Submitted to the EC on 20/05/2012

COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME

ICT Policy Support Programme (ICT PSP)

e-CODEX

e-Justice Communication via Online Data Exchange

ICT PSP call identifier: CIP-ICT-PSP-2009-4

ICT PSP main Theme identifier: CIP ICT PSP 2010 5.2 3: E-JUSTICE SERVICES

Project full title: e-Justice Communication via Online Data Exchange

Grant agreement n°: 270968

Deliverable 5.3 Concept of Implementation

Deliverable ID:	D5.3
Deliverable Name :	Concept of Implementation
Status :draft	V1.0
Dissemination Level :	e-CODEX consortium
Due date of deliverable :	4.5.2012
Actual submission date :	20.05.2012
Work Package :	e-CODEX WP5
Organisation name of lead partner for this deliverable :	Bundesministerium für Justiz, Ministerio de la Justicia
Author(s):	AT and ES
Partner(s) contributing :	DE, EE, NL

The main goal of this document is to describe a convergence approach for building an electronic transport platform able to fulfil the requirements of the Justice domain and which building blocks could be reused in other sectors. The solution described in this document incorporates the inputs arising from a convergence effort in cooperation with the European Commission, other LSPs and standardisation organisations. This effort is intended to create the foundation to design and develop a basic implementation of a cross-border European e-Delivery solution.

The present document will serve as the basic reference to start the development of the pilot implementation of a general electronic transport platform that enables the exchange of documents in different sectors although specially focused on the judicial domain. The challenge faced is to have an operative solution running before end of 2012 based on cutting edge standards like ebMS 3.0, ETSI REM and WS-stack.

This is a living document, as the specifications it contains should be updated according to the experience and results obtained during the next development and integration phases.

History

Version	*Date*	*Changes made*	*Modified by*
0.1	10.01.2012	1st draft of Deliverable 5.3	AT and ES
0.2	18.01.2012	Structural Changes according to the telco on 17Th of January	AT
0.5	31.01.2012	2nd draft ready for review	AT and ES
0.6	02.03.2012	3rd draft update with results from revision	AT, DE, ES, NL
0.7	14.03.2012	4th draft update with new e-Delivery content and reviewed e-Payment chapter	AT, DE, ES
0.9	18.04.2012	Final draft. Updated with inputs from WP5 Madrid meeting, 26-27.MAR.2012 and revision from e-CODEX members, other LSPs and OASIS	AT, DE, ES, NL
1.0	05.04.2012	First complete version. Includes revision to v0.9.	AT, DE, ES

Table of contents

List of Figures

List of Tables

List of Abbreviations

Acronym	Explanation
ACL	Access Control List
ADM	Architecture Development Method
AICPA	The American Institute of Certified Public Accountants (AICPA)
API	Application programming interface
APDU	Application Protocol Data Unit
AS	Applicability Statement AS1[1], AS2[2], AS3[3] and AS4[4] are a family of protocols specifying how to transport data securely and reliably over the Internet.
AT	Austria
BC	Business Collaboration,
BD	Business Document
BE	Belgium
BT	Business Transaction
BusDox	Business Document Exchange Network (PEPPOL)
CA	Certification Authority
CAdES	CMS Advanced Electronic Signatures, published by ETSI as TS 101 733
CBPKI	Cross-border Public Key Infrastructure (Estonia)
CISA	Certified Information Systems Auditor
CMS	Cryptographic Message Syntax, see "CAdES"-Description
COM	Commission
C-PEPS	Citizen Country PEPS Request Invocation Method
CRL	Certificate Revocation List, see "RFC 5280"; http://www.ietf.org/rfc/rfc5280.txt
CROBIES	Cross-Border Interoperability of eSignatures
CZ	Czech Republic
D.I.M	Distributed Identity Management; Technical framework for dealing with identities in the context of web service
DA	Delivery Agent
DAC	Discretionary Access Control
DE	Germany
DES	Data Encryption Standard
DGP	Delivery Gateway Protocol
DGJUST	Directorate General for Justice
DNIe	Documento Nacional de Identidade Electrónico (National ID card / Spain)
DPC	Data Protection and Confidentiality
Driver	Software allowing computer programs to interact with a hardware device

[1] AS1 specification, RFC 3335, http://www.ietf.org/rfc/rfc3335.txt

[2] AS2 specification, RFC 4130, http://www.ietf.org/rfc/rfc4130.txt

[3] AS3 specification, RFC 4823, http://tools.ietf.org/html/rfc4823

[4] AS4 conformance profile,

http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/csprd03/AS4-profile-v1.0-csprd03.odt

DSP	Delivery Service Provider
DSS	Digital signature Standard (NIST)
E2EE	End-to-End Encryption
ebBP	ebXML Business Process, part of ebXML stack
ebMS	ebXML Messaging Services
ebXML	Electronic Business using eXtensible Markup Language, commonly known as e-business XML
EC	European Commission
ECC	Elliptic curve cryptography (NIST)
eCM	e-CODEX member
e-CODEX	e-Justice Communication via Online Data Exchange
ED-GW	Electronic Delivery Gateway
EDI VAN	EDI[5]: Electronic Data Interchange, VAN: Value Added Network
EE	Estonia
eID	Electronic Identity
eIDM	Electronic Identity Management
EIF	European Interoperability Framework
EOPIC	Equivalent of a personal identification code
EPO	European Payment Order
ES	Spain
ETSI	European Telecommunications Standards Institute
EU	European Union
FR	France
GR	Greece
GUI	Graphical User Interface
GW	Gateway
HPC	Health Professional Card
HU	Hungary
ICT	Information and Communications Technology
ID / eID	Identity Document / electronic Identity Document
IDABC	Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens
IEEE	Institute of Electrical and Electronic Engineers
IEEE 830	IEEE Software Requirements Specification
IOP	Interoperability
ISSP	Information System Security Policy
IT	Italy
JHA	Justice and Home Affairs Council
LDAP	Lightweight Directory Access Protocol – RFC 4511
LSP	Large Scale Pilot
LTV	Long Time Validity
MAC	Mandatory Access Control
MIME	Multipurpose Internet Mail Extensions[6]

[5] EDI standards include: UN/EDIFACT, ANSI ASC X12, TRADACOMS, ODETTE

MS	EU Member State
MS-CAPI	Cryptographic Application Programming Interface (Microsoft)
MT	Malta
mTAN	Mobile Transaction Authentication Number
MTOM	Message Transmission Optimization Mechanism (SOAP)
MW	Middleware
NCP	National Contact Points
NIF	National Interoperability Framework
NL	The Netherlands
OCSP	Online Certificate Status Protocol, see "RFC 2560" http://www.ietf.org/rfc/rfc2560.txt
OpenSC	Set of software tools and libraries to work with smart cards with cryptographic capabilities http://www.opensc-project.org
OS	Operating System
PAdES	PDF Advanced Electronic Signature, published by ETSI as TS 102 778
PDP	Policy Decision Point
PEGS	Pan-European e-Government Services
PEPPOL	Pan-European Public Procurement Online (http://www.peppol.eu/)
PEPS	Pan-European Proxy Services (STORK)
PERMIS	PrivilEge and Role Management Infrastructure Standards http://www.openpermis.org
PET	Privacy Enhancing Technologies
PIC	Personal Identification Code
PKCS	Public-key cryptography standards (PKCS #1 - PKCS #15)
PMI	Permission Management Infrastructure
P-Mode	Processing Mode
PT	Portugal
PVP	"Portalverbundprotokoll" (Austrian solution for connecting public authorities)
QAA	Quality Authentication Assurance
QC	Qualified Certificate
RBAC	Role-Based Access Control
REM	Registered E-mail (ETSI TS 102 640)
RFC 3161	RFC 3161 timestamp protocol (The RFC 3161 builds on the IETF standard "Cryptographic Message Syntax", published as the RFC 2630.)
RFC 3161 (GT)	Guardtime RFC 3161 timestamp (GuardTime timestamping protocol and timestamp format are based on the IETF standard "Internet X.509 Public Key Infrastructure Time-Stamp Protocol", published as the RFC 3161.)
RO	Romania
S.A.F.E.	Secure Access to Federated e-Justice / e-Government (German eID Solution)
SAM card	Secure Authentication Module card
SAML 2.0	Security Assertion Markup Language v2.0, authentication request and response format (http://www.oasisopen.org/specs/index.php#saml)

[6] MIME Part One: http://tools.ietf.org/html/rfc2045. Links to additional parts of the specification are given therein.

SHA	Secure Hash Algorithm (NIST)
SOAP	Simple Object Access Protocol
SP	Security Policy
S-PEPS	Service Provider PEPS (STORK)
SP-MW	Middleware Service Provider (STORK)
SPOCS	Simple Procedures Online for Cross- Border Services (http://www.eu-spocs.eu/)
SSCD	Secure Signature Creation Device
SSL V3+	Secure Sockets Layer v3
SSO	Single Sign-On Profile
STORK	Secure idenTity acrOss boRders linked (https://www.eID-stork.eu/)
TAN	Transaction Authentication Number
Time Mark	Timestamp alternative defined in XAdES specification
TLS	Transport Layer Security
TLS 1.0+.	Transport Layer Security Version 1.0 + (RFC 2246, http://tools.ietf.org/html/rfc5246)
TOGAF	The Open Group Architecture Framework
Token	Physical device that an authorized user of computer services is given to ease authentication.
TR	Turkey
tScheme	tScheme is the independent, industry-led, self-regulatory scheme set up to create strict assessment criteria, against which it will approve Trust Services. (http://www.tscheme.org/)
TSL	Trust-service Status List, published by ETSI as TS 102 231
TSP	Trusted Service Provider
TTP	Trusted Third Party
UC	Use Case
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business
VIdP	Virtual Identity Providers
VIdP	Virtual IDP. A system component helping to abstract Pan-European eID interoperability. It either serves as a delegation component between the SP-MW or S-PEPS and the needed SPware (appropriate MW server Component) or enables SP-MW to communicate with other C-PEPS.
WP	Work Package
WP29	Article 29 Data Protection Working Party
WP4	Work Package 4 of the e-CODEX project, Identity (eID for natural and legal persons, roles, mandates and rights) and eSignatures
WSDL	Web Services Description Language
WS-I	Web Services Interoperability[7]
W3C	World Wide Web Consortium
XACML	eXtensible Access Control Markup Language http://saml.xml.org/xacml-oasis-standard
XAdES	XML Advanced Digital signatures, published by ETSI as TS 101 903

[7] http://www.oasis-ws-i.org/

Executive Summary

The e-Justice Communication via Online Data Exchange (e-CODEX) project aims to improve the access of citizens and business to legal means cross border in Europe as well as to improve interoperability between legal authorities within the EU. The goal is to achieve this objective with ideally no impact to the existing national ICT solutions.

In this context transport of data and documents is a key piece of the solution. Any functionality to be developed for a cross-border e-Justice service will necessarily mean transport of information from one country to another also including communication between the e-Justice Portal and some national solution. For this reason a work package explicitly dedicated to transport of data and documents has been defined within e-CODEX: Exchange of documents/data, e-Filing and e-Payment (WP5).

Additionally, it is expected that electronic payment (e-Payment) is to be addressed within the e-CODEX project. E-Payment was decided to be included within this Work Package 5 for transport of data and documents and, for this reason, will be also addressed in this document.

The objective of this document is to define the specifications needed for implementation of the e-CODEX e-Delivery functionality. The basic architecture of this e-Delivery functionality is set up by national gateways which are bilaterally connected to each other. Consequently there is no central hub in the middle. These national gateways interconnect to the national systems respective applications by adapters which handle the mapping between the national format and the standard format used by e-CODEX.

D5.3, chapter 3, now describes the implementation specification of the e-Delivery system. Since the industry standard ebMS V3.0 is used as the base for this implementation D5.3 contains some introduction to ebMS in general (chapter 3). This general description contains the basic architectural concepts of ebMS plus the specification of the ebMS message structure. The final solution as proposed here in D5.3 includes the specification of the transport solution based on ebMS V3.0 extended by the convergence concept that has been developed together with the other LSPs, especially with SPOCS and PEPPOL.

The evidences and the format of these which are necessary for the data exchange are specified based on the ETSI REM standard, which is the same concept as used by SPOCS. The description of these evidences is part of chapter 3 as well.

For the addressing of the gateways, i.e. the addressing of the endpoints, the concept from PEPPOL (SML and SMP) has been considered. However, as a short term solution for e-CODEX it has been decided to use a static addressing solution which should be convenient for the pilots.

Chapter 4 describes the actual implementation and set up of a national gateway in line with the implementation concepts developed in chapter 3. This chapter is the guide and hence of high importance for the piloting Member States to set up their national gateway.

The specification needed for e-payment is done in chapter 5. This specification has the focus on e-payment information access, e-payment execution and e-payment receipt handling. The piloting Member States have different ways how the e-payment is handled. The possibilities vary from direct debit handling outside the e-CODEX process to online payment done with a national system parallel to the e-CODEX process and handing over the payment receipt to the e-CODEX process. The information and means needed for these possibilities are described in chapter 5.

The identification of the competent court is of crucial importance for the civil justice and criminal justice proceedings considered for the pilots. D5.3, chapter 6, specifies how the competent court has to be identified. Originally, the European Court Atlas at the e-Justice Portal has been considered as the database for getting this competent court. However, due to timing reasons with regards to the project schedule, it has been decided together with the EC to provide first national web-services for providing the competent court for the pilot proceedings[8]. The specification of the interface needed for such a web-service is subject of chapter 6.

Chapter 7 defines the test environment needed for the transport infrastructure together with the test scenarios that have to be done in the course of the pilot implementation. The test scenarios cover data and document exchange, e-payment scenarios and the detection of the competent court.

D5.3 is the final document before the implementation of the e-CODEX transport infrastructure is being started. It specifies the building blocks of D5.2 in detail by considering the results of the e-Delivery Task Force. The document is the base for the piloting Member States to set up their national gateway and to define the interface respectively the mapping of their national adapter to this gateway.

Results from development and integration should validate the concept described in D5.3 or may they require updating the content of the document according to the proved implementation. Because of this possible situation, D5.3 should not be considered a finished document; on the contrary it is a living document that should remain open to review also during piloting phase.

[8] This solution can be re-used/replace when the new European Court Atlas is in place.

1. Introduction

1.1. Scope and Objective of Deliverable

This document is the deliverable D5.3 "Concept of Implementation" of Work Package (WP5) of the e-CODEX project.

The goal of this document is to describe how the e-CODEX e-Delivery and e-Payment solutions are to be implemented. These solutions must fulfil the requirements stated in deliverable D5.1 "Requirements" and consider the approach followed by D5.1 that lead to the development of D5.2 "Reusable Assets".

An effort has been done to keep deliverable D5.3 simple although the solution it attempts to describe is not simple. The European Commission asked to align the e-Delivery solution described in this document with the approach described in the "Scenario for the Convergence of LSP e-Delivery solutions" (included as appendix in D5.2) elaborated by the European group that lead to the creation of the current 'European e-Delivery Task Force'[9]. Reasoning for the decision on accepting this scenario as basis for e-CODEX e-Delivery is detailed in an internal document[10].

Nevertheless, an effort has been done to keep deliverable D5.3 simple, as the e-Delivery solution described in this document has been also asked by the EC to align with the approach described in the "Scenario for the Convergence of LSP e-Delivery solutions" (included as appendix in D5.2) elaborated by the European group that lead to the creation of the current 'European e-Delivery Task Force'[11]. Reasoning for the decision on accepting this scenario as basis for e-CODEX e-Delivery is detailed in an internal document[12].

The e-Delivery solution described in deliverable D5.3 will be:

- Compliant with the European convergence e-Delivery solution being defined by the European e-Delivery Task Force.
- The first European convergence e-Delivery solution to come into service.
- Basis for the future broader European cross-border e-Delivery solution.

The document is stored in the project internal workspace (BSCW https://www.jol.nrw.de/pub/bscw.cgi/d605296/index-de.html) and will be available in the download section of the official e-CODEX portal at http://www.e-codex.eu/index.php once accepted by the EC.

---

[9] Members of the e-Delivery Task Force are: DIGIT (European Commission), PEPPOL and SPOCS (other EU co-funded projects), OASIS and ETSI (standardization bodies). A first meeting between the LSPs and the standardization bodies was held on the 12th of January 2012 in Brussels.

[10] "Considerations Regarding the Choice of Transport Infrastructure for the e-CODEX Project and e-Delivery Convergence", is stored on the BSCW https://www.jol.nrw.de/pub/bscw.cgi/d605296/index-de.html and is available upon request.

[11] Members of the e-Delivery Task Force are: DIGIT (European Commission), PEPPOL and SPOCS (other EU co-funded projects), OASIS and ETSI (standardization bodies). A first meeting between the LSPs and the standardization bodies was held on the 12th of January 2012 in Brussels.

[12] "Considerations Regarding the Choice of Transport Infrastructure for the e-CODEX Project and e-Delivery Convergence", is stored on the BSCW https://www.jol.nrw.de/pub/bscw.cgi/d605296/index-de.html and is available upon request.

1.2. WP5 General Objectives and Vision

e-CODEX is a Large Scale Project in the domain of e-Justice[13] that aims to provide to citizens, enterprises and legal professionals an easier access to justice in cross border procedures and to make cross border collaboration of courts and authorities easier and more efficient by creating interoperability of the existing national ICT solutions.

When structuring the work of the e-CODEX project, various considerations were followed to find an optimal organizational structuring. The project aims to develop the interoperability building blocks for e-Justice services in Europe that address the horizontal issues between Member States. Furthermore, these building blocks will need to be proven in real e-Justice services in the countries involved. The project organization will thus need to support these goals properly to ensure that they can also be achieved from a managerial perspective.

Based on the initial building block breakdown for the large scale pilot implementation candidates, WP5 aims to deliver the capability to bind together documents and data that need to be routed or exchanged to enable European cross-border processes in e-Justice.

1.3. Relations to Internal e-CODEX Environment

It is clear that there are dependencies between the different WPs in the context of e-CODEX. The WP5 is strongly linked to WP6 that enhances the overall functionality for e-Justice Services with the "content" of the documents. Another link is to WP7 that provides the IT-groundwork and architecture for interoperability between the systems to be connected, including the security and legal aspects. Beyond that WP4 would establish the identification and electronic signature requirements. WP3 is defining the underlying business processes of the judicial proceedings considered within e-CODEX. Requirements resulting from these business processes have to be considered for the transport infrastructure defined by WP5.

The implementation as basis for the piloting will be done by evaluating work products and assets from other EU-projects that might contribute to this WP. This also means verifying the architectural fit for e-Justice Services (link to WP7) and specifying respectively developing components that are not yet available but which are needed for interoperability of e-Justice services.

Interoperability must be considered with its different dimensions (according EIF v 2.0): there is legal, organizational, semantic and technical interoperability. In practice, interoperability works through inter-administrative agreements, standards definition and use, integration of Public Administrations infrastructures, basic services and systems including the supporting documents and other relevant information. And interoperability must also take into consideration the temporal dimension that may guarantee access to information at any time. The major part of semantics will be in WP6.

[13] See also e-CODEX: towards an interoperable European e-Justice system http://ec.europa.eu/information_society/activities/egovernment/docs/cip/e-CODEX_26_01_2011.pdf

1.4. Relations to External e-CODEX Environment

WP5 has a strong relation to all other LSPs, especially there is a strong relation to SPOCS and PEPPOL with regards to the transport infrastructure developed within these projects. The results, documents and expertise gained by SPOCS and PEPPOL are under consideration from the very beginning. They are part of the analysis done for the category "European Solutions". The analysis targets at the identification of possible re-use of results of PEPPOL and SPOCS especially but also from other LSPs like STORK or ECRIS.

The re-using approach evolved to the question on how e-Delivery platforms of existing Large Scale Pilots (PEPPOL, SPOCS and STORK) can converge towards a single solution, which is suitable for them as well as for the new LSP e-CODEX. As underlined by the European Commission the acceptance and the support coming from the industry is a very important success factor. Therefore such an e-Delivery solution should also include industry standards like ebXML in the area of B2B and ETSI REM.

The Commission, together with some EU co-funded projects (e-CODEX, PEPPOL and SPOCS) and standardization bodies (OASIS and ETSI) have created the aforementioned 'European e-Delivery Task Force' aimed to work on the definition of a European convergence e-Delivery solution capable of fulfil the requirements of current and future LSPs. This solution should be based on standards and capable to evolve. Although considering that the European convergence e-Delivery solution goals are broader than e-CODEX project, the e-Delivery solution described in this document is the first product arising from this task force, ensuring it is completely aligned with the future European cross-border e-Delivery solution.

1.5. Quality Management

Deliverable 5.3 has been provided as a first draft (version 0.5) for a commentary review 3 months before the final delivery was planned. The review participants are all work package partners plus the External Quality Manager. The review comments gained are collected by the work package leaders[14] and processed for the updated version (0.7) which is delivered 2 months before the deadline of the final delivery. A second commentary review has been done using this version 0.7 of D5.3 resulting in the creation of v0.9, the final draft. The participants are all work package partners plus the External Quality Manager and members of the European e-Delivery Task Force. Once the final draft has been accepted by all stakeholders v1.0 is delivered to the European Commission.

The processing of all review comments is documented in the inspection report, which lists the review comments line by line including a statement how the respective review comment has been processed. The inspection report is published together with the update document of D5.3.

Additionally to this commentary review the document has been inspected and discussed in detail in the workshop in Madrid on 26th and 27th March 2012 by the WP5 partners.

[14] WP5 is co-lead by Austria and Spain

1.6. Risk Management

The risks as identified in the course of the creation of deliverable D5.3 and their probability and possible impact are as follows:

ID	Description	Probability		Impact		Expected value		Response	Owner
		inherent	residual	inherent	residual	inherent	residual		
1	The MS decide not to participate in WP5.	medium	low	high	high	high	high	reduce	WP5
2	D5.3 is not finished in time due to lack of availability of resources from contributors.	medium	medium	medium	medium	medium	medium	accept/ transfer	WP5
3	The individual solutions of the MS are very proprietary and have hardly anything in common.	medium	medium	high	high	high	high	reduce	all
4	Legislative resp. judicial reasons jeopardizing a common standard, e.g. different judicial (national) rules are	medium	medium	high	high	high	high	reduce	all

ID	Description	Probability		Impact		Expected value		Response	Owner
		inherent	residual	inherent	residual	inherent	residual		
	jeopardizing a common technical solution								
5	D5.3 is not finished in time since there is no aligned calendar of the different WPs	high	High	high	high	high	high	accept/ transfer	WP5/ WP1
6	D5.3 is not finished in time due to dependencies from other WPs	high	high	high	high	high	high	accept/ transfer	WP5/ WP1
7	D5.3 is not finished in time due to alignment with European e-Delivery Task Force could take too much time.	medium	medium	medium	medium	medium	medium	accept/ transfer	WP5
8	D5.3 is not finished in time due missing legal clarifications	medium	medium	high	high	high	high	accept/ transfer	WP5/ WP7
9	Interconnection to	high	high	high	high	high	high	Having/	WP5/

ID	Description	Probability		Impact		Expected value		Response	Owner
		inherent	residual	inherent	residual	inherent	residual		
	judicial atlas not available in time							executing plan B (WS-IF to pilot MS-directory)	WP1

Table 1: Risks

1.7. Structure of the document

The document is structured as follows:

Chapter	Description
1. Introduction	Present the document and describe the work done
2. Methodology of work	Description on how the work presented in present document has been developed
3. European e-Delivery transport infrastructure	Description of the selected standard (ebMS), message specifications, discovery, addressing and endpoint capabilities, evidences, encryption and other details
4. Gateway	Collection of all the gateway specifications
5. e-Payment specifications	Description of the e-Payment building block
6. Directory of Judicial Atlas	Description of the directory approaches
7. Specification validation tests	Description of the test to be used to validate the specifications collected in this document
8. Traceability	Specifications included in the present document are traced against requirements described in D5.1
9. Conclusion	Gather the main conclusions derived from the work presented in present document

Table 2: Document Structure

2. Methodology of work

The specifications collected in this document stem from:

- Deliverable 5.1 Requirements
- Deliverable 5.2 Reusable Assets
- Scenario for the Convergence of LSP e-Delivery solutions

A first content structure was proposed by WP5 leaders in a face to face meeting (Vienna, 24-25th October 2011) where the development of the specifications was launched. After modifications were agreed by the contributors the structure served as basis for the tasks distribution.

Item	Chapter	Who **bold** is "lead"
Introduction	1	**ES**
Methodology	2	**ES**
European e-Delivery Transport Infrastructure	3	**DE**, EE
Evidences	3.5	**DE**, IT[15], AT
Gateway	4	**NL, AT**, FR
e-Payment	5	**ES**, AT
Directory of Judicial Atlas	6	**NL**, FR
Specifications validation tests	7	**AT, ES**
Traceability	8	**ES**
Conclusions	9	**AT, ES**

Table 3: D5.3 task distribution

WP5 internal coordination of the development has been carried out through telephone conferences.

Within e-CODEX, inputs from Business Process Modelling activities have been considered and alignment with the other technical work packages took place in a face to face meeting held in Tallinn in the 19th and 20th December 2011.

Coordination with other stakeholders has been carried out through face to face meetings:

- European Commission – requirements for the e-Justice Portal (11th January 2012 and 14th March 2012)
- e-Delivery Task Force[16] – definition of a convergence solution (12th January 2012)

Specifications collected in D5.3 will lay the foundations for the European e-Delivery solution that the e-Delivery Task Force aim to define and the European Commission intend to develop broadly in a

[15] IT shall especially support for ETSI REM

[16] Members of the e-Delivery Task Force are: DIGIT (European Commission), e-CODEX, PEPPOL and SPOCS (other EU co-funded projects), OASIS and ETSI (standardization bodies).

future LSP. Involving the e-Delivery Task Force in the revision cycle of deliverable D5.3 has been considered more than appropriate.

The calendar for the development of the D5.3 document settled by WP5 is:

Date	Step	Who
31.01.2012	Updated v0.5	WP5
01.02.2012	Start 1st review	WP5
20.02.2012	End 1st review	
29.02.2012	Updated v0.6	WP5
01.03.2012	Start 2nd review	WP5
14.03.2012	End 2nd review	
14.03.2012	**Updated v0.7**	**WP5**
14.03.2012	Start 3rd review	**WP5+e-Delivery Task Force MB + PD + EQM**
22.03.2012	End 3rd review	
26-27.03.2012	**D5.3 meeting**	**WP5+e-Delivery Task Force**
18.04.2012	**Updated v0.9**	**WP5**
18.04.2012	Start Final review	WP5+MB+PD+EQM
25.04.2012	End Final review	
04.05.2012	**Updated v1.0**	**WP5**
04.05.2012	**D5.3 v1.0 Ready to be delivered**	**WP5+WP1**

Table 4: D5.3 calendar

3. The European e-Delivery Transport Infrastructure

Several European Projects, including the Large Scale Pilots (LSPs) have each developed their own transport infrastructures. The European Commission as well as these projects themselves have emphasised the benefits of reuse, and it was stipulated that the e-Delivery solutions of LSPs PEPPOL, SPOCS and e-CODEX should "converge" over time towards a common standard to be used by all three of these projects and future European initiatives.

This converged solution should incorporate the main result from PEPPOL and SPOCS, and it was considered beneficial to also build upon an established and widely used B2B communication standard which has already gained some industry acceptance.

To reach the above goals a group of technical experts (from the LSPs and other projects and also standardization organisations) created a scenario for a common transport solution, which was subsequently presented to the European commission and agreed upon by the LSPs as the way to move forward in a cooperation towards a European e-Delivery platform. E-CODEX will start to implement its transport solution based on this scenario.

Participants from the LSPs and standardisation bodies (constituting the „European e-Delivery Task Force") have come together to start the standardization activities and to define and set up the process (e.g. some of them participate in the OASIS BDX TC) as well as involve all the relevant stakeholders.

The e-CODEX e-Delivery transport infrastructure will provide a solution for cross-border communication for all LSPs based on the OASIS ebMS 3 standard. The ebMS 3 specification defines the technical interconnection and security standards and the message structure. A detailed description can be found in the ebXML Messaging Core Specifications[17], an overview is given in *D5.2 Reusable Assets* and summarized below.

For non-repudiation, delivery evidences as defined in the ETSI REM standard will be used.

3.1. Introduction

3.1.1. Vision

The vision for the convergence of e-Delivery platforms is based upon requirements assigned to the different levels of interoperability as defined in the European Interoperability Framework (EIF).

[17] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007

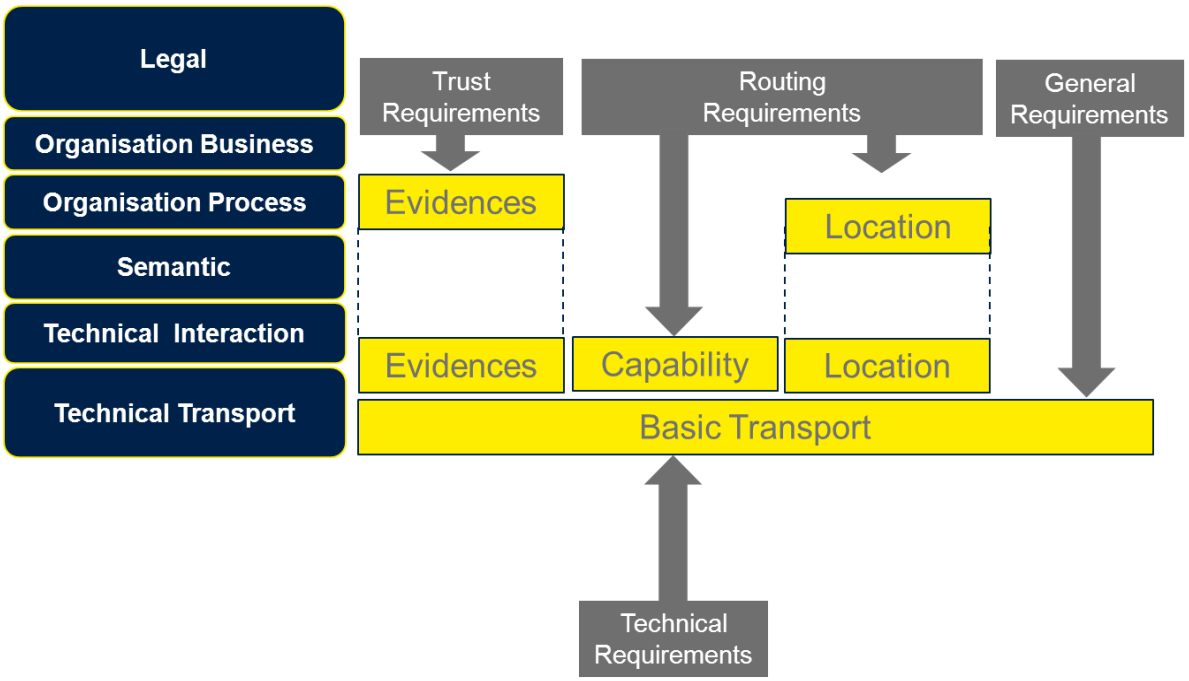


Figure 1 – Requirements for the proposed solution[18]

On the different levels, groups of requirements can be identified:

- Business Requirements
  - Payload agnostic
  - 4-Corner model
  - End-to-end non-repudiation of origin and receipt (for some applications)
- Technical Solution requirements
  - Confidentiality between gateways
  - Authenticity of sender gateway (and of sender end-point)
  - Integrity between gateways
  - Basic reliability in transmission between gateways or access points[19]
- Location Requirements (Routing)
  - Discovery of gateways and their physical addresses dynamically
  - Discovery of capabilities of gateways dynamically[20]

[18] Figure provided to the e-Delivery Task Force by PEPPOL.

[19] Although in the different LSPs the different terms Gateway and Access Point are used, the meaning is quite similar. For better readability further on the term Gateway will be used in this document.

[20] For the e-CODEX piloting phase it has been decided to use static routing with preconfigured gateways.

3.1.2. Basic Architecture

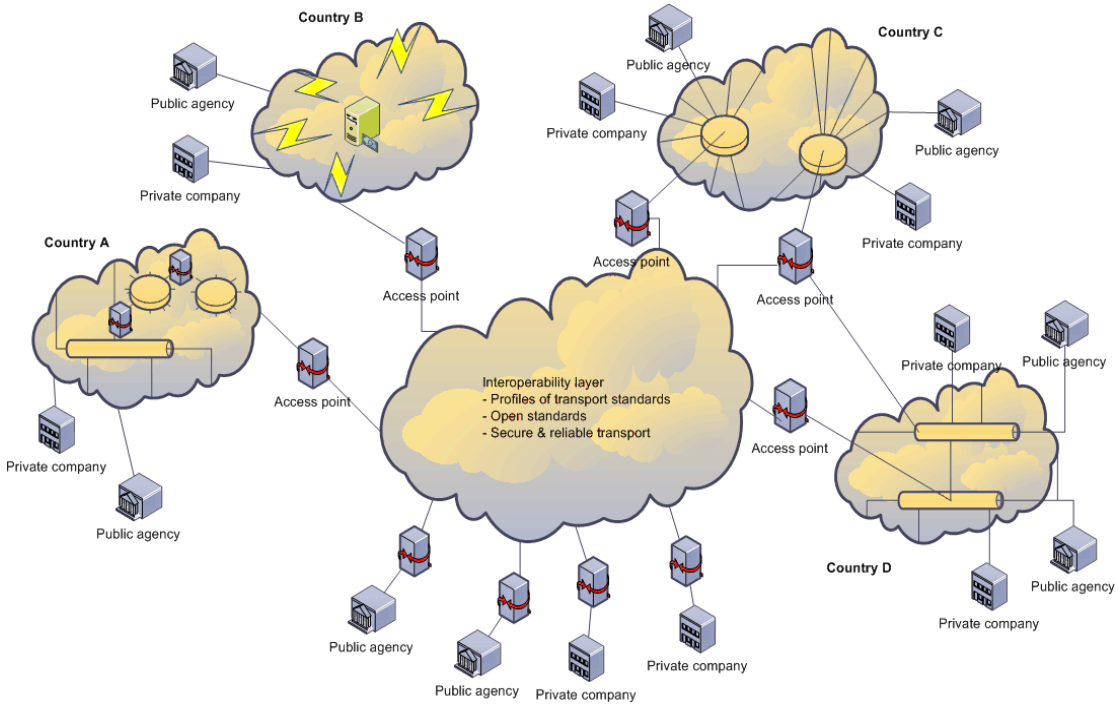


Figure 2 – High Level architecture model [21]

The goal for the European transport infrastructure is to provide cross border communication via gateways, similar to the existing LSP PEPPOL and SPOCS solutions. Behind the gateways there are the national domains (or other communities using their own transport solution[22]), which should remain unchanged.

The function of the gateway is the mapping to existing national domains or infrastructures and to guarantee secure and reliable messaging between the gateways. To guarantee such a messaging between the actual endpoints located within the national domains a so-called “circle of trust”, based on legal agreements, must be established. To be able to provide reliability and non-repudiation between endpoints the scenario for e-Delivery convergence also foresees standardized evidences (ETSI REM).

Concerning the routing and the discovery of partner gateways as well as their capabilities, the SML/SMP approach of PEPPOL seems to fit well and could be reused by aligning the SMP concept with the P-Modes as defined by the ebMS standard.

Furthermore, for trust establishment, i.e. for the retrieval of certificates associated with gateways, the SPOCS approach will be considered, which relies on the concept of TSL lists.

One difficulty arising from the goal to have a one-for-all e-Delivery solution is that there are different actors involved. For PEPPOL the communicating parties are public agencies and private companies, for SPOCS they are mostly private companies and public authorities, and for e-CODEX in most cases

[21] PEPPOL Deliverable 8.3 Transport Infrastructure - PEPPOL EIA (Enterprise Interoperability Architecture).

Please note that what in this document is referred to as “gateways” is in PEPPOL called “access points”.

