Electronic message-exchange according to the Contrast Model

Version 0.6, April 16, 2007

Icelandic Transport Layer Group

Markus Gudmundsson, Rue de Net, Editor (markus@ruedenet.is)

Orn Kaldalons, ICEPRO (icepro@icepro.is)

Bergthor Skulason, State Finance (bergthor.skulason@fjs.is)

Georg Birgisson, Eykur, (georg@eykur.is)

Tryggvi M. Thordarson, HB International (tryggvi@hbi.is)

0. Authors

This document is the effort of the following team:

Company	Who	Email
ICEPRO	Orn Kaldalóns	icepro@icepro.is
State Finance	Bergthor Skulason	bergthor.skulason@fjs.is
Unimaze Software (Rue de Net partner)	Markus Gudmundsson, Editor	markus@ruedenet.is
Eykur	Georg Birgisson	georg@eykur.is
HB International	Tryggvi M. Thordarson	tryggvi@hbi.is

1. Abstract

To ensure institutions, companies and government utilise quickly the business logic in the NESUBL work it is necessary to have a clear set of rules for the transport of messages.

Today's technology affords and offers many different ways in which to enable the transport, but, at the same time the multitude of possible solutions makes the choice that harder to make.

To start off with, the choices have to be few and clear, maybe to the point of taking extreme viewpoints and ignoring the middle.

This document explores and proposes two extremes, basic and advanced, black and white, which the authors think might make the use and implementation of NESUBL be quicker and more seamless.

2. The need for contrasting solutions

Albeit requirements differ vastly when exchanging electronic documents between parties, this document assumes all instances fall under one of two categories. These two catagories are:

- An extremely simple messaging model with basic functionality, derived from well-known standards. This simple solution enables the exchange of electronic documents in casual and simple transfers between individuals and small companies and even from these parties to larger businesses and governmental institutions.

- A fully defined messaging system based on international standards.

The challenge is not that simple solutions do not exist, but that there is no mutual agreement on which solution to adapt. In the remainder of this document a proposal for such a behaviour is suggested under the name Contrast Model for electronic message exchange.

3. The Contrast Model for electronic message exchange

Many companies of all sized require a full-blown message handler for delivering and receiving messages that supports:

- Reliable Messaging
- Non-repudiation
- User authentication
- Message Integrity
- Acknowledgement of receipt
- Duplicate elimination

There are different needs where smaller companies, even individuals, not capable nor interested in a complete messaging solution, but, which need a solution that is:

- Simple, yet effective
- Serves companies with the most basic need
- Serves casual exchange of electronic documents

The idea is not to serve the need of large-scale interactions between companies and governmental institutions where business processes are involved, although they could support this low-end solution as a service to smaller companies and individuals.

The requirements may also differ between documents and NESUBL profiles within the same corporation.

For example the messaging requirements for an account statement may be lower than for invoices and requirments for simple profiles that use standalone documents (e.g. invoice or order) may be lower than for complex and fully integrated profiles such as advanced procurement where each step in the workflow is dependant on the previous one.

4. Contrast Model – Advanced Implementations

This suggests the use of a full-blown message handling mechanism. ICEPRO, in August 2005, recommended the use of ebXML Messaging Services specification (ebMS). There

are other message handling standards such as WS-I (Web Services Interoperability) and related standards which need to be investigated further.

In either case, framework guidelines must fully specify the use of these standards. ebMS has options and a few open ends, while WS-I standards are from different standard groups and are overlapping.

No further description will be given here other than the following examples: Guidelines from STAR (the automotive industry, are very complete, both ebMS and web services), ebMS use from Sweden and use of the web services stack from Denmark.

5. Contrast Model – Basic Implementations

The proposed solution here is to use SMTP (Simple Mail Transport Protocol) with MIME attachments to exchange electronic documents. SMTP and MIME have been used successfully for decades for the exchange of electronic mail and is, along with HTTP, the most used internet protocol we know today.

The advantage of using SMTP over other popular protocols is that:

- People are both sending and receiving documents using this protocol today.
- There is already established mail client software is on the market today and has widespread use.
- People can easily understand the concept and procedures of "sending and receiving electronic mail" without the need to learn using new software or technologies.

Technologically SMTP has the advantage over HTTP (and FTP as well), for instance that:

- Companies, no matter how small, already have established mail servers, either within their company or with their internet operator.
- There is no need to set up and maintain a web server.
- Solution is feasible for companies that have very strict firewall policies.
- The exchange of documents can be made manually, whereas the sender and receiver can opt to use a standard mail client (Outlook, Lotus, Thunderbird, Eudora, etc.)

Besides, as with other established technologies, processing and monitoring of the exchange of electronic documents can easily be implemented by automating the tasks of sending (SMTP) and receiving (POP or IMAP from mail servers) electronic mail.

6. Contrast Model – Basic Implementation Protocol

To make automation efficient, the following guidelines have been established.

6.1. Sending a message

All parameters should use only numbers and characters from the English Alphabet. Automatic receivers should ignore differences between lower and upper case.

- Send protocol used: SMTP (RFC 821, Simple Mail Transfer Protocol)
- Envelope format: Multipart MIME (RFC 2045-2049, Multipurpose Internet Mail Extensions)
- Reply addressing (reply-to tag): SMTP header must be set to the sender's e-mail address for electronic document exchange.

