Visible Signature Profile of the OASIS Digital Signature Services Version 1.0

Committee Draft 01

22 April 2009

Specification URIs:
This Version:
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/cd01/oasis-dssx-1.0-profiles-visualsig-cd1.doc
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/cd01/oasis-dssx-1.0-profiles-visualsig-cd1.html
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/cd01/oasis-dssx-1.0-profiles-visualsig-cd1.pdf
(Authoritative)

Previous Version:
N/A

Latest Version:
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/oasis-dssx-1.0-profiles-visualsig.doc
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/oasis-dssx-1.0-profiles-visualsig.html
http://docs.oasis-open.org/dss-x/profiles/visualsig/v1.0/oasis-dssx-1.0-profiles-visualsig.pdf

Technical Committee:
OASIS Digital Signature Services eXtended (DSS-X) TC

Chair(s):
Juan Carlos Cruellas, *UPC-DAC* <cruellas@ac.upc.ede>
Stefan Drees, Individual Member, <stefan@drees.name>

Editor(s):
Ezer Farhi, *ARX,* <ezer@arx.com>

In Memory of Uri Resnitzky, ARX, an active member of OASIS DSS-X Committee.

Related work:
This specification is related to:

- OASIS Digital Signature Service Core Protocols, Elements and Bindings. Version 1.0.

Abstract:
The visible signature profile enables to embed visible signature characteristics into documents as part of a digital signature operation and also validate these characteristics as part of the verify signature operation.

Status:

This document was last revised or approved by the OASIS DSS-X 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/dss-x/.

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/dss-x/ipr.php.

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

Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

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 section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, 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 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 name "OASIS", is a trademark 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

1 Introduction 5
1.1 Terminology 5
1.2 Normative References 5
1.3 Non Normative References 5
1.4 Namespaces 6
2 Overview 7
3 Profile Features 10
3.1.1 Identifier 10
3.1.2 Scope 10
3.1.3 Relationship to Other Profiles 10
3.1.4 Element <dss:SignatureObject> 10
4 Profile of Signing Protocol 11
4.1.1 Element <dss:SignRequest> 11
4.1.2 Element <dss:SignResponse> 18
5 Profile of Verifying Protocol 19
5.1.1 Element <dss:VerifyRequest> 19
5.1.2 Element <dss:VerifyResponse> 20
6 Conformance 21

1 Introduction

1.1 Terminology

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 **[RFC 2119]**. These keywords are capitalized when used to unambiguously specify requirements over protocol 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: `<ns:Element>`, **`Attribute`**, **Datatype**, **`OtherCode`**.

1.2 Normative References

[Core-XSD]	OASIS Standard, *DSS Schema* February 2007.
[VisSig-XSD]	OASIS Committee Draft, *Visible Signatures profile Schema,* April 2009.
[DSSCore]	OASIS Standard, *Digital Signature Service Core Protocols and Elements*. February 2007.
[AdES-DSS]	OASIS Advanced Electronic Signature Profiles of the OASIS Digital Signature Service Version 1.0., April 2007.
[DSS-MultVerRep]	OASIS Profile for comprehensive multi-signature verification reports for OASIS Digital Signature Services Version 1.0, TBD
[XML-ns]	T. Bray, D. Hollander, A. Layman. *Namespaces in XML.* http://www.w3.org/TR/1999/REC-xml-names-19990114, W3C Recommendation, January 1999.
[XMLSig]	D. Eastlake et al. *XML-Signature Syntax and Processing.* http://www.w3.org/TR/1999/REC-xml-names-19990114, W3C Recommendation, February 2002.
[ISO-8601]	ISO 8601:2004, Data elements and interchange formats – information interchange – representation of dates and times
[W3CDT]	M. Wolf, C.Wicksteed. W3C Date and Time Formats – September 2007 - http://www.w3.org/TR/NOTE-datetime
[ISO-32000]	ISO 32000-1*, Document management – Portable document format – Part 1: PDF 1.7*
[ODF]	OASIS Standard, *Open Document Format for Office Applications (Open Document) v1.1*, February 2007
[ooxml]	Ecma-376, Open Office XML File Format - 1st edition - December 2006, 2nd edition – December 2008