[22] At least in the case of PEPPOL,some organisations may also connect directly, rather than via a gateway

citizens, courts and legal professionals (where in a court case some of the participants may not even be voluntarily involved). The consequence of this is that there are different requirements and possibilities for end user reliability and security. (E.g. it is very much different to set up legal community rules or contracts with private companies as opposed to judicial authorities). On the other hand, in the healthcare domain the epSOS legal agreements are signed between the relevant public authorities, hence more relevant to e-Justice. This could also influence the possibilities of end-to-end security and reliability based on evidences. In other words, end to end security and reliability can be realised with the e-Delivery platform but it is optional depending on the use cases and actors.

3.1.3. Scope

For e-CODEX the development of prototypes for the pilots, based on the defined convergence scenario has the highest priority in order to reach the goal to start the piloting phase by end of 2012. On the other hand parallel activities together with standardization bodies like OASIS will start to create a standard which is suitable for all LSPs including their specific requirements. Within this specification also all technical details (e.g. routing) will be worked out together with the partners. During the implementation phase of the e-CODEX pilots the results of the standardization process (if available) will be included to have a proof of concept for the European e-Delivery platform.

3.1.4. Involved parties

In the development of the e-Delivery convergence solution the following parties are involved:

- The European Commission: Overall responsible for the European e-Delivery solution and co-financing the LSPs.
- SPOCS: LSP project which has already realized a cross border e-Delivery infrastructure.
- PEPPOL: LSP project which has already realized a cross border e-Delivery infrastructure.
- E-CODEX: providing the first prototype of the upcoming e-Delivery solution in order to realize its pilots. This should include as much as possible of the evolving specifications from the standardization bodies OASIS and ETSI.
- OASIS: standardization body. Providing knowledge in the area of reliable and secure messaging as defined in the ebMS 3 standard.
- ETSI: standardization body. Providing knowledge and specifications in the area of evidences (ETSI REM)

It is expected that the list of involved parties will be expanded in the future to include additional actors also in view of the upcoming new LSP.

3.2. ebMS overview

ebMS (**e**bXML **M**essaging **S**ervice) is an open standard for B2B (Business to Business) messaging based on SOAP/Web services. It is developed and maintained by OASIS and has been adopted by ISO as ISO 15000. The ebMS specification “describes a communication-protocol neutral method for exchanging electronic business messages. It defines specific enveloping constructs supporting reliable, secure delivery of business information.”[23]

[23] OASIS Standard, OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features, October 2007, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/ebms_core-3.0-spec.pdf

In view of choosing ebMS as an existing, established B2B protocol over creating new specifications from scratch, it should be noted that other widely accepted protocols exist. The following table compares ebMS with other B2B standards (some of them considered outdated).

Feature	ebMS	AS1/AS2	EDI VAN	WSDL[24]	WS-I
Open public specification	Yes	Yes	Messaging	Yes	Yes
EDI payloads support	Yes	Yes	Yes	No	No
XML payload support	Yes	Yes	Yes	Yes	Yes
PDF and binary attachments support	Yes	Yes	Yes	Yes	Yes
Secure messaging with authentication	Yes	Yes	Partial	Yes	Yes
Reliable message delivery mechanism	Yes	Pending[25]	Partial	No	Yes
Legal receipt verification support	Yes	No	Partial	No	No
Built-in audit log and tracking	Yes	Yes	Yes	No	Yes
Business process workflow enabled	Yes	No	No	No	Partial
Role and action use support in envelope	Yes	No	No	No	No
Conformance suite for implementations	Yes	Yes	No	No	Yes
Digital certificates and encryption	Yes	Yes	Yes	Yes	Yes
XML encryption support	Yes	No	No	Yes	Yes
Open source implementations available	Yes	Yes	No	Yes	Yes
Uses web services infrastructure (Apache/SOAP)[26]	Yes	No	No	Yes	Yes
Asynchronous and Synchronous support	Yes	AS2 only	No	No	No[27]
SMTP delivery support	Yes	Yes	Yes	No	No
Message authorization	Yes	No	Yes	Limited	Yes

Table 5: The ebMS standard compared with other B2B standards[28]

The older protocols such as EDI VAN and AS1/AS2 have a rather large uptake industry. They were developed for use with proprietary networks (VAN), SMTP and FTP, whereas recent developments leverage technologies (security and reliability) that are nowadays available and standardized in the context of web services.

Even though such Web Service technologies as offered through the WS-* stack (WS-Security, WS-RM etc.) are in themselves standardised, they are also rather generic, and require profiling to really allow for interoperability. Where the LSPs PEPPOL and SPOCS chose to define their own profiling of this stack, tailored to the needs of these projects, ebMS is a more general approach that tries to apply the experiences from B2B communications to Web Services.

[24] Note that strictly speaking, WSDL is not a transport protocol, but a service description language. The quoted publication uses the term somewhat synonymous to „plain SOAP with WSDL for collaboration agreement“.

[25] Signed MDNs are considered to provide NRR like ebMS 2.0 receipts.

[26] The meaning of this is that the protocol can be implemented using off-the-shelf (open source or commercial) SOAP toolkits

[27] Asynchronous support is true for WS-I if using WS-Addressing

[28] Comparing ebXML messaging (ebMS) AS2 for EDI, EDI VAN and Web Service messaging, http://www.oasis-open.org/committees/download.php/23458/Comparing%20messaging%20systems%20for%20B2B.pdf

The IDABC Study on Business to business frameworks for IDA networks - September 2003[29] recommended ebMS for B2B exchanges already in 2003.

ebMS has been shown to be able to fulfil the basic transport requirements of all three LSPs PEPPOL, SPOCS and e-CODEX (see section 3.2.3 "ebMS and the Requirements of the Large Scale Pilots" below). ebMS 2 is today itself quite widely adopted by industry, whereas ebMS 3 is newer and better compliant with newer Web Service standards (see 3.2.2 "ebMS 3" below).

3.2.1. ebMS concepts

For the understanding of the following sections a basic knowledge of the terminology defined in the ebMS 3 core specification[30] is required. The following paragraphs summarize some key concepts:

1. **Messaging Service Handler (MSH), Producer, Consumer** – an *MSH* is an entity that is able to generate or process messages that conform to the ebMS specification, and to act as sender or receiver. A *Producer* is an entity (e.g. application) that interacts with a Sending MSH (i.e. an MSH in the Sending role) to initiate the sending of a user message. A *Consumer* is an entity that interacts with a Receiving MSH (i.e. an MSH in the Receiving role) to consume data from a received user message.
2. **Message, User Message, Signal Message** – a *Message* is a logical unit which consists of User Messages or Signal Messages or both. A *User Message* is a message that contains a User Message unit (an *eb:Messaging/eb:UserMessage* XML structure). A *Signal Message* is an ebMS message that contains a Signal Message unit (an *eb:Messaging/eb:SignalMessage* XML structure).

   In other words there exist two types of messages in the ebMS standard: the first type allows transmitting data interpreted by a Consumer and the second type allows transmitting data interpreted by an MSH as a signal (e.g. a pull signal).
3. **Message Exchange Pattern (MEP)**, One-Way/Push, One-Way/Pull, Two-Way/Sync MEP – a *MEP* is an agreement between sending and receiving MSHs. Some aspects of MEPs supported in the messaging layer include:
   - specifying the correlation between messages sent and received in the message header.
   - message binding to the underlying transfer-protocol.

   One-Way/Push, One-Way/Pull or Two-Way/Sync MEPs describe agreements between MSHs as stated above.

   The *One-Way/Push MEP* for example specifies a situation when a sending *MSH* which has agreed to use the *One-Way/Push MEP* sends a message to a receiving *MSH* which has agreed to use *One-Way/Push MEP* as well. In this case the message that would be sent is most likely a message carrying the user data. (It can also be a signal message e.g. error message.) After the reception the receiving *MSH* would send a non-user message (i.e. a signal message) to the sending *MSH* to confirm the reception. Different user messages do not have any reference to each other.

[29] http://ec.europa.eu/idabc/en/document/3516/5585.html

[30] OASIS Standard, OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features, October 2007, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/ebms_core-3.0-spec.pdf

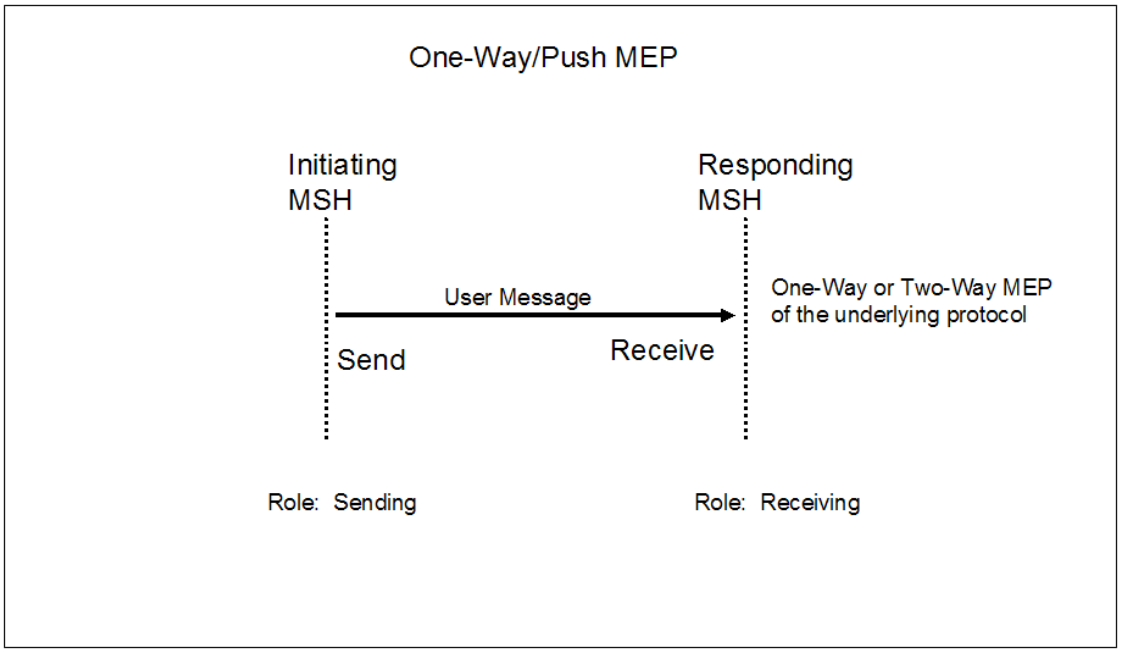


Figure 3 – One-Way/Pushl MEP[31]

The *One-Way/Pull MEP* specifies a situation when a receiving *MSH* which has agreed to use the *One-Way/Pull MEP* sends a signal message to a sending *MSH* which has also agreed to use the *One-Way/ Pull MEP.* After the reception of the signal message the sending *MSH* would send a user message to the receiving MSH. Different user messages do not have any reference to each other.

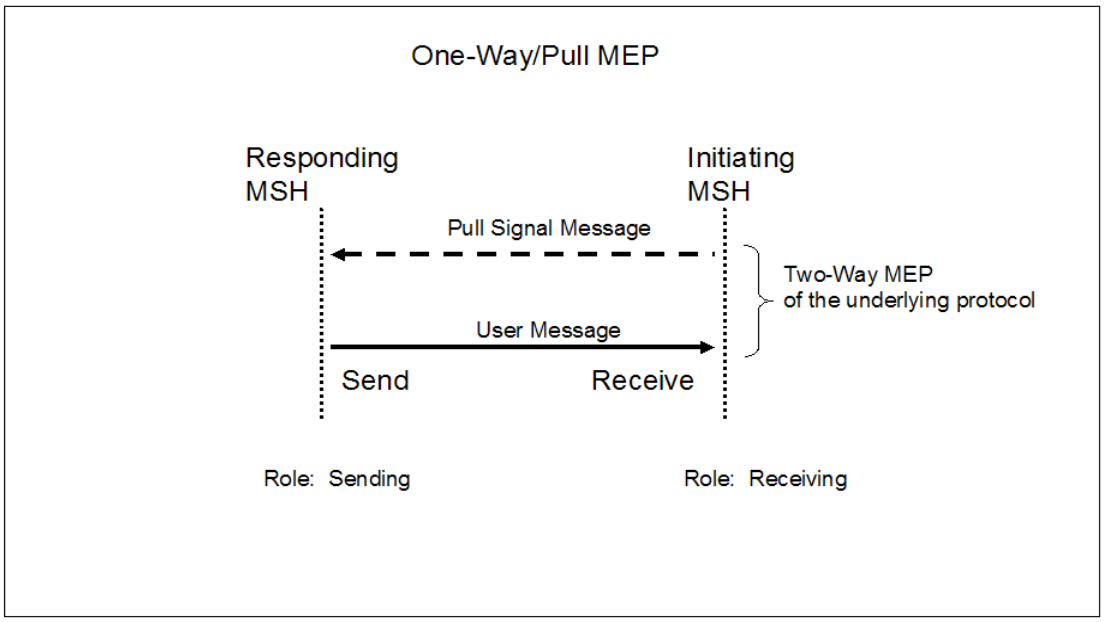


Figure 4 – One-Way/Pull MEP[32]

[31] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007

The *Two-Way/Sync MEP* specifies a situation when a *MSH* which has agreed to use the *Two-Way/Sync MEP* would send a user message to another *MSH* which has also agreed to use the *Two-Way/Sync MEP.* After the reception of the user message the receiving *MSH* would send a user message to the sending MSH. The second user message refers to the ID-field specified in the first user message sent.

Generally in the ebMS MEP context *pushing* means that the sender initiates the message exchange (for HTTP this implies that the sender is an HTTP client, and the receiver a server). *Pulling* in the ebMS MEP context means that the receiver initiates the message exchange (so the receiver would be an HTTP client and the sender an HTTP server).

The ebMS standard specifies more MEPs than listed in the previous section, like: *Two-Way/Push-and-Push*, *Two-Way/Push-and-Pull*, *Two-Way/Pull-and-Push*. MEPs like *Two-Way/Push-and-Pull* and *Two-Way/Pull-and-Push* support asynchronous exchanges between parties.

4. **Processing Mode (P-Mode)** - A P-Mode is the contextual information that governs the processing of a particular message (thus is basically a set of configuration parameters). The P-Mode associated with a message determines, among other things, which security and/or which reliability protocol and parameters, as well as which MEP is being used when sending a message.

   The technical representation of the P-Mode configuration is implementation-dependent. (For specific examples regarding the Open Source product Holodeck[33], which will be the basis of the e-CODEX implementation, see section 4.4.4.2 “P-Modes”).

P-Mode parameters to be used in e-CODEX communication are listed in Appendix III – P-Mode Configuration.

3.2.2. ebMS 3

The previous ebMS version, numbered 2.0, is quite widely adopted in the market. The current version 3.0 supports and extends the core functionality of version 2.0. It improves version 2.0 on several aspects such as message pulling, non-repudiation of receipt, and full compliance with Web service protocols such as WS-ReliableMessaging and WS-Security. Version 3.0 is also compliant with related WS-I Basic Profiles and more compliant with the SOAP processing model, decoupling the message body from the message delivering mechanics.

One significant change in the version 3.0 regards the message body: The SOAP body has been freed for business payload. The ebMS header is just a SOAP extension among others.

For e-CODEX, ebMS 3 was chosen, because in a new and innovative project it makes sense to build upon the newest version of a specification. From a technical point of view, a major reason for this choice is its better compliance with the WS-* protocol stack.

---

[32] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007

[33] http://holodeck-b2b.sourceforge.net/docs/index.html

3.2.2.1. AS4 Profile

The AS4 profile aims at leveraging some of the benefits of the rather successful EDIINT B2B protocols AS1, AS2[34] (and AS3) by defining a subset of functionality "to trim down ebMS 3.0 into a more simplified and AS2-like specification for Web Services B2B messaging".[35]

The Holodeck Open Source solution supports this profile (though with some limitations, e.g. regarding compression and receipts), and any additional profiling in the e-CODEX specifications should be as close to the AS4 profile as possible. The impact of said limitations is expected to be small; however some adaptations may be necessary for the e-CODEX implementation.

3.2.3. ebMS and the Requirements of the Large Scale Pilots

Among the e-CODEX requirements identified in D5.1[36], D5.2[37] and also in the combined deliverable D3.2 & D7.2[38], the ones that concern the transport infrastructure have to be fulfilled by the selected protocol – notably Transport Signatures, Original Sender Authentication, Reliable messaging mechanisms, Encryption and Time Stamping.

The "sister LSPs" PEPPOL and SPOCS have similar requirements, and have created their own transport protocols to fulfil them. PEPPOL uses a communication standard called Secure Trusted Asynchronous Reliable Transport (START)[39]. SPOCS uses a communication standard named REM-MD SOAP Binding Profile[40].

START, the REM-MD SOAP Binding Profile and ebMS are all based on the SOAP standard.

A solution based on ebMS can fulfil the requirements of all three LSPs, as explained in the document "Scenario for the Convergence of LSP e-Delivery solutions"[41]. The following subsections give a summary. The scenario described in that document has been accepted as a basis for the transport infrastructure by the European e-Delivery Task Force.

---

[34] Particularly AS2, being an http-based protocol which does itself not specify the payload (thus allowing for the use of conventional EDI formats like EDIFACT or ANSI X12) has been widely adopted by companies which previously used conventional EDI – among them very large companies such as Walmart.

[35] *AS4 Profile of ebMS 3.0 Version 1.0*. 25 May 2011. OASIS Committee Specification Draft 04 / Public Review Draft 03. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/csprd03/AS4-profile-v1.0-csprd03.html.

[36] e-CODEX Deliverable D5.1 "Requirements"

[37] e-CODEX Deliverable D5.2 "Reusable Assets"

[38] e-CODEX Deliverable D3.2 & D7.2 "Requirements Finalisation and Description of Test Scenarios"

[39] START Profile,

http://www.peppol.eu/Archive/final-public-documents-and-presentations/publications/peppol-v-0-95-specifications-for-infrastructure/start-profile/at_download/file

[40] REM-MD SOAP Binding Profile,

http://www.arrow-net.eu/sites/default/files/D4.1_State_of_the_art_guidelines_standards.pdf

[41] "Scenario for the Convergence of LSP e-Delivery solutions", Appendix to e-CODEX Deliverable D5.2 "Reusable Assets"

- **Transport Signatures**

ebMS supports signing of messages as defined in Web Services Security 1.0[42] and 1.1[43] (based on the XML Signature standard[44]) and the X.509 Certificate Token Profile.[45]

- **Encryption**

As for signatures, encryption of messages is supported according to Web Services Security 1.0 and 1.1 (based on the XML Encryption standard[46]), and the Web Services Security X.509 Certificate Token Profile is required for ebMS implementations.

- **Time Stamping**

All ebMS message types contain an element named *MessageInfo*. This element contains a required child-element called *Timestamp*. The value of this element represents the date at which the message header was created. For the receiving MS, a timestamp will be available in the delivery evidence. This timestamp from the receiving MS is the one considered to be valid for legal periods.

- **Reliable Messaging Mechanisms**

The ebMS standard specifies the Reliable Messaging Module abstract which can be implemented according to the WS-Reliability 1.1[47] or WS-ReliableMessaging 1.1[48] standards.

- **Original Sender Authentication**[49]

Even though the protocols defined by all three LSPs are intended for the use between gateways / access points only (i.e., not end-to-end), both START and the REM-MD SOAP Binding Profile specify elements to convey the identity of the original sender of a message. In some cases (particularly where documents don't carry a personal digital signature from the sender), this requirement exists in e-CODEX as well.

It is not fulfilled by ebMS out-of-the box (as ebMS is not inherently a multi-corner protocol), but can be fulfilled through profiling (see section 3.3.3.4.2 "End Entity Authentication (Original Sender)").

---

[42] Web Services Security: SOAP Message Security 1.0 (WS-Security 2004), OASIS Standard 200401, http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf, March 2004

[43] Web Services Security: SOAP Message Security 1.1, http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-os-SOAPMessageSecurity.pdf, 1 February 2006

[44] XML Signature Syntax and Processing, http://www.w3.org/TR/xmldsig-core/, 10 June 2008

[45] Web Services Security X.509 Certificate Token Profile, http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0.pdf, March 2004

[46] XML Encryption Syntax and Processing, http://www.w3.org/TR/xmlenc-core, 10 December 2002

[47] WS-Reliability 1.1, http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/wsrm-ws_reliability-1.1-spec-os.pdf

OASIS Standard, 15 November 2004

[48] Web Services Reliable Messaging (WS- ReliableMessaging) Version 1.1, http://docs.oasis-open.org/ws-rx/wsrm/200702/wsrm-1.1-spec-os-01.pdf, OASIS Standard 14 June 2007

[49] B2B Protocols for Multi-Corner Messaging.odt-1.odt, http://www.oasis-open.org/committees/document.php?document_id=43769&wg_abbrev=bdx

3.3. Message specifications

3.3.1. Packaging

The ebXML message structure supports the delivery of messages, within a MIME multipart to allow payloads or attachments to be included, but also packaged as a plain message. The ebMS message is packaged as a SOAP 1.1 or 1.2 message.

The ebMS 3.0 Core Specification defines the structure of the Message Package[50]:

"There are two logical sections within the Message Package:

- The first section is the ebMS Header (i.e. the eb:Messaging SOAP header block), itself contained in the SOAP Header.
- The second section is the ebMS Payload, which itself comprises two sections: (a) the SOAP Body element within the SOAP Envelope and in case of MIME packaging, (b) zero or more additional MIME parts containing additional application-level payloads. The SOAP Body and MIME parts are also referred to as ebMS Payload Containers. The SOAP Body is the only payload container that requires XML-structured content, though non-XML content may be included within an appropriately typed (binary or otherwise) element inside the Body."

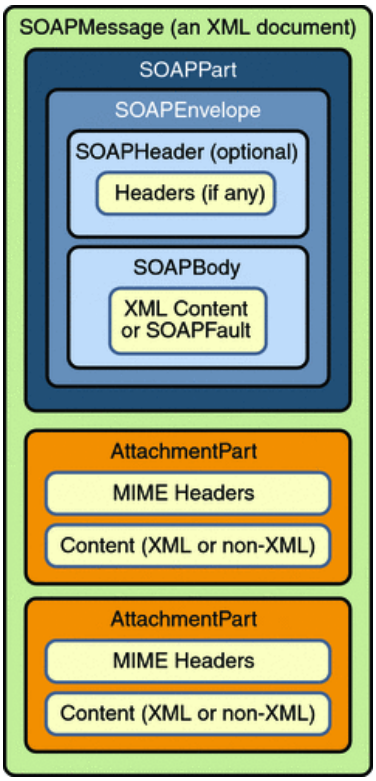


Figure 5 – MIME Message Transport[51]

[50] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007

As explained above, the ebXML message can be sent as a MIME multipart message. The MIME headers of the Message Package have to conform to the SOAP Messages with Attachments W3C Note[52].

"Because implementations MUST support non-multipart messages, an ebMS Message with no payload may be sent either as a plain SOAP message or as a [SOAPATTACH] multipart message with only one body part (the SOAP Envelope)."[53] (In other words, when no additional MIME parts are needed for extra payloads, it is not necessary to use multipart messages.)

The basic structure of an ebXML non-multipart message and the containing elements of the SOAP envelope part:

- SOAP Header Element - Contains an ebXML SOAP Extension block (*"eb:Messaging"*). Extends the SOAP Header with ebXML header information, e.g. MessageType and MessageInfo elements. Other Header elements like WS-Security and Reliability and Reliable Messaging blocks may be present.
- SOAP Body Element - "Unlike ebMS v2, ebXML Messaging 3.0 does not define or make use of any elements within the SOAP Body, which is wholly reserved for user-specified payload data."[54]

In case of sending a multipart message, the SOAP message including the ebMS header is the root body part of the message (see Figure 6). In addition to the SOAP Body and the included user-specified payload, other Payload Containers may be present in the message package. There can also be additional MIME attachments that are not payload containers and which are outside the scope of the MSH processing (see Figure 5).

3.3.2. Message Structure

A detailed description of the XML structure of an ebXML Message can be found in the ebMS Core Specification.[55]

The *eb:Messaging* structure inside the SOAP Header can carry either a *UserMessage* or a *SignalMessage* structure.

---

[51] http://docs.oracle.com/cd/E19879-01/819-3669/bnbhf/index.html

[52] J. Barton, et al, SOAP Messages with Attachments, 2000. http://www.w3.org/TR/SOAP-attachments

[53] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007 Page 36

[54] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007 Page 38

[55] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007 Chapter 5

3.3.2.1. UserMessage

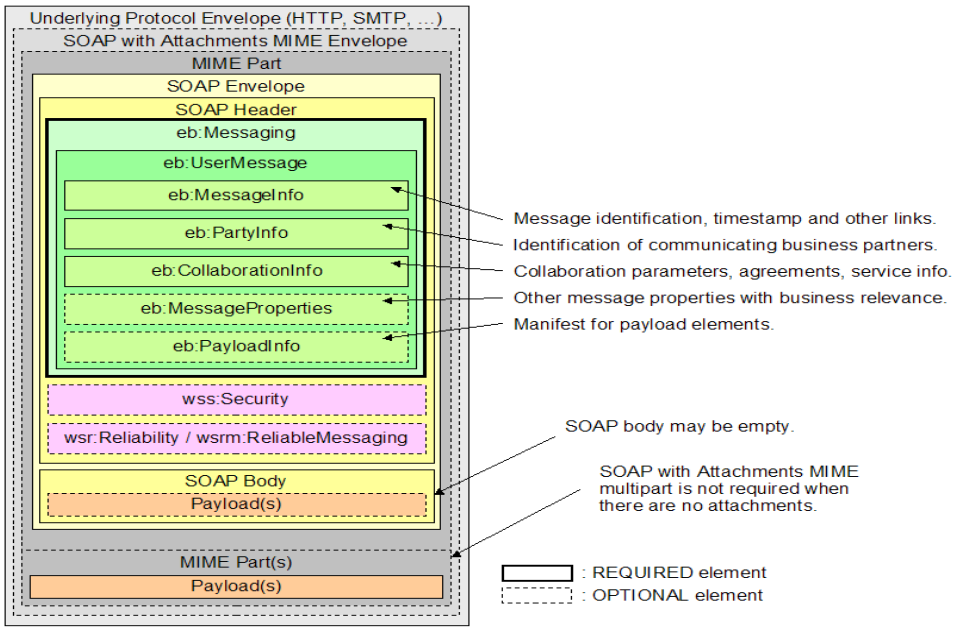


Figure 6 – SOAP Message with UserMessage [56]

The above figure shows the general message structure of an ebMS *UserMessage*. The different parts of the *UserMessage* element are shown in the picture below.

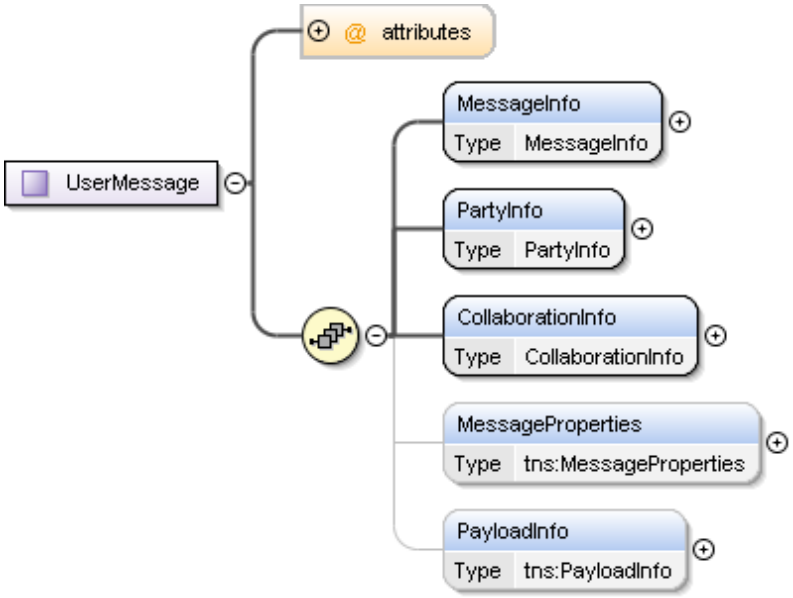


Figure 7 – UserMessage Structure

[56] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007 Page 34

3.3.2.2. SignalMessage

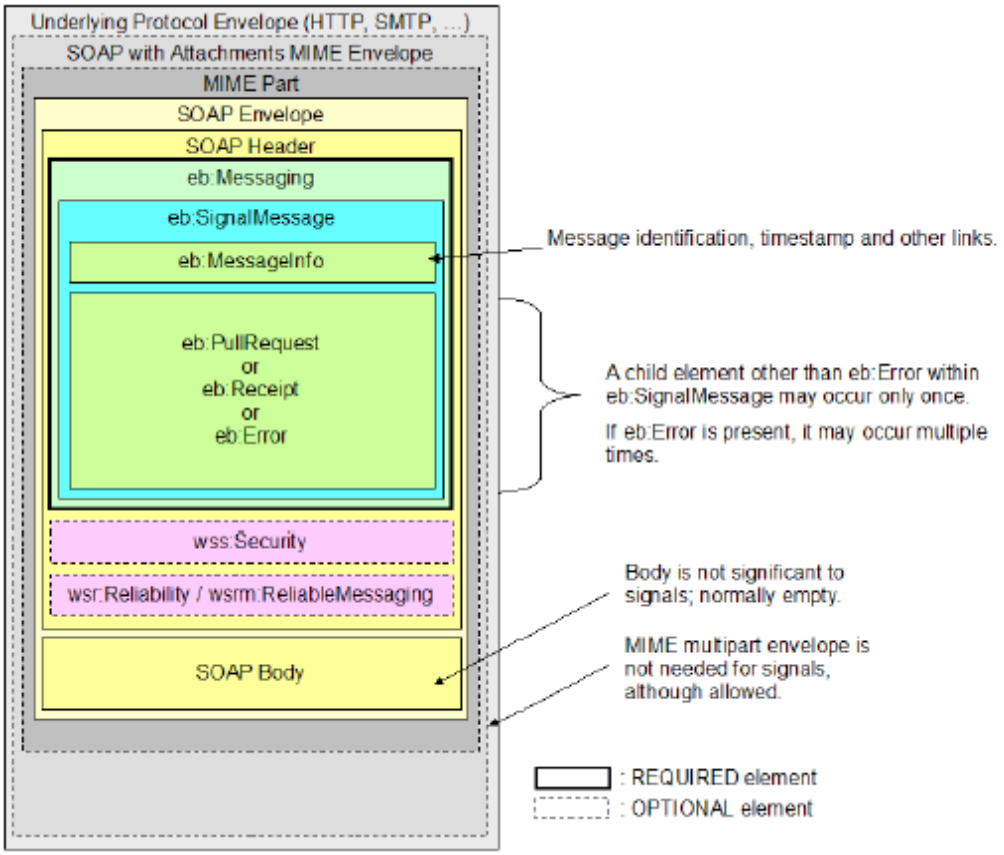


Figure 8 – SOAP Message with SignalMessage[57]

The above figure shows the general message structure of an ebMS *SignalMessage*. The different parts of the *SignalMessage* element are shown in the picture below.

[57] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features Committee Specification 02, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/cs02/ebms_core-3.0-spec-cs-02.html, 12 July 2007 Page 35

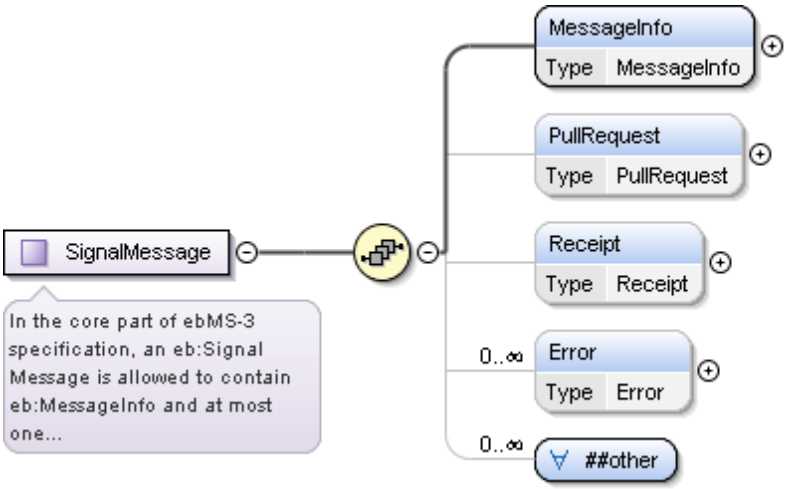


Figure 9 – SignalMessage Structure

3.3.3. Usage of ebMS in a Four-Corner-Model

In the European transport infrastructure different domains using proprietary standards are connected via gateways, and a common protocol is used only between those gateways.

In this type of architecture (also known as a *Four-Corner-Model*[58]) some additional agreements beyond what is already specified in the ebMS standard are required, as described in the following sections.

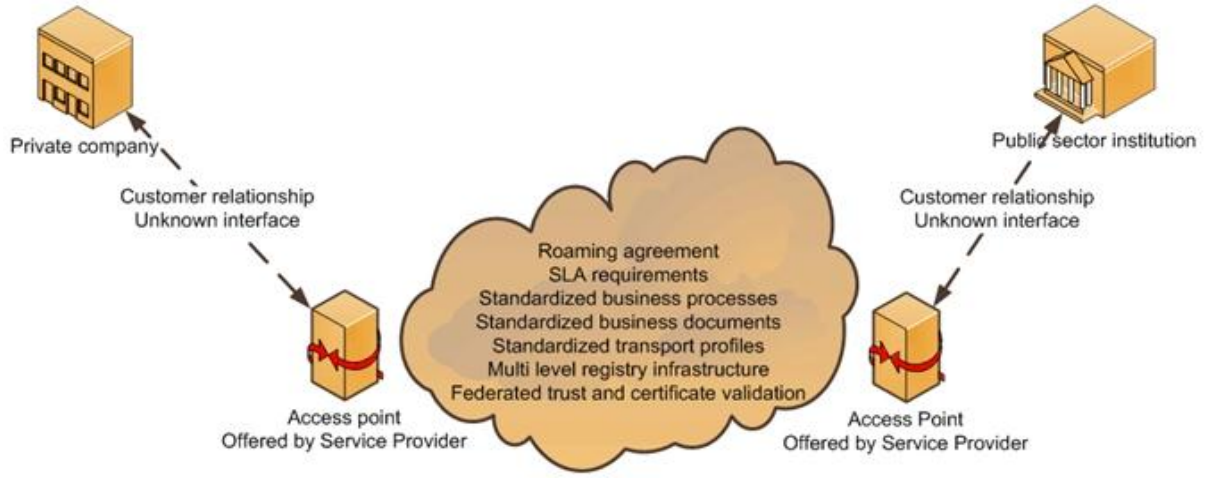


Figure 10 – Four-Corner-Model[59]

[58] It is called a „Four-Corner-Model“ because a message passes four „corners“ on its way: the original sender, the sending gateway/access point, the receiving gateway/access point and the final recipient.

[59] PEPPOL Transport Architecture, http://www.peppol.eu/events/peppol-conferences/conference-pan-european-eprocurement-with-peppol/Conference%20Presentations/20091022-peppol-conference-3-architecture-overview-fremantle/view

Note that these sections also cover profiling aspects. Relevant P-Mode settings are listed in Appendix III – P-Mode Configuration.

3.3.3.1. Addressing

In the usual scenarios where the ebMS protocol is used between end entities, it is obvious that the *From* and *To* fields in the *UserMessage* will be used for the sender and receiver respectively. However in a four-corner-model (i.e., an architecture that uses gateways to connect existing proprietary infrastructures) the senders / recipients of ebMS messages are the gateways, not the end entities.

In order to keep the option open to use out-of the box messaging products, *From/PartyId* and *To/PartyId* shall therefore in this case denote the addresses of gateways.

The addresses of end entities will then be transmitted as ebMS properties.

