BIAS Messaging Protocol v1.0

Editor's Draft 02, 22 August 2006

Document Identifier:
bias-1.0-biasmp-ed-05

Location:
Current: http://docs.oasis-open.org/bias/bias-1.0-biasmp

Technical Committee:
OASIS Biometric Identity Assurance Services (BIAS) Integration TC

Chair(s):
Cathy Tilton, Daon

Editor(s):
Ash Parikh, Raining Data Corporation
Murty Gurajara, Raining Data Corporation

Abstract:
This document specifies a service-oriented XML messaging protocol that implements the BIAS abstract operations specified in INCITS [Project 1823-D].

Status:
This document is updated periodically on no particular schedule.
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 www.oasis-open.org/committees/bias.

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 (www.oasis-open.org/committees/bias/ipr.php).

The non-normative errata page for this specification is located at www.oasis-open.org/committees/bias.
# Table of Contents

1  Introduction ................................................................................................................................. 64
   1.1 Terminology ............................................................................................................................... 64
   1.2 Normative References ............................................................................................................... 64
   1.3 Non-Normative References ...................................................................................................... 64
2  Conformance .................................................................................................................................. 75
3  Terms and definitions ..................................................................................................................... 86
4  BIAS architecture .......................................................................................................................... 108
   4.1 BIAS implementation ................................................................................................................. 119
   4.2 BIAS messages ......................................................................................................................... 119
   4.3 BIAS endpoints .......................................................................................................................... 119
   4.4 BIAS links ................................................................................................................................. 119
   4.5 Transport protocol bindings ...................................................................................................... 119
   4.6 Creation and destruction of BIAS links .................................................................................... 119
5  BIR format .................................................................................................................................. 1240
6  Biographic data format .................................................................................................................. 1341
7  Identification of BIAS endpoints and applications ........................................................................ 1442
8  General provisions ......................................................................................................................... 1543
9  BIAS message syntax ..................................................................................................................... 1614
10  BIAS types ................................................................................................................................... 1846
11  BIAS Client API methods and corresponding BIAS messages ..................................................... 1947
12  Conceptual tables .......................................................................................................................... 2048
   12.1 The MasterEndpoints conceptual table .................................................................................. 2048
       General ....................................................................................................................................... 2048
       Components .............................................................................................................................. 2048
       Entry deletion ............................................................................................................................ 2048
       Life cycle ................................................................................................................................. 2048
   12.2 The VisibleEndpoints conceptual table ................................................................................... 2149
       General ....................................................................................................................................... 2149
       Components .............................................................................................................................. 2149
       Entry deletion ............................................................................................................................ 2149
       Life cycle ................................................................................................................................. 2149
13  WSDL for the services .................................................................................................................... 2220
   13.1 Schemas ................................................................................................................................... 2220
       13.1.1 Schema header .................................................................................................................. 2220
       13.1.2 Global elements ............................................................................................................... 2220
       13.1.3 Types .............................................................................................................................. 2321
   13.2 Messages .................................................................................................................................. 2422
   13.3 Operations ............................................................................................................................... 2422
   13.4 Specification of the SOAP/HTTP binding .............................................................................. 2422
       13.4.1 General provisions .......................................................................................................... 2422
14  Security ....................................................................................................................................... 2624
   14.1 Security considerations .............................................................................................................. 2624
15  Error Handling and Notification .................................................................................................. 2725
1 Introduction

This standard, the BIAS messaging protocol, specifies the syntax, semantics, and encodings of a set of messages ("BIAS messages") that enable an application to invoke remote biometric identity assurance operations in a server component across node or process boundaries.

The BIAS messaging protocol is designed so that a conforming implementation does not have to support the entire functionality specified in this standard. Several conformance classes are defined in this standard to accommodate various degrees of support of such functionality.

Clause 4 describes the architecture of the BIAS messaging protocol, including some examples of possible system configurations using BIAS.

Clause 5 describes the format of the biometric data transferred by BIAS.

Clause 6 describes the format of the biographic data transferred by BIAS.

Clause 7 describes the identification of BIAS endpoints and applications.

Clause ... provides an overview of BIAS message exchanges.

Clause ... contains general provisions which are invoked by other clauses.

