COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME
ICT Policy Support Programme (ICT PSP)

Towards pan-European recognition of electronic IDs (eIDs)

ICT PSP call identifier: ICT-PSP/2007/1
ICT PSP Theme/objective identifier: 1.2

Project acronym: STORK

Project full title: Secure Identity Across Borders Linked
Grant agreement no.: 224993

D5.7.3 Functional Design for PEPS, MW models and interoperability

Deliverable Id :	**D5.7.3**
Deliverable Name :	**D5.7.3 Functional Design for PEPS, MW models and interoperability**
Status :	**Final**
Dissemination Level :	**Public**
Due date of deliverable :	**May 31st 2011**
Actual submission date :	**October 7th 2011**
Work Package :	**WP5**
Organisation name of lead contractor for this deliverable :	**ES-MAP**
Author(s):	**Diana Berbecaru, Eva Jorquera, Martine Schiavo, Adrian Johnston, Antonio Lioy, Arnaldur F. Axfjörð, Carlo Luyten, Carlos Ribeiro, Clemens Orthacker, Daniel Martínez, Joaquín Alcalde-Moraño, Luís Felix, Marc Stern, Mario Stoltz, Matthias Schwan, Renato Portela, Sigurður Másson, Tarvi Martens, Thomas Rössler, Wolfgang Bauer.** **Final redaction: John Heppe**
Partner(s) contributing :	**AT, BE, DE, EE, ES, FI, FR, GR, IS, IT, LU, NL, PT, SE, SI, SK, UK**

Abstract:
This document specifies functionally what the STORK Platform will do. Thus it specifies the data (definition and messages) and processes or functionalities.
This version is also meant to guide a detailed discussion between all participants, and will be revised accordingly, and also including the specifications of the interface with (MS-)specific functionalities.

Project co-funded by the European Community under the ICT Policy Support Programme

History

Version	*Date*	*Modification reason*	*Modified by*
0.1	4/10/2010	Initial version, equal to D5.7.2	J. Heppe
0.2	02/09/2011	Inclusion of explanation due to reviewer's comments. Version for Quality review	J. Heppe
Final 1.0	07/10/2011	Quality review and Finalization	A. v Overeem, S. Koppius, R. Wannee

Intermediate internal versions, e.g. for quality reviews, have been omitted.

Table of contents

List of figures

List of tables

List of abbreviations

<Abbreviation>	<Explanation>
AP	Attribute Provider
AT	Austria
BE	Belgium
DE	Germany
DOW	Description of Work
eID	electronic Identity
EC	European Commission
EE	Estonia
ES	Spain
EU	European Union
FR	France
IDABC	Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens
IDM	Identity Management
IDP	Identity Provider
IS	Iceland
MW	Middleware
MS	STORK Member State
NL	Netherlands
PEPS	Pan European Proxy Service
PT	Portugal
RA	Registration Authority
SP	Service Provider
STORK	Secure idenTity acrOss boRders linKed
STORK QAA	STORK Quality Authentication Assurance
WP	Work Package
UK	United Kingdom

Executive summary

This document is the functional design of the STORK platform, which is achieves the interoperability of electronic identifiers all over the 16 participating states.

It substitutes the original document D5.7.1 and its successor D5.7.2. This version includes explanations due to reviewer's comments on D5.7.2. All these documents are based on the findings of other workgroups, reflected especially in the deliverables D2.1, D2.2, D2.3 of WP2, and the process flows of WP4.

If we are interchanging personal data across borders, the first thing we worry about is the meaning of these data. We must guarantee that these interchanged data are usable in the destination country, allowing a European citizen to register for a service (new to him), and to authenticate in further visits. A special type of data is application logging, which is included as the last subchapter.

In the actual definition of the STORK project, the defined data include all maximum available data, which are:

- eIdentifier
- givenName
- surname
    - inheritedFamilyName
    - adoptedFamilyName
- gender
- nationalityCode
- maritalStatus
- dateOfBirth
- countryCodeOfBirth
- age
- isAgeOver
- textResidenceAddress
- canonicalResidenceAddress
- residencePermit
- eMail
- title
- pseudonym
- signedDoc
- citizenQAAlevel
- fiscalNumber

In most countries only a subset of these data is available; please refer to D5.1, chapter 5 for more details about available data in the different member states. For the future we may expect more data to be integrated in the system, requiring the connection of *Attribute Providers* in the different national implementations. If such new *Attribute Providers* can provide new attributes, these can easily be included in the definitions.

In the second place this document describes what common functionalities will be implemented, as a more detailed and complete description of the business processes as described by Work Package 4. This description is as "technology transparent" as possible, just to centre all discussions on WHAT the system will do, and not HOW this shall be done.

These common functionalities mainly describe the way a Service Provider may request his national representative to obtain a STORK identification; and how this request is passed to the citizen's country. An important point is the consent: there are 2 points in the logic foreseen for explicit user consent to send his data to a foreign organisation: before data collection (consent for data-types, no values included), and just before sending them abroad: consent includes the values.

No data is sent anywhere without user's consent.

These functionalities are presented for the 3 business processes (authentication, attribute transfer and certificate validation), each in 4 scenarios corresponding with the different architectural combinations: PEPS – PEPS, PEPS – MW, MW – PEPS and MW – MW.

The authentication and attribute transfer business processes have especially marked functions, which are the member state specific functionalities, which every member state will have to develop. The certificate validation business process has only one MS-specific function: the check on the status of the presented certificate.

1 Introduction

This document, as result of WP5.1, substitutes the original document D5.7.1[13] and its successor D5.7.2[14], which on their turn substitute the old deliverables D5.2 and D5.5. This version includes explanations due to reviewer's comments on D5.7.2[14]. All these documents are based on the findings of other workgroups, reflected especially in the deliverables D2.1, D2.2, D2.3 of WP2, and the process flows of WP4. From D2.3 we would like to highlight that the most important impact has been due to their investigation on legal restrictions around the usage of eIdentifiers (explained in chapter 2.2 of this document), as well as restrictions on sending data abroad.

Furthermore we've taken into account the architectural design, as documented in D5.1 of this same WP, and detailed it where necessary.

And, last but not least, feedback from the development is also included.

1.1 Objective

This document is the functional specifications of the STORK platform, which will achieve the interoperability of electronic identifiers all over the 16 participating states.

These functional specifications have 2 main objectives:

1. to agree with all our partners on what the system should do, at least as common functionalities and data flows
2. to form a base for the technical design.

To fulfil the first objective, this description should be functionally complete and exhaustive. As far as we could avoid describing technologies, we left them out, because they might produce distortion. And furthermore, some of the technical details still have to be decided, and this document should leave all options open for now.

1.2 New partners

This document has also been adapted due to the needs expressed by the different pilots and new member states (FI, GR, LT and SK). The fact that other partners have "left" (NL and UK remain as observer, which means that they have validated this document but will not produce their national STORK node) hasn't led to changes, thus the project would not be affected if they would decide to build their STORK node.

The incorporation of these new member states has led to some minor changes; no new business processes are defined.

1.3 Contents of the document

If we are interchanging personal data across borders, the first thing we worry about is the meaning of these data. We must guarantee that these interchanged data are usable in the destination country, allowing a European citizen to register for a service (new to him), and to authenticate in further visits.

In the second place this document describes what common functionalities will be implemented, as a more detailed and complete description of the business processes as described by Work Package 4. This description is as "technology transparent" as possible, just to centre all discussions on WHAT the system will do, and not HOW this shall be done.

The specific functionalities for each member state are described in separate but related documents; one for each country, in which the interface between the common parts and the specific parts is specified.

1.4 Scope

This document describes the common specifications of the European eID Interoperability Platform. The chapter about data describes the data and messages to be interchanged between the different components that are part of the platform (PEPSes and V-IDPs); as all components have to understand each other, these data and messages MUST be implemented this way; for other communications, each component may choose his own protocol, messages and languages to communicate with his national IDP's, AP and SP's, although we recommend the use of the same standards for the new communications, especially with their SP's. In this chapter also some paragraphs are dedicated to the user interface; these are recommendations for the user interface of the PEPS/V-IDP of the citizen, and obliged for the component in the SP's country.

The common functionalities, described in chapter 3, are common to several countries. In all countries there will be complementary (specific) functionality, and the interface is described in the national annexes; these can also be used for the interface between the national PEPS and their local IDPs and APs. These specific functionalities and local interfaces will be constructed by each member state in WP5.3.

The document defines the functional design as technology-neutral as possible. Given that actual technology choices to be made at later stages may impose constraints and given the STORK project structure where detailed pilot specifications will follow after the completion of this deliverable revision may be needed.

1.5 Approach

This document follows the ISO 12207 standard in a more or less free format. The different chapters about system development of this standard are interpreted in the following manner:

1. Implementation (of life cycle) process. This process is considered out of the scope of this project.
2. Requirement analysis. The requirements of the system are specified in the Description of Work, and the documents D2.1 – D2.3.
3. Architectural design. Due to existing infrastructures in the different countries, the architecture was imposed to this project. The design we did in D5.1 is the conceptual interoperability model, describing the communications and circles of trust.
4. Software requirement analysis, resulting in the functional specification, the actual document. According to the standard, this document should contain:
    a. Functional and capacity specifications. Functionalities are specified in **chapter 3**.
    b. External interfaces. These are specific to all Member states, so will be included in MS specific documents.
    c. Quality requirements. These are specified in subchapter 1.5
    d. Specifications for physical security. These are specific to all member states. Some general principles will be elaborated in the next phase, although responsibility will not be assumed by the project.
    e. Specifications for Access Security. This topic is interpreted as a logical security specs; which is wider then just access security. This topic is being elaborated by a special workgroup, which will deliver their results at the same time as D5.8.
    f. Specifications for human factor engineering (ergonomics). This topic is described in several points all along the complete description.
    g. Data definition and DB requirements. The data definition and the messages to be interchanged are specified in **chapter 2.** At present stage, no complex database is foreseen.
    h. Acceptance Criteria. These are specified in subchapter 1.5.

i. Other topics. Installation, operation, execution and maintenance requirements are not specified.

5. Software architecture design. Will be elaborated as part of D5.8.
6. Detailed system design. Will be specified in D5.8 Technical design.
7. Coding and unit test. Will be done in WP5.2(-1).

8-12. Integration up to implementation (installation in production environment) are to be done in WP5.2 (-2 and -3).

Thus the actual document contains a complete and exhaustive functional description of what the STORK Platform should do (specifications). As such, it is a product of many discussions the members of WP5 have had, and reflects the taken decisions on functional topics.

This document has been elaborated based on the previous documents, D5.7.1[13] and D5.7.2[14], as well as the knowledge of what has been implemented and how for the *old* member states, and knowing the requirements of the *new* member states.

1.6 Quality management & risk management

The most important topic within the quality is the assurance that all member states of this project agree on the contents of this document, and even those partners in the project who don't contribute to this WP. This assurance was already achieved in the previous version of this document (D5.7.1[13]), and the updates on D5.7.2[14] have been verified by all member states between publishing the draft version (22 July 2010) and the final version of 20 September 2010. The changes in D5.7.3 are minor, only due to the EC-review.

1.7 Glossary

The glossary can be accessed at the corporate STORK Website, clicking the following link: http://www.eid-STORK.eu/index.php?option=com_smf&Itemid=33&topic=42.0.

For readability, a brief enumeration can be found on page 9 of this document.

2 Data

As this project has as main objective to interchange personal data across borders, the meaning (and format) of these data must be agreed on. This is described in the first subchapter, and reflects the different cultures in Europe and the consensus we've achieved.

The next subchapter discusses **THE** data-item in this project: the citizen's identifier and the different variants (simple identifiers, sector identifiers, unusable identifiers and non persistent identifiers), their usage within this project and for the pilots.

Subchapters 2.3-2.5 describe various aspects about the data to interchange and the effects for the user and the service provider.

The subchapter 2.6 describes the messages to be interchanged.

Finally, subchapter 2.7 describes auditing, which must guarantee the traceability of the systems, without storing personal data. This is mainly inspired on the fact that standard logging mechanisms either don't store enough data or store too many data, especially personal data.

2.1 Data Definition

2.1.1 Data Model

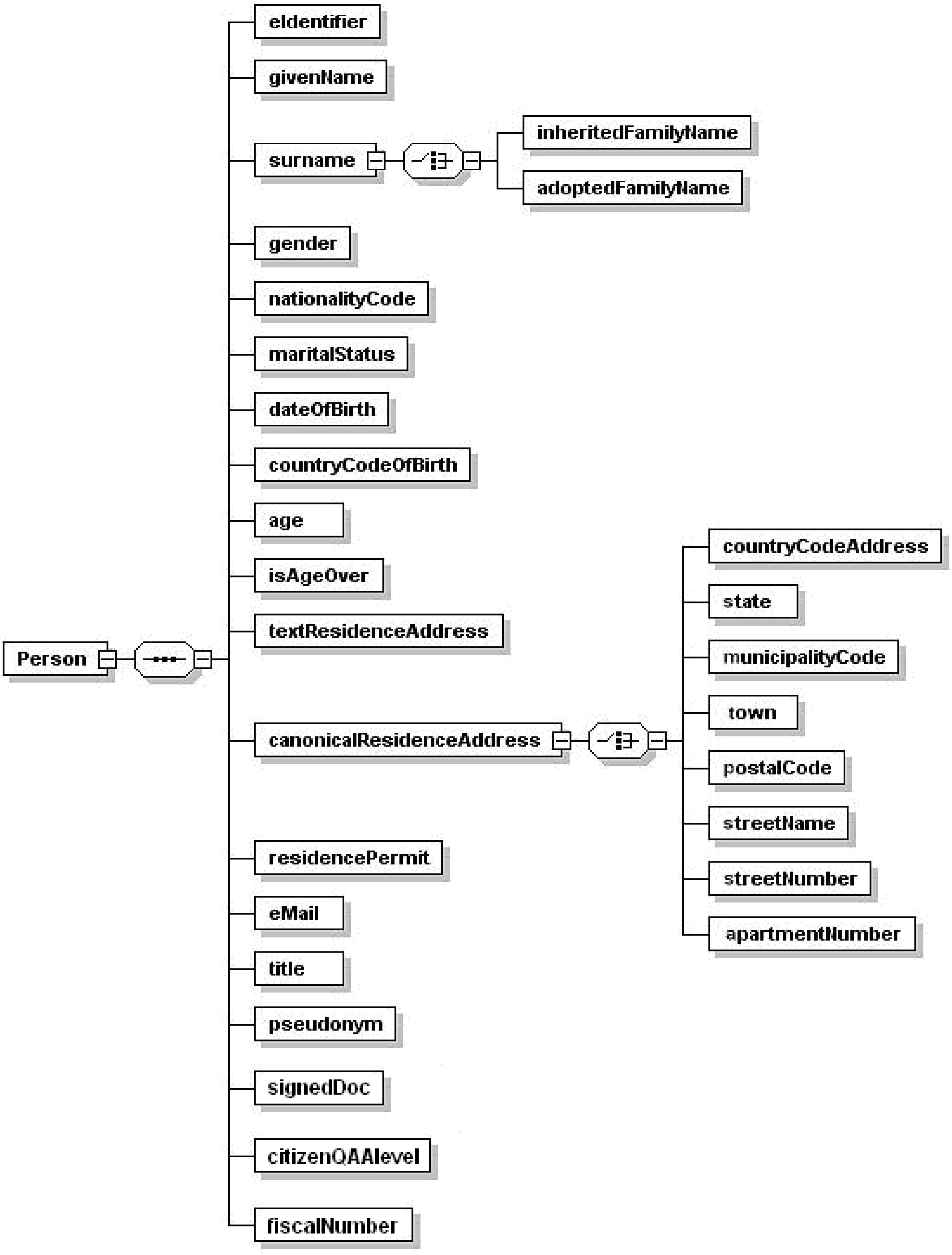


Figure 1: Data model

2.1.2 Personal Attributes

STORK uses UTF-8 as our character encoding to support multiple language.

Field	Type	Values and comment
eIdentifier	String	NC/NC/xxxxxxxxxx…. (NC=NationalityCode, the first one the country of origin of the eIdentifier, the second one the destination country)
givenName	String	
surname	String	inheritedFamilyName / adoptedFamilyName
inheritedFamilyName	String	
adoptedFamilyName	String	
gender	String(1)	F (Female) / M (Male)
nationalityCode	String(2)	*ISO 3166-1 alpha-2*
maritalStatus	String(1)	S (Single) / M (Married) / P (Separated) D (Divorced) / W (Widowed)
dateOfBirth	Date(basic format of *ISO 8601*)	YYYYMMDD / YYYYMM / YYYY
countryCodeOfBirth	String(4)	*ISO 3166-3*. Please note that this code is equal to ISO3166-1 alpha-2 in the majority of countries, but includes 4 letter abbreviations for disappeared countries. E.g. DDDE for the DDR or YGCS for Yugoslavia.
age	Number	In years (0..130)
isAgeOver	Boolean	Logically this is Boolean, in technical design another domain may be chosen
textResidenceAddress	Text	
canonicalResidenceAddress	XML	
residencePermit	String	
eMail	String	*RFC 822*
title	Text	
pseudonym	String	
signedDoc		
citizenQAAlevel	Number	[1,2,3,4]
fiscalNumber	String	

Table 1: Summary of data

2.1.3 Detailed specification for each attribute

Remind that each attribute is optional, i.e. even mandatory ones (see 2.3) in some cases may be denied by the user, (e.g. DE), may be denied by PEPS/V-IDP, or just might not be known.

2.1.3.1 eIdentifier

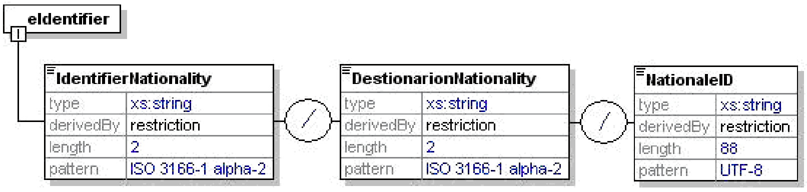
Name	Description
Field Name	eIdentifier
Definition	Cross-Border Electronic Identity Number. Is a number which uniquely identifies a person within a service. See also 2.2
Domain	String of digits, upper and lower case letters (26 letters A-Z and a-z), and "+", "/", the characters of base64. Max length: 94 (3+3+88)
Original / Derived	Original
Description	String of characters composed by three parts: • The first part is the Nationality Code of the identifier • This is one of the *ISO 3166-1 alpha-2* codes, followed by a slash ("/") • The second part is the Nationality Code of the destination country • This is one of the *ISO 3166-1 alpha-2* codes, followed by a slash ("/") • *ISO 3166-1 alpha-2* codes are two-letter country codes defined in ISO 3166 standard published by the International Organization for Standardization (ISO) to represent countries. • The third part is a combination of readable characters, which uniquely identify a person in the country of the origin of the identifier. Examples: - ES/AT/02635542Y (Spanish eIDNumber for an Austrian SP), - GB/NL/274136A (UK eIDNumber for a Dutch SP).
Graphical representation	 eIdentifier IdentifierNationality type xs:string derivedBy restriction length 2 pattern ISO 3166-1 alpha-2 / DestionarionNationality type xs:string derivedBy restriction length 2 pattern ISO 3166-1 alpha-2 / NationaleID type xs:string derivedBy restriction length 88 pattern UTF-8

Table 2: eIDNumber

2.1.3.2 givenName

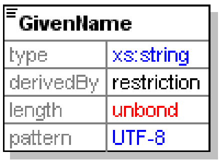
Name	Description
Field name	givenName
Definition	The primary name or given name of a person
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters. Considerations: - Consecutive spaces are not allowed. - A person can have multiple words in a given name
Graphical representation	 GivenName type xs:string derivedBy restriction length unbond pattern UTF-8

Table 3: GivenName

2.1.3.3 surname

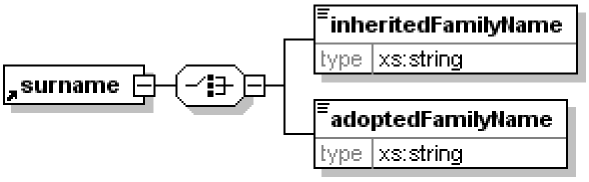
Name	Description
Field name	surname
Definition	A surname or family name or last name is the part of a person's name indicating the family to which the person belongs. The use of family names is widespread in cultures around the world. Each culture has its own rules as to how these names are applied and used.
Domain	String
Original / Derived	Derived. Adopted family name in AT, BE, DE, EE, FR, GR, IT, PT, SE, SI, SK and UK; inherited family name in ES, FI, LU.
Description	Combination of any readable character from UTF8 set of characters. For STORK, this field contains the *inheritedFamilyName* or the *adoptedFamilyName*, depending on which of them is most usual in his nationality country. Considerations: - Consecutive spaces are not allowed.
Graphical representation	 surname inheritedFamilyName type xs:string adoptedFamilyName type xs:string

Table 4: Surname

2.1.3.4 inheritedFamilyName

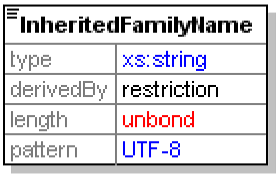
Name	Description
Field name	inheritedFamilyName
Definition	The part of a person's name that describes the family, clan, tribal group he descends from.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters. Considerations: - Any allowable character from UTF8 set of characters. - Consecutive spaces are not allowed. - A person can have multiple words in a family name.
Graphical representation	 InheritedFamilyName type xs:string derivedBy restriction length unbond pattern UTF-8

Table 5: InheritedFamilyName

2.1.3.5 adoptedFamilyName

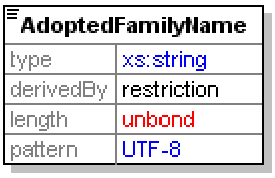
Name	Description
Field name	adoptedFamilyName
Definition	The part of a person's name that describes the family, clan, tribal group he belongs to. This family may e.g. change after a marital association.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters. Considerations: - Any readable character from UTF8 set of characters. - Consecutive spaces are not allowed. - A person can have multiple words in a family name.
Graphical representation	 AdoptedFamilyName type xs:string derivedBy restriction length unbond pattern UTF-8

Table 6: AdoptedFamilyName

2.1.3.6 gender

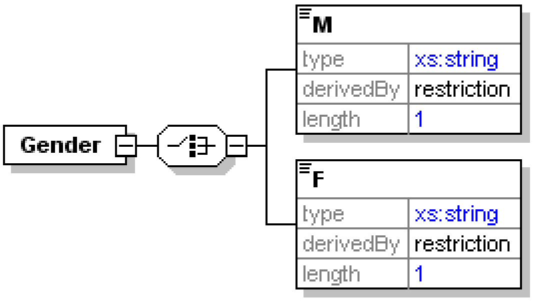
Name	Description
Field name	gender
Definition	State of being male or female
Domain	String(1)
Original / Derived	Original
Description	The following values are permitted: M and F. • "M" = Male • "F" = Female
Graphical representation	 Gender M type xs:string derivedBy restriction length 1 F type xs:string derivedBy restriction length 1

Table 7: Gender

2.1.3.7 nationalityCode

Name	Description
Field name	nationalityCode
Definition	country code of the person's nationality.
Domain	String (2). *ISO 3166-1 alpha-2*
Original / Derived	Original

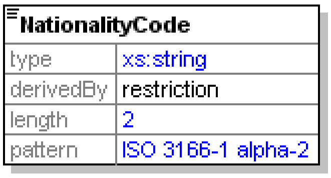
<table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td><em>Description</em></td><td><em>ISO 3166-1 alpha-2</em> codes.
<em>ISO 3166-1 alpha-2</em> codes are two-letter country codes defined in ISO 3166 standard published by the International Organization for Standardization (ISO) to represent countries.
Examples:
<table>
<tr><td>AT</td><td><em>Austria</em></td><td>BE</td><td><em>Belgium</em></td></tr>
<tr><td>DE</td><td><em>Germany</em></td><td>EE</td><td><em>Estonia</em></td></tr>
<tr><td>ES</td><td><em>Spain</em></td><td>FI</td><td><em>Finland</em></td></tr>
<tr><td>FR</td><td><em>France</em></td><td>GR</td><td><em>Greece</em></td></tr>
<tr><td>IS</td><td><em>Iceland</em></td><td>IT</td><td><em>Italy</em></td></tr>
<tr><td>LT</td><td><em>Lithuania</em></td><td>LU</td><td><em>Luxembourg</em></td></tr>
<tr><td>PT</td><td><em>Portugal</em></td><td>SE</td><td><em>Sweden</em></td></tr>
<tr><td>SI</td><td><em>Slovenia</em></td><td>SK</td><td><em>Slovakia</em></td></tr>
</table></td></tr>
<tr><td><em>Graphical representation</em></td><td>
NationalityCode
<table>
<tr><td>type</td><td>xs:string</td></tr>
<tr><td>derivedBy</td><td>restriction</td></tr>
<tr><td>length</td><td>2</td></tr>
<tr><td>pattern</td><td>ISO 3166-1 alpha-2</td></tr>
</table></td></tr>
</table>

Table 8: NationalityCode

2.1.3.8 maritalStatus

Name	Description
Field name	maritalStatus
Definition	This field contains an indicator to identify the legal marital status of a person.
Domain	String(1)
Original / Derived	Original
Description	The following values are permitted: - “S” = Single - “M” = Married or legally equivalent - “P” = Separated - “D” = Divorced - “W” = Widowed

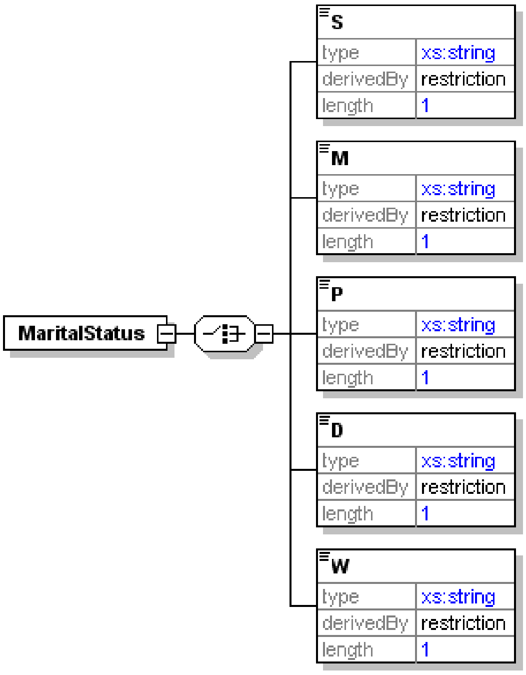
Name	Description
Graphical representation	 MaritalStatus S type xs:string derivedBy restriction length 1 M type xs:string derivedBy restriction length 1 P type xs:string derivedBy restriction length 1 D type xs:string derivedBy restriction length 1 W type xs:string derivedBy restriction length 1

