

ATTACHMENT II
FUNCTIONAL AND TECHNICAL SPECIFICATIONS
ATTACHED TO THE INVITATION TO TENDER

**Invitation to tender No. AIB-2009-HUB-P-002 concerning the development,
implementation and support of a message transfer and monitoring facility**

I. DOCUMENT

A Related Documents

AIB-2009-Hub-P-002_Attachment III.pdf:

AIB-PRO-SD03 - EECS Registration Databases contains information about data standards, transfer interface protocol, performance standards and testing of systems and transfer links.

AIB-PRO Fact Sheet 5, Types of energy sources and technologies

B Intellectual Property Rights and Copyright

This document is the copyright of, and all rights are reserved on behalf of, the Association of Issuing Bodies.

II. CONTENTS TABLE

I.	Document	2
A	Related Documents	2
B	Intellectual Property Rights and Copyright	2
II.	Contents Table.....	3
1	Introduction	5
1.1	Purpose	5
1.2	Background.....	5
1.3	Document Structure.....	6
2	Function of the Hub	6
2.1	Register	6
2.2	Messaging.....	7
2.3	Logging	13
2.4	Testing	13
2.5	Common Data Management.....	13
2.6	Security Certificate Management	15
2.7	Reporting.....	15
2.8	Management	17
2.9	User Interface	17
3	Detailed functional Requirement Specification.....	18
3.1	Register	18
3.2	Messaging.....	19
3.3	Logging	20
3.4	Testing	21
3.5	Common Data Management.....	22
3.6	Security Certificate Management	23
3.7	Reporting.....	24
3.8	Management	25
3.9	User Interface	25
3.10	Requirements not in the process	26
4	Technology requirements.....	28
4.1	Server.....	28
4.2	Database	28
4.3	Clients	28
4.4	Interfaces	28
4.5	Security	29
4.6	Load and performance requirements	29
4.7	Usability.....	30
4.8	Role and permission model	30
4.9	Error messages and protocols	30
4.10	Flexibility	30
5	Operational requirements	31
5.1	Documentation.....	31
6	Service requirements	31
6.1	Application support.....	31
6.2	Application maintenance	31
6.3	Service levels	31

1 INTRODUCTION

1.1 Purpose

This document details the Functional Requirements for a new Hub as a part of the RFI/RFP. The Hub supports message transfer, monitoring, test and report facilities to be implemented on behalf of the Association of Issuing Bodies. In addition to these core functions, the Hub service will also be used to provide common data management and operational reporting.

1.2 Background

The AIB currently defines a market infrastructure based on a set of electronic Registries, one in each Domain. The issue and redemption of certificates, and transfers of holdings within the Registry, are handled entirely within the Registry, and these actions are private to that Domain. Transfers of holdings from one Domain to another are handled by an import/export protocol defined by the AIB. This protocol specifies content and format for the data concerning the transfer, and it also specifies the transport protocol to be used between Registries.

Import/export data files are exchanged directly between Registries or by the current AIB-Hub using email (SMTP).

The use of the AIB-Hub is mandatory for members from 1-1-2010.

A number of types of Hub have been identified, ranging from simple message routing through to a more comprehensive monitoring facility.

- Messaging

This is a simple message forwarding system where all transfers go through the Hub with no further processing than is necessary to deal with the Registry to Hub interfaces. Such a messaging system could be designed to handle different transport protocols for each Registry.

- Transaction log (simple)

This is an extension to the message forwarding Hub where the data is extracted and logged. The log itself could be a simple audit trail, or the data could be used to maintain some form of mirror Registry.

- Transaction log (full)

This is an extension of the transaction log that requires Registries to report all issue and redeem activities. It might be extended to include reports of all transfer activities.

This functional specification assumes that the new Hub:

- Covers functionality of old Hub, messaging, testing and certificate authority
- Allows for state-of-the-art security and encryption
- Is prepared for additional functionality such as:
 - Transaction/certificate statistics

- Common Data Registry, i.e. common between domains
- Production Device Registry (linked with domain Registries)
- Content verification
- Testing capabilities for integration with member applications
- Configurable reporting

1.3 Document Structure

Section 2 provides a descriptive overview of the process that the Hub is to support. This is designed to show how the detailed requirements fit together in context. Each requirement is separately identified in section 3. Section 4 list the technical requirements, section 5 the operational requirements and section 6 the service requirements.

2 FUNCTION OF THE HUB

2.1 Register

Registry and user details need to be registered and maintained in the Hub.

2.1.1 Registry registration

A new Registry should be registered in the Hub by the Hub operator or Hub support. After registration, the Registry information is available for other Registries, can be updated by the Registry itself or the Hub operator, and can be deleted by the Hub operator.

Information to be registered:

- Name of the Registry
- Status: Test or live environment
- Identifier(s) that can be used in the header of the XML files to identify the sender of recipient.
- E-mail address for messages
- E-mail address for system notifications
- Certificate type(s) permitted in Registry
- Domain(s) permitted in Registry

Also useful but not required:

- E-mail address for operator notifications
- Contact information, address, e-mail, telephone

2.1.2 User name password, role.

Each Registry, the Hub operator and the Hub support has one or more users who will have access to the Hub information specific for this user and role. Those users have access to the Hub with a username and password.

The Hub operator is allowed to create, read, update and delete a user.

Each user is forced by the system to change the password on a regular basis. The Hub operator or support organisation should be able to reset the password.

2.2 Messaging

The primary function is to provide a facility for transferring messages from one Registry to another. This is initially intended to support the protocol for transfer of certificate data between Registries, although other protocols may be supported in the future.

The messaging facility itself is, in principle, only required to support file exchange. That is, Registry A sends a file to Registry B by sending the message to the Hub. The Hub checks the format, content and then forwards the message to Registry B. This supports the certificate exchange protocol where Registry A sends a description of the certificates to be transferred to Registry B, and Registry B sends an acknowledgment back to Registry A.

Messages are currently transferred using SMTP. The Hub could potentially support a range of state-of-the-art message transport protocols. However, it is required initially to support the current SMTP protocol, with an extension to HTTP(S) or other state-of-the-art protocols expected later.