Clause 9 specifies the general syntax of a BIAS message.

Clause 10 specifies the mapping between BIAS types and the corresponding types that occur as components of BIAS messages.

Clause 11 specifies the syntax of individual BIAS messages and the actions to be performed when receiving a BIAS method call or a BIAS message related to a BIAS method call.

Clause 12 specifies a number of conceptual tables to be used by an implementation.

Clauses ..., to ..., contain detailed provisions which are invoked by other clauses.

Clause 13 is normative and specifies the SOAP/HTTP binding of BIAS.

Clause ..., is normative and contains the complete formal specification of BIAS messages.

Clause 14 specifies the security provisions and specifications for BIAS exchanges.

Clause 15 specifies error handling requirements.

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


[Reference] INCITS [Project 1823-D], Biometric Identity Assurance Services (BIAS), currently at working draft stage

1.3 Non-Normative References

[Reference] [Full reference citation]
2 Conformance

Annex A specifies the conformance requirements for systems/components claiming conformance to this standard.
3 Terms and definitions

The terms and definitions specified in INCITS [Project 1823-D] also apply to this Standard.

The following additional terms and definitions are used:

acknowledgement BIAS message

BIAS message conveying a response (acknowledgement) to a prior notification BIAS message

NOTE Not all notification BIAS messages have a corresponding acknowledgement BIAS message.

BIAS implementation

software entity that is capable of creating, processing, sending, and receiving BIAS messages in close relationship with BIAS client API method calls

BIAS endpoint

runtime entity, identified by an endpoint IRI, capable of sending and receiving BIAS messages, and containing a running BIAS implementation

BIAS link

logical connection between two BIAS endpoints, consisting of a mandatory request/response link channel and an optional notification/acknowledgement link channel, in which one BIAS endpoint plays the role of master and the other plays the role of slave

NOTE At most one BIAS link can exist between any pair of BIAS endpoints for either assignment of roles. If a BIAS link has two BIAS link channels, one of the two BIAS endpoints will be the master endpoint in both BIAS link channels, and the other one will be the slave endpoint in both. A BIAS link cannot exist between a pair of BIAS endpoints if both are master-role-capable or both are slave-role-capable.

BIAS link channel

logical connection from one BIAS endpoint to another BIAS endpoint using a specific transport protocol binding, in which one BIAS endpoint plays the role of master and the other plays the role of slave

BIAS message

message that can be sent from a BIAS endpoint to another BIAS endpoint through a BIAS link channel

CBEFF

data elements and BIR formats specified in ISO/IEC 19785-1

endpoint IRI

IRI that unambiguously identifies a BIAS endpoint

NOTE There are no constraints on the form of this IRI. In general (unless a binding specification prescribes otherwise) there is no relationship between the IRI scheme of the endpoint IRI and the binding(s) supported by the underlying implementation. This implies that in general, an endpoint IRI may not provide, by itself, sufficient information for locating the BIAS endpoint on a network.

fault BIAS message

BIAS message conveying information related to a fault condition experienced during the processing of a BIAS message
master endpoint (of a given BIAS endpoint)

BIAS endpoint that has been logically connected with a BIAS link to the given (slave) BIAS endpoint, and which is capable of sending request BIAS messages to the given BIAS endpoint and processing notification BIAS messages received from it.

NOTE A given BIAS endpoint may play both the role of master and the role of slave at the same time, with any number of other BIAS endpoints, provided that an appropriate set of BIAS links are established.

notification BIAS message

BIAS message conveying the notification of an event of interest to the receiving BIAS endpoint.

notification/acknowledgement link channel

BIAS link channel through which notification and acknowledgement BIAS messages are transferred.

NOTE A notification/acknowledgement link channel can only exist as part of a BIAS link. Its presence in a BIAS link is optional.

request BIAS message

BIAS message conveying a request for an action to be performed by the receiving BIAS endpoint.

request/response link channel

BIAS link channel through which request and response BIAS messages are transferred.

