D32.2 Requirements Report for eID Service of FutureID Client

Document Identification	
Date	28/02/2012
Status	Final
Version	1.0

Related SP / WP	SP3/WP2	**Document Reference**	Insert Reference #
Related Deliverable(s)	D32.1, D22.1-7	**Dissemination Level**	PU
Lead Participant	IBM	**Lead Author**	Gregory Neven
Contributors	Thomas Gross (UNEW) Moritz Horsch (TUD) Meiko Jensen (ULD) Peter Lipp (TUG) Frank-Michael Kamm (GD) Jon Shamah (EEMA) Harald Zwingelberg (ULD)	**Reviewers**	Christopher Ruff (USTUTT)

This document is issued within the frame and for the purpose of the FutureID project. This project has received funding from the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318424

1. Abstract

This deliverable describes requirements for the eID Service for the FutureID client that will integrate existing eID technology and trust infrastructures, emerging federated identity management services and modern credential technologies to provide a user-centric system for the trustworthy and accountable management of identity claims.

The requirements were extracted from a number of selected core use cases, covering scenarios as broad as customer identification in the financial industry, direct democracy, third-party issuers, electronic contract signing, age verification, and smart cards.

We extracted a list of concrete requirements that include general high-level requirements as well as more specific technical requirements with respect to the client's eID services. Among the technical requirements, the call for a broad support of various well-established standards, at all levels in the protocol stack, is very prominent. Privacy was a recurring theme in the more general requirements, which among other things asked for clear transaction information for the user and the support of privacy-preserving authentication technologies.

2. Document Information

2.1 Contributors

Name	Partner
Thomas Gross	UNEW
Moritz Horsch	TUD
Meiko Jensen	ULD
Peter Lipp	TUG
Frank-Michael Kamm	GD
Gregory Neven	IBM
Jon Shamah	EEMA
Harald Zwingelberg	ULD

2.2 History

Version	Date	Author	Changes
0.1	14/02/2013	All	Import contributions from Wiki
0.2	15/02/2013	Jon Shamah, Moritz Horsch	Minor changes & clarifications
0.3	15/02/2013	Thomas Gross	Minor changes
1.0	28/02/2013	Gregory Neven	Addressing reviewer comments by Chrisopher Ruff.

3. Table of Contents

4. Introduction

The FutureID project will build a comprehensive, flexible, privacy-aware and ubiquitously usable identity management infrastructure for Europe, which integrates existing eID technology and trust infrastructures, emerging federated identity management services and modern credential technologies to provide a user-centric system for the trustworthy and accountable management of identity claims.

This deliverable describes requirements for the eID Service for the FutureID client. The requirements were extracted from a number of selected use cases that were identified as core FutureID use cases at the FutureID Requirements Meeting in Finse, Norway, in January 2013.

4.1 Scope

The main focus of this deliverable is on technical requirements that concern services that will be provided by eID on the side of the client (i.e., the private user to whom the eID is issued). However, while describing the use cases, we also surfaced a number of more general requirements that also apply to other parts in the FutureID identity management infrastructure. Rather than discarding these as out of scope for this document, we chose to preserve these requirements in the document

The requirements summarized in this deliverable may show a considerable overlap with those that will be formulated as part of the dedicated requirements Work Package 22, in particular, in the areas of technology, security, privacy, and usability (Tasks 22.1, 22.2, 22.3, and 22.4). We also foresee a considerable overlap with the requirements that will be formulated in Work Package 31 (Interface Device Service), Work Package 33 (eSign Services), Work Package 34 (User Interface), Work Package 35 (Trustworthy Client Platform), and Sub Project 4 (Backend Integration), each of which have their own dedicated requirements Tasks.

Given the early timing of this deliverable and the very preliminary state of other Work Packages' deliverables at this point, we decided to preserve all requirements that we derive from our use cases, even if they are not strictly within the scope of this Task. We felt that doing so will be more helpful as input to the rest of the project than trying to steer strictly clear from the waters of other Work Packages, and thereby risking to give an incomplete picture.

4.2 Outline

This document first considers a number of specific use cases and derives a list of concise requirements from each of these use cases. In a next step, we consolidate the derived requirements in a merged list that eliminates double requirements that appeared in several use cases and categorizes the requirements as high-level requirements (that may interfere with the scope of other deliverables) and concrete technical requirements for the eID client services (that are the main focus of this document).

The list of use cases that we selected for closer inspection in the subsequent sections are:

1. Know Your Customer
   A number of compliance regulations in the financial industry impose financial institutions to know exactly which customers they're doing business with. By leveraging eIDs and third-party certified attributes, many processes that are currently paper-based can be made more efficient digitally.

2. Direct Democracy
   Many aspects of social life, from small clubs or private entities to local and state governments, would profit from broader participation in joint decision making or opinion building (e.g., polls or internal elections). Online participation based on eID could help, but there are strong security and privacy requirements to take into account.

3. Third-Party Issuers
   Governments usually only include basic identification attributes such as name, address, and civic registration numbers as part of an eID. Other third parties, both governmental and non-governmental, may want to issue additional verifiable attributes in connection with the eID.

4. eSigning and Validation
   Electronic document signing is arguably the "killer application" of eIDs. Huge cost savings can be expected from eliminating the "media break" in many digitized workflows that often still require an approval step based on ink signatures.

5. Age Verification
   Protection of minors online is another important use case of eIDs, both to prevent minors from accessing goods that are considered harmful (e.g., tobacco or alcohol), as well as to protect minors from malicious adults (e.g., teenage chatrooms). But how to enforce age restrictions without requiring children to reveal a highly identifying attribute such as their exact birthdate?

6. Smart Cards and Secure Elements
   The security of all currently deployed eID solutions boils down to the trustworthiness of a small piece of hardware at the heart of the device, called a smart card or a secure element. Many technical standards are in place, so the FutureID client must ensure to support the relevant ones.

In Section 11, we give an overview of all collected requirements, in two separate categories of general high-level requirements or core technical requirements.

4.3 Terminology

We will use the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document to indicate priority levels of requirements. They are to be interpreted as described in [RFC 2119].

5. Know Your Customer

5.1 Use Case Description

5.1.1 Background to "Know Your Customer"

Know Your Customer (KYC) refers to the due diligence activities that financial institutions and other regulated companies must perform to ascertain relevant information from their customers for the purpose of doing business with them. The term is also used to refer to the bank regulation which governs these activities. KYC processes are also employed by companies of all sizes for the purpose of ensuring their proposed agents', consultants' or distributors' are Anti Money Laundering (AML) compliant. Banks, insurers and export credit agencies are increasingly demanding that customers provide detailed anti-corruption due diligence information, to verify their probity and integrity. Today, KYC policies are becoming increasingly important globally to prevent identity theft, financial fraud, money laundering and terrorist financing, amongst other activities.

5.1.2 Goal of the Use Case

By the re-use of data and attributes available to Identity Service Providers, individuals and legal entities, and with the granting of permission, the cost of document verification and identity proofing as part of the KYC process can be radically reduced across the range of financial services subsectors. For both citizen and legal entity cases, the use case will identify common protocols for use in an eventual standard for KYC information provision which will respect data protection and privacy.

5.1.3 Unique Value Proposition

By use of a federated eID scheme, the cost of trusting KYC verification credentials, often of cross-border origins, should be considerably reduced. Additionally the time taken to verify the trust in these credentials and associated attributes will also be reduced.

5.1.4 Measurement of External Stakeholder Value

- Applicability to KYC processes across subsectors in EU member states.
- Ease of integration with existing processes and attribute service providers
- Reduction in cost of KYC process across a number of subsectors
- Reduction in time to perform the KYC process across a number of subsectors
- Measured increase in reliability of the KYC process across a number of subsectors

5.2 Definitions

5.2.1 User

This is defined as the entity that is the subject of the KYC enquiry. This could be an indiviual or organisation.

5.2.2 Identity Sevice Provider (IdsP)

The originator and authority of the eID. Typically this will be a Government Agency, but may be any 'Notifiable eID' under the proposed EU Trusted Services Regulations (June 2012)

5.2.3 KYC Service Application

This is defined as either an automated application seeking to gather information to verify the identity of a user to certain policy levels for KYC purposes or part of a manual process which requires the user to verify its identity and permission the release of relevant verification.

5.2.4 Identity Attribute

There are 3 classes of verifiable identity attribute data.

5.2.4.1 Signed declaration

Typically a PDF or image which states the required proof (such as: a current "utility bill") and which is digitally signed by the originator of the proof.

5.2.4.2 Signed Affirmation/score

Typically a signed token confirming a status, such as: "This person has a utility account at this address", or "this person has a credit score of 6".

5.2.4.3 Virtual Company Dossier (VCD)

Defined by PEPPOL, SPOCS and ESENS (in 2013), the VCD is a container comprising all the documents needed to establish the identity of a business for working in an EU member state or for entering an EU public procurement. For an individual, the VCD should contain mandates to confirm an individual's role within a company. By implication, a VCD should be sufficient for KYC.

Attributes may also be self-declared/verified, where the user signs the attribute itself, and is typically only of value if the attribute is 'incidental', such as 'my favourite porridge'.

http://www.peppol.eu/peppol_components/virtual-company-dossier

5.2.5 Attribute Originator

An organisation which is trusted within the policies of the KYC and signs a declaration or affirmation attribute concerning the user. (An example could be a Government Driving License Agency)

5.2.6 Attribute Manager

An attribute manager assists the user in controlling the large number of personal attributes that should rest in the control of the user. Best practise dictates that attributes should be kept close to the originating source, but often, for service level reasons, these can be retained and stored by, or on behalf of the user itself. If these attributes are signed by the attribute originator, and that signature (whose valididity duration mirrors the validity of the attribute) is neither revoked, nor expired, the identity attribute can be positively verified by the KYC Service Application.

The Attibute Manager can either be entirely independent, or hosted at the IDsP as a white-labelled value-add service. The IdsP must be informed of the location of the Attribute Manager.

5.2.6.1 Electronic Bank Account Management (eBAM)

A special case of identity attribute communication which is similar to KYC is eBAM. This is the electronic management of high value bank accounts and instructions through digital channels.
An example would be a large corporate with (say) 150 bank accounts in 50 different countries and in different banks. The corporate treasury needs to keep track of bank signing mandates and jurisictional requirements as well as managing cross account transfers and setting up new accounts. This is a major cost and delay in corporate flexibility and also is a major overhead at the banks.

As a result corporates use eBAM software (either proprietary or COTS) which utilises a standard XML schema (as defined by ISO 20022 - a generic financial messaging standard) and includes: all generic KYC information; bank required specific information; and jurisdictional information, needed to conduct a transaction. In general channels include

- SWIFT Messages (corporate-2-swift-2-bank)
- BAM Proprietary specific bank systems (corporate-2-bank)
- eBAM using standard messaging.(corporate-2-many banks)

All the channels use ISO 20022 as the standard XML scheme. (Note that ISO 20022 will soon be superceded by ISO 20022 v2) Many of the KYC attributes are required for eBAM.

