Telecom SOA Requirements Version 1.0

Committee Specification 01

16 June 2010

Specification URIs:
This Version:
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cs01/t-soa-req-01-cs-01.html
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cs01/t-soa-req-01-cs-01.pdf (Authoritative)
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cs01/t-soa-req-01-cs-01.doc

Previous Version:
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cd02/t-soa-req-01-cd-02.html
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cd02/t-soa-req-01-cd-02.pdf (Authoritative)
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/cd02/t-soa-req-01-cd-02.doc

Latest Version:
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/t-soa-req-01.html
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/t-soa-req-01.pdf (Authoritative)
http://docs.oasis-open.org/soa-tel/t-soa-req1.0/t-soa-req-01.doc

Technical Committee:
OASIS SOA for Telecom (SOA-Tel) TC

Chair(s):
Enrico Ronco, enrico.ronco@telecomitalia.it

Editor(s):
Enrico Ronco, enrico.ronco@telecomitalia.it

Related work:

This specification replaces or supersedes:

- N/A

This specification is related to:

- OASIS Telecom Use Cases and Issues Version 1.0

Declared XML Namespace(s):

- N/A

Abstract:

This document is the second deliverable produced within the OASIS SOA-TEL TC and has the objective of collecting requirements related to technical issues and gaps of SOA standards (specified by OASIS and other SDOs) utilized within the context of Telecoms. Such technical issues are documented in SOA-TEL’s TC first deliverable “Telecom Use Cases and Issues, v.1.0”.

For each of the issues within the “Telecom Use Cases and Issues, v.1.0”, specific requirements are provided within this document. Where possible, non prescriptive solution proposals to the identified issues and requirements are also described, in order to possibly assist those Technical Committees (within OASIS and other SDOs) responsible for the development and maintenance of the SOA related standards.

Status:

This document was last revised or approved by the OASIS SOA for Telecom (SOA-Tel) TC on the above date. The level of approval is also listed above. Check the “Latest Version” or “Latest Approved Version” location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee’s email list. Others should send comments to the Technical Committee by using the “Send A Comment” button on the Technical Committee’s web page at http://www.oasis-open.org/committees/soa-tel/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (http://www.oasis-open.org/committees/soa-tel/ipr.php).

The non-normative errata page for this specification is located at http://www.oasis-open.org/committees/soa-tel/.

Notices

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", “SOA-TEL” are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/who/trademark.php for above guidance.

Table of Contents

Table of Figures

1 Introduction

Part of the work being undertaken by the OASIS SOA-TEL TC is to understand how SOA-related specifications and standards are used within the scope of the telecommunications environment and determine if there are any issues when used in this manner.

This is the second deliverable of the SOA-TEL TC, and its objective is to collect requirements to address technical issues and gaps of SOA standards (specified by OASIS and other SDOs) utilized within the context of Telecoms. Such issues are documented in SOA-TEL's TC first deliverable "Telecom Use Cases and Issues, v.1.0".

For each of the issues within such document, specific requirements are provided. Where possible, non prescriptive solution proposals to the identified issues and requirements are also described, in order to possibly assist those Technical Committees (within OASIS and other SDOs) responsible for the development and maintenance of the SOA related standards.

For each of the issues identified within "Telecom Use Cases and Issues, v.1.0", a section composed of

- "References",
- "Requirement",
- "Description",
- and "Proposed solution"

is included in this Requirements document.

In order to facilitate future activities, each requirement is identified by means of a reference, with the syntax [SOA-TEL Req. x.y].

The document is organized in the following sections:

- Section 2, Issues on "Intermediaries";
- Section 3, Issues on "Security";
- Section 4, Issues on "Management";
- Section 5, Issues on "SOA collective standards usage".

Moreover, Appendix B, SOA-TEL Requirements, groups all exposed requirements within one single view.

The next steps related to this activity will be taken within the OASIS Telecom Member Section. Most likely, issues and related requirements will be grouped according to categories, and sent and presented to the TCs or Working Groups considered as "owners" of the affected specifications, in order to verify if such groups will want to analyze them and provide their solution. Other alternatives may also be evaluated on a case by case approach. Nevertheless the solution of identified issues and the addressing of the requirements hereafter listed is not to be considered as part of SOA-TEL's TC Charter.

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in **[RFC2119]**.

1.2 Normative References

[RFC2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.

[WSDL 1.1] W3C Note (15 March 2001): "Web Services Description Language (WSDL) 1.1". http://www.w3.org/TR/2001/NOTE-wsdl-20010315.

[SOAP 1.2] W3C SOAP v.1.2, available at http://www.w3.org/TR/soap12-part1/

[SOA-TEL 1.0] OASIS Committee Specification 01, “Telecom SOA Use Cases and Issues Version 1.0”, March 2010. http://docs.oasis-open.org/soa-tel/t-soa-uci/v1.0/cs01/t-soa-uc-cs-01.html

[WS-N 1.3] OASIS Standard, “Web Services Base Notification 1.3 (WS-BaseNotification) Version 1.3”, October 2006. http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.htm.

[WS-A 1.0] W3C Web Services Addressing 1.0 – Core W3C Recommendation 9 May 2006, http://www.w3.org/TR/2006/REC-ws-addr-core-20060509/.

[WS-S 1.1] OASIS Standard, “Web Services Security Specification Version 1.1”, February 2006. http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

[WSDM-MOWS] OASIS Standard, “Web Services Distributed Management: Management of Web Services (WSDM-MOWS) Version 1.1”, August 2006. http://docs.oasis-open.org/wsdm/wsdm-mows-1.1-spec-os-01.htm

[SOA RM 1.0] OASIS Standard, “OASIS Reference Model for Service Oriented Architecture 1.0”, October 2006. http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf

[SCA Assembly 1.1] OASIS Committee Draft, “Service Component Architecture Assembly Model Specification Version 1.1”, January 2010. http://docs.oasis-open.org/opencsa/sca-assembly/sca-assembly-1.1-spec.pdf

[SOA RA 1.0] OASIS Committee Draft 01 Public Review 01, “Reference Architecture for Service Oriented Architecture Version 1.0”, April 2008. http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf

[WSDL 2.0] W3C Web Services Description Language (WSDL) Version 2.0 Part 0: Primer, http://www.w3.org/TR/2007/REC-wsdl20-primer-20070626/, June 2007

[SAML 2.0] OASIS Standard, “Security Assertion Markup Language (SAML) Version 2.0”, March 2005. http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

1.3 Non Normative References

N/A

2 Requirements on Intermediaries

Some existing specifications used by Service Oriented Architectures do not allow for the presence of intermediaries in message exchanges. The lack of standards for intermediaries has led to workarounds and proprietary solutions. This section develops the requirements for intermediaries in message exchanges.

OASIS SOA-TEL TC considers that addressing the specific requirements provided in this section may be the first step for a more general revision of the SOA specifications in order to extend their coverage to include the management of intermediaries.

2.1 Requirements on Transaction Endpoints Specification

2.1.1 Identification of Use Case

There is no standard way to specify in a message that is subject to a process or transaction, the end point to which the message should be sent at the end of the process or transaction.

The lack of endpoint specification in messages is more fully documented in [SOA-TEL 1.0], 3.1 Transaction Endpoints Specification.

2.1.2 Requirement(s)

Req. 1

The WS Addressing specifications, [WS-A 1.0], must include additional fields (in addition to the ones already present) containing remote destinations to which reply messages must be sent.

- The sender of a message must assign the fields when it wants to specify the destination for the reply message, but the node that has to use such destination information (i.e. the node that has to send the reply message) may not necessarily be the direct receiver of the request message.
- The receiver of a message, which needs of information on the endpoint destination to which send a reply message, can obtain the information by these additional fields.
- The receiver of a message has to forward to the next receiver all the additional destinations (present in these additional fields) that it does not use.

2.1.3 Description

The [WS-A 1.0] must include additional information to indicate nodes to which messages replies should be sent (in addition to the one already present).

Specific endpoints should be inserted when the message is part of a transaction involving more participants. Such endpoints must be forwarded, through the chain of invocations, to those nodes that will need to use these endpoints.

The generic node that starts a transaction should be able to specify endpoints for the nodes following in the transaction, in addition to the (already available) “reply_to” endpoint for the message’s direct receiver.

In complex scenarios involving more than 3 nodes, the generic node N that receives a message may not be conscious of the specific transaction of which it is part of, or of other participant nodes, but could obtain the endpoint to which it must send a reply message by fetching such new proposed endpoint element.

Moreover, the current “reply to” element within the WS-A specification could not be utilized for this objective because even the direct sender to node N may not be aware of the final destination for the message.

2.1.4 Solution proposals

The following text is provided in order to illustrate some possible ways to address the Requirement. They are suggestions and are by no means to be considered as mandatory, as other possible options could be identified which are not represented hereafter.

To the best knowledge within OASIS SOA-TEL TC, the requirements presented hereafter could be addressed by the W3C Web Services Addressing (WS-A) WG, which by the way is in status “Completed”.

The WS-Addressing v1.0 specification [WS-A 1.0] defines the following elements:

```
wsa:To>xs:anyURI</wsa:To> ?
<wsa:From>wsa:EndpointReferenceType</wsa:From> ?
<wsa:ReplyTo>wsa:EndpointReferenceType</wsa:ReplyTo> ?
<wsa:FaultTo>wsa:EndpointReferenceType</wsa:FaultTo> ?
<wsa:Action>xs:anyURI</wsa:Action>
<wsa:MessageID>xs:anyURI</wsa:MessageID> ?
<wsa:RelatesTo RelationshipType="xs:anyURI"?>xs:anyURI</wsa:RelatesTo> *
<wsa:ReferenceParameters>xs:any*</wsa:ReferenceParameters> ?
```

Another element could be added to contain a “remote” endpoint reference, named for example

```
<wsa:RemoteReplyTo> wsa:EndpointReferenceType</wsa:RemoteReplyTo> *.
```

It should be possible to add more RemoteReplyTo elements, in a LIFO (Last In First Out) criteria.

The generic receiver can use the last inserted endpoint and delete the element.

The following example is provided.