1.3 Non Normative References

[AustriaSig] An Official implementation of Visible Signature in Austria - http://www.oasis-open.org/committees/document.php?document_id=29553

[Adobe] Implementation of Visible Signatures in Adobe Acrobat and Adobe Reader – http://www.adobe.com

[MSOffice2007] Implementation of Signature Line in Office 2007 – http://www.microsoft.com

1.4 Namespaces

The structures described in this specification are contained in the schema file **[VisSig-XSD].** All schema listings in the current document are excerpts from the schema file. In the case of a disagreement between the schema file and this document, the schema file takes precedence.

This schema is associated with the following XML namespace:

`urn:oasis:names:tc:dssx:1.0:profiles:VisibleSignatures:schema#`

Conventional XML namespace prefixes are used in this document:

- The prefix **`dss:`** (or no prefix) stands for the DSS core namespace **[Core-XSD]**.
- The prefix **`ds:`** stands for the W3C XML Signature namespace **[XMLSig]**.

Applications MAY use different namespace prefixes, and MAY use whatever namespace defaulting/scoping conventions they desire, as long as they are compliant with the Namespaces in XML specification **[XML-ns]**.

2 Overview

For many processes that incorporate a digital signature operation or a verification of a digital signature, there is a need to view displayed information that is related to the binary digital signature.
This visible display of information can include displaying the signer's identity, the time when the digital signature operation was performed, and additional information as well.
The visible information comes in addition to the actual digital signature data, and is aimed at providing end users with information that closely relates to the digital signature act. By no means does this visible information replace the important element of the digital signature.

Visible signatures are strongly associated with documents. The visible signatures will normally be displayed in the document in addition to other displayed information such as text and images.
There are several documents types and applications that already support digital signatures in conjunction with visible signatures:

- Adobe Reader/Adobe Acrobat using digital signatures inside PDF documents. For more information refer to **[ISO-32000]** and **[Adobe]**.
- Microsoft Office 2007 using signatures Line inside OOXML documents. For more information refer to **[ooxml]** and **[MSOffice2007].**
- Other solutions that enable the use of visible signatures as well as digital signatures in other documents types such as TIFF, Office XP/2003, and other document types.
  As an example of such implementation refer to **[AustriaSig]**.

Other types of standards or applications such as Open Document Format [ODF] do not support visible signatures yet, but already support non-visible digital signatures.

The target of the Visible Signatures Profile is to define mechanisms that will enable clients that interact with a digital signature service, based on DSS core, to incorporate visible signatures into documents as part of a digital signature operation.

The signature operation can be applicable for any type of document and can be displayed with any tool that displays the document's content.

The signature verification service may incorporate some visible indications to signature field as part of the signature verification service.

Digital Signature Operation

There are several types of usage scenarios that involve visible signatures as part of a digital signature operation:

- **Submission of a document to be signed**
  In this scenario, an unsigned document is submitted to be signed by the digital signature service. As part of the submission, the client needs to provide some information that will be used by the signature service in order to build a visible content that relates to the digital signature. The visible signature may also include some information extracted of the signer's certificate. This information will be extracted by the digital signature service during signature operation.
  Depending on the type of document, the visible content may be included as part of the signed content. This means that any modification to the visible signature will invalidate the digital signature.
  In this type of scenario there is a single digital signature in the document.
- **Digital Signature operations as part of a workflow process**
  In this scenario, the document will already contain visible signature placeholders (named "signature fields") that are uniquely identified in the document. As part of the digital signature operation, the client will need to specify which signature field should be signed.
  The signature field may already contain metadata such as the display configuration. In such documents several signature fields may be included in the document.

- **Simple Workflow Operation**
  This is a simple case of the above general workflow scenario. In this case only a single field will be signed as part of the digital signature service.
- **The General Signature Scenario**
  In this scenario several signature fields can be signed. In some of the fields a display configuration can be passed as well.