An example is given in "B2B Protocols for Multi-Corner Message Exchange"[60]

```
<eb3:UserMessage>
  <eb3:MessageInfo>
    <!-- Omitted -->
  </eb3:MessageInfo>
  <eb3:PartyInfo>
    <eb3:From>
     <eb3:PartyId>http://edelivery.de/gateway</eb3:PartyId>
     <eb3:Role>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/initiator</eb3:Role>
    </eb3:From>
    <eb3:To>
     <eb3:PartyId>http://edelivery.nl/gateway</eb3:PartyId>
     <eb3:Role>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/responder</eb3:Role>
    </eb3:To>
  </eb3:PartyInfo>
  <eb3:CollaborationInfo>
    <eb3:Service>http://docs.oasis-open.org/ebxml-msg/as4/200902/service</eb3:Service>
      <eb3:Action>http://docs.oasis-open.org/ebxml-msg/as4/200902/action</eb3:Action>
      <eb3:ConversationId>ecae53d4-7473-45a6-ad70-61970dd7c4b0</eb3:ConversationId>
  </eb3:CollaborationInfo>
    <eb3:MessageProperties>
      <eb3:Property
        name="FromPartyId">123456789</eb3:Property>
      <eb3:Property
        name="FromPartyIdType">urn:oasis:names:tc:ebcore:partyid-type:iso6523:0002</eb3:Property>
      <eb3:Property
        name="ToPartyId">192837465</eb3:Property>
      <eb3:Property
        name="ToPartyIdType">urn:oasis:names:tc:ebcore:partyid-type:iso6523:0106</eb3:Property>
      <eb3:Property name="Service">urn:www.cenbii.eu:profile:BII06:ver1.0</eb3:Property>
      <eb3:Property name="Action">RejectOrder</eb3:Property>
    </eb3:MessageProperties>
  <eb3:PayloadInfo>
     <eb3:PartInfo />
  </eb3:PayloadInfo>
</eb3:UserMessage>
```

To denote the different types of national transport infrastructures, values for the address type will have to be defined within e-CODEX, e.g.

```
<eb:Property name="FromPartyId">govello-1234567890123-456789012</eb:Property>
<eb:Property name="FromPartyIdType">urn:oasis:names:tc:ebcore:partyid-type:unregistered:egvp</eb:Property>
```

[60] http://www.oasis-open.org/committees/document.php?document_id=43769&wg_abbrev=bdx

3.3.3.2. Reliability

For the e-CODEX piloting, reliability will be based on WS-Reliability 1.1. Implementations must support this in order to connect to other e-CODEX gateways.

For the European transport infrastructure, it is yet to be determined if this needs to be changed to (or supplemented by) the newer WS-ReliableMessaging standard, which is today supported by a larger number of SOAP toolkits (for Apache Axis through the Sandesha[61] library). In particular this may become necessary to support application servers other than Tomcat (e.g. Oracle Weblogic). The use of AS4 receipts might also be considered.

3.3.3.3. Non-repudiation between gateways

In addition to end-to-end non-repudiation through REM evidences (see section 3.5 "Evidences (Workflows)"), for domains that do not use business-level evidences an option is to use AS4 signals between gateways (not required by e-CODEX).

3.3.3.4. Trust establishment / Security

3.3.3.4.1. Gateway Authentication

For the mutual authentication of gateways according to WS-Security certificates (X509 v3) will be used.

In user groups with a small number of participants it is common to exchange these certificates between communication partners in advance. However with a large number of gateways (as for example in PEPPOL) this will not be feasible. Possible solutions are Trusted Service Lists (TSLs) as employed by SPOCS or a dedicated PKI (Public Key Infrastructure) as used by PEPPOL. For the European Transport Infrastructure this will have to be further examined and possibly a configurable solution needs to be built.

For the e-CODEX pilots, having a limited number of gateways, the out-of-the box available mechanisms for import of certificates in Holodeck will be used[62]. Member states choose their own certificates and exchange them out-of-band, i.e. by means outside e-CODEX infrastructure.

3.3.3.4.2. End Entity Authentication (Original Sender)

Before submitting messages users will usually have to authenticate themselves to either a national transport solution or to the European e-Justice Portal. The type and level of authentication required varies in each case. For the European e-Justice Portal it is anticipated that this authentication is done via users' STORK IDs, and also for other services such as national transport infrastructures it is to be expected that over time standardized authentication mechanisms will be used and that proof of this authentication should be submitted to the receiving gateway as part of the message.

Both PEPPOL and SPOCS have therefore devised means to transmit a SAML[63] token for end entity authentication in their messaging.

---

[61] http://ws.apache.org/sandesha/

[62] Note that the SPOCS SSL implementation is already available for reuse, so combining this with the e-CODEX solution will likely not be too difficult.

[63] Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 March 2005, http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

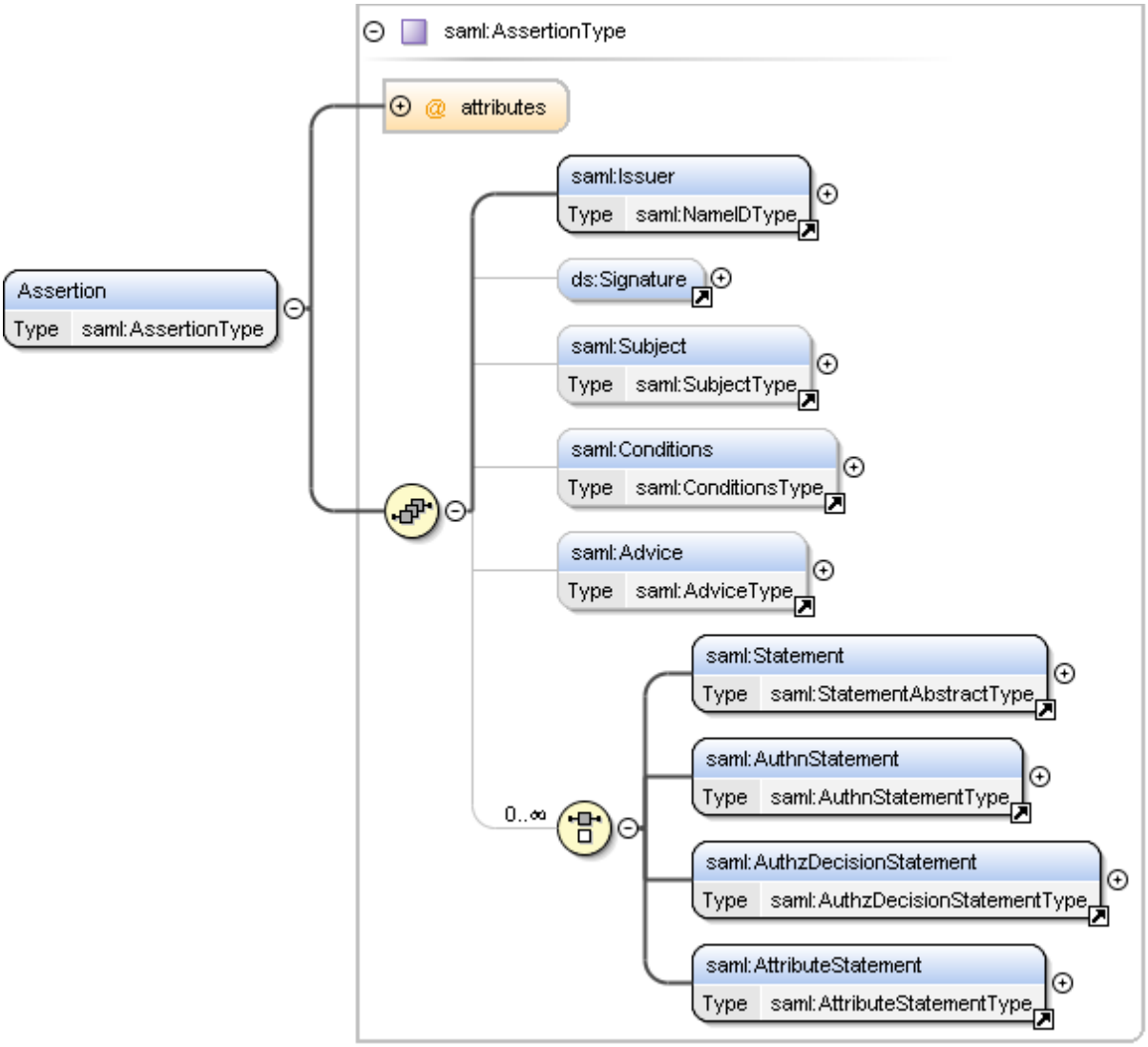


Figure 11 – SAML Assertion

For the European Transport Infrastructure, this SAML token shall be added as a separate message part.

For which business exchanges this type of special payload is required, can be configured in the *BusinessInfo.PayloadProfile* setting of the P-Mode. (That is, communication partners can in this setting agree upon whether they need the SAML token or not.)

The *PayloadInfo* element of the ebMS *UserMessage* structure will then contain a corresponding *PartInfo* structure referencing this additional message part, and the messages parts will be required to appear in a given order (see section 3.3.3.5 “Summary: Message parts” below).

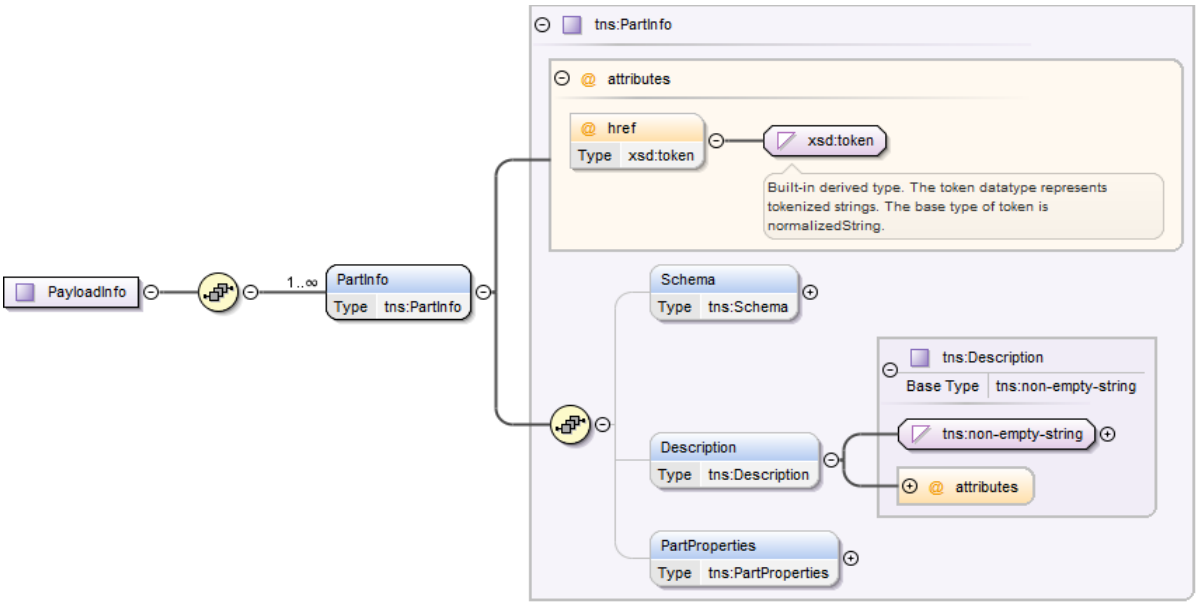


Figure 12 – PartInfo in the UserMessage

Example:

```
<eb3:UserMessage>
  <eb3:MessageInfo>
    <!-- Omitted -->
  </eb3:MessageInfo>
  <eb3:PartyInfo>
    <eb3:From>
      <!-- Omitted -->
    </eb3:From>
    <eb3:To>
      <!-- Omitted -->
    </eb3:To>
  </eb3:PartyInfo>
  <eb3:CollaborationInfo>
    <!-- Omitted -->
  </eb3:CollaborationInfo>
  <eb3:PayloadInfo>
     <eb3:PartInfo>
         <eb3:Description>XML business document in SOAP body</eb3:Description>
     </eb3:PartInfo>
     <eb3:PartInfo href="cid:8563d9f0-86e2-11e1-b0c4-0800200c9a66@edelivery.de">
         <eb3:Description>SAML Token</eb3:Description>
     </eb3:PartInfo>
  </eb3:PayloadInfo>
  <!– Other payloads -->
</eb3:UserMessage>
```

A profile for the SAML token is given in Appendix I – SAML Token Profile.

Example:

```
<saml2:Assertion
    Version="2.0"
    ID="12345678901234567890123456789012345678901234567"
    IssueInstant="2012-03-08T14:22:00">
    <saml2:Issuer>http://edelivery.de/gateway</saml2:Issuer>
    <ds:Signature><!-- Omitted --></ds:Signature>
```

```
    <saml2:Subject>
        <saml2:NameID Format="type">SomeUser</saml2:NameID>
        <saml2:SubjectConfirmation><!-- Omitted --></saml2:SubjectConfirmation>
    </saml2:Subject>
    <saml2:Conditions><!-- Omitted --></saml2:Conditions>
    <saml2:AuthnStatement AuthnInstant="2012-03-08T14:22:00">
        <saml2:AuthnContext>
            <saml2:AuthnContextClassRef>
                <!-- Omitted -->
            </saml2:AuthnContextClassRef>
        </saml2:AuthnContext>
    </saml2:AuthnStatement>
    <saml2:AttributeStatement><!-- Omitted --></saml2:AttributeStatement>
</saml2:Assertion>
```

The *NameID* element in the *Subject* must contain the end entity identifier in the same format as given for the original sender in the end entity adressing field (see section 3.3.3.1 “Addressing” above).

3.3.3.5. Summary: Message parts

The following table summarizes the different message parts for a business message, their order and their meaning.

Message Part No	Contents
1	Payload (XML business document) in SOAP body
2	SAML token for end entity authentication
..	Additional Payloads (attachments)

Table 6: Message parts

In the *PMode[1].BusinessInfo.PayloadProfile[]* setting communication partners can agree which message parts must be present and which are optional. For the European Transport Infrastructure the use of the SAML token will be configured according to bilateral agreements.

For all business messages (i.e. messages which are not evidences) a *PayloadInfo* element MUST be provided in the ebMS Header.

For all message parts listed above, if present, *PartInfo* elements MUST appear in the given order.

Note that (with the exception of the first part, which is the SOAP body) *PartInfo* XML structures are linked to the corresponding MIME parts via the MIME content-ID.

Example:

```
Content-Type: Multipart/Related; boundary=MIME_boundary;
type=application/soap+xml;
start="<0e6dedc0-8734-11e1-b0c4-0800200c9a66@edelivery.de>"
--MIME_boundary
Content-Type: application/soap+xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <0e6dedc0-8734-11e1-b0c4-0800200c9a66@edelivery.de>
<?xml version='1.0' ?>
<S12:Envelope xmlns:S12="http://www.w3.org/2003/05/soap-envelope"
xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/">
<S12:Header>
  <eb:Messaging S12:mustUnderstand="true">
```

```
    <!-- ... -->
    <eb:PayloadInfo>
      <eb:PartInfo>
        <eb:Description>Actual business document payload</eb:Description>
      </eb:PartInfo>
      <eb:PartInfo href="cid:2baedf50-8736-11e1-b0c4-0800200c9a66">
        <eb:Description>SAML Token</eb:Description>
      </eb:PartInfo>
      <eb:PartInfo href="cid:f4f8b7f0-8736-11e1-b0c4-0800200c9a66">
        <eb:Description>Additional attachment</eb:Description>
      </eb:PartInfo>
    </eb:PayloadInfo>
    </eb:Messaging>
  </S12:Header>
<S12:Body>
  <eCodex:EPO>
  <!-- ... -->
  </eCodex:EPO>
</S12:Body>
</S12:Envelope>
--MIME_boundary
Content-Type: application/samlassertion+xml
Content-Transfer-Encoding: 8bit
Content-ID: <cid:2baedf50-8736-11e1-b0c4-0800200c9a66@edelivery.de>
<?xml version='1.0' ?>
<saml2:Assertion
  Version="2.0"
  ID="12345678901234567890123456789012345678901234567"
  IssueInstant="2012-03-08T14:22:00">
  <!-- ... -->
</saml2:Assertion>
--MIME_boundary
Content-Type: image/tiff
Content-Transfer-Encoding: binary
Content-ID: <cid:f4f8b7f0-8736-11e1-b0c4-0800200c9a66@edelivery.de>
...scanned document binary...
--MIME_boundary--
```

Remark: It is still under discussion in e-CODEX WP4, if the so-called “Trust-OK token” (which gives information about signature verification and/or user authentication in human-readable form) should be packaged together with the signed documents in a container. In this case the whole container would be one attachment. Otherwise it might be considered to treat the “Trust-OK token” similar to the SAML token.

3.3.3.6. Message Exchange Patterns

In general, the simplest way to send a message is by making an HTTP request, so implementations are expected to support (One-Way) push mode at minimum. If both gateways support being sender and receiver in this pattern, the Two-Way/Push-and-Push pattern technically poses no extra challenges. On the logical level it introduces the possibility to have messages related to each other, that is, to implement question- and-answer scenarios.

Pulling patterns are primarily useful for communication partners with limited capabilities (for example mobile devices which cannot act as HTTP servers). Therefore Member states might choose

to use this features to implement lightweight clients as backend interfaces, however for interoperability between gateways it will not be required[64].

e-CODEX gateways will support a least the One-Way/Push and Two-Way/Push-and-Push MEPS.

3.3.4. Open issues

The following topics need further elaboration, and will have to be discussed in the European e-Delivery Task Force:

- Reliability
- Addressing of End Entities, Address format
- Dynamic Discovery (see also the following section)
- Trust establishment
- Evidences (end-to-end, further profiling)

3.4. Discovery, addressing and endpoint capabilities

To send an electronic message from sender to receiver address resolution or routing is needed. In order to enable routing of documents received from the sender to the correct recipient the messages will have to be routed using the already existing e-Delivery solutions of the Member States.

[64] Note that the AS4 profile requires support for the One-Way/Pull MEP as well. Even so, obviously any AS4-compliant implementation would still satisfy the e-CODEX requirements.

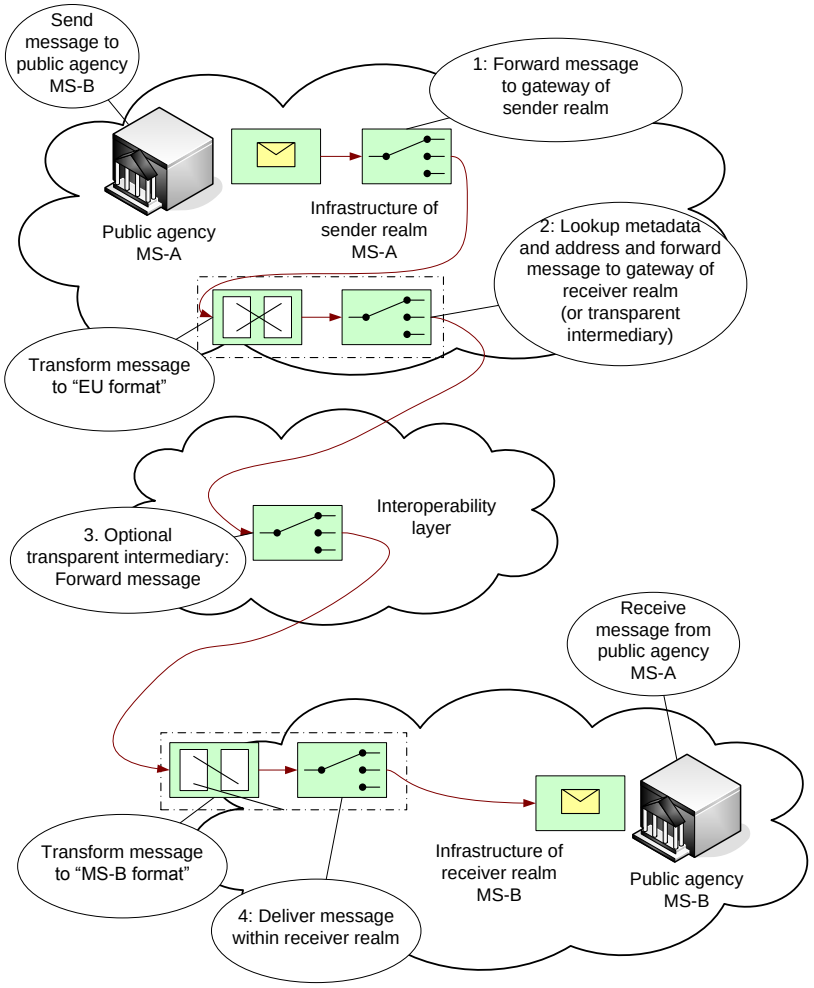


Figure 13 – Message routing

In this case the list of participants is numerous. For example in Romania alone the number of courts is well above 400. So we can safely assume that a sending party has no knowledge of all existing recipients in all the 27 Member States let alone knows the details for routing between the different infrastructures that are needed for cross-border messaging. Therefore it is preferable to have a dynamic system to discover the corresponding gateways and whether they are capable of receiving the message including the metadata for document type. Part of the solution to discover partner gateways and their capabilities can be found in the service discovery process of PEPPOL.

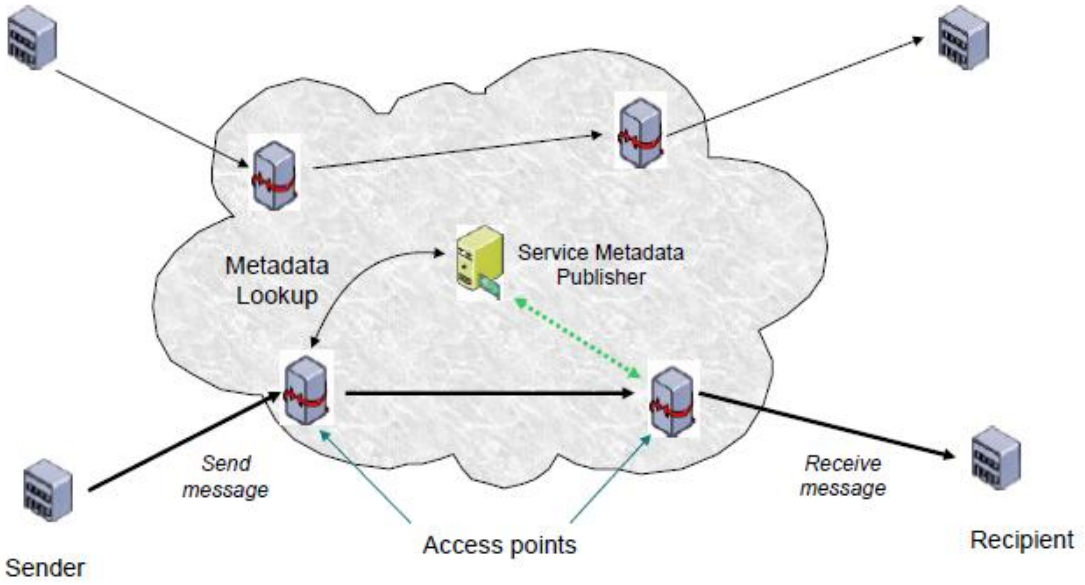


Figure 14 – Discovery infrastructure

The gateways communicate in a peer-to-peer like model across the internet and discover partner gateways and their routing addresses via the SML/SMP infrastructure (as described in a later chapter).

3.4.1. Service Metadata Locator (SML)

For a sender the Discovery process begins with the establishment of the location of the Service Metadata relating to the particular gateway to which the sender wants to transmit a message. Each participant can be a sender and a recipient and is identified by Participant ID. A Participant ID is a unique identifier of a particular participant.

The concept is explained in PEPPOL Specification "PEPPOL Transport Infrastructure Service Metadata Locator (SML)"[65]:

"The Service Metadata Locator service specification is based on the use of DNS (Domain Name System) lookups to find the address of the Service Metadata for a given participant ID [DNS-1034[66]] [DNS-1035[67]]. This approach has the advantage that it does not need a single central server to run the Discovery interface, with its associated single point of failure."

The following diagram represents the lookup flow.

[65] https://joinup.ec.europa.eu/svn/peppol/PEPPOL_EIA/1-ICT_Architecture/1-ICT-Transport_Infrastructure/13-ICT-Models/ICT-Transport-SML_Service_Specification-101.pdf

[66] http://www.ietf.org/rfc/rfc1034.txt

[67] http://www.ietf.org/rfc/rfc1035.txt

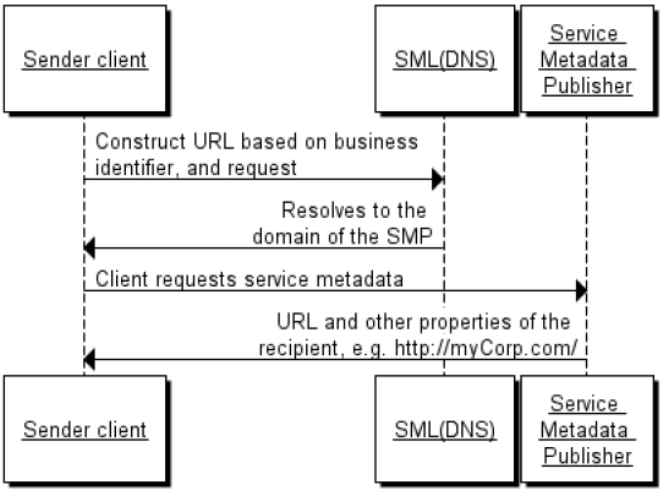


Figure 15 – Endpoint lookup with Service Metadata

- Each participant can be a sender and a recipient and is identified by Participant ID. A Participant ID is a unique identifier of a particular participant.
- The SML service itself plays the role of providing controlled access to the creation and update of entries in the DNS. The sender constructs the address for the service metadata for a given recipient participant identifier using a standard format, as follows:
  http://<hash over recipientID>.<schemeID>.<SMLdomain>/<recipientID>/services/<documentType>
- Each participant ID gets a unique URL for its Metadata
  http://<recipientID>.<schemeID>.<SML domain>
- Via DNS lookup this URL resolves to the address of Metadata Publisher server
- The Metadata Publisher hosts Metadata for each participant ID at a predefined URL
  /<recipientID>/services/<documentType
- The sender uses this URL in a HTTP GET operation which returns the metadata relating to that recipient and the specific document type (for details, see the Service Metadata Publishing specification [BDEN-SMP][68]).

The sender can obtain the information necessary to transmit a message containing that document type to that recipient from the returned metadata. This sequence is shown in the following figure:

[68] https://joinup.ec.europa.eu/svn/peppol/PEPPOL_EIA/1-ICT_Architecture/1-ICT-Transport_Infrastructure/13-ICT-Models/ICT-Transport-SMP_Service_Specification-101.pdf

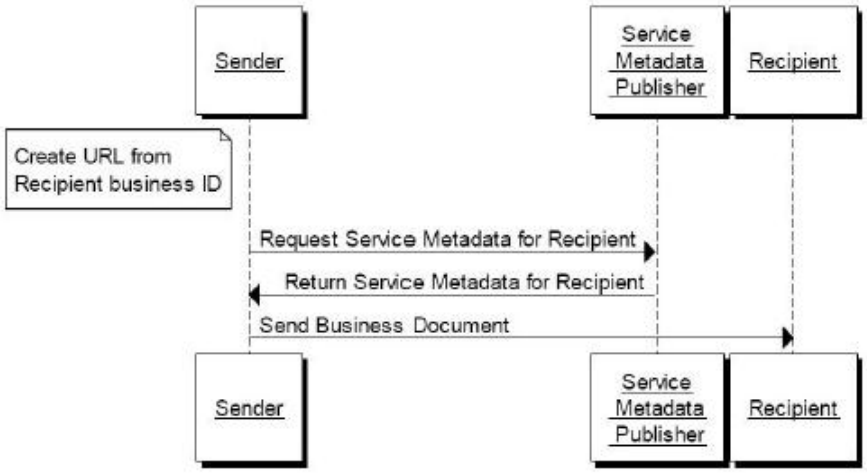


Figure 16 – Sequence Diagram for Sender transmitting Document to Recipient[69]

3.4.2. Service Metadata Publisher (SMP)

In order to enable routing of documents received from the sender to the correct recipient, the infrastructural information resides on servers called Service Metadata Publishers (SMPs). They store information about the receiving capabilities of participants connected to the network, provide the details about the business document types supported and the business collaboration profiles that can be processed through the infrastructure.

Every network participant is being registered in only one SMP registry. The gateway must identify the correct connection in order to retrieve the information about that specific recipient party. The gateway then receives the document and forwards it to the receiving gateway that will send an acknowledgement and pass it on to the recipient.

3.4.3. Processing Modes, CPPs and CPAs

As explained above, Processing Modes or P-Modes are introduced in ebMS 3.0 to define the configuration for ebMS Message Handlers. P-Modes are necessary if you want to setup an agreement between two partners as to how messages must be processed, on both the sending and receiving sides. Both partners must be able to associate the same P-Mode with a message for consistent processing (e.g. security, reliability, encryption, signing). So before a message is sent, the sender must be able to determine which P-Mode is used for this message.

In ebMS 3 the message-exchange capabilities of a party are thus expressed as a set of P-Modes. In a more general context, this concept is called a Collaboration-Protocol Profile (CPP). The OASIS Collaboration-Protocol Profile and Agreement Specification[70] describes a CPP as follows:

"A CPP defines the capabilities of a Party to engage in electronic Business with other Parties. These capabilities include both technology capabilities, such as supported communication and

[69] https://joinup.ec.europa.eu/svn/peppol/PEPPOL_EIA/1-ICT_Architecture/1-ICT-Transport_Infrastructure/13-ICT-Models/ICT-Transport-SML_Service_Specification-101.pdf

[70] OASIS, Collaboration-Protocol Profile and Agreement Specification Version 2.0, http://www.oasis-open.org/committees/ebxml-cppa/documents/ebCPP-2_0.pdf

messaging protocols, and Business capabilities in terms of what Business Collaborations it supports."

An agreement between two parties can be described by a Collaboration-Protocol Agreement (CPA). A CPA is therefore composed of the CPPs that the parties use, or of the P-Modes they have configured.

Examples of both CPPs and CPAs can be found in the Collaboration-Protocol Profile and Agreement Specification.

The SMP service metadata XML structures are very similar to the ebXML Collaboration Protocol Profile (CPP). A CPP defines what can be sent and what can be received while SMP only defines the messages they can receive. The OASIS BDX Technical Committee started a work item to look at ways to use SML/SMP and to retrieve configuration information for ebMS 3.

Also note that the *eb:CollaborationInfo/eb:AgreementRef* element in the ebMS *UserMessage* does not necessarily have to point to an instance of a CPA (though this is RECOMMENDED in the ebMS 3 specification).

3.4.4. Short-Term solution for e-CODEX

How dynamic discovery via SML/SMP can be made to work together with ebMS CPP/CPA mechanisms and P-Modes will need further elaboration together with PEPPOL and SPOCS. To be able to start its piloting according to the established time planning, e-CODEX will for a first implementation use static configuration as is available in out-of the box ebMS messaging products (e.g. in the OS solution Holodeck) , while working on the topic of dynamic discovery in parallel.

Since the ebMS standard does not specify the way that configuration information is retrieved (and, as mentioned above, the Collaboration Agreement reference can point to something different from a CPA), implementations are very much free to decide how to handle this. On interoperability this decision has little or no impact.

3.5. Evidences (Workflows)

For the transmission of legal documents it is very important to send and receive evidences indicating certain points in time, for example when a document is received by the recipient. It is so important because by receiving a document a legal period of time (e.g.20 days) starts, in which the recipient is allowed to react.

For the definition of such evidences the ETSI REM Specification and its defined workflows will be used. In ETSI there are two main workflows for sending a message including all attachments and for sending a message without the attachments, which can be downloaded later on from a repository.

According to the architecture of e-CODEX and the use of ebMS 3.0 as the underlying transport protocol (including the fact that there is no download repository existing) only the evidences defined for sending a message including all attachments will be considered.

The following figure describes the ETSI REM Workflow. The supported Evidences are marked with red circles.

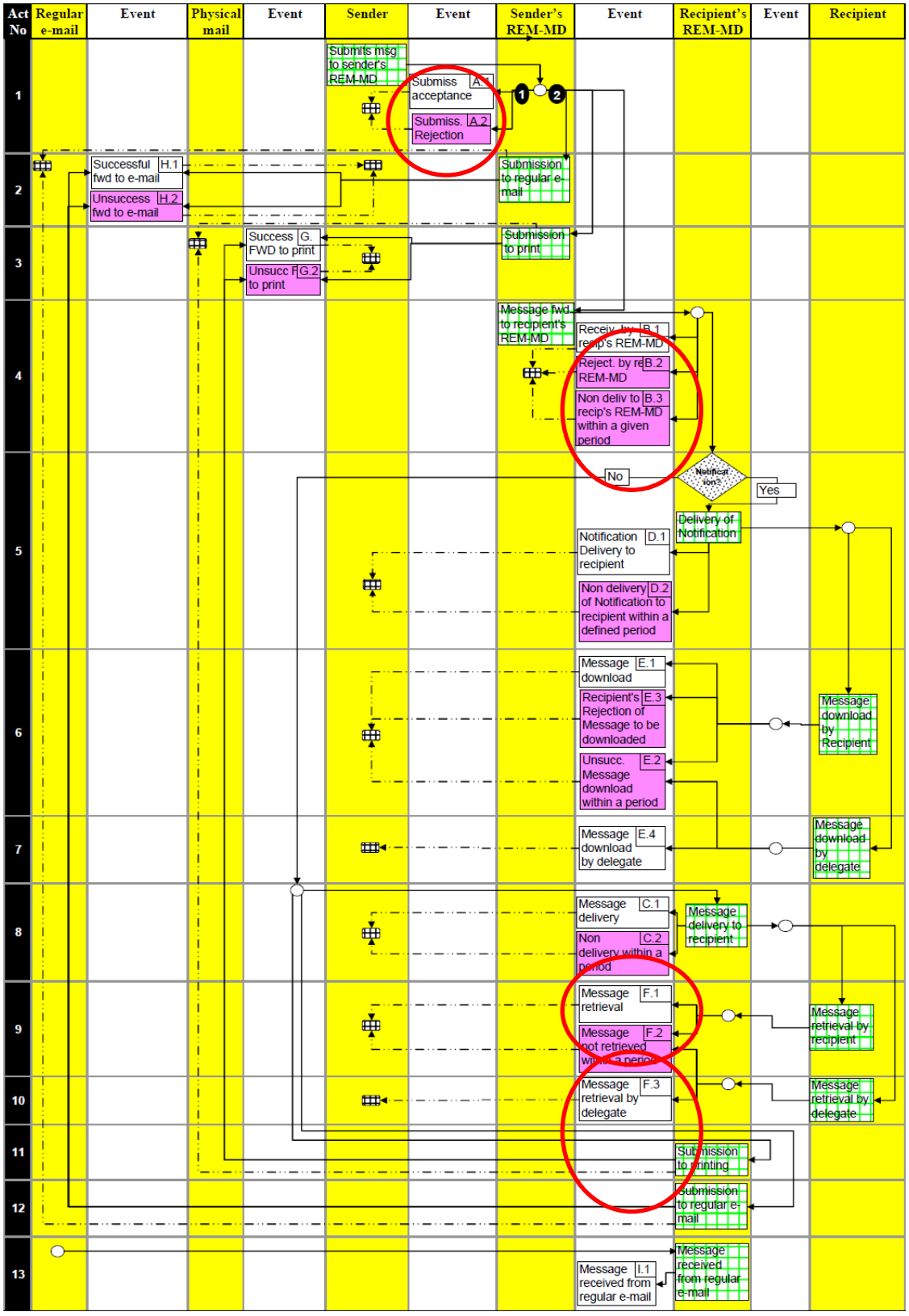


Figure 17 – ETSI REM Workflow (ETSI TS 102 640-1) *Fehler! Textmarke nicht definiert.*

The following table gives a summary and explanation of the supported Evidences.