Suppose that *node_1* calls *node_2*.

node_1 states that the endpoint for the response is *node_n*, but it doesn’t know which node will be sending the final response to *node_n* at the end of the transaction, so it inserts the information (*node_n* endpoint) in the RemoteReply element, not in ReplyTo one. Figure 1 illustrates the example.

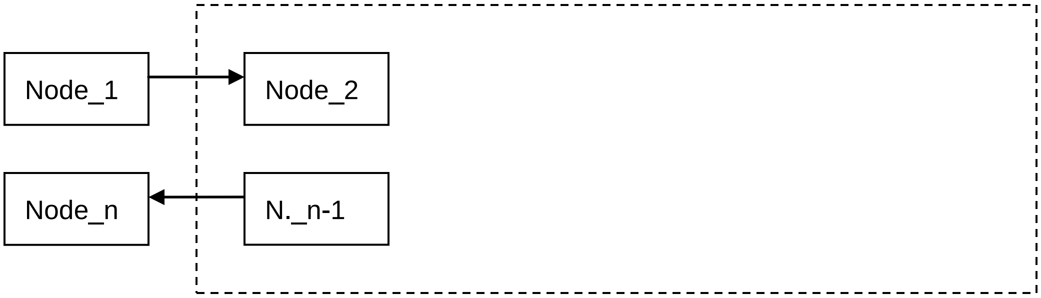


Figure 1: Example for SOAP nodes interaction (1)

The following is an example of the resulting message (in red color the proposed addition to the WS-A specification).

```
<soap:Envelope...>
  <soap:Header>
    <wsa:To> http://host_a/node_2 </wsa:To>
    <wsa:RemoteReplyTo>
      <wsa:Address>
              http://host_b/node_n
      </wsa:Address>
    </wsa:RemoteReplyTo>
    ...
  </soap:Header>
  <soap:Body>
    ...
  </soap:Body>
</soap:Envelope>
```

Suppose now that *node_i* in the transaction, calling *node_i+1*, starts a nested transaction (with *node_j* as final destination) in the main transaction. Also in this case, *node_i* does not know which will produce the response for the *node_j*, so it adds a RemoteReply element, to the message. Figure 2 illustrates the example.

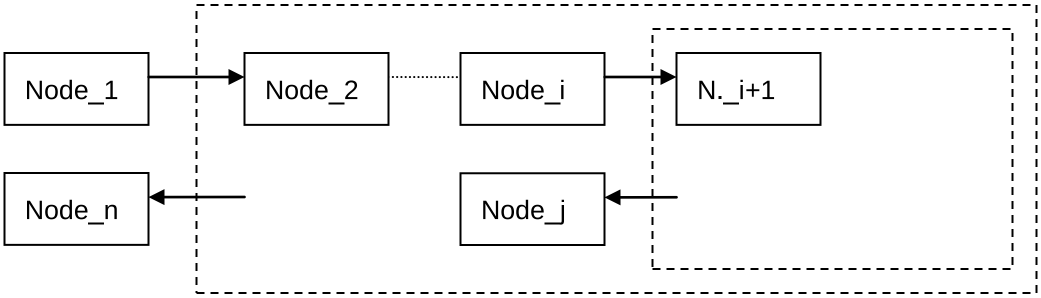


Figure 2: Example for SOAP nodes interaction (2)

The resulting message should be the following.

```
<soap:Envelope...>
  <soap:Header>
    <wsa:To> http://host_c/node_i+1 </wsa:To>
    <wsa:RemoteReplyTo>
      <wsa:Address>
              http://host_d/node_j
      </wsa:Address>
              </wsa: RemoteReplyTo>
    <wsa:RemoteReplyTo>
      <wsa:Address>
              http://host_b/node_n
      </wsa:Address>
              </wsa:RemoteReplyTo>
    ...
  </soap:Header>
  <soap:Body>
    ...
  </soap:Body>
</soap:Envelope>
```

Suppose now that *node_j-1* ends the nested transaction.

node_j-1 needs a reply destination, so it fetches the endpoint by the first RemoteReplyTo element, obtaining the information “http:// host_d/node_j”; it then deletes the element in the header and replies to *node_j*.

node_n-1, last node of the main transaction, should perform in the same way with the remaining RemoteReplyTo element. Figure 3 illustrates the example.

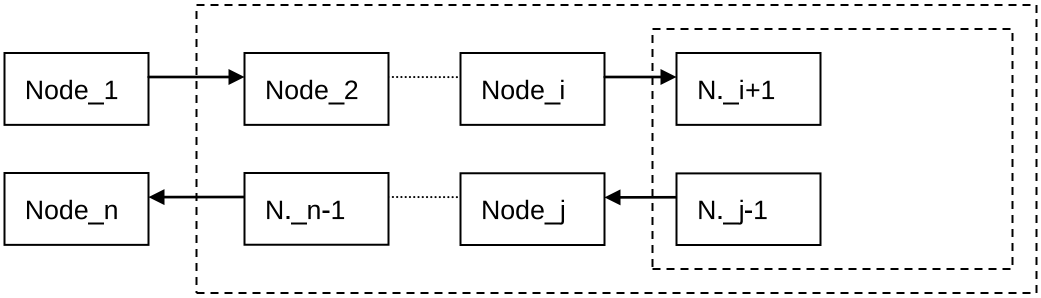


Figure 3: Example for SOAP nodes interaction (3)

2.2 Requirements on WS-Notification

2.2.1 Identification of Use Case

If adopting the WS-Notification [WS-N 1-3] specification, in presence of intermediaries, there is no formal way for the Provider to specify the endpoint to which the final notification should be sent.

Refer to [SOA-TEL 1.0], 3.2 of the SOA-TEL “Telecom Use Cases and Issues” document, in which the technical issue is documented.

2.2.2 Requirement(s)

Req. 2

The WS-Notification specification must provide a mechanism to describe and regulate a scenario in which one or more intermediaries are present; it must standardize the terminology, concepts, operations, WSDL and XML needed to express the roles of the intermediaries (involved in publish and subscribe Web services for notification message exchange).

According to the WS-Notification terminology, the standard must be extended and modified so that:

- a *Subscriber* can require a *Subscription* to a *NotificationProducer* also in the case they do not communicate directly but do so by means of one or more intermediaries;
- likewise a *NotificationProducer* can send a *Notification* to a *NotificationConsumer* also in the case that they do not communicate directly, but by means of one or more intermediaries.

2.2.3 Description

The WS-Notification specification must provide a well specified mechanism whereby a Subscriber can interact (by means of “subscribe”, “unsubscribe” and the other provided operations) with a NotificationProducer also in presence of one or more intermediaries between itself and the NotificationProducer.

Moreover the WS-Notification specification must provide a well specified mechanism by which a NotificationProducer can send notifications to a given NotificationConsumer also via one or more intermediaries.

In the new context, the Subscriber must be able to send a subscription message (different from the ones allowed by the current specification) to an intermediary; the intermediary must be able to request the subscription to the NotificationProducer or to send the request to the next intermediary. As a consequence an intermediary can receive a subscription request from another intermediary.

Moreover the new subscription response message must be managed and forwarded by intermediaries in a similar way.

Conversely, the NotificationProducer must able to send a notification addressed to a NotificationConsumer to an intermediary, and this intermediary must be able to forward the notification to the NotificationConsumer or to the next intermediary. In consequence of that an intermediary can receive a notification from another intermediary.

This requirement is closely connected to the requirement over WS-Addressing, described in Section 2.1 of this document (Requirements on Transaction Endpoints Specification) for two reasons:

- the two requirements introduce and regulate ”intermediaries management” in the WS-Addressing and WS-Notification specifications
- WS-Notification specification characterizes and identifies the actors (such as Subscriber and NotificationProducer) by means of the WS-Addressing standard.

2.2.4 Solution proposals

The following text is provided in order to illustrate some possible ways to address the requirement. They are suggestions and are by no means to be considered as mandatory, as other possible options could be identified which are not represented hereafter.

To the best knowledge within OASIS SOA-TEL TC, the requirements presented hereafter could be addressed by the OASIS WS-Notification Technical Committee (WSN TC), which by the way is in status “Completed”, or possibly, by the W3C Web Services Addressing (WS-A) WG, which by the way is as well in status “Completed”.

Another Working Group potentially interested to receive this requirement is W3C Resource Access since the topic dealt by the specifications (WS-Transfer, WS-ResourceTransfer, WS-Enumeration, WS-MetadataExchange and WS-Eventing Member Submissions) for which this group is responsible may potentially solve the present issues with WS-N specification.

There are several approaches to solve the requirement: the solution to adopt depends on the chosen perspective, on the use cases that are to be covered, and on the scope to assign to the new specification.

Two different lines of solution, not antithetical, but complementary, are provided below. In the first proposal the intermediary plays an active part in the notification services, while the second proposal is more general, and is based on the fact that WS-Notification is supported by WS-Addressing.

First proposal (intermediary plays an active part in the notification services)

The WS-Notification specification should define a new role in addition to the ones already defined (NotificationConsumer, NotificationProducer, SubscriptionManager, Subscriber).

The new role could be named, for example, “Intermediary”, and its description could be:

- *an entity acting on behalf of a Subscriber; it receives a subscription request and asks for the subscription to the NotificationConsumer specified in the request, or forwards the request the next Intermediary;*
- *an entity acting on behalf of a NotificationProducer; it receives a notification and sends it to the NotificationConsumer specified in the notification message, or forwards the request to the next Intermediary.*

To be noted that an Intermediary node could contemporarily have both behaviours: acting on behalf of a *Subscriber* to request a subscription to a *NotificationProducer*, and acting on behalf of a Notification Producer to send a notification message to a *Subscriber*.

The protocol should be extended in such as way to define a new message exchange pattern in which even the Intermediary behaviour is comprised.

The syntax of the subscription request and that of the notification should be extended so that it becomes possible to specify, in the new messages, one or more intermediary destinations and the final destination.

For example, for the subscription operation, if the Subscriber knows the NotificationProvider location, it can make a subscription request in which it inserts an endpoint reference element for the NotificationProvider, and then sends the message to the Intermediary; the Intermediary consumes (reads and deletes) the reference and so it is able to send a subscribe request to the NotificationProvider.