The following specification is aimed to address all above scenarios. The resulting visible signature and digital signature are very similar in all above scenarios.

Visible Signature Content

The structure of the visible signature is made of components (normally strings and images) that are displayed in a certain location inside the visible signature.
The following is information that may be included as part of the visible signature. The parameters are not listed in the order of their importance.

- **Signer Information** - Information of the signer that performs the digital signature operation. The information will be extracted from the signer's certificate.
  Besides the signer's *Common Name*, additional information can be displayed from the signer's certificate such as: *serial number, role, organization*, or any other specific information that is located inside the signer's certificate.
  Several elements that are retrieved from the signer's certificate can be displayed in the visible signature.
- **CA Information** - Information of the Certificate Authority that produced the certificate for the signer. The information will be extracted from the signer's certificate.
  In addition to the CA's *Common Name*, additional information can also be displayed from the signer's certificate or CA Certificate, such as: *CA's country, organization.*
  Several elements that are retrieved from the signer's certificate can be displayed in the visible signature.
- **Signature Time** - The date and time of the digital signature operation. The format of the date and time to be displayed will be provided according to **[ISO-8601]** and **[W3CDT].**
- **Signer's Related Image** - End users and organizations appreciate the ability to view images such as the end user's hand-written signature, as part of the visible signature. This will normally be a scanned or a captured image of the user's hand-written signature.
  In some cases, depending on the context of the signature, this field can also be used to contain an organizational logo.
- **Additional Application Info** - In certain cases, additional information should be displayed in the visible signature. For example, some applications require adding the *Reason* for the digital signature operation.
- **Digital Signature Value** - This value is a base64 encoding or other scanable representation of the binary digital signature. This value can be scanned out of the printed document and used for digital signature verification purposes. In relation to other visually displayed components, this field must not be included as part of the content to be signed.

The following information may be sent to the digital signature service as part of the digital signature information, so that the service is able to embed the visible signature into the document.

- **Document Type** - This value defines the format of the provided document so that the digital signature service can embed both a visible signature and a digital signature into the document, according to the document format.
- **Position** - The visible signature is visually located inside the document. Therefore, the position of the visible signature needs to be specified. This parameter is essential for the Document Submission Scenario. The position may be specified according to page number and location in a page, or any other metric that determines the exact location and size of the visible signature inside the document.

- **Signature Field Identification** - This identification can be used in most of the above scenarios. Through this approach, the signature service can locate the signature field that will contain both the visible signature and digital signature.
- **Visible Signature Configuration and Policy** - This parameter is optional and may direct the service how to configure the visible signature. The configuration will include which elements will be displayed inside the visible signature and their position inside the visible signature will be. For a given specified configuration, all elements must be incorporated into the visible signature.

Some of the information described in the Visible Content section should be sent to the digital signature service to be embedded into the visible signature.

Digital Signature Verification Operation

The verification operation will reply information that is bounded to the signature field. Also, it will be possible in some documents type to add visible indication to the replied document.
The visible indication will include a general verification remark as well as some additional content such as the date and time of the verification operation.

Note:

The visible signature profile does not include signature field management operations such as signature field creation operation or clearing a signature field operation. These operations might be defined in another profile.

3 Profile Features

3.1.1 Identifier

urn:oasis:names:tc:dssx:1.0:profiles:VisibleSignatures

3.1.2 Scope

This document profiles the DSS signing and verifying protocols defined in **[DSSCore].**

3.1.3 Relationship to Other Profiles

The profile in this document is based on the **[DSSCore].** The profile in this document may be implemented.

This profile provides means for the explicit management of signature policies with **[DSSCore]** and other existing profiles like **[AdES-DSS],** and as such, it may be used in conjunction with these specifications.

3.1.4 Element <dss:SignatureObject>

This profile supports requests for generation of a digital signature in conjunction with a visible signature field that is embedded inside a given document.
The profile also supports replied information as well as visible indication as part of a signature verification operation.

4 Profile of Signing Protocol

This profile is based directly on the **[DSSCore].**

This profile is intended to be combined with other profiles freely.