	Evidence	TS reference	Description	SPOCS Reference
1	S-REM-MD Acceptance	6.2.1 – A 1	The senders gateway has accepted the message for sending	SubmissionAcceptanceRejection
2	S-REM-MD Rejection	6.2.1 – A 2	The senders gateway has rejected the message for sending	SubmissionAcceptanceRejection
3	REM Object Delivery	6.2.3 – C 1	The message has successfully been sent to the recipient (including national subsystem)	DeliveryNonDeliveryToRecipient
4	Non delivery within a given retention period	6.2.3 – C 2	The message has NOT been sent to the recipient (including national subsystem)	DeliveryNonDeliveryToRecipient
5	R-REM-MD Acceptance	6.2.2 – B 1	The message has been successfully delivered to the gateway of the recipient (excluding national subsystem)	RelayToREMMDAcceptanceRejection
6	R-REM-MD Rejection	6.2.2 – B2	The message has been rejected by the gateway of the recipient.	RelayToREMMDAcceptanceRejection
7	Expiration of time to deliver to R-REM-MD	6.2.2 – B3	The message has NOT been successfully delivered to the gateway of the recipient	RelayToREMMDFailure
8	Retrieval	6.2.3 – F1	The message has been accepted (opened) by the recipient	RetrievalNonRetrievalByRecipient
9	Expiration of time for Retrieval	6.2.3 – F2	The message has NOT been accepted (opened) by the recipient	RetrievalNonRetrievalByRecipient

Table 7: Supported Evidences

The third column in the table references to the ETSI REM Specification. For a detailed description please refer to this specification[71].

The last column in the above table references to the ETSI REM definition for the LSP SPOCS. SPOCS has defined the structure of an evidence in an XML Format which can be to some extent reused.

One basic reason for the implementation of evidences is the legal need to have an exact timestamp when a message or a document has been delivered. This is necessary because from this point of a time a statutory period is starting. In other words by receiving an evidence 6.2.3 –C1 a timestamp can be set. Please refer also to the e-CODEX D5.1 requirements specification for more information.

[71] ETSI TS 102 640-1, V2.1.1 (2010-01), Electronic Signatures and Infrastructures (ESI) Registered Electronic Mail (REM), Part 1: Architecture

3.5.1. Format of an evidence

There is already a format for evidences via SOAP defined by ETSI. The following figure gives an overview about this format.

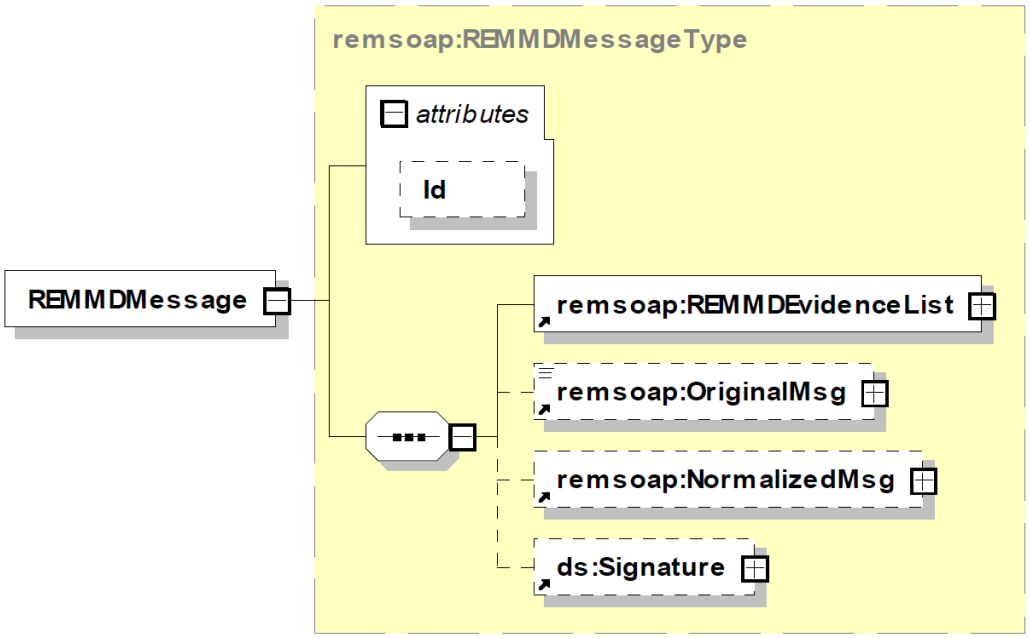


Figure 18 – REMMDMessage (ETSI TS 102 640)

The evidence includes an id, the original message and a signature provided by the gateway.

To take over the solution defined by ETSI and provided by the SPOCS LSP has the advantage to be able to reuse also the code for the generation of the evidence.

The evidence itself is transported within an ebMS message inside the message body. As for any other message the ConversationID (as part of the message header) will be set accordingly to be able to correlate the evidence to previous messages. A special action parameter identifies the message to be of the type evidence.

3.6. End-to-End Encryption

As the ebMS communication is between gateways only, a complete end-to-end encryption is not foreseen and will not be provided by e-CODEX. According to the ebMS 3 standard the communication between the MSH would be encrypted for each direct communication between two participants. As described in D5.2 “Reusable Assets“ the encryption/decryption of the message will be done at the e-CODEX gateway and the message will then be sent via the e-Delivery platform to the other Gateway.

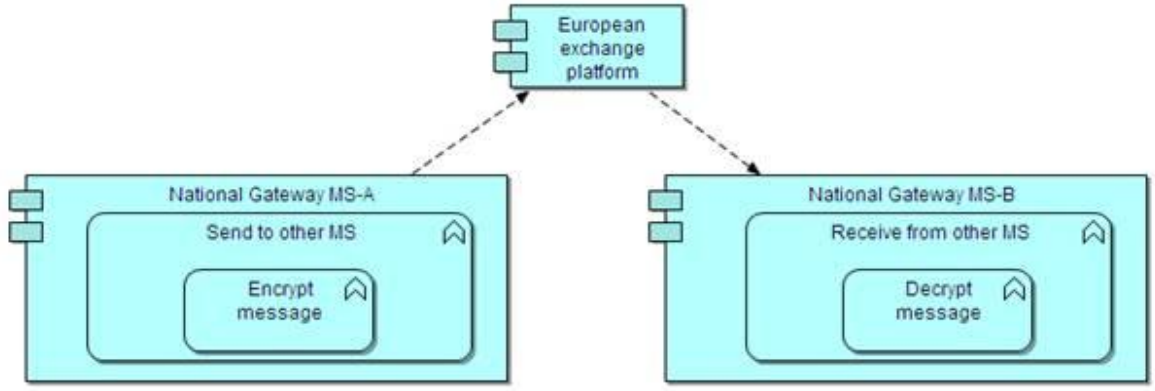


Figure 19 – Encryption Procedure

Notwithstanding the above, encryption will be done point-to-point for each communication step, as shown in the picture below.

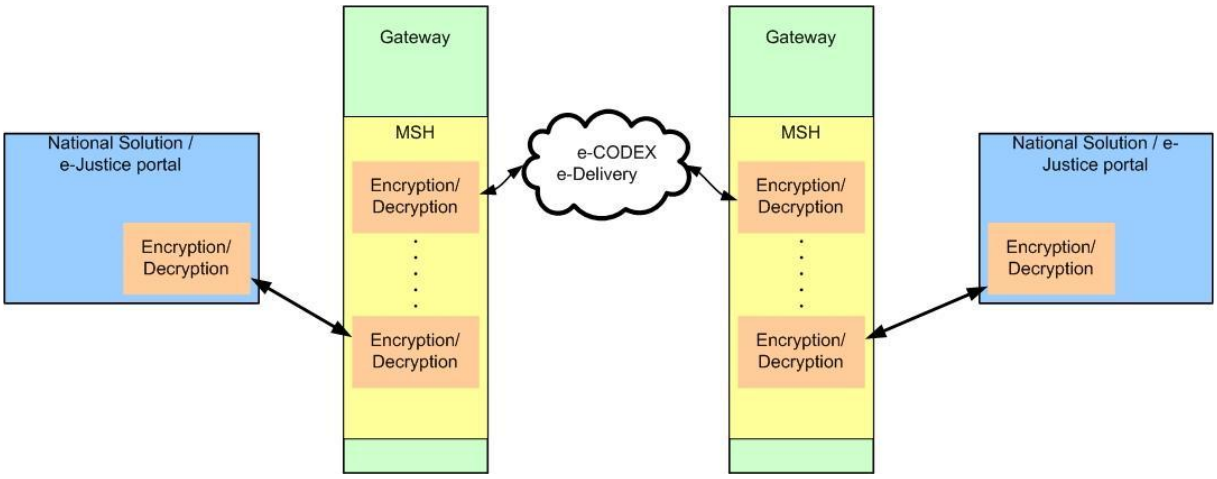


Figure 20 – Point2Point Encryption

A message submitted to a national transport solution will be encrypted and transported by that infrastructure to the e-CODEX gateway, where it is decrypted on reception. In the case of the European e-Justice portal the messages are created at the location of the gateway. (Browser communication between citizen and the portal will be SSL-encrypted).

In the gateway the e-CODEX software (the Pan-European transport platform) encrypts and sends the message. On the receiving end the gateway decrypts it and transfers it into the national transport solution, which re-encrypts and sends it.

To ensure a higher security level it is recommended that the adapter to the national transport solution and the MSH should be installed on the same system.

Note that if the national transport infrastructure does not in itself provide encrypted transport, the encryption has to be implemented in the adapter to that national solution, and decryption at the receiving side (e.g. in the connected back office system).

For example, if on the national side, messages were transported through regular e-mail, a separate application could encrypt all messages received by the gateway of that country before sending them on via e-mail, and the final recipient would then use another separate application to decrypt the received messages.

The encryption level to be applied at the national transport solution level should conform to the e-CODEX general security policy minimum requirements.

In addition, where real end-to-end encryption for documents is desired, this can be done by the original sender using means outside e-CODEX, and decryption will then have to occur at the ultimate receiver. The exchange of keys is in this case up to the end users. This can however not be done for structured data that require a transformation in the e-CODEX gateways.

3.7. Message size

The first step for defining a size limit for the e-CODEX solution is to analyse the existing size limits in the participating countries and their solutions. The table below provides an overview of the known restrictions, as collected by WP5.

Country	Maximum Message Size
Turkey	no restriction
Netherlands	no restriction
Italy	30 MB
France e-barreau	4 MB
France COMEDEC	Not planned to be connected
Spain	3 MB
Estonia Xtee (X-road)	no restriction
Germany EGVP	30 MB (max 100 attached files)
Czech Republic	10 MB
Austria	12 MB

Table 8: Message Sizes

The table shows that the lowest common denominator would be a limit for the overall transportation of 3MB for the whole message. However to enforce a common limitation for all e-CODEX communications does not seem feasible.

For the e-CODEX piloting a simple solution would be to enforce this limit on the sending side. For the European e-Justice portal this could easily be done as part of the form validation process when uploading documents as attachments. However since the size limits vary largely between Member States, it seems better to make the size limit dependant on the receiving country.

When supplying messages via a national transport infrastructure, it is the responsibility of the participating Member States to ensure size limits are not exceeded. Where this is not possible, the e-CODEX gateway will transmit an error message back to the original sender. This functionality should be built into the national adapter to that gateway (where needed), not the gateway itself.

In the context of further applications in the domain of e-Justice beyond the piloting, and also of e-Delivery convergence, the ability to handle large files will be a requirement. Therefore the transport platform must be capable to handle files significantly larger than the limits given above. Even so there will be the necessity to take care of receiving parties with size limits, and with more participants from different domains it will likely not be feasible to rely on the senders' software to enforce such limits.

A routing/addressing mechanism with a capability lookup as developed by PEPPOL and, as part of the convergence effort, intended to be used for the e-CODEX platform as well (see section 3.4 "Discovery, addressing and endpoint capabilities"), might offer a way out for the size limit problem as well: if the supported maximum size for its connected parties (e.g. national solutions in the e-CODEX case) are part of the metadata information provided for a gateway, the sending gateway could query this information and if needed provide error messages to the original sender – or if they so choose, applications on the original sender's side might even be enabled to get this information from their associated gateway before the actual sending occurs.

For the e-CODEX piloting, size limits for the national e-CODEX gateways will be configured via the *PMode.BusinessInfo.PayloadProfile.maxSize* setting.

3.8. Payload

Similar to what has been described for file sizes in chapter 3.7 Message size e-CODEX WP5 has also analysed the payload restrictions of the national solutions (see the table below).

Country	Payload
Turkey	no restriction
Netherlands	EDIFACT, XML, PDF, JPEG, flatfile
Italy	no restriction
France e-barreau	XML, PDF, RTF, ODF
France comdec_2	N/A
Spain	RTF, PDF, ODT, JPG, TIF, XML, ZIP
Estonia	Unknown
Germany	TIFF, ASCII, Unicode (UTF-8), PDF(/a), RTF, XML, MS Word (up to MS Word 2003), MS Excel (up to MS Excel 2003), filename restriction on usage of ä, ö, ü etc.
Czech Republic	Unknown
Austria	XML, PDF

Table 9: Payload Types

The EU-regulations relevant for the planned pilots require the ability to transport any kind of data. Connected national solutions accordingly have to support any kind of payload format. Where national law applies, format restrictions may exist, but to enforce such restrictions is out of scope for WP5. For the e-CODEX piloting it is recommended that the piloting countries agree on a small number of acceptable formats to be determined by WP6.

For the e-CODEX piloting, accepted file types for the national e-CODEX gateways will be configured via the *PMode.BusinessInfo.PayloadProfile* setting.

4. Gateway

In this chapter the Gateway should be described in more detail. The solution is the same whether for the European (hosted by the European e-Justice Portal) or for the national gateway. The open source product Holodeck will be used in order to implement the ebMS 3.0 standard. This will serve as the basis for the e-CODEX Gateway. The reason for choosing this product is that it is freely useable (open source), easy extensible and it is at the moment the only open source implementation of the ebMS 3.0 stack available at the market. Furthermore, according to the timing and budget constraints an own implementation of the ebMS 3.0 specification within the e-CODEX project is not possible.

4.1. Architectural overview

The following figure will give an overview about the architecture of the e-CODEX Gateway.

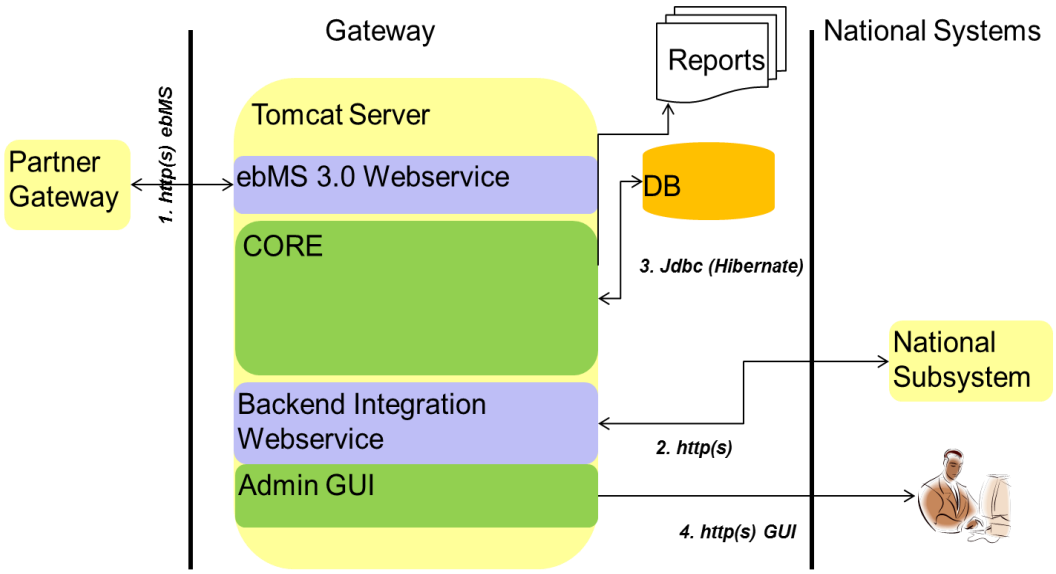


Figure 21 – Gateway architecture

The Gateway software is using a web application server as the runtime environment and its installation includes a customized and extended Holodeck implementation contained within dedicated web applications. As the standard application server the product TOMCAT has been chosen, because it is already integrated with Holodeck and most of the available J2EE servers use TOMCAT internally. Nevertheless any other compliant J2EE server could be used according to the needs and requirements of the piloting MS. Also Oracle Weblogic will be tested, if it turns out to be a requirement from the European Commission to host the Gateway in their data centre.

It should be mentioned the Holodeck itself is based on the open source AXIS Web Service Stack used for handling the Web Services. Due to the fact that AXIS is the one of the most used Web Service Stack, it can be assumed to be interoperable and well tested.

AXIS provides also the possibility for adding user defined modules (described in chapter 4.3.2.4), which can be used to customize the GW.

External Interfaces

The solution provides the following external interfaces:

- A secured ebMS based connection to the partner gateways.
- Web Service Interface to the national Backend Systems

Core Services

Besides the external interfaces a core set of services (please refer to 4.3.1) and DB instances holding the logging information exist. The physical DB installation is not part of the solution and can be done according to the needs and the existing infrastructure of the MS. The integration of the Gateway software with the DB is done using Hibernate.

User Interface

Additionally to the core services a web based graphical user interface implementing an access to the logging data is to be provided.

ebMS 3.0

The main part of the e-CODEX Gateway solution is an ebMS 3.0 Web Service handling all the traffic to and from the other partner gateways according to the ebMS 3.0 standard specified by OASIS. This web service includes already the security handling on the network layer based on SSL client server communication as well as on the data level by encrypting and signing the messages. This web service includes also reliable messaging feature with a guaranteed once and only once delivery.

Backend Integration

Besides the ebMS 3.0 Web Service another Web Service Interface called Backend integration is foreseen.

It propagates the messages and their data towards the existing national electronic legal communication systems for further processing.

For detailed information on packaging please refer to chapter 4.2

4.1.1. Hardware Setup

Please note: Of course the setup is within the responsibility of the MS, therefore this chapter is only a recommendation.

To be able to achieve a high availability and exclude a single point of failure according to the requirements the following Hardware (HW) setup is recommended:

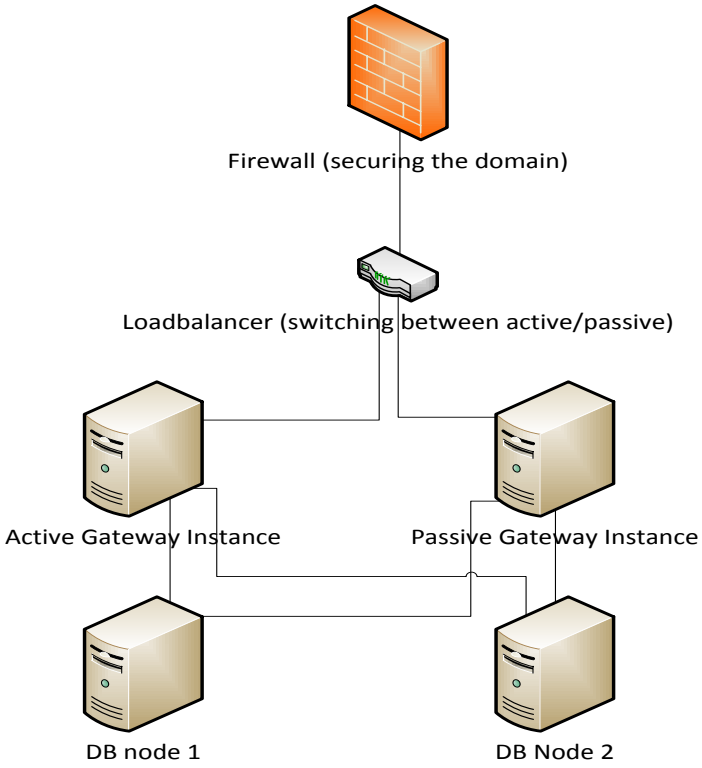


Figure 22 – HW Setup

Two separate HW instances (or virtualised instances) which are hosting the Gateway software in an active /passive mode. In front of these server instances there is a HW load balancer checking the availability of the server instances and in case the active server is not responding any more it switches to the other passive one. Also the DB is running in a separate process preferable also within a cluster to eliminate again a single point of failure. Today most database solutions (e.g. Oracle or MySQL) are supporting this cluster mode.

Due to the fact that the whole implementation is based on Java, all OS and platforms which support Java are supported.

4.1.2. Interfaces

The following external interfaces according to the above figure will be provided by the Gateway.

4.1.2.1. Partner Gateway

This is the interface towards the other MS Gateways participating on the e-CODEX project. Technically this interface is based on ebMS 3 (for details please refer to chapter 3 The European e-Delivery Transport Infrastructure). On the network layer it is an http(s) connection with client –server authentication.

It has been defined to start for the pilots with a point to point communication between all the partner gateways similar to what has been done in the other LSP's SPOCS and PEPPOL. If the European Commission is willing to host in future a general hub (platform) in the middle and if it is needed to be able support a huge number of different cross border proceedings in different governmental areas, then this point to point setup can be changed in an extensions phase to a central hub. The following figure describes the network setup for the e-CODEX project.

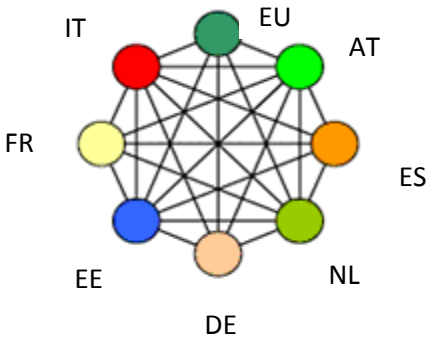


Figure 23 – e-Delivery network setup

4.1.2.2. Backend Integration - National System

This is the interface to the national backend system hosted behind the gateway. It is Web Service based and defined in chapter 4.3.2.2. On the network layer it is an http connection. For the e-CODEX Pilots it is highly recommended to use a SSL connection with client – server authentication.

4.1.2.3. Database Interface

This is the internal interface from the gateway to the internal DBs as the logging DB or the temporary message DB for reliable messaging. This interface is only visible to outside in case of querying the log entries. It is based on JDBC and DB independent implemented with the JPA and Hibernate. Therefore every piloting MS can choose its own DB according to his infrastructural needs.

4.1.3. National TSL or authentication and authorization systems (IDM)

Please note: This chapter is just for clarification of the whole process for generating the Trust Ok Token. It is a functionality of the national system and not of the gateway.

The national systems are in the responsibility to check if the author of a message is authenticated, trusted and the content is not manipulated. This can be proven either by digital electronic signatures or by using a closed system like the ERV in Austria. This is a specific requirement to e-CODEX and maybe not needed for other LSP's. Anyway a so called Trust Ok Token must be generated according to the above mentioned checks.

The authentication of the author of a message by the national adaptors will be done according to the provisions (minimum requirements) of the overall project's security policy. This security policy (also called Circle of Trust) has been defined in the combined deliverable D3.2 & D7.2[72] as follows:

As e-CODEX Service Provider for the civil use cases, e-CODEX will accept and will only accept an "advanced electronic system". An advanced electronic system is an electronic system which meets the following requirements:

[72] D7.2 Requirements Finalisation & D3.2 Described Test Scenarios

(a) the created document is uniquely linked to the user;
(b) the system is capable of identifying the user;
(c) the document is created using means that the user can maintain under his control; and
(d) any subsequent change of the data of a created document is detectable;

This checks are implemented either via a connection to the national TSL to verify if the certificates used for signing the data and the signature as well is valid. In case a document signature is not available the credentials within a closed system (e.g. ERV) according to the above definition of an advanced electronic system are checked.

4.1.4. Call Flow

The following chapter describes the two different main call flows implemented by the gateway.

- Receiving Messages or Evidences
- Sending Message or Evidence

4.1.4.1. Receiving Message or Evidence

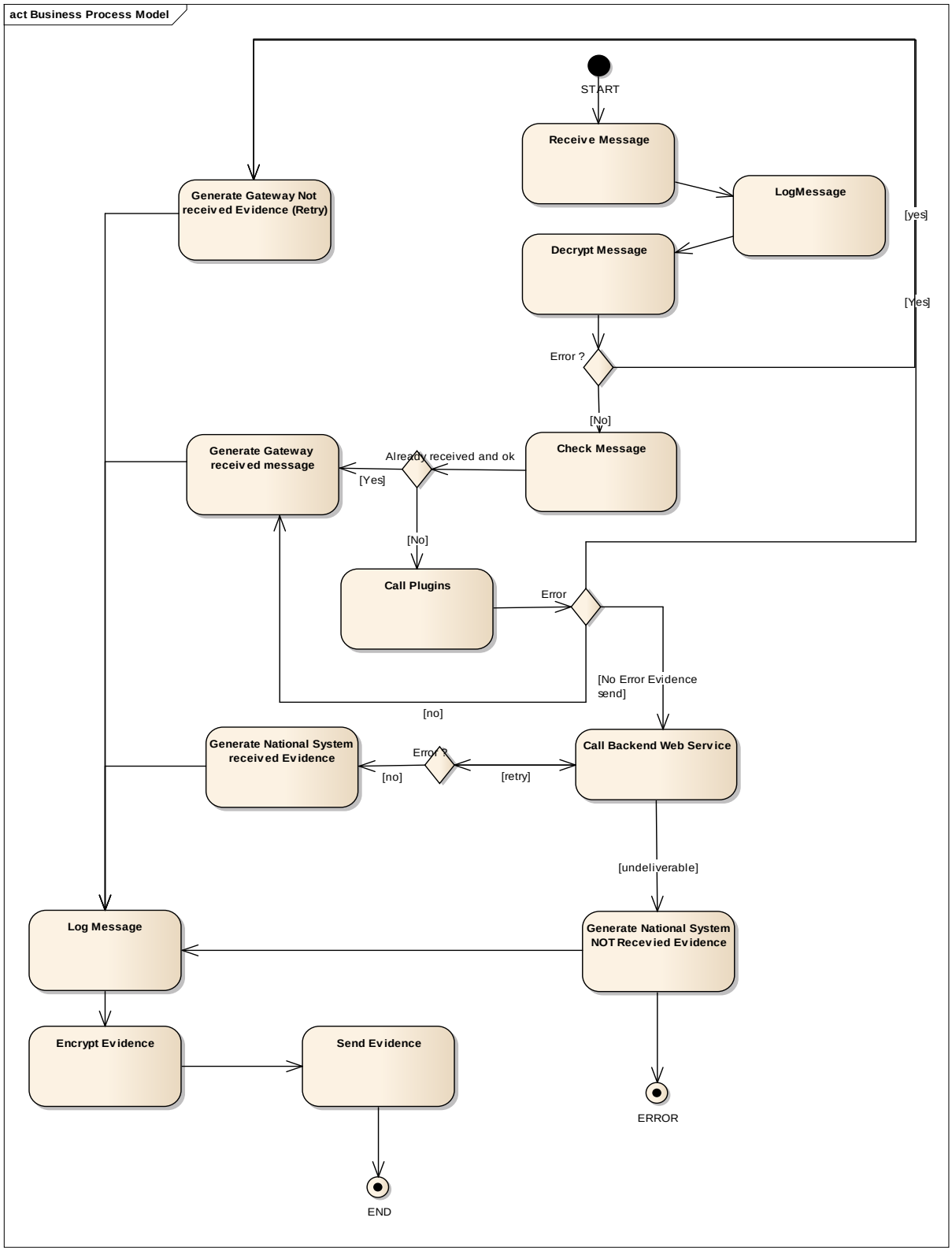


Figure 24 – Receiving call flow at the gateway

This use case always starts by receiving a message or evidence from a partner Gateway. As soon as a message arrives at the Gateway a log entry is created with the state RECEIVE. Then the message is decrypted and checked if it has been already received. In this case evidence (6.2.3 – C 1, see Table 7: Supported Evidences) is sent back and the state in the log DB is changed to DUPLICATE. If the message is an evidence to a previous sent message then the state in the log DB is changed accordingly. If the message is new then the national Plugins are called to further process the message. If the processing failed then an evidence (6.2.2 – B 2, see Table 7: Supported Evidences) is sent back and the message state is change to RECEIVE_FAILED. Otherwise an evidence (6.2.2 – B 1, see Table 7: Supported Evidences) is sent back to the partner gateway and the state is changed to RECEIVE_GATEWAY_FINAL.

Then the message is sent via the Backend Integration Web Service to the national subsystem. If the sending to the national subsystem is successful (including maybe some configurable retries) then an evidence (6.2.3 – C 1, see Table 7: Supported Evidences) is sent back to the partner system. A successful sending is defined if the national subsystem is calling the method setState of the Backend Integration Interface (chapter) with the state RECEIVE _FINAL. Otherwise an evidence (6.2.3 – C 2, see Table 7: Supported Evidences) is sent back and the state is changed to RECEIVE _FAILED.

RECEIVE_GATEWAY_FINAL means that the message has been received at the gateway successfully, whereas RECEIVE _FINAL means that also the national Subsystem has successfully received the message.

4.1.4.2. Sending Message or Evidence

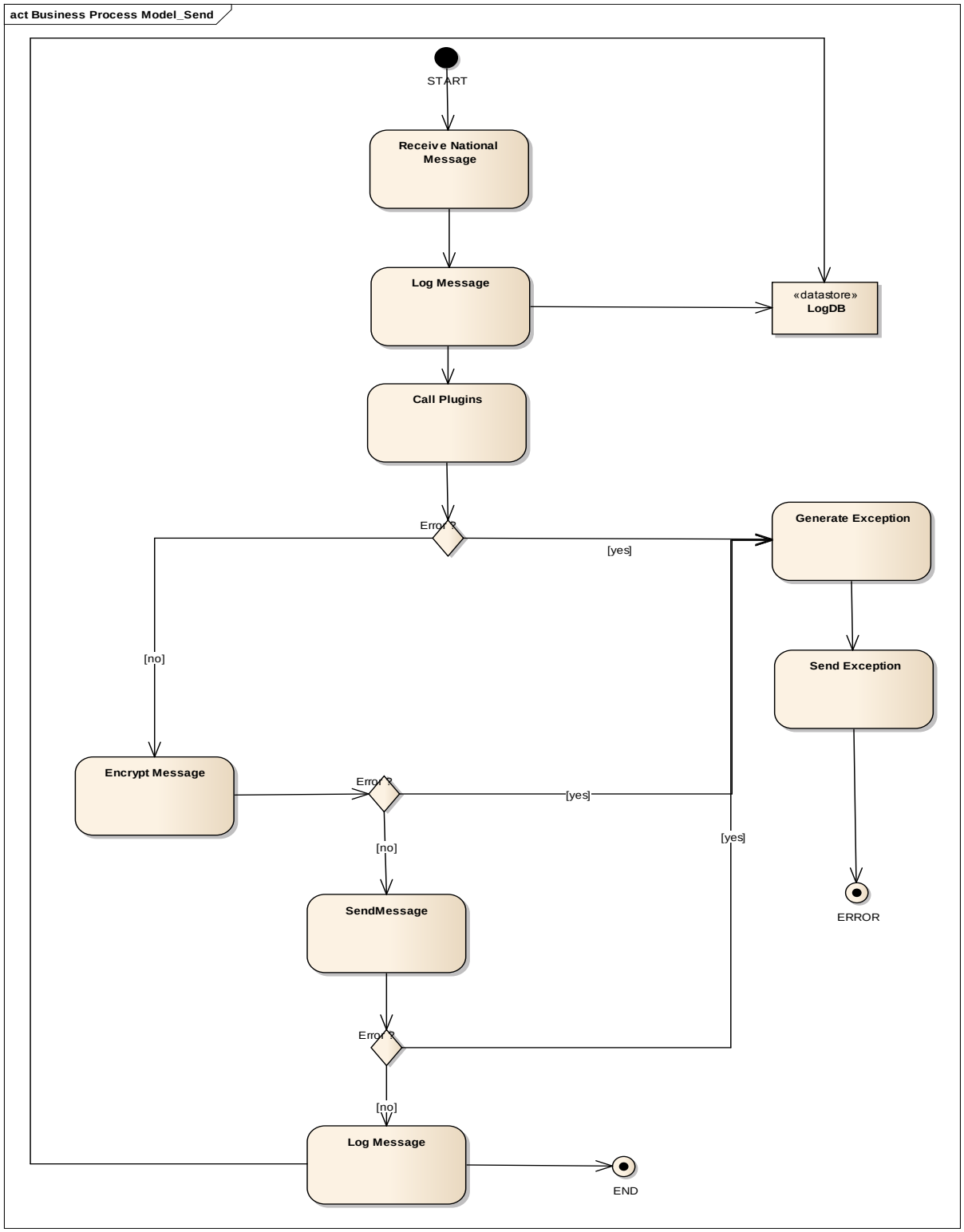


Figure 25 – Sending call flow at the gateway

Sending a message always starts by receiving a message from the national legal communication subsystem. After the message is received at the Gateway by the backend integration interface (chapter 4.4) a log entry is written with the State SENDING. Then all available and configured national plugins are called to further process the message. In case of an exception during the processing of the plugins an exception is thrown and sent back to the national subsystem. Additionally the log state of the message is changed to SEND_FAILED. If the plugin processing was successful then the message is sent to the desired partner Gateway of the receiving Member State and the state of message is changed to SEND_FINAL. Otherwise in case of an error during the processing an exception is thrown and sent back to the national subsystem. If later on an evidence (6.2.3 – C 1 or 6.2.3 – C 2, see Table 7: Supported Evidences) from the partner gateway arrives then the state is set to SEND_FINAL_CONFIRM in case of success or otherwise to SEND_FAILED. These changes are transported via the method setState of the Backend Integration Interface (chapter 4.4) to the national sub system. In case of an error the message is resent several (configurable) times. If the resending is still not successful then the state of the message is changed to SEND_FAILED and an exception is sent back to the national subsystem.

4.1.4.3. Overview about the possible message states

SENDING	Message has been received from national backend system, but not sent to the partner gateway
SEND_FAILED	Message has been received from national backend system, but sending to the partner gateway was not successful or the internal processing at the gateway failed.
SEND_FINAL	Message has been send to the partner gateway successful.
SEND_FINAL_CONFIRM	Message has been send to the partner gateway successful and to the recipients' inbox on the national system.
RECEIVE	Message has been received and processing at the gateway starts.
RECEIVE_FINAL	Message has been received from the partner gateway and propagated to the national subsystem successfully.
RECEIVE_FAILED	The received message could not be processed at the gateway.
DUPLICATE	Message has been already received.
RECEIVE_GATEWAY FINAL	Message has been received, but could be propagated to the national subsystem successful.

Table 10: Message States

4.2. Deployment and Installation

First of all the whole product consists of one web application file (WAR) which is deployed within a Tomcat server. This web application consists of the following elements:

Other Web Applications for:

- Logging
- Backend Integration
- Administration

AXIS 2.0 Modules for:

- ebMS 3.0
- Security
- Reliability
- e-CODEX

Periodic jobs (JAVA) for:

- reporting
- generating evidences
- retrieving/sending messages from/to the national gateway (file, Web Service)

The installation itself is very easy and consists of the following steps:

1. As a precondition the backend integration (e.g. availability of the corresponding national web Service) has to be finished and up and running.
2. The war file has to be copied into the Tomcat server.
3. Afterwards the specific configurations as described in chapter 4.4.4 need to be done.
4. The DB has to be setup and configured correctly.
5. Ensure that all the firewall settings are correct to be able to send and retrieve message from the partner gateways.
6. At the end just restart the Tomcat server.

4.3. Components

The standard e-CODEX Gateway will be built up on an already existing open source product called Holodeck, consisting of a list of AXIS Modules and some basic functionality on top described in chapter 4.3.1.

Additionally there will be some extensions provided by e-CODEX, consisting of an own AXIS Module and some other components like e.g. a backend integration web service. This description will be found in chapter 4.3.2.