Table 9: MaritalStatus

2.1.3.9 dateOfBirth

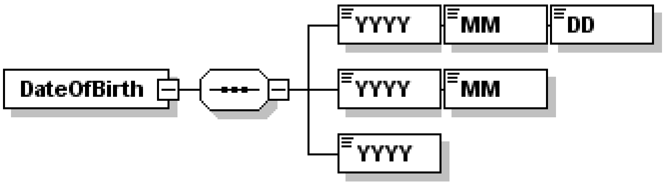
Name	Description
Field name	dateOfBirth
Definition	The date on which a person was born or officially has been deemed to be born.
Domain	Date (basic format of *ISO 8601*)
Original / Derived	Original
Description	Format is the day, month, year and century, or a combination of these elements. The date (in ISO 8601 format) must be YYYYMMDD, YYYYMM or YYYY. [YYYY] indicates a four-digit year, 0000 through 9999. [MM] indicates a two-digit month of the year, 01 through 12. [DD] indicates a two-digit day of that month, 01 through 31. Or said in another way, Date of birth is xs:union memberTypes=xs:date xs:gYearMonth xs:gYear
Graphical representation	

Table 10: DateOfBirth

2.1.3.10countryCodeOfBirth

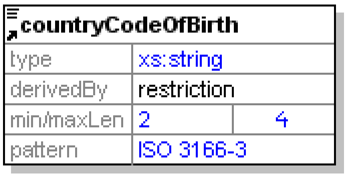
Name	Description
Field name	countryCodeOfBirth
Definition	country where someone was born.
Domain	String (4). *ISO 3166-3*
Original / Derived	Original
Description	*ISO 3166-3* *ISO 3166-3* codes are two-letter country codes defined in ISO 3166-1 standard for existing countries, and 4 letter codes for countries, which don't exist anymore. This standard was published by the International Organization for Standardization (ISO) to represent countries.
Graphical representation	 countryCodeOfBirth type xs:string derivedBy restriction min/maxLen 2 4 pattern ISO 3166-3

Table 11: CountryCodeOfBirth

2.1.3.11age

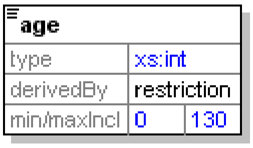
Name	Description
Field name	Age
Definition	The field contains a number with the age (number of years) of the person.
Domain	Integer (0..130)
Original / Derived	Derived (from *dateOfBirth*) The age is the difference (in years) between current date and *dateOfBirth*.
Description	Considerations: - If *dateOfBirth* is of type "YYYYMM" or "YYYY" then the *age* is an empty value. - If *dateOfBirth* is empty, *age* is empty.
Graphical representation	 age type xs:int derivedBy restriction min/maxIncl 0 130

Table 12: Age

2.1.3.12isAgeOver

Name	Description
Field name	isAgeOver

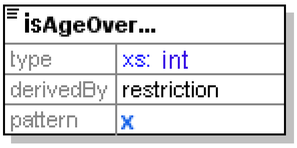
<table>
<tr><td><em>Definition</em></td><td>Is age X years or more?</td></tr>
<tr><td><em>Domain</em></td><td>Boolean</td></tr>
<tr><td><em>Original / Derived</em></td><td>Derived (from <em>age</em>)</td></tr>
<tr><td><em>Description</em></td><td>At the request an <em>age X parameter</em> is given
(<em>see D.5.8. Technical Design for details)</em>
The result it’s a number field, possible values:
<table>
<tr><td>X</td><td>The <em>age</em> is X years or more</td></tr>
<tr><td>empty</td><td>The <em>age</em> is less than X years</td></tr>
</table>
Considerations:
- If <em>age</em> is empty, <em>isAgeOver</em> is empty.</td></tr>
<tr><td><em>Graphical representation</em></td><td>
isAgeOver...
type xs: int
derivedBy restriction
pattern x</td></tr>
</table>

Table 13: IsAgeOver

2.1.3.13textResidenceAddress

Name	Description
Field name	textResidenceAddress
Definition	The address of a postal delivery point. This is usually a building and usually comprises a name/number, street, town and county/state/province.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters, including <newline>s. The format of this text field must be in the original country address style. It is a free text with 5 lines maximum.
Graphical representation	textResidenceAddress type xs:string

Table 14: TextResidenceAddress

2.1.3.14canonicalResidenceAddress

Name	Description
Field name	canonicalResidenceAddress
Definition	The address of a postal delivery point.

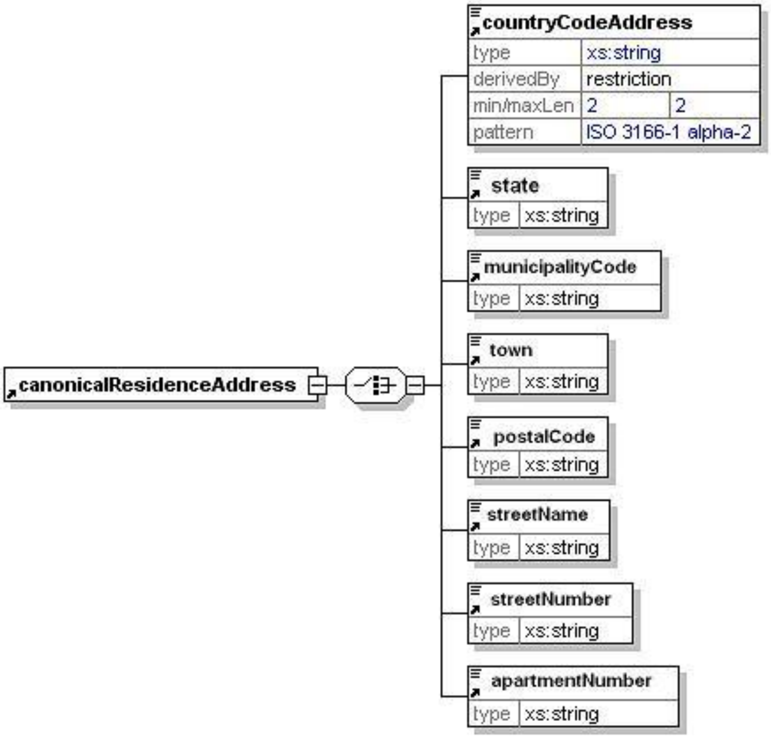
Domain	XML
Original / Derived	Original
Description	See also *Specification of the Address Canonical Data Model* (STORK Work Package 6.5)
Graphical representation	 countryCodeAddress type xs:string derivedBy restriction min/maxLen 2 2 pattern ISO 3166-1 alpha-2 state type xs:string municipalityCode type xs:string town type xs:string postalCode type xs:string streetName type xs:string streetNumber type xs:string apartmentNumber type xs:string canonicalResidenceAddress

Table 15: TextResidenceAddress

2.1.3.15residencePermit

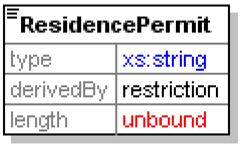
Name	Description
Field name	residencePermit
Definition	It is a text field for storage country information about residence permits.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters.
Graphical representation	 ResidencePermit type xs:string derivedBy restriction length unbound

Table 16: ResidencePermit

2.1.3.16eMail

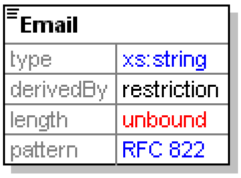
Name	Description
Field name	eMail
Definition	A person’s email address
Domain	String (RFC 822)
Original / Derived	Original
Description	The field contains Internet Email Address according to the grammar laid out in RFC 822 (Standard for the format of ARPA Internet text messages). Considerations: - Note that RFC 822 limits the character repertoire to ASCII.
Graphical representation	 Email type xs:string derivedBy restriction length unbound pattern RFC 822

Table 17: email

2.1.3.17 title

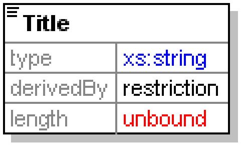
Name	Description
Field name	title
Definition	Academic or noble titles according to the country that issued the ID.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters. It is a free text. Considerations: - Any readable character from UTF8 set of characters.
Graphical representation	 Title type xs:string derivedBy restriction length unbound

Table 18: title

2.1.3.18pseudonym

Name	Description
Field name	pseudonym

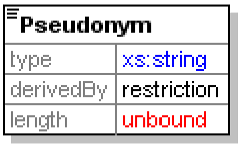
Definition	Personal pseudonym, religious name or stage name. E.g. Madonna, Prince, Benedictus XVI or Marilyn Monroe.
Domain	String
Original / Derived	Original
Description	Combination of any readable character from UTF8 set of characters. It is a string with 70 characters maximum. Considerations: - Any readable character from UTF8 set of characters.
Graphical representation	 Pseudonym type xs:string derivedBy restriction length unbound

Table 19: Pseudonym

2.1.3.19signedDoc

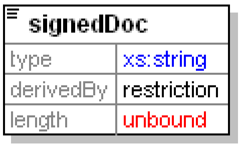
Name	Description
Field name	signedDoc
Definition	At the request a string is given. The result is the previous string signed.
Domain	String
Original / Derived	Original
Description	*See D.5.8.2bInterface Specification for detail.*
Graphical representation	 signedDoc type xs:string derivedBy restriction length unbound

Table 20: signedDoc

2.1.3.20citizenQAAlevel

Name	Description
Field name	citizenQAAlevel
Definition	QAA Level with which the citizen has been authenticated.
Domain	Integer (1)
Original / Derived	Original
Description	*See D.5.8.2bInterface Specification for detail.*

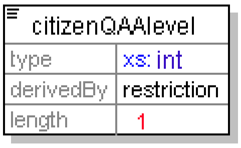
Graphical representation	 citizenQAAlevel type xs: int derivedBy restriction length 1

Table 21: citizenQAAlevel

2.1.3.21fiscalNumber

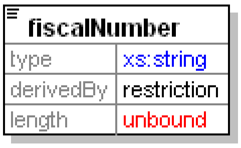
Name	Description
Field name	fiscalNumber
Definition	Number which identifies the person in the national tax administration.
Domain	String
Original / Derived	Original
Description	Fiscal number.
Graphical representation	 fiscalNumber type xs:string derivedBy restriction length unbound

Table 22: fiscalNumber

2.2 Details about identifiers

THE data item in STORK is the person's identifier. There is no guarantee that normal personal data (name, surname, dateofbirth, …) can always identify a person. The more data items we add, the fewer *data twins* we'll get, but still no guarantee for uniqueness is given. Furthermore, even if this was true, some of these data may change, which make them less suitable as items to identify a person.

For these reasons, all actual STORK member states have defined identifiers to substitute the use of normal personal data for this purpose. So all administrative information systems, as well the manual ones, as the automated ones, use such numbers. Originally every administration had its own number (employee-number, student-number, etc.), but nowadays there is a trend to unify this number to something like a "national persons number".

As there are many cultural al legal differences (as explained in D2.3) between the member states, this chapter clarifies the differences, solutions and implications. This chapter isn't meant to discuss the cultures or laws, and supposes that the user gives his consent to use the data item.

2.2.1 Simple identifier

Several countries use one and only one identifier for any purpose. Frequently this number is used by the tax office as primary user identifier, and the social security and other administrations often also use this ID as a primary identifier.

This scheme is the simplest one, as this ID can be used *just as it is* in foreign countries. It is a persistent and unique ID, and the same one all over Europe.

2.2.2 Sector identifier

Many countries use sector identifiers. I.e. a company of sector 1 will identify a person with one number, while a company of sector 2 will use a different number to identify the same person. The purpose of the number determines the value; so in a personnel administration the employee's fiscal number will be known.

In STORK new sectors will be defined, having one "sector" for each country. Some countries will allow for (though not oblige to use) unique identifiers for each service provider. Thus this solution offers a persistent and unique ID, which is different between different countries.

2.2.3 Non usable identifiers

Not all identifiers used at the national level can be used outside their country of origin; some have legal constraints, others have policy constraints, which make them unusable outside the country's frontiers. And even if they could be used it may be desirable to use an alternative from the privacy point of view.

2.2.3.1 General principles

The following general principles will be used:

1. Up to the point where data actually crosses a border between countries participating in STORK, the legal rules of the country of origin will apply.
2. The user data is sent to another country only after explicit consent of the user. That consent is given in a specific context and for a specific purpose. This consent cannot be interpreted as an implicit permission to use that data in other contexts or share it with other parties.

3. After the data has arrived in the destination country, the regime of this country will govern the further use the data. However, the minimal interpretation of the consent given by the user requires that the usage of this data is restricted as far as possible, within the bounds of this regime.

2.2.3.2 Alternatives for national identifiers

The obvious alternative for using national identifiers is to use ID-pseudonyms: identifiers generated especially for the purpose of exchanging data with other countries. ID-pseudonyms will be created and maintained by the PEPS in each STORK member country using a PEPS.

In fact, pseudonyms can be preferred to the direct use of national identifiers for the following reasons:

- It limits the possible use of the data; sharing the data with organisations in other STORK countries, including organisations in the originating country, is less trivial (though clearly not impossible).
- It serves as an isolation layer between the national numbering systems and the STORK participants in other countries. Even if a numbering system in any given country has a lifespan of 50 years, the laws of statistics indicate that every 2 years one of the EU member states will see a need for an overhaul of their national numbering system. If ID-pseudonyms are used, such a change can be carried through without side effects for the other STORK participants.

The following requirements and considerations apply to the use of pseudonyms:

1. National identifiers will be used cross-border only if the rules of the originating country require this or if the use of ID-pseudonyms presents insurmountable difficulties. In all other cases ID-pseudonyms will be used.
2. ID-pseudonyms shall be unique. This is obvious; it implies that a pure random algorithm to generate the pseudonyms is not sufficient and that a number, once assigned to a person, will not be re-used for another person.
3. ID-Pseudonyms shall be invariable. This implies that once a number has been assigned to a person, if that person is identified again through STORK the same number will be used to designate the person – even if years have elapsed between the two moments that STORK is used. Service providers can rely on this: they will receive the same pseudonym in each contact with the person in question.
4. A full history may be required for ID-pseudonyms. There may be exceptions where rule 3 cannot be maintained and a new number must be assigned to a person. To allow such a person to be traced over multiple visits, it is essential that the service which requested the authentication gets an indication that the ID has changed. The easiest way to implement this is to have the PEPS maintain a full list of historical ID's, and to deliver those as additional attributes in the identification or get_attributes request. Since changing an ID is a relative rare occurrence, this will not generate a significant overhead in the communication. For privacy reasons, such a history should be avoided.
5. Different ID-pseudonyms will be generated by a PEPS for different countries. This aids to implement general requirement 3 above, since it limits the possibilities to share the data between different STORK membership countries.
6. ID-pseudonyms should preferably consist of digits only. This is generally more efficient when using pseudonyms as keys in a database and allows various numerical algorithms to be used to generate and manipulate pseudonyms.

7. The first two positions of a pseudonym will be reserved for a code that indicates the originating PEPS. This is the easiest way to insure that there will never be a collision between to pseudonyms issued by two different PEPS systems.
8. It is acceptable if the destination country (and where applicable the sector – see next paragraph) can be derived from a pseudonym. Otherwise, the number used as an ID should be meaningless. Especially, there should be no algorithm to correlate one country specific ID with another ID for the same person.
9. The definition of a pseudonym should allow sufficient room to generate all possible combinations of source and destination country for all inhabitants of the EU in the foreseeable future. This requires a minimum of 10 digits. In combination with requirement 7 this gives a minimum length of 12 digits for the pseudonym. In alignment with the SAML-2 guidelines for identifiers, a maximum length of 32 digits is proposed.

2.2.3.3 Example: the proposal for the Dutch PEPS

Note: this is a proposal, strictly for illustrative purposes. There is no guarantee that a Dutch PEPS, if and when it is built, will actually use this algorithm.

The most common identifier in the Netherlands is the BSN (Citizen’s Service Number). Legislation for the BSN does not explicitly cover usage of the BSN outside the Netherlands, but the wording suggests that is intended for use inside the Dutch government only. Thus, in accordance with rule 1 we will use pseudonyms for communication with foreign SP’s.

The legislation does not limit the usage of BSN’s to Dutch nationals or even Dutch residents. Any person doing official business with a branch of the Dutch government can be assigned a BSN. Since most information systems in the government are built around the BSN as a primary key, acceptance of STORK will be much quicker if we do in fact assign a BSN to all people using STORK.

The pseudonyms generated by the Dutch PEPS will be 24 positions long and generated as follows:

Position	Usage
1-2	Code indicating the Netherlands (where the pseudonym was generated)
4-5	2 digit country code indicating the country where the ID and corresponding attributes are being sent to
7-12	6-digit sector code. If no sector code is present in the authentication request or attribute request, zeroes will be used,
13-22	10-digit random number
23-24	2 check digits, generated using the IBM algorithm[1]

Table 23: Proposed composition of Dutch eID for international use

[1] See e.g. http://augustana.ab.ca/~mohrj/algorithms/checkdigit.html

2.2.4 Non persistent identifiers

Most of the ID numbers are persistent, i.e. a person will have the same number during all of her/his life, at least within the scope of its usage (country, sector, service provider). Nevertheless, due to legal restrictions, in some countries, e.g. Germany and Greece, the ID number identifies the *eID card*, and not the person. So, when the card expires or gets lost, a new card with a new ID number is issued.

In their user registration procedure, non national Service Providers have to take this into account. For most critical applications, they should register more attributes of this person in addition to basic data (name, surname, date of birth, nationality, address), like for instance place of birth, legal address in home country or other data. For less critical applications they should at least foresee that the eID number may not be permanent.

Note: This consequence must be made clear to all Service Providers from other countries then Germany and Greece.

2.3 Mandatory or optional attributes

In many cases, Service Providers need to know about the values of some user attributes in order to perform his registration, to determine his capacity of carrying out a specific act, or simply to allow the user to receive the service that they are providing. Those attributes could be provided by three possible ways:

- Service Provider could ask for the required attributes to the user through a web form;
- Some required attributes can be found into the credential used by the user during the authentication phase;
- Service Provider could ask for concrete attributes to an Attribute Provider of trust in the country of origin of the user.

From the Service Provider point of view, asking for the attributes directly to the user is the simplest way to obtain them. Through a web form, users can plead information about them, and Service Provider could assume these data are true for some purposes, but he doesn’t have any guarantee.

The use of qualified digital certificates during the authentication process solves this situation partially. Digital certificates are issued by organizations which grant the validity of their content. For instance, the legal representative of a company in Spain could demonstrate his attribution by using a digital certificate of representative issued by the Spanish Chamber of Commerce. In this case, the attribute “legal representative” is ensured not only by the user, but also the Chamber of Commerce.

A similar guarantee is offered by the Member States participating in STORK: with independence of the method of authentication, they guarantee that the delivered data correspond with a real person, who is traceable.

Within usual considerations of minimal disclosure, the Service Provider, when requesting some user’s data through the STORK platform, may mark some of them as mandatory, being the others optional. Mandatory attributes imply that, in his business model, and for the requested function, a user (e.g. customer, supplier, employee) can’t exist if this data item is unknown.

E.g. for an employee it’s required to know his given name, family name and date of birth, when gender might be relevant, though not required. In such a case, the request for the first three mandatory attributes, and the last one optional implies that:

1. The user will not be allowed to uncheck any of these attributes. If he doesn’t want to send one of them, he may cancel the transaction, and not any attribute will be sent.
2. If any mandatory attribute is not found, no data will be sent to the Service Provider.

2.4 Derived data – flexibility and parameters

Whenever a member state provides authenticated personal attributes of its citizens, it should do so observing the minimum disclosure security principle. This principle states that a SP should be given the minimum set of attributes required to fulfil the service, even if under the law and the user consent a wider set of attributes could be provided.

Some of the attributes stored in the information systems of each member state have a higher precision than necessary for most SP. For instance, the information system of every member state possess a “date of birth” attribute for their citizens, however a service provider may only require to know if a user is over 18, which is a much less precise information about the same attribute.

STORK will provide mechanisms to comply with the minimum disclosure security principle (MDP) by defining queries for attributes with several levels of precision for the same data.

2.4.1 Policy establishment and fulfilment model

In STORK there are four major policy makers: the service provider and its member state, the citizen and the citizen’s state.

In some countries (e.g. UK and DE) the SP’s Member State enforces the minimal disclosure with a profile for each SP.

It will be very difficult for the citizen’s state to enforce the minimal disclosure because it would need to know in detail the business model of every foreign SP.

The SP is encouraged to the MDP by requiring only the attributes needed to perform the service with the necessary precision.

The least protective method is based on the fact that the citizen requiring the service knows in detail the business model of the SP he’s accessing and is able to enforce the MDP: he can reject the excessive attributes individually, as long as they’re optional. Mandatory attributes can’t be rejected individually: if the citizen considers any mandatory attribute excessive, then he can reject the complete request.

When a SP receives an incomplete reply to his request, he’s free to request these data from the user. Thus he would obtain them, but from a not validated source. Depending on each business case, this might be a valid procedure.

2.4.2 Implementation of the model

Usually, the member states do not have citizen’s attributes with different levels of precision in their information systems. Only the most precise attribute of each type is stored, the others must be computed out of most precise one. In STORK the PEPS and V-IDP module will be responsible for computing this type of attributes.

The PEPS/V-IDP will have two main responsibilities:

1. On receiving a query, the Citizen’s PEPS/V-IDP should verify if the required attribute is available in the attribute provider (AP) and if not, translate it to one available, by increasing the precision.
2. On receiving the attribute from the AP, the PEPS/V-IDP should compute the requested attribute request out of the received, more precise, attribute value.

2.4.3 Precision aware queries

STORK has identified several potential types of attributes with different levels of precision:

1. Date of birth;
2. Nationality;
3. Place of birth;
4. Residency;

Of these types, only the date-of-birth related attributes will be derived. This is the most obvious one. The number of different levels of precision that this attribute may have is very high, for instance the SP may ask if the user is above or below a certain age, and that age may be any number. To cope with this diversity STORK may define three types of queries with parameters:

1. Is age-above <value>
2. Is age-below <value>
3. Is age-between <value-A> <value-B>

The answer to any of these queries is <yes> or <no>

In the actual project we'll implement just the first query; the second is just the opposite of it and the third can be solved with 2 queries, which seems less complex, as well for STORK as for Service Providers.

Please note that isAgeOver is logically a Boolean, but in the technical design another domain may be assigned.

2.5 User consent and language considerations

2.5.1 User consent

In most member states, the privacy legislation requires that the user gives his consent to the use of his data. But the explanation of this requisite, and thus its implementation may be very different from one MS to another MS.

So this general objective to request the consent of the user to send his/her attributes to a Service Provider in another Member State leads to the following consent-schemes. The consent is requested by the PEPS or by the SPWare/Middleware of the user's MS.

There are three possible cases :

1. The requested attributes (types) are displayed and the user's consent is given by just choosing the attributes he/she allows to transfer.
2. The obtained values of the requested attributes are displayed and the user's consent is given by just choosing the attributes he/she allows to transfer.
3. The requested attributes are not displayed because the user's consent is not necessary. It was given for example when the user registered to the ID Provider.

2.5.2 Language considerations

Although the system aims at being as neutral and user-friendly as possible, using symbols rather then text, sometimes the use of text is unavoidable. The most intensive dialog between a user and the STORK Platform is implemented at the PEPS or V-IDP of the Member State of his eID. Owning an eID of a country supposes a certain knowledge of the native language of this system.

This means that every dialog between the user and his PEPS/V-IDP should be in its native language; especially the consent (attribute names and values), and IDP selection. On the consent-request the only data item which can't be translated will be the name of the Service Provider.

The only dialog between the user and the foreign system is the country selection: the name of the MS is written in the MS language (e.g. Deutschland for Germany). So this list will be multilingual. To increase the ease of use, it is recommended also to display the flag of the country.

A common look&feel will be defined for all messages displayed by PEPSes and V-IDPes, at least for the dialog with the foreign system, which will be used by every EU citizen. Nevertheless, the implementation of this look&feel is a member state specific responsibility, so its implementation is just recommended.

2.6 Messages

The following messages are interchanged between the different parties in the communications. The description is functional, i.e. specifies the main data to be interchanged, and nothing about the envelope (protocol), and associated data.