4.1.1 Element <dss:SignRequest>

This clause profiles the <dss:SignRequest> element.

4.1.1.1 Element <dss:InputDocuments>

The document type and the document content will be provided as part of <dss:InputDocument> element where the <Base64Data> element contains the document content encoded in base64 format and the MimeType attribute defines the Document Type (for example application/pdf).

It is also possible to send the document using an <AttachmentReference>. In this case, the MimeType is taken from the attachment reference.

The Mime Type is a mandatory attribute.

If several documents are sent to the signature service, each document should be identified with an xs:ID attribute. This ID will be used for binding a certain visible signature configuration to a specific document.

4.1.1.2 Element <dss:OptionalInputs>

This profile does not impose any restrictions on any optional input specified in the **[DSSCore]** or other profiles.

This profile defines a new Optional Input as indicated below.

4.1.1.2.1 New Optional Inputs

4.1.1.2.1.1 Optional Input <VisibleSignatureConfiguration>

This optional input includes several items that together provide all of the required information for performing a signature operation that includes a visible signature.

This input covers all the above scenarios. The service will restrict input parameters according to the specified visible signature policy (or scenario).

This optional input includes the following items:

FieldName

The parameter enables the digital signature service to perform a signature operation on a specific field in the document. This parameter is more relevant to the workflow scenarios.
This field can be omitted only when submitting a document to be signed.

VisibleSignaturePolicy

This parameter indicates the type of scenario that is used when performing a visible signature operation.

DocumentRestrictionLevel

In some types of documents, the digital signature operation will make the document more restricted to modifications after the document was signed. The content of this element will be numeric and will be implemented according to the document type.

In the case of PDF, there is a special digital signature operation called *Certify*. The Certify operation performs a digital signature operation and also makes the document restricted to changing document's

content beside some certain content modifications such as entering comments or entering data inside form's fields. For more information refer to **[ISO-32000]**, section 12.8.2.2 – DocMDP. The description of the P parameter contains the permissible restriction levels.

VisibleSignaturePosition

Information that relates to the location of the visible signature in the given document. This parameter is more relevant to the document submission scenario.

The position will be defined as an abstract type since the document type defines how to position a visible signature into the document. In the general case, the position includes a page number and (x,y) coordinates inside the given page based on the document's displayable unit definition.

VisibleSignatureItemsConfiguration

Information that will enable the signature service to incorporate visible items into the document.

Other

Additional information related to a visible signature

The schema for this element is listed below:

```
<xs:element name="VisibleSignatureConfiguration"
type="VisibleSignatureConfigurationType" />

<xs:complexType name="VisibleSignatureConfigurationType">
   <xs:sequence>
         <xs:element ref="VisibleSignaturePolicy"/>
         <xs:element name="FieldName" type="xs:string" use="optional"/>
         <xs:element name="DocumentRestrictionLevel" type="xs:integer"
use="optional"/>
         <xs:element ref="VisibleSignaturePosition" use="optional"/>
         <xs:element ref="VisibleSignatureItemsConfiguration" use="optional"/>
         <xs:element name="other" type="dss:AnyType"/>
   </xs:sequence>
</xs:complexType>
```

4.1.1.2.1.2 Optional Input <VisibleSignaturePolicy>

The type of above scenario that is used will be indicated.

In the case that a certain scenario is defined, some restrictions will be checked. The restriction will make sure that adequate parameters are passed in the request.

The restrictions are specified in the sections below.

```
<xs:element name="VisibleSignaturePolicy" type="VisibleSignaturePolicyType"/>

<xs:simpleType name="VisibleSignaturePolicyType">
        <xs:restriction base="xs:string">
          <xs:enumeration value="DocumentSubmissionPolicy" />
          <xs:enumeration value="SimpleWorkflowPolicy" />
          <xs:enumeration value="WorkflowPolicy" />
          <xs:enumeration value="GeneralPolicy" />
        </s:restriction>
</s:simpleType>
```

4.1.1.2.1.2.1 Optional Input <FieldName>

This optional input will define the identity of a signature field to be signed. This parameter will be sent when it is required to incorporate a visible signature into the given field.