5.2.6.2 Other Initiatives

Other initiatives can be found at:

- Mydex(Personal Data Store): http://mydex.org
- Project Danube (Personal Data Store): http://projectdanube.org/
- Attributes In Motion (Kantara Initiative): http://kantarainitiative.org/confluence/display/AIM
- Attribute Exchange (OIX): http://openidentityexchange.org/projects/axn-pilots

5.3 Technical Architecture

The probable architecture would comprise of a number of components and integration with stakeholder services and processes. The exact KYC Application architecture and Trust Policies are out of scope of this document but a representative technical design is shown below. There are very few operational data sources for identity attributes that are available. Organisations such as Experian, Eurochambre and others could probably be sources of reliable and trusted information that can be verified and certified.

There are two architectures available for communication.

1) Collection of the certified identity attributes by the IDsP and then onward transmission with the SAML identity assertion (example #2a or #2b), with the ID Broker acting as the despatcher of data

2) Direct delivery of the certified identity attributes from the source directly to the KYC Service Application with FutureID acting as Authenticator only.

For the purpose of this use case and taking into account scalability and practical issues, it is considered that the functional use of FutureID is as per Figure 1.

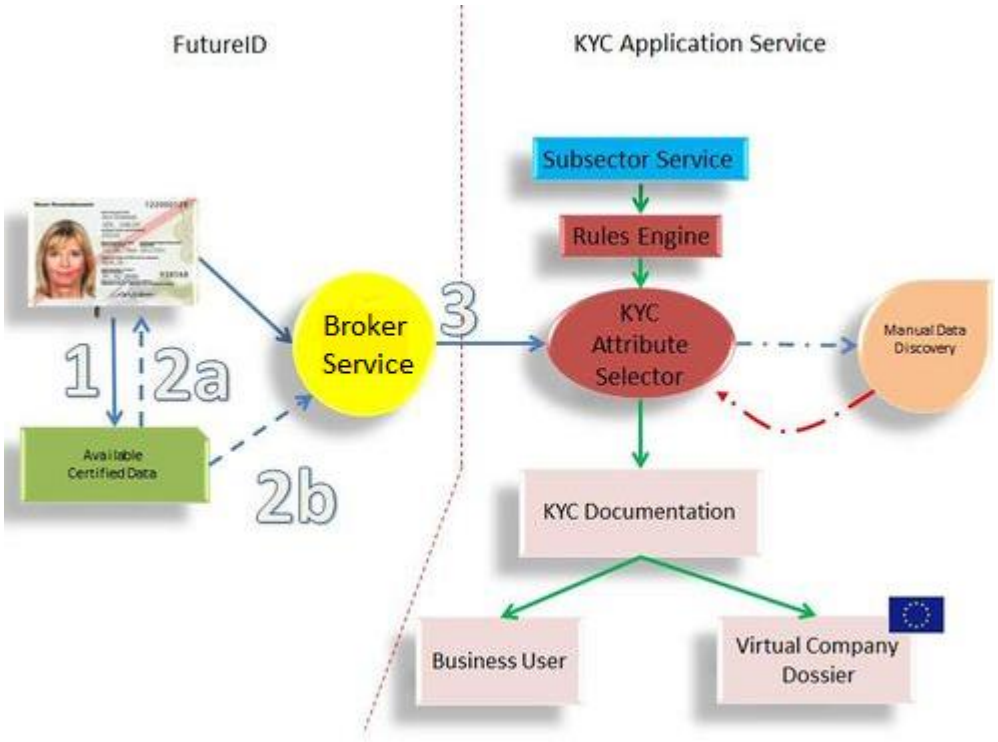

Figure 1 Technical architecture

There are a number of methodologies being proposed in the market and as of the date of publication of this document, no standards have been proposed for the management of identity attributes but some methodologies are being established (see below). For this reason, the requirements for this use case will include all possible variations. It should be noted that these requirements will be common with many other use cases.

5.3.1 Process

- The user enters the KYC Application page or is approached as part of an automated process
- The KYC Application requests the user to authenticate with its eID.
- The user is asked to release required declarations/affirmations by the KYC Application.
- The Attribute Manager holding the attribute data is requested to provide the declarations/affirmations or their URLs.

- The Attribute Manager holding the attribute data forwards the signed declarations/affirmations or their URLs to the KYC Application, via the Broker Service Despatcher (and via a despatcher at the IdsP if needed)
- The KYC process continues (using other methods if verified information is insufficient.

A possible architecture for the process is described in Figure 2.

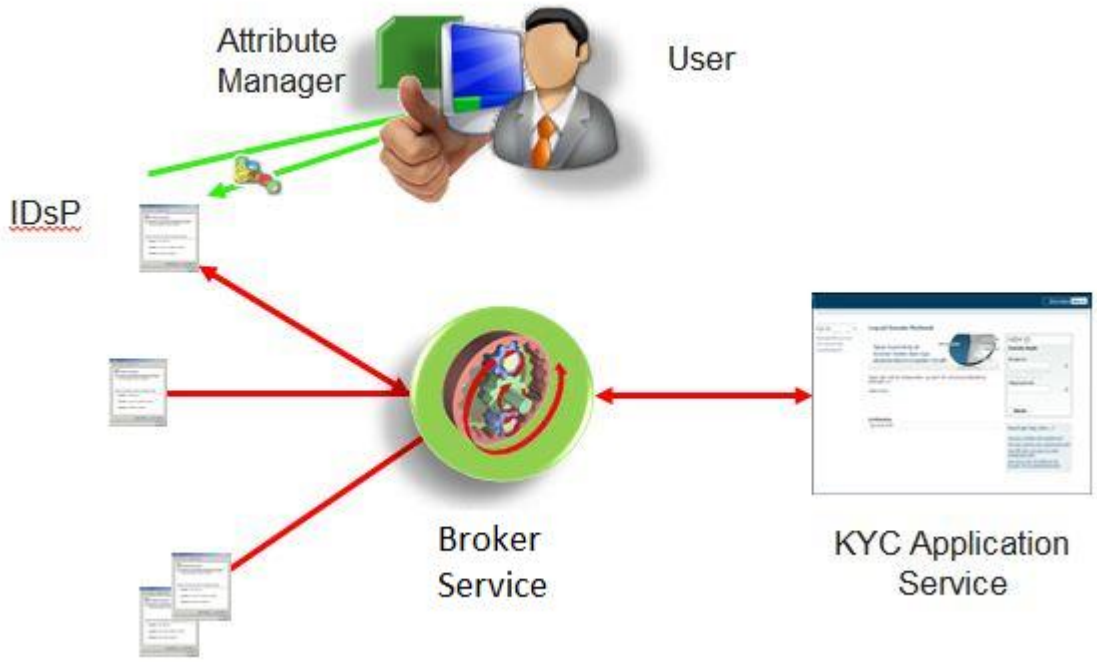

Figure 2 Process architecture

5.3.2 Communications with eID Client

There are a number of architectures that could be used at the eID client to carry out this process, depending on the capability and program storage of the client device. If a common client desgn is desired, then the smallest client possible must be the standard. In this case, it is unlikely that there could be sufficient space available to provide the flexibility in capability needed to handle all the varied use cases.

Therefore it is assumed here that the eID client will be merely an authenticator and common interface to code residing at the central FutureID components. (Broker Service or Trusted Repository).

This is Out of Scope for this document. However, these are typical of the communications expected.

5.3.2.1 Request: Received via FutureID

The broker service (or other central component) receives a request from the KYC Application to provide one or more signed declarations (9.2.1) or affirmations (9.2.2) for the authenticated eID. For scalability

this should not be communicated directly to the user client. This is dispatched to the client and takes the form of an ISO 20022 XML message.

5.3.2.2 Response: Signed identity attributes (declarations/affirmations) sent to KYC Application through FutureID

The attributes (declarations/affirmations) are dispatched through FutureID to the KYC Application and takes the form of an ISO 20022 XML message.

5.3.2.3 Response: Signed URLs of identity attributes sent to KYC Application through FutureID

The attributes (declarations/affirmations) are not dispatched, but only their unique URLs which are then used by the KYC Application to retrieve directly and takes the form of an ISO 20022 XML message.

5.3.2.4 Error Conditions

There would be a series of errors corresponding to each communication and step with partial or complete failure of the transaction or business process

5.3.2.5 Error Messages

Error messages should use a library to display in the appropriate language to all parties in the process

5.3.2.6 Error Recovery

There should be recovery modes to deal with roll-back or fail/stop.

5.4 Requirements

Nr.	Description	Desirability	Mapped to
R5.1	Secure login	MUST	G1
R5.2	Restrict request to authenticated relying party	MUST	T1
R5.3	Receive XML (schema to ISO 20022) from relying party	MUST (OPTIONAL FOR eBAM)	T2
R5.4	Receive other protocols (which may be used as an alternative to ISO 20022, such as SAML2) from relying party	SHOULD (OPTIONAL FOR eBAM)	T3
R5.5	Forward XML documents to appropriate attribute management system	MUST (OPTIONAL FOR eBAM)	T5
R5.6	Forward other protocols (which may be used as an alternative	SHOULD (OPTIONAL FOR	T3

	to ISO 20022, such as SAML2) from relying party to appropriate attribute management system	eBAM)	
R5.7	Manage any errors in the XML or other protocols received	RECOMMENDED (OPTIONAL FOR eBAM)	T4
R5.8	Receive XML from attribute management system	MUST	T2
R5.9	Receive other protocols (which may be used as an alternative to ISO 20022, such as SAML2) from attribute management system	SHOULD	T3
R5.10	Forward XML to specified relying party	MUST	T5
R5.11	Forward other protocols (which may be used as an alternative to ISO 20022, such as SAML2) to specified relying party	SHOULD	T3
R5.12	Manage any errors in the XML or alternative protocols received	RECOMMENDED	T4
R5.13	Display transaction error messages	SHOULD	G2
R5.14	Secure logout	MUST	G1
R5.15	Update transaction log either locally or at a remote location	RECOMMENDED	G3

6. Direct Democracy

6.1 Use Case Description

6.1.1 Background to the "Electronic Direct Democracy" use case

Electronic direct democracy (eDC) refers to the deployment of the possibilities provided by electronic communication means for the shaping of opinions, drafting of decision documents and deciding on such documents.

In times of low voter turnouts of political elections, missing civil engagement and widespread political disinterest, a broader, better, and more vivid participation in opinion building and decision processes on any level of social life can be considered a value in itself. Involvement of all societal parties concerned is necessary independent of the level of importance on which the decisions shall be made—from small clubs or private entities to major decisions shaping the political future of a nation to at least partly revert the process of growing disinterest in politics and societal processes. Also electronic democracy could allow the inclusion of groups that are now factually excluded from discussions. Beyond eInclusion for persons with some kind of physical or mental disability, eDC could also allow better participation of factually hindered persons such as poorer members of society not able to afford travel to meetings of organisations or parents of young children which are factually excluded from physical presence at debates unless some kind of childcare is available, which is typically missing in the evening hours. Online participation systems could aid these groups of persons to participate in relevant decisions and preceding discussions.