2.6.1 PEPS – PEPS scenario

2.6.1.1 Preliminary Sequence Diagram

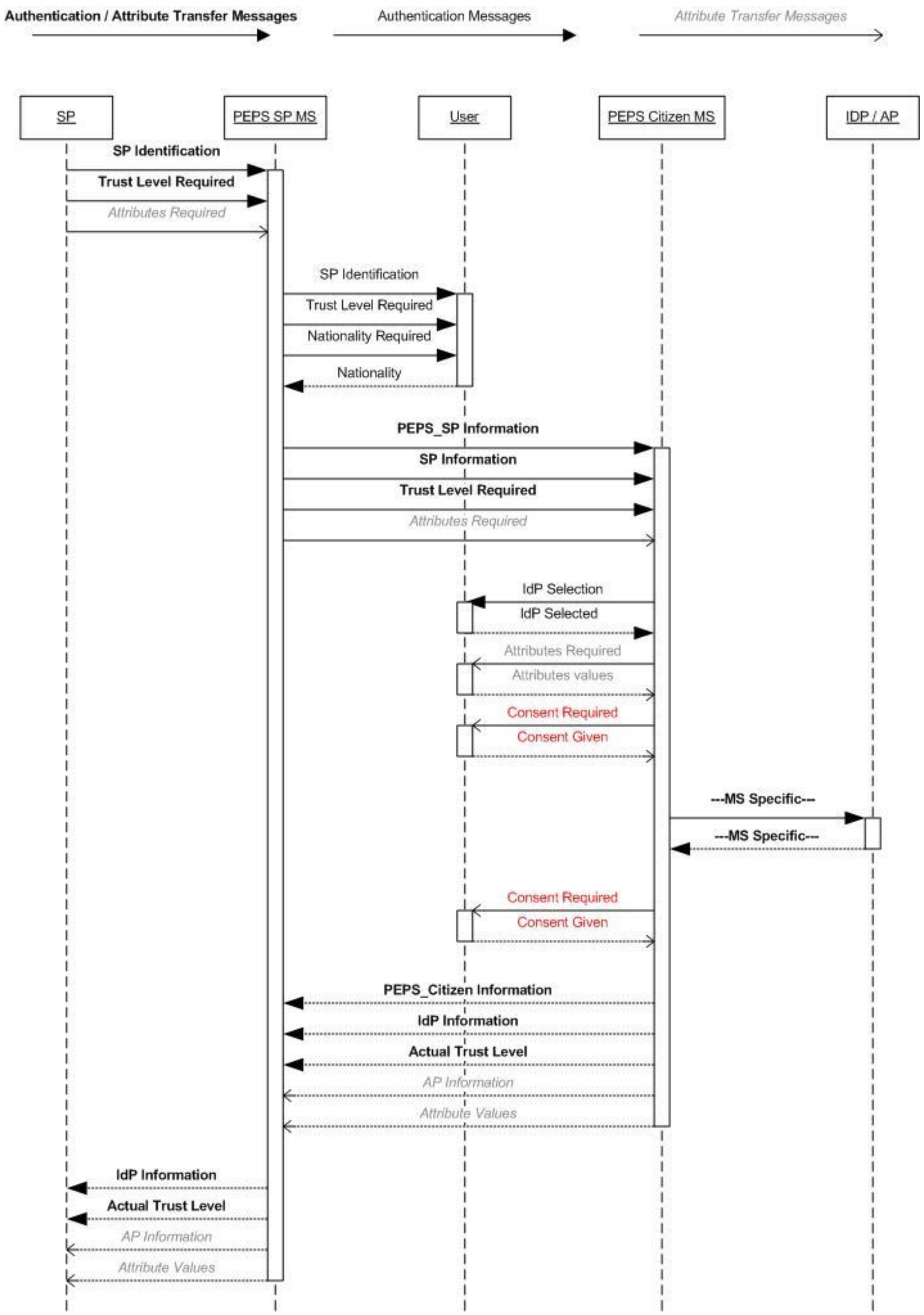


Figure 2: Preliminary Sequence Diagram for PEPS-PEPS scenario

2.6.1.2 Messages

Actors	Data included in Message
SP → PEPS SP MS	- Relying Party Identification (SP Identification) - Trust Level Required - Attributes Required (mandatory/optional) In case of Attribute Transfer: eIDNumber
PEPS SP MS ←→ User	- Relying Party Identification (SP Identification) - Trust Level Required - Country Selection
PEPS SP MS → PEPS Citizen MS	- Relying Party Information (PEPS SP) - Original Relying Party (SP) - Trust Level Required - Attributes Required (mandatory/optional) In case if Attribute Transfer: eIDNumber
PEPS Citizen MS ←→ User	- IdP Selection - Credentials in case of certificate - Data-type user consent required (AT) - Consent given (AT)
PEPS Citizen MS ←→ IdP / AP	Member State specific
PEPS Citizen MS → PEPS SP MS	- Relying Party Information (PEPS Citizen) - IdP Information - Attribute Provider Information (AT) - Actual Trust Level - Attribute Values (mandatory/optional) (AT) In case if Attribute Transfer: eIDNumber
PEPS SP MS → SP	- IdP Information - Attribute Provider Information (AT) - Actual Trust Level - Attribute Values (mandatory/optional) (AT) In case if Attribute Transfer: eIDNumber

Table 24: Contents of messages in PEP-PEPS scenario

2.6.2 PEPS/MW Scenario

2.6.2.1 Sequence Diagram

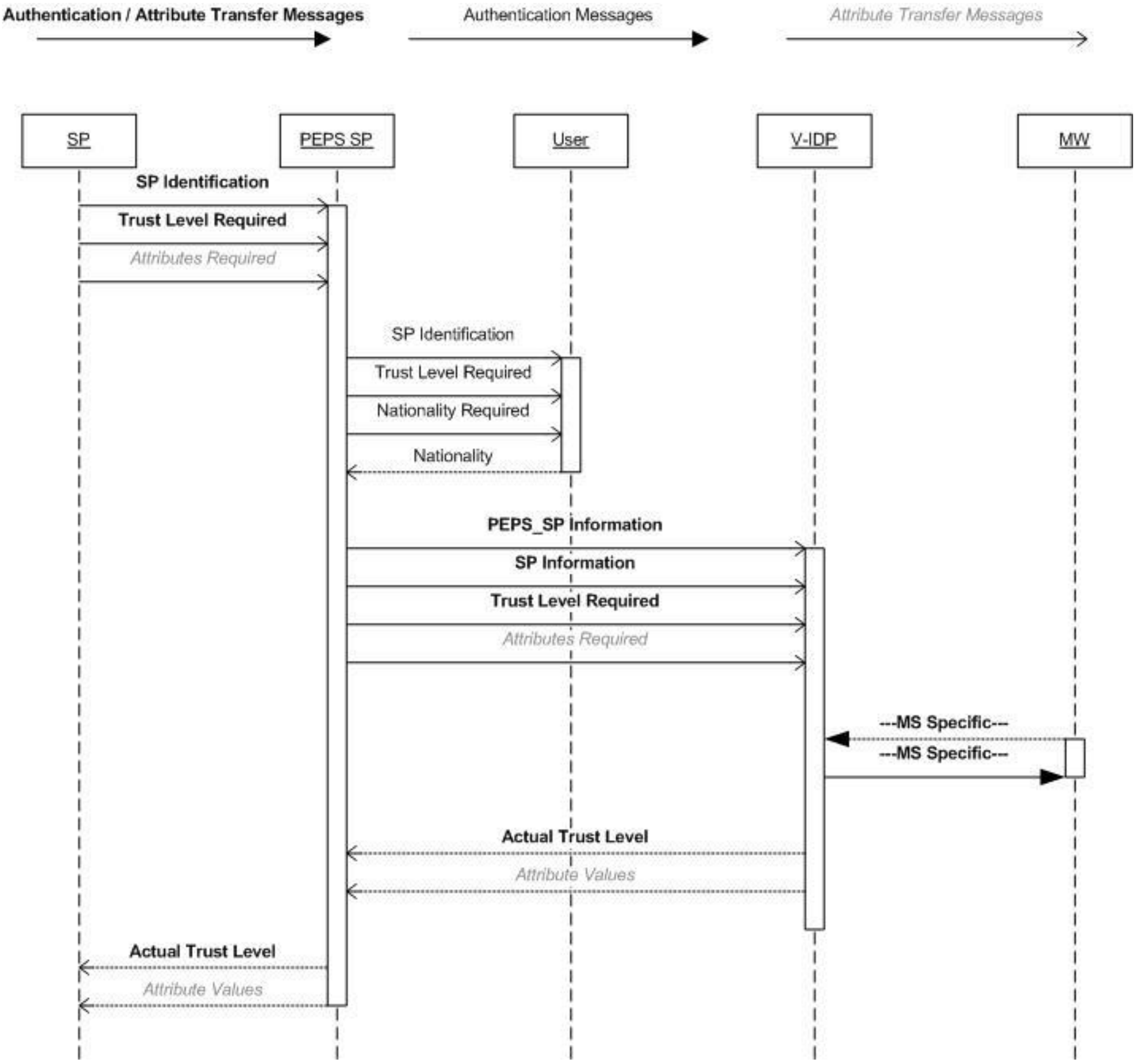


Figure 3: Preliminary Sequence Diagram for PEPS-MW scenario

2.6.2.2 Messages

Actors	Data included in Message
SP → PEPS SP	• Relying Party Identification (SP Identification) • Trust Level Required • Attributes Required (mandatory/optional)
PEPS SP → User	• Relying Party Identification (SP Identification) • Trust Level Required • Country Selection
PEPS → V-IDP	• Relying Party Information (PEPS SP) • Original Relying Party (SP) • Trust Level Required • Attributes Required (mandatory/optional)

Actors	Data included in Message
V-IDP ←→ MW	Member State specific
V-IDP → PEPS SP	• Actual Trust Level • Attribute Values (mandatory/optional) (AT)
PEPS SP MS → SP	• Actual Trust Level • Attribute Values (mandatory/optional) (AT)

Table 25: Contents of messages in PEP-MW scenario

2.6.3 MW/PEPS scenario

2.6.3.1 Sequence Diagram

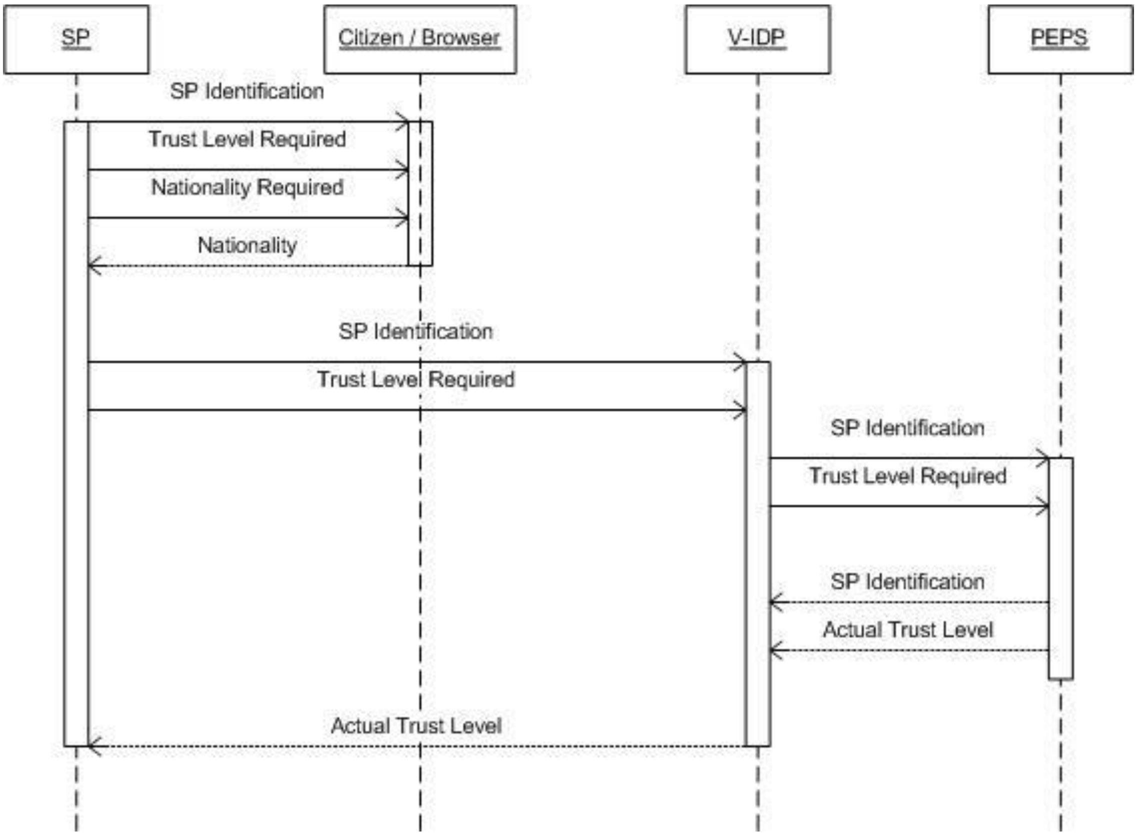


Figure 4: Preliminary Sequence Diagram for MW-PEPS scenario

2.6.3.2 Messages

Actors	Data included in Message
SP ←→ User	• Relying Party Identification (SP Identification) • Trust Level Required • Country Selection

SP → V-IDP	• Actual Trust Level • Relying Party Identification (SP Identification)
V-IDP ←→ PEPS	• Actual Trust Level • SP Identification
V-IDP → SP	• Actual Trust Level

Table 26: Contents of messages in MW-PEPS scenario

2.6.4 MW/MW scenario

In the MW/MW scenario the messages between the MW, the SPware, and the SP are closely related to the deployed eID tokens and technical implementation and existing standards (e.g. European citizen card in Germany or SAML in Austria). Thus a stronger technology relation to Member State implementations exists then aimed for in this deliverable that aims on the functional aspects (WHAT) rather than the technologies (HOW). We therefore suppressed messages and sequence diagrams in this document.

The common specifications in the MW-MW scenario are to be described with the detailed specifications when technology choices are to be made.

2.7 Auditing

2.7.1 Introduction

According to NIST (US National Institute for Standards and Technology), auditing is a review and analysis of management, operational and technical controls. The audit trails are the implementation of these controls and serve the purpose of maintaining a record of system activity both by systems and application processes and by user activity of systems and applications, providing also a means to help accomplish several security-related objectives, including:

1. individual accountability;
2. reconstruction of events;
3. intrusion detection;
4. problem analysis and error detection.

The audit trails or (more frequently) ‘logs’ are a central part of security not only in computer system security but also in analyzing financial and other non-technical systems. As part of this process, it is often necessary to reconcile logs from different sources.

But what is a ‘log’ anyway? According to the definition in [5], a *log* is a record of the events occurring within an organization’s systems and networks. Logs are composed of log entries; each entry contains information related to a specific event that has occurred within a system or network. These computer security logs are generated by many sources, including:

- security software, such as anti-virus software, firewalls, and intrusion detection and prevention systems;
- operating systems on servers, workstations, and networking equipment;
- applications.

The number, volume, and variety of different systems operating in the STORK’s distributed architecture created the need for computer security log management, which also involves protecting the confidentiality, integrity, and availability of logs.

Due to requirements of confidentiality, some of the standard log-tools can't be used, so in this chapter we'll describe the application log.

The STORK architecture can be seen as a producer-consumer system composed of set of components (e.g. SPs, IDPs, PEPS, V-IDP), each of which generates, processes and sends out data (as part of a transaction) that is going to be consumed by one or more components in the STORK architecture (see Figure 5).

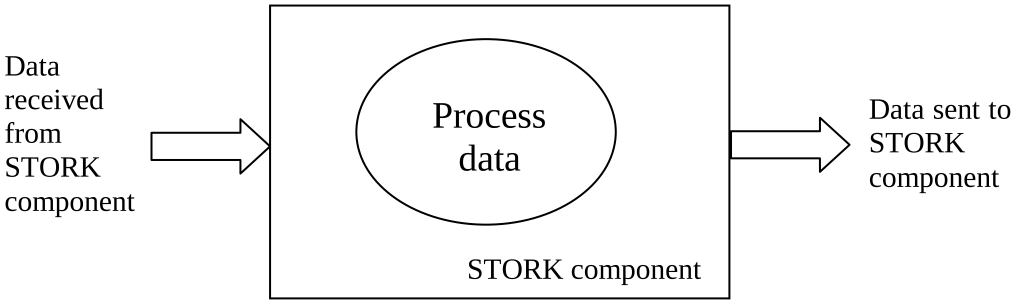

Data received from STORK component
Process data
STORK component
Data sent to STORK component

Figure 5: High-level view of a STORK component (e.g. SP, IDP, PEPS)

For auditing purpose, we need to define a set of requirements and functionalities to be adopted by all STORK components in order to improve interoperability, conflict resolution and problem solving for transactions or workflows using STORK.

The main objectives of the current chapter are to:

- identify the information to log and the operation outcome to log;
- define access policy to the audit log;
- identify the steps to be done to complete an auditing process in the STORK architecture;
- define the audit log management policy (log generation, storage, analysis);
- discuss privacy and audit security issues in the STORK architecture.

Normally we should expect audit analysis requests to be end-to-start, i.e. a user detects strange movements with his account and contacts his service provider. This entity analyses his own log for authentication events for this user, and, if he detects that the event came from STORK, he'll request to analyse the audit trail.

Due to confidentiality restrictions, we can't use the standard logging from e.g. Web servers, so in this chapter we describe the application logging we need.

2.7.2 Audit Definitions

Auditing in STORK is guaranteed by the implementation of audit trails in the form of monitoring and logging components for all the individual sub-systems that are part of a STORK component and their environment.

Among other things, knowing what information to log and having the correct definition of systems/operations to monitor, are vital components in the definition of an Audit Policy that ensures that the audit logs are examined regularly and frequently, and that appropriate action is taken, over any irregularities discovered.

2.7.2.1 Audit Trails in STORK

The audit trails can be application logs or other equivalent means of evidence, as long as they contain sufficient information to trace back the complete details of an operation (in case of success or failure) and can be searched or queried in an automated way.

2.7.2.2 Distributed Audit Services

In a stand-alone system we can isolate all the security relevant activity in individual components, thus allowing us to maintain a time-ordered list of all actions and events (e.g. audit trail) that happened in the system and enabling us to audit the system by simply following the information trail in a single place, obeying to a defined notation and in a specific timeline.

In distributed systems (such as STORK) the security relevant activity, most probably, will span several components enabling the use of different resources all throughout the system instead of just in a server/workstation.

The differences above mean that we need to plan carefully when defining which audit data to collect and how these should be managed, due to the lowered feasibility of interpreting what gets collected (because of the different audit philosophies, different rules and/or regulations, the possibility of having differing naming policies and values, misuse of authority by authorised users, etc).

With this in mind the following definition of Audit Logs functional components has to comply with two very important requirements:

1. The possibility of reconstructing an entire transaction by linking all related request/response identifiers throughout the complete path from Service Provider to Identity Provider and back; starting from the end.
2. The association between an incoming request and (all related) outgoing request(s), as well as the corresponding replies

2.7.2.3 Privacy Issues

STORK, as any eIDM infrastructure mainly deals with the exchange of citizen/user personal information for means of authentication in different services. This means that in the day to day operation of such a system, any unwanted disclosure of personal information (and subsequent misuse) would not only constitute a violation of the citizen's privacy rights, but could also put at risk his assets such as physical property, financial assets, reputation, among others.

Another thing to consider is that threats to privacy and loss of anonymity may strongly demotivate citizens from using such a system. This makes adequate privacy protection an important goal to have in mind when defining an audit policy for our system.

2.7.2.4 Audit Logs Functional Components

As is defined in [8], security functional components describe the desired security behaviour expected of a product, expressing security requirements intended to counter threats in the assumed operating environment of the product and/or cover any identified organisational security policies and assumptions.

Security functional components for a product can cover a span of different functional classes, families and components. In our case, we are only interested in the definition of a subset of these functional components, specifically the ones that are directly or indirectly related with the theme of audit in a STORK infrastructure.

There are already available, well defined and commonly recognized frameworks ([8],[10]) that define requirements to implement an audit function in large scale systems designed to work in a distributed environment in the STORK image. In our particular case, we will base our requirements in the work already developed under the Common Criteria Recognition Agreement[8].

The following functional components based on these criteria are included in this definition. Some definitions have dependencies with other functional requirements. When this happens the dependent functional requirements are identified by their unique identifier in the "Dependencies" section of each component.

1. **FAU_GEN.1 Audit data generation**
   Dependencies: FPT_STM.1 Reliable time stamps

   FAU_GEN.1.1 The STORK component shall be able to generate an audit record of the following auditable events:
      a. Start-up and shutdown of the audit functions;
      b. All incoming and outgoing messages

   Note that several other events (log in tries, change of system parameters, etc.) should also be logged, but this type of logging is part of system logging, so is out of the scope of this document. Here we limit the description to application logging, as described in the last two paragraphs of 2.7.

   FAU_GEN.1.2 The STORK component shall record within each audit record at least the following information:
      a. Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
      b. For each audit event type, based on the auditable event definitions of the functional components, origin and destination of message, request identifier (according to origin), parent request id, request or reply hash.

2. **FAU_SAR.1 Audit review**
   This component will provide authorised users [with] the capability to obtain and interpret the [audit] information. In case of human users this information needs to be in a human understandable presentation. In case of external IT entities the information needs to be unambiguously represented in an electronic fashion.
   As we expect that the frequency of this process will be very little, standard tools will be used; like standard text editors if the log file is in text format.

3. **FAU_SAR.2 Restricted audit review**
   FAU_SAR.2.1 The STORK component shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

4. **FAU_STG.1 Protected audit trail storage**
   FAU_STG.1.1 The STORK component shall protect the stored audit records from unauthorized deletion.
   FAU_STG.1.2 The STORK component shall be able to prevent or detect modifications to the audit records.
   FAU_STG.1.3 The STORK component shall ensure that at least eighteen[2] months stored audit records will be maintained.
   FAU_STG.4.1 The STORK component shall overwrite oldest stored audit records if the audit trail is full. *This requirement may be omitted in the actual definition of STORK, as log is kept for a period larger then the duration of the project.*

[2] This minimum time must be configurable for different MS, but the lowest value is 18.

5. **FPT_STM.1 Reliable time stamps**
   FPT_STM.1.1 The STORK component shall be able to provide reliable time stamps for its own use.

6. **FMT_MTD.1 Management of STORK component data**
   FMT_MTD.1.1 The STORK component shall restrict the ability to query the audit log information to members of the administration group.

7. **FIA_UID.1 Timing of identification**
   FIA_UID.1.1 The STORK component shall require each user to be successfully identified as an administrator before allowing any other STORK component-mediated actions on behalf of that user.

2.7.3 Log Management and Security Issues

2.7.3.1 Logging requirements and goals in STORK

Starting from the requirements identified in [6], we identified several logging requirements specific to the STORK architecture. In practice, any STORK component should record and retain audit-logging information sufficient to answer the following questions:

1. Who performed the logging (the STORK component or an external module/party)?
2. What activity was performed? (e.g. Input, Process, Output)
3. Who or what performed the activity, including where or on what system the activity was performed from (subject)?
4. What the activity was performed on (object)?
5. When was the activity performed?
6. What tool(s) was used to perform the activity?
7. What was the status (such as success vs. failure), outcome, or result of the activity?

2.7.3.2 Mandatory requirements and suggested recommendations

In practice, in the auditing process, what is needed is the ability to derive/reconstruct the workflow based on the state/log information locally stored by each STORK component (such as PEPS, V-IDP, MW) and the communicating entities involved in STORK use cases, such as IDP or SP.

Since the workflow in STORK is coordinated by the exchange of messages, the only state information which can be interpreted by all parties is the sequence of exchanged messages. In addition, each STORK component should log state information about the operations (i.e. processing) it performed locally.

All the different components of STORK should be able to perform monitoring and logging of all their systems. In particular PEPSes, V-IDPs, IDPs and SPs must produce audit trails on their systems for all relevant operations, configuration changes, access attempts, exceptions and key management functions (if applicable) critical to the security of the reliance being placed on processes and operations.
The audit trails must be stored in a long-term support and a tamper evident format, such that illicit addition, modification or deletion of any audit trail can be detected.
The dates and times registered in the audit trails must be obtained from a trusted time source to ensure accuracy and a reasonable clock synchronization of the different parties

2.7.3.3 Log management infrastructure

2.7.3.3.1 Log generation

In the STORK architecture and for auditing purposes, a message handled in a STORK workflow is represented by:

- its sender S (divided in SName and SIP) for incoming messages;
- its recipient R (divided in RName and RIP) for outgoing messages;
- the message identifier (request id);
- the date/time of the message.

Also, additional information must be defined, corresponding to the 7 questions identified in the above section (Logging requirements and goals in STORK):

- LogEntryCounter, to identify the log sequential number.
- OpType (e.g. Auth(entication), Attribute Transfer(AttT), etc), to distinguish between the types of operations encountered on the STORK component PEPS-C.
- msg_hash, identifies the complete object handled in the STORK workflow. The SHA-256 hash is over the functional data; this excludes e.g. the timestamp of sending.
- originator and originating msgid, identifies the message that caused this message msg to be generated, absent in case of incoming requests.
- datetime, is the date and time when the operation was performed on message msg.

Consequently, one proposed format of the log entry corresponding to the sequence of exchanged messages in STORK is:

LogEntryCounter#datetime#OpType#[S_{Name}#S_{IP}#][R_{Name}#R_{IP}#]request_id#

msg_hash#[originator$_{Name}$#orig_msg_id#]

meaning message msg was sent by S to R at datetime, and where # character is used as delimiter, and the optional components are marked with [and]. S or R is present depending on whether message is incoming or outgoing. Originator and originating message id are absent in case of incoming requests.

Important notes: The security applying to the log entries individually or to the log file (as a whole) will be analysed in a separate section. Furthermore, in future phases of this project new attributes may be added.

For example, for the STORK workflow:

Citizen) <--> Service Provider <--> PEPS SP MS <--> PEPS Citizen MS <--> IDP-C

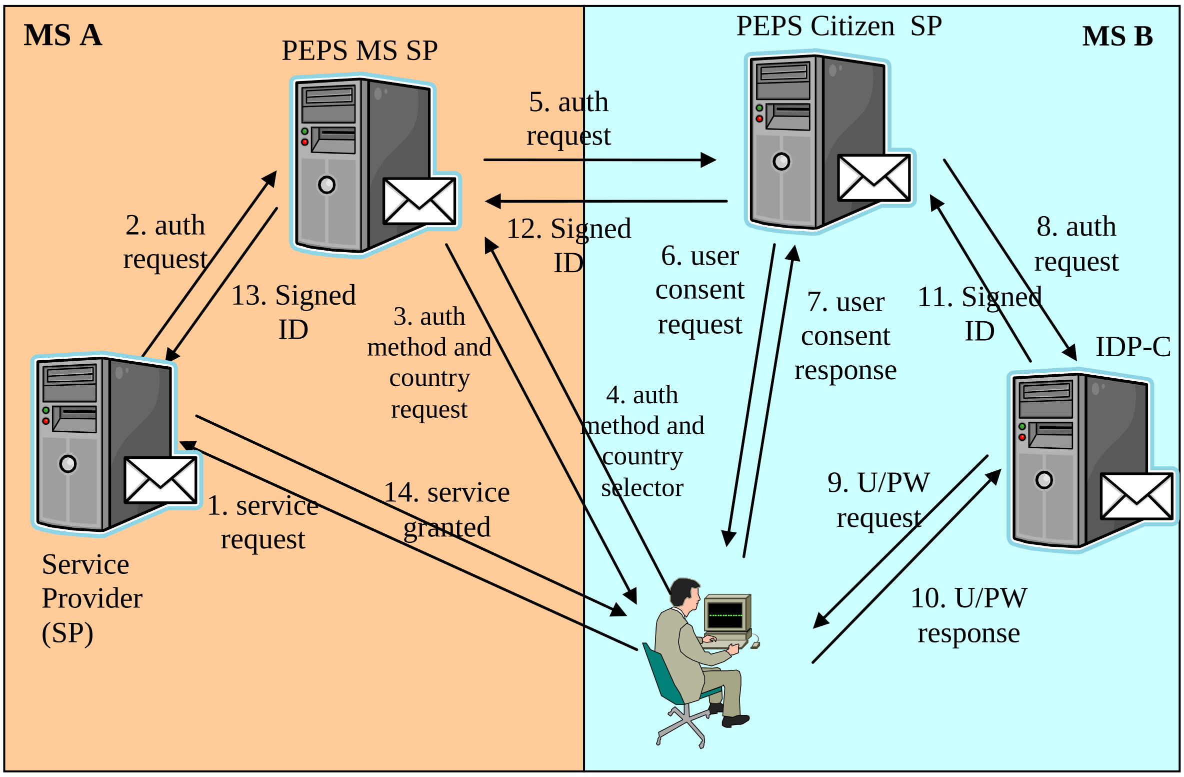


Figure 6: STORK workflow for eID transfer involving several STORK components.

used for e-ID transfer, we should see the following log entries on each STORK component (involved in the STORK workflow) corresponding to the simplified sequence of exchanged messages illustrated in Figure 6. Please note in bold letters the originator Name and original msg_id.