In the cases of the *General Scenario* as well as the *Document submission scenario*, it is possible to pass a name of a non existing field. This will indicate the service to generate a new signature field with the given name. In these scenarios, if the FieldName is not provided, then a new signature field will be added to the document and signed as part of the digital signature operation.

In the workflow scenarios, if the given field does not exist in the given document, the signature operation will fail where the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultminor:FieldNotExist.*

4.1.1.2.1.2.2 Optional Input <VisibleSignaturePosition>

This optional input will define the location of the newly generated visible signature in the document. This parameter will be provided in the case of a submitted document or the general signature scenario. Since a position of a visible signature in a document is very dependant on a way the document is specified, an abstract type is defined. Also, two simple position types are defined that are based on a page number, horizontal and vertical coordinates in the given page, and the dimensions (width and height) of the boundary of the visible signature field. The given coordinates as well as width and height are based on definitions related to the document type. The first type is based on pixel based documents, while the other is a more general one and is defined similarly to the defined in **[ODF].**

In all scenarios, if neither an existing *FIeldName* nor a valid *VisibleSignaturePosition* is provided, the signature operation will fail where the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:PositionIsRequired*.
In all scenarios, if an existing *FIeldName* is provided and also a *VisibleSignaturePosition* is provided, the signature operation will fail where the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:PositionIsAmbiguity*.

The schema for this element is listed below:

```
<xs:element name="VisibleSignaturePosition"
type=”VisibleSignaturePositionType”>

<xs:complexType name=”VisibleSignaturePositionType” abstract=”true”/>

<xs:complexType name=”PixelVisibleSignaturePositionType”>
   <xs:complexContent>
     <xs:extension base=”VisibleSignaturePositionType”>
        <xs:sequence>
          <xs:element name=”PageNumber” type=”xs:integer”/>
          <xs:element name=”x” type=”xs:integer”/>
          <xs:element name=”y” type=”xs:integer”/>
          <xs:element name=”Width” type=”xs:integer” use=”optional”/>
          <xs:element name=”Height” type=”xs:integer” use=”optional”/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:complexType name=”GeneralVisibleSignaturePositionType”>
   <xs:complexContent>
     <xs:extension base=”VisibleSignaturePositionType”>
```

```
        <xs:sequence>
          <xs:element name="PageNumber" type="PageNumberType"/>
          <xs:element name="x" type="MeasureType"/>
          <xs:element name="y" type="MeasureType"/>
          <xs:element name="Width" type="MeasureType" use="optional"/>
          <xs:element name="Height" type="MeasureType" use="optional"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
```

4.1.1.2.1.2.3 Optional Input <VisibleSignatureItemsConfiguration>

This optional input will define the design of the visible signature. This input will direct the digital signature service how to embed the visible content of visible signature in the document. Since this parameter is optional, the signature service can have its own definitions of how to embed the content of the visible signature in the document.

The configuration is based on native sub-elements or items, each can be based on a string or an image. Each of the items will be located in a certain position in the visible signature. In addition, some general design parameters can be provided.

There are cases where the items' values are provided by the signature service (for example: signature time). In other cases, the request will include the items' values (for example, a reason for the digital signature operation).

In the case that a value is included in the request when it is not supposed to, an error will be replied as follows: the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:ValueNotRequired*.

In the case that a required value should be passed as part of the request and the value is missing in the request, the following error will be replied: the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:ValueNotExist*.

In the case that a required value should be passed as part of the request and the value has the wrong type in the request, the following error will be replied: : the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:ValueWrongType*.

