# TCG Infrastructure Working Group
# Simple Object Schema Specification

**Specification Version 1.0**
**Revision 1.0**
**17 November 2006**
**FINAL**

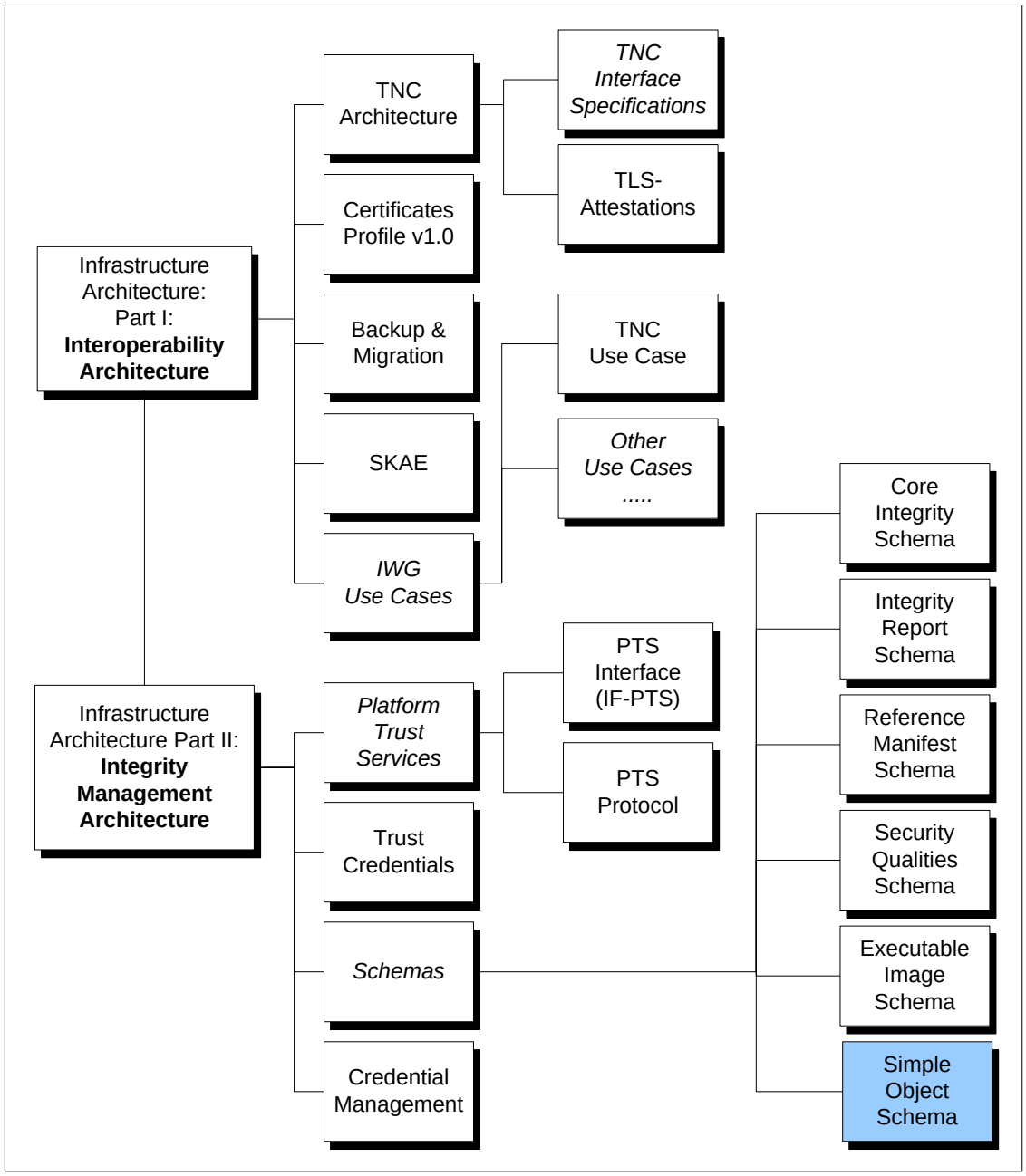
**Contacts:**

GregK@wavesys.com **(Editor)**
Ned.Smith@intel.com **(Co-Chair)**
thardjono@signacert.com **(Co-Chair)**

TCG

**Public**

# IWG Document Roadmap



# Acknowledgement

The TCG wishes to thank all those who contributed to this specification. This document builds on considerable work done in the various working groups in the TCG.

