OASIS Service Provisioning Markup Language (SPML) v2 - XSD Profile

Committee Draft 1.0
2005 September 14

Document identifier: pstc-spml2-xsd-profile-cd-01.pdf

Location: http://www.oasis-open.org/committees/provision/docs/

Send comments to: pstc-comment@lists.oasis-open.org

Editor:

Jeff Bohren, BMC (Jeff_Bohren@bmc.com)

Contributors:

Robert Boucher, CA
Doron Cohen, BMC
Gary Cole, Sun Microsystems
Cal Collingham, CA
Rami Elron, BMC
Marco Fanti, Thor Technologies
Ian Glazer, IBM
James Hu, HP
Ron Jacobsen, CA
Jeff Larson, Sun Microsystems
Hal Lockhart, BEA
Prateek Mishra, Oracle Corporation
Martin Raepple, SAP
Darran Rolls, Sun Microsystems
Kent Spaulding, Sun Microsystems
Gavenraj Sodhi, CA
Cory Williams, IBM
Gerry Woods, SOA Software

Abstract:

This specification defines usage of XML and XSD as a data model (profile) for SPML v2.

Status:

This is a candidate Committee Specification that is undergoing a vote of the OASIS membership in pursuit of OASIS Standard status.

If you are on the provision list for committee members, send comments there. If you are not on that list, subscribe to the provision-comment@lists.oasis-open.org list and send comments there. To subscribe, send an email message to provision-comment-request@lists.oasis-open.org with the word "subscribe" as the body of the message.

Table of contents

1. Introduction (non-normative)

1.1. Concepts

SPML Version 2 (SPMLv2) defines a core protocol **[SPMLv2]** over which different data models can be used to define the actual provisioning data. The combination of a data model with the SPML core specification is referred to as a binding. The use of SPML requires that a specific binding is used, although the choice of which binding is used to negotiated out-of-band by the participating parties.

This document describes the use of the XML and XSD as a data model for SPML based provisioning. This binding is optional.

1.2. Terminology

Within this document:
- The term “requestor” always refers to a Requesting Authority (RA).
- The term “provider” always refers to a Provisioning Service Provider (PSP).
- The term “target” always refers to a Provisioning Service Target (PST).
- The term “object” (unless otherwise qualified) refers to a Provisioning Service Object (PSO).
- The term “client” (unless otherwise qualified) refers to a Requesting Authority (RA).
- The term “server” (unless otherwise qualified) refers to a Provisioning Service Provider (PSP).

2. Notation

This specification contains schema conforming to W3C XML Schema and normative text to describe the syntax and semantics of XML-encoded policy statements.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in IETF RFC 2119 **[RFC2119]**

> *"they MUST only be used where it is actually required for interoperation or to limit behavior which has potential for causing harm (e.g., limiting retransmissions)"*

These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

This specification uses the following typographical conventions in text:

Format	Description	Indicates
`attributeName`	monospace font with *first letter lower-cased*	The name of an XML *attribute*.
`SPMLElementName`	monospace font with *first letter capitalized*	The name of an XML *element* that is defined as part of SPMLv2.
`ns:ForeignElementName`	monospace font *with namespace prefix*	The name of an XML element that is *defined by another specification.*

`<SPMLElement>`	monospace font *surrounded by* <>	*An instance of an XML element* that is defined as part of SPMLv2.
`<ns:ForeignElement>`	monospace font *with namespace prefix* surrounded by <>	An instance of an XML element that is *defined by another specification*.

Terms in ***italic bold-face*** are intended to have the meaning defined in the Glossary.

```
Listings of SPML schemas appear like this.

```

```
Example code listings appear like this.
```

Conventional XML namespace prefixes are used throughout the listings in this specification to stand for their respective namespaces as follows, whether or not a namespace declaration is present in the example:
- The prefix `saml:` stands for the SAML assertion namespace [**SAML**].
- The prefix `ds:` stands for the W3C XML Signature namespace [**DS**].
- The prefix `xsd:` stands for the W3C XML Schema namespace [**XS**].

3. Overview (non-normative)

3.1. XML PSOs

A PSO is represented in this binding by an XML structure. Thus structure should be defined by the XSD that is returned as the schema for the containing target.

3.1.1. PSO Identifier

The PSO Identifier may be any opaque identifier for the PSO, such as a GUID, URN, or XPath expression. If an XPath expression is used, it must resolve to a single PSO.