Security of transfer has two aspects: preventing others viewing the messages in transit, and so-called non-repudiation, whereby the sender of the message cannot deny message content.

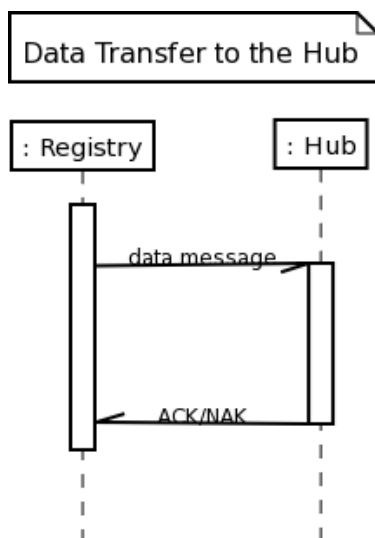
Non-repudiation is handled by using digital signatures. The sender digitally signs the message before transfer to the Hub. The Hub then re-signs the message before forwarding. Thus all messages received by any Registry are signed by the Hub.

In practice, the current AIB protocol standard requires messages to conform to the S/MIME standard and to be signed and encrypted. This standard is to be supported under SMTP, and it will continue to be supported under HTTP(S).

2.2.1 Message Flow – Sending data to the Hub

Although no such messages are defined yet, it is expected that it will become necessary for Registries to send data to the Hub, for example, for update of centrally held data.

The message flow protocol for transferring data to the Hub is:

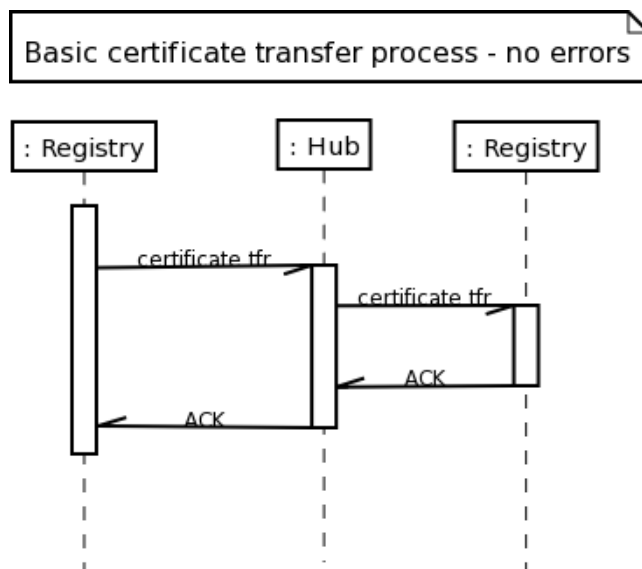


It is expected that the Hub will directly acknowledge a data message with an ACK if the message was read and understood, and with a NAK otherwise.

2.2.2 Message Flow – Certificate Transfer

2.2.2.1 **Basic Process**

The message flow protocol for a certificate transfer request from Registry A to Registry B is:



The sending Registry receives an acknowledgement from the receiving Registry. There is no intermediate acknowledgement from the Hub that the message has been received and is being forwarded.

The Hub retains a continuous thread of activity for the message. That is, the Hub recognises that a certificate transfer is in progress so that the returning ACK can be matched with the original message.

2.2.2.2 Errors detected by the Hub

The Hub may reject the message due to errors in the XML, unrecognised values in any of the routing fields, or unrecognised or otherwise invalid content in any other field.

Invalid content can be recognised by message validation, but also by business rules which are described in AIB-PRO-SD03: EECS Registration Databases.

New business rules need to be implemented if necessary. The present business rules are listed below. Validation may in future be done against the stored common data.

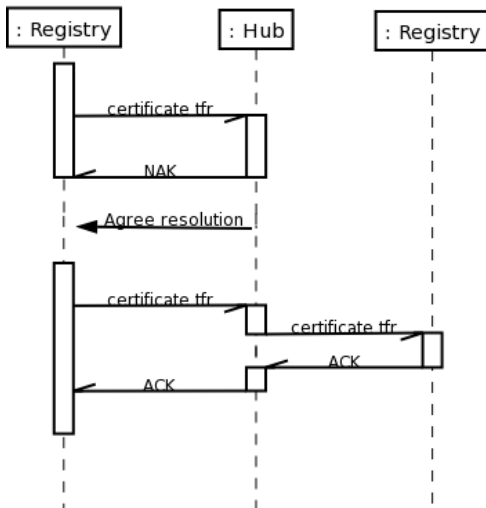
- checksum and/or common data on either seller or buyer ID or sender or receiver ID.
- checksum and/or common data on the Production Device.
- date rules, for example issue date can not be before production end date or start date can not be later than end date.
- technology code should be a code from the list (common data).
- earmark should be a code from the AIB list (common data)
- total (nroc).should be equal to all the certificate numbers in the range.
- length of ID's
- Issuing Body is only allowed to send certificates from the EECS scheme's they are member of. (common data)
- Use of heat is one of the codes from the list (common data)
- Certificate type not permitted in recipient Registry (common data)
- No response from recipient Registry within defined period¹

In such a case the Hub returns a NAK to the sender, with appropriate reason code where relevant to the message format².

¹ Response times will be included in SD03.

² SD03 may be modified to accommodate reason codes in NAK files.

Basic certificate transfer process - format or data error rejection by hub

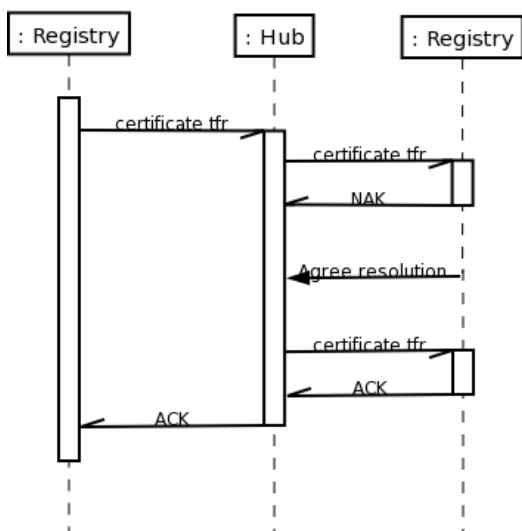


The rejection by the Hub is assumed to terminate any process that might exist in the sending Registry. The Registry and Hub operators handle resolution of the problem off-line. The Hub treats the subsequent re-transmission of the certificate transfer request as a new request.