The use case presented here explicitly excludes the area of public elections of representatives for parliaments and other political bodies. Generally we acknowledge that the people's decision about the persons that will represent them for a period requires a very high level of secrecy during the voting process and transparency in the process of determining the outcome of the vote by openly counting ballots. While the anonymous and pseudonymous authentication methods that may be suggested within FutureID for polling systems may be leading the path towards electronic means fulfilling some requirements for public elections, but such formal elections are not object of this use case. Proposing a system that could fulfill the conflicting requirements may easily cover the work plan for another Integrated Project. E.g., regarding the necessary full transparency which requires effective means for an examination of the voting process due to the public nature of elections and at the same time regarding integrity preventing the manipulation of the results for such systems and confidentiality forbidding the retraceability of each citizen's voting behaviour as they have been demanded, e.g., by the German Bundesverfassungsgericht (supreme constitutional court).

But also polls, internal elections and processes for opinion building and discussion with lesser impact and importance than public elections have requirements regarding the confidentiality and unlinkability of votes on one hand and transparency of the processes on a very high level. Therefore eID systems

supporting eDC may be a key enabler if they could provide the variety of necessary features needed for the authentication process on such systems. Such systems highly vary regarding the targeted area for their deployment with some having a much higher demand on transparency and trustworthiness than others, e.g. where a poll has direct influence on a decision the factual influence is much greater and with it the according requirements than polls that merely provide a opinion towards a board of formally elected representatives. Even within the same setting some of the content addressed may have a greater sensitivity than the normal content depending on the circumstances so a discussion about the architecture of the town hall is in most cases considerably less sensitive than, e.g., a discussion about the legality of a same-sex-marriage being highly influenced by sexual preferences and ethical and religious beliefs. This use case description therefore cannot and does not provide requirements regarding functionalities of eID systems valid for all possible types of participation and voting systems let alone suggest solutions for all problems in this area. FutureID will attempt to provide a high-level set of requirements for both types of systems. Please refer to the upcoming legal deliverable on privacy requirements (planned as D22.3) which will elaborate on a system of data protection and security protection goals allowing an assessment of the potential conflicts as shown in Figure 3. For further details please refer to [ZwiHan12].

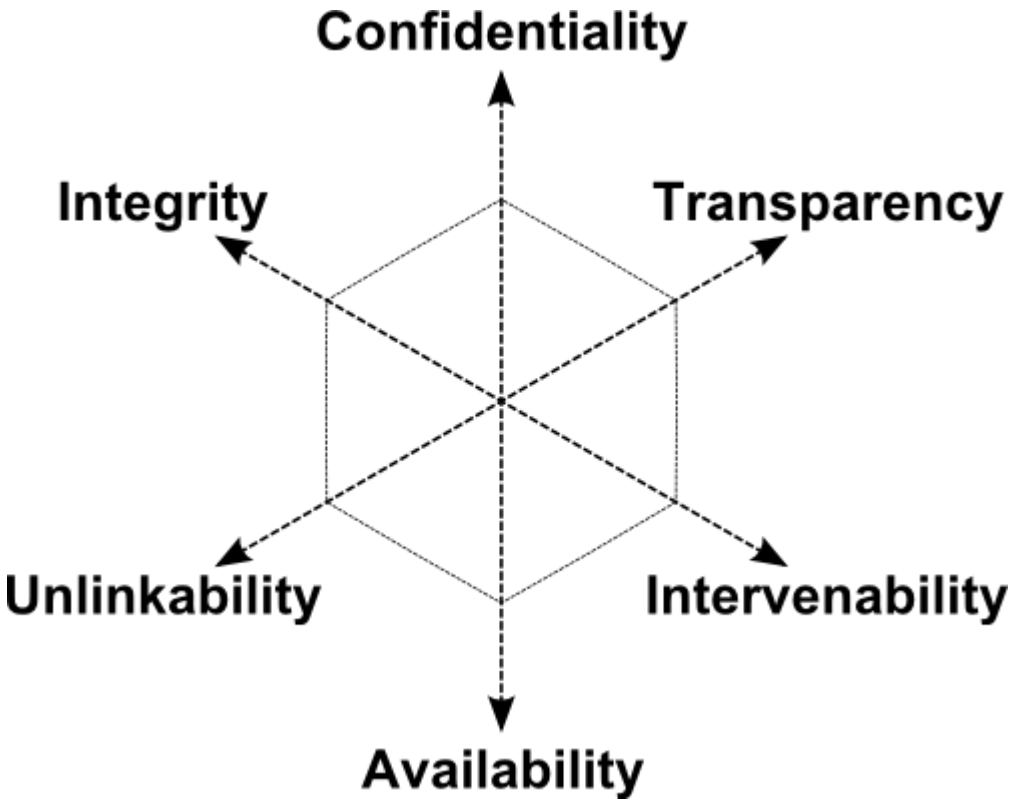

Figure 3 Potential requirement conflicts

6.1.1.1 Online Surveys

In the private sector, organisations from private clubs to large joint stock companies have built a wide variety of methods to find joint decisions with as many different requirements regarding the participation. Some of these are based or influenced by legal requirements, e.g., the participation rights of stock holders in corporations others are purely based on joint decision of their members. To break down complexity we compare such participation methods to surveys in general. Surveys may include boards allowing participants to state opinions and reflect the voting aspect of the aforementioned political discussion board. The preceding discussion may be anonymous in many cases requiring authentication only insofar as the membership to the group of eligible participants. For the voting process, on the other hand, a series of rules and restrictions may apply, e.g., that participants may be allowed to cast only one vote or further. For surveys often the person setting up the survey may determine the requirements for participation usually including:

1. Membership of a given group
2. Participating only once or up to x times
3. Casting one or more votes per option
4. Delegation of the power to participate to another entity is allowed or not and requirements for the delegation, e.g. that the delegate must be eligible member of the group of participants
5. Verifying additional attributes not genuinely stored on eID such as being a member, being a registered lawyer, student etc. (third party issuers)

6.1.1.2 European e-Petitions and European citizens' initiatives

European e-Petitions and European Citizens Initiatives Petitions are a possibility for citizens to get involved in the political process during parliamentary terms and on specific topics. The different national rights for petitions or initiatives foresee varying requirements and have a variety of legal consequences, but in many cases the result is not binding for the legislative body addressed.

We here focus on the petition to the European Parliament which is foreseen in Art. 227 of the Treaty on the functioning of the European Union.[1] Details are laid down in Rules 201-2013 of the European Parliament's rules of procedure [1]. According to these rules every "citizen of the European Union or natural or legal person residing or having its registered office in a member state" has the right to address a petition to the European Parliament. The requirements for addressing the European Parliament with a petition may be derived from Rule 201 as follows:

- Petitioner is eligible as
  - petitioner is a citizen of the EU,
  - or natural or legal person residing in the EU,
  - or legal person or having its registered office in a member state

[1] Consolidated Version of the Treaty on the functioning of the European Union, online: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:C:2008:115:0047:0199:en:PDF.

    - otherwise the petition may nevertheless be considered by the committee, on its own decision.
- The petitioner is affected by the subject matter directly.
- The petition shall show the **name, nationality, and permanent address** of (each) petitioner.
- If petition is supported by several petitioners, these shall **appoint representative and deputy representative.**
- Petition may be **withdrawn** by the petitioner at any time / all petitioners for a petition submitted by a group of persons.
- If asked to by the petitioner, the Parliament must withheld the petitioners name to protect her or his privacy.
- The petition may be submitted in writing or via an online form.

A series of these requirements may be supported by eIDs making the process easier and faster to access. In particular for providing name, nationality, and permanent address of the petitioners eIDs may be used to actually gain these information in a verified and trustworthy quality. Having the identities verified would support the transparency principle according to which all petitions registered by the Parliament as a general rule shall become public documents including the name of the petitioner and the content of the petition, see. Rule 201 No 9 of the European Parliament's rules of procedure. However, without proper verification it would be easy to impersonate someone and the publicity of the documents may heavily compromise the impersonated, e.g. imagine a petition for same-sex marriage maliciously addressed to the Parliament under the identity of a catholic bishop or a conservative politician. Therefore it would be advisable to verify the identity of the petitioner. For this the existing national eID solutions may very well be deployed and on a European level unification is foreseeable in the future with the draft regulation on electronic identification and trust services for electronic transactions in the internal market eID Regulation.

On the other hand identifying petitioners would contradict the privacy respecting paradigm of Rule 201 Numbers 10 (anonymity) and 11 (confidentiality of the content). Here the authentication with most of the existing eID schemas in European member states will nevertheless reveal the identity to the Parliament. However, if a petition is allowed anonymously and confidential no vital reasons can be seen to not allow a first submission in an anonymous form. As a back-channel for further communication with the petitioner e-mail may serve the purpose, provided that the Parliament gives up on its practice to continue proceedings by letter[2]. For the authentication that the petitioner is either citizen or resident advanced authentication schemas may be used. The German national eID already allows to anonymously verify the age, or place of living of the holder. Authentication solutions deploying privacy preserving

[2] Any further correspondence concerning action taken on the petition will be communicated by post.", source: http://www.citizenhouse.eu/index.php?option=com_gov2ufaq&view=category&id=6&Itemid=117&lang=en#quest76

attribute-based credentials[3] (Privacy-ABCs) could allow verifying any type of attribute that may be issued by a trustworthy issuer. Privacy-ABCs would in addition offer the optional possibility to imbed the identity in an encrypted format during the otherwise anonymous authentication (inspection). This necessarily requires that the petitioner is properly informed about the additional optional transmission of the identifying information and that she gets presented the conditions under which her identity may be revealed. It also may become necessary in the course of proceedings to ask the petitioner to provide her identity, e.g. to verify that the petitioner is self-affected by the subject matter. In this case the identification procedure may nevertheless take recourse to existing solutions, communicate via e-mail with the petitioner or recourse to the inspection feature if the petitioner was properly informed about this reason within the inspection grounds.

Similarly to the petition, European citizen initiatives bring the requirement to authenticate participating citizens. The citizen initiative is based on Article 11 paragraph 4 of the Treaty on European Union[4]. Compared to petitions, an initiative is more demanding on the side of the organizers starting the initiative and those supporting it. But if at least 1 million citizens from different member states must support the initiative the organizers shall be given opportunity to present the initiative in a public hearing at the European Parliament with appropriate representation of the commission.

For filing the European initiative the details are laid down in Regulation 211/2011 on the citizens' initiative[5]. The privacy concern already mentioned above for petitions have been seen and addressed by the regulators and the regulation consequently contains a series of detailed data protection requirements, see Article 12 of Regulation 211/2011.

While the data protection requirements addressed the current solutions to protect the privacy of the supporters, better eID solutions could provide the same level of trust into the validation of the list of supporters while not collecting personal information that may be compromised or lost in the first place. Advanced privacy preserving eID solutions could allow citizens to anonymously submit statements of support. eIDs supporting Privacy-ABCs enable users to verify that the supporter is citizen of the particular member state and has only supported the initiative once. In general more information about the identity of the supporters is not needed to verify the list of supporters. Depending on the European federation

[3] For details and latest research see the results of the European project Attribute-based Credentials for Trust, ABC4Trust, http://www.abc4trust.eu.