4.3.1. Core Components

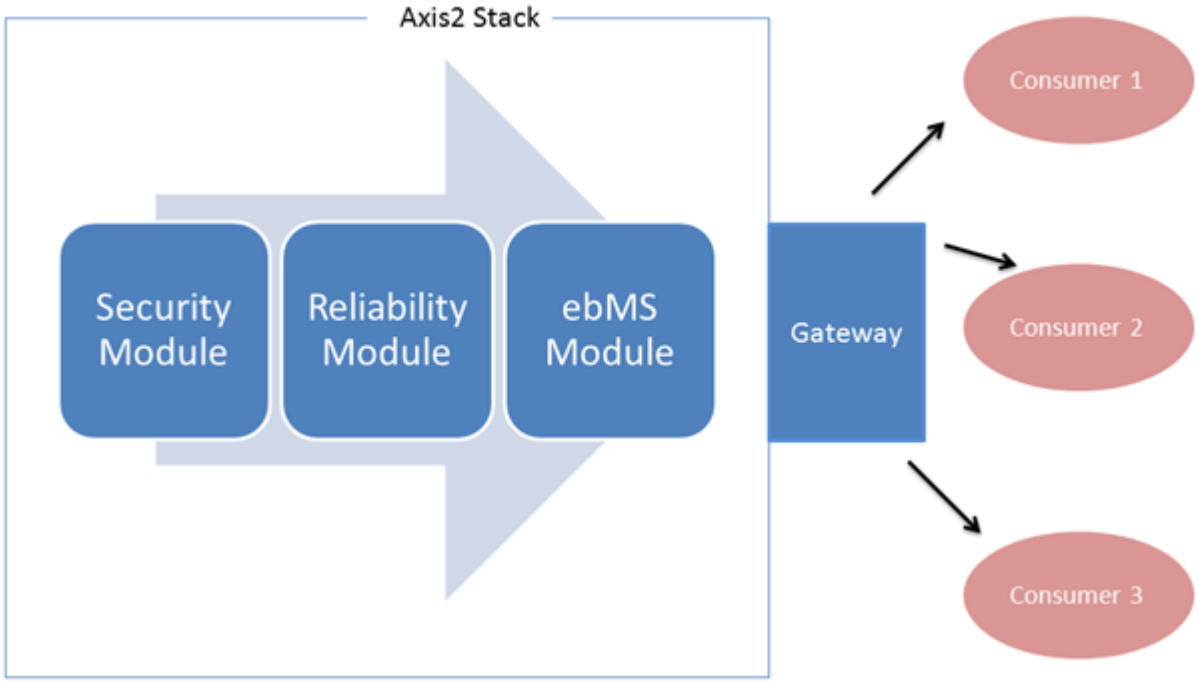


Figure 26 – Holodeck core components[73]

When a message arrives it is processed by different modules. The first is the security module, which will verify signatures and decrypt the message if necessary, then follows the reliability module and then the ebMS module. After the message passed these modules the gateway component can determine the consumer of the message.

4.3.1.1. Gateway

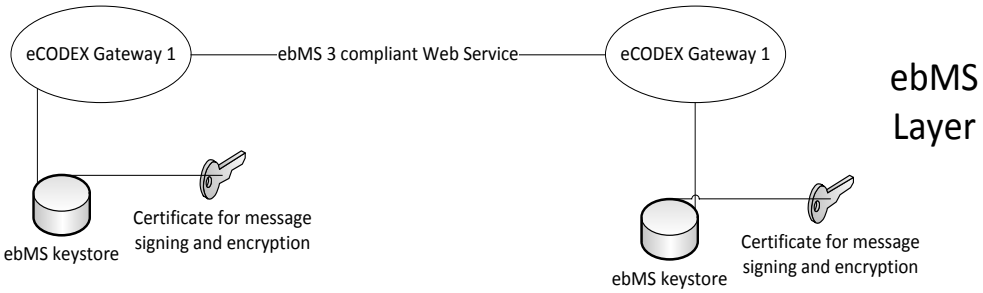
This component is responsible for providing the general functionalities like:

- Database access
- Configuration

For Database Access an intermediary between the real database and the access to it is established. A standard tooling called Hibernate will be used for this purpose. So it is up to the Member States which DB will be used. All widely used DBs solutions like e.g. MySQL or Oracle will be supported. This is a gateway internal component and not visible to developers as well as administrators so it will not be described in detail in this document. For the configuration a set of XML Files are provided. A detailed Description of the parameters and their functionality can be found in chapter 4.4.2.

[73] http://holodeck-b2b.sourceforge.net/docs/images/picture4-a.png

4.3.1.2. Security



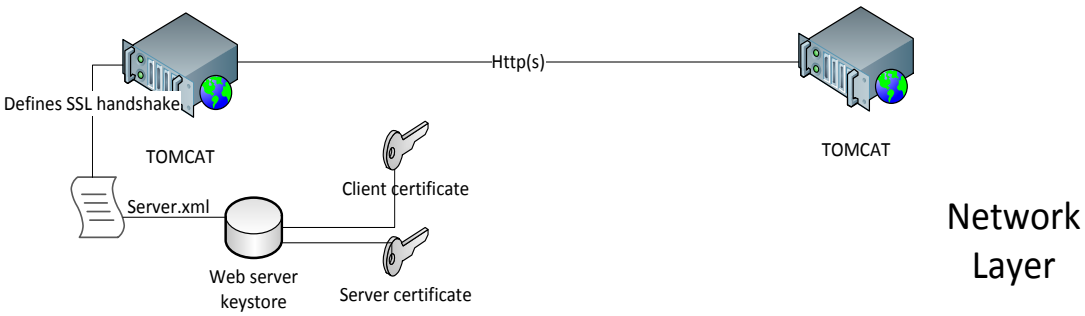


Figure 27 – Basic security setup

It has been decided that for the piloting phase of e-CODEX a static routing together with predefined certificates for signing and encrypting between the Gateways will be used which has the following consequences:

- The Gateway to Gateway identification is done in a static way and in advance by configuring the Endpoints URLs within the P-Mode configuration (defined by the ebMS specification). Please refer to chapter 4.4.4.
- The Gateway certificates used for signing and encrypting the messages are also configured in advance and stored locally at the gateways side in specific keystores. (Please refer to chapter 4.4.4). There is no dynamic certificate allocation and /or verification. This functionality will be provided at a later stage.
- The Gateway certificates used to secure the Gateway to Gateway connection on a Https level are also stored in keystores used by the underlying application server.

4.3.1.3. Reliability

This basic module supports the redelivery in case of errors in different predefined policies and according to the ebMS specification concerning reliable messaging. To be able to do so Holodeck provides a small DB for the outgoing messages, where all the messages are stored as long as they are not finally delivered. In case they are delivered or they could not be delivered within a certain time period then an evidence is sent back to the national system (in case the national system is able to handle evidences) and the status (please refer to Interface description 4.3.2.2.1.3) is set accordingly. Afterwards the messages are automatically removed from this DB. The time period for redelivery is

configurable within the corresponding reliability profile (please refer to chapter 4.4.4.3). It is a legal requirement for the e-CODEX project not to store the message content after delivery.

To see how this module can be configured please refer to chapter 4.4.4.3.

4.3.1.4. ebMS 3.0

In this basic module the specification of ebMS is implemented, meaning the basic communication protocols as well as the basic configuration like P-Modes. For a more detailed explanation on ebMS and the protocols please refer to chapter 3.2 and the corresponding specification[74].

4.3.2. Extensions provided by e-CODEX

Additionally there are some extensions to be able to fulfil the requirements of e-CODEX. For example the generation and sending of evidences as well as the provision of audit logs are not part of the standard open source product.

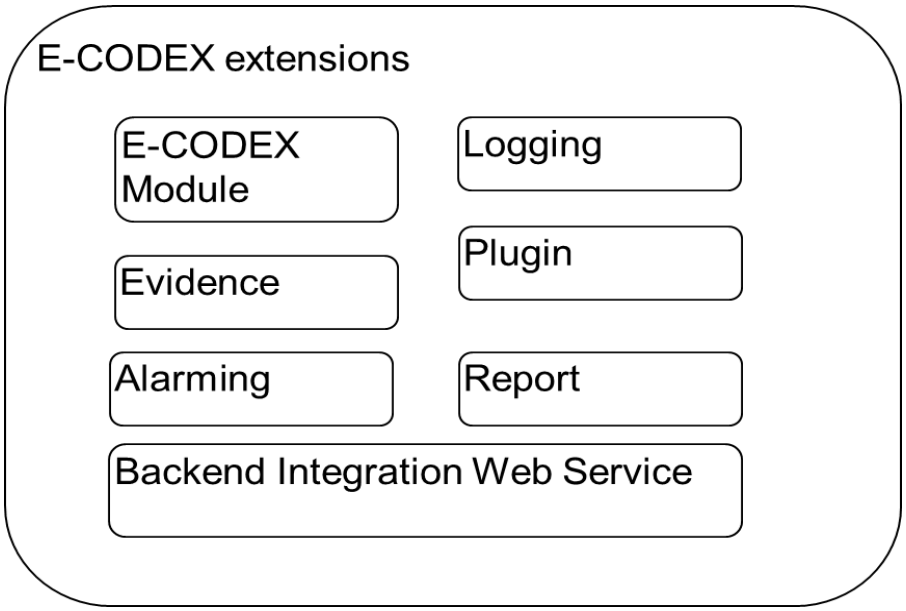


Figure 28 – Gateway e-CODEX Extensions

4.3.2.1. e-CODEX Module

This component consists of a set of call flow handlers responsible for checks and transformations needed to be done on outgoing and incoming messages. The following list gives an overview about the different handlers depending if they are used for received messages (incoming flow) or sent messages (outgoing flow).

- Outgoing Flow: According to the e-CODEX pilots it is necessary to validate the signature of the applied documents and to generate a so called Trust Ok Token. Due to the fact that there are some MS, which do not have a signed document according to the definition of an

[74] OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features, http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/ebms_core-3.0-spec.pdf

"advanced electronic signature system" the corresponding authentication information is used to generate the Trust Ok Token.
Anyway it has been decided to provide a module for the generation of the trust ok token by WP4 and also the call of this module should be done outside the GW in the national systems.

- Both directions: A handler responsible for storing and writing logging information in the DB as well as in a file for bug fixing and debugging.
- Both directions: The sending and receiving Gateways are resolved by the usage of the P-Mode configurations (please refer to 4.4.4.2). The end user address needs to be resolved at the national backend systems. The successful or unsuccessful delivery to the end recipient is forwarded from the national backend system to the Gateway by setting the status (please refer to the Interface definition 4.3.2.2.1.3) accordingly and by sending an evidence if possible.
- Outgoing Flow: A handler responsible for creating the end user identification of the sender and the receiver as additional ebMS header parameter. The definition of these parameters can be found in chapter 3.3.3.1.

4.3.2.2. Backend Integration Web Service Interface

For the interconnection towards the national e-Delivery systems the e-CODEX Gateway supports a Web Service interface in both directions.

- Outgoing: The gateway implements a Web Service which will be called by the national system. e-CODEX is acting as a server.
- Incoming: The national system will implement a Web Service, which will be called by e-CODEX. e-CODEX is acting as a client.

For both directions the same interface will be used, which is described in the next chapter.

The following figure gives an overview in case of Austria. The parts which represent the existing unchanged national solution (ERV) are marked in blue. The new Web Service to be implemented by the MS is marked in red colour and the parts provided by e-CODEX are orange.

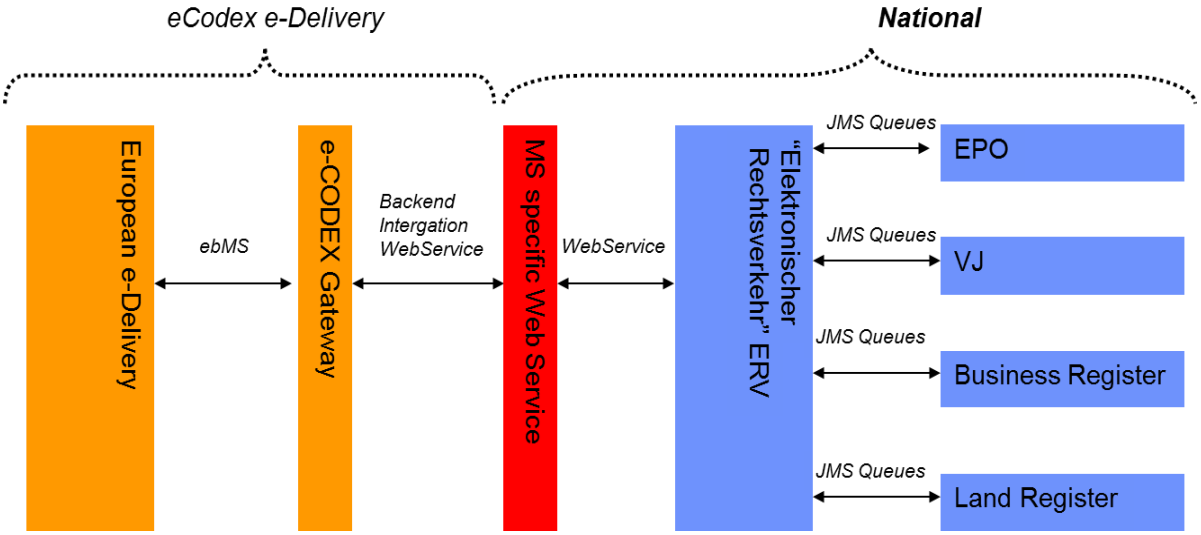


Figure 29 – Backend Integration

4.3.2.2.1. Interface

The Interface between the e-CODEX gateway and the national backend is the same in both directions. The following two figures describe the call flow for each direction.

For sending a message from the gateway to the national backend infrastructure the gateway calls the *sendMessage* method of the national Web Service implementation as a client. Alternatively the call of the method *sendMessageWithReference*, means that only a reference to the payload is transmitted and not the payload itself. To indicate a successful receipt of the message the national subsystem calls the *setState* method of the Web Service at the gateway. The gateway can therefore create and forward an evidence to the partner gateway.

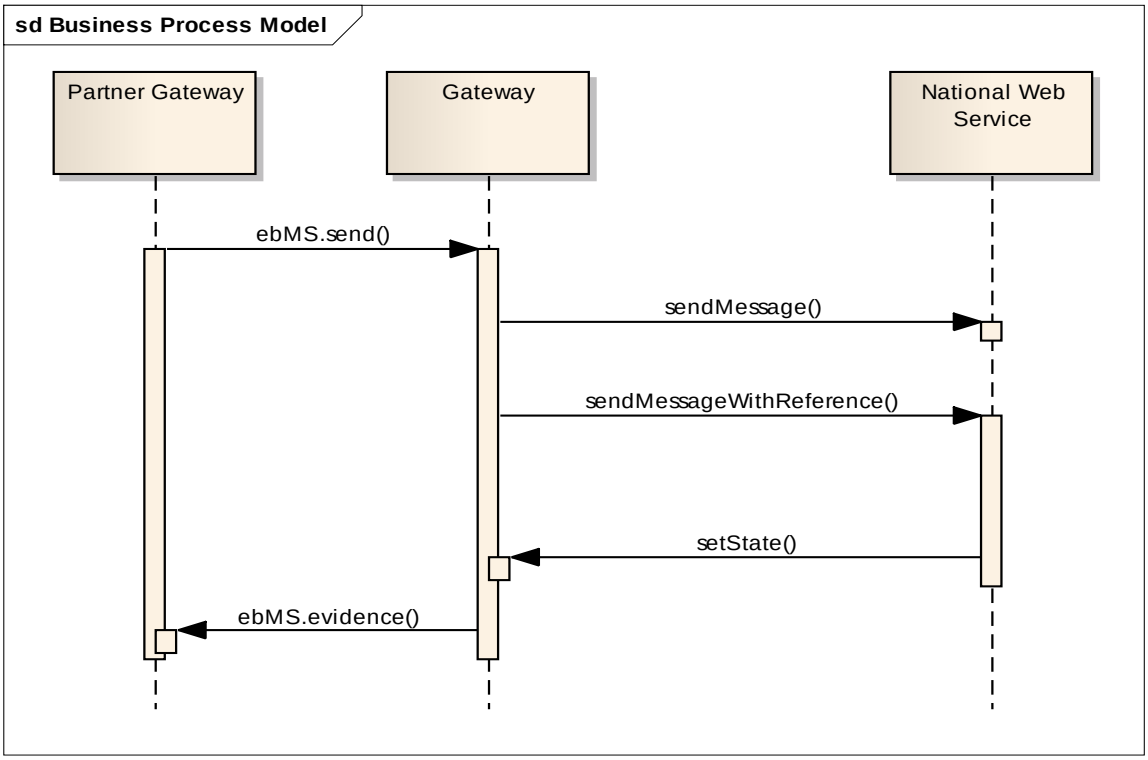


Figure 30 – Gateway to Backend System

For sending a message from the national backend infrastructure to the gateway the national subsystem calls the *sendMessage* method of the Gateway Web Service as a client. Alternatively the call of the method *sendMessageWithReference* means that only a reference to the payload is transmitted and not the payload itself. To indicate a successful receipt (triggered by receiving a corresponding evidence) of the message the gateway calls the *setState* method of the Web Service at the national subsystem.

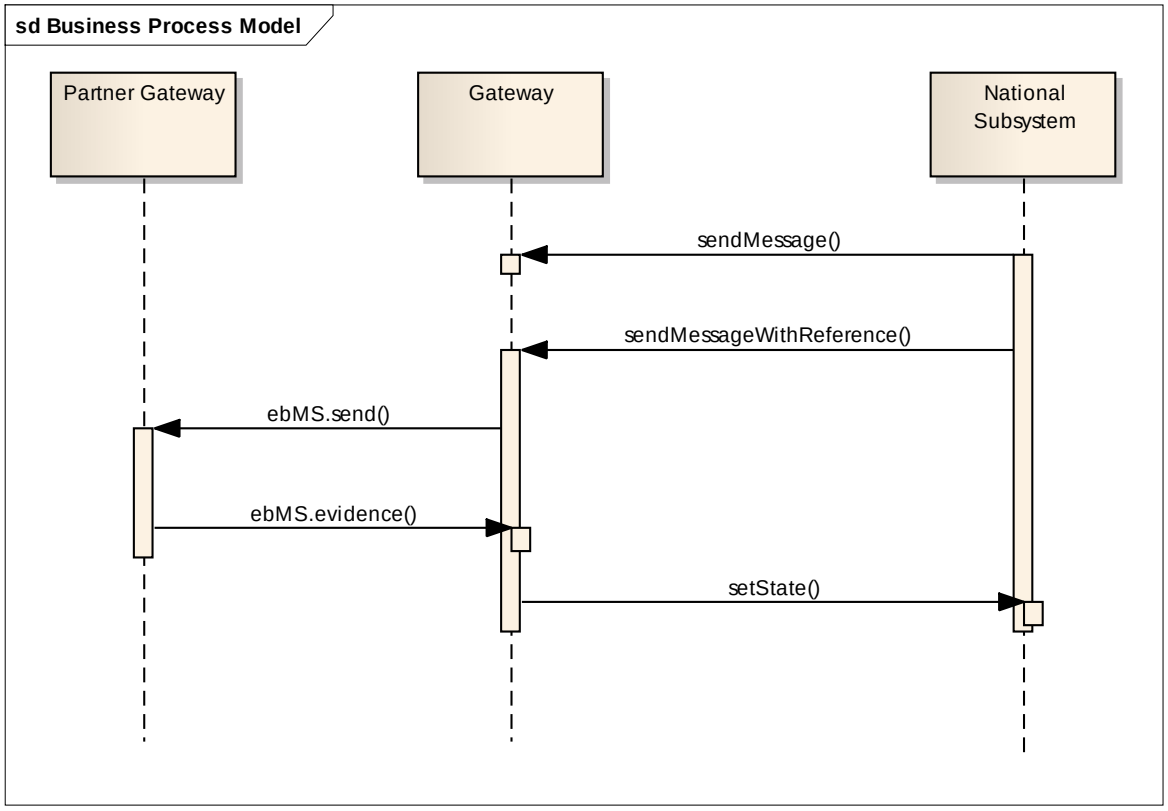


Figure 31 – Backend System to Gateway

4.3.2.2.1.1. sendMessage

This method will be invoked by the gateway when forwarding a message to the national backend web service or when the national backend system is transferring a message to the gateway. This method is only valid for a dedicated message.

Input:

Fieldname	Type	Mandatory	Description
Sender	STRING	Yes	Holds the national end address of the sender (e.g. ERV or EGVP address)
Receiver	STRING	Yes	Holds the national address of the receiver (e.g. national court id of the receiving country)
MS	STRING	Yes	The ID of the Gateway to which the message should be sent or from which the message has been received.
Proceeding	INT	Yes	Could be for the first release one of the selected pilots (EPO, Small Claims, Arrest Warrant, Secure Data Exchange). Defined by constants.
Correspond	INT	No	If the message belongs to a previous

			sent or received message then this message id is stored.
action	STRING	No	The dedicated action for a proceeding. (e.g. in case of EPO Form A)
ID	STRING	Yes	The unique ID of the message produced by the national subsystem
Payload[]	Array of BINARY"s	No	Any arbitrary list of attachments for the proceeding. This will maybe vary from MS to MS.

Table 11: Input Parameters – SendMessage

The method returns only one parameter which is the state of message.

4.3.2.2.1.2. sendMessageWithReference

This method is exactly the same as above but with the difference, that instead of the Payload itself in form of a Binary only an URL is provided pointing to the store of the attachment. This method could be useful to reduce internal network traffic and to work around any SOAP with attachment issues.

Fieldname	Type	Mandatory	Description
Payload[]	Array of URL	NO	Any arbitrary list of attachments for the proceeding. This will maybe vary from MS to MS.

Table 12: Different payload – SendMessageWithReference

The method returns only one parameter which is the state of message.

4.3.2.2.1.3. setState

This method is necessary to set the state SEND_FINAL_CONFIRM at the national subsystem to indicate a successful delivery to the recipient or to set the state RECEIVE_FINAL at the gateway to trigger the gateway to send an evidence 6.2.3 – C 1, see Table 7: Supported Evidences.

Input:

Fieldname	Type	Description
ID	STRING	The unique ID of the message
state	INT	Could be SEND_FINAL_CONFIRM or RECEIVE_ FINAL

Table 13: Input Parameters – SetState

There is no return parameter for this method.

A WSDL of the message can be found in the Appendix II – Backend Integration WSDL.

4.3.2.3. Logging

This component contains the functionality of writing logging information transactional into a configurable DB, so that this information can be used later on for reports and auditing purposes. Of course the logging information contains only message headers like date and time sender and recipients but in no case any message content is stored.

Structure of the logging DB:

Fieldname	Type	Default	Description
Date	DateTime	--	Holds the date and time the message has been sent or received
Status	INT	--	Holds the status of the message
Sender	STRING		Holds the address of the sender
Receiver	STRING		Holds the address of the receiver
Proceeding	INT		Could be for the first release one of the selected pilots (EPO, Small Claims, Arrest Warrant, Secure Data Exchange)
Type of Message	INT		A Message could be a message(m) or an evidence (e)
Correspond	INT		If the message belongs to a previous send or received message then this message id is stored.
ID	STRING		The unique ID of the message

Table 14: Structure Logging DB

There will be also a web based GUI providing a statistical overview on message traffic running over the Gateway.

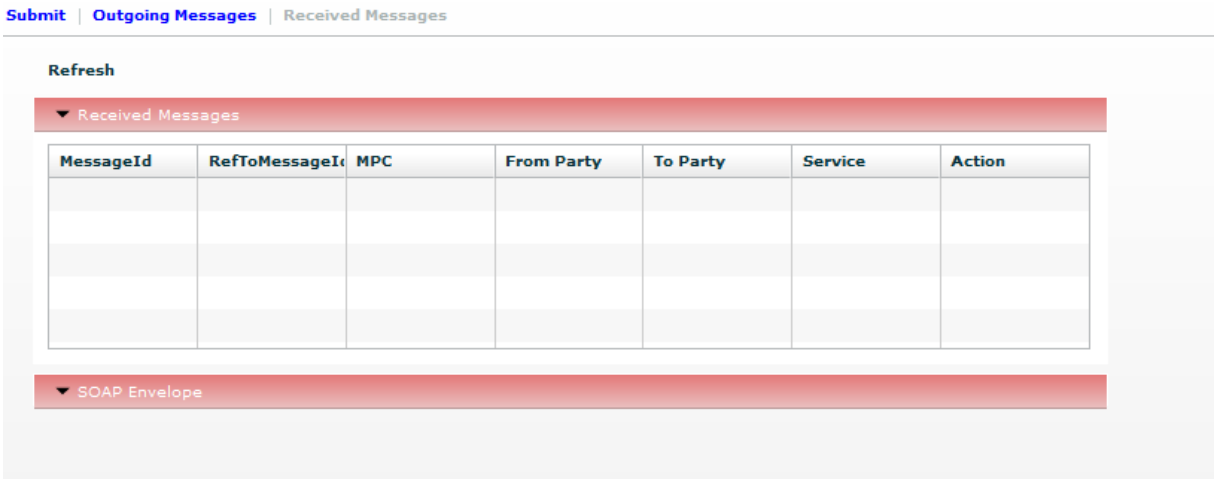


Figure 32 – Example for Logging GUI

4.3.2.4. Plugin

To some extent it is necessary to customize the gateway functionality to the national needs for example to check payment receipts and format transformation for the national system.

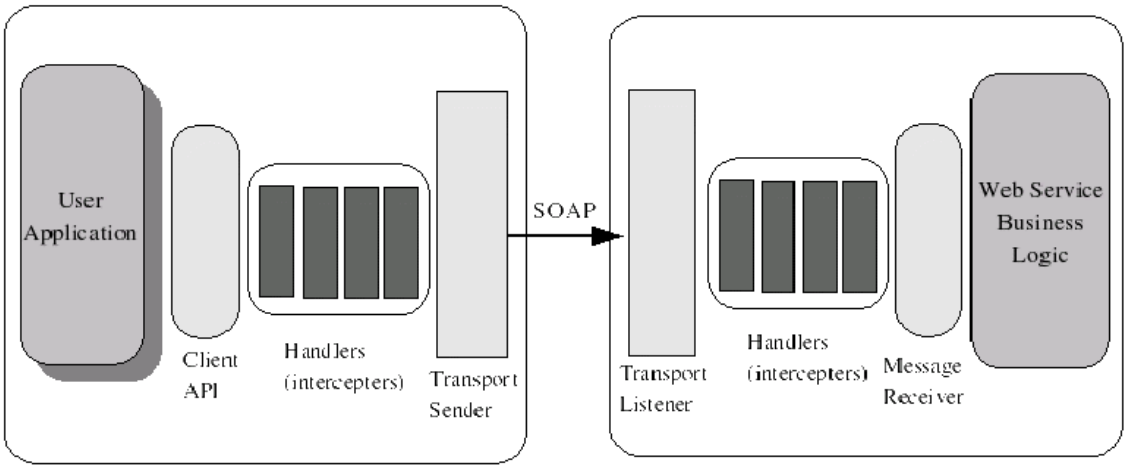


Figure 33 – Plugin overview

The plugin will be called directly within the message processing flow after the message has been received, decrypted and checked and before the message is transferred to the national backend system (Please refer also to Figure 30). Within the plugin the message can be checked, extended with additional national information or also manipulated due to national needs. Reasons for implementing a plugin could be:

- Checking of national routing information (e.g. search and transferring national electronic id's)
- Validating national rules
- Payment checks
- Checking certificates
- …..

Please note: e-CODEX only provides the possibility for such a plugin, but the implementation is in the responsibility of the MS, although the plugin itself has to be installed at the gateway as part of it.

4.3.2.4.1. Implementation

To provide a plugin the standard mechanism of AXIS is used. To be able to hook up with the message flow in both directions in and out a java class must be implemented derived from the AXIS AbstractHandler class:

```
*/
public class ReceivedUserMsgHandler extends AbstractHandler
{

 public InvocationResponse invoke(MessageContext msgCtx) throws AxisFault
 {
  ……..
  return InvocationResponse.CONTINUE;
 }
}
```

To manipulate the message the method *invoke* has to be implemented. From the parameter msgCtx there is full access to the whole message data. After writing the handler it must be configured within the module.xml file and packed into a Module Archive.

```
<InFlow>
  ……
  <handler name="ReceivedUserMsg" class="org.holodeck.ebms3.handlers.ReceivedUserMsgHandler">
    <order phase="ebms3InPhase" after="RespPackager"/>
  </handler>
  ……..
</InFlow>
```

For more detailed information please refer to the Axis 2 description.[75]

4.3.2.4.2. Installation

For installation only the module Archive (*mar File) containing the implemented Handler code must be installed in the Library path of the Gateway and the Gateway needs to be restarted.

4.3.2.5. Evidences

The Gateway will be able to handle and generate ETSI REM Evidences as defined within the Standard. For details about the ETSI REM standard please refer to http://www.etsi.org/deliver/etsi_ts/102600_102699/10264004/02.01.01_60/ts_10264004v020101p.pdf

For a detailed description of all supported evidences please refer to chapter 3.5.

In general the e-CODEX gateway generates evidences synchronously depending on messages and their status in the log DB. This evidence messages are transported within a normal ebMS message. The structure of such a message will be a XML fragment described in chapter 0.

Which evidences are supported can be defined per proceeding in a separate policy file. A detailed description can be found in chapter 4.4.4.5.

4.3.2.6. Report

The data basis for the reports is the logging data. It is essential for the administrators to know the utilization of the Gateway and how many erroneous and not successful messages have been sent or received to find and solve any runtime problems. Additionally it is also essential from a legal point of view to get reports for auditing purposes and to prove message transfers.

For Reports a command line tooling will be provided which access the Logging DB and generates the following Reports:

- ERR: Provides the list of all erroneous messages within a specific time range
- ALL: Provides the list of all messages within a specific time range
- SUM: Provides only the summary of the messages not the message details within a specific time range

The format of the report files will be csv, which can be imported easily into Microsoft Excel for further processing.

The tooling can be executed by the administrators via the command line:

```
generateReport  <TypeOfReport> <Date_From>  <Date_To> <location> <filename>
```

Example

```
generateReport  ERR 01-01-2012 01-02-2012 /home/reports example_reports
```

[75] http://axis.apache.org/axis2/java/core/docs/modules.html

4.3.2.7. Alarming

In case of erroneous conditions an alarm should be raised to inform the national administrators, who are responsible for the national gateway. The following main errors will be alarmed:

- DB not reachable or full
- File system full
- Web Service not reachable
- Unexpected error in a log file

For monitoring the above mentioned resources, scripts will be provided which can be integrated within a monitoring tool like e.g. Nagios. The scripts will provide the specific UNIX exit codes to indicate the monitoring tool the existence or the clearance of an alarm.

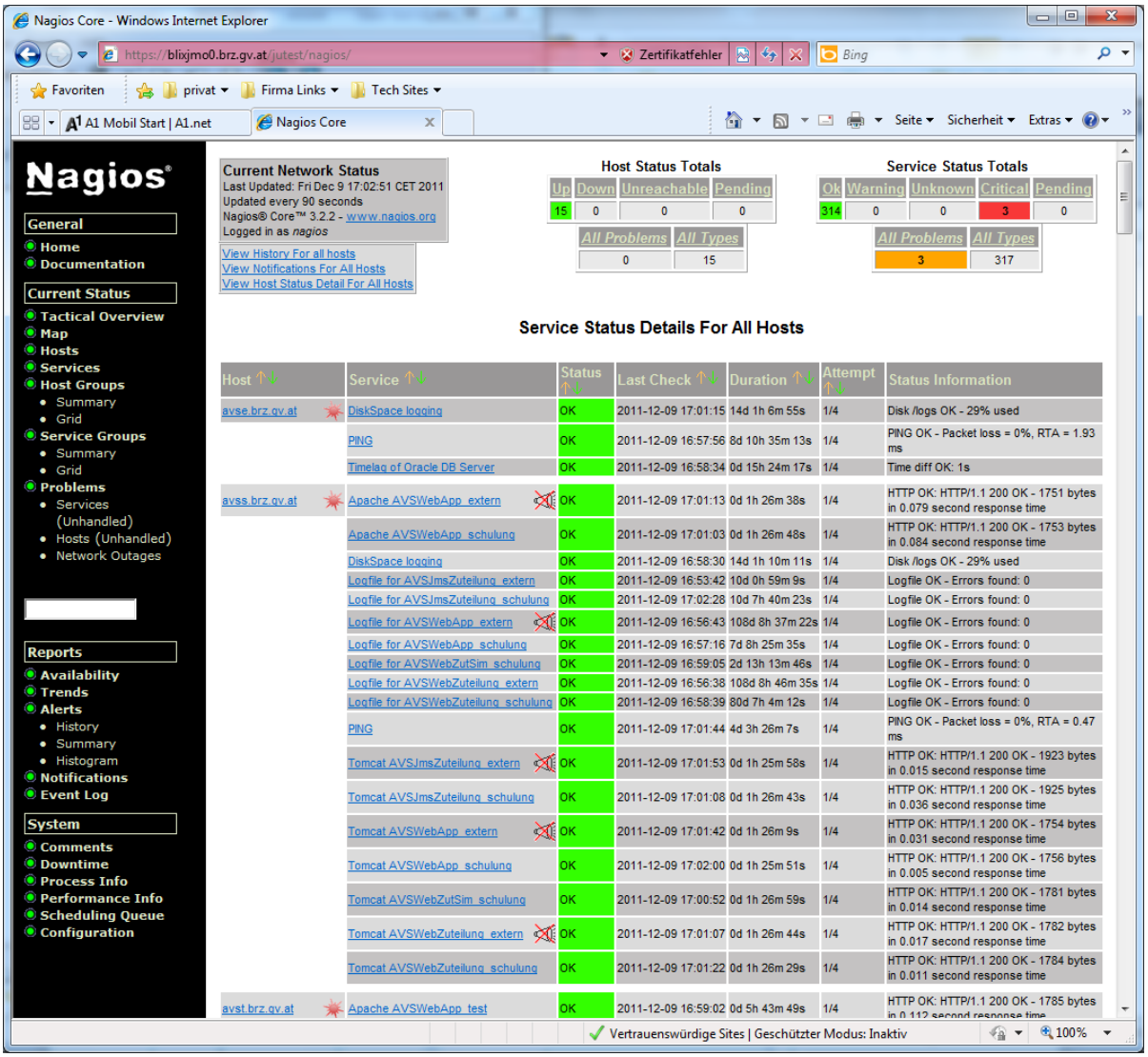


Figure 34 – Example for monitoring GUI (NAGIOS)

Such tools are used by administrators for monitoring and forwarding the alarms via a mail. Please note for the first release the e-CODEX Gateway will not support SNMP Traps.

It is in the responsibility of the national administrator to solve the alarms, because otherwise the gateway will not be able to receive or send messages.

4.4. Needed National adaptations and configurations of the standard gateway

In this chapter all the necessary tasks are described which are needed to be done by the Member States on top of the standard e-CODEX Gateway.

4.4.1. Backend Integration

There are different possibilities available at the gateway to integrate the national subsystems. The MS can decide whether they want to have a tight coupling with the gateway by implementing some specific Worker classes) or having a very loose coupling by implementing a Web Server and having an additional process in between the e-CODEX Gateway and the national subsystem.

Please note: Depending on the different type of Integration the possibilities for providing specific evidences could be restricted!

In general the following possibilities are available.

- File based
- Direct by implementing Handler and Worker class
- Web Service based
- *Optionally also a JMS based integration has been started by Holodeck but it is not finished and available yet.*

4.4.1.1. MS specific Handler and Worker class

If the MS decide to have a very tight coupling between the e-CODEX gateway and the national subsystem it can integrate the call of the national subsystems as well as the content transformation directly within the gateway. The following example gives an overview what has been done to implement the File based integration.

To send messages from the partner gateways to the national subsystems a so called ***EbConsumer*** Interface has to be implemented.

```
package org.holodeck.ebms3.consumers;

import org.apache.axis2.context.MessageContext;
import org.holodeck.ebms3.module.MsgInfo;
import java.util.*;

/**
 *  This interface is implemented by any type of consumer party behind an MSH.
 *
 * @author Hamid Ben Malek
 */
public interface EbConsumer
{
  public void push(MsgInfo msgInfo, MessageContext outMsgCtx);
  public void pull(MsgInfo msgInfo, MessageContext outMsgCtx);
  public void setParameters(Map<String, String> parameters);
}
```

The method push is called by the gateway whenever a message arrives. Within this method MS specific code can be executed to forward the message to the national subsystems.