2.2.2.3 Error in format detected by recipient

Registry B may reject the message due to errors in the XML. Since the Hub checks the XML from Registry A, an error here can only happen because the Hub has introduced it.

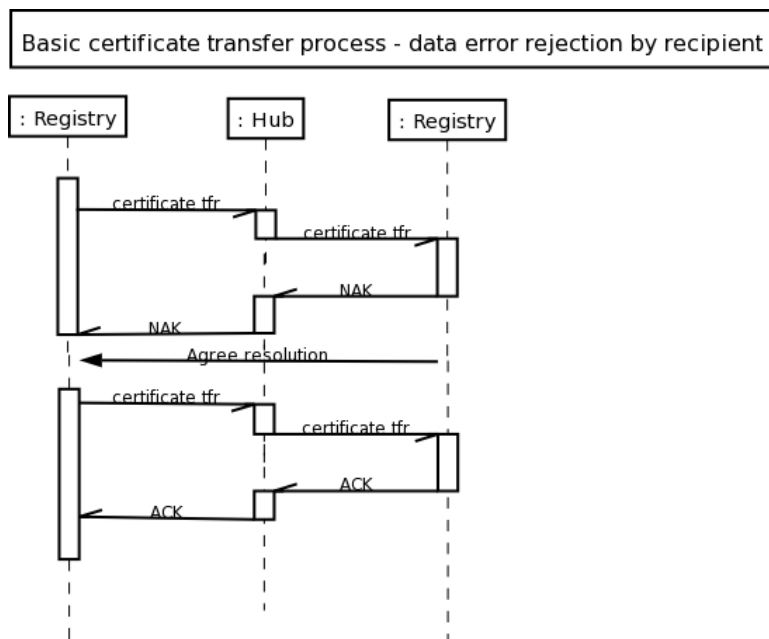
Basic certificate transfer process - format error rejection by recipient



The Hub retains a continuous thread of activity for the message. That is, the Hub recognises that a certificate transfer is in progress so that the returning ACK can be matched with the original message.

2.2.2.4 Error in content detected by recipient

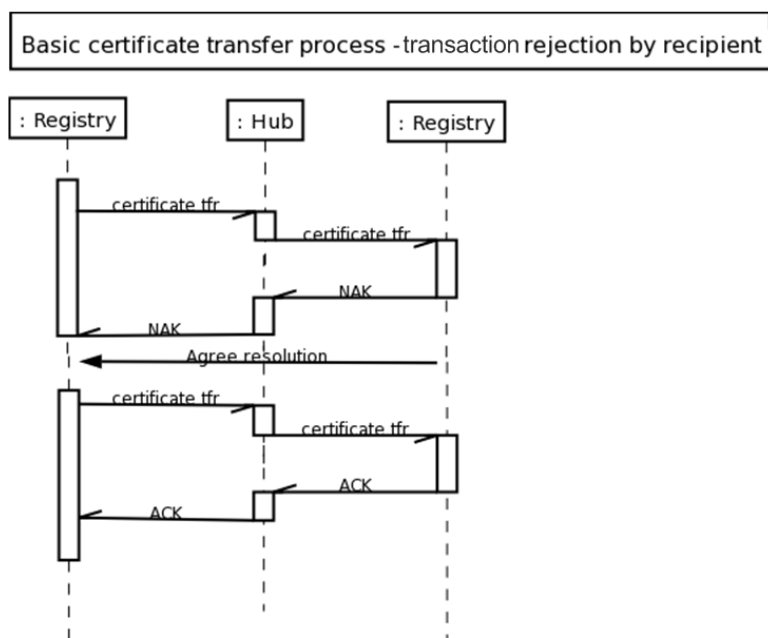
Registry B may also reject the message due to errors in content. The Hub does limited checking on content, so the error can be either an error in the data sent by Registry A, or incomplete data held by Registry B. In either case, the Hub is not involved in the correction process.



The rejection by the receiver is assumed to terminate any process that might exist in the sending Registry. The Registry operators handle resolution of the problem off-line. The Hub treats the subsequent re-transmission of the certificate transfer request as a new request.