NOTE A request/response link channel can only exist as part of a BIAS link. Its presence in a BIAS link is mandatory.

response BIAS message

BIAS message conveying a response to a prior request BIAS message.

slave endpoint (of a given BIAS endpoint)

BIAS endpoint to which the given (master) BIAS endpoint has been logically connected with a BIAS link, and which is capable of processing request BIAS messages received from the given BIAS endpoint and sending notification BIAS messages to it.

NOTE A given BIAS endpoint may play both the role of master and the role of slave at the same time, with any number of other BIAS endpoints, provided that an appropriate number of BIAS links are established.

transport protocol binding

physical realization of a BIAS link channel, specifying what transport protocol to use, how to encode BIAS messages, how to compose transport-level messages carrying the encoded BIAS messages, and other details about the usage of the transport protocol.
4 BIAS architecture

The BIAS Architecture as shown in Figure 1:

1. A Client request to the BIAS Web services may be triggered by a human interaction OR any proxy system such as an ESB.
2. Client makes a call to the BIAS Implementation using OASIS Service Interfaces and Bindings (via WSDL)
3. Client sends and receives messages that conform to the BIAS schemas
4. The BIAS implementation maps the service call to the appropriate internal API or set of APIs and returns data according to the service interface.
5. The Bio API Services, INCITS services and Web Services are represented as circles.

At the heart of the BIAS messaging protocol are the concepts of BIAS message, BIAS endpoint, BIAS link, master/slave endpoints, and transport protocol binding.
4.1 BIAS implementation

4.2 BIAS messages

4.3 BIAS endpoints

4.4 BIAS links

4.5 Transport protocol bindings

4.6 Creation and destruction of BIAS links
5 BIR format

A number of BIAS message types defined in this Standard may contain one or more BIRs. When transferring such messages, the BIRs can be represented in any CBEFF patron format that meets certain requirements (see Error! Reference source not found.).

Do we want to always XML encode the metadata (CBEFF header) and then send the binary form of the entire BIR as Base64 encoded BLOB or as an attached object or ... We need to have a consistent approach for how this data is to be packaged/transmitted.

Patron formats are registered with the CBEFF registration authority (managed by the International Biometric Industry Association, IBIA, http://www.ibia.org/cbeff%5Fnew/) and are represented by a 4-byte integer value (2-byte Owner + 2-byte Type).
6 Biographic data format

Some BIAS messages involve the exchange of biographic data associated with an identity. BIAS will support the encoding of this data in any standard format for which a format identifier has been assigned.

The following biographic data formats are currently supported:

<table>
<thead>
<tr>
<th>Format Identifier</th>
<th>Format</th>
<th>Governing Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD</td>
<td>NIST Type 2 logical data record (XML encoding)</td>
<td>ANSI/NIST ITL1-2006</td>
</tr>
<tr>
<td>GJXDM</td>
<td><a href="http://www.it.ojp.gov/topic.jsp?topic_id=133">http://www.it.ojp.gov/topic.jsp?topic_id=133</a></td>
<td></td>
</tr>
<tr>
<td>CIQ</td>
<td>Customer Information Quality (CIQ) specification, <a href="http://www.oasis-open.org/committees/ciq">http://www.oasis-open.org/committees/ciq</a></td>
<td></td>
</tr>
<tr>
<td>UN/CEFACT</td>
<td>UN/CEFACT Core Components Technical Specification (UN/CEFACT CCTS)Naming and Design Rules (NDR)</td>
<td></td>
</tr>
</tbody>
</table>

Need to include here an XML schema containing:
- Format Identifier
- Biographic data (in the format indicated by the format identifier)
7 Identification of BIAS endpoints and applications

<planned content> describes and identifies of BIAS endpoints, applications, and an overview of BIAS message exchanges.
8 General provisions

<planned content> contains general provisions which are invoked by other clauses.
9 BIAS message syntax