-on Service Provider(SP):

```
1#13Feb200918:22:59#Auth#POLITO#201.202.203.204#IT-PEPS#193.194.195.196#12345# 654ACEFD#
……
25#13Feb200918:23:21#Auth#IT-PEPS#193.194.195.196#POLITO#201.202.203.204#34567#ABE5737D#POLITO#12345
……
```

- on IT-PEPS:

```
23#13Feb200918:23:00#Auth#POLITO#201.202.203.204#IT-PEPS#193.194.195.196#12345# 654ACEFD#
24#13Feb200918:23:00#Auth#IT-PEPS#193.194.195.196#User#197.198.199.200#34500#EE4578BA#POLITO#12345#
….
29#13Feb200918:23:05#Auth#User#197.198.199.200#IT-PEPS#193.194.195.196#34500#7554321#POLITO#12345#
30#13Feb200918:23:05#Auth#IT-PEPS#193.194.195.196#PT-PEPS#123.45.67.89#34510#DAC547FE#POLITO#12345#
….
40#13Feb200918:23:21#Auth#PT-PEPS#123.45.67.89#IT-PEPS#193.194.195.196#45678#CBA98765#POLITO#12345#
41#13Feb200918:23:21#Auth#IT-PEPS#193.194.195.196#POLITO#201.202.203.204# CBA98765#POLITO#12345#
….
```

2.7.3.3.2 Log storage

Considering the format proposed in the above section, it will be assumed that each STORK component stores locally the information in a dedicated file. The techniques to be used for data retrieval from the log will be analysed in a separate section.

2.7.3.3.3 Log transmission

Not considered (for the moment).

2.7.3.3.4 Log analysis

As log analysis is expected to be a very little frequent process, this is executed manually, and standard text editors can be used.

2.7.3.3.5 Log disposal

Taking into account the different laws, policies and regulations regarding the conservation of log information in eID systems already implemented by the MS, the minimum time interval defined for the conservation of the log information must be (at least) equal to the smallest time frame (months), currently being used in any of the MS.

After a maximum time (months) defined by the MS has passed, the logs must be automatically deleted.

In the scope of the STORK pilot, the generated logs must comply with *FAU_STG.1 Protected audit trail storage*.

2.7.4 Log Security Issues

2.7.4.1 Malicious Actions

In analysing the security requirements and the appropriate solutions for logging data in STORK, we start from considering several malicious actions that could be performed either by a STORK component itself or by a communicating party involved in a STORK transaction (eID transfer, attribute transfer, etc.).

In the following we consider parties to log malicious actions under the prerequisite that the STORK workflow is consistent, that is, it does not contain deadlocks.

The following cases involving malicious actions could be encountered for a single message exchange:

1. A single party logs malicious actions (operations), while the other one logs truthfully.
   In this case, it can be detected that two different messages have been logged, although it can be differentiated which party cheated. It could either be the sender, who logged a message he hasn’t sent to the recipient, who is logging truthfully, or the sender is logging truthfully while the recipient logs a malicious action.

2. Both parties log malicious actions.
   In case the parties are logging different messages, again the difference can be detected although it cannot be decided who or whether at least one acted truthfully. In case the two parties agree on which malicious operations to log (the two parties conspire), then the malicious logging remain undetected in a first step.

2.7.4.2 Integrity for Decentralized Log Model in STORK.

Each STORK component has to support the integrity of the log data. In other words, any modification of the logged data, as well as the insertion, deletion and replacement of log data in the log store must be detected.
To achieve this we can use one of several solutions available (among others not covered here):

1. Write Once Devices
   Without using cryptography, this is the best form of protecting log data. The entire log that is written cannot be changed latter on and there are also obvious improvements to the physical storage management process. However using this method there is also the possibility of having bottlenecks during the log writing process, there are the possibility of hardware malfunction (mitigated by using redundancy) and also, the possibility of the log data being tampered with before being written to the device without later detection of this compromise.

2. Electronic Signatures
   The use of cryptography is an obvious improvement to the integrity guarantee of the log data and also makes good use of legislation widely available in most countries. Using electronic signatures to sign the log we can always detect if a certain line has been tampered with, and there are also signature schemes that associate the previous log line with the next one, thus allowing the detection of deletions/replacements of certain log lines. Also, since the process is electronic we always have the possibility of consulting/managing the log data easily and automatically.
   The downside of using electronic signatures is the very real possibility of self provoked Denial of Service attacks. The log signature is a time consuming operation that has to be done sequentially, so log intensive applications will take a lot of time in the process of log signing/writing.
   To prevent against these self provoked DoS attacks we can define a signature policy to the log. As such we can define a set period (be it a time period, line interval, etc) to sign the data. This way, instead of signing all log lines, we just sign the log data in specific occasions thus reducing the number of signatures made and still achieving not only reasonable guarantees against tampering but also (combined with other systems in the infrastructure) reasonable tampering detection capabilities.

3. Remote Storage
   Remote storage is a mechanism in which the log file is automatically stored on a remote machine, to which the administrator of the PEPS doesn't have access rights. Such a remote machine could be a colleague PEPS or any other server in the national network infrastructure.

As described above, each solution has its advantages and disadvantages and none of then is absolutely perfect. Each STORK component should choose its implementation according to its internal policies, but always respecting what is defined in the Audit Logs Functional Components and having in mind concerns about performance. The methods 2. and 3. require the STORK software to include these facilities.

3 Functionalities of business processes

3.1 Introduction

This chapter describes the functionalities of the STORK system. Thus this document acts as input for the subsequent design. To describe the functionality a use case based approach is used, which is explained in more detail in the following section.

3.1.1 Methodology

There are many different approaches to describe functional requirements of a system. In STORK we follow the well known use case based approach. Thereby we start with a use case description (which is a textual description) of the main "business processes". As primary input we use the 1st draft of process flows of WP4. These use case descriptions do not consider special architectural constraints, but define the functionality of STORK from an abstract point of view.

Afterwards each use case is analysed with respect to the given reference architecture. This means that we take the pan-European proxy service (PEPS) and middleware (MW) approaches and their main components into consideration. Thereby we describe each "scenario" independently to avoid functional decomposition. To describe these scenarios we use the Unified Modelling Language (UML) that is state of the art and known by most IT engineers. However, we use UML not in a strict way, but use it as a tool to describe the functionalities. Therefore some of the diagrams may not even be valid UML but provide nevertheless a good understanding of the system.

This chapter provides an exhaustive description of the system's functionality. Thereby we focus on the question WHAT needs to be done (not HOW). This means the current version is completely technology independent and thus may appear to be partially very abstract. During the design phase these use case descriptions must be refined (and possible adapted) to fit the chosen technology.

3.2 High level business processes

The functionality of STORK is basically defined by the following three processes.

1. **Authentication** is the process that allows a user to access privileged data. Usually this process ends with a fully identified user, which means that his eID is transferred to the service provider (SP), and this SP recognises this user as a known customer, student, partner, or whatever relationship this person may have with the SP.

   Nevertheless, other service providers or applications may exist that have fewer requirements; they allow access to privileged data without fully identifying the user. In the pilots of the STORK project we have examples of the access to university library, which is reserved to students, or the access to several rooms of Saferchat, which is restricted to people of a certain range of age, or of a certain gender. In both these cases we should call this process to allow users to access privileged data authentication, even though they're not fully identified in the SP. Other examples exist of authentication processes that don't end up with a fully identified user against SP. The user is identified in their country but only the data required to access is sent to the SP.

2. **Attribute Transfer** is the process that allows a service provider to access additional attributes from an attribute provider (AP), other than those required for the basic authentication. This process is initiated with a fully identified user, but his eID is not recognised as a known customer at the SP.

The attributes that can be requested are any combination of data-types recognised by STORK, and the STORK platform will do whatever it can, to retrieve the requested data, and, if the user allows so, pass the data to the service-provider that requests this info. This user-consent will be implemented in accordance with the legal data protection requirements of each member state (MS).

The implementation of authentication and attribute transfer as two consecutive processes might require the user to identify himself twice to his IDP, give twice his consent to the transfer of data, etc., so such a construction should be considered as improvable. But several applications exist, which contain such an implementation, so their support within STORK is unavoidable.

3. **Certificate validation** handles the scenario, where the service provider (SP) needs to verify a user-created digital signature. Whereas the signature validation is out of scope, STORK offers the functionality of validating certificates for signatures.

Sometimes a **fourth process** is mentioned, the **registration at a service provider**. This fourth process should be executed when a new user wants to register at a service and clicks the corresponding option. Although, as a business process, this is a separate process that exists and is recognised as such, and even probably in technological sense it may be necessary to construct it. In a functional sense it can be realized with the existing authentication process. So, considering this, and having agreed on this premise with all the member states that participate in the WP5, the leaders of this WP have suggested that we should not include a description of this fourth business process in this functional design.

The rest of the functionalities section is structured as follows. The following section gives an implementation independent description of the business processes and other overall functional requirements. This is done by means of use cases. Afterwards a refined analysis of these use cases is done to work out the functional requirements for each scenario in more detail.

3.3 System overview

3.3.1 Actors

Name	Description
Citizen (CIT)	A citizen is any person of a country of one of the member states, who owns an electronic identifier. The only citizens taken into account are the people that want to use eServices offered by service providers in foreign member states, which allow the use of STORK's eID interoperability platform.
Service Provider (SP)	A service provider is an institution, public or private, that offers people the facility to execute business transactions electronically.
Identity Provider (IDP)	An entity which provides electronic credentials (eID) and optionally some attributes of the citizens.
Attribute Provider (AP)	An organisation which provides certain attributes of citizens.

Table 27: Actors

3.3.2 Use Cases Overview

The following diagram shows the system level use-cases offered by the STORK system.

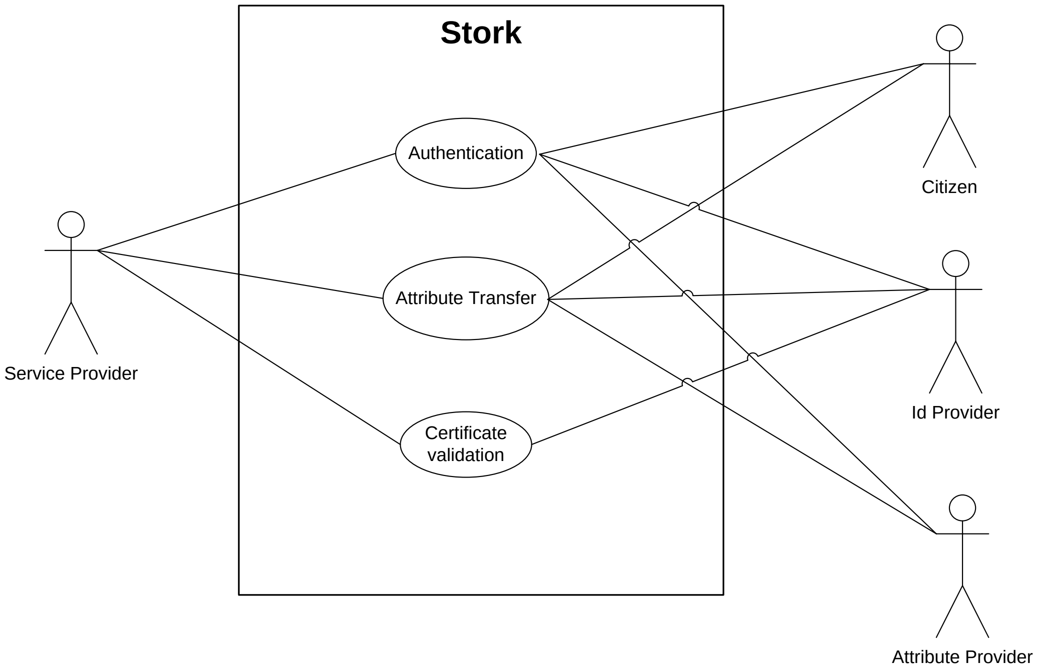


Figure 7: System Use Cases

3.4 Use Case Authentication

Authentication is the process of proving user's credentials (issued by an EU member state) to a service provider that the user is trying to access. A more exact description is given in paragraph 3.2.

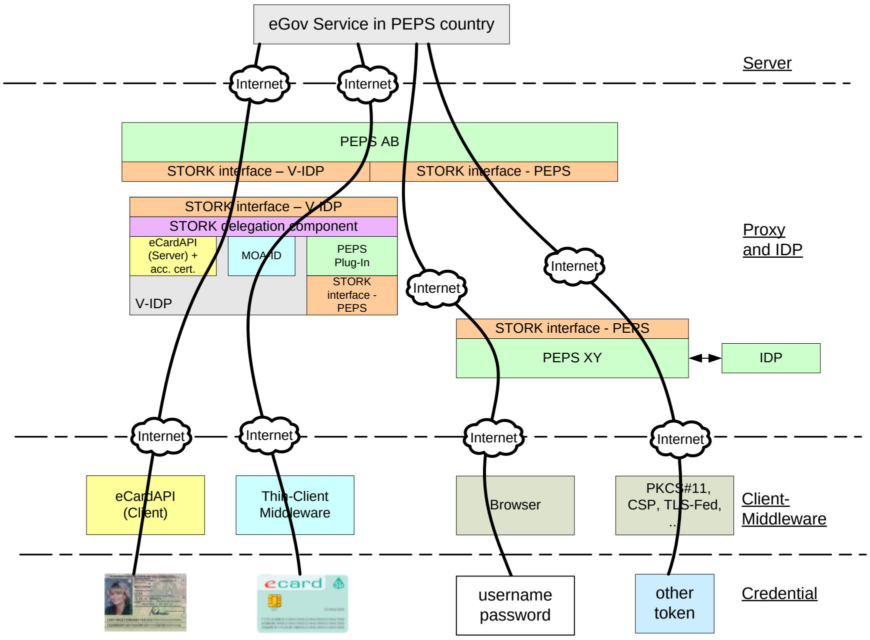


Figure 8: STORK Environment if Service Provider is of a PEPS Member State

The figure above depicts the playing field for this use case together with some basic business rules.

If Service provider is in a **PEPS member state**, he always contacts his national PEPS only. Any cross-border communication is channelled either through a local-PEPS-to-remote-PEPS connection or through the national virtual IDP (V-IDP) typically hosted at the same site as the PEPS, both of them act as a proxy for foreign eID services. In that case, the V-IDP's role would be limited to handling traffic from user clients.

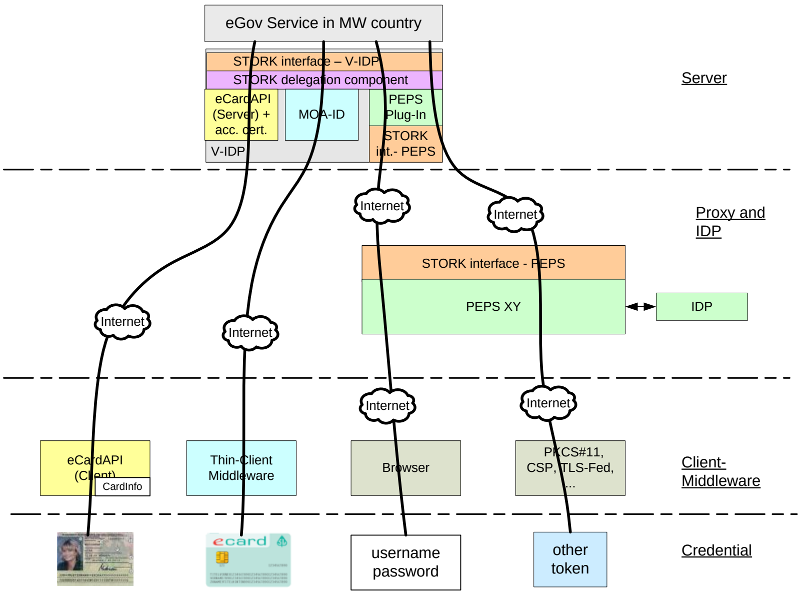


Figure 9: STORK Environment if Service Provider is of a MW Member State

If SP is of a **MW member state**, SPs will communicate directly with eID services of MW countries, or through V-IDPs, which on their turn access the eID services of PEPS countries, depending on where the user's eID is from.

3.4.1 General description

The standard case envisioned by the members of the STORK functional work group is as follows. An EU resident wishes to access an eService (including, but not necessarily limited to eGovernment services) offered by or in an EU member state. To prove the resident's entitlement to use this service, they present their national eID (or equivalent) to STORK, which will transfer his data to the eService.

3.4.1.1 Brief Description

To be able to access an eService, the user presents his eID to the same client PC[3] he is currently using. The STORK interoperability layer, accessible from the website of the service provider, provides the validation of the eID presented and, after collecting explicit user consent, transfers requested supporting eID data to the service provider.

3.4.1.2 Preconditions

ID	Description
AU-PRE-1	Citizen uses client PC[3] with internet connection (client MW installation may be required; smart card reader may be required)
AU-PRE-2	Service provider has embedded STORK functionality into his website for eService authentication (server MW installation may be required; contract with national PEPS may be required; nation-specific access certificate may be required)
AU-PRE-3	Citizen holds valid eID issued by EU member state that is also a member of STORK

Table 28: Preconditions for UC-AU

3.4.1.3 Postconditions

ID	Description
AU-POS-1	Service provider has received the data that he considers prerequisite to providing the requested service to the citizen.

Table 29: Postconditions for UC-AU

3.4.1.4 Main flow of events

ID	Description
AU-MFE-1	Citizen selects an eService that requires authentication. Some code can be embedded into the service provider's webpage.
AU-MFE-2	Service provider prompts the citizen to select the country of his identifier. Depending on

[3] Client PC is meant as a notion for the most common citizen's Internet browsing component. This does not rule out other components such as PDAs, smart phones, etc.

ID	Description
	the type of service, this may not be the full list of STORK members.
AU-MFE-3	Citizen selects his member country
AU-MFE-4	Service provider requests authentication data from STORK (depending on context, the request may go to a PEPS or to a V-IDP)
AU-MFE-5	STORK requests the citizen to perform MS specific authentication procedure. STORK can obtain also attributes from national Attribute Providers or verify attributes introduced by the user, only for some special attributes.
AU-MFE-6	STORK requests citizen consent to deliver the requested data to the service provider
AU-MFE-7	Citizen selects the data elements to be released from STORK to the service provider (from a list of those elements originally requested by the service provider)
AU-MFE-8	STORK delivers the consented data to the service provider

Table 30: Main flow of events for UC-AU

3.4.1.5 Alternative flows

ID	Condition	Description
AU-ALF-4-1	The Service Provider is not allowed to send any request	Alternative sequence to AU-MFE-4: If the service provider is not whitelisted by a STORK component as legitimate requester, STORK will deny the service provider's request for data.
AU-ALF-4-2	The Service Provider is not allowed to ask for some attributes	The SP is not allowed to request (these) attributes.
AU-ALF-5-1	The citizen does not present a valid eID	Alternative sequence to AU-MFE-5: If the citizen does not present a valid eID or it can not be successfully verified, STORK will deny the service provider's request for data.
AU-ALF-6-1	The Service Provider asks for mandatory attributes that are not consented by the user.	Alternative sequence to AU-MFE-6: If the citizen does not consent to release at least a minimal set of data to the service provider, STORK will deny the service provider's request for data.
AU-ALF-8-1	The Service Provider asks for mandatory attributes that are not found or not allowed in the MS.	Attribute not found or not allowed. Authentication Request is rejected by STORK.

Table 31: Alternative flows for UC-AU

3.4.1.6 Special Requirements

ID	Description

UC-AU-SR-1	The attribute transfer request may also contain derived attributes.
UC-AU-SR-2	The Citizen must give her/his consent to the attributes to be transferred. Depending on member state specific requirements this consent might be given over the data-types or over the values. The user can deny the consent to send the attributes to his MS. In this case the contact with the Attribute Providers is avoided and the request rejected.
UC-AU-SR-3	The Citizen must have the possibility to disable optional attributes. In this case the disabled attributes will not be transferred to STORK and further to the Service Provider.
UC-AU-SR-4	The attributes must be protected from unauthorized modification and disclosure during the transfer between IDP/AP and SP.

Table 32: Special Requirements for Authentication

3.4.1.7 Other Requirements

ID	Description
UC-AU-OR-1	This process is supposed to be executed through the Internet, using standard browsers (Microsoft Internet Explorer, Mozilla Firefox or similar)

Table 33: Other Requirements for Authentication

3.4.2 Authentication PEPS-PEPS: UC-AU-PP

In this scenario a citizen of a PEPS member state wants to consume a service within another PEPS member state.

3.4.2.1 Reference Architecture

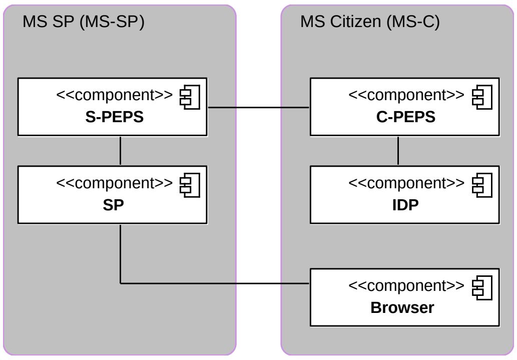


Figure 10: Reference Architecture for UC-AU-PP

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS located in the SP's country.
Browser	The citizen uses the browser to consume a service of the SP.
SP	The service provider requiring user authentication.
C-PEPS	The PEPS located in the citizen's country.

IDP	The identity provider (IDP) is an important part of the authentication process.

Table 34: Components for UC-AU-PP

3.4.2.2 Actors

ID	Abbreviation	Description
AU-PP-AC-1	CIT	Citizen using the browser to interact with the system.
AU-PP-AC-2	SP	See according component description.
AU-PP-AC-3	S-PEPS	See according component description.
AU-PP-AC-4	C-PEPS	See according component description.
AU-PP-AC-5	IDP	See according component description.

Table 35: Actors for UC-AU-PP

3.4.2.3 Activity Diagram

To complete the final process flow it is necessary to add some redirections, where the citizen redirects the session from one actor to another. As redirections are part of the technical implementation the possible redirections are not drawn in the picture below. The final process with all redirections will be defined and drawn later in the design phase.