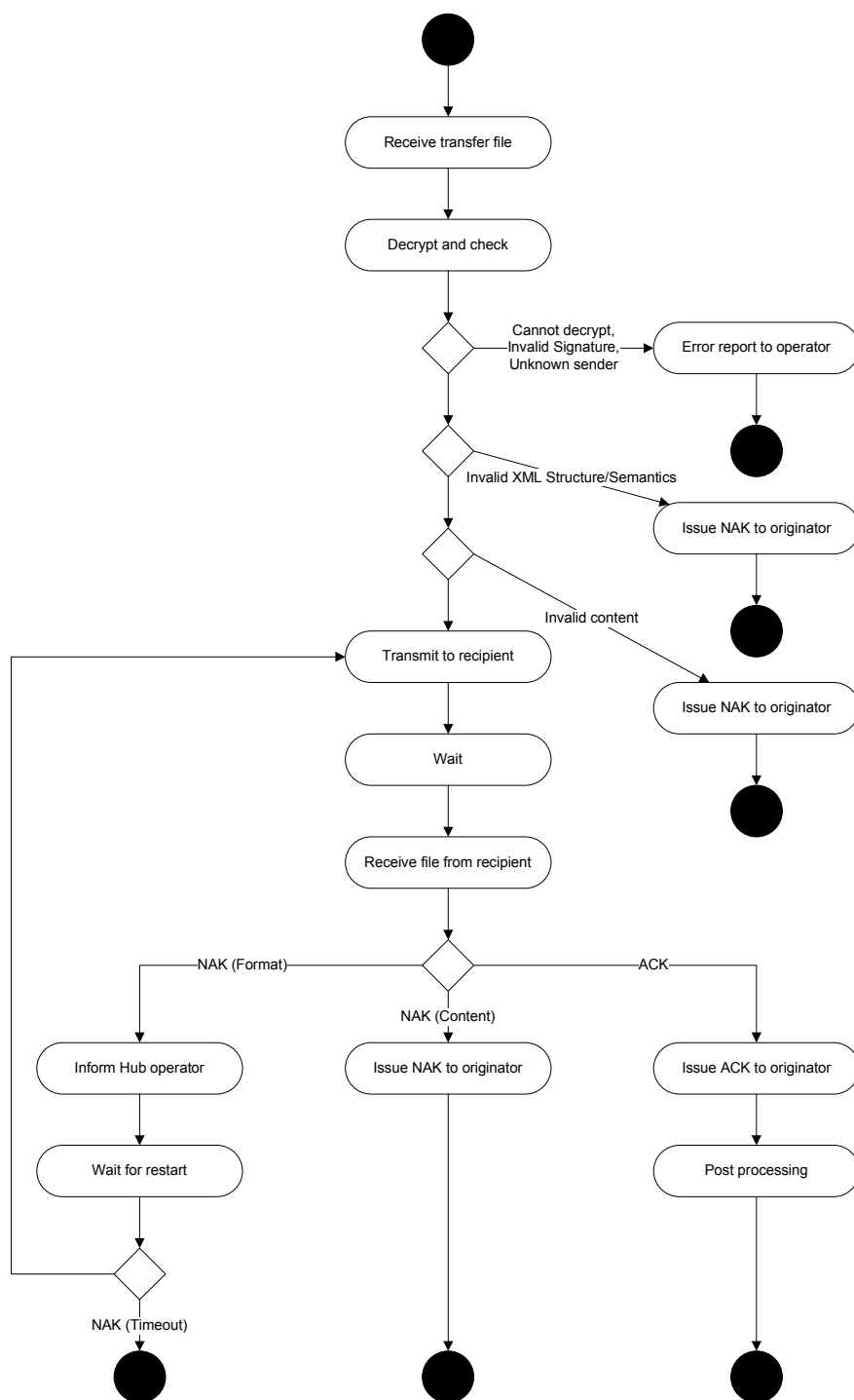
2.2.2.5 Transaction rejected by recipient

Registry B may also reject the message should the recipient reject the transaction. In this case, the Hub is not involved in any correction process that may take place.



The rejection by the receiver is assumed to terminate any process that might exist in the sending Registry. The Registry operators handle resolution of the issue off-line. The Hub treats the subsequent re-transmission of the certificate transfer request as a new request.

The overall action diagram for these processes is:



2.2.3 Routing

Messages between Registries are defined in the document AIB-PRO-SD03: EECS Registration Databases with a header that contains an <r:to> field and an <r:from> field. The Hub will expect these fields to refer to the sending and receiving Registries respectively and will use the field content to identify the actual address and the transport mechanism to be used for forwarding. Subject to any standards described in SD03, each Registry may choose to be represented by one or more of: CMO EAN number; email address; or, url.

2.2.4 File formats

File formats are based loosely on the SOAP standard. That is, they are formatted using XML and contain <s:header> and <s:body> elements.

The formats for certificate transfer files and the corresponding ACK/NAK files are given in the document AIB-PRO-SD03: EECS Registration Databases

It is expected that ACK/NAK files for other data transfer types will have the similar format.

The XSD scheme's which are supported should be available in the Hub, for scheme validation.

The Hub needs to be prepared for implementation of new XSD scheme's in the future.

2.3 Logging

The purpose of logging is to provide a basis for subsequent reporting. Therefore it is a requirement of the Hub that transfer requests and other messages are interpreted before forwarding and that the data is stored in a database ready for reporting.

The Hub must verify that the message content is reasonable. This should be done before forwarding so that the Hub can send an appropriate message back to the sender if an error is found.

2.4 Testing

The Hub must have a facility that allows Registry operators to test their ability to send and receive messages from the Hub. The facility will not test the internal workings of the Registries, but it should allow Registries to confirm that they can handle all message types, including rejection responses.

2.5 Common Data Management

The Hub needs the sender and receiver addresses of the messages and naturally the keys and certificates used for digital signatures and encrypting.

In a later phase the Hub requires certain data about account holders and production devices in order to validate messages before logging them. This data may be limited to the relevant identifier and the issuing domain in which the account or device is registered. This is not available in the present Hub, so new procedures and structures should be agreed, and implemented. Also procedures for add, change and removal of this data.

The Hub must make this common data available to Registries in the form of downloadable files or web pages. Common data information which is foreseen is listed in the paragraphs below. This list may be expanded, both before and after implementation.

2.5.1 Account details

It should be possible to store account details in the Hub.

Account number: This is to take any form that is allowed by the interface specification. Multiple account numbers may be needed for each entity.

Account name: Text for the name of the account holder as registered in the relevant domain.

Registrant: The identifier of the registrant for the account number.

2.5.2 Production Device details

It should be possible to store production device details in the Hub.

Production Device Identifier: This is to take any form that is allowed by the interface specification.

Production Device name: Text for the name of the production device as registered in the relevant domain.

Technology: Type of energy source and/or technology of the production device.

Registrant: The identifier of the registrant for the production device.

2.5.3 Technology codes

It should be possible to store technology codes in the Hub.

Technology codes are listed in PRO Fact Sheet 5, Types of energy sources and technologies.

Technology codes: Code of the source and technology used.

Technology description: Description of the code.

2.5.4 Earmarks

It should be possible to store earmarks in the Hub.

List of codes belonging to type of support of a production device.

Earmark description: Description of the different type of support.

Earmark code: Code linked to the earmark description.

This information is available in AIB-PRO-SD03: EECS Registration Databases

2.5.5 Use of Heat

It should be possible to store possible "Use of Heat" details in the Hub.

This information is available in AIB-PRO-SD03: EECS Registration Databases

2.5.6 EECS scheme information

It should be possible to store different EECS schemes in the Hub.

Issuing Bodies (IB) are member of an EECS scheme, the member and EECS scheme they are allowed to use need to be stored.