A BIAS message consists of an abstract value of the ASN.1 type `BIASMessage` specified below.

```
BIASMessage ::= SEQUENCE {
    nature        CHOICE {
        request       BIASRequest,
        response       BIASResponse,
        notification      BIASNotification,
        acknowledgement     BIASAcknowledgement
    },
    ...}

BIASRequest ::= SEQUENCE {
    masterEndpointIRI    EndpointIRI,
    slaveEndpointIRI    EndpointIRI,
    linkNumber      UnsignedInt,
    requestSequenceNumber   UnsignedInt,
    params        CHOICE {
        addMaster       AddMaster-RequestParams,
        deleteMaster      DeleteMaster-RequestParams,
        ...}
}

BIASResponse ::= SEQUENCE {
    slaveEndpointIRI    EndpointIRI,
    masterEndpointIRI    EndpointIRI,
    linkNumber      UnsignedInt,
    requestSequenceNumber   UnsignedInt,
    params        CHOICE {
        addMaster       AddMaster-ResponseParams,
        deleteMaster      DeleteMaster-ResponseParams,
        ...}
    returnValue      BIAS-RETURN
}
```
BIASNotification ::= SEQUENCE {
  slaveEndpointIRI    EndpointIRI,
  masterEndpointIRI    EndpointIRI,
  linkNumber      UnsignedInt,
  notificationSequenceNumber UnsignedInt,
  params        CHOICE {
    masterDeletionEvent
    MasterDeletionEvent-NotificationParams,
    ...
  }
}

BIASAcknowledgement ::= SEQUENCE {
  masterEndpointIRI    EndpointIRI,
  slaveEndpointIRI    EndpointIRI,
  linkNumber      UnsignedInt,
  notificationSequenceNumber UnsignedInt,
  params        CHOICE {
    ...
  },
  returnValue      BIAS-RETURN
}

BIASFault ::= SEQUENCE {
  masterEndpointIRI    EndpointIRI,
  slaveEndpointIRI    EndpointIRI,
  linkNumber      UnsignedInt,
  notificationSequenceNumber UnsignedInt,
  params        CHOICE {
    ...
  },
  returnValue      BIAS-RETURN
}
10 BIAS types

<planned content> specifies the mapping between BIAS types and the corresponding types that occur as components of BIAS messages.
11 BIAS Client API methods and corresponding BIAS messages

<planned content> specifies the syntax of individual BIAS messages and the actions to be performed when receiving a BIAS method call or a BIAS message related to a BIAS method call.
12 Conceptual tables

This clause specifies type definitions for a series of conceptual tables. These types are defined to aid the specification of the behaviour of a BIAS implementation, but their abstract values do not occur in any BIAS message exchanged between BIAS endpoints, and therefore are never encoded. A conforming master BIAS endpoint is required to support all the tables specified in this clause. A conforming slave BIAS endpoint is required to support all the tables specified in this clause for slave endpoints. A BIAS endpoint can use any suitable representation for the conceptual tables and is not required to support serialization of their content.

NOTE A BIAS endpoint implementation may choose to support such serialization for administrative or debugging purposes, but this is not required. An example of this capability would be a management interface that supports a query of the current content of the conceptual tables.

12.1 The MasterEndpoints conceptual table

This conceptual table shall be present in all slave endpoints and is defined in ASN.1 as follows:

```
MasterEndpoints ::= SET OF endpoint MasterEndpoint

MasterEndpoint ::= SEQUENCE {
  masterEndpointIRI  EndpointIRI
}
```

General

An entry of this table represents a master endpoint that is a BIAS endpoint linked to the local endpoint.

Components

The component masterEndpointIRI shall contain the endpoint IRI of the master endpoint. This shall be different from the local endpoint IRI.

There shall not be two or more entries with the same value of the component masterEndpointIRI.

Entry deletion

This subclause only applies as explicitly invoked by other clauses of this Standard when an entry of the MasterEndpoints table is deleted.

Let masterEndpointIRI be the value of the component masterEndpointIRI of the entry being deleted.

For each entry of the UnitEventNotificationDisablers table (see Error! Reference source not found.) where the component referrerEndpointIRI has the value masterEndpointIRI, the BIAS implementation shall delete the entry of the UnitEventNotificationDisablers table (subclause Error! Reference source not found. applies).

Life cycle

An entry can be added to the MasterEndpoints table:

when the BIAS implementation receives an addMaster request BIAS message from a BIAS endpoint (see Error! Reference source not found.).
An entry of the MasterEndpoints table can be deleted:

when the BIAS implementation receives a call to the method ... from the local application (see Error! Reference source not found.); and

when the BIAS implementation receives a deleteMaster request BIAS message from a BIAS endpoint (see Error! Reference source not found.).