In the subscription request, the endpoint reference of the Intermediary to which notifications should be sent, could be also included.

The subscribe message could be as the following:

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/Intermediary/SubscribeRequest
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:Subscribe>
      <wsnt:ConsumerReference>
        <wsa:Address>
          http://www.example.org/NotificationConsumer
        </wsa:Address>
      </wsnt:ConsumerReference>
      <wsnt:ProducerReference>
        <wsa:Address>
          http://www.example.org/NotificationProducer
        </wsa:Address>
      </wsnt:ProducerReference>
      <wsnt: IntermediaryReference>
        <wsa:Address>
          http://www.example.org/Intermediary
        </wsa:Address>
      </wsnt: IntermediaryReference>
      <wsnt:Filter>
        <wsnt:TopicExpression Dialect=
     "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          npex:SomeTopic
        </wsnt:TopicExpression>
        <wsnt:MessageContent
            Dialect="http://www.w3.org/TR/1999/REC-xpath-19991116">
          boolean(ncex:Producer="15")
        </wsnt:MessageContent>
      </wsnt:Filter>
      <wsnt:InitialTerminationTime>
        2005-12-25T00:00:00.00000Z
      </wsnt:InitialTerminationTime>
    </wsnt:Subscribe>
  </s:Body>
</s:Envelope>
```

The Intermediary receives the above message and makes a subscription request to the notification consumer with the following message:

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-
2/NotificationProducer/SubscribeRequest
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:Subscribe>
      <wsnt:ConsumerReference>
        <wsa:Address>
          http://www.example.org/NotificationConsumer
        </wsa:Address>
      </wsnt:ConsumerReference>
      <wsnt: IntermediaryReference>
        <wsa:Address>
          http://www.example.org/Intermediary
        </wsa:Address>
      </wsnt: IntermediaryReference>
      <wsnt:Filter>
        <wsnt:TopicExpression Dialect=
     "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          npex:SomeTopic
        </wsnt:TopicExpression>
        <wsnt:MessageContent
            Dialect="http://www.w3.org/TR/1999/REC-xpath-19991116">
          boolean(ncex:Producer="15")
        </wsnt:MessageContent>
      </wsnt:Filter>
      <wsnt:InitialTerminationTime>
        2005-12-25T00:00:00.00000Z
      </wsnt:InitialTerminationTime>
    </wsnt:Subscribe>
  </s:Body>
</s:Envelope>
```

The notification message could be the similar to these defined with the current specification, but sent by the NotificationProducer to the Intermediary rather than directly to the NotificationConsumer, as showed in the next figure; in this message the final destination should be present.

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/Intermediary/Notify
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:Notify>
      <wsnt:NotificationMessage>
        <wsnt:SubscriptionReference>
          <wsa:Address>
            http://www.example.org/SubscriptionManager
          </wsa:Address>
        </wsnt:SubscriptionReference>
        <wsnt:Topic Dialect=
     "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          npex:SomeTopic
        </wsnt:Topic>
      <wsnt:ConsumerReference>
        <wsa:Address>
          http://www.example.org/NotificationConsumer
        </wsa:Address>
      </wsnt:ConsumerReference>
        <wsnt:ProducerReference>
          <wsa:Address>
            http://www.example.org/NotificationProducer
          </wsa:Address>
        </wsnt:ProducerReference>
        <wsnt:Message>
          <npex:NotifyContent>exampleNotifyContent</npex:NotifyContent>
        </wsnt:Message>
      <wsnt:NotificationMessage>
    </wsnt:Notify>
  </s:Body>
</s:Envelope>
```

Second proposal (more general proposal, is based on the fact that WS-Notification is supported by WS-Addressing)

The WS-Addressing specification should be extended so that it expresses the concept of “final destination” of the message, by adding a new element, named for example <was:FinalTo>, in addition to those already present.

In this way the subscriber could specify both the NotificationProducer and the NotificationConsumer as final destinations in the subscription message.

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-
2/NotificationProducer/SubscribeRequest
    </wsa:Action>
    <wsa:FinalTo>
      <wsa:Address> http://www.example.org/NotificationProducer
</wsa:Address>
    </wsa:FinalTo>
    ...
  </s:Header>
  <s:Body>
    <wsnt:Subscribe>
      <wsnt:ConsumerReference>
         <wsa:FinalTo>
          <wsa:Address>
            http://www.example.org/NotificationConsumer
          </wsa:Address>
         </wsa:FinalTo>
      </wsnt:ConsumerReference>
      <wsnt:Filter>
        <wsnt:TopicExpression Dialect=
     "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          npex:SomeTopic
        </wsnt:TopicExpression>
        <wsnt:MessageContent
            Dialect="http://www.w3.org/TR/1999/REC-xpath-19991116">
          boolean(ncex:Producer="15")
        </wsnt:MessageContent>
      </wsnt:Filter>
      <wsnt:InitialTerminationTime>
        2005-12-25T00:00:00.00000Z
      </wsnt:InitialTerminationTime>
    </wsnt:Subscribe>
  </s:Body>
</s:Envelope>
```

The intermediary can send the message to the NotificationProducer without the necessity to make any interpretation of the message.

As a consequence, the NotificationProducer knows the endpoints of the NotificationConsumer and of the intermediary to which reply to; so it can send a notification to the intermediary, specifying the NotificationConsumer as final destination.