This information is found at the AIB website:

http://www.aib-net.org/portal/page/portal/AIB_HOME/AIB_MEM

2.6 Security Certificate Management

The security of data in transit depends on the use of public/private key pairs issued to Registries. Each Registry will use it's own private key to digitally sign messages before transmission to the Registry, and the Hub will use the corresponding public key to confirm the message integrity. Similarly, the Hub will use a private key to sign forwarded messages, and the recipient will use the Hub's public key to confirm the message integrity. If necessary, and depending on the transport mechanism used, the messages will be encrypted by the relevant public key.

The Hub must provide a facility for the generation of key pairs if required by any Registry.

2.7 Reporting

The Hub is, later, expected to be capable of providing reports on volumes of certificates transferred between Registries, issued and cancelled certificates and transfers in Registries broken down by Registry, date and technology. These reports must be available as displayed reports and as downloadable files for import into a spreadsheet. The exact data on the reports is not yet defined.

AIB want a user-configurable reporting tool about the content of the messages. This means that data of the messages need to be in a database. The Hub should not store data in the database if the recipient of a transfer request or similar message rejects that request.

Reporting should be possible for different roles and different aggregation level of the data. Selection criteria in the paragraphs below the roles are described and examples are given.

2.7.1 Public

Domain import and export statistics. Selectable by: certificate issue date range or end of generation period date range; single domain or all domains; single technology or all technologies.

Standard reports on an aggregated level.

2.7.2 Private to Registry operators

Domain import and export transactions for the operator's domain only. Selectable by: certificate issue date range or end of generation period date range; single technology or all technologies.

Account details: selectable by domain or all domains.

Production Device details: selectable by domain or all domains.

Hub interaction details for the operator's domain only. Selectable by: transfer date range; other party.

Test activities for the operator's domain only. Selectable by: date range

2.7.3 Private to Hub operator

Domain import and export transactions for the operator's domain only. Selectable by: certificate issue date range or end of generation period date range; domain or all domains; single technology or all technologies.

Account details. Selectable by: domain or all domains.

Production Device details. Selectable by: domain or all domains.

Hub interaction details. Selectable by: transfer date range; sending party; receiving party.

Test activities. Selectable by: date range; domain.

2.7.4 Interpretation of reporting selection criteria

The report classes listed in 2.7.1, 2.7.2 and 2.7.3 include specifications of selection criteria. This section describes how the list of selection criteria is to be interpreted, and shows how the data for the resulting report is to be grouped.

2.7.4.1 **Omitting a selection**

The user may omit all or any selection criteria. In this case the report selection simply excludes that selection, all values for the relevant data item are allowable and no record is rejected on the basis of the relevant data item.

The resulting report could

- summarise the data for all values of the omitted selection field and exclude that data field from the report, or,
- report on every distinct value of the data field, and include the data field in the report.

The selection mechanism must allow the user the option of choosing one of these two possible report types.

2.7.4.2 **Date ranges**

Lower dates are inclusive. Records match if the relevant field is greater than or equal to the given criterion.

Upper dates are exclusive. Records match if the relevant field is less than the given criterion.

Date ranges may be incomplete. That is, if the lower data range is given the report includes all records with relevant dates equal to that date and above, and if an upper date is given the report includes all records with relevant dates below that date.

2.8 Management

The Hub must be able to report on various aspects of its status, the status of messages, numbers of messages processed and errors in messages.

Errors in messages or in transmission, or any other operationally detectable error, must be flagged up immediately to defined people, including the Hub operator and the relevant Registry operator. It must be possible to manage errors, and it must be possible to resend messages. Making changes to the content of messages must be prohibited.

The Hub should operate automatically as far as possible.

2.9 User Interface

All aspects of management and use of the system must be by remote secure access. Access to data entry, market reporting, management reporting and basic management functions will use https. Access to the Hub for low level problem resolution, software upgrades and system administration will be through a secured remote access facility.

3 DETAILED FUNCTIONAL REQUIREMENT SPECIFICATION

The Requirements of the Functional Details are described below. For each of the requirements the following details are provided:

- Requirement Title
- Textual description
- Desirability of Requirement
 - o 1 – Functionality in the existing Hub – Required
 - o 2 – New wish for near future – Hub should be prepared for this
 - o 3 – Nice to have
- Implementation Considerations

3.1 Register

3.1.1 Registry registration

3.1.1.1 Configuration functions
The Hub must provide functions for: adding or removing Registries, adding or removing private keys, adding or removing public key certificates, adding or removing an intermediate or root certificate.
1 – Functionality in the existing Hub – Required

3.1.2 User name password, role.

3.1.2.1 User Identification
The Hub identifies each user by means of a username and password combination.
1 – Functionality in the existing Hub – Required

3.1.2.2 Password Management
The Hub requires each user to set their own password, and forces the user to change the password from time to time.
1 – Functionality in the existing Hub – Required
The timetable is not specified yet.

3.1.2.3 Secure access
Low level access for problem resolution, software upgrades and other activities requiring direct access to the Hub server must be secured so that unauthorised access is prevented.
1 – Functionality in the existing Hub – Required
This will generally require the use of SSH or similar secure log-in facilities. Other facilities that are not considered secure must be disabled.

3.2 Messaging

3.2.1.1 Multiple communication protocol

The Hub must be capable of sending and receiving files using SMTP or HTTP(s) or other state-of-the-art protocols.

1 – Functionality in the existing Hub – Required

3.2.1.2 Receive using any transport mechanism – a

The Hub must allow a remote party to send files using any of the defined transport mechanisms.

1 – Functionality in the existing Hub – Required

3.2.1.3 Receive using any transport mechanism – b

The Hub must not require notice from the remote party as to the transport mechanism that party is about to use for sending a file.

1 – Functionality in the existing Hub – Required

3.2.1.4 Send using any transport mechanism

The Hub allows a receiving Registry to specify in advance the transport mechanism that is to be used when sending messages to that Registry.

1 – Functionality in the existing Hub – Required

3.2.1.5 AIB Standards

The Hub supports the standards for messages defined in AIB-PRO-SD03: EECS Registration Databases

1 – Functionality in the existing Hub – Required

3.2.1.6 Communication protocol

The Hub must be capable of sending and receiving files using SMTP.

1 – Functionality in the existing Hub – Required

Hub should be open for other state-of-the-art protocols in the future

3.2.1.7 Security protocol

The Hub must be capable of sending and receiving and interpreting files using S/MIME.