12.2 The VisibleEndpoints conceptual table

This conceptual table shall be present in all BIAS endpoints and is defined in ASN.1 as follows:

```
VisibleEndpoints :::= SET OF endpoint VisibleEndpoint
VisibleEndpoint :::= ....
```

General

An entry of this table represents a visible endpoint, which is a BIAS endpoint whose components are visible to the local application.

This table has one entry for the local endpoint and one entry for each slave endpoint of the local endpoint (zero or more). Each entry contains a copy of the BIAS implementation schema present in the (local or slave) BIAS endpoint.

Components

The component hostingEndpointIRI shall contain the endpoint IRI of the visible endpoint. The visible endpoint shall be either the local endpoint or a slave endpoint.

The other components shall contain a copy of the attributes of the BIAS implementation schema in the BIAS endpoint.

There shall not be two or more entries with the same value of the component hostingEndpointIRI.

Entry deletion

This subclause only applies as explicitly invoked by other clauses of this International Standard when an entry of the VisibleEndpoints table is deleted.

Let hostingEndpointIRI be the value of the component hostingEndpointIRI of the entry being deleted.

Life cycle
13 WSDL for the services

13.1 Schemas

13.1.1 Schema header

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="../BIAS"
targetNamespace="../BIAS"
attributeFormDefault="qualified"
>

13.1.2 Global elements

13.1.2.1 Global element request and type BIASRequest

<xs:complexType name="BIASRequest">
  <xs:sequence>
    <xs:element name="masterEndpointIRI" type="EndpointIRI"/>
    <xs:element name="slaveEndpointIRI" type="EndpointIRI"/>
    <xs:element name="linkNumber" type="xs:unsignedInt"/>
    <xs:element name="sequenceNumber" type="xs:unsignedInt"/>
    <xs:choice>
      <xs:element name="addMaster"
        type="AddMaster-RequestParams"/>
      <xs:element name="deleteMaster"
        type="DeleteMaster-RequestParams"/>
      ....
    </xs:choice>
  </xs:sequence>
</xs:complexType>

The semantics and the use of this XSD type are defined by interpreting an element of this type as the EXTENDED-XER encoding of an abstract value of the ASN.1 type BIASRequest specified in clause 98.

13.1.2.2 Global element response and type BIASResponse

<xs:complexType name="BIASResponse">
  <xs:sequence>
    <xs:element name="slaveEndpointIRI" type="EndpointIRI"/>
    <xs:element name="masterEndpointIRI" type="EndpointIRI"/>
    <xs:element name="linkNumber" type="xs:unsignedInt"/>
    <xs:element name="sequenceNumber" type="xs:unsignedInt"/>
    <xs:choice>
      <xs:element name="addMaster"
        type="AddMaster-ResponseParams"/>
      <xs:element name="deleteMaster"
        type="DeleteMaster-ResponseParams"/>
      ....
    </xs:choice>
  </xs:sequence>
</xs:complexType>
The semantics and the use of this XSD type are defined by interpreting an element of this type as the EXTENDED-XER encoding of an abstract value of the ASN.1 type BIASResponse specified in clause 9.

13.1.2.3 Global element notification and type BIASNotification

   <xs:element name="notification" type="BIASNotification"/>

   <xs:complexType name="BIASNotification">
     <xs:sequence>
       <xs:element name="slaveEndpointIRI" type="EndpointIRI"/>
       <xs:element name="masterEndpointIRI" type="EndpointIRI"/>
       <xs:element name="linkNumber" type="xs:unsignedInt"/>
       <xs:element name="sequenceNumber" type="xs:unsignedInt"/>
       <xs:choice>
         <xs:element name="masterDeletionEvent" type="MasterDeletionEvent-NotificationParams"/>
         ....
         <xs:element name="slaveDeletionEvent" type="SlaveDeletionEvent-NotificationParams"/>
         ....
       </xs:choice>
       <xs:any minOccurs="0" maxOccurs="unbound"/>
     </xs:sequence>
   </xs:complexType>

The semantics and the use of this XSD type are defined by interpreting an element of this type as the EXTENDED-XER encoding of an abstract value of the ASN.1 type BIASNotification specified in clause 9.