[4] A consolidated Version of the Treaty on European Union is available online: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:C:2012:326:0013:0046:EN:PDF

[5] Regulation (EU) No 211/2011 of the European Parliament and of the Council of 16 February 2011 on the citizens' initiative, online: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2011:065:0001:0022:EN:PDF

systems for the national eIDs or the availability of an European citizen card in the future may even make the involvement of the member states obsolete.

Participating in politics just with by swiping an eID at one's personal computer might invite more persons to participate in the political debate. In the view of the authors a broad participation in democratic processes is a value on its own worth being supported.

6.1.2 Goal of the Use Case

The electronic Direct Democracy use case will point to possibilities how eIDs may be deployed for a better participation in participation systems. It will show requirements that need to be fulfilled, in particular with respect to anonymous and pseudonymous but yet properly authenticated participation in such systems.

6.1.3 Regulatory Considerations

The regulatory requirements vary broadly depending on the detailed sub-use case to be considered. However, requirements for participation often flow from the legal framework and may include:

- Required attributes for participants, e.g. being member of a club, being registered barrister, being citizen of a particular municipality or having a particular sex or age.
- Number of votes that can be cast.
- Whether a once given vote may be changed or revoked.
- Requirements for delegating votes, e.g. whether this is possible at all, attributes of the delegate, such as being member of the organisation itself, etc.

6.2 Requirements

Nr.	Description	Desirability	Mapped to
R6.1	For votes and polls it must be possible to determine that a particular person has participated already	MUST	T6
R6.2	The client should support as many different eID services as possible	MUST	T7
R6.3	The client should run at any technical customer platform available (PC, tablet, smartphone, ...)	MUST	T8
R6.4	When attesting anonymous attributes the client must provide a graphical interface that clearly illustrates what information the recipient of the attestation will get to know	MUST	G4
R6.5	The user interface of the FutureID client must be designed so that it can be	SHOULD	G5

	used by elderly, young, or physically or mentally impaired people without restrictions		
R6.6	While generally security means for the protection of abuse of the client should be done on the side of the eID infrastructure the FutureID client should nevertheless be protected so that unauthorized persons are not able to use (i.e. misuse) the eID without permission (e.g. grabbing a stored copy of a software certificate from a stolen smartphone)	MUST	T9
R6.7	The FutureID client should support selective activation and deactivation of a user's different eID tokens for use in FutureID contexts (e.g. to block access from a stolen/lost smartphone)	MUST	T9
R6.8	One FutureID user must not be able to read the votes/survey answers of other participants without permission (e.g. local copy of inputs, or client in kiosk mode in public places)	MUST	G6
R6.9	One FutureID user must not be able to change the votes/survey answers of other participants without permission	MUST	G6
R6.10	A user should be able to easily configure the time, circumstances, and means of communication by which a survey/vote operator is allowed to contact him during a survey/poll	MUST	G7
R6.11	The client should support a communication channel between the responsible person for the survey (server) and the user.	SHOULD	G8
R6.12	A user should be able to easily configure the time, circumstances, and means of communication by which a survey/vote operator is allowed to contact him after a survey/poll	MUST	G7
R6.13	The FutureID user client should support the delegation features of the underlying architecture: • Delegates should know whom they are currently acting for, how many votes they currently cumulate, easily select whether they act as delegate or on behalf of themselves. Where the identity of the delegator is not necessary to be known to the delegate, anonymous delegation should be possible. • Delegators should be supported by viewing whom they have delegated their rights to. If foreseen in the poll, the delegator should be able to claim back his rights and where desirable overwrite the	SHOULD	T10

"own" vote done by the proxy in the poll.
- Delegators should be supported revoking the power from the delegate.

R6.14	A user should not be able to provide false values for attributes that are required for determining eligibility for participation in a certain survey/poll. It must, however, be possible to not send a value for a required attribute – usually with the consequence of not being able to participate.	MUST	G4
R6.15	A user should be able to participate in two separate polls/surveys in such a way that the poll/survey operators cannot determine that the user was the same for both surveys/polls, e.g. by means of anonymization or pseudonymization	MUST	G10
R6.16	The user client should enable the user to determine what data is stored onto her eID tokens	MUST	G11
R6.17	The user client should show the attributes and attribute values before sending them to the relying party e.g. on a confirmation screen	MUST	G4
R6.18	Where supported by the survey or poll, participation should be possible pseudonymously or anonymously, e.g. only verifying the relevant attribute for participation and the fact that one has not yet participated	SHOULD	G10
R6.19	History function: The FutureID client should be able to show a history of recent transactions of the active user, containing all details of these transactions (e.g. attributes attested, information disclosed, context of use, identity of involved entities). Note that this feature needs a separate protection (see Hansen et al., 2009).	SHOULD	G13
R6.20	If one user changes the votes/survey answers of another user with permission (delegation), that change must be announced to that other user as soon as possible with all change details (i.e. values before change, values after change, date, time, identity of changing user) unless the other user is the delegate and the initial participant was her delegator	SHOULD	G14
R6.21	A user should be able to easily configure which other users are allowed to read/change her votes/survey answers	SHOULD	T10
R6.22	A user should be able to selectively delete data that was stored on her eID	SHOULD	G13

	tokens by past polls whose deadlines expired already		
R6.23	It must be guaranteed that the authentication/evaluation of eligibility for participation in a survey/poll is strictly separated from the votes/survey answers given.	SHOULD	G10
R6.24	If a survey/poll is performed under pseudonym, linkage of a user's pseudonym to her identity may only be disclosed to other entities with consent of the affected user or must not be possible at all (to be set by the survey/poll operator)	SHOULD	G10
R6.25	To prevent extortion or buying of votes it should be possible for voters to change their vote after they have voted while the survey is still open for participation (without identification of the person voting of course!)	OPTIONAL	G15
R6.26	The FutureID client should run in an energy-efficient mode	OPTIONAL	T11
R6.27	A user should be able to selectively pause, interrupt, or delete any ongoing communication with any of its eID tokens or the FutureID client (if in doubt about illegitimate communication), without disrupting the FutureID client's or eID token's integrity or operability	OPTIONAL	G4
R6.28	If a survey/poll is performed completely anonymous, a user should be able to plausibly deny her participation at this survey/poll	OPTIONAL	G10
R6.29	If a survey allows changing of answers within its duration the user may establish a link to the own answers given but not any entity on the side of the server side. For the user client follows that it must support a pseudonym feature allowing secure pseudonyms.	SHOULD	G10

7. Third-Party Issuers

7.1 Use Case Descriptions

7.1.1 Overview

Paper-based certificates, licences, attestations, and tickets are widespread used in our everyday life. They differ in their frequency of use, physical characteristics and complexity. For instance, in the tourism sector licences for fishing or lift passes for skiing are used. Further examples are tickets for public transport and electronic prescriptions.

Such credentials are generally issued by a non-governmental third party that must manage and operate the issuance, the verification and the transmission medium. That causes high costs, security issues and reduced convenience for the user due to different and incompatible systems.

7.1.2 Use Case

Fishing in Germany is regularized and requires passing a fishing test and a fishing license for the particular areas. The fishing test must be passed only once and is valid for a lifetime. The certificate for a passed exam is issued by any municipality in Germany. In addition, a temporary fishing license for a particular area must be bought by local municipalities. Both, the certificate and the license must be shown for inspection.

The issuance and verification process is very inconvenient for both parties. People must carry their identity card, fishing certificate, and fishing license to go fishing. The paper-based certificate and license may get lost, destroyed, or just forgotten. The verification of the license should be easy. It is issued by a local municipality and only valid for the local area. Hence, the physical characteristics of such a license should be common to the inspector. However, fishing certificates are issued by every German state with different laws, which makes is hard to verify.

To make the process more reliable, the certificate and the licence should be replaced by a digital credential. That would improve the issuance and the verification. In addition, it should be possible to store the credentials on mobile devices or identity cards. FutureID can provide the infrastructure and the required client and server components to enable such functionality.

7.1.3 Technical Architecture

7.1.3.1 Presentation

The presentation of the credential during an inspection can be done in a graphical way, for instance, in form of a QR code or barcode. This presentation form is well-known from mobile ticking in public transportation or check-ins on airports and can easily be established if the third party issued credentials are stored on mobile devices.

In the case of storing credentials on a smart card, which has no graphical interfaces in general, the credentials must be transmitted to an inspector's verification device. Such a device must equip with necessary communication interfaces for receiving the credentials and performing verification protocols. Depending on the type of verification protocol additional interfaces, for instance, a mobile Internet connection to query OSCP requests, are required.

7.1.3.2 Issuance

Each third party issuer should operate its own Certificate Authority (CA). To take advantage of the existing PKI infrastructure used on the Internet, the third party issuer should obtain an intermediate CA certificate by an established company. The data structures of X.509 certificates according to RFC 5280 should be sufficient for the issued credentials. Further information should be added through X.509 certificate extensions.

7.1.3.3 Validation and Revocation

Nowadays certificate validation and revocation is based on Certificate Revocation Lists (CRL) and the Online Certificate Status Protocol (OCSP). CRLs are issued periodically by the CA and contain the serial numbers of the revoked certificates. The validation is performed by checking the expiration date and locking up the serial number of the certificate in the CRL. If the serial number is not in the list the certificate is valid, otherwise the certificate is revoked. Due to the fact, that CRLs are issued periodically the certificate's status might be outdated.

OCSP is a protocol to query the current status of the certificate by the CA at the current time. The request contains the serial number of the certificate and the signed response denotes the status. Each certificate validation requires one OCSP request and each response must be signed by the server. That implies message overhead and complex operations caused by the signature as well as a huge number of requests.

In general, the revocation is managed by the same organization that issues the credentials. Hence, third party issuers are also responsible for providing revocation status of their credentials. Such revocation information must be public if the credentials should be verified by other organizations. In the case of the Third Party Issuers there are some issues using CRLs or OCSP. To provide information about the certificate status via OCSP the issuer must operate the necessary infrastructure. That might not be feasible for small third party issuers. Verifying certificates using OCSP also requires a connection to the Internet or the issuer's infrastructure. In the case of performing verification on the way, for instance, checking the fishing licence at a lake, the request can be realized through the mobile network, but that requires appropriate devices and creates extra costs. Hence, OCSP should be preferred for credentials that rely on real-time validation to prevent misuse. CRL are only updated and issued periodically, which might not be sufficient in all use cases. Hence, real-time validation and revocation information are not available. Certificates for websites or encrypting e-mails are typically revoked because the corresponding private keys are compromised. In the use case Third Party Issuers the issued certificates are more similar to credentials that must be revoked if they get stolen. The credential is stored on a smart card or mobile

device that is equipped with an access control itself (e.g., a PIN code). That should protected the credentials until the loss is reported and the CRLs are updated.