Item Identification

Follows the list of items that can be part of a visible signature:

- Signer's info – All of the following values will be taken out of the signer's certificate:
  - "**Subject:CommonName**" – The Common Name of the signer
  - "**Subject:Title**" – The title of the signer
  - "**Subject:Org**" – The organization of the signer
- "**CertSerialNum**" – The serial number of the user certificate
- CA's info – All following values will be taken out of the issuer fields in the signer's certificate:
  - "**Issuer:CommonName**" – The Common Name of the CA
  - "**Issuer:Country**" – The country of the CA
  - "**Issuer:Org**" – The organization of the CA

- "**SignatureTime**" – The time of the digital signature operation. This value is determined by the signature service.
- "**SignerImage**" – An image that will be incorporated into the visible signature. The image may contain a capture of the user's hand-written signature or a company logo. As an example, the provided value may be a base64 encoding of a JPEG-encoded image.
  Alternatively, a URI of an image can be provided so that the signature service can locate the value of the image and incorporate it into the visible signature.
- "**SignatureReason**" – The reason for the digital signature operation. This sub-element is used in PDF documents.
- "**SignerContactInfo**" – Textual information for contact information of the signer.
- "**SignatureProductionPlace**" – Textual information for the location where the signature was produced.
- "**CustomText**" – Textual information that can be added to the Visible Signature.
- "**SignatureValue**" – A signature value will be encoded into the visible signature. The computed digital signature of the document will be incorporated into the visible signature either by a scanable image or a base64 output.

  In cases such as PDF documents, such value cannot be displayed since the digital signature itself is calculated upon the visible signature as well. If such an element is requested to be included in the visible signature, the signature operation will fail.

Position of an item in the visible signature

This abstract type enables the signature service to design the location of the item in the visible signature. There are two general ways to position the item inside the visible signature either by providing a document related coordinates or providing percentage values that enables the service to position the item in relation to the whole visible signature rectangle.

Additional information for an item

When the request includes an item, both type and value of the item may be provided. The following types are supported:

- **String** – the provided value is of type string.
- **Image** – the provided value is an image encoded in base64 format.
- **DateTime** – the item represents a date and time. In this case a DateTime format string may be provided.
  As part of the string format it should be defined whether to display a GMT offset as well.
  The format of the string will be according to **[ISO-8601]** or **[W3CDateTime].**
- **ItemValueURI** – the provided value is a URI. This value can be used to get a required image to be included into the visible signature.

In the case that the item is a string, the request can include a font name and a font size so that the item can be visualized using a specific font. If the required font and its size are not available, an error will be replied to the client as follows: the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:FontNotExist*.

General Parameters

The following general design parameters can be passed as part of the sign request:

Display Caption
If this parameter is true, all string information will be preceded with a caption that is relevant to the item.

Orientation

This parameter will direct the service to design the whole content of the visible signature in a certain orientation. There are 4 orientation values supported: 0, 90, 180 and 270, while the default value is 0 indicating that the visible signature is aligned with the text in the given page. The orientation parameter is calculated counterclockwise.