For instance if an opaque GUID is used for the PSO ID:

```
<spml:pso>
   <spml:psoID ID="2244" targetID="target2"/>
   …
</spml:pso>
```

If for instance an XPath is used for the PSO ID:

```
<spml:pso>
   <psoID ID="/Person/email='jdoe@acme.com'" targetID="target2"/>
   …
</spml:pso>
```

3.1.2. PSO Data

The PSO Data element contains a root XML element that conforms to the XSD schema defined by the target.

```
<spml:pso>
   …
   <spml:data>
      <user>
```

```
         <cn>John Doe</cn>
         <uid>jdoe<uid>
         <email>jdoe@acme.com</email>
         <phone>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </user>
   </spml:data>
</spml:pso>
```

3.2. Schema

The schema defines the allowed attributes and elements. For the XSD Profile, the PSO schema is defined using XSD. The XSD can be defined by inclusion in the spml:schema element, or by reference to an external or well known XSD schema URI.

For instance if the XSD is defined by inclusion:

```
<spml:schema>
   <xsd:schema>
      …
   </xsd:schema>
</spml:schema>
```

If the XSD is defined by reference:

```
<spml:schema ref=http://www.acme.com/schemas/user.xsd />
```

3.3. Core Operations

3.3.1. Add Request

The Add Request creates PSOs. The Add Request must contain a <data> element that contains an XML element that defines the new PSO. The Add Request may also pass a PSO Identifier (<psoId> element). If a PSO identifier is not defined in the Add Request, the new PSO Identifier must be returned in the Add Response.

```
<spml:addRequest targetID="target2">
   <spml:data>
      <user>
         <cn>John Doe</cn>
         <uid>jdoe<uid>
         <email>jdoe@acme.com</email>
         <phone>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </user>
   </spml:data>
</spml:addRequest >
```

3.3.2. Add Response

The Add Response would contain the status. If the request is successful, the response could include the new PSO ID and data. For instance:

```
<spml:addResponse status = "spml:success">
```

```
   <spml:psoID ID="2244" targetID="target2"/>
   <spml:data>
      <user>
         <cn>John Doe</cn>
         <uid>jdoe<uid>
         <email>jdoe@acme.com</email>
         <phone>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </user>
   </spml:data>
</spml:addResponse>
```

3.3.3. Modify Request

The Modify Request modifies PSOs. The Modify Request always contains the PSO Identifier. The modification type can be either add, replace, or delete. If the modification is not being made to the root XML element of the PSO data, the request would specify a selector XPath that uniquely identifies the sub-element being modified.

For instance to add a sub-element to the root element of the PSO data:

```
<spml:modifyRequest >
   <spml:psoID ID="2244" targetID="target2"/>
   <spml:modification modificationMode = "spml:add">
      <spml:component path="./phone" namespaceURI="http://www.w3.org/TR/xpath20"/>
      <spml:data>
         <mobile>555-1212</mobile>
      </spml:data>
   </spml:modification>
</spml:modifyRequest>
```

To replace a sub-element:

```
<spml:modifyRequest >
   <spml:psoID ID="2244" targetID="target2"/>
   <spml:modification modificationMode="spml:replace" >
      <spml:component path="./phone" namespaceURI="http://www.w3.org/TR/xpath20"/>
      <spml:data>
         <phone>
            <mobile>555-1212</mobile>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </spml:data>
   </spml:modification>
</spml:modifyRequest>
```

To delete a sub-element:

```
<spml:modifyRequest >
   <spml:psoID ID="2244" targetID="target2"/>
   <spml:modification modificationMode = "spml:delete" >
      <spml:component path="./phone" namespaceURI="http://www.w3.org/TR/xpath20"/>
   </spml:modification>
</spml:modifyRequest>
```

3.3.4. Delete Request

The Delete Request deletes PSOs. The Delete Request always contains the PSO Identifier.

```
<spml:deleteRequest>
   <spml:psoID ID="2244" targetID="target2"/>
</spml:deleteRequest >
```

3.3.5. Lookup Request

The Lookup Request returns the data for an identified PSO. The Lookup Request always contains the PSO Identifier.

```
<spml:lookupRequest returnData = "spml:everything">
   <spml:psoID ID="2244" targetID="target2"/>
</spml:lookupRequest>
```