In the Third Party Issuers use case the credentials are generally issued for a specific period of time. For instance, a fishing licence is valid for a fishing season and a skiing pass for a few days or weeks. Hence, revocation is not so important and verification based on the expiration date should be sufficient. Furthermore, revocation information issued by CRLs should be appropriate.

7.1.3.4 Binding

In some case it may be necessary to bind credentials on particular device. That can be archived directly or indirectly.

In the case of an identity card the binding can be done indirectly. We consider a credential stored on an ID card and including the owner's first and last name. During an inspection the owner presents his ID card to the inspector, who is able to compare the name of the credentials with the name printed on the card surface. A special binding is not necessary, because it is done by using an ID card as a transport medium.

Direct binding can be done in many different ways, for instance, encryption, signing or constant serial numbers. In the case of the fishing use case, for instance, the certificate and the license contain the same serials number, which enables to verify the binding between both credentials.

7.2 Requirements

Nr.	Description	Desirability	Derived Nr.
R7.1	Support for the Identity Management Protocols SAML 2.0 and OpenID 2.0 for exchanging authentication data.	SHOULD	T12
R7.2	Provide interfaces and data structure for protocols to support various authentication protocols. Authentication protocols are used to verify the legitimate user or owner of the credential. Verification protocols are used to verify the validity of the credential, for instance, OCSP or simply fetching CRLs from a URL.	MUST	T7
R7.3	Provide interfaces and data structure for protocols to support various verification protocols. Verification protocols are used to verify the validity of the credential, for instance, OCSP or simply fetching CRLs from a URL.	MUST	T7
R7.4	Support CardInfo files (CIF) according to CEN TS 15480 to support various	MUST	T13

types of credentials.

R7.5	Support for selecting and managing credentials by the user.	SHOULD	G11
R7.6	Support for binding credentials, for instance, binding to other credentials, device, or eID application.	OPTIONAL	T15
R7.7	Support for transmitting or exchanging credentials, for instance, via e-mail or NFC (Android Beam).	OPTIONAL	T14

8. eSigning and Validation

8.1 Use Case Descriptions

8.1.1 Background to Use Case

The Business Process Management re-engineering and digitisation of a complex workflow is acknowledged as a sure method to reduce costs and increase efficiency. Often workflows require a signature to legally agree or confirm the process at some point. When the workflow is digitised, the application of the signature in the digital domain require the actors in the process having a digital signature capability. Otherwise the workflow would need to revert to a paper methodology so that an ink-based signature can be used.

To avoid such media-breaks, the FutureId-Client will need to support as many digital signing technologies as are available with a special focus on the European market.

8.1.2 Use Case Examples

8.1.2.1 Granting of a Bank Loan

In this real case example there are a number of documents required to complete the application for a bank loan (depending on country and bank, so e.g.):

- Application
- Guarantor
- Contract
- Terms and conditions
- Consumer Procection Information sheet

These documents will need to be signed by the customer and by the bank. Both, customer and bank, may be represented by multiple persons:

- The customer may e.g. be a family requesting a loan which is only granted if both family representatives are made liable for that grant, so both have to sign
- The bank may require signatures from multiple persons, depending on their procedures and the roles of the employees involved.

These signatures will typically need to be applied in a certain order and from different individuals who may have to have their identities validated for compliance. In this case real examples have shown savings between €50 and €100 per application and each large bank may process many thousands of applications per week.

For the case of 10,000 applications per month, the equivalent annual saving can be up to €12 million per year. It is clear that the more applicants that can use this application, the greater the savings.

Goal of the Use Case

The goal of the use case is to:

- Demonstrate the practicality and economics of increasing the number of acceptable eIDs and the use of their associated digital signatures in a typical 'generic' transaction in either a business-to-business case or a business-to-consumer case.
- Determine the priority of addressing different signing methods
- Explore the legal and trust issues arising.
- Incorporate into a real workflow or BPM case.

8.1.2.2 eGovernment Services Use Case

This use case reflects usage of digital signatures in eGovernment. Typically citizens need to send an document (application-form, tax declaration or similar) to an eGovernment service. Such documents will need to be signed by the citizen. Reciprocally the citizen will receive documents from eGovernmental services that have been signed, either by an officer or an automated service using an "administrative signature"

This includes cross-border applications, where signatures from foreign governmental signatures need to be verified and documents have to be signed in a way acceptable by a foreign service. This may include cross-border authentication using STORK (resp. STORK 2.0), where cross-border recognition of electronic identities has been put in place. STORK 2.0 is also working on cross border digital signature issues.

8.2 Regulatory Considerations

- Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures.
- Action Plan on e-signatures and e-identification to facilitate the provision of cross-border public services in the Single Market" [COM(2008) 798 final]
- Commission Decision 2003/511/EC of 14 July 2003 on the publication of reference numbers of generally recognised standards for electronic signature products in accordance with Directive 1999/93/EC of the European Parliament and of the Council [Official Journal L 175, 15.7.2003].
- The impact of the 4th June 2012 eSignature proposals on the use of 'notifiable and non-notified e-signatures' for private-sector use.
- Electronic Services Directive (2006/123/EC) defining baseline signature format profiles

8.3 Technical Architecture

A high level view of the FutureID scheme integration into a workflow is shown in Figure 4. In this case the document(s) (in XML) are introduced for signing. The FutureID component 'normalizes' the signature to match the required signing method.

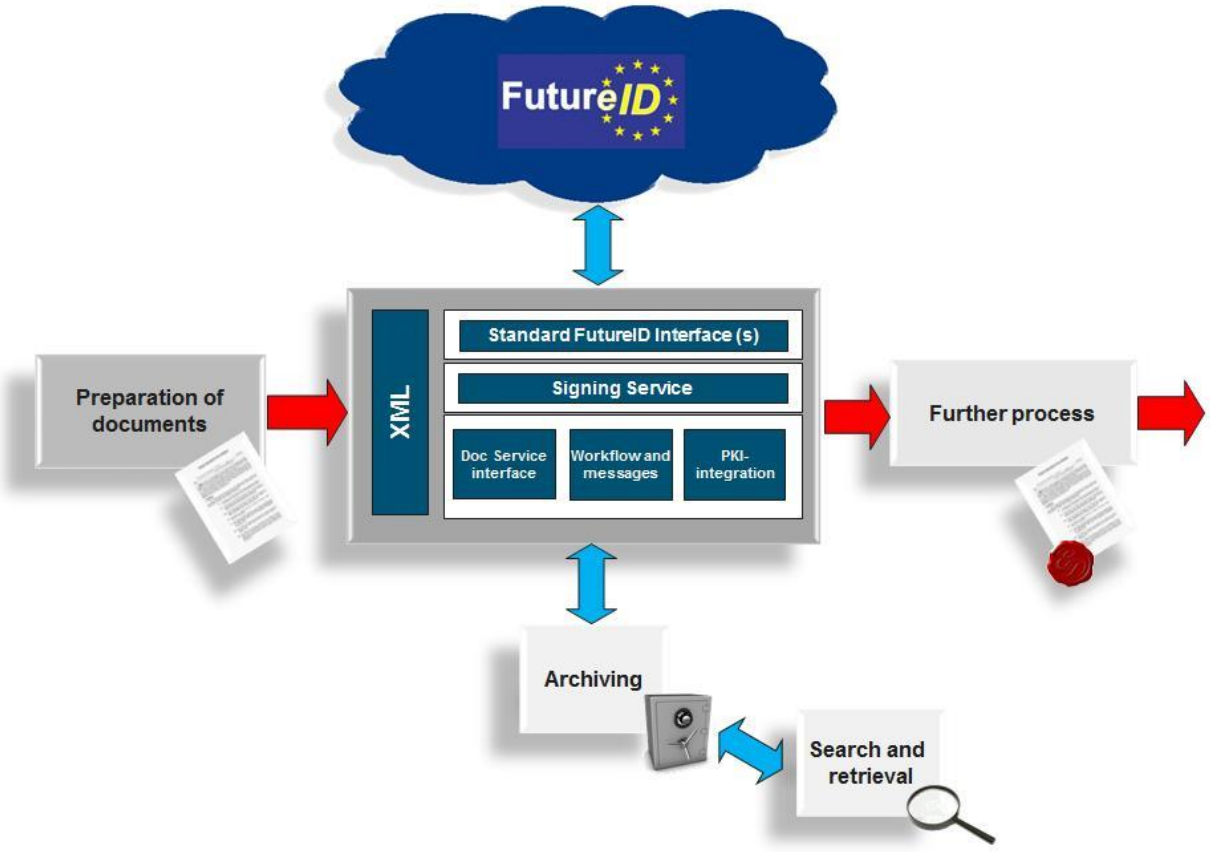

Figure 4 FutureID integration into a workflow

8.3.1 Documents

While there is no single standard electronic format for electronic contracts, PDF and XML are generally usable formats that also are heavily used in practice. Within FutureID, no restrictions on the document format is necessary. It is advisable though that there is a displaying software available that allows the user to view the document signed properly (WYSIWYS-principle).

8.3.2 Signature Formats

Signature formats are typically closely related to the document formats used. For a use case covering electronic contracts, using qualified electronic signature will be advisable if not required. Thus adequate formats will need to be used.

Document Format Signature Format

PDF PAdES

XML	XAdES
Other	CAdES, XAdES

8.3.3 Signing Tokens

This use case involves all sorts of users in different countries having different means to produce electronic signatures:

- Software signatures
- Smart Cards (bank cards, citizen cards, private smart cards)
- Mobile Signatures using secure elements on mobile phones
- Mobile Signatures using server signing

8.3.4 Signature Validation

Contract signing also requires validation of signatures produced by other parties. Besides the pure format- and crypto-handling, there are trust issues involved. The tools used must allow for flexible and easy handling of trust anchor management processes. This includes

- Manual setting of trust to different CAs
- Recognition of European CAs that issue qualified certificates
    - This includes support for Trusted Lists, a mechanism used by the commission to distribute information on trust services in Europe.
- Web-of-trust type trust handling (like in PGP) where no central CA is used
    - this will require support for signed keys (as in PGP) and TSLs as a means to distribute trust information. This allows e.g. people to import trust settings configured by people they trust and they know to be more knowledgable than themselves.

8.3.5 Other signature schemes

Conventional digital signatures require the signer to be available during signature creation, e.g., when a contract is signed. To overcome this limitation, the concept of proxy signatures has been introduced. Basically, a proxy signature scheme allows an entity (the delegator) to delegate his signing capabilities to another entity (the proxy) that can then construct signatures on behalf of the delegator. This concept has seen a considerable amount of interest since then. In a practical application, the delegator may not want to give the proxy the power to sign any message on behalf of the signer, the delegation by warrant approach was proposed. Here, a signed warrant is used to describe the delegation. Thereby, any type of security policy may be included in the warrant to describe the restrictions under which the delegation is valid. E.g. the bank employee may be delegated the right by the administration to sign grants for loan that follow a specific structure with a limited set of options, but not just any grant possible.