1 – Functionality in the existing Hub – Required

Hub should be open for other state-of-the-art protocols in the future

3.2.1.8 Messaging Protocol

The Hub must provide the ability to perform the protocols outlined in section 2.2

1 – Functionality in the existing Hub – Required

Hub should be open for other state-of-the-art protocols in the future

3.2.2 Message Flow – Sending data to the Hub

3.2.2.1 Format validation

The Hub validates the XML structure of all incoming messages

1 – Functionality in the existing Hub – Required

3.2.3 Message Flow – Certificate Transfer

3.2.3.1 Content validation

The Hub validates the content of all incoming messages

1 – Functionality in the existing Hub – Required

This includes all data items that can be validated, if possible to common data. Specifically it will include validating check digits on account numbers and production device numbers and other issues mentioned in 2.2.2.2.

3.2.4 Routing

3.2.4.1 Receive message

The Hub checks if the sending party, is the same party as the sending party in the message

1 – Functionality in the existing Hub – Required

3.2.4.2 Deliver message

The Hub send the message only to the party who is the receiving party in the message

1 – Functionality in the existing Hub – Required

3.2.5 File formats

3.2.5.1 Check file format

The Hub checks if the file which is used, is one of the known file formats

1 – Functionality in the existing Hub – Required

3.2.5.2 Check file format for specific Registry

The Hub checks if the Registry sending a file, is allowed to use this specific file version

3 – Nice to have

3.3 Logging

3.3.1.1 Transaction Log simple

Record all transfers between Registries.

1 – Functionality in the existing Hub – Required

3.3.1.2 Transaction Log

Record details of all transfers between Registries.

2 – New wish for near future – Hub should be prepared for this
Most of the implementation considerations arising from this requirement relate to other requirements, such as market reporting.

3.3.1.3 Record full details
The Hub records all of the details provided for in AIB-PRO-SD03: EECS Registration Databases
3 – Nice to have

3.4 Testing

3.4.1.1 Test Facility
The Hub must provide for Registries to initiate one or more test sequences for the XSD scheme's requested by the Registry.
1 – Functionality in the existing Hub – Required

3.4.1.2 Protocol Coverage
The full set of test sequences exercise all of the protocol paths identified in 2.2
1 – Functionality in the existing Hub – Required

3.4.1.3 Process Coverage
Test sequences for receiving from the Hub to include valid files, files with errors in format, and files with errors in data.
1 – Functionality in the existing Hub – Required
Each XSD scheme, has a specific set of test files, and error checks like the checks described in 2.2.2.2

3.4.1.4 Testing Formal Requirements Support
The test facility must allow a Registry operator to build up a test portfolio that conforms to the testing standard defined in AIB-PRO-SD03: EECS Registration Databases
1 – Functionality in the existing Hub – Required
This standard is different for the different EECS scheme's or file formats

3.4.1.5 Piecemeal testing
The test facility must allow a Registry operator to test both single files and groups of files so that tests may be run at a level of granularity that suits the requirements of the Registry operator at that time.
1 – Functionality in the existing Hub – Required
Groups of files may be pre-defined. It is not necessary that the Registry operator should be able to define the content of a group. However, a range of groups should be provided.

3.4.1.6 Status confirmation
The result of each test, as perceived by the Hub, must be available to the Registry operator, including EECS scheme and XSD-version
1 – Functionality in the existing Hub – Required

--

3.4.1.7 Status content
The result report from each test must include adequate information for the Registry operator to understand the nature of any failure and to facilitate fault fixing.
1 – Functionality in the existing Hub – Required
This might be expected to include full reports from XML parsing or details of data items that are at fault.

3.4.1.8 Completion status
The Hub provides a facility for the Registry operator to indicate the result of the test as perceived by the Registry operator.
1 – Functionality in the existing Hub – Required

3.4.1.9 Test Log
All test activities must be logged.
1 – Functionality in the existing Hub – Required

3.4.1.10 File Coverage
Test sequences for sending to the Hub to cover all variations in data file type.
1 – Functionality in the existing Hub – Required
In the subsequent implementation this must include all file types defined in other parts of the specification.

3.5 Common Data Management

3.5.1 Common Data handling

3.5.1.1 Publish Common Data
The Hub must provide facilities for downloading all common data in a form that can be imported into databases.
2 – New wish for near future – Hub should be prepared for this

3.5.1.2 Manage Common Data
The Hub provides facilities for the Hub operator to modify or add to all common data.
2 – New wish for near future – Hub should be prepared for this

3.5.1.3 Update Common Data
The Hub provides facilities for Registry operators to modify or add to the data relevant to their own domain.
2 – New wish for near future – Hub should be prepared for this
In the interests of reducing changes required in existing Registries, these facilities are expected to be form based.

3.5.2 Account details

3.5.2.1 Store Account data

The Hub must store data relating to all accounts recognised across all Registries.

2 – New wish for near future – Hub should be prepared for this

3.5.3 Production Device details

3.5.3.1 Store Production Device data

The Hub must store data relating to all production devices recognised across all Registries.

2 – New wish for near future – Hub should be prepared for this

3.5.4 Technology codes

3.5.4.1 Store Technology codes

The Hub must store data relating to Technology codes

2 – New wish for near future – Hub should be prepared for this

Technology codes are listed in PRO Fact Sheet 5, Types of energy sources and technologies.

3.5.5 Earmarks

3.5.5.1 Store Earmarks

The Hub must store data relating to Earmarks

2 – New wish for near future – Hub should be prepared for this

Earmarks are listed in SD03

3.5.6 Use of Heat

3.5.6.1 Store "Use of Heat"

The Hub must store data relating to possible "Use of Heat"

2 – New wish for near future – Hub should be prepared for this

Uses of Heat are listed in SD03

3.5.7 EECS-scheme and XSD information

3.5.7.1 Store EECS-scheme in relation to XSD versions

It's possible that different AIB-scheme's are supported by different XSD files. This should be stored and checked.