Special thanks to the members of the IWG contributing to this document:

| Name | Company |
|---|---|
| Mark Redman | Freescale Semiconductor |
| Malcolm Duncan | CESG |
| Diana Arroyo | IBM |
| Lee Terrell | IBM |
| Markus Gueller | Infineon |
| Ned Smith (IWG Co-Chair) | Intel Corporation |
| Thomas Hardjono (IWG Co-Chair) | SignaCert |
| Wyllys Ingersoll | Sun Microsystems |
| Jeff Nisewanger | Sun Microsystems |
| Paul Sangster | Symantec |
| Greg Kazmierczak (Editor) | Wave Systems |
| Len Veil | Wave Systems |

## Table of Contents

# 1 Scope and Audience

This specification is integral to the TCG Infrastructure Working Group's (IWG) reference architecture, and is directly related to the TCG's Integrity Management Model. Specifically, the simple object XML schema defines the structure with which integrity measurements are included within integrity reports.

Architects, designers, developers, and technologists interested in the development, deployment, and interoperation of trusted systems will find this document necessary in providing a specific mechanism for communicating integrity information.

# 2 Introduction

The purpose of this document is to provide a detailed description of the TCG Infrastructure Working Group's simple object XML schema, hereafter referred to as the *simple object schema*. The simple object schema is derived from the Core Integrity Metadata XML Schema [1].

The simple object schema allows instantiation of interoperable integrity report and snapshot integrity measurements. This schema is intended for use as a child of the Integrity Report Schema [9] and the Reference Manifest Schema [5] and allows implementers to populate integrity measurements. One use of integrity reports and snapshot structures is in the Trusted Network Connect (TNC) use models [7] whereby a Platform Trust Service (PTS) [8] creates integrity reports containing snapshots to be sent by IMCs to their corresponding IMVs for verification of acceptable platform state prior to network access. Another use is by a Reference Manifest Publisher who populates Reference Manifest records with file measurements.

## 2.1 Normative Specification Content

The contents of this document should be considered to be NORMATIVE except for the XML schemas and associated structural diagrams. For XML schemas, the XML in this document is generated from the XSD files. While it is the intention of the authors to keep these representations consistent, the XSD files are considered NORMATIVE for all XML and any XML representations in this document are INFORMATIVE.

## 2.2 Schema Version

The report schema's version number is defined using the `version` attribute of the schema's root-level `schema` element:

```
version="version_number"
```

This document refers to version 1.0 of the simple object schema.

## 2.3 Schema Namespace

The simple object schema's namespace is defined using the `targetNamespace` attribute of the schema's root-level `schema` element:

```
targetNamespace="namespace"
```

The schema's namespace reflects the schema version, and is currently defined as follows:

```
http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#
```

## 2.4 Dependent Schema Definitions

### 2.4.1 W3C XML Schema Syntax

The simple object schema relies upon data structures defined by the World Wide Web Consortium's (W3C) XML-Schema syntax. Consequently, the simple object schema imports the W3C's XML schema with the following namespace:

```
http://www.w3.org/2001/XMLSchema
```

The report schema associates the abovementioned schema with the "xs" namespace prefix.

### 2.4.2 W3C XML-Signature Syntax

The simple object schema relies upon data structures defined by the World Wide Web Consortium's (W3C) XML-Signature digital signature syntax. Consequently, the simple object schema imports the W3C's digital signature XML schema with the following namespace:

```
http://www.w3.org/2000/09/xmldsig#
```

The report schema associates the abovementioned schema with the "ds" namespace prefix.

### 2.4.3 TCG Core Integrity Schema Syntax

The report schema relies upon data structures defined by the TCG Core Integrity Schema syntax, [1]. Consequently, the report schema imports the TCG Core Integrity Schema with the following namespace:

```
http://www.trustedcomputinggroup.org/XML/SCHEMA/Core_Integrity_V1_0_1#
```

The report schema associates the abovementioned schema with the “core” namespace prefix.

### 2.4.4 Schema Diagram Conventions

The schema diagrams in this specification contain attributes and elements that are either mandatory or optional to populate. Those that are mandatory to populate are depicted by solid lines surrounding the attributes and elements. Those that are optional to populate are depicted by dashed lines surrounding the attributes and elements.