Recently, we defined the novel concept of a blank digital signature scheme. Here, an originator, i.e., the bank adminstration delegating signing permissions to the employes of the grant department, can define and sign a message template, describing fixed parts of a message as well as several choices for

exchangeable parts of a message. One may think of a form with blank fields, where for such fields the originator specifies all the allowed strings to choose from. Then, a proxy (the employee of the grant department) is given the power to sign template instantiations of the template given by the originator by using some secret information. The resulting message signature can be publicly verified under the originator's and the proxy's signature verification keys. Thereby, no verifying party except the originator and the proxy learn anything about the "unused" choices from the message template and, consequently, about the template given a message signature.

8.3.6 Process

A typical signature process will be

- One party signs a document. If there are multiple signers required, the business process will take care of managing that.
  - If a signed document needs to be signed by a further signer, the signer will need to validate any signatures on the document before signing.
  - A parallel or serial signature can then be applied be the signer.
- As soon as all signers have signed the document, a time stamp may be added to the signed document, depending on the business rules applied.
- It is typical that the signed document must be verifiable long after any of the certificates used in the signing process have expired. In this case, business processes may need to ensure the verifiability either by using long-term validation formats (AdES-A) or ensure this feature by different means, like storing documents, signatures and valdiation material in a long term archival system suitable for that purpose.

8.4 Requirements Summary

Nr.	Description	Desirability	Derived Nr.
R8.1	Support for CMS and CAdES BES/EPES	SHOULD	T17
R8.2	Support for XML and XAdES BES/EPES	SHOULD	T18
R8.3	Support for PDF-Signatures and PAdES BES/EPES	MUST	T16
R8.4	Support for long term forms (AdES-T, AdES-C, AdES-XL and AdES-A)	SHOULD	T18
R8.5	Support for ASIC	SHOULD	T20
R8.6	Support for multiple citizen cards and other tokens	MUST	
R8.7	Support for mobile signatures	MUST	T21

R8.8	Support for server side created signatures	MUST	T22
R8.9	No restrictions on type of documents signed	MUST	G16
R8.10	Supports different trust models (PKI, Web of Trust)	SHOULD	G17
R8.11	Support for other signature schemes (proxy signatures, blank signatures, blind signatures...	OPTIONAL	G19
R8.12	Support for Trusted Lists and Trust Status Lists as trust anchor distribution format	SHOULD	G17
R8.13	(signed) trust settings can be imported from other trusted users	OPTIONAL	G18
R8.14	Meaningful default settings, user ideally needs to configure only things he understands	MUST	G20
R8.15	Support for STORK 2.0 cross border signatures and delegation	OPTIONAL	T23
R8.16	Supports Validation with understandable (as much as possible) user feedback in case of failed validation	SHOULD	G21

9. Age Verification

9.1 Background

Date of birth and the age of the user is one of the major attributes certified and verified in the eID space. The major driver behind the use cases on age verification is legal compliance, as most European countries hierarchized age requirements for transactions. The legal regulations are usually part of the protection of minors and state that the transactions in question can only be performed, the goods in question can only be obtained if the citizen has reached a certain age. A typical example of transactions performed is agreeing to terms of services of Internet sites. A typical example of goods obtained is that minors may not obtain alcohol or tobacco. Thus, the major direction of the use cases in this section will be age verification according to legal requirements.

Given that these legal requirements are in place for the protection of minors, the protection of the minors privacy is a direct corollary. Privacy protection serves as protection against harassment, stalking and other crimes. The date of birth certified as basis for age verification is a sensitive piece of Personal Identifiable Information (PII). Depending on the size of the country in question, a combination of date of birth and location (Zip code or home town) can already be uniquely identifying. In any case, the date of birth is a quasi-identifier that decreases the size of the user's anonymity set tremendously. From this consideration springs a requirement to make the age-verification privacy-preserving, which means allowing service providers to verify the age without disclosing the certified date of birth.

It needs to be noted however that the date of birth of a user serves for identification of users in police and border control scenarios and as reference for lookup of citizen database entries. Therefore, there may be use cases in which a full disclosure of the date of birth is required. These cases fall under identification and are not strictly age verification discussed in this section.

We can structure age-verification use cases according to three factors:

- Age proof constraints
- Disclosure
- Online/Offline

9.1.1 Age Proof Constraints

Most cases of age verification require that the age of the user shall be *greater than* a given constant age. Most age verification for the protection of minors falls into this class. However, we see the similar use cases not only for legal requirements, but also for terms of services for sites, such as Facebook. Age verification of similar style may also be applied to social benefits for senior citizens.

The second class are age proofs that require that a user's age is *less than* a given constant value. In discussions with eID providers, use cases such as kids corners on Internet sites and social networks were proposed. In these cases, children shall be shielded from the incursion of adults.

The third class are range proofs, that is, for instance to prove that the user's age is between 20 and 30 years. These cases serve statistical purposes.

9.1.2 Disclosure

As noted before, the user's date of birth is sensitive Personal Identifiable Information (PII) that may lead to a small anonymity set or unique identification. Therefore, we can distinguish use cases into those that disclose the date of birth itself and those that keep it confidential only disclosing the age or even the truth of a statement about the user's age.

There are approaches in eID card proposals in Europe to certify the date of birth as part of the card's sensitive information, however not to disclose the date of birth itself in age verification. These cards achieve that goal without the use of Privacy-ABCs by proving the authenticity of the card and then disclose a so called *derived attribute* about age, e.g., ageGreatherThan18. In this case, the service provider checks the authenticity of the card and trusts the card to make a correct statement of the user's age without obtaining a cryptographic proof in the presentation token with it.

9.1.3 Online/Offline Use Cases

Most cases discussed for eID cards consider an online service provider, that is, a service provider with internet access. Consider for example an Internet vendor. However, European countries have proposed use cases with offline devices as well. In this case, the device has no connection to the internet (and is potentially only updated with a revocation lists). Use cases along those lines include vending machines for media or tobacco.

9.2 Use Case Descriptions

9.2.1 Age Verification for Legal Compliance

9.2.1.1 Obtaining Restricted Goods

Driver: Protection of Minors

Examples:

- Alcohol (Sale over the counter, verification at Point-of-Sale terminal)
- Tobacco (Tobacco vending machine)
- Medication (Internet Drugstore)
- Media (DVD rental service)

9.2.1.2 Performing Transactions

Driver: Maturity to enter contracts

Examples:

- Agreeing to terms of services
- Transactions with money (e-banking, auctions, ...)

Example Policy Statement:

User must prove that he or she has an age *greater than* 18 to complete the transaction.

9.2.2 Age Verification for Social Benefits

Obtaining social benefits tied to a citizen's age.

Driver: Social welfare

Examples:

- Reduced prices in museums and cultural organizations.

Example Policy Statement:

User must prove that he or she has an age *greater than* 65 to gain the benefit.

9.2.3 Range Proofs for Statistical Classification

Gaining a statistical classification of the user's ages for customer insight.

Driver: Customer base analysis

Example Policy Statement:

User must prove that he or she has an age in one of the intervals given by the classification.

9.2.4 Kids' Corner

Protected area in Internet services that is only accessible to citizens with a maximal age and certified social workers.

Driver: Protection of Minors

Examples:

- Walled social networks only accessible to minors.

Example Policy Statement:

User must prove that he or she has an age *less than* 16 to enter the kids' corner.

Observe that the use case may be inhibited by the fact the countries only roll out eID cards to citizen with a certain minimal age.

9.3 Requirements

9.3.1 High-Level Requirements

Nr.	Description	Desirability	Mapped to
R9.1	The age verification must allow to prove adulthood.	MUST	G12
R9.2	The age verification must allow disclosure of the user's date of birth.	SHOULD	G12
R9.3	The age verification should be privacy-preserving.	SHOULD	G12
R9.4	Data minimization, in particular, the date of birth should not be disclosed.	SHOULD	G12
R9.5	A preferable implementation will only disclose the user's age (age = 27) or even better the truth of an statement on the user's age (The statement "The user's age is *greater than* 18" is true).	OPTIONAL	G12
R9.6	The age verification should be anonymous.	SHOULD	G10
R9.7	As a side requirement, it is important that age verification proofs cannot be repeated many times by the terminal. If the terminal can send the eID many age verification requests with different age requirements it can box in the user's date of birth and eventually learn the date itself.	SHOULD	G24
R9.8	The age verification should be flexible enough to allow a variety of use cases by supporting *less than* statements.	OPTIONAL	G12
R9.9	The age verification should be flexible enough to allow statistical classification use cases by allowing interval proofs.	OPTIONAL	G12
R9.10	The age verification shall work with online as well as offline devices.	SHOULD	T24
R9.11	The policies and eID client should allow for informed consent in communicating to the user that an age verification is requested	SHOULD	G4
R9.12	The policies and information displayed by the eID client should be precise enough to communicate whether the date of birth is disclosed itself or a	OPTIONAL	G4

statement about the age only.

R9.13	The informed consent procedure at the eID client should allow an informed user decision on date of birth disclosure.	SHOULD	G4
R9.14	The disclosure of the date of birth should be opt-in disclosure of Personal Identifiable Information (PII).	OPTIONAL	G4

9.3.2 Low-Level Requirements

The access to Privacy-ABC such as U-Prove and Identity Mixer will be mediated through the ABC4Trust interface. Requirements of these Privacy-ABC technologies will impose a requirement to be compatible with its interface.

Nr.	Description	Desirability	Mapped to
R9.20	The age verification must be executable directly with the terminal (in online or offline scenario).	MUST	T25
R9.21	The age verification (with disclosure of date of birth) should be transmittable as SAML 2.0 attribute exchange.	SHOULD	T12
R9.22	The age verification (with disclosure of date of birth) should be transmittable with the STORK platform.	SHOULD	T23
R9.23	An age verification (with disclosure of date of birth) should be transmittable by OpenID.	SHOULD	T12
R9.24	An age verification (with disclosure of date of birth) should be transmittable with U-Prove.	SHOULD	T26
R9.25	An age verification (with disclosure of date of birth) should be transmittable with Identity Mixer.	SHOULD	T27
R9.26	An age verification proving fulfillment of an age class (e.g., age > 18) with U-Prove.	SHOULD	T28
R9.27	An age verification proving fulfillment of an age class (e.g., age > 18) with U-	SHOULD	T29

	Prove.		
R9.28	An age verification with presentation of an age class (e.g., age > 18) with Identity Mixer.	SHOULD	T30
R9.29	An age verification with presentation of with a variable interval specification on the age, e.g., as specified with the ABC4Trust policy language, should be transmittable with Identity Mixer.	OPTIONAL	T30

10. Secure Element related Requirements

10.1 Available Components

10.1.1 Secure Elements

In addition to the classical smart cards as a separate hardware device, other form factors of secure elements exist which support a similar or the same functionality as a smart card. These secure elements can typically be found in mobile devices where they support a large variety of mobile applications.