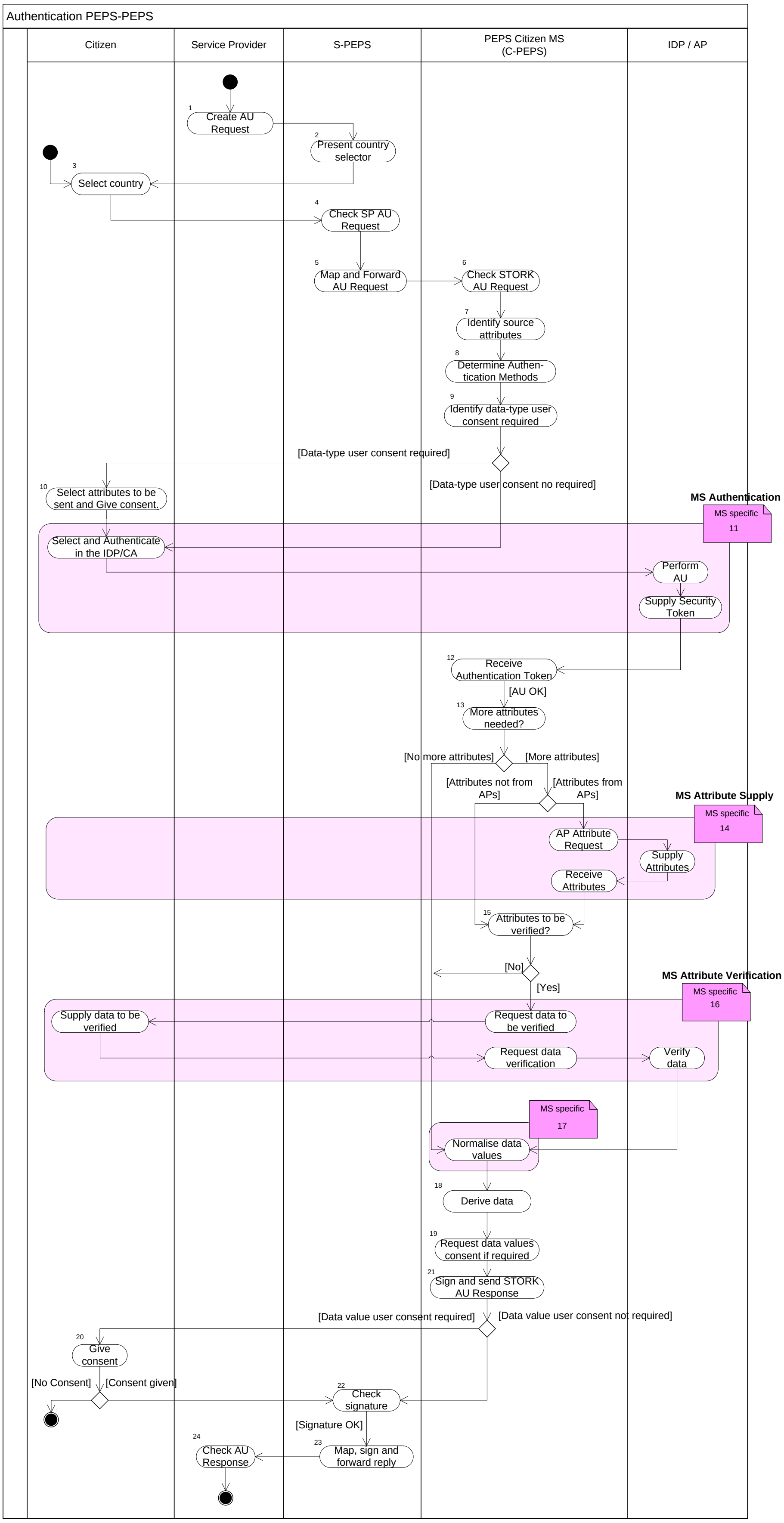

Authentication PEPS-PEPS
Citizen
Service Provider
S-PEPS
PEPS Citizen MS
(C-PEPS)
IDP / AP
1
Create AU Request
2
Present country selector
3
Select country
4
Check SP AU Request
5
Map and Forward AU Request
6
Check STORK AU Request
7
Identify source attributes
8
Determine Authen-tication Methods
9
Identify data-type user consent required
[Data-type user consent required]
[Data-type user consent no required]
10
Select attributes to be sent and Give consent.
MS Authentication
MS specific
11
Select and Authenticate in the IDP/CA
Perform AU
Supply Security Token
12
Receive Authentication Token
[AU OK]
13
More attributes needed?
[No more attributes]
[More attributes]
[Attributes not from APs]
[Attributes from APs]
MS Attribute Supply
MS specific
14
AP Attribute Request
Supply Attributes
Receive Attributes
15
Attributes to be verified?
[No]
[Yes]
MS Attribute Verification
MS specific
16
Supply data to be verified
Request data to be verified
Request data verification
Verify data
MS specific
17
Normalise data values
18
Derive data
19
Request data values consent if required
21
Sign and send STORK AU Response
[Data value user consent required]
[Data value user consent not required]
20
Give consent
[No Consent]
[Consent given]
22
Check signature
[Signature OK]
23
Map, sign and forward reply
24
Check AU Response

Figure 11: Activity Diagram for UC-PP-AU

3.4.2.4 Description of Actions

ID	Actor	Description
AU-PP-DOA-1	SP	**Create AU Request** The Citizen has selected a service that requires authentication. The service provider sends request for User Authentication to the S-PEPS. The request includes the needed authentication trust level (QAA), attributes (mandatory and optional) and some ID for the SP.
AU-PP-DOA-2	S-PEPS	**Present country selector** The S-PEPS delivers a code that can be embedded into the web page of the SP, allowing the Citizen to choose the nationality of her/his eID. The Web page has the nationality flag of each STORK country. With each flag is the name of the country in the native language(s) of the country. All PEPSes must use the same format on this web page.
AU-PP-DOA-3	CIT	**Select country** When a citizen selects a country an Authentication Request is sent to the S-PEPS. This Request will include: QAA, mandatory and optional attributes, SP identification and citizen’s country.
AU-PP-DOA-4	S-PEPS	**Check SP AU Request** The S-PEPS receives the User Authentication request. The S-PEPS checks the Authentication Request of the Service Provider: origin, format and content. <u>Check origin</u> Each country MUST implement their local control policies determining which SP can access their PEPS. In the first place check whether this SP is in the list of known SPs: - In some countries this list is empty (e.g. BE accepts all requests coming from national SPs). - If the SP is not in the list, in some countries we reject the request (e.g.UK only accepts requests coming from known SPs) - Else, if the SP is not in the list and a request coming from an unknown SPs is accepted: · the PEPS must execute some checks to validate that the SP may request authentication to this PEPS. E.g. it may check the domain of the request to validate that the SP from who is receiving the request is from his country. · the number of requests for the last 60s is checked; if this number exceeds a maximum value, the incoming request is rejected (to avoid DoS). <u>Check contents</u> The contents of the request are checked on validity and format: - If the format is incorrect the request is rejected. - The required QAA level and citizen’s country must be specified. <u>Identify the destination</u> The country code is received in the authentication request. If not recognised, the request is rejected. With the country code we obtain the destination (PEPS or V-IDP) where

ID	Actor	Description
		the Authentication Request has to be sent.
AU-PP-DOA-5	S-PEPS	**Map and Forward AU Request** Map the attributes The attributes must be translated to STORK terms. Only defined attributes can be requested. E.g a country may allow requesting *moradaTexto* (address in Portuguese), which should be translated in STORK to textResidenceAddress. The mapping is a specific functionality. In some countries, only a restricted list of attributes can be requested by SPs, according to the SPs profile. If the request includes attributes which are not allowed for him, the request is rejected. Send Authentication Request The S-PEPS uses the gathered information, to prepare a request for User Authentication of the Citizen to C-PEPS. The request data package is signed by the S-PEPS and sent over to the C-PEPS following the security and auditing requirements.
AU-PP-DOA-6	C-PEPS	**Check STORK AU Request** The C-PEPS receives and checks the request for User Authentication. Check origin Check whether the request comes from a trusted colleague (PEPS or V-IDP). If not, the request is rejected.
AU-PP-DOA-7	C-PEPS	**Identify source attributes (include derived data and mapping)** If the S-PEPS is trusted the C-PEPS extracts the request parameters from the request for User Authentication. Check the contents and format of the request. - If the format is incorrect the request is rejected. Check content: Map the attributes Identify for each attribute requested which is the correspondent attribute in the MS. For the data to derive, identify which is the attribute/attributes that can be used to obtain the requested attribute. E.g: If the citizen's age is requested, the C-PEPS must know that he has to obtain the date of birth. This relation must be defined previously. Check completeness Check whether for each of the mapped mandatory attributes there is a national credential (card, certificate, etc) or Attribute Provider that can reveal the required attributes. Identify also the national credentials or Attribute Providers for optional attributes. *Although for the moment only age and IsAgeOver are derived, please note that in the future we may foresee more data to be derived.*
AU-PP-DOA-8	C-PEPS	**Determine Authentication Methods** **QAA**

ID	Actor	Description
		Determine if the national credentials identified in the previous step accomplish with the QAA required by the SP. **IDP selection** If the C-PEPS needs the Citizen to choose which IDP to use, the selection will be done in activity AU-PP-DOA-10.
AU-PP-DOA-9	C-PEPS	**Identify data-type user consent required** Each PEPS applies the legal requirements in the MS. So, in some countries data-type consent is required before asking for the attributes. In other MS's the consent must be given when the values obtained are presented to the citizen. **9. A-** In the first case we should request consent in activity AU-PP-DOA-10. Thus, the user consent described in activity AU-PP-DOA-19 will not be required. **9. B-** In the second case we should request consent directly in activity AU-PP-DOA-19 when the values for the requested attributes are known.
AU-PP-DOA-10	C-PEPS/ CIT	**Select attributes to be sent and give consent** Attributes requested by the SP The data-types requested by the SP are shown to the citizen in the C-PEPS's native language, and user is requested to give his consent. The user can give his consent only for some attributes of the total shown. The user will not be able to disallow mandatory attributes; if he doesn't want to send these, the complete consent is rejected. For each attribute the system will show the user if the attribute is mandatory or optional. If the user consent is denied, or not all the mandatory attributes are allowed or any other case where data were not enough or the required consent wasn't given, C-PEPS will reject the request.
AU-PP-DOA-11	C-PEPS/ CIT	**MS Authentication (Select and Perform Authentication in the IDP/CA)** IDP Selection The C-PEPS provides a list of IDP's that fulfil the trust level requirements from the "request for User authentication". Authentication Authentication is a country specific activity or group of activities. Within this activity, some of the requested attributes may be collected. The Token is verified before ending this activity.
AU-PP-DOA-12	C-PEPS	**Receive Authentication Token** When the authentication has been completed, the C-PEPS issues an Assertion that includes the requested information from the "User Authentication request". If the request was only for e.g. confirmed age, the Assertion will only include confirmation of the age. In other cases the Assertion might include more information such as confirmed name and address of the Citizen, all based on the "User Authentication Request".
AU-PP-DOA-13	C-PEPS	**More attributes needed?** If these attributes can be obtained from APs the process flow goes to step

ID	Actor	Description
		AU-PP-DOA-14 else the process flow goes to step AU-PP-DOA-15. If there are some data to be requested to Attribute Providers, the C-PEPS will request these attributes to the Attribute Providers of his country.
AU-PP-DOA-14	C-PEPS	**MS Attribute Supply (Obtain attributes from APs)** This functionality is a country specific activity or group of activities. For each group of attributes with a common attribute provider, a request is sent to this AP. Each AP will return the values for requested attributes.
AU-PP-DOA-15	C-PEPS	**Attributes to be verified?** If some of the required attributes have to be verified the process flow goes to step AU-PP-DOA-16 else the process flow goes to step AU-PP-DOA-17.
AU-PP-DOA-16	C-PEPS/ CIT	**MS Attribute Verification (Verify attributes)** This functionality is a country specific activity or group of activities. The attributes will be requested to the citizen. The C-PEPS will construct and send the attribute the request to verify the attributes.
AU-PP-DOA-17	C-PEPS	**Normalise data values** The normalisation of data is specific for each country. This function translates the national coding and formats to STORK codings and formats. E.g. gender might locally be indicated as M(ännlich) and W(eiblich), while STORK uses M(ale) and F(emale). Map the value attributes and derive data The attributes received are mapped over the attributes requested by the SP through the S-PEPS. Data values are mapped to STORK nomenclature.
AU-PP-DOA-18	C-PEPS	**Derive data** The attributes received in the C-PEPS are used to form the derived data when needed. The attributes to be derived are constructed, according to the specifications in their description (part 2 of this document). *Although for the moment only age and IsAgeOver are derived, please note that in the future we may foresee more data to be derived.*
AU-PP-DOA-19	C-PEPS	**Request data values consent if required** In those countries where consent must be given for the transmission of data knowing the values to be sent, these data-types and corresponding values are shown and user's consent is requested. The data-types are shown in the C-PEPS's native language. Data-values shown are the original ones, before mapping, except for derived attributes. Data values in text format are not translated. Usually when the value data consent is needed at this point, the data-type consent (activity AU-PP-DOA-10) is avoided.
AU-PP-DOA-20	CIT	**Give data value consent** The user can select to send some attributes of the total shown. He will not be able to disallow the sending of mandatory attributes. C-PEPS receives the data value user consent. If the user consent is denied, C-PEPS will reject the request.

ID	Actor	Description
		If the citizen accepts, the data will be sent together to the S-PEPS.
AU-PP-DOA-21	C-PEPS	**Sign and send STORK AU Response** The C-PEPS signs and sends the Assertion with all the data collected to the S-PEPS. This step will recommend the application of the security and auditing requirements.
AU-PP-DOA-22	S-PEPS	**Check signature** When the S-PEPS has received the assertion, it validates the assertion. If it comes from a trusted PEPS the process flow continues else the Authentication is rejected. This step will follow the security and auditing requirements.
AU-PP-DOA-23	S-PEPS	**Map, sign and forward reply** If the assertion is valid the S-PEPS extracts the content of the assertion. Map the attributes Identify for each attribute received in the Attribute Transfer Response from STORK which is the correspondent attribute in the SP. Build, sign and send response to the SP Build the response with the mapped attributes. Sign and forward the response to the SP. This step will recommend the application of the security and auditing requirements.
AU-PP-DOA-24	SP	**Check AU Response** SP should check the origin of the received response and the content. The SP validates the assertion and will grant access to the requested service if he trusts the assertion.

Table 36: Description of actions for UC-AU-PP

3.4.2.5 Special Requirements

ID	Description
AU-PP-SPR-1	If an error occurs, the user will be notified.
AU-PP-SPR-2	Before data is transferred from C-PEPS to S-PEPS the Citizen must be prompted to give its consent. Depending on the actual implementation (MS specific) this might be done before requesting the data or before sending the data. Furthermore, the user must be able to reject the transmission of optional attributes.
AU-PP-SPR-3	The AU response data must be kept confidential. The origin and integrity of the authentication requests and responses must be ensured. This may be implemented by signing the data and/or by other means (e.g. having a trust relation between communicating parties) following the security and auditing requirements.

Table 37: Special Requirements for UC-AU-PP

3.4.3 Authentication PEPS-MW: UC-AU –PM

In this scenario a citizen of a MW country needs to authenticate to use a service in a PEPS country. The PEPS country has installed a V-IDP. For each MW country a SPWare component (a server side middleware which communicates with the MW running on the Citizen's machine) is interacting with the V-IDP. From the SP's view the process is like a PEPS-PEPS case (except perhaps for different transaction time).

3.4.3.1 Reference Architecture

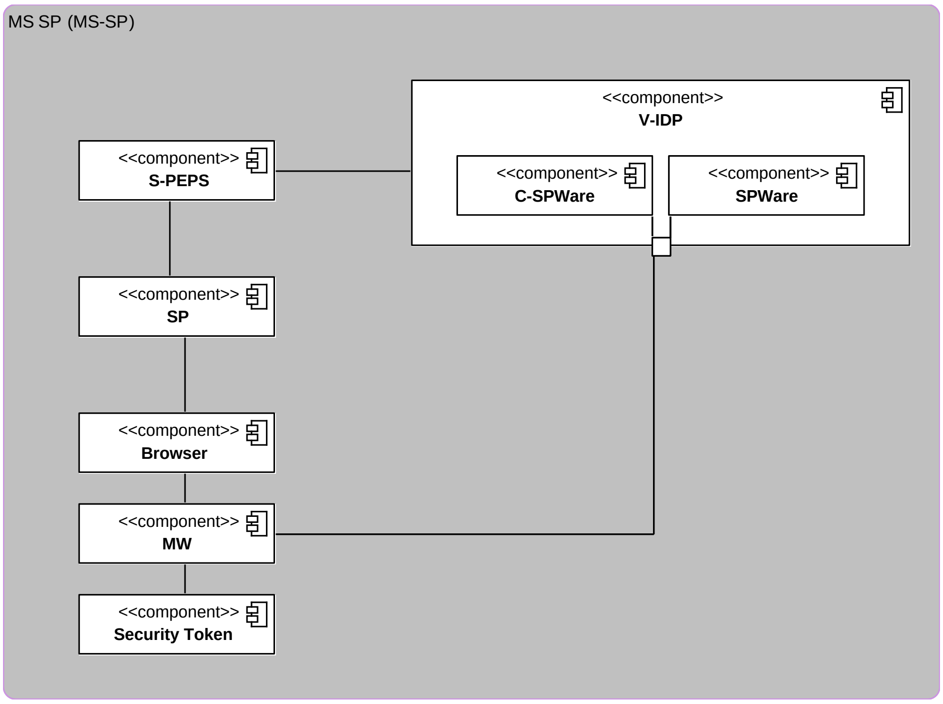


Figure 12: Reference Architecture for UC-AU-PM

Please note that all server components may be geographically located in the SP's country. The following table provides a short description of the used components.

ID	Description
S-PEPS	This SP's national PEPS component. In this scenario it is acting as bridge between SP and V-IDP.
V-IDP	The virtual identity provider. It handles the S-PEPS requests and translates it to the according SPWare requests and vice versa. One V-IDP can handle multiple SPWare components.
SPWare (C-SPWare)	The part of the middleware application that interacts with the Virtual Identity Provider (and on the other hand with the MW).
MW	The part of the middleware application that interacts with the security token (and on the other hand with the SPWare)
Security Token	Token used for authentication and identification.
Browser	The citizen uses the browser to consume a service.
SP	The service provider requiring user authentication/identification.

Table 38: Components for UC-AU-PM

3.4.3.2 Actors

ID	Abbreviation	Description
AU-PM-AC-1	CIT	Citizen using the browser to interact with the system.
AU-PM-AC-2	SP	See according component description.
AU-PM-AC-3	S-PEPS	See according component description.
AU-PM-AC-4	V-IDP	See according component description.
AU-PM-AC-5	C-SPWare	See according component description.

Table 39: Actors for UC-AU-PM

3.4.3.3 Activity Diagram

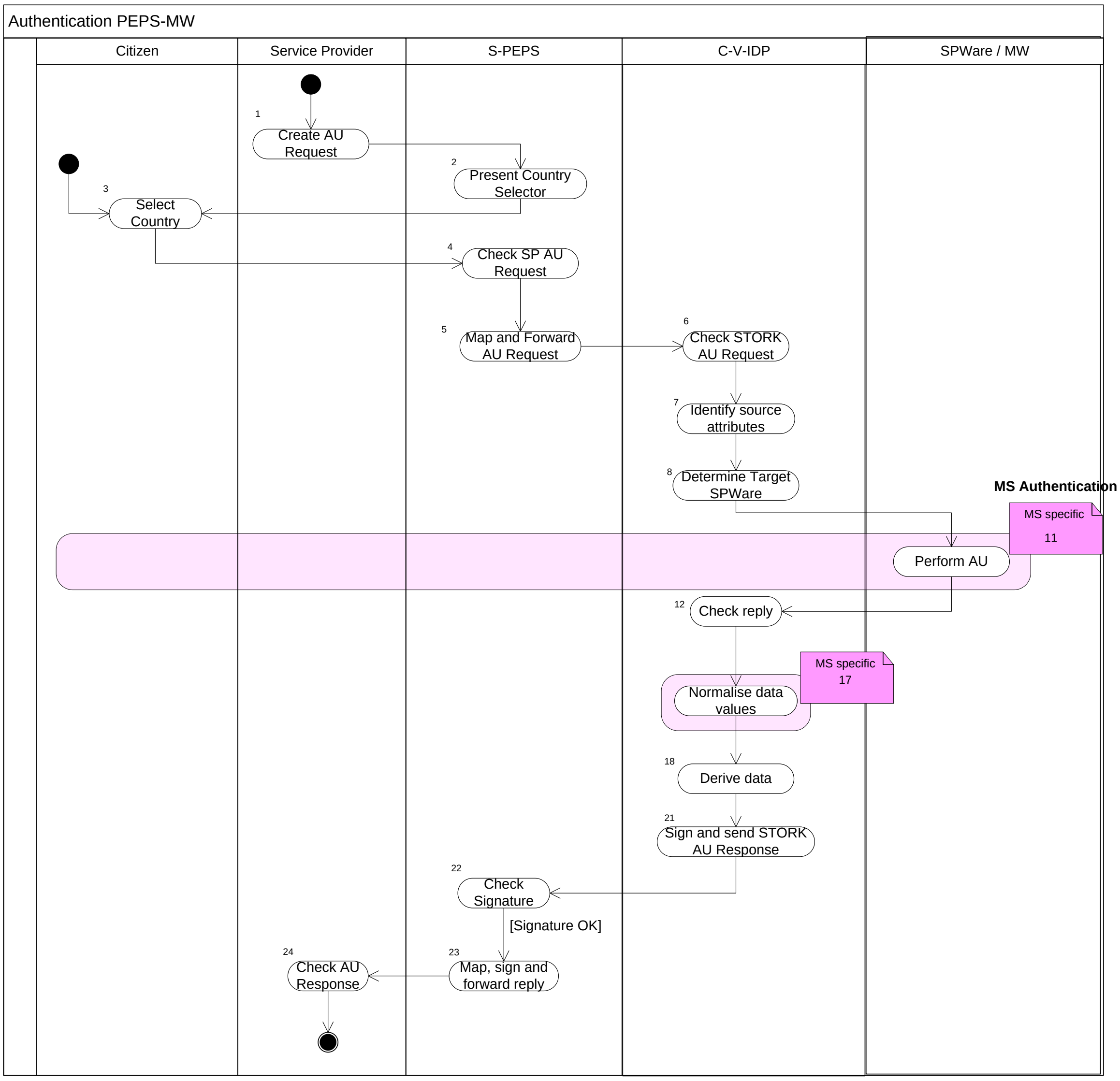


Figure 13: Activity Diagram for UC-AU-PM

3.4.3.4 Description of Actions

Most of the actions below are related or equal to those of section 3.4.2.4. Therefore, we only give a reference to the according action here.

ID	Actor	Description
AU-PM-DOA-1	SP	See AU-PP-DOA-1: **Create AU Request**
AU-PM-DOA-2	S-PEPS	See AU-PP-DOA-2: **Present country selector**
AU-PM-DOA-3	CIT	See AU-PP-DOA-3: **Select country**
AU-PM-DOA-4	S-PEPS	See AU-PP-DOA-4: **Check SP AU Request**
AU-PM-DOA-5	S-PEPS	See AU-PP-DOA-5: **Map and Forward AU Request**
AU-PM-DOA-6	C-V-IDP	See AU-PP-DOA-6: **Check STORK AU Request**
AU-PM-DOA-7	C-V-IDP	See AU-PP-DOA-7: **Identify source attributes (include derived data and mapping)**
AU-PM-DOA-8	C-V-IDP	**Determine Target SPWare** V-IDP determines the Citizen's nationality and selects the according SPWare (C-SPWare).
AU-PM-DOA-11	SPWare /MW	**MS Authentication (Perform AU)** Authentication is a country specific activity or group of activities. Within this activity, some of the requested attributes may be collected. C-SPWare performs the actual authentication process. This step depends on the MS specific MW solution. During this step the user also gives her consent to transfer the data. If any error occurs in the authentication process, then the C-V-IDP will be informed.
AU-PM-DOA-12	C-V-IDP	**Check Reply** V-IDP checks the authentication data it received.
AU-PM-DOA-17	C-V-IDP	See AU-PP-DOA-17: **Normalise data values**
AU-PM-DOA-18	C-V-IDP	See AU-PP-DOA-18: **Derive data**
AU-PM-DOA-21	C-V-IDP	See AU-PP-DOA-15: **Sign and send STORK AU Response**
AU-PM-DOA-22	S-PEPS	See AU-PP-DOA-22: **Check signature**
AU-PM-DOA-23	S-PEPS	See AU-PP-DOA-23: **Map, sign and forward reply**
AU-PM-DOA-24	SP	See AU-PP-DOA-24: **Check AU Response**

Table 40: Description of actions for UC-AU-PM

3.4.3.5 Special Requirements

ID	Description
AU-PM-SPR-1	The client PC must run the client side middleware (either have it installed or in case of e.g. the minimum footprint solution it will be downloaded as needed). If there is no client middleware available the process will abort.

Table 41: Special Requirements for UC-AU-PM

3.4.4 Authentication MW-PEPS: UC-AU–MP

In this scenario a citizen from a PEPS country wants to consume a service from a MW country. For this purpose, the SP forwards the citizen to her/his PEPS, where the actual authentication and identification takes place.

3.4.4.1 Reference Components

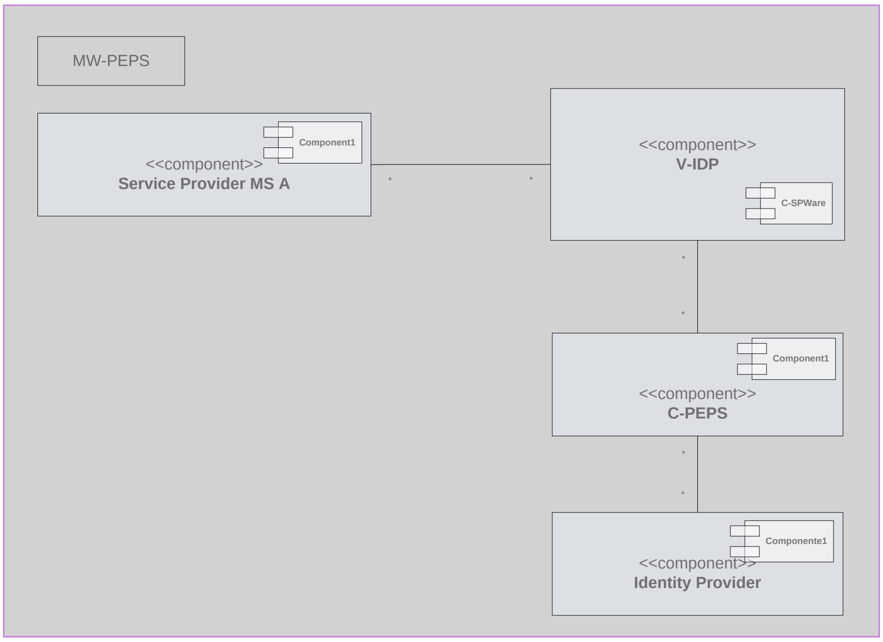


Figure 14: Reference Architecture for UC-AU-MP

The following table provides a short description of the used components.

ID	Description
SP	The service provider requiring user authentication/identification.
V-IDP	Virtual Identity Provider – software provided by MW country running (possibly) in the S-PEPS host.
C-PEPS	The PEPS of the Citizen's member state
IDP	The Identity Provider(s) (IDP) registered at Member State of citizen.

Table 42: Components for UC-AU-MP

3.4.4.2 Actors

ID	Abbreviation	Description
AU-MP-AC-1	CIT	Citizen using the browser to interact with the system.
AU-MP-AC-2	SP	See according component description.
AU-MP-AC-3	V-IDP	See according component description.
AU-MP-AC-4	C-PEPS	See according component description.