To receive messages from the national subsystems and forward them to the partner gateways the messages must be stored in a DB. This is done by a helper class ***org.holodeck.ebms3.module.Constants*** and the call

```
Constants.store.save(msg);
```

Msg is a ***org.holodeck.ebms3.persistent.UserMsgToPush*** class and represents the whole message including metadata and attachments.

4.4.1.2. File based

It is possible to define a dedicated directory in which the gateway persist the incoming messages and another one which is polled periodically for new files to be sent out. This can be done by setting a configuration parameter within the consumer configuration of the gateway.xml

```
<consumption>
  <consumer className="org.holodeck.ebms3.consumers.impl.SaveToFolder">
   <!-- a parameter to indicate to the consumer application "ConsumerTest"
       where to save the attachment in the request. The value could be
       either an absolute path or relative to the "Received_Messages_Folder"
       (directory where the MSH saves attachments of the received messages) -->
   <parameter name="directory">Messages</parameter>
  </consumer>
 </consumption>
```

The ebMS message to be sent has to be provided already in a file with the correct format. Within the ebMS XML File also the additional end user identification within the message properties has to be provided. An example of the format can be found in chapter 3.3.3.1.

```
<Metadata>
  <pmode>Gateway_NL</pmode>
  <ConversationId>123456</ConversationId>
  <From>
    <PartyId type="name">Gateway_AT</PartyId>
  </From>
  <Properties>
    <Property name=”FromPartyId ">EVR.national.Id</Property>
    <Property name=”ToPartyId ">NL.national.Id</Property>
  </Properties>
  <Payloads>
    <Payload>payload.xml</Payload>
  </Payloads>
</Metadata>
```

Any additional payload is provided as an additional file.

There is no possibility to provide the evidences concerning the successful delivery to the end-users inbox.

4.4.1.3. Web Service

If the MS chooses to use the Backend Integration via Web Service the following steps has to be done.

(1) Providing a Web Service implementing the WSDL described in chapter 4.3.2.2 to be able to retrieve messages from the e-CODEX Gateway.
(2) Extend the national subsystem to call the Web Service described in chapter 4.3.2.2 in order to forward a message to the e-CODEX Gateway.

The end user identification will be provided by the WSDL API directly. The Web Service must be able to support the delivery of the information that a message has been forwarded to the end-users inbox. (getState method)

4.4.2. End User Authentication and Signature validation

The signature validation and the generation of the Trust Ok Token will be done outside the gateway on the national side with the help of a module or library provided by WP4. So this is transparent for the gateway as the Trust Ok Token is just an additional document as part of the message provided by the national systems.

The same is valid for the end user authentication in form of a SAML Token. A proposal for such a SAML Token can be found in chapter 3.3.3.4.2. It has to be mentioned that the provisioning of an SAML Token is not necessary in e-CODEX and therefore not part of the piloting phase.

4.4.3. Content mapping

The XML format of the forms for the e-CODEX pilots is defined by WP6. A mapping of this data format towards any existing MS specific data formats is right now out of scope for the gateway. If a MS wants to do this mapping inside the gateway then it can install a plugin like it is described in chapter 4.3.2.4.

Please note that any mapping must be done in an additional XML document attached to the original ones. A change of an existing XML form is not possible, because then the signature would be destroyed.

4.4.4. Configuration and Administration

Holodeck is implemented as a set of modules of Axis2, e.g., ebMS modules, reliability module, security module. The configuration of these modules is done by configuration files which are loaded at the start-up and they are located in the \config folder of the gateway.

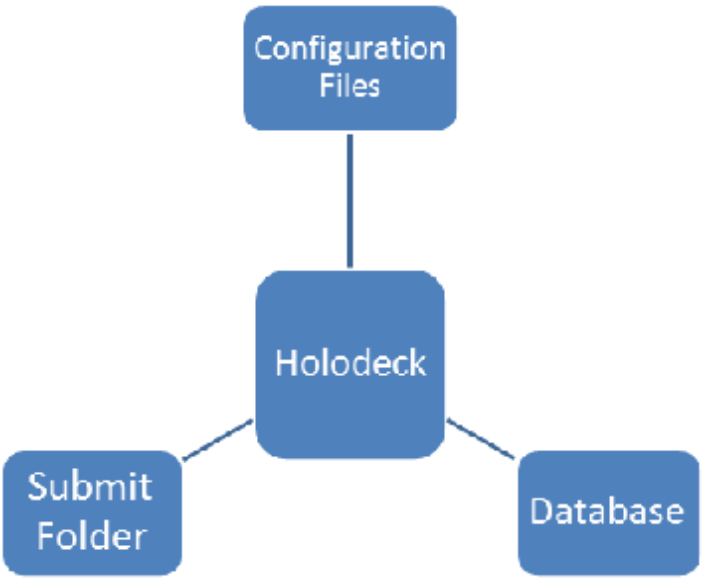


Figure 35 – External structure of Holodeck[76]

[76] http://holodeck-b2b.sourceforge.net/docs/images/picture2-a.png

The Pickup folder or Submit folder is located at \store\send. In this folder the payloads can be dropped so that Holodeck can pick them up and store them in the database. The database is only being used to store outgoing messages.

4.4.4.1. The Gateway component

The configuration file of the gateway module is called *gateway.xml*. In this file you can find the various consumers with their filters. This filter tells which consumer meets the criteria and who he can route the message to. If no filter is defined the consumer is capable of receiving all messages.

The following picture shows part of the content of the gateway.xml.

```
<consumption>
  <consumer className="org.holodeck.ebms3.consumers.impl.SaveToFolder">
    //------------------------------------------------------------------------
        a parameter to indicate to the consumer application "ConsumerTest"
        where to save the attachment in the request. The value could be
        either an absolute path or relative to the "Received_Messages_Folder"
        (directory where the MSH saves attachments of the received messages)
    //------------------------------------------------------------------------
    <parameter name="directory">Messages</parameter>
  </consumer>
</consumption>
<consumption>
  <consumer className="org.holodeck.ebms3.consumers.impl.SaveToTopic">
    <parameter name="destination">HOLODECK</parameter>
    <parameter name="url">tcp://localhost:61616</parameter>
  </consumer>
</consumption>
<consumption>
```

Figure 36 – The consumer segment of gateway.xml file

Each consumer object is declared by an xml entry <consumption>. The definition of a consumer contains an optional element called <filter>

```
<filter>
  <service>Shipment</service>
</filter>
```

Figure 37 – The filter element of gateway.xml file

In this example, the consumer has only one filter and wants to consume only messages that have SOAP header <eb:Service> with a value equals to "Shipment".

If a consumer wants to consume messages in which the value of the SOAP header "eb:Messaging/eb:UserMessage/eb:CollaborationInfo/eb:Action" is "NewPurchaseOrder" and the value of the SOAP header eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:From/eb:PartyId is "PartnerID", then the filter would be defined as the following:

```
<filter>
  <fromParties>
    <party type="">ParnerID</party>
  </fromParties>
  <action>NewPurchaseOrder</service>
</filter>
```

Figure 38 – The filter element of gateway.xml file

4.4.4.2. P-Modes

The instances of Holodeck exchanging messages need to agree on a set of P-Mode documents. So one P-Mode document is agreed upon and exchanged so that the instances of Holodeck have the same P-Mode document. When a message is submitted it must tell Holodeck the name of the P-Mode it is associated with. P-Modes are located in a folder under /config/pmodes.

```
<PModes>
    // ---------------- Define many producers -------------------
  <Producer name="name-it-anything">
  </Producer>
  <Producer name="second-producer">
  </Producer>
  etc...
  // ---------------- Define many UserServices -----------------
  // ------------ first User Service ------------
  <UserService name="name-it-something">
  </UserService>
  // ------------ second User Service -----------
  <UserService name="secondService">
  </UserService>
  // ------------ third User Service ------------
  <UserService name="thirdService">
  </UserService>
 etc...
  // ---------------- Define many bindings ----------------------
  // ------------ first binding -----------------
  <Binding name="firstBinding">
  </Binding>
  // ------------ second binding ----------------
  <Binding name="secondBinding">
  </Binding>
 etc...
  // ---------------- Define many pmodes ------------------------
  // ------------ first PMode -------------------
  <PMode name="firstPMode">
  </PMode>
  // ------------ second PMode ------------------
  <PMode name="secondPMode">
  </PMode>
</PModes>
```

Figure 39 – Example of a P-Mode Document

4.4.4.3. Reliability

Reliability is configured in the *reliability-config.xml* file. References to reliability are made in the P-Mode documents. In this example that you have a reliability service called "Order-Callback".

```
<Reliability name="ORDER-CALLBACK">
  <AtMostOnce>true </AtMostOnce>
  <AtLeastOnce>true </AtLeastOnce>
  <InOrder>false </InOrder>
  <AckReply ackTo="http://localhost:8080/holodeck/services/wsrm">Callback </AckReply>
  <RetransmissionInterval>40000 </RetransmissionInterval>
  <ExponentialBackoff>false </ExponentialBackoff>
  <MaximumRetransmissionCount>5 </MaximumRetransmissionCount>
</Reliability>
```

Figure 40 – The reliability-config.xml file

The reliability service is then specified in the element <Leg> of a P-Mode document as an attribute called "reliability". To use the service "Order-Callback" within a P-Mode definition, you choose which transport leg you want to use as it is shown in the following example.

```
<Binding name="uploadBinding">
  <MEP name="One-Way/Push">
    <Leg number="1" mpc="UploadMPC" userService="ConsumerTest"
      soapAction="upload" reliability="ORDER-CALLBACK">
      <Endpoint address="http://localhost:8080/holodeck/services/msh"
        soapVersion="1.1" />
      <As4Receipt>Response </As4Receipt>
    </Leg>
  </MEP>
</Binding>
```

Figure 41 – A segment of a P-Mode document

4.4.4.4. Reporting

The policy which reports are available is in the *report-config.xml* file. References to reports are made in the P-Mode documents. In this example you have a report profile called "daily-error-report".

```
<report-policy name="daily-error-report">
    <Period>daily</Period>
    <MessageTypes>error</MessageTypes>
    <FileName>./report</FileName>
    <ReportType>detail</ReportType>
</report-policy>
```

Figure 42 – Reporting configuration

The reports itself are stored on the file system under the name defined in the tag <FileName>.

4.4.4.5. Evidences

The possible policies which evidences are supported will be configured in the *evidence-config.xml* file. References to evidences are made in the P-Mode documents. In this example you have an evidence profile called "basic-evidence-support ".

```
<evidence-policy name="basic-evidence-support">
    <SubmissionAcceptanceRejection>true</SubmissionAcceptanceRejection>
    <RelayREMMDFailure>true</RelayREMMDFailure>
    <DeliveryNonDeliveryToRecipient>false</DeliveryNonDeliveryToRecipient>
    <RetrievalNonRetrievalByRecipient>false</RetrievalNonRetrievalByRecipient>
    <AcceptanceRejectionByRecipient>false</AcceptanceRejectionByRecipient>
    <DownloadNonDownloadByRecipient>false</DownloadNonDownloadByRecipient>
    <RelayToNonREMSystem>true</RelayToNonREMSystem>
    <ReceivedFromNonREMSystem>true</ReceivedFromNonREMSystem>
</evidence-policy>
```

Figure 43 – Evidence profile

By defining different evidence profiles the difference between a secure e-Delivery system and a registered e-Delivery system can be modelled.

The evidence level is then specified in the element <Leg> of a P-Mode document as an attribute called "evidence" and therefore bound to a P-Mode. To use the policy "basic-evidence-support" within a P-Mode definition, you choose which transport leg you want to use it as is shown in the following example.

```
<Binding name="binding2">
  <MEP name="One-Way/Push">
    <Leg number="1" mpc="shipment" userService="Customer" producer="Supplier" reliability="CALLBACK" security="sign-encrypt-body-header"
      evidence-policy="basic-evidence-support">
      <Endpoint address="http://192.168.1.7:8085/holodeck/services/msh" soapVersion="1.1" />
    </Leg>
  </MEP>
</Binding>
```

Figure 44 – Evidence configuration within the P-Mode

4.4.4.6. Security

Security profiles are configured in the *security-config.xml* file. References to security are made in the P-Mode documents. In this example that you have a security service called "sign-encrypt-body-header ".

```
<security name="sign-encrypt-body-header">
  <policy>policy-sign-body-header.xml</policy>
  <initiator>
    <user callbackClass="org.holodeck.security.sample.PWCBHandler">client</user>
    <encryptionUser>service</encryptionUser>
    <keystore type="JKS" password="apache">client.jks</keystore>
  </initiator>
  <recipient>
    <user callbackClass="org.holodeck.security.sample.PWCBHandler">service</user>
    <encryptionUser>client</encryptionUser>
    <keystore type="JKS" password="apache">service.jks</keystore>
  </recipient>
</security>
```

Figure 45 – Security configuration

The security service is then specified in the element <Leg> of a P-Mode document as an attribute called "security". To use the service "sign-encrypt-body-header" within a P-Mode definition, you choose which transport leg you want to use it as is shown in the following example.

```
<Binding name="binding2">
  <MEP name="One-Way/Push">
    <Leg number="1" mpc="shipment" userService="Customer" producer="Supplier" reliability="CALLBACK" security="sign-encrypt-body-header">
      <Endpoint address="http://192.168.1.7:8085/holodeck/services/msh" soapVersion="1.1" />
    </Leg>
  </MEP>
</Binding>
```

Figure 46 – Security configuration within P-Mode

4.4.4.7. Database

For the database different configurations are provided out of the box defined in the file *persistence.xml*.

Within this file the different DB connection parameters such as URL, password and driver class are defined.

```
<persistence-unit name="ebms3-mysql" transaction-type="RESOURCE_LOCAL">
 <provider>org.hibernate.ejb.HibernatePersistence</provider>

 <properties>
  <property name="hibernate.archive.autodetection" value="class, hbm" />
  <property name="hibernate.show_sql" value="false" />
  <property name="hibernate.format_sql" value="true" />
  <property name="hibernate.connection.driver_class" value="com.mysql.jdbc.Driver" />
  <property name="hibernate.connection.url" value="jdbc:mysql://localhost:3306/ebms3" />
  <property name="hibernate.connection.username" value="ebms3" />
  <property name="hibernate.connection.password" value="ebms3" />
  <property name="hibernate.dialect" value="org.hibernate.dialect.MySQLDialect" />

  <property name="hibernate.c3p0.min_size" value="5"/>
  <property name="hibernate.c3p0.max_size" value="20"/>
  <property name="hibernate.c3p0.timeout" value="300"/>
  <property name="hibernate.c3p0.max_statements" value="500"/>
  <property name="hibernate.c3p0.idle_test_period" value="3000"/>
  <property name="hibernate.c3p0.maxStatementsPerConnection" value="500"/>
 </properties>
</persistence-unit>
```

Which DB Configuration will be used is defined in the module file defining the corresponding module file *module.xml*.

```
<module name="holodeck-ebms3" class="org.holodeck.ebms3.module.Ebms3Module">
 <parameter name="PersistenceUnit">ebms3-derby</parameter>
 ……..
```

The database is used within the gateway for the storage of configuration and log data, as well as for implementation of the reliant messaging.

5. e-Payment

There are some MS where, according to their national legal requirements, payment of court fees is mandatory to start a judicial process. Payment of court fees is closely related to each MS legal requirement and the type of payment evidences that are accepted by the court. From a legal point of view, not all the Member States need payment in their judicial context, in fact not all the regulations[77] associated to e-CODEX pilots include payment as a requirement, e.g. Small Claims.

WP5 focuses only on the payment of court fees and not on the transfer of money for penalties derived from the decisions for a proceeding[78]. National laws, rules and derived contracts with financial entities form a scenario that is not much flexible as to allow normalising e-Payment among countries easily. Moreover, although banks play an important role in the payment process, involving banks in the solution would be too complex and should therefore be out of scope of the e-CODEX project. Furthermore, only a few countries require payment of court fees, so e-Payment must be an optional feature for cross-border filing. If a Member State does not require any payment of court fees it should not be forced to implement it. Of course, the e-Payment solution defined by e-CODEX should be suitable for all the Member States, covering all their basic needs and restrictions.

Besides, a payment receipt must be recognized and accepted by the destination judge. This is the capital reason for using the appropriate e-Payment national solution. These national solutions obviously take into account the national legal requirements for e-Payment. For use cases in which payment of court fees is required the user must be re-directed to the appropriate e-Payment national solution of each Member State. Even for future use cases the legal settings in most national countries should be observed, as they define and establish the solutions that are recognized as valid to execute payment.

If a central reference is to be considered, a suitable place to be published would be the e-Justice portal. This website would provide users with references to the appropriate information about the payment process for the country in which he is filing the claim and links to the e-Payment solutions.

Definitions on e-Payment[79] settled by the Legal and Security Subgroup are:

- The user should have the **instructions about how to pay/when to pay court fees.** e-CODEX should provide access to the instructions/user guidance regarding how the payment is to be made. This information is to be provided by the MS calling for the payment of court fees.

  The translation of documents related to e-Payment is up to the countries, translation is possible but not necessary. The documents are in the language of the competent court because this is part of the judicial procedure[80]
- For the payment of court fees (when needed): **the user is offered a link to the e-Payment national solution** provided by the receiving MS[81]. There should also be an electronic receipt.

[77] See REGULATION (EC) No 861/2007 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 11 July 2007 establishing a European Small Claims Procedure

[78] agreed in the WP5 meeting held in Madrid on May 2011

[79] settled in the meeting on key legal issues held in Düsseldorf on 30th September 2011

[80] decision made in e-CODEX Management Board, 2-3 November 2011 – Berlin, Germany

[81] See statement in D5.2, at the end of § 7.1

e-CODEX does not support the payment itself but the attachment of the payment evidence will be supported, according to receiving country's rules.

- For the use cases (i.e. Small Claims and EPO) where the form permits the user to include the bank information for the payment of fees, there should be a **check box** stating that the claimant accepts the fact that the court can request for the payment to be done using the claimant's bank details.
- At least it should be possible to send the e-Payment evidence as an attachment or the e-Payment id as a reference. If this is to be included in the form, WP6 have to consider if a place for the e-Payment id and evidence must be provided.

5.1. e-Payment overview

The functionalities to be offered to the users are:

- information on when payment of court fees is required, how much should be paid and how to proceed;
- a way to access the online e-Payment solutions for executing the payment and
- means to handle the corresponding payment receipts.

The e-CODEX project aims to develop the interoperability building blocks for e-Justice services in Europe that address the horizontal issues between Member States. The e-Payment Building Block (e-Payment BB), which could be subdivided into smaller subBBs, has been identified in D5.2. The e-Payment BB will implement the use case relevant for e-CODEX WP5 handling court fees, which have to be paid before a proceeding can be initiated. It has to be mentioned that there is another use case, which handles the payment of penalties between the claimant and the opponent. It has been agreed that this use case is out of scope of e-CODEX and WP5.

e-Payment BB includes the handling and sending of e-Payment receipts together with the message as an additional business document proving a successful payment of the respective court fees. Due to the fact that this payment receipt is an additional business document and that the destination court will only accept recognized receipts, it must be delivered 1:1 as received from the payment service.

The receipt can be an attachment (pdf, xml) or a number (pre-payment). A structured way to transport these kinds of payment receipts will be considered. As a minimum requirement this payment envelopes must include the associated case ID or reference and the amount of money paid by the applicant. If the case ID uniquely references the applicant then the applicant does not need to be included.

The online payment itself via direct bank or credit card interfaces is out of scope for WP5. The reason for that is that normally, in every country, well defined and widely used electronic payment applications are already in place, which should be used further on. The e-Payment BB here only initiates the payment by calling the correct national e-Payment interface and receives a receipt for the successful/unsuccessful payment.

In case of online pre-payment of court fees, there is a need for cross-referencing the case data to be submitted with the payment:

- The case data, which is to be submitted to the court, has to include the , evidence to prove the successful payment of the correct amount of court fees in the way accepted by the court.

- The payment data must include a reference to the (preliminary) case or document number to be used by the national accounting system for transferring the payment to the right court *(e.g.: some countries use a unique document number, pre-printed on forms to be filled and submitted. You have to specify this document number on your payment of court fees, so the court can check, if the payment of court fees was received, before proceeding with the case)*.

Table below lists those MS where payment is mandatory to start a judicial process including some others and their situation.

Country	Mandatory	Solution available
AT	YES	YES
DE	YES	YES
ES	NO	Unnecessary
FR	YES	YES
IT	YES	YES
NL	TBD	Pilot
PT	YES	Order for payment procedure

Table 15: Inputs on Payment from the MS

5.2. e-Payment Business process

The steps that explain e-Payment are shown in the following picture:

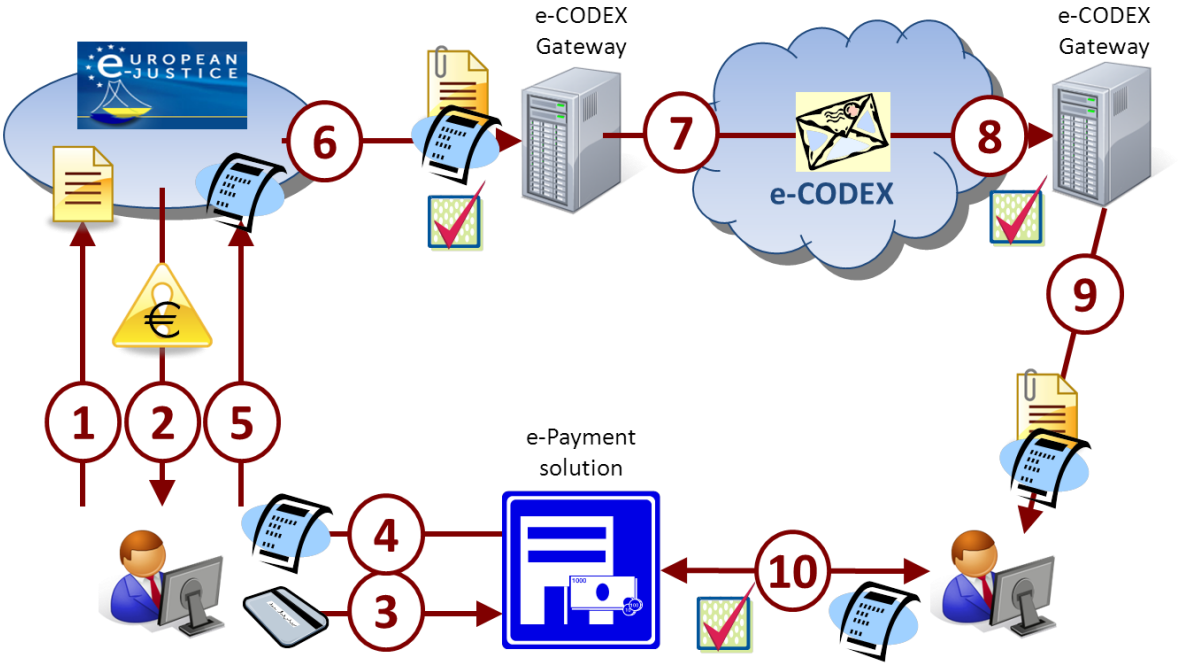


Figure 47 – e-Payment business process

1. The user asks to initiate a proceeding by accessing the appropriate form to be filled.
2. The user is advised that payment is required for that destination and is informed how to do the payment (information to be provided by the MS calling for the payment of court fees).
3. The user accesses the appropriate e-Payment solution and proceeds with the payment.
4. The user receives the payment receipt.
5. The user fills the form and attaches the payment receipt.
6. The e-CODEX gateway identifies the receiving MS. When payment is required, it is up to the sending country to check, in the 'national adapter'[82] of the gateway, that the appropriate receipt is attached.
7. The sending e-CODEX gateway sends the "e-CODEX message" through "e-CODEX infrastructure" to the receiving e-CODEX gateway of the addressed MS.
8. The message is delivered to the e-CODEX gateway of the receiving MS. When payment is required, it is up to the receiving country to check, in the 'national adapter' of the gateway, that the payment receipt is attached and is valid.
9. Finally the receiving e-CODEX gateway forwards the message to the addressed user.

[82] The 'national adapter' is the part of the gateway in charge of the particular functionalities required by the country or participant, i.e. adapting the received message to the national schema and transforming the national message to the e-CODEX message.

10. Receiver should be able to check the receipt with his respective e-Payment national solution (this is up to the MS's rules).

In this scenario it is necessary to provide an access point, mainly for citizens. Most countries only provide national solutions for legal professionals and judicial authorities, but not for citizens. Therefore, the European e-Justice Portal seems to be the best option to include information to the citizens about judicial procedures, and to offer forms (e.g. Small Claims and EPO) for initiating some of them. Currently, the e-Justice Portal, in its "Going to court" section there is a page called "Costs of proceedings"[83] where information about cost of proceedings is provided.

5.3. e-Payment specifications

Three different functionalities are to be considered for the description of the e-Payment Building Block:

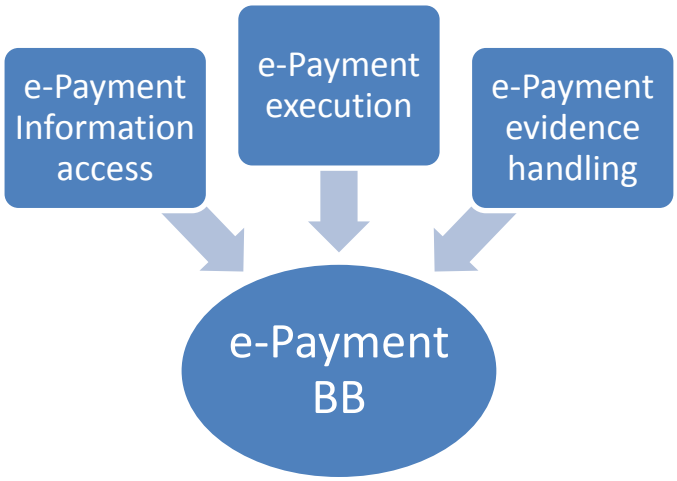


Figure 48 – e-payment scenarios

#BB Code: BBP001	**BB Title:** e-Payment information access
BB description: As the users will need to be aware of the basic aspects: regulation setting the court fees, which processes require payment of fees, the amount to be paid and the rules to calculate it, when should they be paid, what are the court requirements about evidences of payment, etc., this BB includes the necessary functionalities to access to e-Payment information already existing in order to know perfectly how, where, how much should be paid on a judicial procedure. Two scenarios are possible: - Citizens accessing to this information directly through the e-Justice Portal	

83 https://e-justice.europa.eu/content_costs_of_proceedings-37-eu-en.do

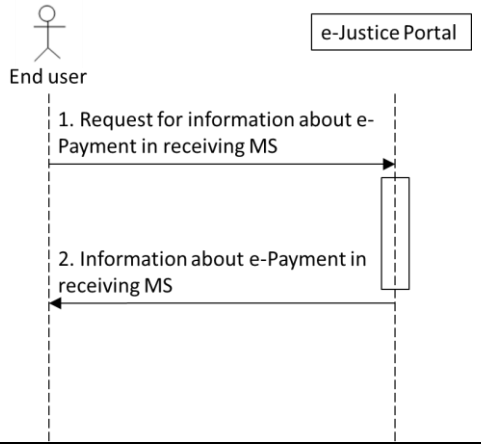
<table>
<tr><td colspan="3">(reusable existing BB).This is the scenario to be developed for e-CODEX.
- Judicial Professionals could have e-Payment information available at their own solutions offered by their service provider (these solutions could connect directly to e-Justice Portal). Developing this scenario is up to the national service providers.
NOTE: Having this information available in the languages of the European Union is up to the countries, translation is possible but not necessary.</td></tr>
<tr><td colspan="2">Other BB dependences:
Transportation
Directory
Administration</td><td>REQ Traceability:
WP5-RQ-F-004 e-Payment national solution redirection.
WP5-RQ-F-005 Interoperability with e-Justice portal</td></tr>
<tr><td rowspan="2">Information Data</td><td colspan="2">Inputs:
Addressee MS</td></tr>
<tr><td colspan="2">Outputs:
Detailed information about e-Payment process, depending on each Member State requirements. (based on the information provided by each MS)</td></tr>
<tr><td>BB Technical Description Functionality</td><td colspan="2"> 
</td></tr>
</table>

Table 16: e-Payment Information

<table>
<tr><td colspan="2">#BB Code: BBP002</td><td>BB Title: e-Payment execution</td></tr>
<tr><td colspan="3">BB description:
This BB provides access to the appropriate e-Payment solution by redirecting to the corresponding national e-Payment service provider where the user is able to pay and obtain the payment receipt.</td></tr>
<tr><td colspan="2">Other BB dependences:
Transportation
Directory
Administration</td><td>REQ Traceability:
WP5-RQ-F-004 e-Payment national solution redirection.
WP5-RQ-F-005 Interoperability with e-Justice portal</td></tr>
<tr><td>Information</td><td colspan="2">Inputs:
- Addressee MS,</td></tr>
</table>

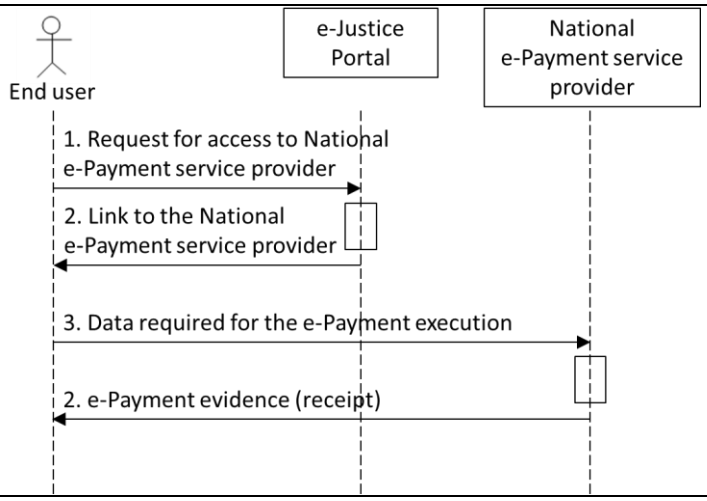
<table>
<tr><td></td><td>- data required for the payment execution (applicant id, case id, court, amount of money to be paid …),</td></tr>
<tr><td></td><td>Outputs:
- Valid e-Payment evidence. Receipt from the national e-Payment service provider complying with the requirements to be valid in front of the court according to the receiving country’s rules.</td></tr>
<tr><td>BB Technical Description Functionality</td><td> 
</td></tr>
</table>

Table 17: e-Payment execution

<table>
<tr><td colspan="2">#BB Code: BBP003</td><td>BB Title: e-Payment evidence handling</td></tr>
<tr><td colspan="3">BB description:
This BB includes the handling and sending of the payment evidence (receipt) together with the message as an additional business document proving a successful payment of court fees.
This evidence (the receipt of the transaction) will be transported as any other data or document attached to the appropriate process form.</td></tr>
<tr><td colspan="2">Other BB dependences:
Transportation
Directory
Administration</td><td>REQ Traceability:
WP5-RQ-F-004 e-Payment national solution redirection.
WP5-RQ-F-005 Interoperability with e-Justice portal</td></tr>
<tr><td>Information Data</td><td colspan="2">Inputs:
Required payment information: Proceeding, case id, value of the claim, court/MS responsible for.
Valid e-Payment evidence from the appropriate national e-Payment service provider with the information required according to the receiving MS legal regulations.</td></tr>
<tr><td></td><td colspan="2">Outputs:
Message containing the payment evidence attached to the appropriate form or document to be sent to the appropriate destination.</td></tr>
</table>

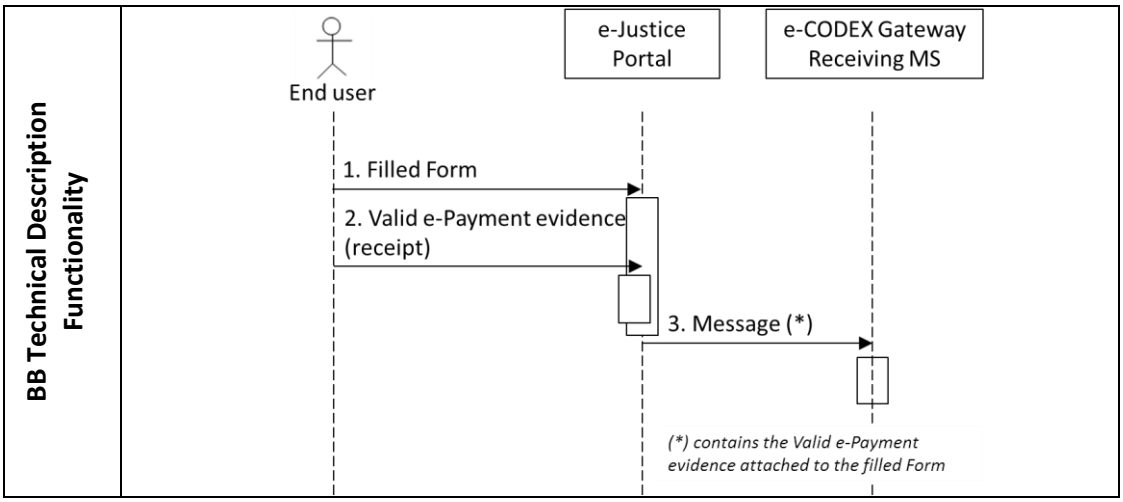


Table 18: e-Payment receipt handling

5.3.1. National e-Payment parameters

The e-Payment parameters provided by the listed MS are included in following table. The enumerated information means:

- **e-Payment user information**: for this document it is enough with the reference to this information, the information itself should be available before starting the pilots. The information to be offered to the users should be enough for the user to be aware of the basic points: regulation setting the court fees, which processes require payment of fees, the amount to be paid and the rules to calculate it, when should they be paid, what are the court requirements about evidences of payment, etc.
- **e-Payment solution**: the reference to be used by a citizen to access to the e-Payment application (electronic address, URL, applet). It is advisable to provide citizen with guidance on the access and use of the national e-Payment solution, i.e. a user manual.
- **Means of payment**: means like credit card, bank transfer, direct debit, voucher, cash, etc.
- **Moment of payment**: pre-paid, pay-now, pay-later, etc.
- **Type of Receipt**: electronic stamp, bar code, pdf-receipt, XML, URL, etc.

This is the information to be included in the European e-Justice portal for the user to have a clear view of the procedure and national details before proceeding whit the payment of the required court fees. For this purpose, a template is being defined together with the Directorate General for Justice (DGJUST) of the European Commission ensuring that MS will provide the information in a homogeneous way.

All e-CODEX participants must fill this template providing their information about:

- payment of court fees is to be included (even if they do not have court fees; if this is the case it would be enough for them to indicate it at the start of the template)
- electronic payment of court fees must be included by those participants providing this option

This template is included as an Appendix IV – European e-Justice portal template for payment information from MS in this document.

By the time updating this section the template is still to be discussed with the DGJUST

MS	e-Payment user information	e-Payment solution	Means of payment	Moment of payment	Type of Receipt
AT[84]	User is only asked to provide IBAN, BIC and account holder	No e-Payment solution needed. Court fees are charged by direct debit	**Currently:** National direct debit transaction **Future:** SEPA cross-border direct debit transaction	Paid after filing	Not necessary, court fees are stated in the court decision
DE	http://www.justiz.nrw.de/JM/online_verfahren_projekte/projekte_fuer_partner_der_justiz/elektronische_kostenmarke/index.php	"Elektronische Kostenmarke" (http://www.kostenmarke.justiz.de/)	via bank transfer or credit card	Pre-Paid	Barcode and pdf-receipt as well as a unique ID, which can alternatively be provided as proof of payment
ES	Not necessary	Not necessary	Not necessary		Not necessary

[84] **Austria uses direct debit transactions to collect court fees for cases filed electronically:**

For all electronically filed cases Austria requires, that the claimant (or the lawyer/representative filing the claim) has to specify IBAN and BIC of his bank account number in addition to the case data of the claim. The back office application of the Austrian courts then calculates the required court fees automatically and withdraws the required amount for court fees from the specified bank account of the claimant automatically via a direct debit transaction.