One of the most widely distributed secure elements for mobile devices is the SIM card which has evolved over the recent years to a **UICC** (universal IC card) that can host several applications in parallel. In this case the classical SIM application is one of several applications on the card. The development of Near Field Communication (NFC) and the increasing availability of NFC phones have further triggered the need to securely connect directly to the NFC interface, a capability that has been added to the UICC using the Single Wire Protocol (SWP). New applications can be loaded onto the UICC via the internet ("over-the-internet", OTI) or via the cellular network ("over-the-air", OTA) allowing the support of additional applications after field rollout.

While the SIM/UICC is owned by the network operator and is interchangeable, the same functionality can be located on an **embedded secure IC** which is owned by the device manufacturer. Since the element is not removable, all applications and credentials stored on it need to be manageable from remote. In principal, an embedded secure element could be used in addition to a classical removable UICC and could support additional security applications independent from the mobile network operator. While the electrical interfaces differ compared to the UICC the logical communication structure remains the same.

Another form factor for secure elements that offers independence from the network operators and the device manufacturers while still being removable is the secure **MicroSD card**. Since many mobile devices support this interface, a secure MicroSD card is a flexible and widely usable form factor. Service providers have the ability to develop their own and independent business models using this type of secure elements.

In a wider definition of secure elements a Trusted Platform Module (**TPM**) for stationary devices and the Mobile Trusted Module (**MTM**) for mobile devices can also offer certain security functionalities like the storage of secret keys, a trustworthy identity and the ability to audit the integrity status. Although they do not offer the same capabilities as secure elements, some limited applications may be envisioned using their security services. As an example, in the new Windows 8 operating system an MTM can be used to create a virtual smart card for secure log-on.

10.1.2 Trusted Execution Environment

Although not a secure element in the strict sense, the Trusted Execution Environment (TEE) for mobile devices offers security against a large range of software attacks on mobile devices. By providing a strict separation of execution environments, applications in the TEE can be protected against malware running in the normal ("rich") Operating System (OS) context. Mechanisms are implemented to protect the integrity of the TEE software itself with a secure boot process and to protect the confidentiality of the other assets within the TEE. The separation of the TEE and the rich OS is deeply supported by the mobile device processor architecture (e.g. ARM TrustZone or Intel TXT). The processor provides a dedicated hardware section to run trusted applications and to establish a trusted environment. Additionally, the access to shared resources (e.g. memory) is controlled and strictly separated.

A TEE can host trusted services for security critical applications like eID services, payment applications and digital rights management (DRM). In cases where tamper proof protection against physical attacks is required however, it will not fully replace a secure element with tamper proof hardware. Therefore, the combination of a TEE offering services on a medium security level with secure elements for the storage of the critical secrets is the most reasonable approach. In the FutureID context, a TEE could be used in combination with a secure element to securely store and access eID credentials on a mobile device.

Trusted Execution Environments are currently standardized by Global Platform.

10.1.3 Comparison of Components

The various components described above show many similarities in their technical properties and the level of provided trust but they also have some major differences. The most important difference can be found between the hardware-based tokens (smart cards, secure elements, other tokens) with tamper-proof hardware and the software-based trusted execution environment (TEE). While the TEE can protect secret data against most types of software attacks, it does not provide tamper-proof hardware and the resulting resistance against local attackers and side-channel attacks. Nevertheless, it can be combined with hardware-based secure elements to provide additional security.

The hardware-based tokens mainly differ in the form factor and the provided electrical interfaces. While smart cards are typically fabricated as plastic cards in the standardized ID-1 format (ISO/IEC 7810) and provide electrical contacts according to ISO/IEC 7816-2 or a contactless interface according to ISO/IEC 14443 (or both), the other tokens have other form factors depending on the actual application and can provide different interfaces (e.g. USB). Typically, these elements also use APDU commands based on ISO/IEC 7816-4. The security chip contained in these elements is tamper-proof and therefore offers a high resistance against hardware attacks that may be conducted by a local attacker being in physical possession of the secure element. In addition, these elements provide a strong protection against side-channel leakage by which an attacker could obtain confidential information without compromising the underlying cryptographic algorithm.

10.2 Relevant Standards

10.2.1 ISO/IEC 24727

The ISO/IEC 24727 series of standards defines programming interfaces for interactions between integrated circuit cards (ICCs) and external applications. A fundamental requirement for these standards is the compliance of the ICCs with the ISO/IEC 7816-4 standard which defines the data structure and command sequence of the cards. ISO/IEC 24727 mainly defines the interfaces for an interoperable implementation of the service access layer and the generic card access layer.

Within this definition, the service interface translates an action request into one or more generic requests and for the response translates one or more generic confirmations into an action confirmation. On the other hand, the generic card interface translates a generic request from the service interface into one or more specific requests for the ICC and in response translates one or more specific confirmations from the ICC into a generic confirmation for the service interface.

By specifying the interfaces, a layered architecture can be achieved with both layers being either part of a separate middleware or both being integrated into the same middleware. In the latter case, the internal communication between both layers is not specified. The interfaces are specified independent of the specific protocols that are required to establish a communication between the client-application and a card-application.

The generic card interface layer is specified in ISO/IEC 24727-2 while the service access layer is specified in ISO/IEC 24727-3. Both provide a more concrete specification of concepts developed in the ISO/IEC 7816 series, especially ISO/IEC 7816-4 for the data structure and commands, ISO/IEC 7816-8 for security operations, ISO/IEC 7816-9 for card management and ISO/IEC 7816-15 for cryptographic information.

One of the key functionalities provided by ISO/IEC 24727 is the capability to discover card applications available on an ICC, to obtain information about a card application and to establish a communication channel between the card application and the client application. The definition of the interface device layer (IFD) is out of the scope of this standard. The requirements for the IFD layer can be found in deliverable D 31.1.

10.2.2 OpenMobile API Service Access Layer

Since more and more applications are envisioned for mobile devices, the SIMAlliance - a non-profit association comprising the main actors of the smart card industry - has developed a concept how applications running in a mobile environment should access secure elements. As a result, an API specification has been published in 2011 which is independent of a specific platform or programming language and can therefore be implemented on any type of device and operating system. Currently, the main focus of implementations being developed lies on Android-based systems.

The API specifies interfaces for available services that cover functionalities like

- managing a communication channel
- transmitting commands and data to a secure element
- managing and performing PIN verification

- storing and retrieving sensitive data in the secure element
- assembling a list of available secure elements
- managing digital signatures

The OpenMobile API can act as an IFD layer and has therefore been described in Deliverable D 31.1 in more detail. One of the requirements for the FutureID client IFD layer states that it must support the OpenMobile API on mobile devices.

In addition to the IFD layer capabilities it also provides a service access layer which can directly support the eID services which are considered in this deliverable. The service layer provides a higher abstraction of secure element functions and thus makes them easier to use by application developers than the underlying transport API. The layer provides various APIs for different purposes, like

- file management,
- authentication,
- cryptographic services,
- application discovery,
- PKCS#15 functions and
- secure storage.

Further details can be found in the OpenMobile API specification (V2.03).[1]. The general architecture can be seen in the next figure (from [1]).

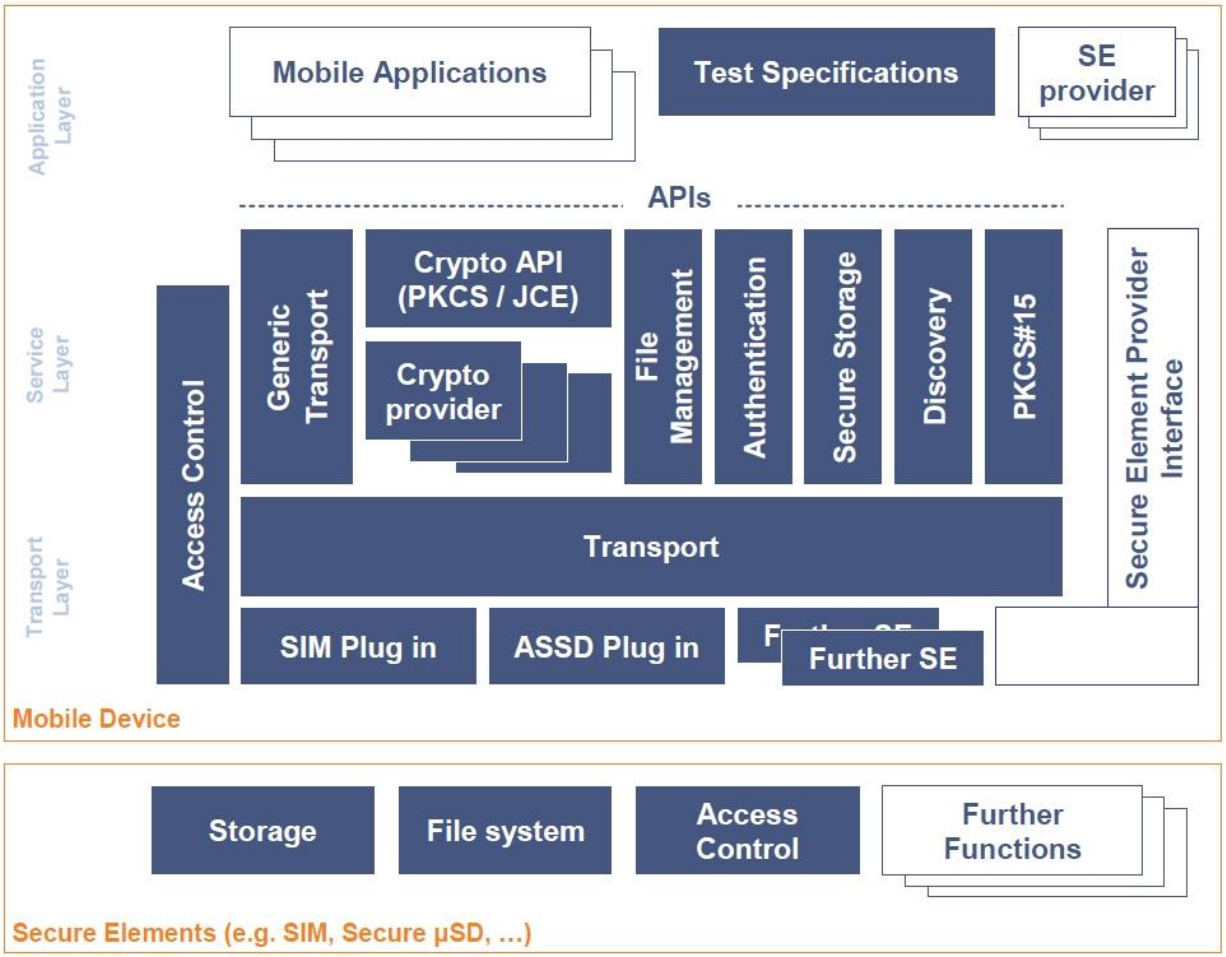

Figure 5 General architecture of the OpenMobile API

The concept of the Secure Elements Provider Interface allows adding new types of future secure elements, as long as the supplier provides the according interface. Therefore, the OpenMobile API ideally fits into the objective of FutureID to support all kinds of eID types and to be flexible enough to incorporate future technologies.