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/NotificationConsumer/Notify
    </wsa:Action>
    <wsa:FinalTo>
      <wsa:Address> http://www.example.org/NotificationConsumer
</wsa:Address>
    </wsa:FinalTo>
...
  </s:Header>
  <s:Body>
    <wsnt:Notify>
      <wsnt:NotificationMessage>
        <wsnt:SubscriptionReference>
          <wsa:Address>
            http://www.example.org/SubscriptionManager
          </wsa:Address>
        </wsnt:SubscriptionReference>
        <wsnt:Topic Dialect=
     "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          npex:SomeTopic
        </wsnt:Topic>
        <wsnt:ProducerReference>
          <wsa:Address>
            http://www.example.org/NotificationProducer
          </wsa:Address>
        </wsnt:ProducerReference>
        <wsnt:Message>
          <npex:NotifyContent>exampleNotifyContent</npex:NotifyContent>
        </wsnt:Message>
      <wsnt:NotificationMessage>
    </wsnt:Notify>
  </s:Body>
</s:Envelope>
```

2.3 Requirements on SOAP

2.3.1 Identification of Use Case

Extract from [SOA-TEL 1.0], section 4.1 (rows 405 to 414):

The perceived technical gap suggested is that the SOAP specification should be modified in order to enable a SOAP Intermediary node to “forward” the SOAP Header in automatic mode (thus without the

Header reinsertion) even if such node performs some processing operation over the body of the SOAP message.

Another way of expressing this perceived gap is to state that currently only 3 roles are allowed for a SOAP Node (i.e. initial SOAP Sender, SOAP intermediary, SOAP ultimate receiver – section 2.1 of the SOAP 1.2 specification), while a probable fourth role enabling the simultaneous body processing and header forwarding of a specific SOAP message may be needed.

2.3.2 Requirement(s)

Req. 3

A new “Message Sender and Receiver concept” must be added in [SOAP 1.2] to model SOAP nodes which must forward the SOAP headers message, but also need to perform changes on the body of the message.

A new SOAP protocol must be added to manage the behavior of such nodes.

2.3.3 Description

As documented in the SOA-TEL TC “Use Cases and Issues” document, some SOAP nodes can’t be classified as “Ultimate SOAP Receivers” because they aren’t the real providers of the service, but can’t be simple “SOAP Intermediaries”, because they need to perform changes on the body of the message: such nodes aren’t requestors or receivers, they need to process the SOAP header blocks, perform some changes on the body, and forward the message to the following node.

Hereafter a proposal definition of the new “SOAP functional intermediary” (the name is provisional and could be different) concept is provided:

- **SOAP functional intermediary**

  - *A SOAP functional intermediary is both a SOAP receiver and a SOAP sender and is targetable from within a SOAP message. It processes the SOAP header blocks targeted at it and acts to forward a SOAP message towards an ultimate SOAP receiver. Moreover a SOAP Functional Intermediary can process the contents of the SOAP body.*

This new concept and its functionalities of both processing the body of a message and of forwarding headers as a usual “SOAP intermediary” are to be included in the SOAP specification.

2.3.4 Solution proposals

The following text is provided in order to illustrate some possible ways to address the Requirement. They are suggestions and are by no means to be considered as mandatory, as other possible options could be identified which are not represented hereafter.

To the best knowledge within OASIS SOA-TEL TC, the requirements presented hereafter could be addressed by the W3C “XML Protocol” Working Group, which produced the SOAP specification. Currently such group is in status “Completed”. For such reason, should the requirement be accepted, some preliminary investigations with W3C representatives are suggested to identify if within this SDO there are some WGs willing to consider and solve the issue.

Some modifications to [SOAP 1.2] are needed (but other parts of the specification may need to be revised and changed):

- Include the new concept definition in Section 1.5.3;
- Modify paragraphs 2.2 and 2.7 of [SOAP 1.2]. In particular, 2 cases are suggested.

Case 1

The SOAP functional intermediary typology is covered by the role “`next`”. In this case the SOAP intermediary and SOAP functional intermediary act in a very similar way.

In this case Table 2 in section 2.2 should be modified as follows, while no changes should be required for table 3 at section 2.7.1.

Table 2: SOAP Roles defined by this specification

Short-name	Name	Description
`next`	"http://www.w3.org/2003/05/soap-envelope/role/next"	Each SOAP intermediary, SOAP functional intermediary, and the ultimate SOAP receiver MUST act in this role.
`none`	"http://www.w3.org/2003/05/soap-envelope/role/none"	SOAP nodes MUST NOT act in this role.
`ultimateReceiver`	"http://www.w3.org/2003/05/soap-envelope/role/ultimateReceiver"	The ultimate receiver MUST act in this role.

Case 2

The SOAP functional intermediary typology is covered by the role “`ultimateReceiver`”. In this case Table 2 should be modified as follows:

Table 2: SOAP Roles defined by this specification

Short-name	Name	Description
`next`	"http://www.w3.org/2003/05/soap-envelope/role/next"	Each SOAP intermediary, and the ultimate SOAP receiver MUST act in this role.
`none`	"http://www.w3.org/2003/05/soap-envelope/role/none"	SOAP nodes MUST NOT act in this role.
`ultimateReceiver`	"http://www.w3.org/2003/05/soap-envelope/role/ultimateReceiver"	The ultimate receiver and SOAP functional intermediary, MUST act in this role.

Moreover, table 3 in section 2.7.1 should be modified as follows:

Table 3: SOAP Nodes Forwarding behavior

Role		Header block	
Short-name	**Assumed**	**Understood & Processed**	**Forwarded**
`next`	Yes	Yes	No, unless reinserted
		No	No, unless `relay` ="true"
user-defined	Yes	Yes	No, unless reinserted

		No	No, unless `relay` ="true"
	No	n/a	Yes
`ultimateReceiver`	Yes	Yes	No, unless reinserted
		No	No, unless `relay` ="true"
`none`	No	n/a	Yes

3 Requirements on Security

3.1 Requirements on Security Token Correlation

3.1.1 Identification of Use Case

Currently it is not possible to correlate a security token with another one, previously created.

Refer section 5-1 of [SOA-TEL 1.0], in which the technical issue is documented.

3.1.2 Requirement(s)

Req. 4

The WS Security specifications must enable to express a relation between two security tokens, a “main” token (e.g. named “*token2*”) and a “related” token (e.g. named “*token1*”).

The characteristics of the relation are that, when the token correlation is used,

- the “main” token can not be built without being in possession of the “related” token,
- the WS-Sec header should not be considered valid if the “related” token is not present.

This token correlation requirement defines a new token security model, in which a “main” token is syntactically and semantically meaningful if it is built and presented in relation with another “related” token.

SOA-TEL Req. 4.1

It must be possible to express “token correlation” also into the SAML assertion.

3.1.3 Description

This token correlation requirement extends the message security models and enforces the security mechanism in environments where the message exchange pattern is more complex than the simple “requestor – provider” pattern.

This model should be useful when the definition and the use of a “simple” token doesn’t guarantee a sufficient level of security, since the authorization to access a specific service also depends on the fact that a previous token was released.

The possible “status” of the “related” token could be valid or expired (i.e. not valid anymore).

In the new token typology to be introduced, the “related” token is not a simple “attribute”, inserted only for traceability purposes into the header, but instead is an integral part of the token.

The identity provider should release the security token directly made up of two parts: the “main” and the “related” tokens.

3.1.4 Solution proposals

The following text is provided in order to illustrate some possible ways to address the Requirement. They are suggestions and are by no means to be considered as mandatory, as other possible options could be identified which are not represented hereafter.

[WS-S 1.1] defines three types of security tokens and how they are attached to messages (“user name token”, “binary security token” and “XML token”), and furthermore the syntax provides 2 elements to include tokens in the security header:

- <wsse:UsernameToken>
- <wsse:BinarySecurityToken>.

A new element should be added, named for example <wsse:AssociatedToken> to the previous ones.

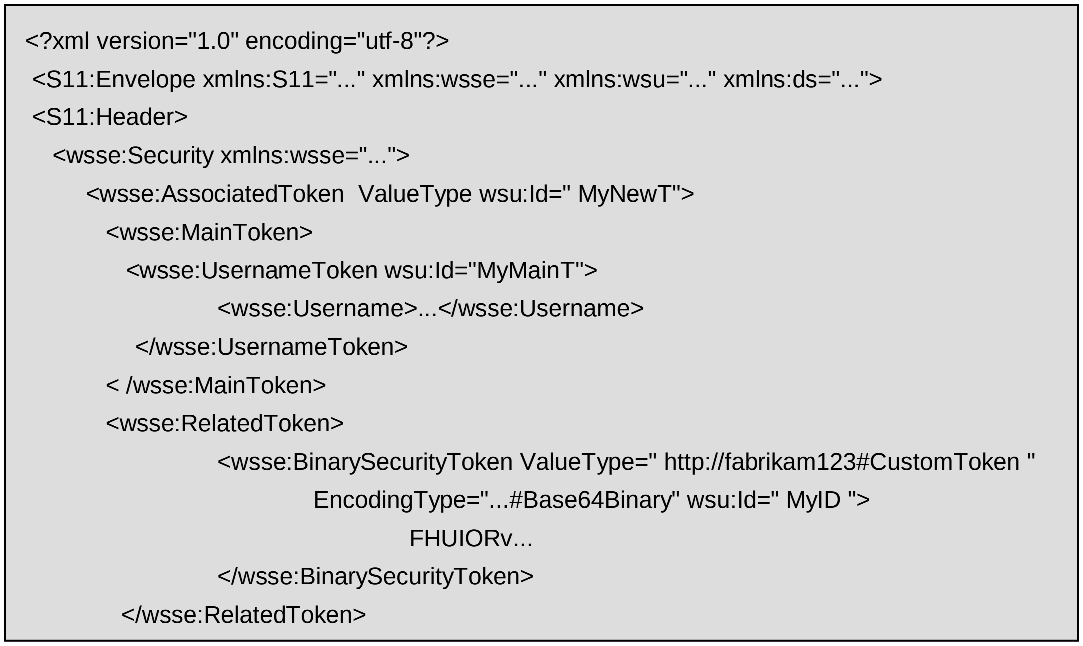
The <wsse: AssociatedToken> could contain (in a recursive manner) a username token, or a binary token, or a XML token element, or again a related token, for the “main” token.

The same should be for the “related” token.

This could be the syntax of the element:

```
<wsse: AssociatedToken>
        <wsse:MainToken>
                ………
        </wsse: MainToken>
        <wsse:RelatedToken>
                ………
        </wsse:RelatedToken>
</wsse:AssociatedToken>
```

This is an example of associated token:



```
<?xml version="1.0" encoding="utf-8"?>
 <S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="..." xmlns:ds="...">
 <S11:Header>
   <wsse:Security xmlns:wsse="...">
       <wsse:AssociatedToken  ValueType wsu:Id=" MyNewT">
         <wsse:MainToken>
           <wsse:UsernameToken wsu:Id="MyMainT">
                     <wsse:Username>...</wsse:Username>
            </wsse:UsernameToken>
         < /wsse:MainToken>
         <wsse:RelatedToken>
                     <wsse:BinarySecurityToken ValueType=" http://fabrikam123#CustomToken "
                               EncodingType="...#Base64Binary" wsu:Id=" MyID ">
                                         FHUIORv...
                     </wsse:BinarySecurityToken>
           </wsse:RelatedToken>
```

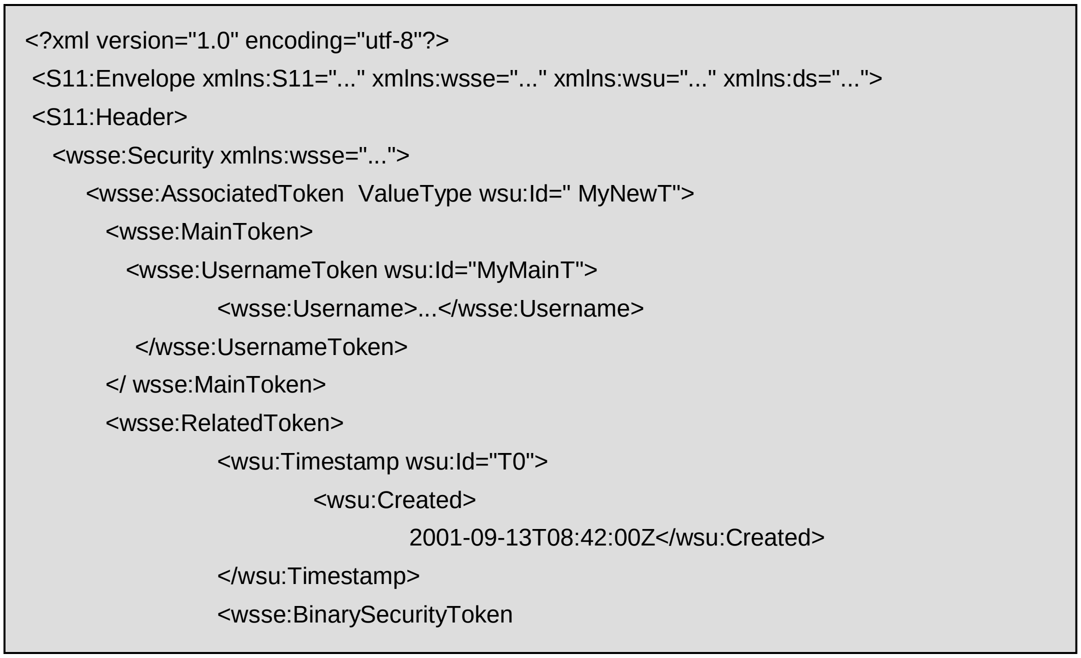
The <wsse:AssociatedToken> element could have other significant elements (other than the related token value) useful to the definition of the context in which the main token was built; for example it could include the timestamp value present in the security header from which the related token derive. Examples of other significant elements may also be (but not limited to) the ones currently defined within the three above mentioned security tokens types.

In other worlds if the related security token belonged to the following header:

```
<S11:Header>
 <wsse:Security>
        <wsu:Timestamp wsu:Id="T0">
                <wsu:Created>
                        2001-09-13T08:42:00Z</wsu:Created>
        </wsu:Timestamp>

 <wsse:BinarySecurityToken
        ValueType="...#X509v3"
        wsu:Id="X509Token"
        EncodingType="...#Base64Binary">
                MIIEZzCCA9CgAwIBAgIQEmtJZc0rqrKh5i...
 </wsse:BinarySecurityToken>
```

The AssociatedToken in the new header should be the following:



```
<?xml version="1.0" encoding="utf-8"?>
 <S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="..." xmlns:ds="...">
 <S11:Header>
    <wsse:Security xmlns:wsse="...">
        <wsse:AssociatedToken  ValueType wsu:Id=" MyNewT">
          <wsse:MainToken>
            <wsse:UsernameToken wsu:Id="MyMainT">
                     <wsse:Username>...</wsse:Username>
             </wsse:UsernameToken>
          </ wsse:MainToken>
          <wsse:RelatedToken>
                     <wsu:Timestamp wsu:Id="T0">
                             <wsu:Created>
                                     2001-09-13T08:42:00Z</wsu:Created>
                     </wsu:Timestamp>
                     <wsse:BinarySecurityToken
```

Clearly this mechanism is particularly meaningful when the related token is a SAML assertion that supplies all the information to describe the context in which the main token was built, that is the objective of the requirement.

In a similar way the SAML protocol could be extended to support the requirement.

In this case a new AssociatedToken element could be added into the SAML syntax, so the related token could be included directly in the SAML assertion constituting the main token, without the necessity of express the relation to the Ws security header level.

3.2 SAML Name Identifier Request

3.2.1 Identification of Use Case

A user device, a Service Provider (SP) and an Identity Provider (IdP) are the actors of this use case. The SP is new to the circle of trust of the IdP. The IdP does not know a name identifier of the user device. The IdP requests a name identifier from the SP, who sends the desired name identifier to the IdP.

Section 5.2.2 in [SOA-TEL 1.0] describes a use case for the proposed SAML Name Identifier Request-Response protocol.

3.2.2 Requirement(s)

Req. 5

In order to make the [SAML 2.0] support name identifier use cases such as that described in section 3.2.1, the Security Services TC must specify a

- <NameIdentifierRequest> message sent from an Identity Provider to a Service Provider to request a name identifier for a User, and a
- <NameIdentifierResponse> message sent from the Service Provider to the Identity Provider to return such a name identifier to the Identity Provider.

This requires extensions to the existing [SAML 2.0] core specification (saml-core-2.0-os) including the SAML 2.0 protocol schema. No modification of the existing SAML 2.0 assertion schema is necessary.

Description

Figure 4 provides a high-level message flow illustrating the proposed SAML Name Identifier request-respone protocol. Messages 4 and 6 belong to the proposed SAML Name Identifier Request protocol. These messages are interlaced into the SAML Authentication Request and Response exchange between SP and IdP and are not specified in SAML V2.0 yet (therefore, marked in red):

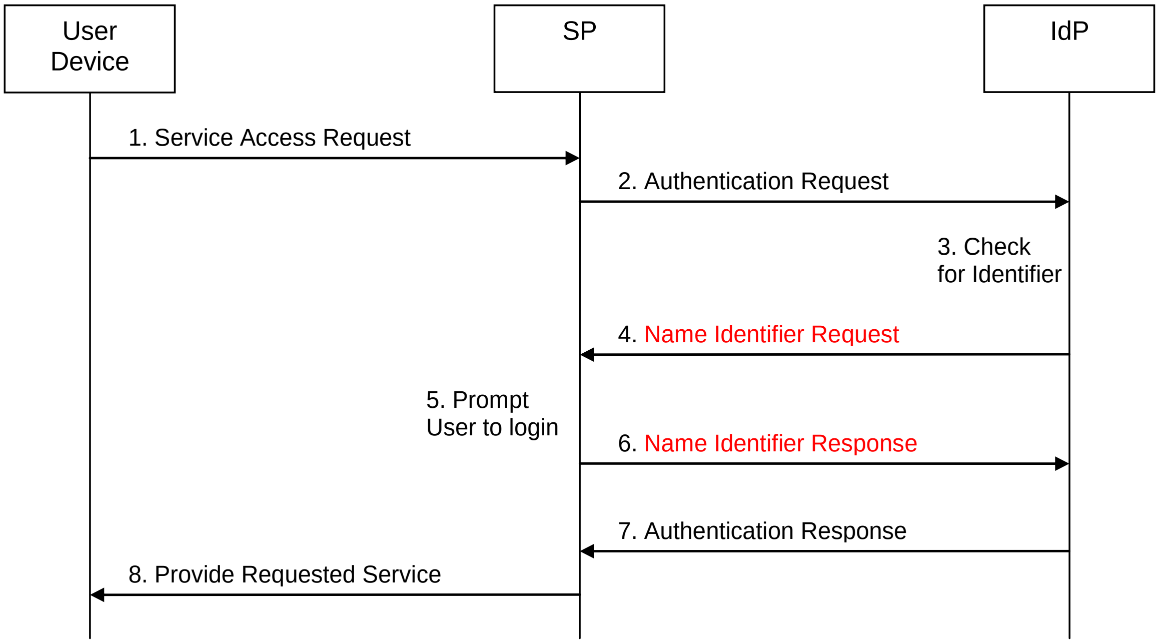

User Device
SP
IdP
1. Service Access Request
2. Authentication Request
3. Check for Identifier
4. Name Identifier Request
5. Prompt User to login
6. Name Identifier Response
7. Authentication Response
8. Provide Requested Service

Figure 4: SAML Name Identifier request-response use case: pictorial representation

The single steps of this use case are as follows:

1) The user requests access to a service offered by a SP. The user device does not include any authentication credentials.
2) Since access to this service requires the User to be authenticated but the request in step 1 does not include any authentication credentials, the SP sends an Authentication Request to the IdP. This Authentication Request may be passed to the IdP via the user device using redirection.
3) The IdP checks the Authentication Request received in step 2, and - as the SP is new to the IdP's circle of trust - the IdP determines that it does not have an identifier stored in its database for the User for the given SP.
4) This step is not defined in SAML V2.0: Since the IdP has realized in step 3 that it does not have an identifier for the combination of the User and the SP, the IdP generates a message called Name Identifier Request and sends it to the SP.
5) Upon receipt of the Name Identifier Request, the SP recognises that the IdP does not have an identifier for the combination of SP and User. Therefore, the SP prompts the User to log in to the SP.
6) This step is also not defined in SAML V2.0: The SP sends a message called Name Identifier Response to the IdP. This response message includes the identifier for the combination of User and SP that the IdP is to use in any further communication and authentication processes.
7) On receipt of the Name Identifier Response, the IdP stores the identifier contained in the Name Identifier Response in its database. The IdP sends an Authentication Response to the SP, which uses the identifier received in step 6.
8) The SP grants the User access to the requested service.

In step 3 of the message exchange illustrating a SAML Name Identifier use case above, conventionally, the IdP would respond to the Authentication Request (step 2) by issuing an error message or a randomly generated identifier. This, however, is problematic: In the former case, the service access request in step 1 breaks down. In the latter case, the SP has to ask the user for his credentials and then send (usually via a backchannel) a message to the IdP indicating that from now on the IdP should use the "real identifier" instead of the random one for the given user (this could be done via the NameIdentifier Management Protocol).

These issues can be resolved on SAML protocol level by defining <NameIdentifierRequest> and <NameIdentifierResponse> messages enabling the Identity Provider to request from a Service Provider a name identifier for a User and the Service Provider to send such a name identifier back to the Identity Provider.

3.2.3 Solution proposal

Extension of the SAML 2.0 protocol schema by <NameIdentifierRequest> and <NameIdentifierResponse> messages, instances of which are exemplified as follows:

Name Identifier Request:

```
<samlp:NameIdentifierRequest
      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
      ID="aaf23196-1773-2113-474a-fe114412ab72"
      Version="2.0"
      IssueInstant="2006-07-17T20:31:40Z">
      <saml:Issuer
            Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
            http://idm.nsn.com
        </saml:Issuer>
    </samlp:NameIdentifierRequest>
```

Name Identifier Response:

```
<samlp:NameIdentifierResponse
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    ID="aaf23196-1773-2113-474a-fe114412ab72"
    Version="2.0"
    IssueInstant="2006-07-17T20:31:40Z">

    <saml:Assertion
        MajorVersion="1" MinorVersion="0"
        AssertionID="128.9.167.32.12345678"
        Issuer="Smith Corporation">
        <saml:Issuer
            Format="urn:oasis:names:tc:SAML:1.1:nameid-
            format:X509SubjectName">
            C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
        </saml:Issuer>
        <saml:Subject>
            <saml:NameID
                Format="urn:oasis:names:tc:SAML:1.1:nameid-
                format:unspecified">
                tom.smith
            </saml:NameID>
        </saml:Subject>

        <saml:AttributeStatement>
            <saml:Attribute
                xmlns:x500="urn:oasis:names:tc:SAML:2.0:
                profiles:attribute:X500"
                x500:Encoding="LDAP"
                NameFormat="urn:oasis:names:tc:SAML:2.0:
                attrname-format:uri"
                Name="urn:oid:2.5.4.42"
                FriendlyName="givenName">
                <saml:AttributeValue xsi:type="xs:string">
                    Tom
                </saml:AttributeValue>
            </saml:Attribute>