Table 43: Actors for UC-AU-MP

3.4.4.3 Activity Diagram

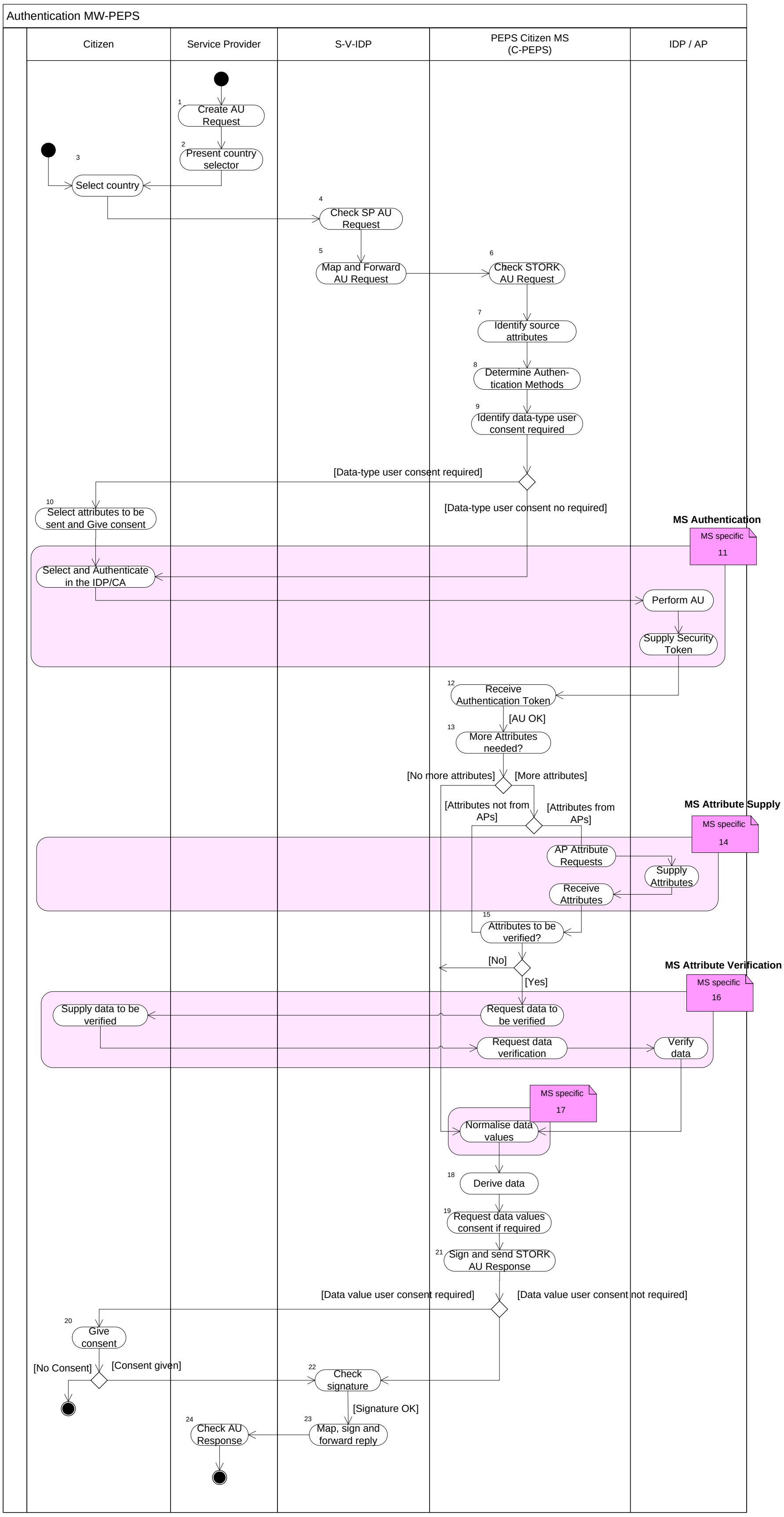

Authentication MW-PEPS
Citizen
Service Provider
S-V-IDP
PEPS Citizen MS (C-PEPS)
IDP / AP
1 Create AU Request
2 Present country selector
3 Select country
4 Check SP AU Request
5 Map and Forward AU Request
6 Check STORK AU Request
7 Identify source attributes
8 Determine Authentication Methods
9 Identify data-type user consent required
[Data-type user consent required]
[Data-type user consent no required]
10 Select attributes to be sent and Give consent
MS Authentication
MS specific 11
Select and Authenticate in the IDP/CA
Perform AU
Supply Security Token
12 Receive Authentication Token
[AU OK]
13 More Attributes needed?
[No more attributes]
[More attributes]
[Attributes not from APs]
[Attributes from APs]
MS Attribute Supply
MS specific 14
AP Attribute Requests
Supply Attributes
Receive Attributes
15 Attributes to be verified?
[No]
[Yes]
MS Attribute Verification
MS specific 16
Supply data to be verified
Request data to be verified
Request data verification
Verify data
MS specific 17
Normalise data values
18 Derive data
19 Request data values consent if required
21 Sign and send STORK AU Response
[Data value user consent required]
[Data value user consent not required]
20 Give consent
[No Consent]
[Consent given]
22 Check signature
[Signature OK]
23 Map, sign and forward reply
24 Check AU Response

Figure 15: Activity Diagram for UC-AU-MP

3.4.4.4 Description of Actions

ID	Actor	Description
AU-PM-DOA-1	SP	See AU-PP-DOA-1: **Create AU Request**
AU-PM-DOA-2	SP	See AU-PP-DOA-2: **Present country selector** This action is performed in the SP instead of being done by the S-PEPS (AU-PP-DOA-2).
AU-PM-DOA-3	CIT	See AU-PP-DOA-3: **Select country**
AU-PM-DOA-4	S-V-IDP	See AU-PP-DOA-4: **Check SP AU Request**
AU-PM-DOA-5	S-V-IDP	See AU-PP-DOA-5: **Map and Forward AU Request**
AU-PM-DOA-6	C-PEPS	See AU-PP-DOA-6: **Check STORK AU Request**
AU-PM-DOA-7	C-PEPS	See AU-PP-DOA-7: **Identify source attributes (include derived data and mapping)**
AU-PP-DOA-8	C-PEPS	See AU-PP-DOA-8: **Determine Authentication Methods**
AU-PP-DOA-9	C-PEPS	See AU-PP-DOA-9: **Identify data-type user consent required**
AU-PP-DOA-10	C-PEPS/ CIT	See AU-PP-DOA-10: **Select attributes to be sent and give consent**
AU-PP-DOA-11	C-PEPS/ CIT	See AU-PP-DOA-11: **MS Authentication (Select and Perform Authentication in the IDP/CA)**
AU-PP-DOA-12	C-PEPS	See AU-PP-DOA-12: **Receive Authentication Token**
AU-PP-DOA-13	C-PEPS	See AU-PP-DOA-13: **More attributes needed?**
AU-PP-DOA-14	C-PEPS	See AU-PP-DOA-14: **MS Attribute Supply (Obtain attributes from APs)**
AU-PP-DOA-15	C-PEPS	See AU-PP-DOA-15: **Attributes to be verified?**
AU-PP-DOA-16	C-PEPS/ CIT	See AU-PP-DOA-16: **MS Attribute Verification (Verify attributes)**
AU-PP-DOA-17	C-PEPS	See AU-PP-DOA-17: **Normalise data values**
AU-PP-DOA-18	C-PEPS	See AU-PP-DOA-18: **Derive data**
AU-PP-DOA-19	C-PEPS	See AU-PP-DOA-19: **Request data values consent if required**
AU-PP-DOA-20	CIT	See AU-PP-DOA-20: **Give consent**
AU-PP-DOA-21	C-PEPS	See AU-PP-DOA-21: **Sign and send STORK AU Response**
AU-PP-DOA-22	S-V-IDP	See AU-PP-DOA-22: **Check signature**
AU-PP-DOA-23	S-V-IDP	See AU-PP-DOA-23: **Map, sign and forward reply**
AU-PP-DOA-24	SP	See AU-PP-DOA-24: **Check AU Response**

Table 44: Description of actions for UC-AU-MP

3.4.4.5 Special Requirements

ID	Description
AU-MP-SPR-1	If an error occurs, the user will be notified.

AU-MP-SPR-2	Before data is transferred from C-PEPS to S-PEPS the Citizen must be prompted to give his/her consent. Depending on the actual implementation (MS specific) this might be done before requesting the data or before sending the data. Furthermore, the user must be able to reject the transmission of optional attributes.
AU-MP-SPR-3	The AU response data must be kept confidential. The origin and integrity of the authentication requests and responses must be ensured. This may be implemented by signing the data and/or by other means (e.g. having a trust relation between communicating parties) following the security and auditing requirements.

Table 45: Special Requirements for UC-AU-MP

3.4.5 Authentication MW-MW: UC-AU–MM

User presents an eID card from a MW country to a service provider located in a MW country.

All necessary negotiation happens only between Client PC and SP server (with the client PC prompting the user for local authentication, option selection and consent). Since there are already MW solutions deployed this section goes more into technical details.

3.4.5.1 Reference Architecture

The following diagram shows the main components and their dependencies.

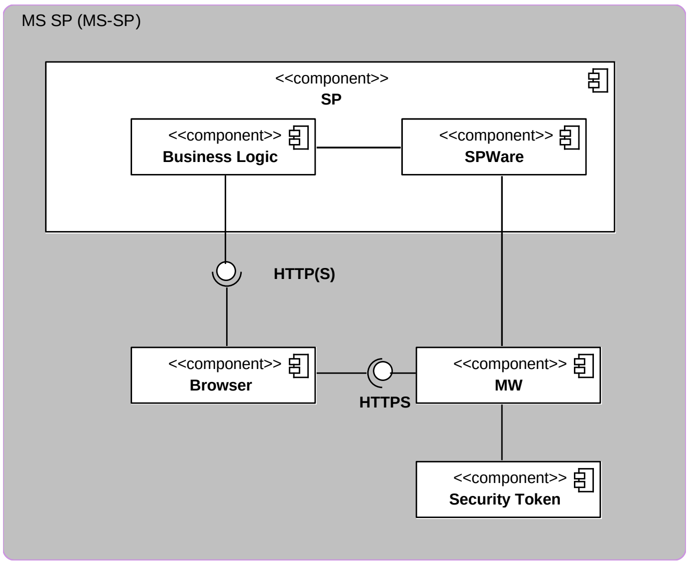


Figure 16: Reference Architecture for UC-AU-MM

The following table provides a short description of the used components.

ID	Description
Business Logic	The SP functionality
SPWare	The part of the middleware application that interacts with the Business Logic (and on the other hand with the MW)

MW	The part of the of the middleware application that interacts with the security token (and on the other hand with the SPWare)
Security Token	Token used for authentication and identification.
Browser	The citizen uses the browser to consume a service (SP)
SP	The service provider. It consists of the Business Logic and the SPWare

Table 46: Components for UC-AU-MM

3.4.5.2 Actors

ID	Abbreviation	Description
AU-MM-AC-1	CIT	Citizen using the browser to interact with the system.
AU-MM-AC-2	SP	See according component description.
AU-MM-AC-3	SPWare	See according component description.
AU-MM-AC-4	Security Token	See according component description.

Table 47: Actors for UC-AU-MM

3.4.5.3 Activity Diagram

The following activity diagram shows the functionality each of the components must implement. The use case is mapped on "technical components". The actor "Citizen" of the use case description is represented by the technical component "Browser" and thus labelled Citizen/Browser.

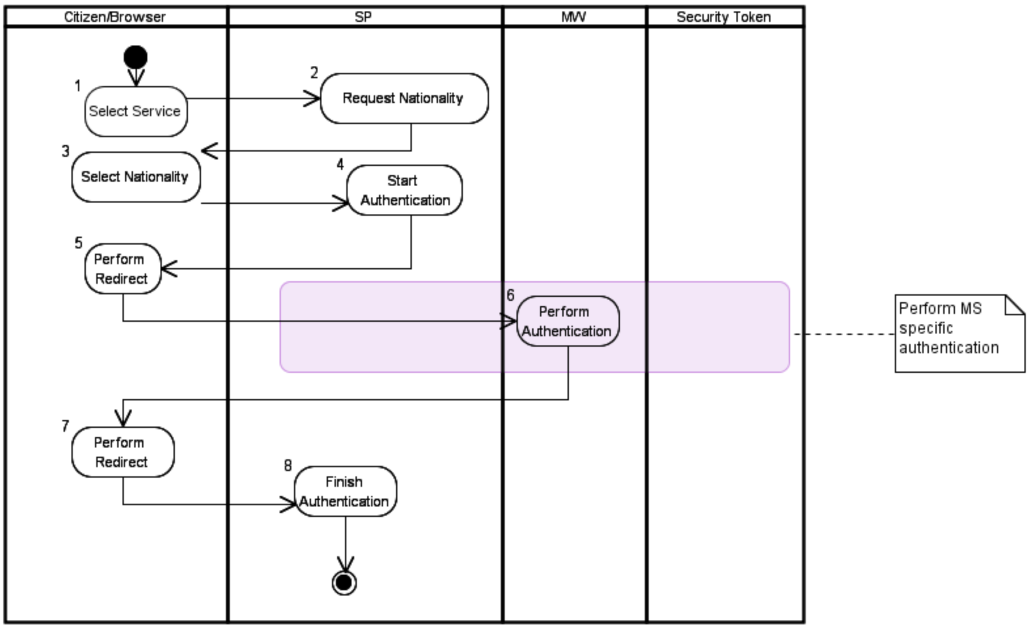


Figure 17: Activity Diagram for UC-AU-MM

3.4.5.4 Description of Actions

ID	Actor	Description
AU-MM-DOA-1	Citizen/Browser	The Citizen requests to use a service and thus uses her browser to contact SP.

AU-MM-DOA-2	SP	SP requests the Citizen for her nationality.
AU-MM-DOA-3	Citizen/Browser	The Citizen selects her nationality.
AU-MM-DOA-4	SP	SP starts the authentication process. In this step the business logic interacts with the SPWare to trigger the authentication process.
AU-MM-DOA-5	Citizen/Browser	The browser redirects the MW, which is running on the Citizen's PC on a dedicated port.
AU-MM-DOA-6	SP, MW, Security Token	The actual authentication and identification process takes place. Here the SPWare the MW and the Security Token interact with each other.
AU-MM-DOA-7	Citizen/Browser	The Browser is requested to redirect to the originating SP.
AU-MM-DOA-8	SP	The SP grants access.

Table 48: Description of actions for UC-AU-MM

3.4.5.5 Sequence Diagram

For clarification the messages of the activity diagram above are shown in the following sequence diagram. It shows the messages after AU-MM-DOA-3 of the activity diagram above. It may serve as an example how this functionality could be implemented in technical sense.

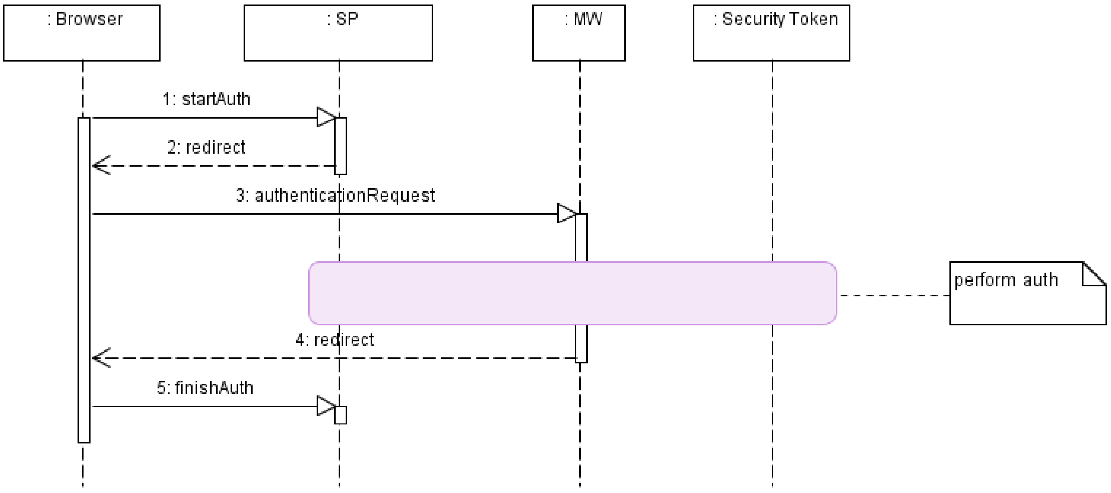


Figure 18: Sequence Diagram for UC-AU-MM

3.4.5.6 Special Requirements

ID	Description
AU-MM-SPR-1	The client PC must run the client side middleware (either have it installed or download it on demand). If there is no client middleware available the process will abort.

Table 49: Special Requirements for UC-AU-MM

3.5 Use Case Attribute Transfer

3.5.1 General description

3.5.1.1 Brief Description

The Service Provider queries some attributes about a user. These attributes are supplied by the IDP and/or by APs.

3.5.1.2 Preconditions

ID	Description
AT-PRE-1	The Citizen is fully authenticated (with his eID)
AT-PRE-2	There exists a contract between the APs and the PEPS, and between PEPSes. In some countries also between the SP and the PEPS.

Table 50: Preconditions of Attribute Transfer

3.5.1.3 Postconditions

ID	Description
AT-POS-1	The Service Provider knows at least the requested attributes he qualified as mandatory.

Table 51: Postconditions of Attribute Transfer

3.5.1.4 Main flow of events

ID	Description
AT-MFE-1	The Service Provider wants to know some attributes about a citizen. Therefore it sends an attribute transfer request to STORK. The request must come from an authorized SP (AT-ALF-1-1). The attribute transfer request consists of mandatory attributes and optional attributes, although SPs are strongly advised to follow the minimal disclosure principle.
AT-MFE-2	STORK recovers the information available for the citizen. STORK extracted information of the received request and the eID: nationality, attributes and QAA required for the SP.
AT-MFE-3	STORK requests data from the Attribute Providers/IDP. STORK can obtain some of the requested attributes from the authentication user credentials (re-authentication is required). This attributes are validated against the corresponding national IDPs/CAs. STORK asks for the user consent ->Depending on the country the consent request will be gathered before asking the APs showing the data-type requested or (as showed in AT-MFE-6) before sending the attributes between MS showing the found values (implicit when the user introduces the PIN in the case of cards) (UC-AT-SR-2 and UC-AT-SR-3). If the user denies consent for mandatory attributes, STORK will reject the request.
AT-MFE-4	STORK looks up the responsible Attribute Provider/IDP and asks for the requested attributes.
AT-MFE-5	The attribute provider/IDP gives attribute values following the member state specific Attribute Supply procedure. If there are some attributes that for legal reasons need to be verified, the citizen introduces

	these attributes and their values; these are verified against the AP/IDP following the member state specific MS Attribute Verification procedure.
AT-MFE-6	STORK requests user consent of values to be transmitted (UC-AT-SR-2 and UC-AT-SR-3). If the user denies consent for mandatory attributes, STORK will reject the request.
AT-MFE-7	STORK checks the received attributes. If some mandatory attributes were not found or not allowed to transfer, the request is rejected and no attributes are sent to the SP. If some optional attributes were not found or not allowed to transfer, STORK sends the found attributes to the SP.
AT-MFE-8	STORK forwards the attributes to the Service Provider.

Table 52: Main flow of events for Attribute Transfer

3.5.1.5 Alternative flows

ID	Condition	Description
AT-ALF-1-1	The Service Provider is not allowed to send any request	The SP must be verified because in some countries any SP can send an Attribute Request to STORK while in others they cannot.
AT-ALF-1-2	The Service Provider is not allowed to ask for some attributes	The SP is not allowed to request (these) attributes. The request is rejected.
AT-ALF-8-1	The Service Provider asks for mandatory attributes that are not found or not allowed in the MS.	Attribute not found or not allowed. Attribute Request is rejected by STORK.

Table 53: Alternative flow of events for Attribute Transfer

3.5.1.6 Special Requirements

ID	Description
UC-AT-SR-1	The attribute transfer request may also contain derived attributes.
UC-AT-SR-2	The Citizen must give her/his consent to the attributes to be transferred. Depending on member state specific requirements this consent might be given over the data-types or over the values. The user can deny the consent to send the attributes above his MS. In this case the contact with the Attribute Providers is avoided and the request rejected.
UC-AT-SR-3	The Citizen must have the possibility to disable optional attributes. In this case the disabled attributes will not be transferred to STORK and further to the Service Provider.
UC-AT-SR-4	The attributes must be protected from unauthorized modification and disclosure during the transfer between IDP/AP and SP.

Table 54: Special Requirements for Attribute Transfer

3.5.1.7 Other Requirements

ID	Description
UC-AU-OR-1	This process is supposed to be executed through the Internet, using standard tools (Microsoft Internet Explorer or Mozilla Firefox)

Table 55: Other Requirements for Attribute Transfer

3.5.2 Attribute Transfer PEPS-PEPS: UC-AT-PP

This section gives a more detailed analysis of the use case attribute transfer in the PEPS-PEPS scenario.

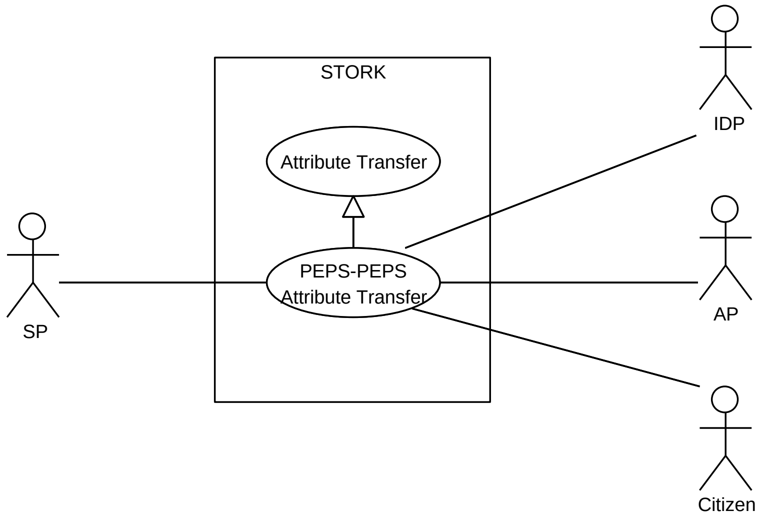


Figure 19: Attribute Transfer Analysis

Figure 19 above shows only one specialization of the Attribute Transfer use case, the "PEPS-PEPS Attribute Transfer". This covers the use case where a Service Provider and the Citizen are located in a PEPS member state.

3.5.2.1 Reference Architecture

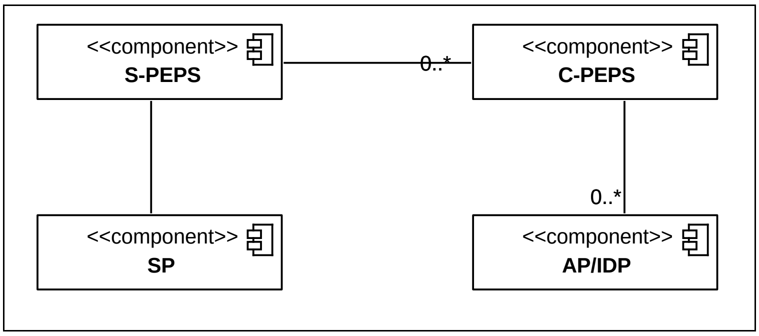


Figure 20: Reference Architecture or UC-AT-PP

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS of the Service Provider's member state

SP	A Service Provider of a PEPS member state.
C-PEPS	The PEPS of the Citizen’s member state
AP/IDP	The Attribute Provider(s) (AP) and Identity Provider(s) (IDP) registered at Member State of citizen

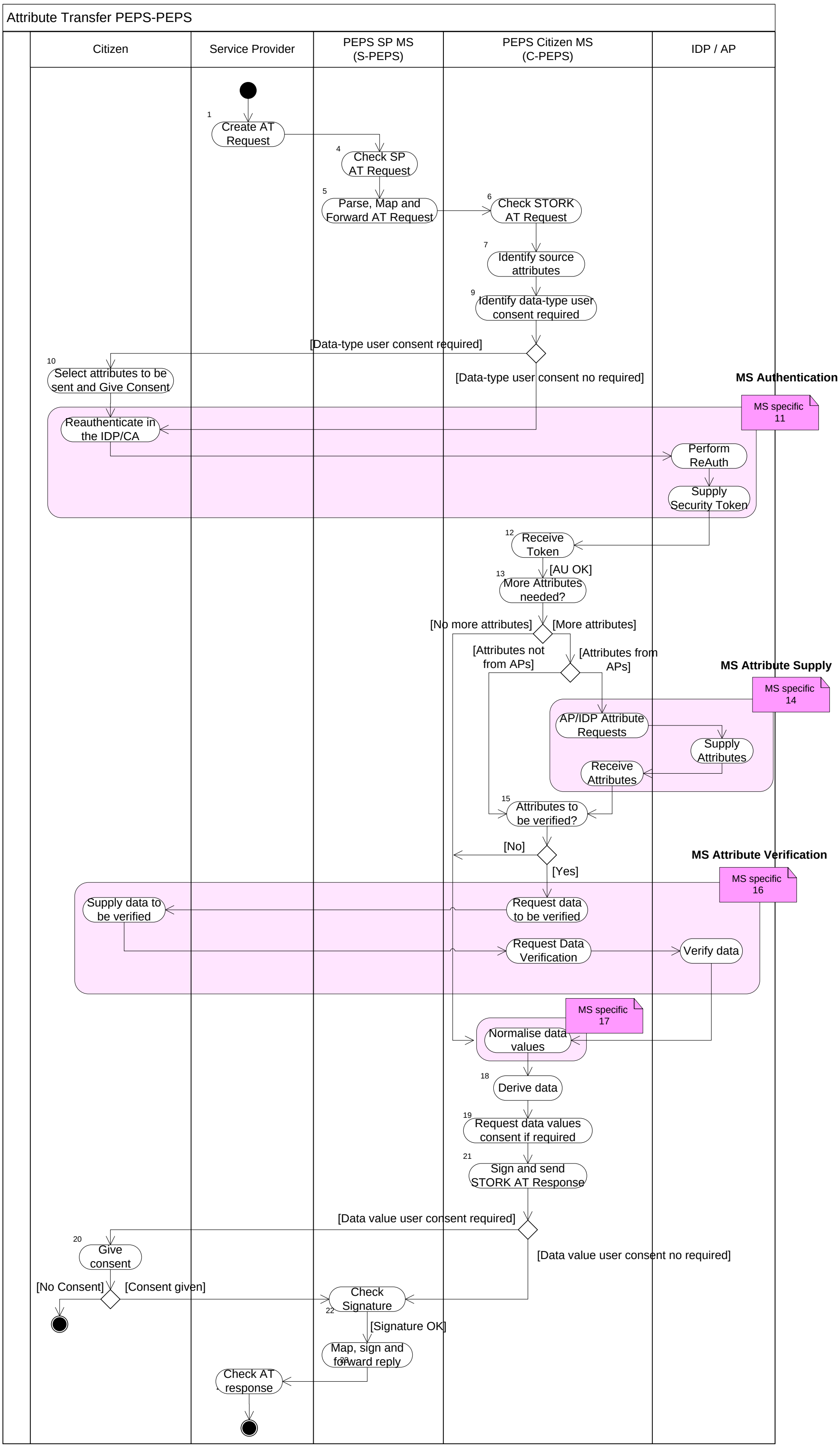
Table 56: Components for UC-AT-PP

3.5.2.2 Actors

ID	Abbreviation	Description
AU-PP-AC-1	CIT	Citizen using the browser to interact with the system.
AU-PP-AC-2	SP	See according component description.
AU-PP-AC-3	S-PEPS	See according component description.
AU-PP-AC-4	C-PEPS	See according component description.
AU-PP-AC-5	AP/IDP	See according component description.