10.3 Requirements

Nr.	Description	Desirability	Mapped to
R10.1	The FutureID client eID service layer must comply with ISO/IEC 24727-2 and ISO/IEC 24727-3 standards.	MUST	T31
R10.2	On mobile devices, the Future ID client eID service layer must support the	MUST	T32

OpenMobile API.

R10.3	The FutureID client eID service layer should comply with the CEN 15480 standard.	SHOULD	T13
R10.4	The FutureID client eID service layer must comply with the CardInfo files defined in the CEN 15480 standard.	MUST	T13

10.4 References

1. ↑ [1.0](#) [1.1](#) SIMAlliance Open Mobile API Specification, V 2.03.

11. Derived Requirements

11.1 General High-Level Requirements

Nr.	Description	Desirability	Derived from
G1	Secure login and logout	MUST	R5.1, R5.14
G2	Display transaction error messages	SHOULD	R5.13
G3	Update transaction log either locally or at a remote location	RECOMMENDED	R5.15
G4	Informed consent: whenever information about a user is transmitted, the client must provide a graphical interface that clearly illustrates what information or properties are being transmitted. The user must have the opportunity to cancel the transmission without disrupting the FutureID client's or eID token's integrity or operability.	MUST	R6.4, R6.14, R6.17, R6.27, R9.11, R9.12, R9.13, R9.14
G5	Accessibility: the user interface of the FutureID client must be designed so that it can be used by elderly, young, or physically or mentally impaired people without restrictions	SHOULD	R6.5
G6	One FutureID user must not be able to read or change the votes/survey answers of other participants without permission.	MUST	R6.8, R6.9
G7	A user should be able to easily configure the time, circumstances, and means of communication by which a survey/vote operator is allowed to contact him during as well as after a survey/poll	MUST	R6.10, R6.12
G8	The client should support a communication channel between the responsible person for the survey (server) and the user.	SHOULD	R6.11
G9	A user should not be able to provide false values for attributes that are required.	MUST	R6.14
G10	Anonymity and pseudonymity: A user should be able to authenticate in an unlinkable way to different relying parties. Linkage of pseudonyms to real identities may only be performed	MUST	R6.15, R6.18, R6.23, R6.24, R6.28, R6.29,

	with consent of the affected user, or must be impossible.		R9.6
G11	The user client should enable the user to determine what data is stored onto her eID tokens.	MUST	R6.16, R7.5
G12	Minimal disclosure: A user should be able to disclose only those attributes that are relevant to the current transaction, or even merely that they satisfy a certain property. In particular, it should be possible to prove that a user's birthday is before or after a certain date.	MUST	R6.18, R9.1, R9.2, R9.3, R9.4, R9.5, R9.8, R9.9
G13	History function: The FutureID client should be able to show a history of recent transactions of the active user. The user should also be able to delete history entries.	SHOULD	R6.19, R6.22
G14	If a delegate exercises his delegation, there should be a way to inform the delegator of the delegate's actions.	SHOULD	R6.20
G15	To prevent extortion or buying of votes it should be possible for voters to change their vote after they have voted while the survey is still open for participation (without identification of the person voting of course!)	OPTIONAL	R6.25
G16	No restrictions on type of documents signed	MUST	R8.9
G17	Supports different trust models (PKI, Web of Trust, Trusted Lists and Trust Status Lists)	SHOULD	R8.10, R8.12
G18	(Signed) trust settings can be imported from other trusted users	OPTIONAL	R8.13
G19	Support for other signature schemes (proxy signatures, blank signatures, blind signatures...)	OPTIONAL	R8.11
G20	Meaningful default settings, user ideally needs to configure only things he understands	MUST	R8.14
G21	Supports Validation with understandable (as much as possible) user feedback in case of failed validation	SHOULD	R8.16
G23	The age verification must allow to prove adulthood.	MUST	R9.1

G24	There must be some mechanism to prevent repeated attribute disclosures or proofs by the same terminal, to avoid "boxing-in" of the user.	SHOULD	R9.7

11.2 Technical Requirements for Client eID Services

Nr.	Description	Desirability	Derived from
T1	Restrict attribute requests to certified and authenticated relying parties	MUST	R5.2
T2	Receive XML (schema to ISO 20022) from attribute management system and forward it to relying party for financial messaging	MUST[6]	R5.3, R5.8
T3	Forward other protocols (which may be used as an alternative to ISO 20022, such as SAML2) to and receive them from appropriate attribute management system and relying party.	SHOULD[6]	R5.4, R5.6, R5.9, R5.11
T4	Manage any errors in the XML or other (alternative) protocols received	RECOMMENDED[6]	R5.7, R5.12
T5	Forward XML (to ISO 20022 schema) to appropriate attribute management system or specified relying party.	MUST[6]	R5.5, R5.10
T6	For votes and polls it must be possible to determine that a particular person has participated already	MUST	R6.1
T7	The client should support various eID services and different authentication and revocation protocols.	MUST	R6.2, R7.2, R7.3, R8.6
T8	The client should run at any technical customer platform available (PC, tablet, smartphone, ...)	MUST	R6.3

[6] OPTIONAL for eBAM

T9	In case of loss or theft of the eID, unauthorized persons cannot (mis)use it without permission from the user. Users should also be able to request their lost or stolen eID to be deactived or revoked.	MUST	R6.6, R6.7
T10	The FutureID user client should support the delegation features of the underlying architecture. Delegates should know whom they are acting for. Delegators should see who they have delegated to and withdraw delegation.	SHOULD	R6.13, R6.21
T11	The FutureID client should run in an energy-efficient mode	OPTIONAL	R6.26
T12	Support for the Identity Management Protocols SAML 2.0 and OpenID 2.0 for exchanging authentication data.	SHOULD	R7.1, R9.21, R9.23
T13	Compliance with the CEN TS 15480 standard, in particular support CardInfo files (CIF) to support various types of credentials.	MUST	R7.4, R10.3, R10.4
T14	Support for transmitting or exchanging credentials, for instance, via e-mail or NFC (Android Beam).	OPTIONAL	R7.7
T15	Support for binding credentials, for instance, binding to other credentials, device, or eID application.	OPTIONAL	R7.6
T16	Support for PDF-Signatures and PAdES BES/EPES	MUST	R8.3
T17	Support for CMS and CAdES BES/EPES	SHOULD	R8.1
T18	Support for XML and XAdES BES/EPES	SHOULD	R8.2
T19	Support for long term forms (AdES-T, AdES-C, AdES-XL and AdES-A)	SHOULD	R8.4
T20	Support for ASIC	SHOULD	R8.5
T21	Support for mobile signatures	MUST	R8.7
T22	Support for server side created signatures	MUST	R8.8
T23	Support for STORK 2.0 cross border signatures, delegation, and age verification.	OPTIONAL	R8.15, R9.22

T24	Authentication and attribute disclosure shall work with online as well as offline terminal devices.	SHOULD	R9.10
T25	Authentication and attribute disclosure must be executable directly with the terminal (in online or offline scenario).	MUST	R9.20
T26	An age verification (with disclosure of date of birth) should be transmittable with U-Prove and Identity Mixer.	SHOULD	R9.24, R9.25
T27	An age verification proving fulfillment of an age class (e.g., age > 18) with a derived attribute named by the card without cryptographic proof for the age statement. Underlying credential may be mEAC based, Identity Mixer or U-Prove key binding.	SHOULD	R9.25
T28	An age verification proving fulfillment of an age class (e.g., age > 18) with U-Prove.	SHOULD	R9.26
T29	An age verification proving fulfillment of an age class (e.g., age > 18) with Identity Mixer.	SHOULD	R9.27
T30	An age verification with presentation of with a free-formed policy statement about the age should be transmittable with Identity Mixer.	OPTIONAL	R9.28, R9.29
T31	The FutureID client eID service layer must comply with ISO/IEC 24727-2 and ISO/IEC 24727-3 standards.	MUST	R10.1
T32	On mobile devices, the Future ID client eID service layer must support the Open Mobile API.	MUST	R10.2

12. Conclusion

We distilled a list of concrete requirements from six different use cases for the FutureID client that we considered. Our requirements include general high-level requirements as well as more specific technical requirements with respect to the client's eID service.

The technical requirements mainly insist on a broad support of existing standards, from low-level smart card and secure element standards such as CEN TS15480 CardInfo-files (T13) and the ISO/IEC 24727-2 and 24727-3 standards (T31), over interface standards such as the Open Mobile API (T32) and security assertion formats such as SAML and OpenID (T12) to application-layer protocols such as ISO 20022 (T2).

Among the general requirements, privacy seems to be a recurring theme. Informed consent (G4) comes up as a requirement in several use cases, meaning that at every use of the eID, it must be clearly communicated to the user what information is being transmitted. Minimal information disclosure (G12) is another recurring requirement, as are anonymous and pseudonymous authentication (G10). These requirements are best met by supporting privacy-preserving authentication technologies such as Identity Mixer and U-Prove (T26, T27).

13. References

[HRSZ10] M. Hansen, M. Raguse, K. Storf, H. Zwingelberg: Delegation for Privacy Management from Womb to Tomb - A European Perspective, in M. Bezzi, P. Duquenoy, S. Fischer-Hübner, M. Hansen, G. Zhang (Eds.): Privacy and Identity Management for Life, Proceedings of the 5th IFIP WG 9.2,9.6/11.4, 11.6, 11.7/PrimeLife International Summer School 2009, IFIP AICT 320, p. 18-33, Springer, 2010

[ZwiHan12] H. Zwingelberg, M. Hansen: Privacy Protection Goals and Their Implications for eID Systems, in J. Camenisch, B. Crispo, S. Fischer-Hübner, R. Leenes, G. Russello (Eds.): Privacy and Identity Management for Life, Proceedings of the 7th IFIP WG 9.2,9.6/11.7, 11.4, 11.6 International Summer School 2011, Springer, 2012

[RFC 2119] S. Bradner: Key words for use in RFCs to Indicate Requirement Levels. RFC 2119, 1997.

[SAML 2.0] OASIS: Security Assertion Markup Language. Version 2.0, 2009.

[ISO/IEC 7816] ISO/IEC: Identification cards - Integrated circuit cards. International Standard, ISO/IEC 7816, Part 1 - 13, 15, 2004 - 2011.

[ISO/IEC 24727] ISO/IEC: Identification cards - Integrated circuit card programming interfaces - Part 3: Application interface. International Standard, ISO/IEC 24727, Part 1 - 6, 2007 - 2011.

[ISO/IEC 24727-3] ISO/IEC: Identification cards - Integrated circuit card programming interfaces - Part 3: Application interface. International Standard, ISO/IEC 24727-3, 2008.

[ISO 20022] ISO agreed XML-based single standardisation approach (methodology, process, repository) to be used by all financial standards initiatives including KYC and eBAM. Note that V2 is shortly to be confirmed

[OpenID 2.0] OpenID Foundation: OpenID Specification. Version 2.0, 2007.

[CEN TS 15480] CEN: Identification card systems - European Citizen Card. TS 15480, Part 1–4.