3.3.6. Lookup Response

The Lookup Response (if successful) will return the data for the identified PSO.

```
<spml:lookupResponse>
   <spml:psoID ID="2244" targetID="target2"/>
   <spml:data>
      <user>
         <cn>John Doe</cn>
         <uid>jdoe<uid>
         <email>jdoe@acme.com</email>
         <phone>
            <mobile>555-1212</mobile>
            <home>555-2323</home>
            <work>555-6767x321</work>
         </phone>
      </user>
   </spml:data>
</spml:lookupResponse>
```

3.4. Search Operations

3.4.1. Search Request

The search request can specify a search base and an XPath selection statement.

```
<spmlsearch:searchRequest>
   <spmlsearch:query scope = "spmlsearch:oneLevel" targetID="target2">
       <spml:select>/user</spml:select>
   </spmlsearch:query>
</spmlsearch:searchRequest>
```

The select clause for the search request treats each target as a document root that (directly or indirectly) contains all other objects as nodes. So, for example,

- "/Person" would select every Person object that the target directly contains.
- "//Person" would select every Person object on a target,
  no matter which container was the Person object's parent.
- "/Group" would select every Group object that the target directly contains.
- "//Group" would select every Group object on a target,
  no matter which container was the Group object's parent.

3.4.2. Search Response

The search response, if successful, would contain all of the PSOs that satisfied the search criteria. For instance:

```
<spml:searchResponse status = "spml:success">
   <spml:pso>
      <spml:psoID ID="2244" targetID="target2"/>
      <spml:data>
         <user>
            <cn>John Doe</cn>
            <uid>jdoe<uid>
            <email>jdoe@acme.com</email>
         </user>
      </spml:data>
   </spml:pso>
   <spml:pso>
      <spml:psoID ID="2245" targetID="target2"/>
      <spml:data>
         <user>
            <cn>Jane Smith</cn>
            <uid>jsmith<uid>
            <email>jsmith@acme.com</email>
         </user>
      </spml:data>
   </spml:pso>
</spml:searchResponse>
```

4. Specification (Normative)

4.1. XPath Support

A provider MUST support the abbreviated syntax for XPath expressions. Put differently, a provider MUST support any XPath location path that does not include an explicit axis specifier.

A provider MAY support explicitly specified axes. A provider MAY support arbitrary XPath expressions. However, a requestor that deals with arbitrary providers should assume only that each provider supports location paths in the abbreviated syntax format.

Abbreviated Syntax. An XPath expression that uses only the abbreviated syntax contains no explicit axis specifier. Each step assumes the "child" axis by default. Any axis other than the “child” axis is specified by one of the following abbreviations:

- "@" is short for "attribute:"
- "//" is short for "/descendant-or-self::node()/"
- "." is short for self::node()
- ".." is short for parent:node()

Each target is a document root. A provider MUST treat each target as a document root that (directly or indirectly) contains all other objects as nodes.

4.2. Core Capability

4.2.1. Element <spml:data>

The <*spml:data*> element MAY contain any number of XML elements. The elements MUST conform to the XSD specified in the *spml:schema* for that target.

4.2.2. Element <spml:modification>

The <*spml:modification*> element MAY contain any number of XML elements. The <*spml:modification*> element MUST define the “modificationMode” attribute to be one of “add”, “replace”, or “delete”.

An <*spml:modification*> element MAY contain at most one <*component*> element. If the modification is on a sub-element of the PSO data, the *component* element MUST be set to the XPath state that uniquely identifies the sub-element withen the PSO data root element. If the modification is on the PSO data root element, the *component* element MAY be omitted.

An <*spml:modification*> element MAY contain at most one <*data*> element. If the <*spml:modification*> contains a <*component*> element, then the <*spml:modification*> MUST contain a <*data*> element.

Modification component. An <*spml:component*> element MUST have a “namespaceURI” attribute and MUST have a “path” attribute.

The value of the “namespaceURI” attribute MUST specify the XML namespace of a query language. The value of the “path” attribute MUST be an expression that is valid in the query language that “namespaceURI” specifies. (For example, if a requestor uses XPath 2.0 as the query language for the “path” attribute, the value of the “namespaceURI” attribute MUST be "http://www.w3.org/TR/xpath20".)

The value of the “path” attribute MUST specify an attribute or a sub-element (or an attribute of a sub-element) of the object that the provider is to modify. The specified attribute or element MUST be valid (according to the schema of the target) for the schema entity of which the object to be modified is an instance.