6.1.1. Subject

The sender of a message should include in the subject line a unique message identifier.

For automated message handlers:

- **[GUID]**
  where a [GUID] is computer generated global unique identifier, expressed in a hexadecimal form: xxx-eee-ggg ….
  This form is used by automated message handlers

or alternatively for manual processing of messages only:

- **{COUNTRYID}[PARTYID]-[DATE]-[MSGID]**
  where [COUNTRYID] is an optional two letter country ISO code only used in international context, [PARTYID] is an identifier of the sender party (unique within country or GLN), [DATE] is a date representation, preferably in the format YYYYMMDD and [MSGID] is a unique identifier within that data. The sender party should format these consistently (to avoid clashing within its own namespace).

6.1.2. Body

The body of a message is a free text and can be used to send regular mail messages that go with the electronic document.

However, to better support automated message handling a few parameters should be sent by automated message handlers. This text must be the first text lines of the mail and include the parameter tags (casing does not matter):

6.1.2.1. Service

NESUBL profile identifier or similar identifier. For example this might be:

```
urn:www.nesubl.eu:profiles:profile4.0
```

6.1.2.2. Action

NESUBL document description or profile action description. For example this might be for Order or CreditNote. Or it might be:

```
urn:www.nesubl.eu:actions:CreateOrder
```

6.1.2.3. Conversation Identifier

The conversation identifier is used to start this process and or NESUBL profile, and should be created according to the following rule:

```
<dateOfInitialMessageTransmission>':'<uniqueNumber>':'<fromPartyId
>
```

With an example being:

```
ConversationId = 20070228:123:is0123456789
```

6.2. Attachments

The first document attachment is always the main document; the document that will be processed and at least one document is mandatory. The first document could refer to other documents or there could be standalone document that are related to the first one but are not necessarily referenced. These could be worksheets, images, etc.

Generally, the first attachment has MIME type text/xml. The suggested name of the attachment is to use the name of the document according to its specification (e.g. CreditNote for UBL 2.0 credit notes) and use conventional file extensions. Also if the document appears more than one in the same conversation, a sequence number can be used after a hyphenation (e.g. the second credit note correction made to the same invoice CreditNote-2.xml.

6.3. Sending acknowledgement

Sending an acknowledgement can happen either manually or automatically. When sending an acknowledgment, the Subject line contains the original message identifier,

prefixed with a "re: " (or the equivalent token for other languages) or "ack: " (for auto-generated acknowledgements). The original attachments are not included in the reply.

When performed manually, the user can simply reply to the electronic message just received.

When performed automatically, the message identifier in the subject line should be prefixed with "ack: ". The body tokens (service, action and conversation) can be repeated from the original message, but this is optional.

The acknowledgement should be sent as soon as the message is received and stored.

6.4. Transport level errors

Automatic message handlers should be able to handle administration mail messages that indicate errors occurring, e.g. if a host was not found, electronic mail address did not exist and tie them to sent messages.

In a same way, if a received message does not fulfill the requirements of the specification (duplicate message, no attachment found, invalid or unknown party) an error should be sent instead of an acknowledgment.

An error is sent by prefixing the message identifier in the subject header with or "error:". The body of the message should not contain anything but an error message.

6.5. Application level errors

During manual and automatic processing of documents after reception some errors may come up. These have to be handled with application responses according to the profile descriptions or mutual agreement. These are handled in the same way as sending a message and may require acknowledgments as well.

6.6. Checking availability of service

It is possible to send a message with to check the availability and integrity of a message handler. A regular message is sent with:

- Service parameters: urn:is:icepro:mshservice
- Action parameter: ping

When the acknowledgment is received, the message handler is considered up and running.

Appendix A. Sample MIME message

```
    MIME-Version: 1.0
    From: Company A, Document-X <document-exchange@company-a.com>
    To: Company B, Electronic Messages <emsgs@company-b.com>
    Reply-to: Company A, Document-X <document-exchange@company-a.com>
    Date: Fri, 07 Feb 2007 16:15:05 -0000 (GMT)
    Subject: 05817520-6B71-4609-BD2B-DE97F61A4712
    Content-Type: multipart/mixed;
                  boundary=unique-boundary-1

    SERVICE: urn:www.nesubl.eu:profiles:profile4.0
    ACTION: Order
    CONVERSATION: 20040510:4567:IS1234567890

    This text will be ignored by automatic handlers,
    but can be used as a message to manual receivers.

    --unique-boundary-1
    Content-type: text/xml; filename=Order.xml

    <?xml version="1.0" encoding="utf-8" ?>
    <Order xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="urn:oasis:names:specification:ubl:schema:xsd:Order-2">
    ...

    --unique-boundary-1
    Content-type: text/plain; charset=US-ASCII;
file=HandlingInstructions.txt

    Items should be packed as follows:
       Item 10023 with item 1023.
       Items 1014 and 1018 with item 1120.

    --unique-boundary-1
    Content-Type: image/jpeg
    Content-Transfer-Encoding: base64

      ... base64-encoded image data goes here ...
```

Appendix B. Revision history

This document has undergone the following revisions:

Version	Date	Version

Version 0.3	March 27, 2007	First published draft
Version 0.4	March 28, 2007	Missing author added
Version 0.5	March 29, 2007	Renaming of concepts and heading structure fixed.