### 2.4.5 Keywords

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 RFC 2119 [11]. This specification does not distinguish blocks of informative comments and normative requirements. Therefore, for the sake of clarity, note that lower case instances of must, should, etc. do not indicate normative requirements.

# 3 Simple Object Schema

schema location: **http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0.xsd**
attribute form default: **Unqualified**
element form default: **Qualified**
targetNamespace: **http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#**

## 3.1 COMPLEX TYPES

The following complex types are specified in this document:

Complex types
SimpleObjectType
ValuesType

Elements which are derived from these complex types are defined in section 3.2.

### 3.1.1 complexType SimpleObjectType

#### 3.1.1.1 Description

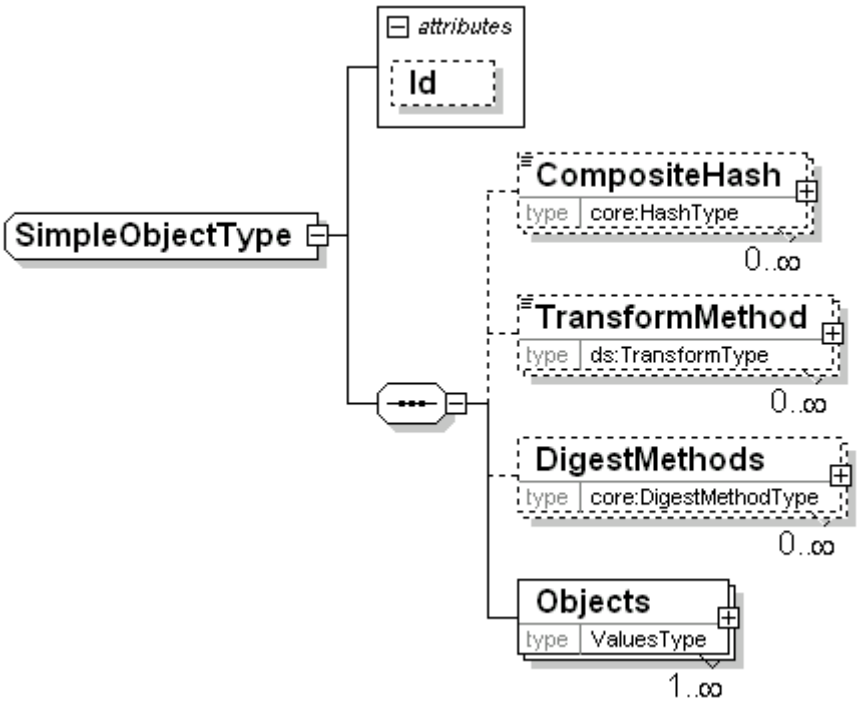
The SimpleObjectType complex type represents component measurement values and the base information necessary to interpret those measurements.

Elements of SimpleObjectType include:

- CompositeHash – If multiple integrity measurement values are included, the CompositeHash element provides a means to combine those measurements into a single hash. The schema allows for multiple CompositeHash elements; one CompositeHash element should be used for each Digest Method (if more than one Digest Method is used).
- TransformMethod – This element identifies an algorithm applied to the measured data prior to a hash computation operation.
- DigestMethods – This element identifies the digest method used to compute hash values.
- Objects – The elements containing the integrity measurements. If raw data (i.e. not digests) are populated in the Object elements, then external software MUST perform any transforms required prior to input of the data to PTS and either a TransformMethod element must be populated either in SimpleObjectType or in the parent Reference Manifest [5] or Snapshot [9].

#### 3.1.1.2 Diagram

diagram




namespace http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#

children **CompositeHash** **TransformMethod** **DigestMethods** **Objects**

used by element **SimpleObject**

attributes

| Name | Type | Use | Default | Fixed |
|---|---|---|---|---|
| Id | xs:ID | Optional | | |

#### 3.1.1.3 Attribute Detail

| Attribute | Description |
|---|---|
| ID | Document unique record instance identifier. ID is used in other parts of the document to reference instances of Simple Objects. This attribute SHOULD be populated if more than one SimpleObject element is instantiated in a Reference Manifest [5] or Snapshot [9] Values element. |

#### 3.1.1.4 XML

source

```
<xs:complexType name="SimpleObjectType">
  <xs:sequence>
    <xs:element name="CompositeHash" type="core:HashType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="TransformMethod" type="ds:TransformType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="DigestMethods" type="core:DigestMethodType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Objects" type="ValuesType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:ID"/>
</xs:complexType>
```