An <*spml:component*> element MAY include <*spml:namespacePrefixMap*> elements that defines the namespace prefixes that are used in the XPath path. Each “prefix” attribute on the <*spml:namespacePrefixMap*> element MUST exactly match one the namespace prefixes used in the Xpath.

Modification data. A requestor must specify as the content of the <data> sub-element of a <modification> any *value* that is to be added to, replaced within, or deleted from the element or attribute that the <component> element specifies.

- In the XML Schema profile, a requestor that specifies a <component> element within a <modification> element with “modificationMode='add'” or (within a <modification> element with) “modificationMode='modify'” MUST specify a value that is to replace the element or attribute that the <component> element specifies.
  - If the <component> element (XPath expression) specifies an XML element, then the *value* (that is the content of the <data> element) MUST be one or more XML elements that are valid (according to the schema of the target) for the element that the <component> element specifies.

  - If the <`component`> element (XPath expression) specifies an XML attribute, then the *value* MUST be valid (according to the schema of the target) for the attribute that the <`component`> element specifies.

- In the XML Schema profile, a requestor that specifies a `<component>` element within a <`modification`> element with "`modificationMode='delete'`" MUST NOT specify a value. The (XPath expression that is the value of the) <component> element MUST specify the set of elements or (MUST specify) the attribute that the provider should delete.

  - If the <`component`> element (XPath expression) specifies a set of XML elements, then each XML element that the <`component`> element specifies must be optional (i.e., "`minOccurs='0'`") according to the schema of the target for the object to be modified.

  - If the <`component`> element (XPath expression) specifies an XML attribute, then the specified attribute MUST be *optional* (according to the schema of the target) for the object to be modified.

4.2.3. Element <spml:schema>

If the schema is included as content of an <*spml:schema*> element, the <*spml:schema*> element MUST contain at least one <*xsd:schema*> element. If the schema is not included as content of an <*spml:schema*> element, the "ref" attribute on the <*spml:schema*> element MUST be set to the URN of the referenced schema. If the schema is included as content of an <*spml:schema*> element, a requestor should ignore any "ref" attribute on the <*spml:schema*> element.

If a provider supports only a subset of the top-level elements that are defined in the schema for a target, then the <*spml:schema*> element MUST contain at least one <*spml:supportedSchemaEntity*> element. Each <*spml:supportedSchemaEntity*> element specifies a top-level schema element that the provider supports for that target.

If the <*spml:schema*> element contains no <*spml:supportedSchemaEntity*> element, then the requestor may assume that the provider supports for that target all of the top-level elements that the schema of the target defines.

4.2.4. Element <supportedSchemaEntity>

The "entityName" attribute on the <*spml:supportedSchemaEntity*> element MUST refer to a top-level element that is defined in the schema for a target. The provider MUST support every sub-element and attribute of the referenced schema element.

4.3. Search Capability

4.3.1. Element <spmlsearch:query>

The <*spmlsearch:query*> element MAY contain an <spml:select> element. If an <*spml:select*> element is defined, it MUST be set to a valid XPath statement for the XSD schema defined by the target. The "XPath Support" section specifies general requirements for XPath support.

4.3.2. Element <spmlsearch:select>

An <*spmlsearch:select*> element MUST have a "`namespaceURI`" attribute and MUST have a "`path`" attribute.

The value of the “`namespaceURI`” attribute MUST specify the XML namespace of a query language. The value of the “`path`” attribute MUST be an expression that is valid in the query language that “`namespaceURI`” specifies. (For example, if a requestor uses XPath 2.0 as the query language for the “`path`” attribute, the value of the “`namespaceURI`” attribute MUST be "http://www.w3.org/TR/xpath20".)

The value of the “path” attribute MUST specify a *filter* that selects objects based on:

- The presence (or absence) of a specific element or attribute
- The presence (or absence) of a specific value in the content of an element
  or (the presence of absence of a specific value) in the value of an attribute

An <*spmlsearch:select*> element MAY include <*spml:namespacePrefixMap*> elements that defines the namespace prefixes that are used in the XPath path. Each “prefix” attribute on the <*spml:namespacePrefixMap*> element MUST exactly match one the namespace prefixes used in the Xpath.