            <saml:Attribute
                xmlns:x500="urn:oasis:names:tc:SAML:2.0:
                        profiles:attribute:X500"
                        x500:Encoding="LDAP"
                        NameFormat="urn:oasis:names:tc:SAML:2.0:
                        attrname-format:uri"
                        Name="urn:oid:1.3.6.1.4.1.1466.115.121.1.26"
                        FriendlyName="mail">
                        <saml:AttributeValue xsi:type="xs:string">
                              trscavo@gmail.com
                        </saml:AttributeValue>
                  </saml:Attribute>
            </saml:AttributeStatement>
      </saml:Assertion>
      <samlp:Status xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">
            <samlp:StatusCode
            xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
            Value="urn:oasis:names:tc:SAML:2.0:status:Success">
            </samlp:StatusCode>
      </samlp:Status>
</samlp:NameIdentifierResponse>
```

3.3 SAML Attribute Management Request

3.3.1 Identification of Use Case

A user wishes to use his attribute information across multiple service providers. Such attribute information can be layout, preferred email address, etc. Today, these attributes are stored locally at each service provider. Thus, the user will have to enter and change the same attributes multiple times in order to ensure they are consistent for each of the different service providers the user has an account with, resulting in a bad user experience.

The user creates a temporary or transient account. The service provider allows the user to set specific settings like coloring, text size, etc. But he/she does not want to set these setting again each time the user logs in because the service provider will not be able to link the attributes for a user's temporary account with the user's permanent account. This is because by the very nature of a temporary or transient account the next time the user logs on to the service provider the user will have a different user name and so the service provider will not be able to link the attributes for a user's temporary account with the user's permanent account.

Section 5.3.2 in [SOA-TEL 1.0] describes a use case for the proposed SAML Attribute Management Request-Response protocol.

3.3.2 Requirement(s)

Req. 6

In order to make the [SAML 2.0] support attribute management use cases such as that described in 3.3.1, the Security Services TC must specify a

- <ManageAttributeRequest> message sent from a Service Provider to an Identity Provider to request a modification or the storage of an attribute, and a

- <ManageAttributeResponse> message sent from the Identity Provider to the Service Provider to return to the Service Provider the result of processing the received <ManageAttributeRequest> message.

This requires extensions to the existing SAML 2.0 core specification (saml-core-2.0-os) including the SAML 2.0 protocol schema. No modification of the existing SAML 2.0 assertion schema is necessary.

3.3.3 Description

Figure 5 provides a high-level message flow outlining the proposed SAML Attribute Management protocol:

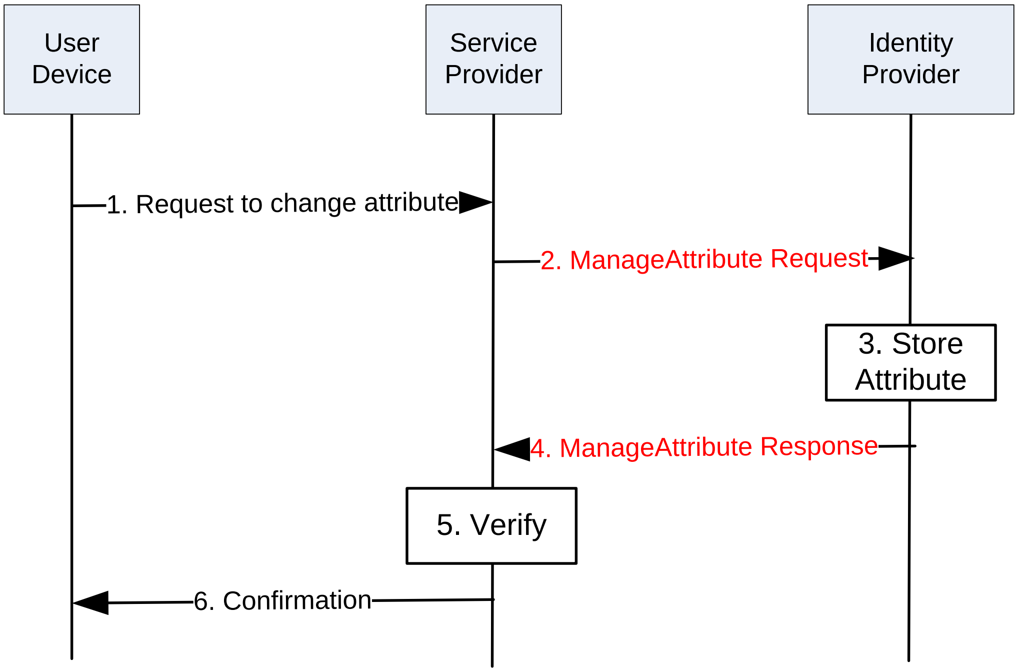


Figure 5: SAML Attribute Management request-response use case: pictorial representation

The Manage Attribute Request and Response messages are marked in red since the SAML 2.0 does not support such messages yet. The ManageAttribute Request allows the Service Provider to manage attributes stored on the Identity Provider side.

3.3.4 Solution proposal

Extension of the SAML 2.0 protocol schema by <ManageAttributeRequest> and <ManageAttributeResponse> messages, instances of which are exemplified as follows:

Manage Attribute Request:

```
<samlp:ManageAttributeRequest
      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
      ID="aaf23196-1773-2113-474a-fe114412ab72"
      Version="2.0"
      IssueInstant="2006-07-17T20:31:40Z">
      <saml:Issuer
            Format="urn:oasis:names:tc:SAML:1.1:nameidformat:
            X509SubjectName">
            C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
      </saml:Issuer>

      <saml:Subject>
            <saml:NameID
                  Format="urn:oasis:names:tc:SAML:1.1:nameidformat:X50
                  SubjectName">
                  C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
            </saml:NameID>
      </saml:Subject>
      <saml:AttributeStatement>
            <saml:Attribute
                  xmlns:x500="urn:oasis:names:tc:SAML:2.0:profiles:
                  attribute:X5 00"  x500:Encoding="LDAP"
                  NameFormat="urn:oasis:names:tc:SAML:2.0:
                  attrname-format:uri"
                  Name="urn:oid:2.5.4.42"
                  FriendlyName="givenName">
                  <saml:AttributeValue
                        xsi:type="xs:string">
                        John
                  </saml:AttributeValue>
            </saml:Attribute>
            <saml:Attribute
                  xmlns:x500="urn:oasis:names:tc:SAML:2.0:profiles:
                  attribute:X500" x500:Encoding="LDAP"
                  NameFormat="urn:oasis:names:tc:SAML:2.0:
                  attrname-format:uri"
                  Name="urn:oid:1.3.6.1.4.1.1466.115.121.1.26"
                  FriendlyName="mail">
                  <saml:AttributeValue
                        xsi:type="xs:string">
                        johndoe@gmail.com
                  </saml:AttributeValue>
            </saml:Attribute>
      </saml:AttributeStatement>
</samlp:ManageAttributeRequest>
```

Manage Attribute Response:

```
<samlp:ManageAttributeResponse
      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
      ID="aaf23196-1773-2113-474a-fe114412ab72"
      Version="2.0"
      IssueInstant="2006-07-17T20:31:40Z">
      <saml:Assertion
            MajorVersion="1" MinorVersion="0"
            AssertionID="128.9.167.32.12345678"
            Issuer="Smith Corporation">
            <saml:Issuer
                  Format="urn:oasis:names:tc:SAML:1.1:
                  nameid-format:unspecified">
                  http://idm.nsn.com
            </saml:Issuer>
            <saml:Subject>
                  <saml:NameID
                        Format="urn:oasis:names:tc:SAML:1.1:
                        nameid10format:X509SubjectName">
                        C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
                  </saml:NameID>
            </saml:Subject>
            <saml:AttributeStatement>
                  <saml:Attribute
                        xmlns:x500="urn:oasis:names:tc:SAML:2.0:
                        profiles:attribute:X500"
                        x500:Encoding="LDAP"
                        NameFormat="urn:oasis:names:tc:SAML:2.0:
                        attrname-format:uri"
                        Name="urn:oid:2.5.4.42"
                        FriendlyName="givenName">
                        <saml:AttributeValue
                              xsi:type="xs:string">
                              John
                        </saml:AttributeValue>
                  </saml:Attribute>
                  <saml:Attribute
                        xmlns:x500="urn:oasis:names:tc:SAML:2.0:
                        profiles:attribute:X500"
                        x500:Encoding="LDAP"
                        NameFormat="urn:oasis:names:tc:SAML:2.0:
                        attrname-format:uri"
                        Name="urn:oid:1.3.6.1.4.1.1466.115.121.1.26"
                        FriendlyName="mail">
                        <saml:AttributeValue
                              xsi:type="xs:string">
                              trscavo@gmail.com
                        </saml:AttributeValue>
                  </saml:Attribute>
            </saml:AttributeStatement>
      </saml:Assertion>
      <samlp:Status
            xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">
            <samlp:StatusCode
                  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
                  Value="urn:oasis:names:tc:SAML:2.0:status:Success">
            </samlp:StatusCode>
      </samlp:Status>
</samlp:ManageAttributeResponse>
```

3.4 User ID Forwarding

3.4.1 Scenario/context

3.4.2 Identification of Use Case

Currently a standard way does not exist to add two (or more) credentials in one message.

Refer to section 5-4 of [SOA-TEL 1.0], in which the technical issue is documented.

3.4.3 Requirement(s)

Req. 7

The WS Security specifications must enable to bring two security credentials in the security header: the “main” credential (e.g. named “*credential2*”) and a “secondary” credential (e.g. named “*credential1*”).

The authentication and authorization process should be performed on the basis of the main credential; the secondary credential should be used to complete the security functionalities.

[SOA-TEL Req. 7.1]

It must be possible to express support two credentials also into the SAML assertion.

3.4.4 Description

The user-id forwarding requirement extends the message security models and enforces the security mechanism in environments where a second security credential is necessary to add functionalities to the basic security process.

This model should be useful when the process of authentication and authorization on the base of the credential provided in the security header is not enough, and other security functionalities have to be executed on a second credential, for example to complete the authorization process or to profile the data.

3.4.5 Solution proposals

The following text is provided in order to illustrate some possible ways to address the Requirement. They are suggestions and are by no means to be considered as mandatory, as other possible options could be identified which are not represented hereafter.

To the best knowledge within OASIS SOA-TEL TC, the requirements presented hereafter could be addressed by the OASIS Web Services Security (WSS) TC, which by the way is in status “Completed”, and possibly by the OASIS Security Services (SAML) TC.

Hereafter some suggestions are proposed.

The WS-Sec v1.1 specification defines the following elements:

/wsse:Security;

/wsse:Security/@S11:actor;

/wsse:Security/@S12:role;

/wsse:Security/@S11:mustUnderstand;

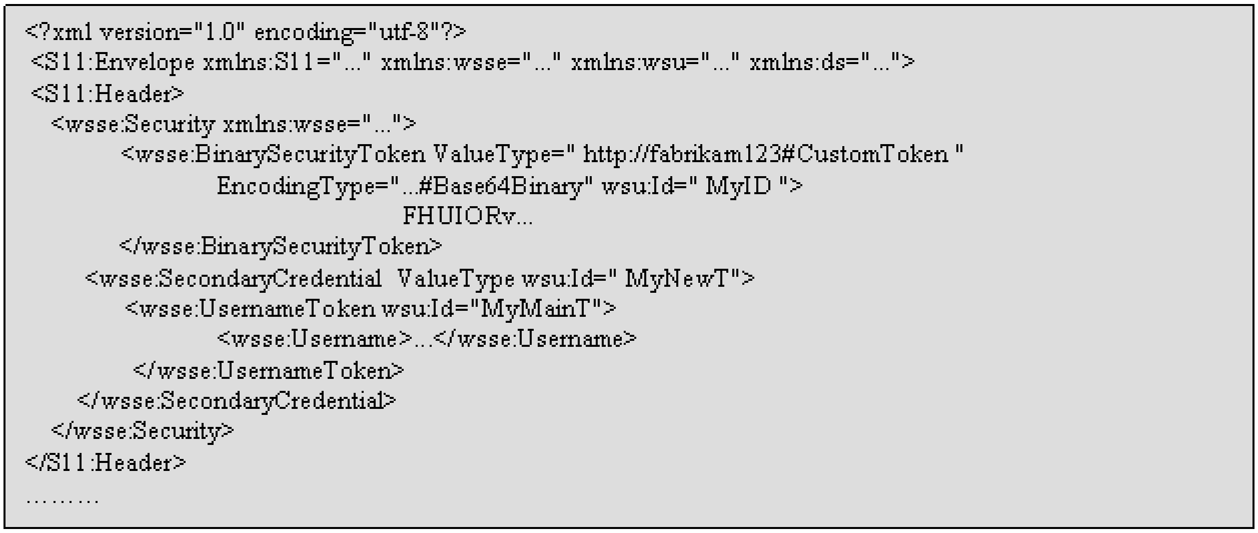
/wsse:Security/{any};

/wsse:Security/@{any};

Another element should be added, named for example:

/wsse:SecondaryCredential. This element should contain a security token, in particular one of the tokens provided by the current WS Security specification.

This is an example of header with a secondary credential, when the main credential is represented by a binary token, and the secondary by a user name and password token:



```
<?xml version="1.0" encoding="utf-8"?>
 <S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="..." xmlns:ds="...">
 <S11:Header>
   <wsse:Security xmlns:wsse="...">
          <wsse:BinarySecurityToken ValueType=" http://fabrikam123#CustomToken "
                    EncodingType="...#Base64Binary" wsu:Id=" MyID ">
                                  FHUIORv...
          </wsse:BinarySecurityToken>
       <wsse:SecondaryCredential  ValueType wsu:Id=" MyNewT">
           <wsse:UsernameToken wsu:Id="MyMainT">
                    <wsse:Username>...</wsse:Username>
            </wsse:UsernameToken>
      </wsse:SecondaryCredential>
   </wsse:Security>
</S11:Header>
………
```

In a similar way the SAML protocol could be extended to support the requirement.

In this case the “secondary credential” element could be added into the SAML syntax. In this way the related token could be included directly in the SAML assertion which constitutes the main token, without the necessity of express the relation to the WS security header level.

As an alternative path, the following hypothesis can be considered. This requirement (User-id forwarding requirement) is “intrinsically” similar to the “Security token correlation” requirement, presented elsewhere in the present document. Thus a common approach in modifying the WS-Security specifications could be adopted to address both the requirements and, more in general, similar security issues.

4 Requirements on Management

4.1 Cardinality of a Service Interface

4.1.1 Identification of Use Case

Extract from [SOA-TEL 1.0], section 6.3:

> [SOA-RM 1.0]: (Section 3.1) “A service is accessed by means of **a** service interface (see Section 3.3.1.4), where the interface comprises the specifics of how to access the underlying capabilities.”
> [SOA-RM 1.0]: (Subsection 3.3.1.4) “**The** service interface is the means for interacting with a service.”
> [SCA Assembly 1.1]: “A Service represents **an** addressable interface of the implementation.”
> Note – SCA definition for Service may be a consequence of the SOA-RM definition, we do not know

------

> [SOA-RA 1.0] (3137 – 3140) “In fact, managing a service has quite a few similarities to using a service: suggesting that we can use the service oriented model to manage SOA-based systems as well as provide them. A management service would be distinguished from a non-management service more by the nature of the capabilities involved (i.e., capabilities that relate to managing services) than by any intrinsic difference. “

4.1.2 Requirement(s)

Req. 8

The SOA Reference Model and Architecture must explain how a service separates and exposes its manageability capabilities to allow other services to manage it.

The Service Delivery Framework specified by TM Forum and depicted below sets such requirement at the SDF Service Management Interface (indicated in red in

Figure 6).

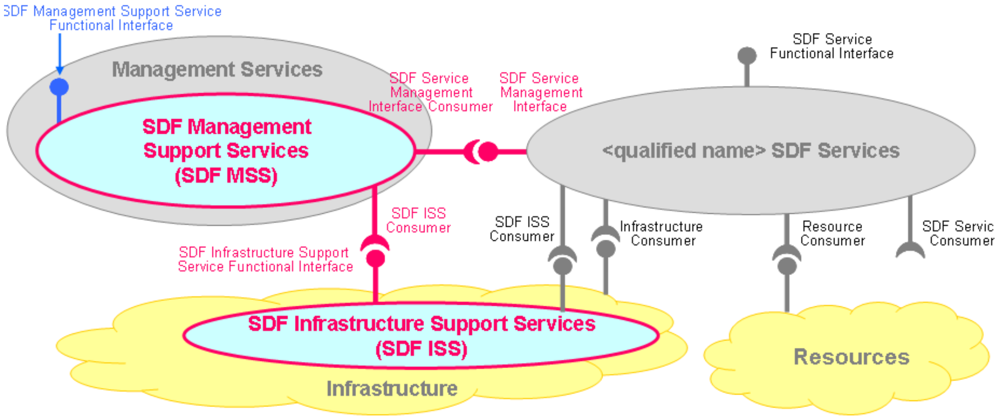

SDF Management Support Service Functional Interface
Management Services
SDF Management Support Services (SDF MSS)
SDF Service Management Interface Consumer
SDF Service Management Interface
SDF Service Functional Interface
<qualified name> SDF Services
SDF ISS Consumer
SDF Infrastructure Support Service Functional Interface
SDF ISS Consumer
Infrastructure Consumer
Resource Consumer
SDF Service Consumer
SDF Infrastructure Support Services (SDF ISS)
Infrastructure
Resources

Figure 6: TM Forum SDF Reference Model

4.1.3 Description

As documented in the SOA-TEL TC “Use Cases and Issues”, interfaces are the ways to interact with and between services and interfaces are the way to expose capabilities. At the same time, TM Forum SDF requires that SDF Services expose both Functional and Management capabilities and recommends this exposure to be made at separate interfaces attached to the SDF Service.

4.1.4 Solution proposals

OASIS SCA Assembly Model specification v1.1 offers a solution to the multiple interfaces problem as well as to “marking” an interface as being a management interface.

Updates to this specification (Committee Draft 03 rev 1.1 June 2009) offer also support for dynamic wiring of “service references” with “services” at run time through “autowire”, policy sets and SCA runtime re-evaluation of targets.

These proposals will be tested through TM Forum’s use case analysis and the results will be sent back to OASIS SCA Assembly team for further discussion.

Observations:

1. SCA Assembly Model covers only design, deployment and runtime as manageable capabilities (or management operations) for software bundles that constitute SDF Services. Other aspects of service lifecycle management such as quality, charging are not part of OASIS charter and will be further investigated by TM Forum in collaboration with other industry organizations.
2. SCA Assembly Model is not yet mapped to the OASIS SOA RA/RM.

4.2 Requirements on Metadata

4.2.1 Identification of Use Case

Extract from [SOA-TEL 1.0], section 6-4:

Specialization in supporting and managing a service during its whole lifecycle requires finer granularity knowledge about that service: properties, supported actions or operations, possible states as well as contracts that may govern interactions with the service (including pre and post conditions for these interactions), what is the “architectural” style for service “composability”, what are its dependencies or what is the level of exposure for its functional capabilities.

The proposed model for the TMF SDF Service is complemented by additional data representation (metadata) in support of SDF Service lifecycle management (ref. Section 6.4 – [SOA-TEL 1.0]). This new data representation containing information about the service in various phases of its lifecycle, aims at covering current gaps in the information available for the purpose of service management (e.g. what is already covered by the SOA Service description) in the overall context of Service Provider's business and operations. Moreover, this metadata is dynamic: it may change from one phase to another of the SDF Service lifecycle.

The SDF Service Lifecycle Metadata consists at least of:

1. Additional information about the SMI of a SDF Service (properties, actions);
2. Management Dependencies of the SDF Service, including cross-domains dependencies;
3. Management State of the SDF Service.

4.2.2 Requirement(s)

Req. 9

A standardization body (most probable TM Forum) must normalize the meta-data of Service Management to address the needs of managing any service from a lifecycle perspective. The meta-data should evolve into a meta-model that can be automatically instantiated into current and future management models which are domain (network or IT), technology (enterprise Java, IP network) or lifecycle phase (service creation, deployment, operation, etc).

4.2.3 Description

As documented in the SOA-TEL TC "Use Cases and Issues", paragraph 6.4, managing a service through its entire lifecycle requires finer granularity information (about the service, its execution environment, its dependencies, etc) than it is available today through management applications and tools. Moreover, this information, even when it is available (and most of it already exists) it comes in "bits and pieces", usually uncorrelated, from many places (tools, interfaces, environments) following diverse data models (SID, CIM, etc).

TM Forum SDF initiative believes that completing and unifying service management information through a well defined meta-data that describes and evolves with the lifecycle of each service instance is key to solving the issue of rapid service creation and launch.

The real problem to address is management across domains; the existence of different standards for metadata is an obstacle to the achievement of such objective.

4.2.4 Solution proposals

TM Forum SDF initiative started to define elements of service lifecycle management meta-data and show how they can be used in a service oriented management framework such as SDF (see fig 23 in OASIS UC document).

Nevertheless, TM Forum is not a data modeling or IT standards organization hence it raises the call to contributions to such organizations through OASIS SOA-Tel in the following areas:

- Representation of actions or state machines into meta-data (maybe OMG – UML 2.x)
- Support of versioning and compatibility of this meta-data
- Support of cohesiveness across metadata elements when they are updated from different sources and along the phases in the lifecycle of a service.
- Best design patterns for building and maintaining a repository for this meta-data

Today there is no clarity as to where to find such standards or if they exist and if they do not exist which organization should take the responsibility of working on them

5 Requirements on SOA collective standards usage

5.1 Common Patterns for Interoperable Service Based Communications

5.1.1 Identification of Use Case

This section is related to the specification of requirements related to the perceived technical issues identified in section 7, [SOA-TEL 1.0].

5.1.2 Requirement(s)

Req. 10

A common communications profile should be defined such that all multi tier web/ mobile applications declaring support for the profile will be able to establish a converged sessions irrespective of the underlying protocols, network domains and access across one or more servers/ services within or across different respective domains.

Such a profile will need to define an agreed to approach to:

1. Establish a session id for the context of converged application.
2. Ability to set up event sync supporting a common set of set of bi-directional event classes (i.e. push, broadcast, pub/sub, etc.).
3. Universally agreed to means to access the meta-data to discover the interface, binding, events classes, capability of service and device.
4. Common and agreed upon means/ nomenclature for an application in real-time to discover, advertise and negotiate device characteristics, codec’s and communication modes with a peer or set of peers.
    - Device attributes, communication protocols and media negotiation achieved through two way services interaction.
    - This interaction can default to common underlying negotiation means if available/ discoverable at setup time.

5.1.3 Description

The Internet has been enormously successful as en environment allowing user centric viral application growth. Its success, among other things, is the result of passing control to the end user and abstracting the underlying network details out of the picture for the application. As the name denotes, The Internet was designed to allow networks to interoperate. Unfortunately, communication oriented application models are more often bound to specific network domains with dependencies across different underlying VoIP protocols, competing standards, discovery data models and session negotiation and establishment.