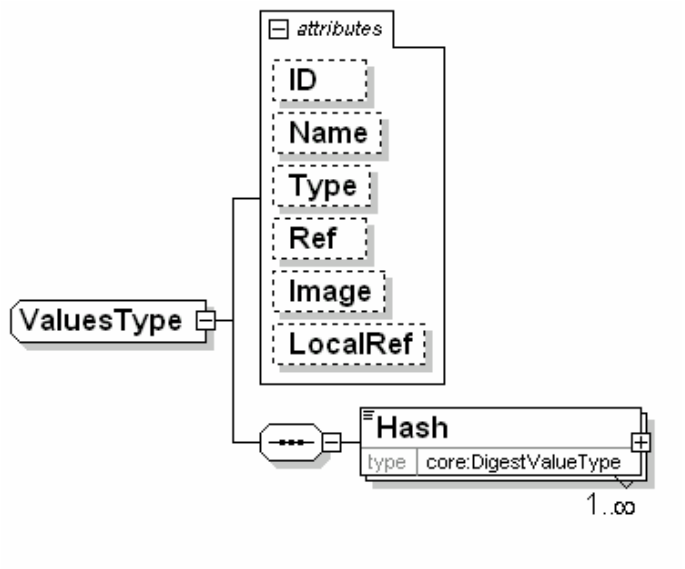
## 3.1.2 complexType ValuesType

#### 3.1.2.1 Description

The ValuesType complex type represents component integrity measurements. ValuesType complex type includes the Hash element which contains the actual digest values.

### 3.1.2.2 Diagram

diagram




namespace http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#

children Hash

used by element SimpleObjectType/Objects

attributes

| Name | Type | Use | Default | Fixed |
|---|---|---|---|---|
| ID | xs:ID | Optional | | |
| Name | xs:normalizedString | Optional | | |
| Type | xs:anySimpleType | Optional | | |
| Ref | xs:anyURI | Optional | | |
| Image | xs:base64Binary | Optional | | |
| LocalRef | xs:IDREF | Optional | | |

### 3.1.2.3 Attributed Detail

| Attribute | Description |
|---|---|
| ID | Document unique record instance identifier. ID is used in other parts of the XML document to reference instances of integrity values. |
| Name | Descriptive name for the included set of integrity values. If the Simple Object is instantiated in a Reference Manifest, then this attribute MAY be populated with a relative pathname for the object on a platform. |
| Type | Type descriptor for the included set of integrity values. |
| Ref | URI reference to the raw data corresponding to the digest value. This MUST be populated in a Reference Manifest if Image is not populated. |
| Image | The actual raw data corresponding to the digest value. This MUST be populated in the Reference Manifest if the Ref is not populated. |
| LocalRef | If a snapshot containing the Simple Object is a sync snapshot (i.e. its PcrHash value is extended to a TPM PCR), then LocalRef is a document internal reference to the CompositeHash of a non-sync snapshot that is populated in a Hash element. |

#### 3.1.2.4 XML

Source

```
<xs:complexType name="ValuesType">
  <xs:sequence>
    <xs:element name="Hash" type="core:DigestValueType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="ID" type="xs:ID"/>
  <xs:attribute name="Name" type="xs:normalizedString"/>
  <xs:attribute name="Type" type="xs:anySimpleType"/>
  <xs:attribute name="Ref" type="xs:anyURI"/>
  <xs:attribute name="Image" type="xs:base64Binary"/>
  <xs:attribute name="LocalRef" type="xs:IDREF"/>
</xs:complexType>
```