2 – New wish for near future – Hub should be prepared for this

3.5.7.2 Store XSD

The Hub must store the actual XSD versions

2 – New wish for near future – Hub should be prepared for this

The XSD version should give access for scheme validation

3.6 Security Certificate Management

3.6.1.1 Configuration functions

The Hub must provide functions for: adding or removing private keys, adding or removing public key certificates, adding or removing an intermediate or root certificate.

1 – Functionality in the existing Hub – Required
--

3.6.1.2 Certificate Authority

The Hub should act as a certificate authority for the issuing of security certificates related to all interactions with the Hub.
--

1 – Functionality in the existing Hub – Required
--

The primary purpose of the certificate authority is to support the security used in message transfer. However, it is expected that the user interface may require a certificate to support HTTPS, and that users of the system will have other needs. The certificate authority should not be limited in the range of uses to which it might be put.
--

3.6.1.3 Issue public/private keys
--

The Hub provides tools for issuing public/private key pairs to Registries.
--

1 – Functionality in the existing Hub – Required
--

The Hub (or Hub operator) takes responsibility for ensuring that Registries have security keys
--

3.6.1.4 Provide access to public keys
--

The Hub should provide the public key for itself and for each Registry to all other Registries
--

1 – Functionality in the existing Hub – Required
--

3.6.1.5 Maintain public/private keys

The Hub provides tools to ensure that security keys remain current.

2 – New wish for near future – Hub should be prepared for this
--

This is intended to avoid delays in file transfer resulting from out of date security keys.

3.7 Reporting

3.7.1.1 Report classes

The Hub must be able to support the report classes outlined in 2.7.

2 – New wish for near future – Hub should be prepared for this
--

3.7.1.2 Public and Private reports

The Hub must be able to restrict access to reports according to the classification in 2.7

2 – New wish for near future – Hub should be prepared for this
--

3.7.1.3 Flexible selection criteria
--

The Hub interprets selection criteria according to 2.7.4
--

2 – New wish for near future – Hub should be prepared for this
--

3.7.1.4 Report available for display

The market reports must be available for display through the user interface.
--

2 – New wish for near future – Hub should be prepared for this

3.7.1.5 Report available for download

The market reports should be available for download in computer processable format.

2 – New wish for near future – Hub should be prepared for this

3.7.1.6 Extensibility

The chosen solution must be flexible to allow further development of management processes and reporting.

2 – New wish for near future – Hub should be prepared for this

3.8 Management

3.8.1.1 Data update on-line

The Hub supports standing data update by means of on-line data entry and modification forms. This applies to all configuration and management option data.

1 – Functionality in the existing Hub – Required

3.8.1.2 Activity logging

The Hub must provide reports of activity, selected by any combination of one or more of: sent/received, local party, remote party, date range.

1 – Functionality in the existing Hub – Required

3.8.1.3 Viewing sent and received files

The Hub must be able to present to the user the clear text versions of any files sent or received by the specific Registry.

2 – New wish for near future – Hub should be prepared for this

This only applies to those files which the system is able to decrypt.

3.8.1.4 Extensibility

The chosen solution must be flexible to allow further development of management processes and reporting.

2 – New wish for near future – Hub should be prepared for this

3.9 User Interface

3.9.1.1 Interface Design

The Hub must have a web front end.

1 – Functionality in the existing Hub – Required

By its nature, the location of the Hub is not critical, as it needs to be accessed for message transfer and reporting from any location. All access must therefore be through web interfaces.

3.9.1.2 Access control

The Hub supports individual access profiles set for each user ID, allowing access only to specific modules and menu options

1 – Functionality in the existing Hub – Required

3.9.1.3 Secure access

The user interface should be designed to minimise the possibility of unauthorised access.

1 – Functionality in the existing Hub – Required

Although user passwords have been specified elsewhere, the design should provide for mechanisms that reduce the possibility of, for example, discovering the user password. This may involve the use of HTTPS and other security techniques.

3.9.1.4 Help System

On–screen context–sensitive help to be available on request from any user interface screens.

2 – New wish for near future – Hub should be prepared for this

3.10 Requirements not in the process

3.10.1 Archiving

3.10.1.1 Backup

The Hub must include provision for backup of all data.

1 – Functionality in the existing Hub – Required

3.10.1.2 Basic Archive

The Hub must maintain a log of all message activity, together with a copy of all messages sent and received.

1 – Functionality in the existing Hub – Required

Archives need not be retained in a local on-line form and may be transferred to archive media such as CDR.

3.10.1.3 Archive management

Ability to select past–period records for transfer to on–line archive files, to keep current files within manageable size limits for enquiry, reporting and back–up purposes, yet allowing archive data to be included in screen enquiries and reports as and when required.

2 – New wish for near future – Hub should be prepared for this

Only designated users should have control over the archiving of transactions. Once transactions have been archived they should remain accessible to all users on request for reporting purposes.

3.10.2 Quantitative Requirements

3.10.2.1 Message volumes

The Hub must be capable of handling: (expected norm) currently 100 messages per day in average; (expected potential maximum) 25,000 messages per day.

1 – Functionality in the existing Hub – Required

This requirement represents the design expectations for the total numbers of messages in both directions. This must be taken into account when considering response times and storage volumes.

3.10.2.2 Message response time

The Hub must begin to process any message (incoming or outgoing) within: (expected) 30 seconds; (maximum) 3 minutes.

1 – Functionality in the existing Hub – Required

There is no difficulty in having the Hub respond faster than this. This requirement is intended to provide an outer envelope for the design.

3.10.2.3 Message sizes (average)

The Hub must be designed to handle messages where the average size is: 2 Kbytes

1 – Functionality in the existing Hub – Required

This requirement is aimed at defining sizes of archive and other accumulated data.

3.10.2.4 Message sizes (limits)

The Hub must be capable of handling messages up to: (expected) unlimited; (minimum) 500 Kbytes

1 – Functionality in the existing Hub – Required

This requirement addresses storage and processing limitations which might apply to an individual message.

3.10.2.5 Data retention

All data is to be kept for a minimum of 7 years.

1 – Functionality in the existing Hub – Required

A search index is needed

3.10.3 Language

3.10.3.1 User Interface Language

The Hub must support the English language

1 – Functionality in the existing Hub – Required

This includes screen text, help text, error messages, as well as report content (headings, descriptions, status words and any other derived text).