There are a growing set of application models that serve a general web and mobile market that can not “build-in” assumptions of the underlying network or multi-modal connection establishment. The communication profile is an attempt to mitigate this problem. It does not seek to enforce one standard over the other but attempts to establish a general framework allowing converged applications to interoperate thru normalized patterns of session establishment and discovery.

6 Conformance

The objective of this document is to collect requirements to address technical issues and gaps of SOA standards (specified by OASIS and other SDOs) utilized within the context of Telecoms. Such issues are documented in SOA-TEL's TC first deliverable "Telecom Use Cases and Issues, v.1.0".

For each requirement listed in this document, a specific conformance rule applies. In the following are listed

Conformance to Requirement 1

A future version of WS Addressing specification must include additional fields (in addition to the ones already present) containing remote destinations to which reply messages must be sent.

Conformance to Requirement 2

A future version of WS-Notification specification must provide a mechanism to describe and regulate a scenario in which one or more intermediaries are present.

Conformance to Requirement 3

A future version of SOAP specifications must include a new "Message Sender and Receiver concept" to model SOAP nodes which must forward the SOAP headers message, but also need to perform changes on the body of the message

Conformance to Requirement 4

A future version of WS Security specifications must enable to express a relation between two security tokens, a "main" token (e.g. named "*token2*") and a "related" token (e.g. named "*token1*").

Conformance to Requirement 4.1

In a future version of the SAML Specification (or a new profile of this specification) it must be possible to express "token correlation" into the SAML assertion.

Conformance to Requirement 5

The SAML 2.0 protocol must support name identifier use cases by means of

- <NameIdentifierRequest> message sent from an Identity Provider to a Service Provider to request a name identifier for a User, and a
- <NameIdentifierResponse> message sent from the Service Provider to the Identity Provider to return such a name identifier to the Identity Provider.

.

Conformance to Requirement 6

The SAML 2.0 protocol must support attribute management use cases by means of

- <ManageAttributeRequest> message sent from a Service Provider to an Identity Provider to request a modification or the storage of an attribute, and a
- <ManageAttributeResponse> message sent from the Identity Provider to the Service Provider to return to the Service Provider the result of processing the received <ManageAttributeRequest> message.

Conformance to Requirement 7

A future version of WS Security specifications must enable to bring two security credentials in the security header: the “main” credential (e.g. named “*credential2*”) and a “secondary” credential (e.g. named “*credential1*”) so that the authentication and authorization process could be performed on the basis of the main credential, while the secondary credential could be used to complete the security functionalities.

Conformance to Requirement 7.1

In a future version of the SAML Specification (or a new profile of this specification) it must be possible to support two credentials into the SAML assertion.

Conformance to Requirement 8

A future version of the OASIS SOA Reference Model and Architecture must explain how a service separates and exposes its manageability capabilities to allow other services to manage it.

Conformance to Requirement 9

A standardization body (most probable TM Forum) must have normalized the meta-data of Service Management to address the needs of managing any service from a lifecycle perspective.

Conformance to Requirement 10

A common communications profile should have been defined such that all multi tier web/ mobile applications declaring support for the profile will be able to establish a converged sessions irrespective of the underlying protocols, network domains and access across one or more servers/ services within or across different respective domains.

Appendix A. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Mike Giordano	Individual
Ian Jones	BT
Paul Knight	Individual
Lucia Gradinariu	LGG Solutions
Orit Levin	Microsoft
Joerg.Abendroth	Nokia Siemens Networks
Christian Guenter	Nokia Siemens Networks
Thinn Nguyenphu	Nokia Siemens Networks
Olaf Renner	Nokia Siemens Networks
Abbie Barbir	Individual
Vincenzo Amorino	Telecom Italia
Luca Galeani	Telecom Italia
Maria Jose Mollo	Telecom Italia
Enrico Ronco	Telecom Italia
Federico Rossini	Telecom Italia
Luca Viale	Telecom Italia

1068 Appendix B. SOA-TEL Requirements

Req. 1	The WS Addressing specifications, [WS-A 1.0], must include additional fields (in addition to the ones already present) containing remote destinations to which reply messages must be sent. • The sender of a message must assign the fields when it wants to specify the destination for the reply message, but the node that has to use such destination information (i.e. the node that has to send the reply message) may not necessarily be the direct receiver of the request message. • The receiver of a message, which needs of information on the endpoint destination to which send a reply message, can obtain the information by these additional fields. • The receiver of a message has to forward to the next receiver all the additional destinations (present in these additional fields) that it does not use.
Req. 2	The WS-Notification specification must provide a mechanism to describe and regulate a scenario in which one or more intermediaries are present; it must standardize the terminology, concepts, operations, WSDL and XML needed to express the roles of the intermediaries (involved in publish and subscribe Web services for notification message exchange). According to the WS-Notification terminology, the standard must be extended and modified so that: • a *Subscriber* can require a *Subscription* to a *NotificationProducer* also in the case they do not communicate directly but do so by means of one or more intermediaries; • likewise a *NotificationProducer* can send a *Notification* to a *NotificationConsumer* also in the case that they do not communicate directly, but by means of one or more intermediaries.
Req. 3	A new "Message Sender and Receiver concept" must be added in [SOAP 1.2] to model SOAP nodes which must forward the SOAP headers message, but also need to perform changes on the body of the message. A new SOAP protocol must be added to manage the behavior of such nodes.
Req. 4	The WS Security specifications must enable to express a relation between two security tokens, a "main" token (e.g. named "*token2*") and a "related" token (e.g. named "*token1*"). The characteristics of the relation are that, when the token correlation is used, • the "main" token can not be built without being in possession of the "related" token, • the WS-Sec header should not be considered valid if the "related" token is not present. This token correlation requirement defines a new token security model, in which a "main" token is syntactically and semantically meaningful if it is built and presented in relation with another "related" token.
Req. 4.1	It must be possible to express "token correlation" also into the SAML assertion.
Req. 5	In order to make the [SAML 2.0] support name identifier use cases such as that described in section 3.2.1, the Security Services TC must specify a • <NameIdentifierRequest> message sent from an Identity Provider to a Service

	Provider to request a name identifier for a User, and a • <NameIdentifierResponse> message sent from the Service Provider to the Identity Provider to return such a name identifier to the Identity Provider. This requires extensions to the existing [SAML 2.0] core specification (saml-core-2.0-os) including the SAML 2.0 protocol schema. No modification of the existing SAML 2.0 assertion schema is necessary.
Req. 6	In order to make the [SAML 2.0] support attribute management use cases such as that described in 3.3.1, the Security Services TC must specify a • <ManageAttributeRequest> message sent from a Service Provider to an Identity Provider to request a modification or the storage of an attribute, and a • <ManageAttributeResponse> message sent from the Identity Provider to the Service Provider to return to the Service Provider the result of processing the received <ManageAttributeRequest> message. This requires extensions to the existing SAML 2.0 core specification (saml-core-2.0-os) including the SAML 2.0 protocol schema. No modification of the existing SAML 2.0 assertion schema is necessary.
Req. 7	The WS Security specifications must enable to bring two security credentials in the security header: the “main” credential (e.g. named “*credential2*”) and a “secondary” credential (e.g. named “*credential1*”). The authentication and authorization process should be performed on the basis of the main credential; the secondary credential should be used to complete the security functionalities.
Req. 7.1	It must be possible to support two credentials also into the SAML assertion.
Req. 8	The SOA Reference Model and Architecture must explain how a service separates and exposes its manageability capabilities to allow other services to manage it. The Service Delivery Framework specified by TM Forum and depicted below sets such requirement at the SDF Service Management Interface.
Req. 9	A standardization body (most probable TM Forum) must normalize the meta-data of Service Management to address the needs of managing any service from a lifecycle perspective. The meta-data should evolve into a meta-model that can be automatically instantiated into current and future management models which are domain (network or IT), technology (enterprise Java, IP network) or lifecycle phase (service creation, deployment, operation, etc).
Req. 10	A common communications profile should be defined such that all multi tier web/ mobile applications declaring support for the profile will be able to establish a converged sessions irrespective of the underlying protocols, network domains and access across one or more servers/ services within or across different respective domains. Such a profile will need to define an agreed to approach to: 1. Establish a session id for the context of converged application. 2. Ability to set up event sync supporting a common set of set of bi-directional event classes (i.e. push, broadcast, pub/sub, etc.). 3. Universally agreed to means to access the meta-data to discover the interface, binding, events classes, capability of service and device. 4. Common and agreed upon means/ nomenclature for an application in real-time to discover, advertise and negotiate device characteristics, codec’s and communication modes with a peer or set of peers. o Device attributes, communication protocols and media negotiation achieved through two way services interaction.

	This interaction can default to common underlying negotiation means if available/ discoverable at setup time.