The schema for this element is listed below:

```
<xs:element name="VisibleSignatureItemsConfiguration
type=”VisibleSignatureItemsConfigurationType” />

<xs:complexType name=VisibleSignatureItemsConfigurationType”>
   <xs:sequence >
         <xs:sequence minOccures=”0” maxOccures=”unbounded”>
               <xs:element ref=”VisibleSignatureItem”/>
         </xs:sequnece>
         <xs:element name=”IncludeCaption” type=”xs:boolean” use=”optional” />
         <xs:element name=”Orientation” type=”OrientationType” use=”optional”
/>
   </xs:sequence>
</xs:complexType>

<xs:element name="VisibleSignatureItem" type=”VisibleSignatureItemType” />

<xs:complexType name=”VisibleSignatureItemType”>
   <xs:sequence>
         <xs:element name=”ItemName” type=”ItemNameEnum”/>
         <xs:element ref=”ItemPosition” use=”optional” />
         <xs:element ref=”ItemValue” use=”optional” />
   </xs:sequence>
</xs:complexType>

<xs:simpleType name="ItemNameEnum">
        <xs:restriction base="xs:string">
          <xs:enumeration value="Subject:CommonName" />
          <xs:enumeration value="Subject:Title" />
          <xs:enumeration value="Subject:Organization" />
          <xs:enumeration value="CertSerialNum" />
          <xs:enumeration value="Issuer:CommonName" />
          <xs:enumeration value="Issuer:Country" />
          <xs:enumeration value="Issuer:Organization" />
          <xs:enumeration value="SignatureTime" />
          <xs:enumeration value="SignerImage" />
          <xs:enumeration value="SignatureReason" />
          <xs:enumeration value="SignerContactInfo" />
          <xs:enumeration value="SignatureProductionPlace" />
          <xs:enumeration value="CustomText" />
          <xs:enumeration value="SignatureValue" />
        </s:restriction>
</s:simpleType>

<xs:element name=”ItemPosition” type=”ItemPositionType” />

<xs:complexType name=”ItemPositionType” abstract=”true”/>

<xs:complexType name=”PixelItemPositionType”>
   <xs:complexContent>
     <xs:extension base=”ItemPositionType”>
        <xs:sequence>
          <xs:element name=”x” type=”xs:integer”/>
          <xs:element name=”y” type=”xs:integer”/>
```

```
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:complexType name="GeneralItemPositionType">
   <xs:complexContent>
     <xs:extension base="ItemPositionType">
        <xs:sequence>
          <xs:element name="x" type="MeasureType"/>
          <xs:element name="y" type="MeasureType"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:complexType name="PercentItemPositionType">
   <xs:complexContent>
     <xs:extension base="ItemPositionType">
        <xs:sequence>
          <xs:element name="x-percent" type="PercentType"/>
          <xs:element name="y-percent" type="PercentType"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:element name="ItemValue" type="ItemValueType" />

<xs:complexType name="ItemValueType" abstract="true"/>

<xs:complexType name="ItemValueStringType">
   <xs:complexContent>
     <xs:extension base="ItemValueType">
        <xs:sequence>
          <xs:element name="ItemValue" type="xs:string" use="optional"/>
          <xs:element name="ItemFont" type="xs:string" use="optional"/>
          <xs:element name="ItemFontSize" type="xs:integer" use="optional"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:complexType name="ItemValueImageType">
   <xs:complexContent>
     <xs:extension base="ItemValueType">
        <xs:sequence>
          <xs:element ref="dss:Base64Data"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<xs:complexType name="ItemValueDateType">
   <xs:complexContent>
     <xs:extension base=" ItemValueStringType">
        <xs:sequence>
          <xs:element name="DateTimeFormat" type="xs:string" use="optional"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
```

```
</xs:complexType>

<xs:complexType name="ItemValueURIType">
   <xs:complexContent>
     <xs:extension base="ItemValueType">
        <xs:sequence>
          <xs:element name="ItemValue" type="xs:anyURI"/>
        </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>
```

4.1.2 Element <dss:SignResponse>

This profile does not impose any restrictions on any optional input specified in the **[DSSCore]** or other profiles.

4.1.2.1 Element <dss:DocumentWithSignature>

The document type and the updated document content will be returned to the client as part of the dss:DocumentWithSignature element where the <Base64Data> element contains the document content encoded in base64 format and the MimeType attribute defines the Document Type (for example application/pdf).

Also it is possible to send the document using an <AttachmentReference>, in this case the MimeType is taken from the attachment reference.

5 Profile of Verifying Protocol

This profile is based directly on the **[DSSCore].**

This profile is intended to be combined with other profiles freely.

This profile can be combined with the multi-signature verification report profile **[DSS-MultVerRep]** to get a verification report to every signature field inside the given document. For each signed field, the report can include the verification status of the signed field.

5.1.1 Element <dss:VerifyRequest>

The input document may contain signed and unsigned fields within the given document. Each signed field may also have a visible signature.

If a general verification request is sent to the verification service, the verification service should reply with the signature status of all signature fields, including unsigned fields.

If a verification request is sent for a specific signature field, then the service will respond with a verification status for the requested field.

5.1.1.1 Element <dss:InputDocuments>

The document type and the document content will be provided as part of the dss:InputDocument element where the <Base64Data> element contains the document content encoded in base64 format and the MimeType attribute defines the Document Type (for example application/pdf).