Austria has a back-up solution, if no account number was specified, or if a direct debit transaction is not possible:

The court can send either a "Request for Payment Letter" or an "Urgent Request for Payment Letter" (with the threat of taking enforcement actions) notifying the claimant to pay the right amount of court fees and specifying the account number of the court and reference information (the case number) for which to pay.

FR	https://www.timbre.justice.gouv.fr/pages/aide/textes.jsp[85]	https://www.timbre.justice.gouv.fr/pages/achat/choixTimbres.jsp	Visa, Mastercard	Pre-Paid	PDF files: - proof of purchase - number of the stamp and barcode
IT	To be provided (new National Portal available soon)	Italian system (new Italian National Portal available soon)	To be provided (when National Portal available: payment cards and money transfer (via bank and postal system))	Pre-Paid	electronic payment receipt (XML)
NL	To be provided	Solution in piloting phase (available soon)	To be defined	To be defined	To be defined
PT	https://igfij.mj.pt/CUSTAS/Paginas/Autoliquidacoes.aspx	Order for payment procedure – The court's system (Citius) | Generally – For generation of the payment code: https://igfij.mj.pt/CUSTAS/Paginas/Autoliquidacoes.aspx	Home banking / ATM / bank branches	Order for payment procedure – after the filing, prior to the treatment of the process | Generally – Prior to case submission to the court Injunctions	Receipt of the e-Payment system: DUC – document with the data to perform the payment | entity, number, value; Receipt of the payment: Generated outside the e-Payment system by an ATM or banking receipt

Table 19: e-Payment parameters provided by MS

5.4. e-Payment Considerations

5.4.1. Standards

This BB will be aligned to standards and guidelines established from WP7 for e-CODEX.

The information associated to a payment service provider transfer would be considered as any other kind of information and included in the message to be sent to court. XML is the standard being considered for the e-CODEX messages.

[85] This information will be soon available in English

5.4.2. Security

The same level and security requirements for e-Payment information as for any other kind of judicial data/information would be taken into account. Payment guarantees are those given by the payment process, e-CODEX will only guarantee the correct and secure handling of payment evidences.

5.4.3. Inputs on e-Payment from MS

Those Member States where e-Payment is mandatory made available significant information about their e-Payment national approach. It has been considered worthy to include this information as Appendix V – Information about e-Payment from MS.

6. Directory of judicial atlas

6.1. Introduction

One of the main requirements for communication is having a sending and a receiving party. The retrieval of the receiving party and its contact details is a challenge in itself. In many cases the sender might not even know whom to address in certain a proceeding. If that is not known, how would this person be able to use the correct parameters for addressing?

As an initiator of a proceeding you will have to have quite extensive knowledge of the proceeding, in order to be able to find your addressee. For each proceeding you might have to turn to a different court or judicial authority. Additionally, each Member State applies different business rules in determining what authority is handling the procedure. It is these business rules that have to be specified.

Since the former "Plan A", where competent authorities are retrieved on the European e-Justice Portal, has been declared out of scope by both the representatives of the European Commission and WP5 leaders, it is no longer part of this document[86]. The former "Plan B" which request competent authority information in the relevant Member State, is described in this paragraph. Still two sub scenarios are to be considered. Paragraph 6.3 describes the details of both scenarios.

In electronic messaging the address of the electronic business document differs from the geographical address of the processing employee. Section 6.2 explains these differences.

6.2. Electronic vs. traditional addressing

In the comparison between traditional postal delivery and the electronic delivery both similarities and differences are found.

The most important difference is that all the addressing information for an end-to-end delivery must be known before sending a physical letter. Traditional postal service does not add any intellect along the line. Postal services cannot interpret the envelope and certainly cannot route on (part of) the information which resides inside the envelope. In electronic messaging interpreting envelope information is possible and if chosen, content based routing is an option. ***Content based in this context is meant as content of a business document header (SBDH) or REM ETSI information. The e-CODEX infrastructure is not to open/read contents of the exchange of business process itself.***

A distinction must be made between old fashioned postal delivery on obtaining the address and putting this information on the envelope. Country, city, street and number are put on the envelope to enable the postman to deliver mail at the correct door. Behind that door, the employee who processes the mail is located. In order to send the letter to that specific employee, the address of his or her organisation must be put on the envelope. If this address is unknown one can do a look up in the yellow pages, phonebook or zip code book. Or if a telephone number is known, the sender can obtain the address information by making a call to the addressee. Once the correct address is obtained, it can be put on the envelope and handed over to the mailman (resembling scenarios 3 & 4).

[86] This is based on a common agreement between EC and e-CODEX because it is expected that the "new Court Atlas" is not in place when the pilot phase of e-CODEX starts. The "Plan B" solution can be "re-used" to integrate the "new Court Atlas" when is available via the European e-Justice portal.

An acknowledgement of receipt contains information on the receiving party of the initial message. This information is needed from a legal point of view. What you see here is that this information will come to you even if you didn't know exactly where you have sent the message in the first place. Normally the acknowledgement of receipt will return to the sender of the initial message containing the business information in a matter of seconds. If, at that moment the addressee information is stored in the information system of the sender of the initial business document, this organisation has all the legally needed information. This will be the acknowledgement of receipt provided by the infrastructure. A business acknowledgement stating that the request was accepted/denied by some clerk will be returned after processing the request. Depending on the electronic address the user provided in the form for 'return traffic' it will be returned to the European e-Justice portal, or to a nationally implemented secure electronic address of the user.

6.2.1. An example

An application for EPO is processed by a court situated in Rome. So the processing department must be addressed in the business document and is, from a legal point of view, essential. From a technical point of view, the addressing could be of a very different nature.

Let's assume that all courts in Italy use a nation-wide system which is administered by a service provider in Milan. In that case, the "electronic address" would be that of the service provider in Milan expressed in a URL, organisation code, etc.., but not as Street+number+postalcode+city and certainly not the geographical of electronic address of a specific court.

The system adds the content of the business document into the (national) case management system. Within this system it is added to the workflow of the employee or department of a specific court in Italy which processes the application.

In this particular example, it is likely that a judge of the processing court in Rome is not interested at all in the fact that the electronic message was picked at the Italian border by a gateway, from there deliver to a national service provider which hosts the Italian case management system and has allocated the contents of the message to the court of Rome.

The point is that the electronic address is of no value from a legal point of view regarding the business process. Likewise, the geographical (traditional mail) address is of no use for electronic delivery to the ultimate addressee. Address information that is needed from a business perspective resides inside the business documents or its attachments, the actual message. The electronic address resides in the message envelope or the message header.

In order to know how to address a specific court (or courts in general) by electronic means, the European e-Justice portal must provide the electronic address to put in the message header or on the message envelope for delivering purposes, as well as the geographical address to put inside the business document (xsd and/or pdf) for legal purposes. The functionality that must be provided either by the European e-Justice portal itself, or a webservice/business transaction initiated from the European e-Justice portal is that:

- A user enters the requested parameters for retrieval a competent authority
- A request is initiated by the e-Justice portal to the relevant Member State, providing the entered parameters
- The Member States responds with both the geographical and the electronic address. Preferably also information on court fees, payment methods, etc… is provided in this respond

- The European e-Justice portal puts the geographic address in the (web)form(s) of the application and temporarily stores the electronic address awaiting the sending of the form by the user. The user needn't sign for the electronic address for it is not part of the legal procedure or used for business purposes. The users signs for the content of it's message, not for the added attributes of technical/routing nature. It is put in an additional document (REM ETSI evidences, Standard Business Document Header, ebMS Envelope, etc…) for delivery purposes.

6.3. Possible scenarios

6.3.1. Approaches and responsibilities

Depending on how the retrieval of ultimate addressees is technically supported, the responsibilities of specifying the required parameters on the retrieval of the contact details of the ultimate addressee might vary. The required parameters will however be equal.

- If it is chosen to have a dedicated web-service for the retrieval of a competent authority, Work Package 5 will be responsible.
- In the case of considering the retrieval of a competent authority as just another Business Transaction for exchanging business information, Work Package 3 will have to perform the specifications of the required parameters and have them modelled into a XML scheme by Work Package 6.

In case of the latter, Work Package 5 will only be responsible for providing the channel and the routing information to deliver the message to the correct national gateway. Also future changes of the required parameters, due to altered or newly supported regulations, will be out of scope for WP5. Extension of parameters will be analysed and modelled by WP3* and translated to an adapted XML schema by WP6* (* = or its equivalent entity in a post e-CODEX situation).

6.3.1.1. Short term solution

The "short term plan B" provides a solution only for the piloting phase of the project and is not sustainable when the e-CODEX project has ceased to exist. It is however, by far the easiest way to implement on a short term notice.

In the piloting phase a restricted number of Member States will participate in a certain use case. Also the expected number of cases during the 1st piloting phase is limited. A suggestion therefore is made to allocate just 1 authority in a Member State processing this particular use case. The result would be a limited list of participants, with a limited list of authorities. This list can be "hard coded" into Collaboration Protocol Agreements, P-Modes or other routing/addressing document. Any Member State should be allowed to decide to implement the sustainable solution, mentioned in the next paragraph. Agreements on this issue are yet to be made, if desired at all.

No additional business rules will have to be applied and after a brief "stock taking" in the participating Member States, the pilot phase is "good to go". The downside is that after the piloting phase, a sustainable solution still has to be implemented, which than will not have been tested during the piloting phase. If not implemented in the 1st piloting phase, it is to be considered to implement this feature in the extension phase of the e-CODEX project. Either in the 1st piloting phase or in the extension phase the sustainable solution will have to be tested before implementing on a large scale during run-time phase. Also, if just one Member State decides to implement the

sustainable solution during piloting (and having agreed so with participants in the same use case), the solution will have to be proven by testing.

6.3.1.2. Sustainable solution

The long term proposal for finding ultimate addressees is a request to a Member State. Based on certain parameters the Member State (not necessarily the national gateway) responds to this request by providing details on the requested competent authority.

By smartly creating a generic Business Transaction or web-service all currently chosen use cases will be supported. Future developments can easily be incorporated in this transaction. The current use cases require only the retrieval of courts, but in future other (judicial) authorities can be found accordingly. (E.g. police departments, district attorneys, official agencies, etc.)

This proposal does require some effort from each of the Member States. They will somehow have to create a repository of authorities and their fields of competence/jurisdiction. This is an exercise Member States will have to perform anyway, because the European e-Justice portal requires this kind of information too. The web-service or ebMS transaction will address this repository.

In future developments of the European e-Justice portal itself, it might use this transaction to dynamically show competent authorities on their website. Deploying this service on the portal will reduce maintenance of this information. The task of keeping the information up-to-date lies in the hands of the respective Member States. The European e-Justice portal will be in the role of the "requestor" and the national responding repository as "provider" (see Figure 49 – Address resolution business transaction). In proceedings in the field of criminal law where two or more authorities communicate without using the e-Justice portal, the requesting authority can initiate the business transaction in the role of "requestor".

By creating such a business transaction and designing it in a generic approach and making it "role based" it will be deployable by the e-Justice portal and national authorities. In future it might even be deployed by notaries, lawyers and other entities, providing they have access to an e-CODEX gateway.

6.4. Specifying the required addressing parameters

Either in a web service or in a Business Transaction according to the ebXML standards, the dynamic retrieval of addressees consists of a requesting and a responding activity. A so called Q and A game. This Question and Answer game is visualized in the image below:

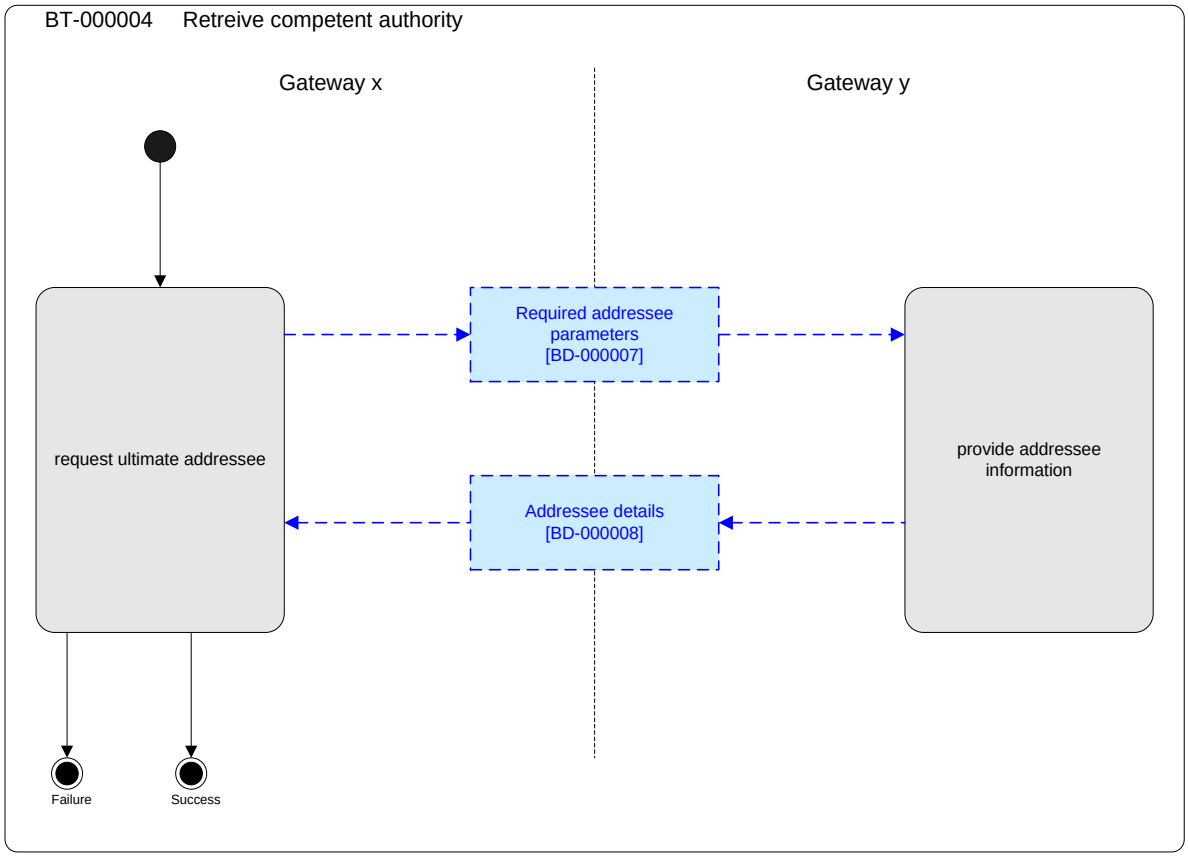


Figure 49 – Address resolution business transaction

1. The business activity on the requesting side (gateway x) is “request ultimate addressee”.
2. The output of this activity is the Business Document “Required Addressee Parameters”. All parameters relevant for the receiving party (gateway y) are provided.
3. The receiving party will perform a “look up” in its own judicial authority repository and…
4. …provides a positive (success) or a negative (failure) answer in the Business Document “Addressee details”. The negative answer could be that, based on the provided parameters no answer was found.
5. The contents of the answer are used to fill the required address fields in the forms of a proceeding and also used in the Business Document Header to enable the messaging infrastructure to route the message over multiple components to the correct (digital) addressee. This could be in the form of a Standard Business Document Header or an ETSI REM evidences document.

6.4.1. Requesting parameters

Each use case has its own parameters for determining what authority is to handle incoming forms and messages. It is recommended not to create a separate web-service or business transaction for each use case. As stated earlier in this chapter, the exhaustive analysis of what parameters are required, is perhaps to be performed by the analysts of WP3. Nevertheless a quick look at the use cases EPO (UC1), Small Claims (UC2) and EAW (UC3) provides the short list below:

Parameter	Description	Use Case				
		1	2	3	4	5
Procedure	Not every court is competent to process any procedure. Based on EU procedure a MS can distinguish Criminal / Civil law, what types of court are competent. (example: EPO, Small Claim, EAW, Financial Penalties, etc..)	√	√	√		
Field of law	In Belgium for instance a claim (EPO) regarding an issue on labour will be processed in a specific court	√	√			
Amount of claim	If the amount of a claim exceeds a certain national limit, another court might be the competent one	√	√			
Location of defendant	Some MS determine the competent court based on the residence of the defendant	√	√	√		

Table 20: Judicial atlas web service input parameter

6.4.2. Responding parameters

Parameter	Description	Use Case				
		1	2	3	4	5
Procedure	Responding to the procedure	√	√	√		
Competent court	Name of the competent court(s). The response might contain multiple competent courts from which a requestor can choose	√	√	√		
Geographical address	To be filled in the forms	√	√			
Electronic address	For electronic routing to correct addressee (see 6.2.1)	√	√	√		
Court fees applied	Information on applied court fees, the amount and payment methods	√	√			
Accepted languages	Applications must be filed in a language, the receiving MS accepts. It is helpful to the user if this information is provided automatically.	√	√	√		

Table 21: Judicial atlas web service output parameter

7. Specifications validation tests

Specifications written in present document should be tested and validated. Basic tests have been described at the time of writing the specification. Required level of detail of these tests is that enough to confirm the specifications could be implemented. Detailed test for each functionality of the pilots are being described in e-CODEX and more involved interoperability and/or conformance tests are still under discussion.

7.1. European e-Delivery Transport Infrastructure

Validation of the transport infrastructure and the gateways are done together, the relevant tests are described together in the next section.

7.2. Gateway

7.2.1. Test environment

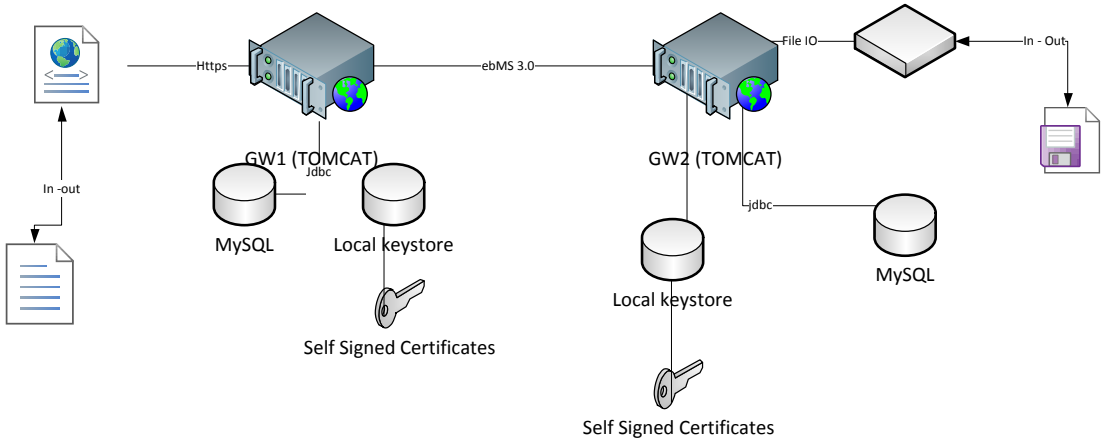


Figure 50 – Gateway test environment

The test environment consists of the 2 GWs up and running communicating with each other. Each of them is hosted on a different HW and has an own DB instance. On one GW a file based backend integration is installed on the other a Web Service based one, in order to test both types of integration.

For test purposes self-signed test certificates are used for the network level as well as the ebMS internal level for signing and encrypting the message.

For test purpose the national subsystems are simulated on one hand by providing a specific set of directories and on the other hand by providing a dummy Web Service.

Please note: The tests will be performed on a Tomcat application server with a MySQL DB in the background.

7.2.2. Test scenarios

The overall test scenarios for the pilots are described in the D10.2 document and are the basic input for the scenarios tests specifically for the GW.

7.2.2.1. Sending a message from A to B

After the successful setup of the test environment a message is sent from one GW to the other. This could be done by providing the corresponding input files in the out directory of the sending GW (if file based integrated) or by calling a Web Service of the GW (if Web service based integrated).

Please note: For tests at this stage test files with dummy data will be used. Only in the final end to end tests representing the scenarios in D10.2 according to the Pilot Uses real EPO, Small Claims or EAW test data (forms) will be used.

Includes the following checks

- ✓ Checking the SSL connection
- ✓ Checking the correct encryption of the message
- ✓ Checking the created Trust OK token
- ✓ Checking if the attachment is included correctly
- ✓ Checking if the end user identification (separate Business document is generated correctly)
- ✓ Checking the log entry
- ✓ Checking the report entry
- ✓ Checking if the message is decrypted and stored correctly at the national subsystem of the receiving GW.
- ✓ Checking if a proper evidence is sent back to the sending GW indicating successful or unsuccessful delivery.

Performing negative tests:

- ✓ Shutdown the national subsystem (file system not available, Web Service not available) and check if the corresponding evidence is sent and the right status has been set in the DB
- ✓ Shutdown temporary the partner gateway and check if the message has been resent if the GW is up and running again according to the configured policy.
- ✓ Shutdown the partner GW and check if the corresponding evidence is sent and the right status has been set in the DB

7.2.2.2. Sending a message from B to A

This test scenario is the same as the above one, but in the other direction.

7.3. e-Payment

The functionalities to be tested are those listed in section 5.3, these are:

- e-Payment information access
- e-Payment execution
- e-Payment receipt handling

7.3.1. e-Payment information access

This functionality should offer the user information on when a payment is required and what and how much should be paid. This information is provided by the MSs as listed in section 5.3.1.

- Checking will consist just in accessing the information and verify the user is able to:
  (1) know whether e-Payment is required;
  (2) how much should be paid;
  (3) and how to proceed.

7.3.2. e-Payment execution

The user will be re-directed to the respective national solution in the appropriate Member State. The way to access the online payment solutions for executing the payment is provided by the MSs as listed in section 5.3.1.

- Checking will be as simple as accessing the national solutions as indicated by the MSs.

7.3.3. e-Payment evidence delivery

Each solution produce a different type of evidences as listed in section 5.3.1, even there are some case in which evidence is not needed. The mean to handle the corresponding payment receipts or evidences would be just attaching the document or information to the message.

For this test, MS are asked to provide a sample of the evidence given by their solutions.

- Checking will be performed just by verifying that the sample evidences provided by the MSs can be delivered together with the message.

7.4. Directory of judicial atlas

According to the business transaction described in chapter 6.4 the test must validate the correct resolution of the competent court for each piloting MS and each procedure (e.g. EPO).

The test scenario covers the sending of a request message (refer to chapter 6.4) to the Gateway of the receiving MS and the return of the response message including the competent court for this specific procedure. How the finding of the court is implemented behind the gateway (dynamically or statically via lists) is in the responsibility of the MS.

If the MS does not pilot the requested procedure then a corresponding error message should be returned.

This scenario will be executed for all piloting MS.

8. Traceability

WP5 requirements are identified and described in D5.1. In this section specification included in the present document are traced against aforementioned requirements. Also traceability with the building blocks identified in deliverable D5.2 is included.

CODE	CLASSIFICATION	REQUIREMENT NAME	BB	Sub-BB	Specification (D5.3)
WP5-RQ-F-001	Functional	Exchange of information/data. e-Filing and e-Payment functionality	e-Delivery	BBT001	Message specifications (3.3.)
			e-Payment	All	e-Payment specifications (5.3.)
WP5-RQ-F-002	Functional	Exchange of information/data	e-Delivery	BBT001 BBT005	Message specifications (3.3.)
			Administration		(Gateway) Configuration and administration (4.4.4.)
WP5-RQ-F-003	Functional	Acknowledgements Management	e-Delivery	BBT006	Evidences (Workflows) (3.5.)
WP5-RQ-F-004	Functional	e-Payment national solution redirection	e-Payment	BBP001 BBP002 BBP003	e-Payment specifications (5.3.)
WP5-RQ-F-005	Functional	Interoperability with e-Justice portal	e-Delivery	BBT001	Gateway (4.)
WP5-RQ-F-006	Functional	Directory	Directory		Possible scenarios (6.3.)

CODE	CLASSIFICATION	REQUIREMENT NAME	BB	Sub-BB	Specification (D5.3)
			e-Delivery	BBT003	Discovery, addressing and endpoint capabilities (3.4.)
WP5-RQ-F-007	Functional	Access Management. Technical Identification and Authorization.	e-Delivery	BBT004	(Gateway) 4.1.3. National TSL or authentication and authorization systems (IDM) (4.1.3.)
WP5-RQ-F-008	Functional	National interface and EU Interface	e-Delivery	BBT001 BBT007	(Gateway) Backend Integration - National System (4.1.2.2.)
WP5-RQ-F-009	Functional	Format Validation	e-Delivery	BBT007	(Gateway) Plugin (4.3.2.4.) (Gateway) Content mapping (4.4.3.)
WP5-RQ-F-010	Functional	Standards conversion	e-Delivery	BBT007	(Gateway) Plugin (4.3.2.4.) (Gateway) Content mapping (4.4.3.)
WP5-RQ-F-011	Functional	Communication information	Administration	WP5_BBAdministration	(Gateway) Logging (4.3.2.3.)
WP5-RQ-F-012	Functional	Statistics information	Administration	WP5_BBAdministration	(Gateway) Logging (4.3.2.3.)
WP5-RQ-NF-001	Non Functional	European frameworks initiatives" requirements	ALL	ALL	ALL

CODE	CLASSIFICATION	REQUIREMENT NAME	BB	Sub-BB	Specification (D5.3)
WP5-RQ-NF-002	Non Functional	e-CODEX - Standards and guidelines	ALL	ALL	ALL
WP5-RQ-NF-003	Non Functional	Mutual recognition in e-CODEX context. Circle of trust.	e-Delivery	BBT004	(Gateway) National TSL or authentication and authorization systems (IDM) (4.1.3.)
WP5-RQ-NF-004	Non Functional	Secure information exchange	e-Delivery	BBT008	(Gateway) National TSL or authentication and authorization systems (IDM) (4.1.3.) (Gateway) Security (4.3.1.2.) End2End Encryption (3.6)
WP5-RQ-NF-005	Non Functional	National Gateways" Identification and authentication	e-Delivery	BBT002 BBT003	Discovery, addressing and endpoint capabilities (3.4) (Gateway) Gateway Authentication (3.3.3.4.1.)
WP5-RQ-NF-006	Non Functional	Integrity	e-Delivery	BBT002	(Gateway) National TSL or authentication and authorization systems (IDM) (4.1.3.)
WP5-RQ-NF-007	Non Functional	Electronic signatures	e-Delivery	BBT002	(Gateway) National TSL or authentication and authorization systems (IDM) (4.1.3.)
WP5-RQ-NF-008	Non Functional	Time stamping	e-Delivery	BBT009	Evidences (Workflows) (3.5)

CODE	CLASSIFICATION	REQUIREMENT NAME	BB	Sub-BB	Specification (D5.3)
figureWP5-RQ-NF-009	Non Functional	Information preservation. Personal Data Protection	e-Delivery Administration	BBT008	4.3.2.3. Logging (4.3.2.3.) (Logging section)
WP5-RQ-NF-010	Non Functional	e-CODEX Infrastructure administration	Administration		(Gateway) Configuration and administration (4.3)
WP5-RQ-NF-011	Non Functional	Administration Restricted Access (No unauthorised access)	Administration		(Gateway) Configuration and administration (4.4.4.)
WP5-RQ-NF-012	Non Functional	Architecture design	ALL	ALL	ALL
WP5-RQ-NF-013	Non Functional	Performance features	ALL	ALL	ALL
WP5-RQ-NF-014	Non Functional	Interoperable Platform	ALL	ALL	ALL
WP5-RQ-NF-015	Non Functional	Error Management	e-Delivery	BBT005 BBT006	(Gateway) Reliability (4.4.4.3.)
WP5-RQ-NF-016	Non Functional	Availability	ALL	ALL	(Gateway) HW Setup (4.1.1)
WP5-RQ-NF-017	Non Functional	National Gateways" requirements	e-Delivery Administration	ALL	Gateway (4.)

Table 22: Traceability between requirements and specifications

9. Conclusions

During the definition of the convergent e-Delivery solution described in chapter 3 the following conditions for the e-CODEX pilots have been defined.

- For the pilots starting at the beginning of 2013 no dynamic routing/discovery will be implemented. The gateway connections are statically defined in the p-mode definitions.
- There will be no central hub in the middle.
- SAML tokens are foreseen but out of scope and not used for the e-CODEX.
- End User Identities will be transported in separate property fields of the message header.

By specifying the generic e-CODEX Gateway in the chapter 4 the following split between the national adapters and the generic Gateway has been defined:

- Trust Ok Token document will be generated within the national adapters by tools provided by WP4. The trust ok token is just another document attached to the message.
- The content conversion from the e-CODEX XML Schemas to the national existing ones will be done in the national adapters.
- The cross border reliable and secure messaging will be provided by the Gateway.
- The Gateway will include a logging, monitoring and configuration facility.
- The Gateway will support the ETSI REM Evidences defined in chapter 3.5

After finishing the D5.3 the development for the generic Gateway starts by establishing a developer team from volunteering MS and a common development environment. Additionally a workshop with the piloting MS will take place (organized by WP3) to clarify the functionality needed by the national adapters.

Regarding e-Payment, national payment rules and legislation vary from one country to another and is not mandatory in all the European countries. The ground required to consider the option of building a common e-Payment solution should be having a homogeneous payment scenario in the Member States. The first step should be done in the legal interoperability layer before a CIP project would be able to define a European solution.

The e-Payment solution defined in the present document aims to provide citizens and other users with the appropriate means to be able to pay court fees in those MS where it is mandatory. The information about court fees and the payment services are to be provided by the MS requiring the payment of court fees. WP5 will be responsible for handling the e-Payment evidences to ensure they arrive at the court together with the process form.

Appendix I – SAML Token Profile

The following profile describes the SAML token to be used for original sender authentication. It is closely modelled after the profiles used by LSPs SPOCS and STORK.

I.1. General

The Assertion MUST be a SAML 2.0 Assertion.

I.2. saml:Issuer

MUST contain the ID of the issuer.

I.3. ds:Signature

The assertion MUST be digitally signed by the issuer. The signature must be an enveloped signature and applied to the saml:Assertion element and all its children. The signature must contain a single ds:Reference containing the saml:Assertion/ID attribute value.

I.4. saml:Subject

MUST contain the end entity identifier in the same format as given for the original sender in the end entity adressing field.

Only the element saml:NameID and saml:SubjectConfirmation are used.

I.4.1. saml:NameId

Mandatory identifier that represents the Subject. The attribute SPNameQualifier shall not be used.

I.4.2. saml:SubjectConfirmation

MUST be present. Attribute "method" MUST be present with one of the following values:

- “urn:oasis:names:tc:SAML:2.0:cm:holder-of-key”
- "urn:oasis:names:tc:SAML:2.0:cm:sendervouches".

Elements <saml:BaseId>, <saml:NameId>, <saml:EncryptedID> shall not be used.

I.4.2.1. saml:SubjectConfirmationData

MUST be present. Rules for attributes of this element:

Attribute	Support	Notes
@NotBefore	MUST	Subject (sender) cannot be confirmed before this time.
@NotOnOrAfter	MUST	Subject cannot be confirmed on or after this time.
@Recipient	MUST	URI reference of the gateway this assertion is being sent to.
@InResponseTo	MUST NOT	Id of the Request that requested this assertion.

@Address	MUST NOT	IP address of user that this assertion was issued to.

I.5. saml:Conditions

MUST be present. Rules for attributes of this element:

Attribute	Support	Notes
@NotBefore	MUST	Assertion not valid before this time.
@NotOnOrAfter	MUST	Assertion not valid on or after this time.

I.5.1. saml:AudienceRestriction

URI reference of the gateway this assertion is being sent to.

I.6. saml:Advice

Advice elements MAY safely be ignored by implementations.

I.7. saml:AuthnStatement

MUST be present. Its attribute SessionIndex shall not be used. Element <saml:AuthnStatement>/<saml:SubjectLocality> shall not be used.

I.8. saml:AuthzDecisionStatement

An Assertion MUST NOT contain an <AuthzDecisionStatement>.

I.9. saml:AttributeStatement

e-CODEX: An Assertion MUST contain at least one <AttributeStatement>.

To provide information about the end entity's initial registration process strength, the following <saml:Attribute> element is defined and may be provided:

@Name =

@NameFormat =

<saml:AttributeValue>: The value of the elements denotes the registration strength level.

