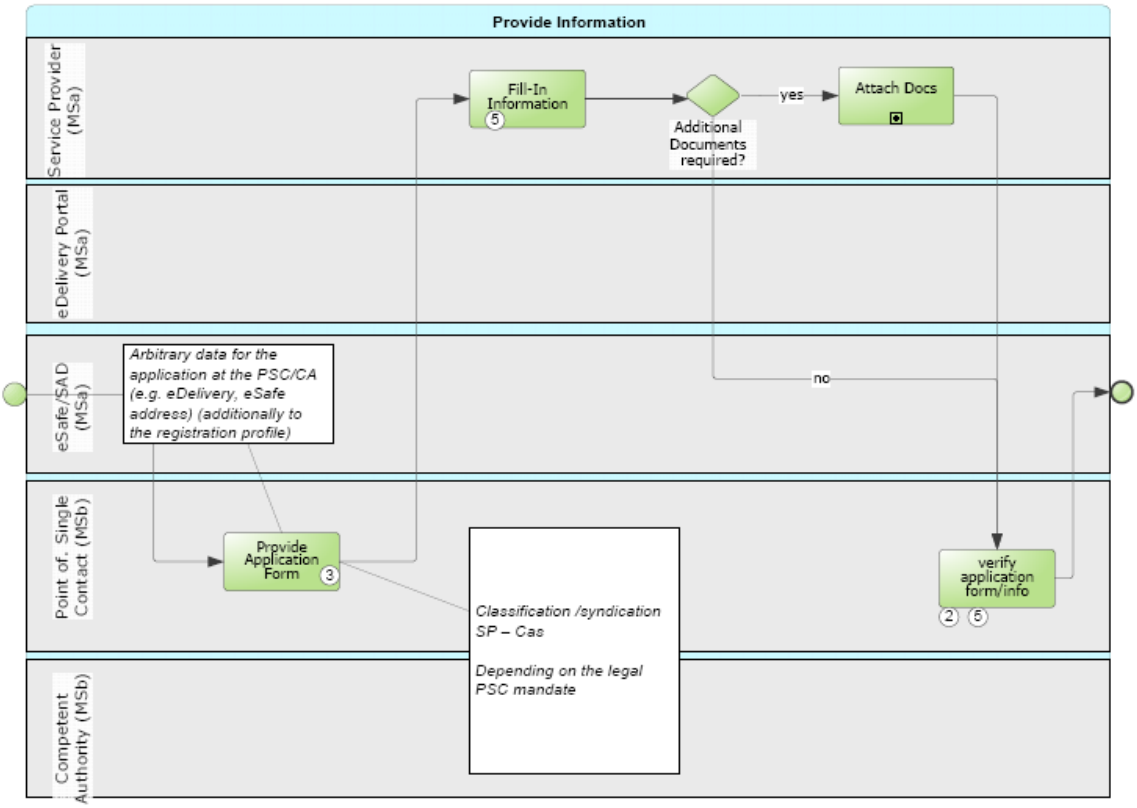

Figure 16: "Provide Information" subprocess overview

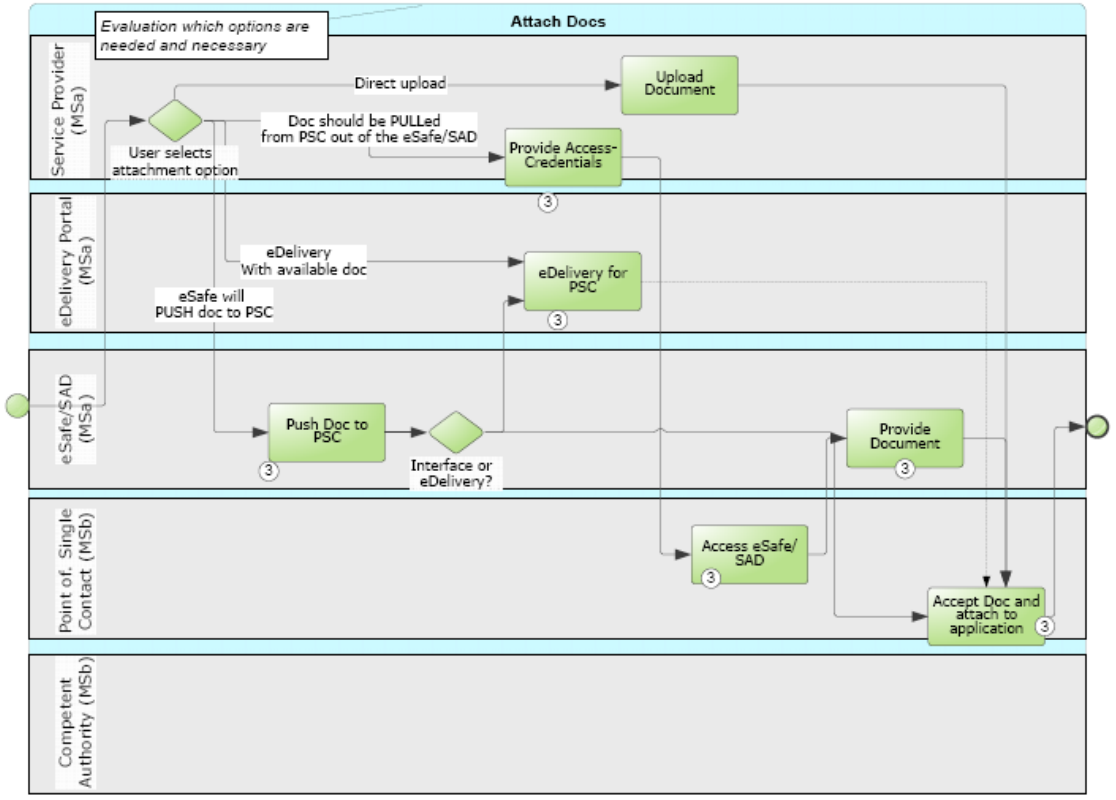

Figure 17: "Attach Docs" subprocess overview

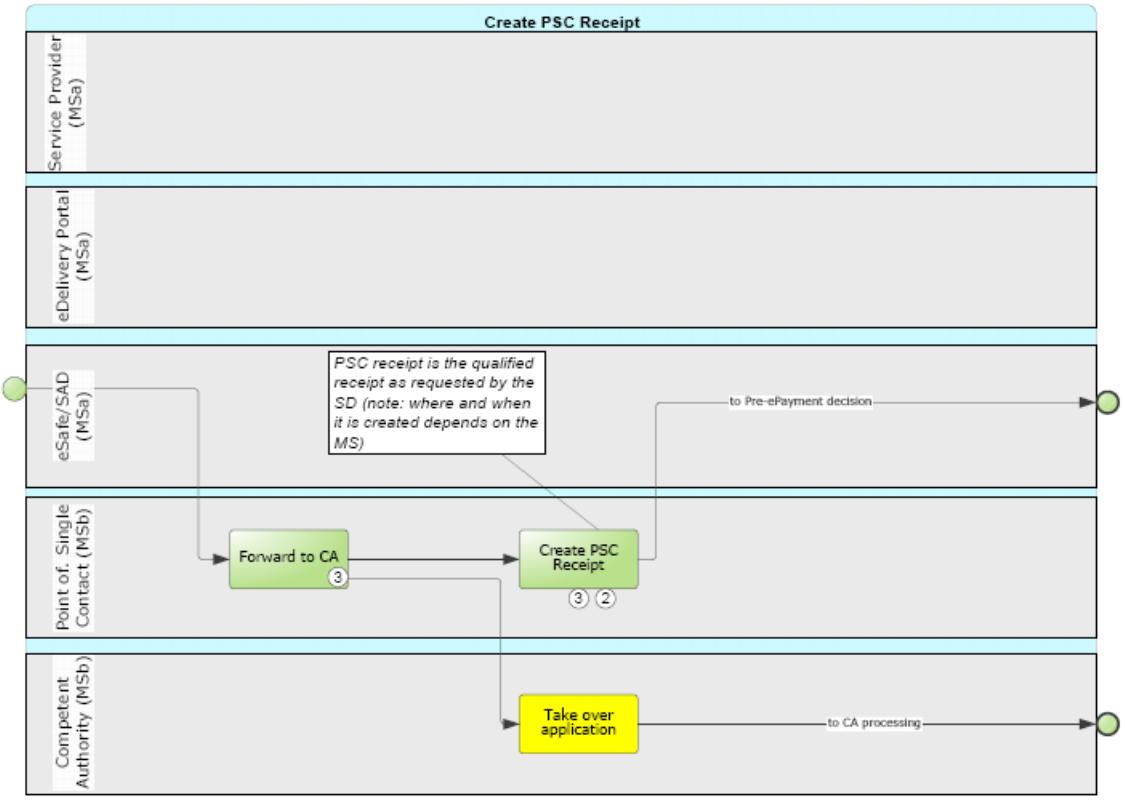

Figure 18: "Create PSC Receipt" subprocess overview

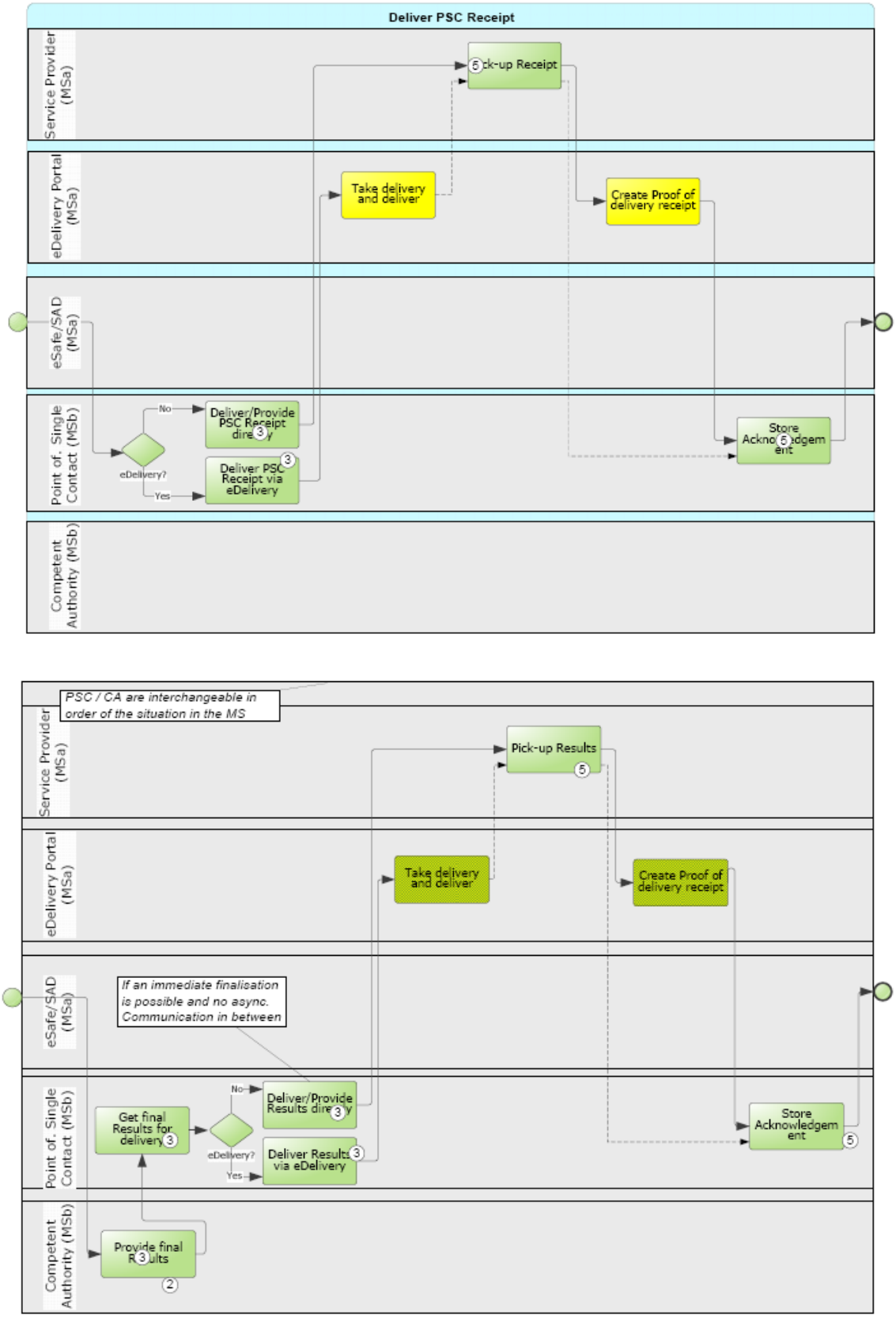

Figure 19: "Deliver PSC Receipt" subprocess overview

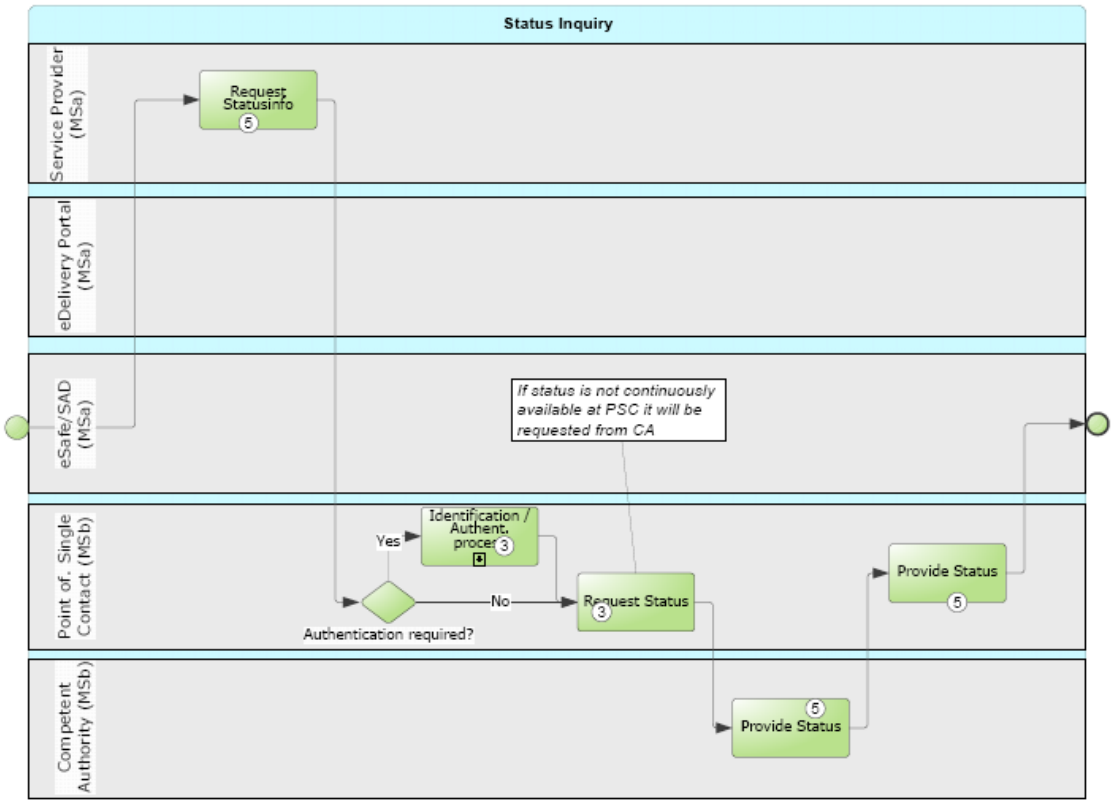

Figure 20: "Status Inquiry" subprocess overview

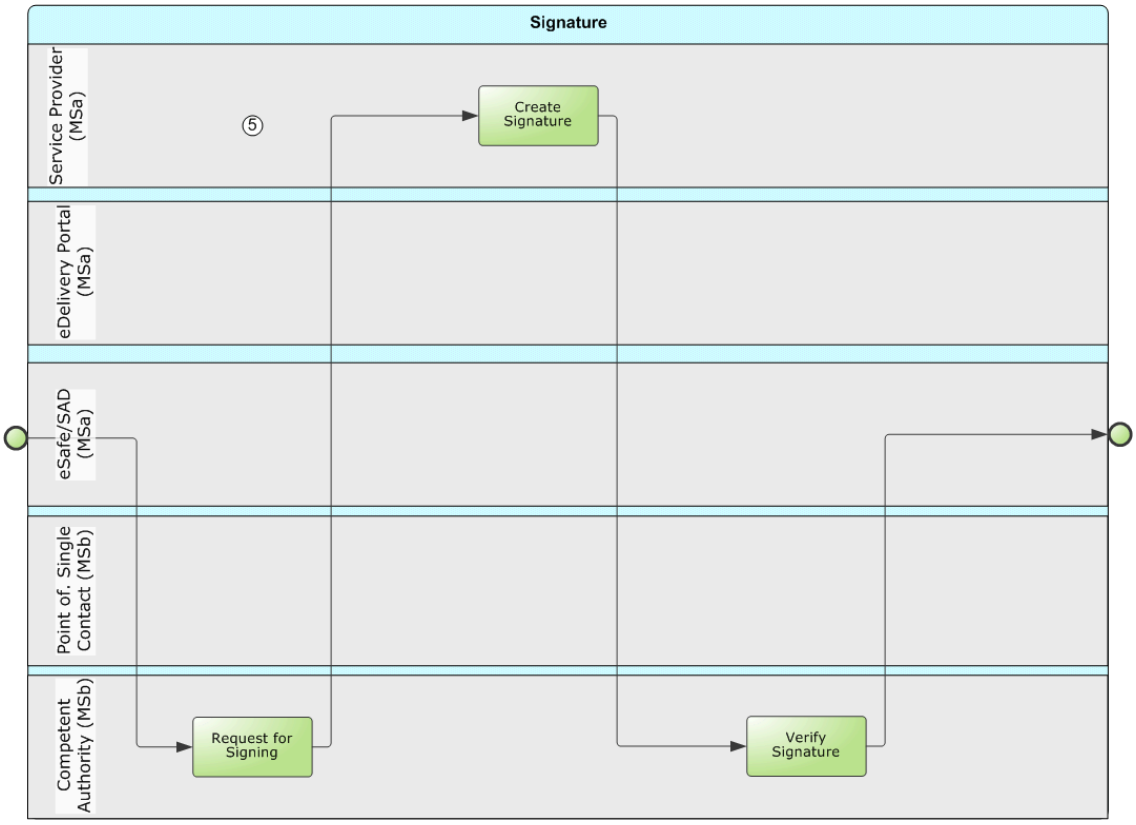

Figure 21: "Apply Signature" subprocess overview

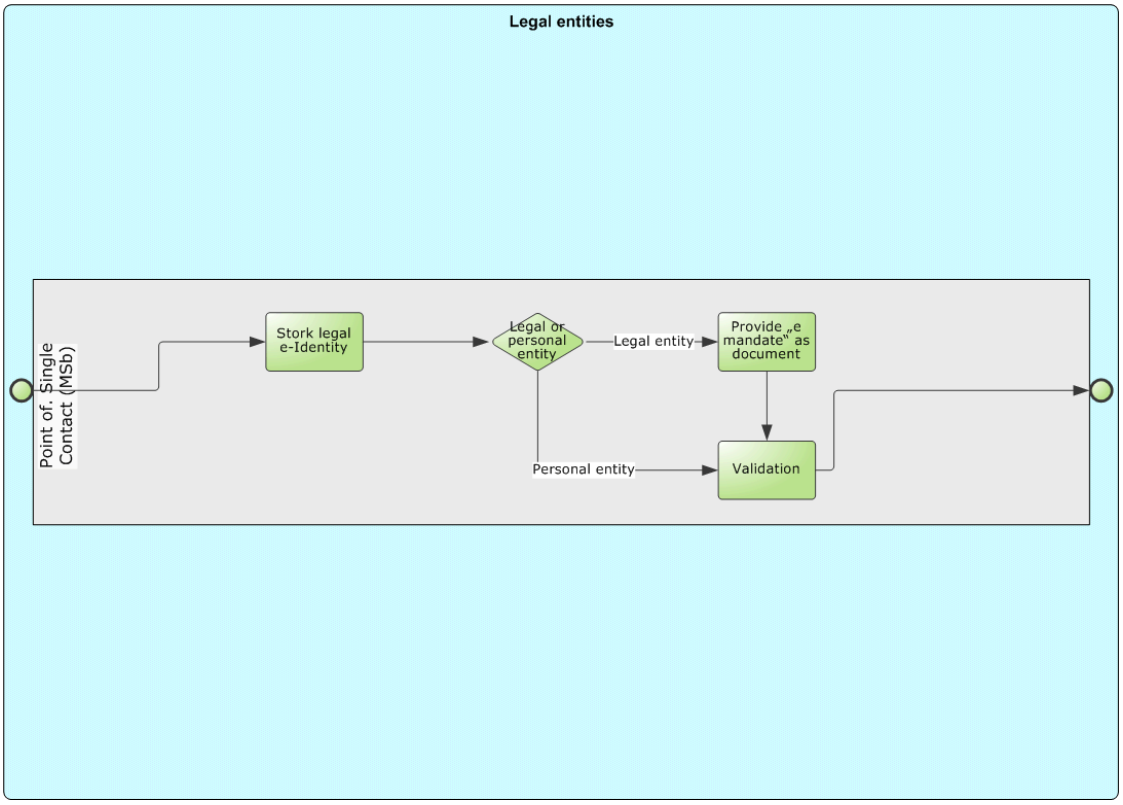

Figure 22: Legal Entities in the Process Model

F. Appendix – eSafe solutions to be integrated with SPOCS

5.1.1 Austrian solutions

eTresor for myhelp.gv.at

Currently, one eSafe service for private users exists in Austria. Access to private personal documents requires authentication by citizen card. In Austria the