Also it is possible to send the document using an <AttachmentReference>, in this case the MimeType is taken from the attachment reference.

The Mime Type is a mandatory attribute.

5.1.1.2 Element <dss:OptionalInputs>

This profile does not impose restrictions on any optional input specified in the **[DSSCore]** or other profiles.

This profile defines a new Optional Input as indicated below.

5.1.1.2.1 New Optional Inputs

5.1.1.2.1.1 Optional Input <FieldName>

This optional input will define the identity of a signature field to be verified. This parameter will be sent in a scenario where it is required to validate only a certain field. In this case the response from the VerifyRequest will include verification status related only to this field.
If the given field does not exist in the given document, the signature operation will fail where the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:FieldNotExist* if no field is specified, then verification statuses for all the signature fields in the document will be replied.

The schema for this element is listed below:

```
<xs:element name="FieldName" type="xs:string"/>
```

5.1.1.2.1.2 Optional Input <VisibileIndicationFormat>

This optional input will define whether the verification service should embed into the visible signature an indication that specifies the verification status of the digital signature and some other information as part of the verification operation. The Visible indication will include the following items:

- **Verification Mark** – a √, X, ? symbols that will indicate a success in the verification procedure, Failure or whether the verification service cannot perform a full signature validation procedure.
- **Verification time** – the time of the signature verification. The service will define the format of the date and time content.
- **Verification Scope Indication** – the scope of verification that was performed (for example, only signature validation, CRL/OCSP check, …)

```
<xs:element name="VisibleIndicationFormat" type="VisibleIndicationFormatType"
use=”optional”/>

<xs:complexType name=VisibleIndicationFormatType”>
   <xs:sequence>
      <xs:element name=”VerificationMark” type=”xs:boolean” use=”optional”/>
      <xs:element name=”VerificationTime” type=”xs:boolean” use=”optional”/>
      <xs:element name=”VerificationScope” type=”xs:boolean” use=”optional”/>
   </xs:choice>
</xs:complexType>
```

5.1.2 Element <dss:VerifyResponse>

This clause profiles the <dss:VerifyRequest> element.

5.1.2.1 New Optional Outputs

5.1.2.1.1 Optional Output <FieldName>

This optional output will define the identity of a signature field that is verified. This parameter will be replied for every signature field that is validated in the document as part of the signature validation service.

The schema for this element is listed below:

```
<xs:element name="FieldName" type="xs:string"/>
```

5.1.2.2 Element <dss:DocumentWithSignature>

This parameter will be returned only if the VisibleIndicationFormat is included in the request and the service is capable of embedding verification information into the visible signature.
The document type and the updated document content will be returned to the client as part of the dss:DocumentWithSignature element where the <Base64Data> element contains the document content encoded in base64 format and the MimeType attribute defines the Document Type (for example application/pdf).

Also it is possible to send the document using an <AttachmentReference>, in this case the MimeType is taken from the attachment reference.

6 Conformance

The following conformance is related to typical usage scenario of the DSS signature service. These scenarios are described in the Overview section and are formalized by sending the *VisibleSignaturePolicy* attribute as part of the Optional Inputs in the SignRequest.

For each of these usage scenarios all components of the request will be analyzed by the signature service to make sure that input parameters are aligned with the described usage scenario. If the parameters are not adequate, the following error will be replied: the <ResultMajor> will be replied with the value of *urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError* and the <ResultMinor> with the value of *urn:oasis:names:tc:dss:1.0:resultminor:ConformanceError*
Some of the restrictions are also described in above sections.

Simple document submission – A single document is submitted to be signed and there is no field name indication in the request.

The request should also include signature position information.
If the documents includes a signature field embedded inside the document an error is replied to the user.

Simple workflow signature operation – A single document is sent to the digital signature service and also a single signature field ID is specified. No signature position as well as signature configuration is passed to the server.

General workflow operation – The sent documents may include several signature fields. No visible signature position as well as configuration is sent as part of the request.

General request – This is the most flexible policy. Any scenario that involves incorporating a visible signature as part of a digital signature operation can use this general policy.