# 3.2 ELEMENTS

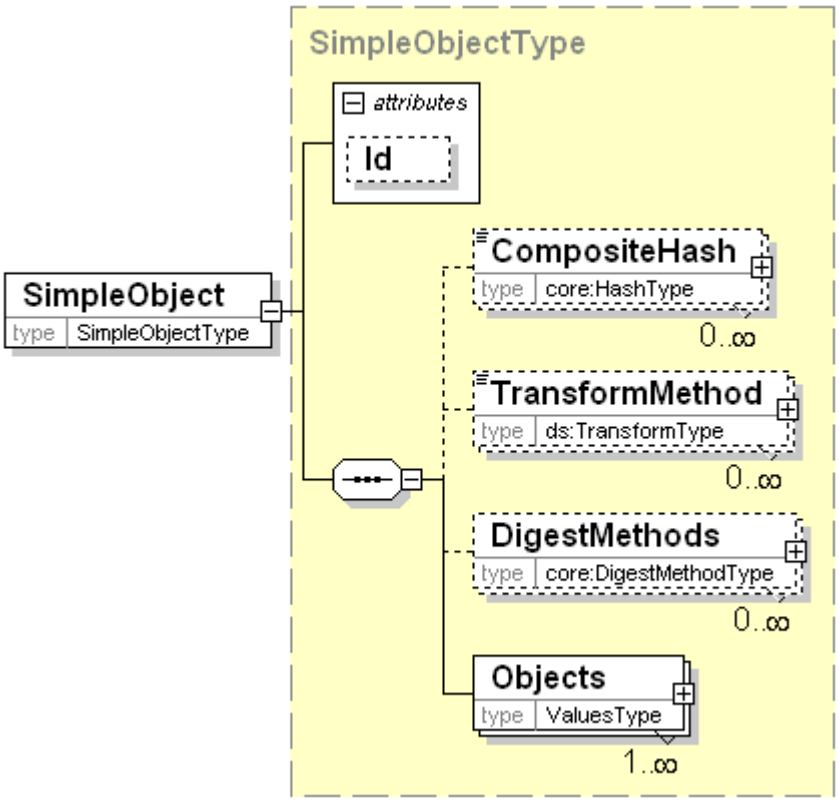
## 3.2.1 element SimpleObject

### 3.2.1.1 Description

The SimpleObject element is an instance of SimpleObjectType (see section 3.1.1). SimpleObject is instantiated by Reference Manifest [5] and Snapshot (within an Integrity Report) [9] XML documents. The DigestMethods element is optional to populate; however at least one Digest Method MUST be populated in either the SimpleObject DigestMethods element or in the parent structure – i.e. in the Reference Manifest DigestMethod or Snapshot DigestMethod structure, such that the CompositeHash hash value and Objects hash values can reference an appropriate Digest Method. Implementers SHOULD check to ensure that the intended Digest Method used to hash raw Object data is supported (i.e. by the component performing the hash function – PTS [8]).

### 3.2.1.2 Diagram

diagram




namespace http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#

type **SimpleObjectType**

properties content complex

children **CompositeHash** **TransformMethod** **DigestMethods** **Objects**

### 3.2.1.3 XML

source

```
<xs:element name="SimpleObject" type="SimpleObjectType"/>
```

## 3.2.2 element SimpleObject/DigestMethods

### 3.2.2.1 Description

The DigestMethods element is defined by the core:DigestMethodType complex type defined in [1]. This element defines the algorithm used in the computation of a SimpleObject digest.

### 3.2.2.2 XML

source
```
<xs:element name="DigestMethods" type="core:DigestMethodType" maxOccurs="unbounded"/>
```

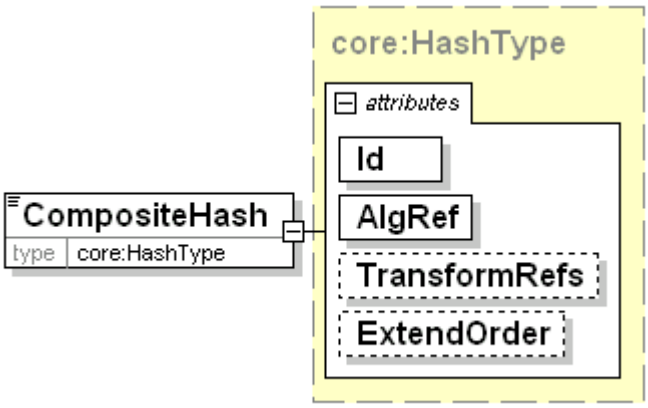
## 3.2.3 element SimpleObjectType/CompositeHash

### 3.2.3.1 Description

The CompositeHash element is defined by the core:HashType complex type defined in [1]. This element defines a composite hash of all the SimpleObject digests. The AlgRef attribute is a reference to the DigestMethod used to compute the composite hash value. The ExtendOrder attribute contains one or more references to the Object hash values and defines the order in which the hash values were used to compute the CompositeHash hash value.