eSafe is a service that is completely separate from any PSC or CA portal. The eSafe as well as the PSC will support the PUSH principle. Therefore the Austrian scenario can serve as an example for the message flow across borders. The eSafe can be replaced by the eSafe implementation of any other pilot country using document PUSH principle.

The eSafe allows the authenticated user to select documents to be made available to other eSafe customers or to third parties. The documents are shown in the other user's personal space or in the so called showbox, a space secured by a generated access token.

For SPOCS, the planned modules will get access to the documents and copy them to an OCD container before the container will be transferred via the PUSH principle to the receiving module at the PSC. Since the PSC and the eSafe have established a session and the session identifier is transferred along with the pushed documents they are allocated to the correct application form. The PUSH scenario does not require the PSC to prove any access rights at the eSafe.

GOG-archives

There is also interest in integrating the GOG-archives. Currently there are some enhancements in realisation and the integration in the pilot is evaluated. If the roadmap allows the participation the GOG-archives will be integrated following the PUSH principle. All documents are electronically signed with a qualified signature from the notary or lawyer archiving the deed and are provided encrypted only for the authorised receipient. Since the security requirements are very high for the GOG-archives, the security requirements from SPOCS will be fulfilled per default.

5.1.2 Greek solution

Citizens' box provided by Ermis portal