Table 57: Actors for UC-AT-PP

3.5.2.3 Activity Diagram


Attribute Transfer PEPS-PEPS
Citizen
Service Provider
PEPS SP MS (S-PEPS)
PEPS Citizen MS (C-PEPS)
IDP / AP
1
Create AT Request
4
Check SP AT Request
5
Parse, Map and Forward AT Request
6
Check STORK AT Request
7
Identify source attributes
9
Identify data-type user consent required
[Data-type user consent required]
[Data-type user consent no required]
10
Select attributes to be sent and Give Consent
MS Authentication
MS specific 11
Reauthenticate in the IDP/CA
Perform ReAuth
Supply Security Token
12
Receive Token
[AU OK]
13
More Attributes needed?
[No more attributes]
[More attributes]
[Attributes not from APs]
[Attributes from APs]
MS Attribute Supply
MS specific 14
AP/IDP Attribute Requests
Supply Attributes
Receive Attributes
15
Attributes to be verified?
[No]
[Yes]
MS Attribute Verification
MS specific 16
Supply data to be verified
Request data to be verified
Request Data Verification
Verify data
MS specific 17
Normalise data values
18
Derive data
19
Request data values consent if required
21
Sign and send STORK AT Response
[Data value user consent required]
[Data value user consent no required]
20
Give consent
[No Consent]
[Consent given]
22
Check Signature
[Signature OK]
Map, sign and forward reply
23
Check AT response

3.5.2.3.1

Figure 21: Activity Diagram UC-AT-PP

3.5.2.4 Description of Actions

ID	Actor	Description
AT-PP-DOA-1	SP	**Create AT Request** Creates a request for attributes required for his services. This request must contain the citizen’s eID (which includes its nationality), attributes (mandatory and optional) and QAA required by the SP.
AT-PP-DOA-4	S-PEPS	**Check SP AT Request.** The S-PEPS receives the Attribute Request. The S-PEPS checks the Attribute Transfer Request of the Service Provider: origin, format and content. Check origin Each country MUST implement their local control policies determining which SP can access their PEPS. In the first place check whether this SP is in the list of known SPs: - In some countries this list is empty (e.g. BE accepts all requests coming from national SPs). - If the SP is not in the list, in some countries we reject the request (e.g.UK only accepts requests coming from known SPs) - Else, if the SP is not in the list and a request coming from an unknown SPs is accepted: · the PEPS must check the domain of the request to validate that the SP from who is receiving the request is from his country. · the number of requests for the last 60s is checked; if this number exceeds a maximum value, the incoming request is rejected (to avoid DoS). Check contents The contents of the request are checked on validity and format: - If the format is incorrect the request is rejected. - The citizen’s ID-number and required QAA level must be specified.
AT-PP-DOA-5	S-PEPS	**Parse, map attributes and Forward the AT request to the C-PEPS** Parse the request Obtain the nationality of the user identifier. If not recognised, the request is rejected. (XX- number. XX = country code. Number = unique identifier for the citizen). With the country code we obtain the destination (PEPS or V-IDP) where the request has to be sent. The citizen’s eID must be included in the Attribute Request sent. We will obtain also the national IDP where the citizen has been authenticated. Map the attributes The attributes must be translated to STORK terms. Only defined attributes can be requested. E.g: a country may allow to request Direction, which should be translated in STORK to Address. This is specific functionality.

ID	Actor	Description
		In some countries, only a restricted list of attributes can be requested by SPs, according to the SPs profile. If the request includes attributes which are not allowed for him, the request is rejected. Send Attribute Request The S-PEPS constructs the Attribute Request. The Attribute Request is signed by the S-PEPS and sent over to the C-PEPS following the security and auditing requirements.
AT-PP-DOA-6	C-PEPS	**Check STORK AT request** The C-PEPS receives and checks the Attribute Transfer Request from STORK. Check origin Check whether the request comes from a trusted colleague (PEPS or V-IDP). If not, the request is rejected.
AT-PP-DOA-7	C-PEPS	**Identify source attributes (including derived data and mapping)** If the S-PEPS is trusted the C-PEPS extracts the request parameters from the Attribute Request. Check the contents and format of the request. - If the format is incorrect the request is rejected. Map the attributes Identify for each attribute requested which is the corresponding attribute in the MS. For the data to derive, identify which is/are the attribute/attributes that can be used to obtain the requested attribute(s). E.g: If the citizen's age is requested, the C-PEPS must know that he has to obtain the date of birth. This relation must be defined previously. Check completeness Check whether for each of the mapped mandatory attributes, these are available in the national credentials provided by a national IDP used in previous authentication. Or else, if there is an Attribute Provider able to provide these data. Else reject the request (not found). Identify Attribute Providers also for optional attributes.
AT-PP-DOA-9	C-PEPS	**Identify data-type and data value user consent to be applied** Each PEPS applies the legal requirements in the MS. So, in some countries data-type consent is required before asking for the attributes. In other MS's the data consent must be given when the values obtained are presented to the citizen. **9.A-** In the first case we should request consent in activity AT-PP-DOA-10. Thus, the user consent described in activity AT-PP-DOA-19 will not be required. **9.B-** In the second case we should request consent directly in activity AT-PP-DOA-19 when the values for the requested attributes are known.
AT-PP-DOA-10	C-PEPS, CIT	**Select attributes to be sent and give consent.** Attributes requested by the SP The data-types requested by the SP are shown to the citizen in the C-

ID	Actor	Description
		PEPS's native language, and user is requested to give his consent. The user can give his consent only for some attributes of the total shown. The user will not be able to disallow mandatory attributes; if he doesn't want to send these, the complete consent is rejected. For each attribute the system will show the user if the attribute is mandatory or optional. If the user consent is denied, or not all the mandatory attributes are allowed or any other case where data were not enough or the required consent wasn't given, C-PEPS will reject the request.
AT-PP-DOA-11	C-PEPS, CIT	**MS Authentication (Perform Re-authentication)** The IDP where the user has to be re-authenticated can be obtained from the Request. Authentication Authentication or Re-authentication is a country specific activity or group of activities. This activity will be performed against the IDP that authenticated previously the citizen. Within this activity, some of the requested attributes may be collected. The Token is verified before ending this activity.
AT-PP-DOA-12	C-PEPS	**Receive Token** When the authentication has been completed, the C-PEPS issues an Assertion that includes the requested information: attributes and re-authentication.
AU-PP-DOA-13	C-PEPS	**More attributes needed?** If these attributes can be obtained from APs the process flow goes to step AU-PP-DOA-14 else the process flow goes to step AU-PP-DOA-15. If there are some data to be requested to Attribute Providers, the C-PEPS will request these attributes to the Attribute Providers of his country.
AT-PP-DOA-14	C-PEPS	**MS Attribute Supply (Obtain attributes from APs)** This functionality is a country specific activity or group of activities. For each group of attributes with a common attribute provider, a request is sent to this AP. Each AP will return the values for requested attributes.
AT-PP-DOA-15	C-PEPS	**Attributes to be verified?** If some of the required attributes have to be verified the process flow goes to step AU-PP-DOA-16 else the process flow goes to step AU-PP-DOA-17.
AT-PP-DOA-16	C-PEPS, CIT	**MS Attribute Verification (Verify attributes)** This functionality is a country specific activity or group of activities. The attributes will be requested from the citizen. The C-PEPS will construct and send the request to verify the attributes.
AT-PP-DOA-17	C-PEPS	**Normalise data values** The normalisation of data is specific for each country. This function translates the national coding and formats to STORK codings and

ID	Actor	Description
		formats. E.g. gender might locally be indicated as M(ännlich) and W(eiblich), while STORK uses M(ale) and F(emale). Map the value attributes and derive data The attributes received are mapped over the attributes requested by the SP through the S-PEPS. Data values are mapped to STORK nomenclature.
AT-PP-DOA-18	C-PEPS	**Derive data** The attributes received in the C-PEPS are used to form the derived data when needed. The attributes to be derived are constructed, according to the specifications in their description (part 2 of this document).
AT-PP-DOA-19	C-PEPS	**Request data values consent if required** In those countries where consent must be given for the transmission of data knowing the values to be sent, these data-types and corresponding values are shown and user's consent is requested. The data-types are shown in the C-PEPS's native language. Data-values shown are the original ones, before mapping, except for derived attributes. Data values in text format are not translated. Usually when the value data consent is needed at this point, the data-type consent (activity AT-PP-DOA-10) is avoided.
AT-PP-DOA-20	CIT	**Give data value consent** The user can select to send some attributes of the total shown. He will not be able to disallow the sending of mandatory attributes. C-PEPS receives the data value and user consent. If the user consent is denied, C-PEPS will reject the request. If the citizen accepts, the data will be sent to the S-PEPS.
AT-PP-DOA-21	C-PEPS	**Sign and send STORK AT Response** The C-PEPS signs and sends the Assertion with all the data collected to the S-PEPS. This step will recommend the application of the security and auditing requirements.
AT-PP-DOA-22	S-PEPS	**Check signature** When the S-PEPS has received the assertion, it validates the assertion. If it comes from a trusted PEPS the process flow continues else the Request is rejected. This step will follow the security and auditing requirements.
AT-PP-DOA-23	S-PEPS	**Map, sign and forward reply to SP** If the assertion is valid the S-PEPS extracts the content of the assertion. Map the attributes Identify for each attribute received in the Attribute Transfer Response from STORK, the corresponding attribute in the SP. Build, sign and send response to the SP Build the response with the mapped attributes. Sign and forward the response to the SP.

ID	Actor	Description
AT-PP-DOA-24	SP	**Check received AT response** SP should check the origin of the received response and the content. Inform the user of the correct receipt of his certified attributes.

Table 58: Description of actions for UC-AT-PP

3.5.2.5 Special Requirements

ID	Description
AT-PP-SPR-1	If an error occurs, the user will be notified.
AT-PP-SPR-2	During the actual attribute transfer (which is part of the member state specific functionality) the user must give her/his consent.
AT-PP-SPR-3	The response data must be kept confidential. The origin and integrity of the requests and responses must be ensured. This may be implemented by signing the data and/or by other means (e.g. having a trust relation between communicating parties) following the security and auditing requirements.

Table 59: Special Requirements for UC-AT-PP

3.5.3 Attribute Transfer PEPS-MW: UC-AT-PM

This section gives a more detailed analysis of the use case attribute transfer in the PEPS-MW scenario.

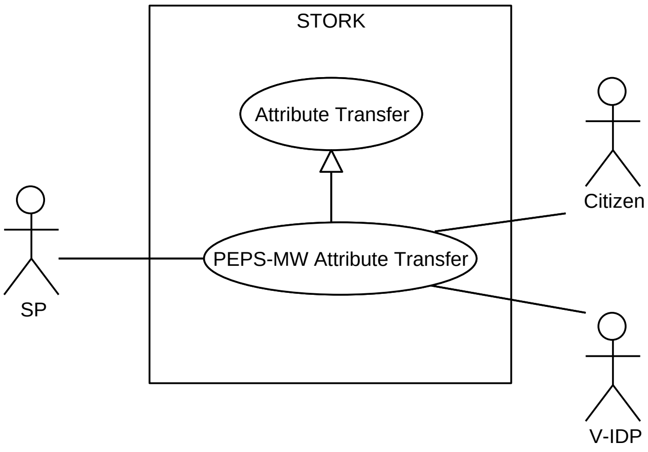


Figure 22: Use Case Diagram for UC-AT-PM

Figure 22 above shows only one specialization of the Attribute Transfer use case, the "PEPS-MW Attribute Transfer". This covers the use case where a Service Provider is located in a PEPS member state and the Citizen is located in a MW member state.

3.5.3.1 Reference Architecture

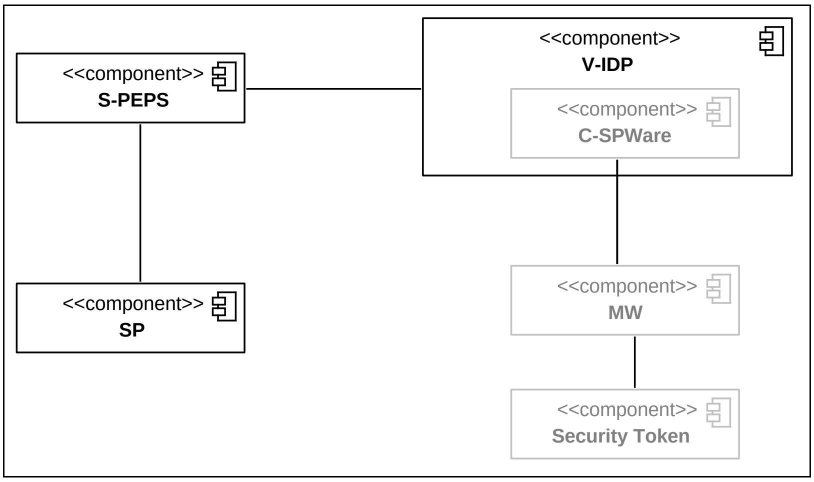


Figure 23: Reference Architecture for UC-AT-PM

The implementation specific components, which actually perform the attribute transfer, are drawn grey.

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS of the Service Provider’s member state
V-IDP	V-IDP interacting with S-PEPS (and geographically located in the country of the S-PEPS). It includes the SPWare of the Citizen’s country.
SPWare (C-SPWare)	The part of the middleware application that interacts with the Virtual Identity Provider (and on the other hand with the MW).
MW	The part of the middleware application that interacts with the security token (and on the other hand with the SPWare)
Security Token	Token used for authentication and identification.
SP	A Service Provider of a PEPS member state

Table 60: Components for UC-AT-PM

3.5.3.2 Actors

ID	Abbreviation	Description
AU-PM-AC-1	CIT	User that interacts with the SP with a national identifier.
AU-PM-AC-2	SP	See according component description.
AU-PM-AC-3	S-PEPS	See according component description.
AU-PM-AC-4	V-IDP	See according component description.
AU-PM-AC-5	C-SPWare	See according component description.

Table 61: Actors for UC-AT-PM

3.5.3.3 Activity Diagram

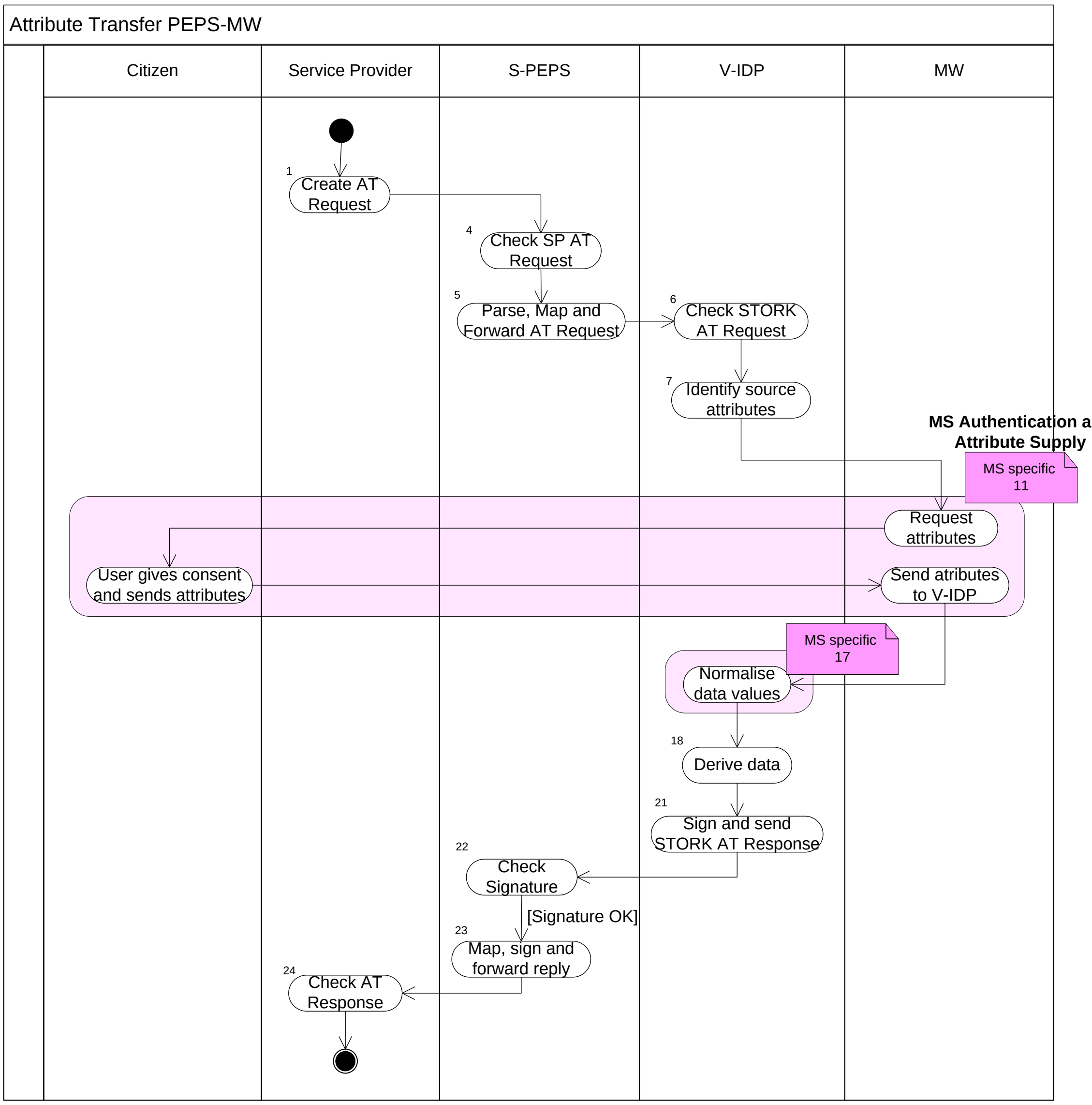


Figure 24: Activity Diagram for UC-AT-PM

3.5.3.4 Description of Actions

ID	Actor	Description
AT-PM-DOA-1	SP	See AT-PP-DOA-1: **Create AT Request**
AT-PM-DOA-4	S-PEPS	See AT-PP-DOA-4: **Check SP AT Request.**
AT-PM-DOA-5	S-PEPS	See AT-PP-DOA-5: **Parse, Map attributes and Forward the AT request to the V-IDP**
AT-PM-DOA-6	V-IDP	See AT-PP-DOA-6: **Check STORK AT request**
AT-PM-DOA-7	V-IDP	See AT-PP-DOA-7: **Identify source attributes (including derived data and mapping)**
AT-PM-DOA-11	V-IDP, MW	**MS Re-Authentication and Attribute Supply** The actual attribute retrieval is member state specific and not analysed here.

ID	Actor	Description
AT-PM-DOA-17	V-IDP	See AT-PP-DOA-17: **Normalise data value**
AT-PM-DOA-18	V-IDP	See AT-PP-DOA-18: **Sign and send STORK AT Response**
AT-PM-DOA-21	S-PEPS	See AT-PP-DOA-21: **Check STORK AT response**
AT-PM-DOA-22	S-PEPS	See AT-PP-DOA-22: **Check signature**
AT-PM-DOA-23	S-PEPS	See AT-PP-DOA-23: **Map, sign and forward reply to SP**
AT-PM-DOA-24	SP	See AT-PP-DOA-24: **Check received AT response**

Table 62: Description of actions for UC-AT-PM

Note that in some cases attributes can be obtained from the eID without reading the certificate. Thus no revocation check is required in this business process. Anyway, if this revocation check has to be performed, it is done by the SPWare or IDP, so it is part of the specific functionalities.

3.5.3.5 Special Requirements

ID	Description
AT-PM-SPR-1	If an error occurs, the user will be notified.
AT-PM-SPR-2	During the actual attribute transfer (which is part of the member state specific functionality) the user must give her/his consent.
AT-PM-SPR-2	The response data must be kept confidential. The origin and integrity of the requests and responses must be ensured. This may be implemented by signing the data and/or by other means (e.g. having a trust relationship between communicating parties) following the security and auditing requirements.

Table 63: Special Requirements for UC-AT-PM

3.5.4 Attribute Transfer MW-PEPS (UC-AT-MP) and Attribute Transfer MW-MW (UC-AT-MM)

The Applications of Service Providers of Middleware Member States are not going to request any additional attributes for the citizen because all the information they use is obtained in the Authentication process. A specific case is the Signature Creation Attribute Request. If used beyond the authentication request, the activity diagram is however different to the one of the authentication request.

So, for this reason the scenarios MW-PEPS and MW-MW are not described for the Attribute Transfer use case.

3.6 Use Case Certificate Validation

STORK foresees limited support for interoperability of digital signatures. There are WP6 pilots which foresee digital signing in the Service Provider web environment whereas the result of signing (i.e. signature format) is left to decide for each Service Provider.

STORK does not provide means for handling and validation of “3rd party-created signatures” i.e. validation of signatures created outside Service Provider environment and control.

3.6.1 General description

Certificate validation (of the user) is an essential feature required by the process of digital signing. After the creation of cryptographic signature with user’s private key there is need to verify whether the corresponding certificate is valid or not. In many cases the proof of validity is stored within a signature data structure.

The Service Provider servicing users with foreign certificates and in need to verify validity of that might be in trouble without STORK help because of the following reasons:

- User’s certificate might not contain information about means for certificate validation such as CRL distribution point and/or OCSP responder address
- OCSP Responder might have access policy restricting free access
- Certificate of OCSP responder required for verification of OCSP response is unknown to Service Provider.

STORK certificate validation functionality provides solutions to abovementioned problems.

Although the primary objective is to cover certificate validation functionality for digital signature (non-repudiation) certificates, it is foreseen to optionally extend this functionality for other end-user certificates (e.g. authentication) as well.

In the context of the services directive already other initiatives exist. Those initiatives or projects put efforts on certificate validation solutions based on trust service lists or signature formats. However, for the progress of STORK an interim solution had been required.

3.6.1.1 Brief Description

The Service Provider queries STORK whether a certificate is valid.

3.6.1.2 Preconditions

SP has retrieved user’s certificate and needs to validate it. This need may occur:

- in the process of creating digital signature
- in authentication in case SP builds TLS connection with client authentication

ID	Description
CV-PRE -1	STORK has capability to find Identity Provider which corresponds to the user’s certificate.
CV-PRE-2	STORK has access rights to query Identity Provider.
CV-PRE-3	SP has capability to verify the signed response.

Table 64: Preconditions for UC-CV

3.6.1.3 Postconditions

ID	Description
CV-POS-1	The Service Provider SP has retrieved a validity confirmation of the certificate in question.

Table 65: Postconditions for UC-CV

3.6.1.4 Main flow of events

ID	Description
CV-MFE-1	SP sends certificate validation request to STORK
CV-MFE-2	STORK looks up the responsible Identity Provider and queries the validity of the certificate in question.
CV-MFE-3	Identity Provider responds to STORK
CV-MFE-4	STORK forwards the response to SP
CV-MFE-5	SP verifies the response

Table 66: Main flow of events for UC-CV

3.6.1.5 Alternative flows

ID	Condition	Description
CV-ALF-1	Identity Provider not found	STORK fails to find the appropriate Identity Provider.
CV-ALF-2	STORK resigns the response	Before forwarding, STORK re-signs the response.

Table 67: Alternative flows for UC-CV

3.6.1.6 Special Requirements

ID	Description
UC-CV-SR-1	User must have X.509v3 certificate.

Table 68: Special Requirements

3.6.1.7 Open Issues

ID	Description
CV- OI-1	The scope of the certificate validation has to be decided. The following description refers to the PEPS-PEPS scenario but the same applies for all other use cases as well. Case 1. In the most complex case, when the SP does NOT have any support for certificate validation, he (the SP) would ask STORK to validate the certificate. In this case, the most appropriate place where to perform the certificate validation task is in C-PEPS (as drawn in the diagram). BUT, most probably in this case SP will have to indicate also to C-PEPS a validation policy, indicating at least the context in which the certificate (to be validated) will be used. Case 2. In a simpler case, SP supports party certificate validation (in the sense that it could have his own modules for constructing the certificate path) but he wants to know the revocation status from STORK. In this case SP creates an OCSP request

ID	Description
	and he wants just to get from STORK the OCSP response. Thus, in this case C-PEPS just forwards the original OCSP response (resigning it if necessary). So, in this case the described use case is wrong, because the certificate validation is done by SP.

Table 69: Open issues on Certificate Validation

Note that certificate validation at a PEPS brings a liability relationship between relying party and a certification service provider.

3.6.2 Certificate Validation PEPS-PEPS: UC-CV-PP

This section gives a more detailed analysis of the use case certificate validation in the PEPS-PEPS scenario.

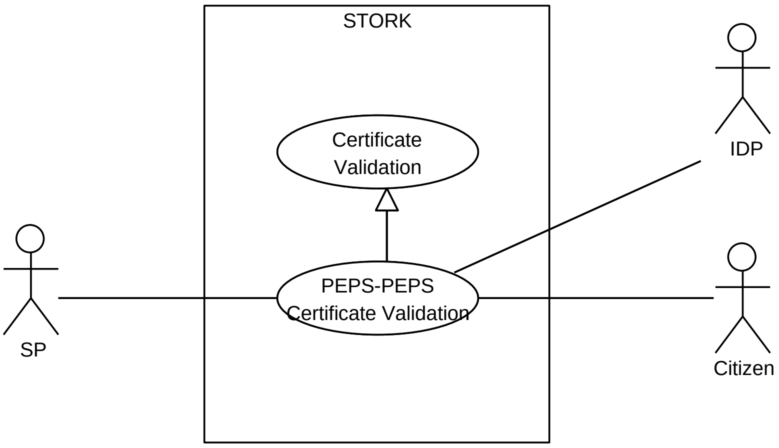


Figure 25: Certificate Validation Analysis

3.6.2.1 Reference Architecture

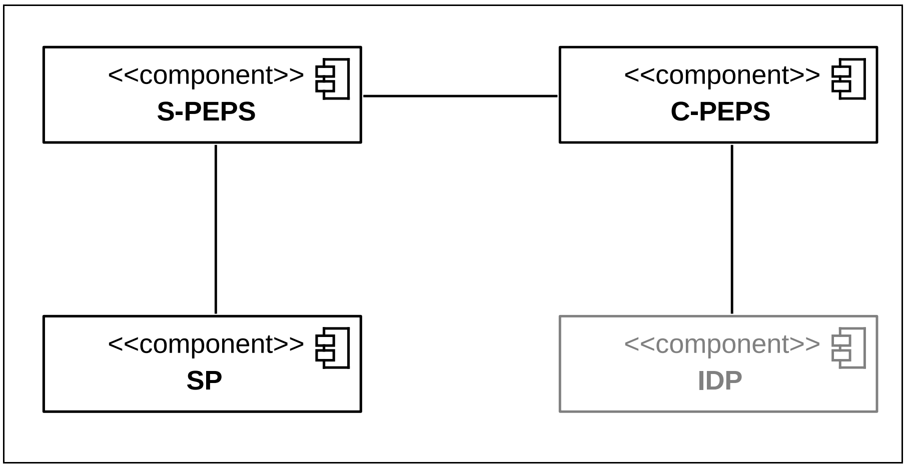


Figure 26: Reference Architecture for UC-CV-PP

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS located in the SP's country.
SP	The service provider requesting certificate validation.
C-PEPS	The PEPS located in the Citizen's country.
IDP	The identity provider (IDP) actually provides the information about the certificate state.