### 3.2.3.2 Diagram

diagram




namespace http://www.trustedcomputinggroup.org/XML/SCHEMA/Simple_Object_v1_0#

type core:HashType

properties
isRef 0
content complex

attributes

| Name | Type | Use | Default | Fixed |
|---|---|---|---|---|
| Id | xs:ID | Required | | |
| AlgRef | xs:IDREF | Required | | |
| TransformRefs | xs:IDREFS | Optional | | |
| ExtendOrder | xs:IDREFS | Optional | | |

### 3.2.3.3 XML

source
```
<xs:element name="CompositeHash" type="core:HashType" minOccurs="0" maxOccurs="unbounded"/>
```

## 3.2.4 element SimpleObjectType/Objects

### 3.2.4.1 Description

The Objects element is defined by the ValuesType complex type (see 3.1.2). It contains the digest values of the measured components plus descriptive information.

### 3.2.4.2 XML

source
```
<xs:element name="Objects" type="ValuesType" maxOccurs="unbounded"/>
```

## 3.2.5 element SimpleObjectType/TransformMethod

### 3.2.5.1 Description

The TranformMethod element is defined by the XML W3C TransformType complex type [3] and describes the algorithm applied to the measured SimpleObject data prior to a hash computation operation.

### 3.2.5.2 XML

source
```
<xs:element name="TransformMethod" type="ds:TransformType" minOccurs="0" maxOccurs="unbounded"/>
```

## 3.2.6 element SimpleSnapshotObject/DigestMethods

### 3.2.6.1 Description

The DigestMethod element is defined by the core:DigestMethodType complex type [1].

### 3.2.6.2 XML

source
```
<xs:element name="DigestMethods" type="core:DigestMethodType" minOccurs="0" maxOccurs="unbounded"/>
```

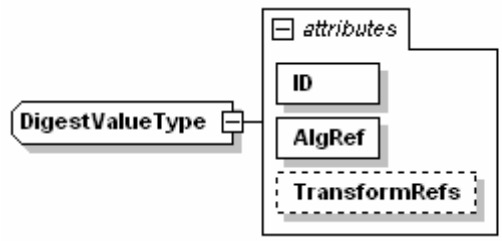
## 3.2.7 element ValuesType/Hash

### 3.2.7.1 Description

The Hash element is defined by the core:DigestValueType complex type [1]. The AlgRef attribute is a reference to the digest method used to calculate the hash value.

### 3.2.7.2 Diagram

diagram




namespace http://www.trustedcomputinggroup.org/XML/SCHEMA/Core_Integrity_v1_0_1#

type extension of ds:DigestValueType

properties
- base ds:DigestValueType

used by
- complexType core:HashType

attributes

| Name | Type | Use | Default |
|---|---|---|---|
| Id | xs:ID | required | |
| AlgRef | xs:IDREF | required | |
| TransformRefs | xs:IDREFS | optional | |

### 3.2.7.3 XML

source
```
<xs:element name="SignatureMethod" type="ds:SignatureMethodType"/>
```

# 4 References

[1] Trusted Computing Group, TCG IWG Core Integrity Schema, Specification Version 1.0, Revision 1.0, October 2006.

[2] Trusted Computing Group, TCG TPM Specification, TPM Main Part 2 TPM Structures, Specification version 1.2, Level 2, Revision 85, 13 February 2005.

[3] W3C, XML Schema, W3C Consortium, October 2004.

[4] Trusted Computing Group, TCG TPM Specification, TPM Main Part 3 Commands, Specification version 1.2, Level 2, Revision 85, 13 February 2005.

[5] Trusted Computing Group, TCG IWG Reference Manifest Schema, Specification Version 1.0, Revision 1.0, October 2006.

[6] Trusted Computing Group, TCG IWG Architecture Part II, Specification Version 1.0, Revision 1.0, October 2006.

[7] Trusted Computing Group, *TNC Architecture for Interoperability*, Specification Version 1.1, May 2006.

[8] Trusted Computing Group, TCG Platform Trust Services Interface IF-PTS, Specification Version 1.0, Revision 1.0, October 2006.

[9] Trusted Computing Group, TCG IWG Integrity Report Schema, Specification Version 1.0, Revision 1.0, October 2006.

[10] Trusted Computing Group, TCG IWG Security Qualities Schema, Specification Version 1.0, Revision 1.0, October 2006.

[11] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, Internet Engineering Task Force RFC 2119, March 1997.