Appendix II – Backend Integration WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="Juseb" targetNamespace="http://org.ecodex.backend"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:tns="http://org.ecodex.backend"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xmime="http://www.w3.org/2005/05/xmlmime">
    <wsdl:types>
     <xsd:schema
   targetNamespace="http://org.ecodex.backend"
   xmlns="http://www.w3.org/2001/XMLSchema">
   <xsd:complexType name="sendRequestType">
     <xsd:sequence>
       <xsd:element name="Sender" type="xsd:string"/>
       <xsd:element name="Receiver" type="xsd:string"/>
       <xsd:element name="MS" type="xsd:string"/>
       <xsd:element name="Proceeding" type="xsd:integer"/>
       <xsd:element name="correspondingID" type="xsd:integer"/>
       <xsd:element name="action" type="xsd:string"/>
       <xsd:element name="ID" type="xsd:string"/>
       <!-- Attachments -->
                                                <xsd:sequence>
                                                    <xsd:element minOccurs="0"
maxOccurs="unbounded" name="payload" nillable="true" type="xsd:base64Binary"
xmime:expectedContentTypes="application/octet-stream"/>
                                                </xsd:sequence>
                            <xsd:sequence>
   </xsd:complexType>
    <xsd:complexType name="sendRequestURLType">
     <xsd:sequence>
       <xsd:element name="Sender" type="xsd:string"/>
       <xsd:element name="Receiver" type="xsd:string"/>
       <xsd:element name="MS" type="xsd:string"/>
       <xsd:element name="Proceeding" type="xsd:integer"/>
       <xsd:element name="correspondingID" type="xsd:integer"/>
       <xsd:element name="action" type="xsd:string"/>
       <xsd:element name="ID" type="xsd:string"/>
       <!-- URL to the Attachments data file -->
                                                <xsd:sequence>
                                                    <xsd:element minOccurs="0"
maxOccurs="unbounded" name="payload" nillable="true" type="xsd:string"/>
                                                </xsd:sequence>
                            <xsd:sequence>
   </xsd:complexType>
                    <xsd:complexType name="sendResponseType">
     <xsd:sequence>
       <xsd:element name="state" type="xsd:integer"/>
     </xsd:sequence>
   </xsd:complexType>
  </xsd:schema>

  <xsd:complexType name="getStateRequestType">
     <xsd:sequence>
       <xsd:element name="ID" type="xsd:string"/>
       <xsd:element name="State" type="xsd:integer"/>
```

```
      </xsd:sequence>
    </xsd:complexType>
        </wsdl:types>


        <wsdl:message name="sendRequestEntry">
                <wsdl:part name="sendRequestEntry" type="tns:sendRequestType"/>
        </wsdl:message>
        <wsdl:message name="sendRequestURLEntry">
                <wsdl:part name="sendRequestURLEntry" type="tns:sendRequestType"/>
        </wsdl:message>
        <wsdl:message name="sendResponseEntry">
                <wsdl:part name="sendResponseEntry" type="tns:sendResponseType"/>
        </wsdl:message>
        <wsdl:message name="getStateRequestEntry">
                <wsdl:part name="getStateRequestEntry" type="tns:getStateRequestType"/>
        </wsdl:message>
        <wsdl:portType name="BackendInterface">
                <wsdl:operation name="sendMessage">
                        <wsdl:input name="sendMessageRequest" message="tns:sendRequestEntry"/>
                        <wsdl:output name="sendMessageResponse" message="tns:sendResponseEntry"/>
                </wsdl:operation>
                <wsdl:operation name="sendMessageWithReference">
                        <wsdl:input                                    name="sendMessageWithReferenceRequest"
message="tns:sendRequestURLEntry"/>
                        <wsdl:output                                   name="sendMessageWithReferenceResponse"
message="tns:sendResponseEntry"/>
                </wsdl:operation>
                <wsdl:operation name="getState">
                        <wsdl:input name="getStateRequest" message="tns:getStateRequestEntry"/>
                </wsdl:operation>
        </wsdl:portType>
        <wsdl:binding name="eCODEX" type="tns:BackendInterface">
                <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
                <wsdl:operation name="sendMessage">
                        <wsdl:input name="sendMessageRequest">>
                                <soap:body use="literal" />
                        </wsdl:input>
                        <wsdl:output name="sendMessageResponse">>
                                <soap:body use="literal" />
                        </wsdl:output>
                </wsdl:operation>
                <wsdl:operation name="sendMessageWithReference">
                        <wsdl:input name="sendMessageWithReferenceRequest">>
                                <soap:body use="literal" />
                        </wsdl:input>
                        <wsdl:output name="sendMessageWithReferenceResponse">>
                                <soap:body use="literal" />
                        </wsdl:output>
                </wsdl:operation>
                <wsdl:operation name="getState">
                        <wsdl:input name="getStateRequest">>
                                <soap:body use="literal" />
                        </wsdl:input>
                </wsdl:operation>
        </wsdl:binding>
        <wsdl:service name="eCODEXBackendService">
                <wsdl:port name="BackendPort" binding="tns:eCODEX">
```

```
                    <soap:address location="http://xxxx:1234/eCODEX"/>
                </wsdl:port>
            </wsdl:service>
</wsdl:definitions>
```

Appendix III – P-Mode Configuration[87]

Note that parameter names in Holodeck are not the same as given in the ebMS3 specification.

Parameter	Usage in e-CODEX
PMode.MEP	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/oneWay
PMode.MEPbinding	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/push
PMode.Initiator.Role	Default value: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/initiator
PMode.Responder.Role	Default value: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/responder
PMode[1].Protocol.SOAPVersion	1.2
PMode[1].BusinessInfo.Service	Will have a value that is constructed as follows: http://e-justice.europa.eu/[name-of-use-case]
PMode[1].BusinessInfo.Action	Will have a value that is constructed as follows: http://e-justice.europa.eu/[name-of-form]
PMode[1].BusinessInfo.Properties[]	
PMode[1].BusinessInfo. PayloadProfile[]	Will be used to specify which message parts are mandatory (see section 3.3.3.5 “Summary: Message parts”).
PMode[1].BusinessInfo.PayloadProfile.maxSize	Will for each communication partner (MS) be set to the maximum message size admissible to the corresponding national solution.
PMode[1].BusinessInfo.MPC	Will not be used.

III.1. Reliability

Reliability will be based on WS-Reliability 1.1

PMode[1].Reliability.AtLeastOnce.Contract	true
PMode[1].Reliability.AtLeastOnce.Contract.AckOnDelivery	false
PMode[1].Reliability.AtLeastOnce.Contract.AcksTo	not used
PMode[1].Reliability.AtLeastOnce.Contract.AckResponse	
PMode[1].Reliability.AtLeastOnce.ReplyPattern	Response
PMode[1].Reliability.AtMostOnce.Contract	true
PMode[1].Reliability.InOrder.Contract	false

[87] As far as these parameter values aren’t e-CODEX specific, they correspond to the recommendations in the AS4 profile (http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/csprd03/AS4-profile-v1.0-csprd03.odt).

PMode[1].Reliability.StartGroup	not used
PMode[1].Reliability.Correlation	not used
PMode[1].Reliability.TerminateGroup	not used

III.2. Security

PMode[1].Security.WSSVersion	1.1
PMode[1].Security.X509.Sign.Element[]	
PMode[1].Security.X509.Sign.Attachment[]	Note that Holodeck cannot out of the box sign attachments. It will however be necessary to sign at least the SAML token for end user authentication and the "Trust-ok-Token".
PMode[1].Security.X509.Signature.Certificate	
PMode[1].Security.X509.Signature.HashFunction	
PMode[1].Security.X509.Signature.Algorithm	
PMode[1].Security.X509.Encryption.Encrypt.Element[]	
PMode[1].Security.X509.Encryption.Encrypt.Attachment[]	Note that Holodeck cannot out of the box encrypt attachments. E-CDOEX will however require encryption of all message parts.
PMode[1].Security.X509.Encryption.Certificate	
PMode[1].Security.X509.Encryption.Algorithm	
PMode[1].Security.X509.Encryption.MinimumStrength	
PMode[1].Security.UsernameToken.username	
PMode[1].Security.UsernameToken.password	
PMode[1].Security.UsernameToken.Digest	
PMode[1].Security.UsernameToken.Nonce	
PMode[1].Security.UsernameToken.Created	
PModeAuthorize	
PMode[1].Security.SendReceipt	false
Pmode[1].Security.SendReceipt.ReplyPattern	not used

III.3. Other Required Features

Any ebMS3 messaging product (possibly other than Holodeck) that would be connected to the European Transport Infrastructure needs (in addition to the settings mentioned in the previous section) to support at least[88]:

- Attachments

[88] Note that this list is not exhaustive and may grow over time.

Appendix IV – European e-Justice portal template for payment information from MS

Introduction

Is a paragraph style to be used for the main introduction text in the start of the article. It could be about 5 sentences or even less. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[Quote the national regulation regarding the payment of court fees.]

[Indicate whether electronic payment of fees is available]

Payment of court fees

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[Explain to the citizen if payment of court fees is required and in which processes]

[List the judicial processes subject to payment of court fees:

- *Small Claims*
- *EPO*
- *…]*

Amount of court fees

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[This point should answer the question 'how much must be paid?'. In case calculations are to be done these should be explained here]

Paying court fees

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[This point should answer the question 'how to pay court fees?', providing simple examples]

[In case of electronic payment option, explain it here]

Link to the electronic payment service

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[In case of electronic payment option available, include the link the citizen should use to access the e-Payment service (follow the instructions for including links in the template)]

Evidence of payment. Receipt

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[Describe the evidence the citizen should obtain from the payment act: information included, format, appearance, size, etc. Detail the requirements for this evidence to be valid in front of the court.]

[In case of electronic payment option available, include the same description for the electronic evidence and detail the requirements for this evidence to be valid in front of the court.]

Submission of payment evidence to the court

Normal – This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times. This is just a sample text repeated several times.

[Explanation on how to send the payment evidence to the court, usually attached to the process form.]

<related-links>
(Link to the appropriate e-Payment solution)

</related-links>

<keywords> court fees, electronic payment, payment evidence, receipt, judicial process </keywords>

<updated>Last update: 09/04/2012</updated>

Appendix V – Information about e-Payment from MS

V.1. e-Payment in Austria (AT)

Austria uses direct debit transactions for automatic collection of court fees for electronically filed cases.

This means – in addition to the case data- the claimant (or his lawyer/representative) has to specify IBAN, BIC and the account holder name of his bank account number, which is to be used for the automatic withdrawal of court fees. The back office application of the Austrian courts then calculates the required court fees automatically and collects the required amount for court fees from the specified bank account of the claimant automatically via a direct debit transaction.

The Austrian back-office application for the courts currently uses national direct debit transactions. For e-CODEX this needs to be extended by using SEPA cross-border direct debit transactions.

Important prerequisites:

- All electronic claim forms, which can start a cross-border case to be processed by an Austrian court, **must contain input fields to specify the bank account number (IBAN, BIC, name of account holder) of the claimant (or his lawyer/representative) to be used for withdrawal of court fees.**

Note:

- The Form A for the European Order for Payment Procedure already contains these fields.
- The forms for the other e-CODEX use cases need to be checked!

V.2. e-Payment in Germany (DE)

V.2.1. General remarks

In Germany pre-payment of Court fees is mandatory. Later payment is not considered feasible (neither for court fees nor for other applications in the judicial domain where payment is required) because of the organizational arrangements this would imply.

Therefore bank transfer and credit card payment are of interest as payment methods. Among these, credit card payment is often preferred because it is immediate, whereas for bank transfer the goods or services that are being paid for need to be held until payment arrives.

In the “Land” North Rhine-Westphalia, the following online procedures which use different payment methods are currently offered:

- Electronic voucher for payment of court fees: bank transfer or credit card
- Online auction platform of the justice of North Rhine-Westphalia : Pre-Paid via bank transfer
- Information from trade register via the internet for registered users: monthly account; direct debit is possible
- Information from land register via the internet for admitted users and authorities: monthly account; direct debit is possible

For e-CODEX, the first of the above solution (electronic voucher: “Kostenmarke”) will be used.

V.2.2. Electronic voucher for payment of court fees

For the “Elektronische Kostenmarke” (electronic voucher for payment of court fees) the procedure is as follows:

- In a web application (www.kostenmarke.justiz.de) users can purchase a voucher.
- Accepted payment methods are bank transfer and credit card.
- A unique identifier is generated, which corresponds to a specific amount of money.
- The receipt contains this unique identifier and a corresponding bar code.
- After reception of payment, this unique identifier will in the database be tagged as “paid”. (Successful payment by credit card is considered immediate reception of payment)
- For a given receipt with its unique identifier submitted to a court, the court clerk can via another web application check if this identifier is tagged “paid”.
- Only if this is the case the proceeding is continued.

V.3. e-Payment in Spain (ES)

V.3.1. Judicial fees

In Spain it is not necessary to pay fees to start a trial.

Spanish citizens do not need to pay fees, costs are covered by taxes, justice is free by law in Spain.

V.3.2. Payments arisen from judicial processes

All payments associated with the development of the judicial process is carried out through an existing bank account number for each court (called ‘Appropriation Account’, and there is one per court), besides it’s necessary to include a special code that the Court gives you and it’s associate to the judicial process.

The management of the payments is made through an application for the Judicial Secretaries. They can consult and review all payments in their Appropriation Account.

We have national regulations for all such payments:

- RD 467/2006, 21 April, by regulating the judicial appropriations cash deposits, effects or values.
- ORDEN JUS/1623/2007, 4 April, by adopting the model forms of income payment orders and transfer orders governed by the RD 467/2006.

V.4. e-Payment in France (FR)

France sent a proposal for an e-Payment approach: it would consist on a central portal where the user is redirected to the national e-Payment solution and receive a stamp after the execution of payment.

In order to pay a tax, the citizen will add the number of the stamp in the form.

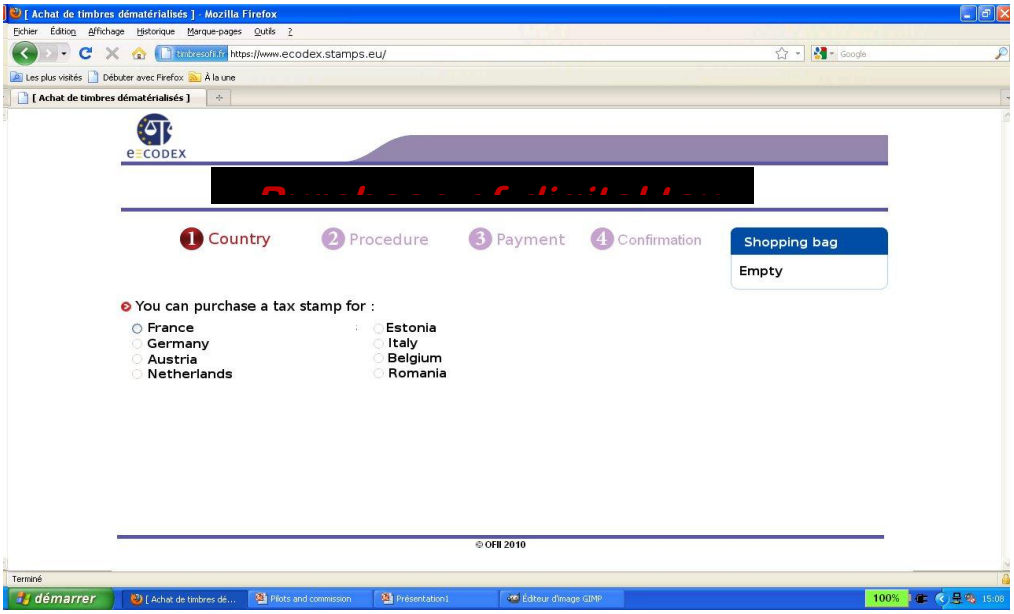


Figure 51 – France proposal of a central interface

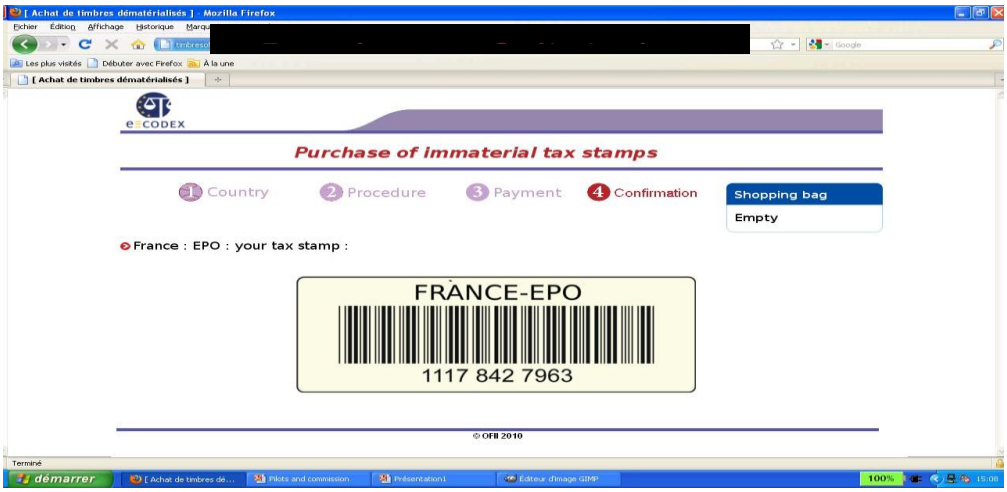


Figure 52 – Tax stamp proposal

V.5. e-Payment in Italy (IT)

V.5.1. Introduction

The Italian system is valid for the payment of taxes by individuals for the benefit of the public administration in order to cover management expenses of proceedings (court costs) or to obtain a service (registration, copy paper, certificate, notifications, etc).

The service is subject to verification of payment.

V.5.2. Phases

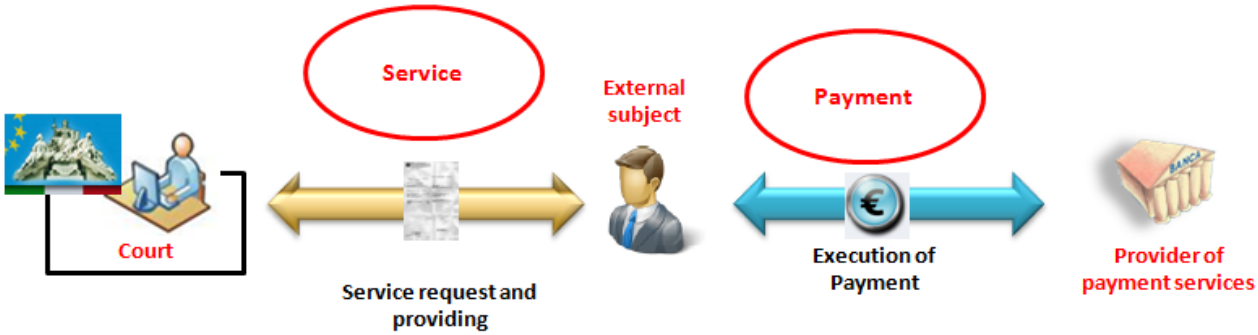
The process of electronic payment for justice costs is divided conceptually into two distinct and independent stages:

1) A phase in which the subject interacts with the Justice domain to request the service
2) and a phase in which the payer interacts with the domain of the financial intermediates (or providers of payment services) to perform the payment of the requested sums.

The two phases, depending on the type of service provided, may be strictly sequential or interleaved in a single stream.

A subject can perform an electronic payment independently by the level of computerization of the courts office: an electronic payment can be accepted and validated in a traditional way.

The financial intermediates are banks and the Italian Postal Service.



Payment phase: the payer performs the electronic payment through a channel that provides the requirements for identification, authorization, and integrity of payment data. The electronic channel gives the payer the ability to use different means for payment, such as debit cards, credit cards, prepaid cards or other means of payment by electronic money.

Through the same channel the subject receives the electronic payment receipt.

Service phase: the subject requires the service. Depending on the type of service, the payment receipt will be forwarded to the office through electronic channel or in traditional mode (paper). The court office verifies the integrity of the receipt, preserves it and provides the requested service.

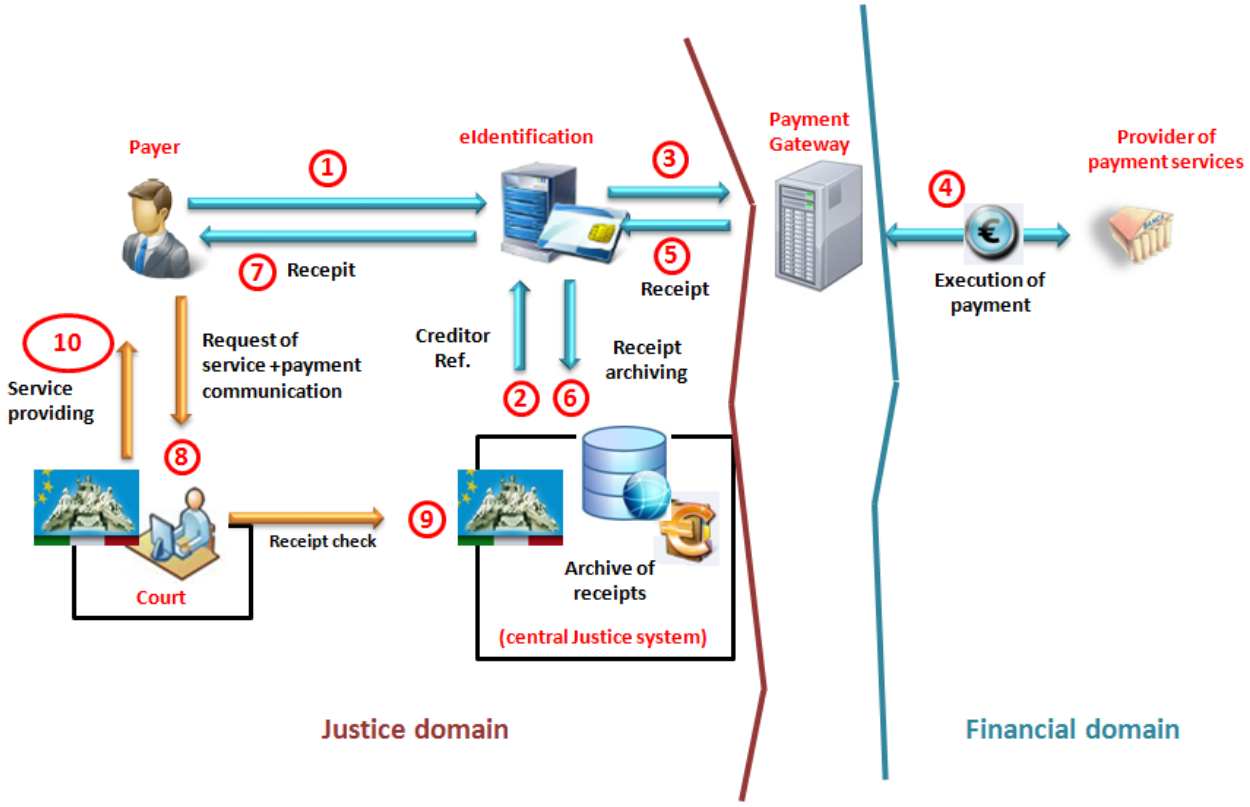
In case of synchronous service, the two processes are managed through protocols and interoperability in a fully automated flow to meet the demands of the payer in an integrated and transparent way.

V.5.3. Logic and functional model

The steps outlined above, while remaining independent and asynchronous, require the definition of information shared between the two areas (financial and justice), and agreement on protocols and application logic.

To implement the interaction between different systems within a single business process, ensuring the independence of each domain in order to architectural choices, infrastructure, formats and streams, a logic-functional component, the ***Payment Gateway***, has been introduced; it provides sharing capabilities of protocols (application and transport) and transformation of formats making the decoupling of infrastructures and functions between the financial domain and the justice domain.

SOA model was adopted as a paradigm for interoperability between applications, each under the control of different domains.



(following the picture's numeration)

Payment Phase

1. The external subject gets access to the payment service through strong authentication on his/her Access Point.[89]
2. The eJustice system provides a unique identifier to be associated to the payment procedure (Creditor Reference - ISO 11694 – is used). This information has to be put into the electronic payment receipt.
3. Data related to the payment (amount, recipient, causal, ...) are transmitted through the exchange of XML structured data to the "payment gateway" which, as shown, defines the gateway to the domain of entities providing the payment services.
4. Within the "banking world", the gateway shall generate a request for transfer according to the SEPA standards and manages and interprets the typical flows of bank transactions. The payment is completed within the systems of the provider of payment services according to their typical work flows.

[89] The Access Point is authorized by the Ministry of Justice and has the responsibility to e-Identify external users

5. Once the payment is completed, the gateway generates a receipt of payment in the form of structured interchange file (XML format) according to an XSD defined in the justice domain. The payment receipt is digitally signed.
6. The payment receipt is stored in a special file system (“archive of the receipts”) managed by the Ministry of justice.
7. The receipt of payment is returned to the payer.

Service Phase

8. The subject sends the receipt to the Court together with the request of service or the indication of the reason for the payment. This can be sent using the electronic channel, if available at the office, or in the traditional way (paper).
9. The court’s system checks the integrity of the payment receipt (e-Signature of receipt), verifies that the receipt states the correct charge and that it has never been used before. The archive of the receipts allows avoiding interaction with the systems of the financial domain to cross-check.
10. The Court accepts the payment and provides the requested service.

V.5.4. Archive of the receipts

The definition of the component “archive of the receipts” allows systems in Courts to verify the payments made independently from the processing state of the economic transaction (reconciliation, reporting, etc.).

Please note that this mode of operation is assured by special legal provisions related to payments made to the State Treasury and the means of payment accepted. Under those provisions, in fact, the payment is effectively made when the financial subject charges the sum on the bank account of the payer.

This component also allows the storage of the statements made by local providers of payment services and the consequent implementation of analysis of data on payments to justice.

V.6. e-Payment in The Netherlands (NL)

The current situation in the Netherlands regarding the collection of court fees via e-Payment is that there is a pilot, not a fully implemented system. Bailiffs, barristers and lawyers connected to the courts in The Hague and Amsterdam receive, if participating in this pilot, digital invoices via a Biller Service Provider (BSP). The courts deliver the BSP the invoices for the court fees. The BSP deals thereupon with the bailiffs, barristers and lawyers. Successful fee collection is not a prerequisite in all cases to have access to the legal procedure. In those cases the fee can be paid afterwards as well. The payment process is not expected to use the e-banking facilities that banks provide their customers with. For bailiffs, barristers and lawyers the possibility to use batch procedures for multiple transactions is highly desirable. This service is not open to private citizens.

During this pilot, maybe afterwards as well and maybe the debtor decides on the way payments are done. BSP is the agent sending the invoice to the debtors if payment to the courts is done electronically. Apart from e-Payment cash payments are right as well. The Financial administration system of the courts is not yet ready for dealing with e-Payment smoothly. Improvements are expected mid next year with the introduction of a new Financial system.

It may be clear from this short notice that courts in the Netherlands are progressing in using e-Payment for collecting court fees. It should also be clear that the experiences so far are limited and a lot can be learned from our colleagues in e-CODEX.

V.7. e-Payment in Portugal (PT)

V.7.1. Introduction

In Portugal there are two situations for which e-Payment is available:

1. Order for payment procedure, available only for lawyers/solicitors
2. General, available to everyone

The payment of taxes in order to cover management expenses of proceedings (court costs) is always prior to the process filing. The latter is subject to the verification of payment.

The process for each of these situations is presented next.

V.7.2. Order for payment procedure

For filing an order for payment procedure, available only for lawyers/solicitors:

1. access the platform Citius
    a. fill all the data needed for filing the case electronically
    b. receive a payment reference (number, entity and amount) to perform the payment
2. perform the payment elsewhere (home banking / ATM / bank branches)
3. when (and only when) the payment is detected by the court system, automatically, with no manual action, will the system allow the case to be treated

V.7.3. Other processes

For each of a number of procedures, a pre-payment process is available for the regular citizen/company:

1. Access the website – self-service payment site (https://igfij.mj.pt/CUSTAS/Paginas/Autoliquidacoes.aspx)

2. Choose the type of judicial case that applies (kind of wizard that guides the user through the process). For instance:

IGFIJ IP

INSTITUTO DE GESTÃO FINANCEIRA E DE INFRA-ESTRUTURAS DA JUSTIÇA, I.P.

Pesquisar: Âmbito: Geral OK Pesquisa Avançada | Perguntas Frequentes | Links | Registe-se | Mapa do Portal | Contactos

Portal Internet IGFIJ » Custas Judiciais » Autoliquidações

05.Abr.12

- IGFIJ
- Documentação e Contratação
- Notícias
- Áreas de Actividade
- SIADAP
- Custas Judiciais
  - Autoliquidações
  - Revalidações
  - Reembolsos
- Depósito de Receitas
- Newsletters

Início > DL 34/2008

Autoliquidação de Taxas de Justiça

Geração de DUC (Documento Único de Cobrança) – Portaria n.º 419-A/2009, de 17 de abril, com as alterações introduzidas pela Portaria n.º 82/2012, de 29 de março.

Ano do processo:

2012

Tipo de pagamento:

- Acções Declarativas (A - Acções Declarativas) - Tabela I
- Acções Declarativas e Recursos (B - Recursos e Situações Especiais) - Tabela I
- Acções Declarativas e Recursos (C - Grandes Litigantes) - Tabela I
- Processos administrativos urgentes (artigos 97.º e 100.º do CPTA) - Tabela II
- Incidente de intervenção provocada principal ou acessória de terceiros e oposição provocada - Tabela II
- Execuções - Diligências não realizadas por oficial de justiça - Tabela II A
- Execuções - Diligências não realizadas por oficial de justiça - Tabela II B (Grandes Litigantes)
- Execuções - Diligências realizadas por oficial de justiça - Tabela II A
- Execuções - Diligências realizadas por oficial de justiça - Tabela II B (Grandes Litigantes)
- Reclamações de créditos - Tabela II
- Oposições à execução ou à penhora/embargos de terceiro - Tabela II
- Incidentes e Procedimentos - Tabela II A
- Incidentes e Procedimentos - Tabela II B (Grandes Litigantes)
- Reclamações, pedidos de rectificação, de esclarecimento e de reforma da sentença
- Processos da competência do Ministério Público previstos no DL 272/2001
- Penal

IA -	Acções declarativas, não contidas nas restantes hipóteses
IB -	Recursos, cada interveniente coligado, cada interveniente associado à parte, assistentes em processo civil, administrativo e tributário.
IC IIB	Acções propostas por sociedades comerciais, que no ano anterior tenham proposto mais de 200 acções, procedimentos ou execuções.

Voltar Próximo

ÁREA RESERVADA

Email do Utilizador:

Senha:

- Registe-se
- Perdeu a senha?

INQUÉRITOS

Para votar no inquérito corrente deverá estar previamente registado no Portal do IGFIJ.

Acha que o site está acessível?

- Não
- Sim

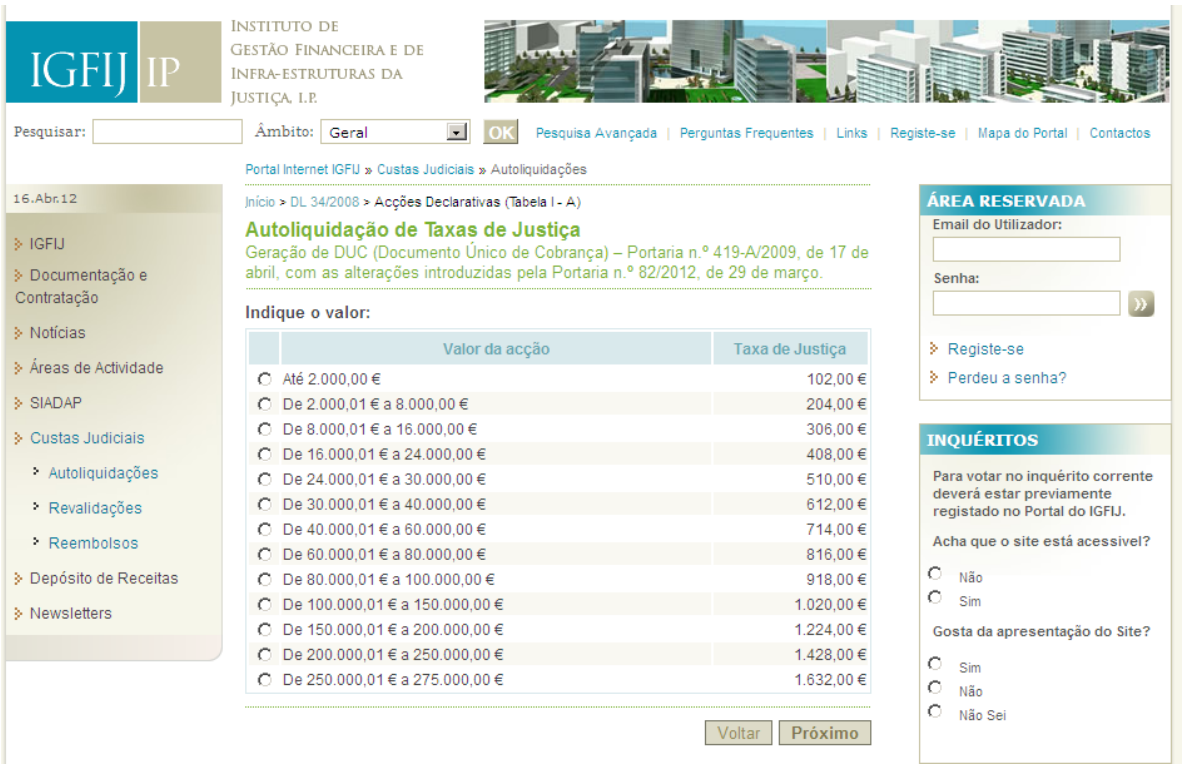
Gosta da apresentação do Site?

- Sim
- Não
- Não Sei

VEJA TAMBÉM

MAIS CONCORRÊNCIA MENOS BUROCRACIA

Citius


IGFIJ IP
Instituto de Gestão Financeira e de Infra-estruturas da Justiça, I.P.
Pesquisar:
Âmbito: Geral
OK
Pesquisa Avançada | Perguntas Frequentes | Links | Registe-se | Mapa do Portal | Contactos
Portal Internet IGFIJ » Custas Judiciais » Autoliquidações
16.Abr.12
IGFIJ
Documentação e Contratação
Notícias
Áreas de Actividade
SIADAP
Custas Judiciais
Autoliquidações
Revalidações
Reembolsos
Depósito de Receitas
Newsletters
Início > DL 34/2008 > Acções Declarativas (Tabela I - A)
Autoliquidação de Taxas de Justiça
Geração de DUC (Documento Único de Cobrança) – Portaria n.º 419-A/2009, de 17 de abril, com as alterações introduzidas pela Portaria n.º 82/2012, de 29 de março.
Indique o valor:
Valor da acção
Taxa de Justiça
Até 2.000,00 € 102,00 €
De 2.000,01 € a 8.000,00 € 204,00 €
De 8.000,01 € a 16.000,00 € 306,00 €
De 16.000,01 € a 24.000,00 € 408,00 €
De 24.000,01 € a 30.000,00 € 510,00 €
De 30.000,01 € a 40.000,00 € 612,00 €
De 40.000,01 € a 60.000,00 € 714,00 €
De 60.000,01 € a 80.000,00 € 816,00 €
De 80.000,01 € a 100.000,00 € 918,00 €
De 100.000,01 € a 150.000,00 € 1.020,00 €
De 150.000,01 € a 200.000,00 € 1.224,00 €
De 200.000,01 € a 250.000,00 € 1.428,00 €
De 250.000,01 € a 275.000,00 € 1.632,00 €
Voltar
Próximo
ÁREA RESERVADA
Email do Utilizador:
Senha:
Registe-se
Perdeu a senha?
INQUÉRITOS
Para votar no inquérito corrente deverá estar previamente registado no Portal do IGFIJ.
Acha que o site está acessível?
Não
Sim
Gosta da apresentação do Site?
Sim
Não
Não Sei


IGFIJ IP
Instituto de Gestão Financeira e de Infra-estruturas da Justiça, I.P.
Pesquisar:
Âmbito: Geral
OK
Pesquisa Avançada | Perguntas Frequentes | Links | Registe-se | Mapa do Portal | Contactos
Portal Internet IGFIJ » Custas Judiciais » Autoliquidações
16.Abr.12
IGFIJ
Documentação e Contratação
Notícias
Áreas de Actividade
SIADAP
Custas Judiciais
Autoliquidações
Início > DL 34/2008 > Acções Declarativas (Tabela I - A) > Gerar DUC
Autoliquidação de Taxas de Justiça
Geração de DUC (Documento Único de Cobrança) – Portaria n.º 419-A/2009, de 17 de abril, com as alterações introduzidas pela Portaria n.º 82/2012, de 29 de março.
Código de segurança
Por questões de segurança, insira o código que vê na imagem seguinte e clique 'Confirmar':
454494
Voltar
Confirmar
ÁREA RESERVADA
Email do Utilizador:
Senha:
Registe-se
Perdeu a senha?
INQUÉRITOS
Para votar no inquérito corrente deverá estar previamente

3. Choose “Emitir Documento” (issue document) by pressing the button, in order for the payment document to be generated, see next figure for instance.
4. With this document, the person has a unique code (702 6980 024 027 405 on the example above) which will be used to pay for the procedure
5. Perform the payment via home banking, on ATM machines or at bank branches (payment to the State)
6. Deliver the receipt (either ATM or banking receipt) together with all the specific information of the case, in order to be able to file it.

INSTITUTO DE GESTÃO FINANCEIRA E DE INFRA-ESTRUTURAS DA JUSTIÇA, I.P.

MINISTÉRIO DA JUSTIÇA

DUC (Documento Único de Cobrança)

Tipo de pré-pagamento	Lei 7/2012 – Regulamento das Custas Processuais
Tipo de acção	Acções Declarativas (A - Acções Declarativas) - Tabela I
Descrição do pagamento	Até 2.000,00 €
Entrega electrónica	Não
Pagamento a prestações	Não

Referência para pagamento	702 680 024 027 405
Montante a pagar	102,00 €
Data de emissão	05-04-2012 19:05:22

O pagamento pode ser efectuado através do Multibanco, da Internet e das instituições de Crédito aderentes (aos balcões ou através da internet), utilizando a referência indicada.
Para efectuar o pagamento pela Internet, utilize o serviço on-line do seu banco, seleccionando «Pagamentos ao Estado». Válido como recibo, após certificação, ou juntamente com o documento emitido pela entidade cobradora.

TAXAS DE JUSTIÇA: O documento comprovativo do pagamento da taxa de justiça perde validade 90 dias após a respectiva emissão, se não tiver sido, entretanto, apresentado em juízo ou utilizado para comprovar esse pagamento, caso em que o interessado solicita ao Instituto de Gestão Financeira e de Infra-Estruturas da Justiça, I. P., no prazo de seis meses, a emissão de novo comprovativo quando pretenda ainda apresentá-lo.

A emissão do novo comprovativo só poderá ser efectuada através da internet, utilizando a funcionalidade "Revalidação de taxas de justiça", bastando para o efeito digitar a referência do pagamento do documento original.

Se o interessado não pretender apresentar o comprovativo em juízo, requer ao Instituto de Gestão Financeira e de Infra-Estruturas da Justiça, I. P., no mesmo prazo, o reembolso da quantia despendida, mediante entrega do original ou documento de igual valor, sob pena de reversão para o referido Instituto.

DEPÓSITOS AUTÓNOMOS: Se o documento comprovativo do pagamento do depósito autónomo não for apresentado em juízo ou utilizado para comprovar esse pagamento, o reembolso da quantia despendida pode ser requerido ao Instituto de Gestão Financeira e de Infra-Estruturas da Justiça, I. P., mediante entrega do original ou documento de igual valor, sob pena de reversão para o referido Instituto."