In the Greek example of an eSafe, each registered user has a private message electronic box provided by Ermis portal, where both digitalised documents and documents produced during an administrative act can be uploaded to the user's eSafe instance. In the former case the document is uploaded either by the SP or an Citizen Service Centres' employee. In the latter case, the document is uploaded automatically if the whole process is integrated either in the Ermis portal or in the Citizen Service Centres' back office system. The documents that are provided by Ermis or the Citizen Service Centres are certified copies digitally singed, the rest of the documents are simple copies. The PSC information system that is hosted by Ermis infrastructure can automatically request the document from the eSafe as long as this is chosen by the SP.

The SP has access to all the documents in his eSafe instance and selects which documents will be accessed by the PSC. When the access is given to the PSC, the PSC may retrieve the selected documents. The PSC creates a dossier for each specific case and the SP has access to this dossier. A document is transferred from the eSafe to the PSC case management system in the predefined dossier of the case. Additionally Web

services are currently being prepared especially for making a document accessible to other systems, provided that proper access rights have been given.

5.1.3 Polish solution

eSafe on ePUAP

At present there is no fully functional Polish eSafe. There is a secure infrastructure (ePUAP platform), which supports setting up private electronic mailboxes for document exchange between CAs and citizens. It is planned to develop the Polish eSafe within the ongoing project of ePUAP including the national and cross-border interoperability features specified for the SPOCS project, without imposing further requirements on the SPOCS modules. The Polish eSafe will then be accessible directly by a web browser as well as through Web services and will fully support SPOCS open modules. However this is an independent project run by the Ministry of Interior.

Furthermore, the Polish eSafe will be integrated with the Polish Citizen ID Card called pl.id[42]. The project is in compliance with EU ID Card recommendation for eID. Every Polish citizen will have an electronic personal identification card with an electronic signature. It will allow accessing the government authority registry and any other trusted registries, which require proof of personal identification.

The pl.id together with technical infrastructure will allow the SP to attach required attestations or any other documents saved in a Competent Authority system or in eSafes. It will diminish requirements imposed on citizens to deliver requested documents in paper form.

[42] http://cpi.mswia.gov.pl/portal/cpi/38/195/plID.html (last visited on 22/08/2010)