Appendix A. References

[AES] National Institute of Standards and Technology (NIST), FIPS-197: Advanced Encryption Standard, **http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf**, National Institute of Standards and Technology (NIST)

[ARCHIVE-1] OASIS Provisioning Services Technical Committee, email archive, **http://www.oasis-open.org/apps/org/workgroup/provision/email/archives/index.html**, OASIS PS-TC

[DS] IETF/W3C, *W3C XML Signatures*, **http://www.w3.org/Signature/**, W3C/IETF

[DSML] OASIS Directory Services Markup Standard, *DSML V2.0 Specification*, **http://www.oasis-open.org/specs/index.php#dsmlv2**, OASIS DSML Standard

[GLOSSARY] OASIS Provisioning Services TC, *Glossary of Terms*, **http://www.oasis-open.org/apps/org/workgroup/provision/download.php**, OASIS PS-TC

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

[RFC 2246] T. Dierks and C. Allen, *The TLS Protocol*, **http://www.ietf.org/rfc/rfc2246.txt**, IETF

[SAML] OASIS Security Services TC, **http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security**, OASIS SS-TC

[SOAP] W3C XML Protocol Working Group, **http://www.w3.org/2000/xp/Group/**

[SPML-Bind] OASIS Provisioning Services TC, SPML V1.0 Protocol Bindings, **http://www.oasis-open.org/apps/org/workgroup/provision/download.php/1816/draft-pstc-bindings-03.doc**, OASIS PS-TC

[SPML-REQ] OASIS Provisioning Services Technical Committee, *Requirements*, **http://www.oasis-open.org/apps/org/workgroup/provision/download.php/2277/draft-pstc-requirements-01.doc**, OASIS PS-TC

[SPML-UC] OASIS Provisioning Services Technical Committee, *SPML V1.0 Use Cases*, **http://www.oasis-open.org/apps/org/workgroup/provision/download.php/988/drfat-spml-use-cases-05.doc**, OASIS PS-TC

[SPMLv2-Profile-DSML] OASIS Provisioning Services Technical Committee, SPMLv2 DSMLv2 Profile, OASIS PS-TC

[SPMLv2-Profile-XSD] OASIS Provisioning Services Technical Committee, SPML V2 XSD Profile, OASIS PS-TC

[SPMLv2-REQ] OASIS Provisioning Services Technical Committee, Requirements, OASIS PS-TC

[SPMLv2-ASYNC] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Async Capability of SPMLv2, OASIS PS-TC

[SPMLv2-BATCH] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Batch Capability of SPMLv2, OASIS PS-TC

[SPMLv2-BULK] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Bulk Capability of SPMLv2, OASIS PS-TC

[SPMLv2-CORE] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Core Operations of SPMLv2, OASIS PS-TC

[SPMLv2-PASS] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Password Capability of SPMLv2, OASIS PS-TC

[SPMLv2-REF] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Reference Capability of SPMLv2, OASIS PS-TC

[SPMLv2-SEARCH] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Search Capability of SPMLv2, OASIS PS-TC

[SPMLv2-SUSPEND] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Suspend Capability of SPMLv2, OASIS PS-TC

[SPMLv2-UPDATES] OASIS Provisioning Services Technical Committee, XML Schema Definitions for Updates Capability of SPMLv2, OASIS PS-TC

[SPMLv2-UC] OASIS Provisioning Services Technical Committee., SPML V2.0 Use Cases, OASIS PS-TC

[WSS] OASIS Web Services Security (WSS) TC, **http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss**, OASIS SS-TC

[X509] RFC 2459 - Internet X.509 Public Key Infrastructure Certificate and CRL Profile, **http://www.ietf.org/rfc/rfc2459.txt**

[XSD] W3C Schema WG ., *W3C XML Schema*, **http://www.w3.org/TR/xmlschema-1/** W3C

Appendix B. Acknowledgments

The following individuals were voting members of the Provisioning Services committee at the time that this version of the specification was issued:

Jeff Bohren, BMC
Robert Boucher, CA
Gary Cole, Sun Microsystems
Rami Elron, BMC
Marco Fanti, Thor Technologies
James Hu, HP
Martin Raepple, SAP
Gavenraj Sodhi, CA
Kent Spaulding, Sun Microsystems

Appendix C. Notices

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's procedures with respect to rights in OASIS specifications can be found at 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 implementors or users of this specification, can be obtained from the OASIS President.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS President.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself does not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

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 RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.