3.10.3.2 User Interface Multi-lingual display

If possible the Hub must support localisation into any European language.

3 – Nice to have

This includes screen text, help text, error messages, as well as report content (headings, descriptions, status words and any other derived text).

3.10.3.3 Language selection

If possible the Hub allows each user to select the language used for the user interface.

3 – Nice to have

4 TECHNOLOGY REQUIREMENTS

4.1 Server

- 4.1.1 The server operating system has to be industry standard with vendor support
- 4.1.2 The server solution has to be industry standard with vendor support
- 4.1.3 The backup solution has to be industry standard with vendor support

4.2 Database

- 4.2.1 The database solution has to be industry standard with vendor support
- 4.2.2 A data log and a data history is required
- 4.2.3 All database transactions have to be logged. For the transaction log at least the following information is required: Transaction, user, date, time
- 4.2.4 A complete documentation of the data model has to be provided
- 4.2.5 Multi-client capability is required
- 4.2.6 Options or clarification regarding database licenses have to be part of the tender

4.3 Clients

- 4.3.1 The client application has to be a web client based on industry standards
- 4.3.2 No specific browser and browser settings have to be required
- 4.3.3 The web user interface has to reflect the corporate design of the AIB (like <http://www.aib-net.org/>)
- 4.3.4 No proprietary technology must be used for the client application

4.4 Interfaces

- 4.4.1 The interfaces have to be implemented according to the business specification
- 4.4.2 Open standards for data import / and export have to be used:
 - 4.4.3 XML with XSD validation
 - 4.4.4 Excel (CSV)
 - 4.4.5 ASCII
- 4.4.6 The standard for certificates has to be XML with XSD validation (according to the AIB specifications)
- 4.4.7 State of the art and transaction safe interface technology.
- 4.4.8 Import and export of certificates has to be implemented to be using e-mail (SMTP)

- 4.4.9 All other interfaces to the Hub have to be implemented as web services (WSDL). This includes reports, upload of shard data etc.
- 4.4.10 Import and export of certificates has to be implemented to be using web services as an alternative to e-mail (SMTP) and should operate in parallel with e-mail (SMTP)
- 4.4.11 Backwards compatibility
- 4.4.12 For the import and export of certificates the technology of the existing hub (email) has to be supported in parallel. This means that certificates can be exported via web service from one domain while the receiving domain can import the certificates via email (and vice versa).

4.5 Security

- 4.5.1 Transactions have to be secure; encrypted; email with certificate
- 4.5.2 The email certificates can be generated via web client by the issuing bodies themselves
- 4.5.3 Login concept:
 - Username
 - State of the art password (the password has to be changed by the user every N month; the N can be configured by the system administrator)
 - Administrator, super user etc. roles can access the Hub from specific computers only (IP/MAC filter)

4.6 Load and performance requirements

- 4.6.1 At least 100 clients have to be handled by the Hub
- 4.6.2 The reaction time (client) has to be state of the art
- 4.6.3 The response time (server) has to be state of the art
- 4.6.4 At least 50 concurrent clients have to be handled by the Hub
- 4.6.5 At least 25000 requests per day have to be handled by the Hub

4.7 Usability

- 4.7.1 The application should be user friendly in general
- 4.7.2 Multi language support is required. The default language is English. The application is delivered in English only. Any additional language can be added easily. Any visible text element of the application can be translated. The translation to any other language is not part of the tender.
- 4.7.3 Error messages have to be user friendly (with error number and error text)
- 4.7.4 The application can be configured by the administrator via graphical user interface

4.8 Role and permission model

- 4.8.1 Flexible role and permission model (like <http://www.aib-net.org/>)
 - Administrator
 - Super User
 - General Secretary
 - Issuing Body
 - Reporting
- 4.8.2 Permissions on GUI, access and data level
- 4.8.3 Several roles per user should be possible. A user with several roles has just the view of the role he is currently logged in.

4.9 Error messages and protocols

The following messages, protocols and logs are required:

- 4.9.1 Error, status and information messages
- 4.9.2 Error reasons
- 4.9.3 Error protocol
- 4.9.4 Error log
- 4.9.5 Action log
- 4.9.6 Data log
- 4.9.7 Batch log

4.10 Flexibility

- 4.10.1 Flexible system architecture: Additional requirements can be implemented easily at reasonable costs
- 4.10.2 Flexibility regarding the number of clients, users and the amount of transactions and data

5 OPERATIONAL REQUIREMENTS

5.1 Documentation

The following electronic (online) English documentation has to be provided:

- 5.1.1 Technical documentation: Application, system, server architecture (including interfaces)
- 5.1.2 Specification
- 5.1.3 Functional documentation
- 5.1.4 User manual
- 5.1.5 Configuration including the defaults

6 SERVICE REQUIREMENTS

This includes application maintenance and 2nd and 3rd level support including all interfaces implemented by the solution provider.
The hosting of the application is not part of the tender.

6.1 Application support

Support organization:

- 6.1.1 1st level support: AIB expert user
- 6.1.2 2nd level support: Helpdesk operated by the solution provider
- 6.1.3 3rd level support: Specialists working for the solution provider

Support process:

- 6.1.4 AIB calls the helpdesk operated by the service provider
- 6.1.5 AIB defines the level of the problem (1 to 4)
- 6.1.6 The service provider analyzes the problem and fixes it according to the service level agreement (this includes a documentation of the problem and the solution)
- 6.1.7 Bug fixes for mayor problems have to be approved by AIB
- 6.1.8 The service provider informs AIB about the solution within the agreed time slot (according to the service level agreement)

6.2 Application maintenance

This includes continues improvements and fixes for the provided application, patches, workarounds and new releases. The service provider has to inform AIB about improvements for the provided application as soon as they are available and to install them. The service provider is responsible to maintain the knowhow for the provided application within the company.

6.3 Service levels

Service levels have to be defined for the following service elements:

- 6.3.1 Service slots (availability of the help desk)
- 6.3.2 Restore time (after forwarding a problem to the help desk); level 1 to 4 (critical, severe, small, trivial)
- 6.3.3 Maintenance slots