Table 70: Components for UC-CV-PP

3.6.2.2 Actors

See according components.

3.6.2.3 Activity Diagram

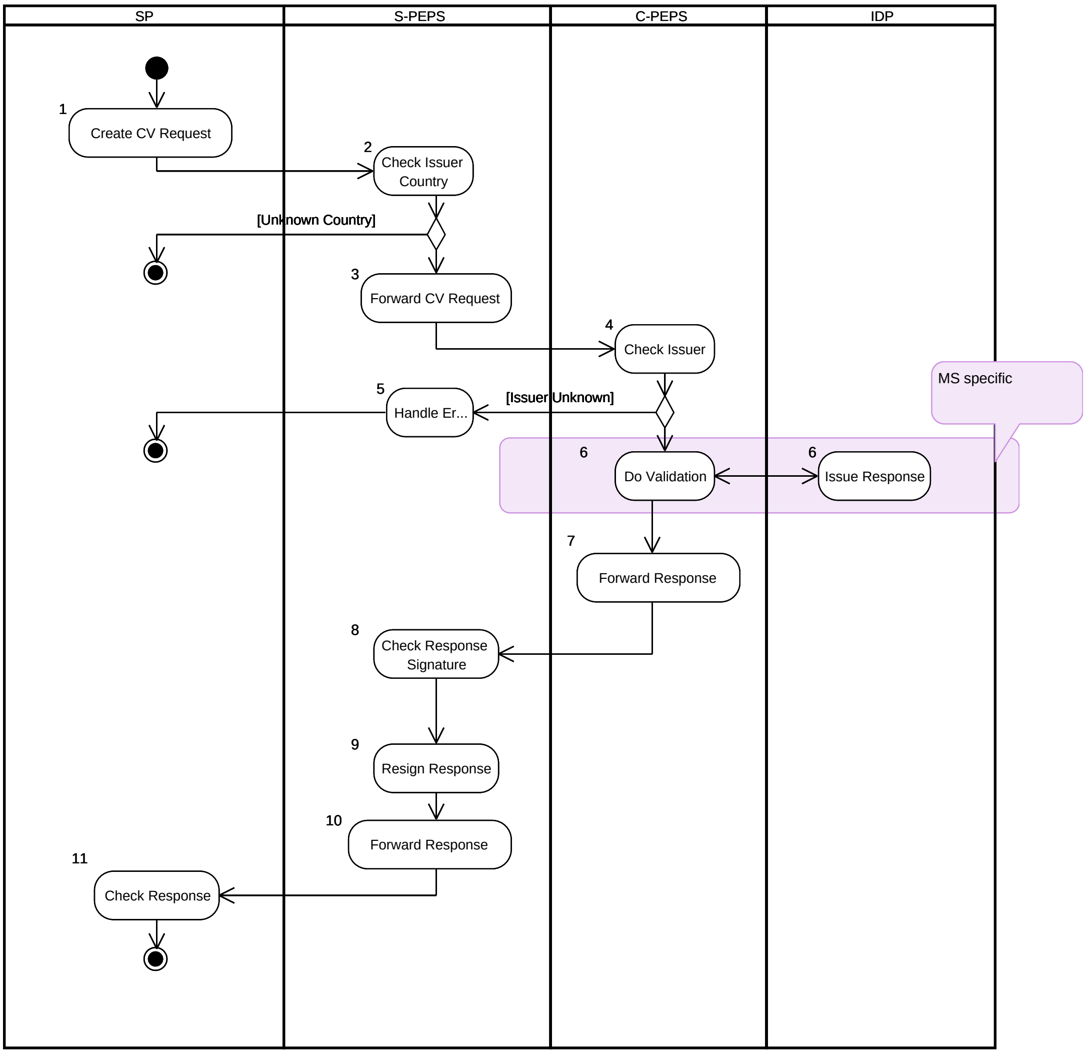


Figure 27: Activity Diagram for UC-CV-PP

3.6.2.4 Description of Actions

ID	Actor	Description
CV-PP-DOA-1	SP	Creates a request for certificate validation.
CV-PP-DOA-2	S-PEPS	S-PEPS checks whether issuer of the certificate in question belongs to country with known PEPS. S-PEPS forwards the request to C-PEPS
CV-PP-DOA-3	S-PEPS	The request is forwarded to C-PEPS.
CV-PP-DOA-4	C-PEPS	Checks, if the issuer of the certificate is known.
CV-PP-DOA-5	S-PEPS	S-PEPS handles the error of an unknown issuer and informs SP about it.
CV-PP-DOA-6	C-PEPS, IDP	In this action the actual certificate validation takes place. Typically, an IDP is involved to get the current state of the certificate. However, the certificate validation procedure is member state specific and not analyzed here.
CV-PP-DOA-7	C-PEPS	Forwards the certificate validity response to S-PEPS.
CV-PP-DOA-8	S-PEPS	Validates the certificate validity response.
CV-PP-DOA-9	S-PEPS	The response is transformed into a member state specific format and resigned.
CV-PP-DOA-10	S-PEPS	The response is forwarded to SP.
CV-PP-DOA-11	SP	SP finally checks the signature of the S-PEPS.

Table 71: Description of actions for UC-CV-PP

3.6.2.5 Special Requirements

ID	Description
CV-PP-SPR-1	In CV-PP-DOA-8 there must be a trust relationship between the C-PEPS creating the certificate validation response and S-PEPS.

Table 72: Special Requirements for UC-CV-PP

3.6.3 Certificate Validation PEPS-MW: UC-CV-PM

3.6.3.1 Reference Architecture

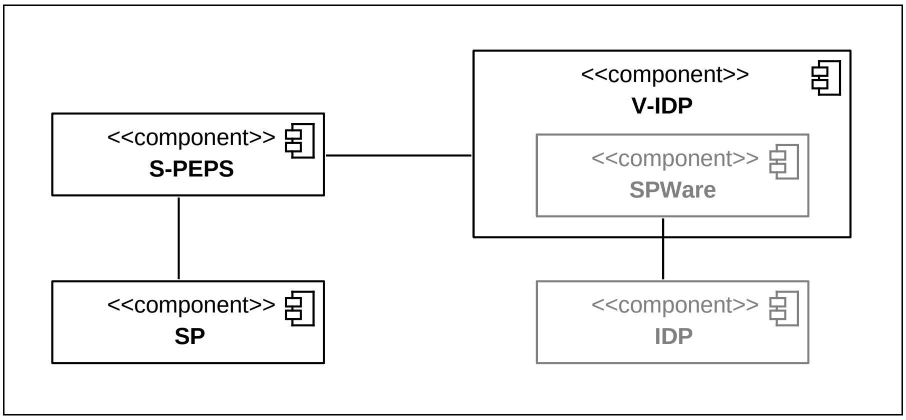


Figure 28: Reference Architecture for UC-CV-PM

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS located in the SP’s country.
SP	The service provider requesting certificate validation.
V-IDP	Virtual Identity Provider – software provided by MW country running (possibly) in the S-PEPS host.
SPWare	SPWare component integrated into the V-IDP that handles member state specific aspects.
IDP	The Identity Provider registered at Member State of citizen.

Table 73: Components for UC-CV-PM

3.6.3.2 Actors

See according components.

3.6.3.3 Activity Diagram

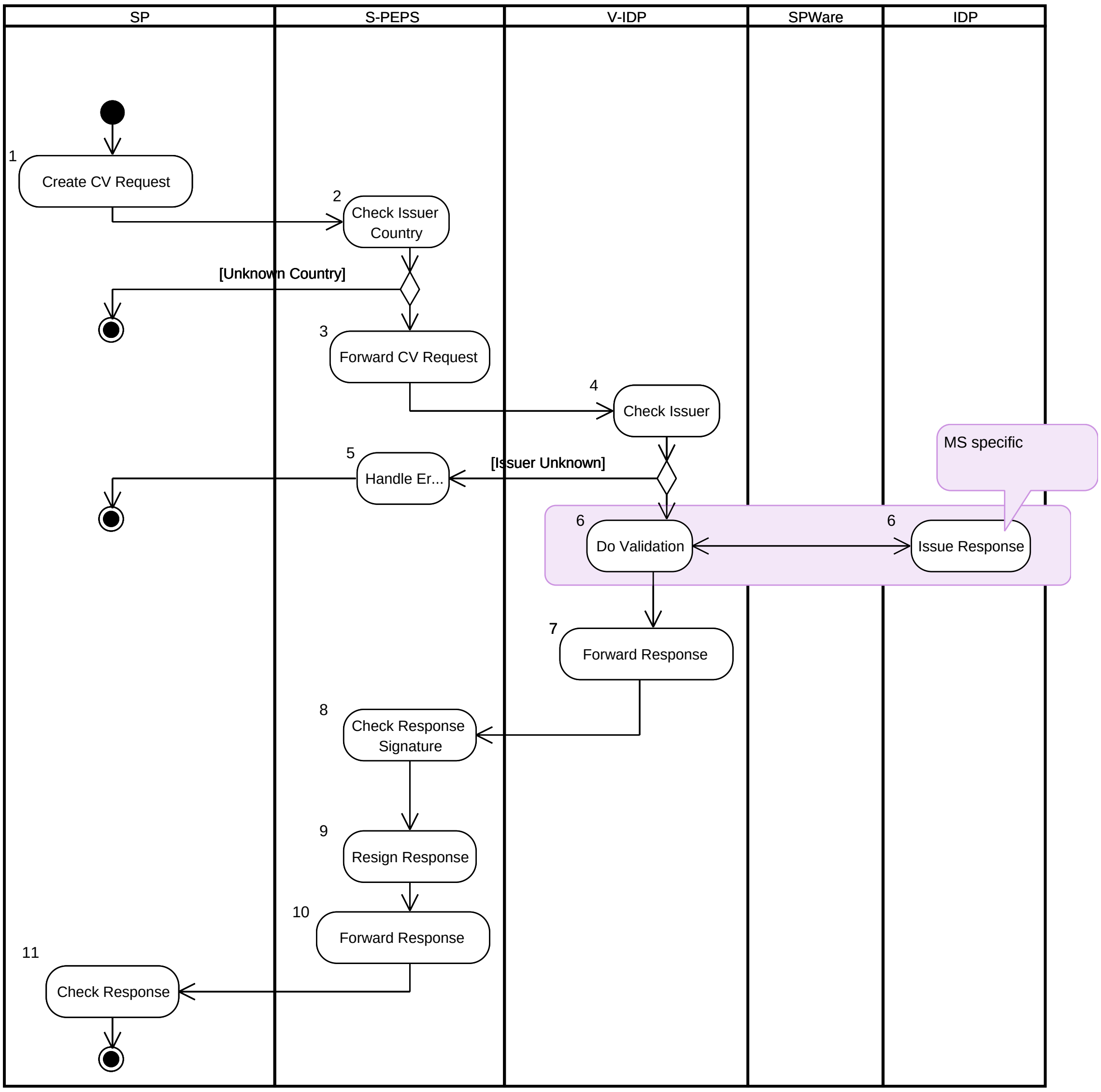


Figure 29: Activity Diagram for UC-CV-PM

3.6.3.4 Description of Actions

ID	Actor	Description
CV-PM-DOA-1	SP	See CV-PP-DOA-1
CV-PM-DOA-2	S-PEPS	See CV-PP-DOA-2
CV-PM-DOA-3	S-PEPS	See CV-PP-DOA-3
CV-PM-DOA-4	C-PEPS	See CV-PP-DOA-4
CV-PM-DOA-5	S-PEPS	See CV-PP-DOA-5
CV-PM-DOA-6	C-PEPS, IDP	See CV-PP-DOA-6
CV-PM-DOA-7	C-PEPS	See CV-PP-DOA-7
CV-PM-DOA-8	S-PEPS	See CV-PP-DOA-8

ID	Actor	Description
CV-PM-DOA-9	S-PEPS	See CV-PP-DOA-9
CV-PM-DOA-10	S-PEPS	See CV-PP-DOA-10
CV-PM-DOA-11	SP	See CV-PP-DOA-11

Table 74: Description of actions for UC-CV-PM

3.6.3.5 Special Requirements

None.

3.6.4 Certificate Validation MW-PEPS: UC-CV -MP

3.6.4.1 Reference Architecture

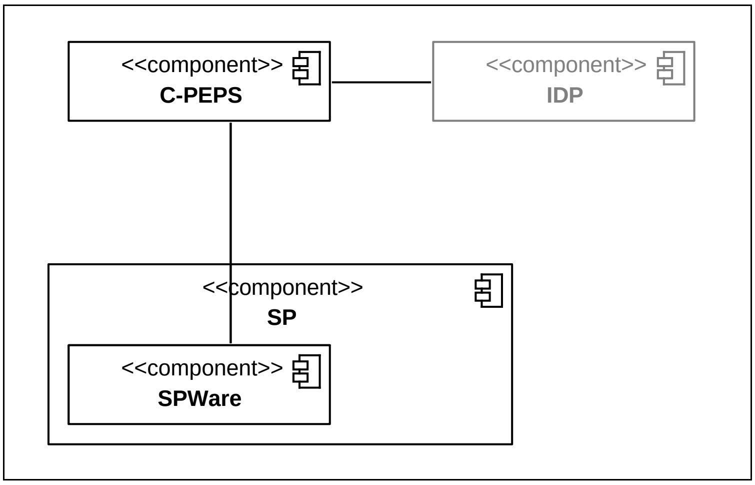


Figure 30: Reference Architecture for UC-CV-MP

The following table provides a short description of the used components.

ID	Description
S-PEPS	The PEPS located in the SP's country.
SP	The service provider requesting certificate validation.
V-IDP	Virtual Identity Provider – software provided by MW country running (possibly) in the S-PEPS host.
SPWare	SPWare component integrated into the V-IDP that handles member state specific aspects.
IDP	The Identity Provider registered at Member State of citizen.

Table 75: Components for UC-CV-MP

3.6.4.2 Actors

See according components.

3.6.4.3 Activity Diagram

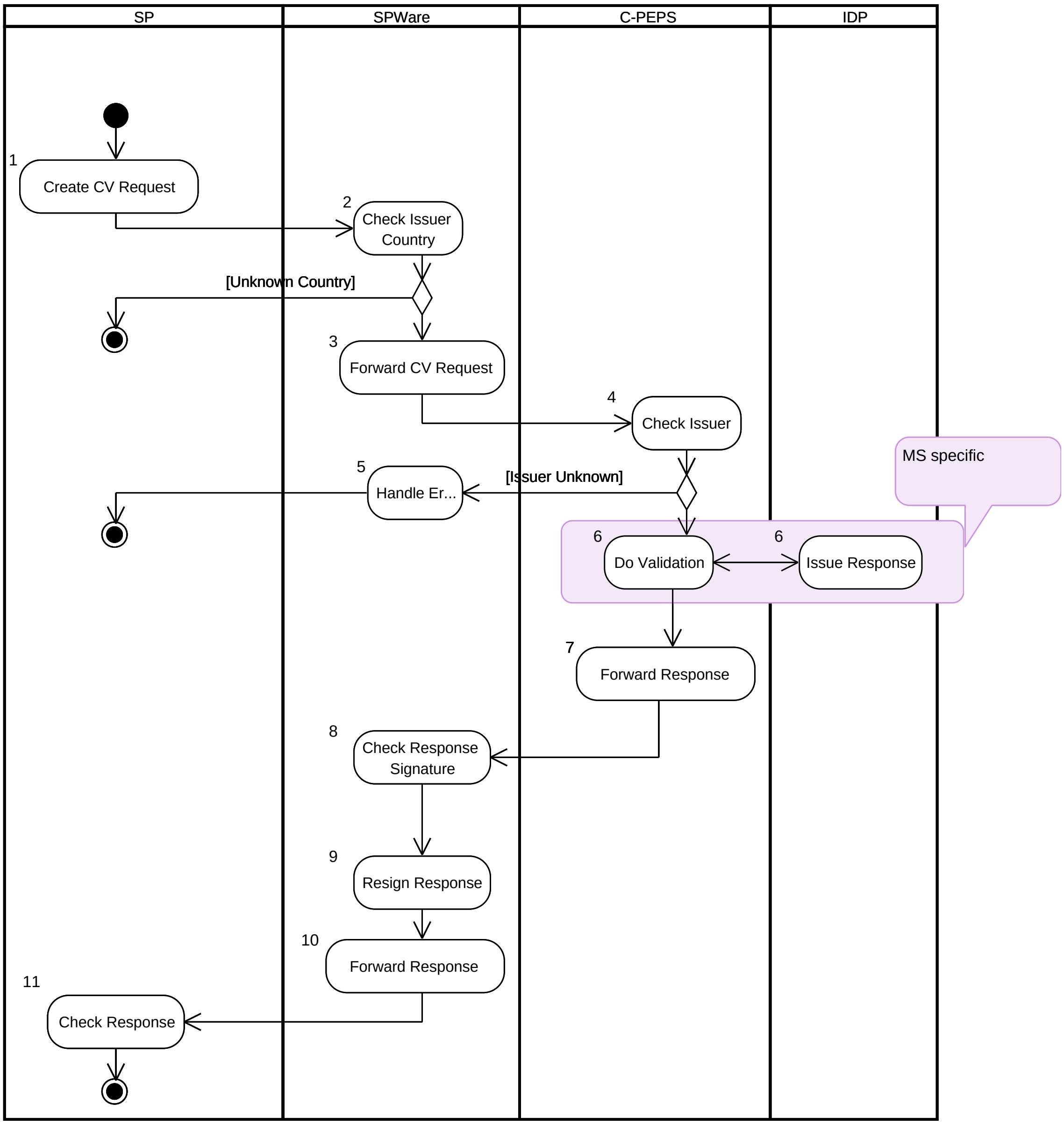


Figure 31: Activity Diagram for UC-CV-MP

3.6.4.4 Description of Actions

ID	Actor	Description
CV-MP-DOA-1	SP	See CV-PP-DOA-1
CV-MP-DOA-2	SPWare	See CV-PP-DOA-2
CV-MP-DOA-3	SPWare	See CV-PP-DOA-3
CV-MP-DOA-4	C-PEPS	See CV-PP-DOA-4
CV-MP-DOA-5	SPWare	See CV-PP-DOA-5
CV-MP-DOA-6	C-PEPS, IDP	See CV-PP-DOA-6

ID	Actor	Description
CV-MP-DOA-7	C-PEPS	See CV-PP-DOA-7
CV-MP-DOA-8	SPWare	See CV-PP-DOA-8
CV-MP-DOA-9	SPWare	See CV-PP-DOA-9
CV-MP-DOA-10	SPWare	See CV-PP-DOA-10
CV-MP-DOA-11	SP	See CV-PP-DOA-11

Table 76: Description of actions for UC-CV-MP

3.6.4.5 Special Requirements

3.6.5 Certificate Validation MW-MW: UC-CV -MM

3.6.5.1 Reference Architecture

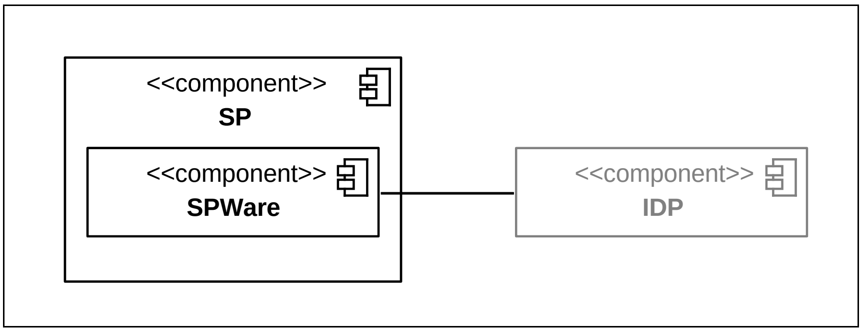


Figure 32: Reference Architecture for UC-CV-MM

The following table provides a short description of the used components

ID	Description
SP	The service provider requesting certificate validation.
SPWare	SPWare component provided by the Citizen’s member state communicating with the IDP.
IDP	The Identity Provider registered at Member State of citizen.

Table 77: Components for UC-CV-MM

3.6.5.2 Actors

See according components.

3.6.5.3 Activity Diagram

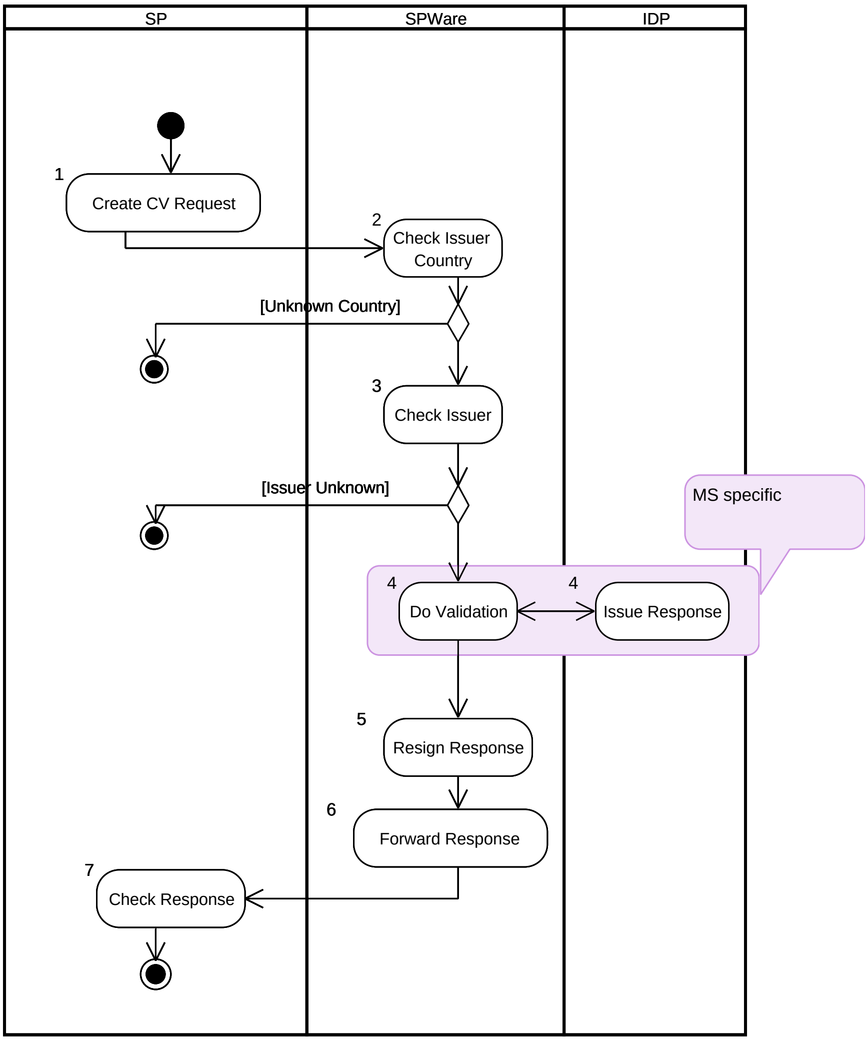


Figure 33: Activity Diagram for UC-CV-MM

3.6.5.4 Description of Actions

ID	Actor	Description
CV-MP-DOA-1	SP	See CV-PP-DOA-1
CV-MP-DOA-2	SPWare	See CV-PP-DOA-2
CV-MP-DOA-3	SPWare	See CV-PP-DOA-6
CV-MP-DOA-4	SPWare	See CV-PP-DOA-5
CV-MP-DOA-5	SPWare	See CV-PP-DOA-9
CV-MP-DOA-6	SPWare	See CV-PP-DOA-10
CV-MP-DOA-6	SP	See CV-PP-DOA-11

Table 78: Description of actions for UC-CV-PP

3.6.5.5 Special Requirements

None.

4 References

[1] Modinis Study on Identity Management in eGovernment: “Common Terminological Framework for Interoperable Electronic Identity Management”, November 23, 2008.

[2] STORK “*Glossary v2.0*”, 2008.

[3] “*Quality Authenticator scheme*”, STORK “*Deliverable 2.3 v.1.0*”, 2009.

[4] “*Technology – induced challenges in privacy & data protection in Europe*”, ENISA AdHoc Working Group on Privacy & Technology, October 2008.

[5] K. Kent, M. Souppaya. Guide to Computer Security Log Management. Recommendations of the National Institute of Standards and Technology, NIST Special Publication 800-92, September 2006

[6] Information System Audit Logging Requirements, SANS Consensus Project, available at http://www.sans.org/resources/policies/info_sys_audit.pdf, accessed 11/02/2009.

[7] P. Ohm, D. Sicker and D. Grunwald. Legal Issues Surrounding Monitoring During Network Research, In Proceedings of IMC’07, October 24-27, 2007, San Diego, California.

[8] Common Criteria: Common Criteria for Information Technology Security Evaluation, Version 3.1, revision 2, October 2007. Part 2: Security Functional Components, http://www.commoncriteriaportal.org/files/ccfiles/CCPART2V3.1R2.pdf

[9] ENISA: Privacy Features of European eID Card Specification, Version 1.0.1, January 2009, http://www.enisa.europa.eu/doc/pdf/deliverables/enisa_privacy_features_eID.pdf

[10] NIST: An Introduction to Computer Security: The NIST Handbook, NIST Special Publication 800-12, December 1997, http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

[11] A. Wombacher, R. Wieringa, W. Jonker, P. Knežević, S. Pokraev. “Requirements for Secure Logging of Decentralized Cross-Organizational Workflow Executions”, In Proceedings of OTM Workshops 2005, LNCS 3762, pp: 526-536, Springer-Verlag, 2005.

[12] E. Gellner. Trust, chesion, and the social order. In “Trust: Making and Breaking Cooperative Relations”, pp: 142-157, Basil Blackwell, 2000.

[13] “D5.7.1 Functional design for PEPS and MW models and interoperability”, September, 2009.

[14] “D5.7.2 Functional design for PEPS and MW models and interoperability”, September, 2010.