13.1.2.4 Global element acknowledgement and type BIASAcknowledgement

   <xs:element name="acknowledgement" type="BIASAcknowledgement"/>

   <xs:complexType name="BIASAcknowledgement">
     <xs:sequence>
       <xs:element name="masterEndpointIRI" type="EndpointIRI"/>
       <xs:element name="slaveEndpointIRI" type="EndpointIRI"/>
       <xs:element name="linkNumber" type="xs:unsignedInt"/>
       <xs:element name="sequenceNumber" type="xs:unsignedInt"/>
       <xs:choice>
         ....
         <xs:element name="masterDeletionEvent" type="MasterDeletionEvent-NotificationParams"/>
         <xs:element name="slaveDeletionEvent" type="SlaveDeletionEvent-NotificationParams"/>
         ....
       </xs:choice>
       <xs:element name="returnValue" type="BIAS-RETURN"/>
       <xs:any minOccurs="0" maxOccurs="unbound"/>
     </xs:sequence>
   </xs:complexType>

The semantics and the use of this XSD type are defined by interpreting an element of this type as the EXTENDED-XER encoding of an abstract value of the ASN.1 type BIASAcknowledgement specified in clause 9.

13.1.3 Types

13.1.3.1 Type EndpointIRI

   <xs:simpleType name="EndpointIRI">
     <restriction base="xs:token"/>
   </xs:simpleType>
The semantics and the use of this XSD type are defined by interpreting an element of this type as the EXTENDED-XER encoding of an abstract value of the ASN.1 type `EndpointIRI` specified in Error! Reference source not found..

### 13.2 Messages

### 13.3 Operations

### 13.4 Specification of the SOAP/HTTP binding

#### 13.4.1 General provisions

13.4.1.1 This clause specifies the use of BIAS with the SOAP protocol using HTTP or HTTPS as a carrier.

13.4.1.2 A transport-level message for the binding specified in this clause shall consist of an HTTP message which carries a SOAP message encoded in XML 1.0, in accordance with the HTTP binding specified in W3C SOAP, clause 7.

13.4.1.3 The version of SOAP shall be either 1.1 or 1.2.

13.4.1.4 The version of HTTP shall be either 1.0 or 1.1. HTTP over TLS (HTTPS) may be used.

13.4.1.5 The HTTP SOAP Transmission Optimization Feature specified in W3C SOAP MTOM, clause 2, may be used. When this feature is used, any binary data (such as the BDB of a BIR) contained in a BIAS message can be included in the HTTP message as a binary block and does not require the use of Base64.

13.4.1.6 A request BIAS message shall be carried in an HTTP request message with the HTTP method POST. The body of the SOAP envelope within the HTTP request message shall contain a single occurrence of the global element `request` specified in Error! Reference source not found.,424.1. The `soapAction` HTTP header (for SOAP 1.1) or the `action` parameter of the MIME media type (for SOAP 1.2) shall be set to "/BIASRequest".

13.4.1.7 A response BIAS message shall be carried in an HTTP response message returned by the slave endpoint in reply to the HTTP request message containing the corresponding request BIAS message. The body of the SOAP envelope in the HTTP response message shall contain a single occurrence of the global element `response` specified in Error! Reference source not found.,424.2.

13.4.1.8 A notification BIAS message shall be carried in an HTTP request message with the HTTP method POST. The body of the SOAP envelope within the HTTP request message shall contain a single occurrence of the global element `notification` specified in Error! Reference source not found.,424.3. The `soapAction` HTTP header (for SOAP 1.1) or the `action` parameter of the MIME media type (for SOAP 1.2) shall be set to "/BIASNotification".

13.4.1.9 An acknowledgement BIAS message shall be carried in an HTTP response message returned by the master endpoint in reply to the HTTP request message containing the corresponding notification BIAS message. The body of the SOAP envelope in the HTTP response message shall contain a single occurrence of the global element `acknowledgement` specified in Error! Reference source not found.,424.4. For notification BIAS messages that do not expect a corresponding
acknowledgement BIAS message, an HTTP response message with an empty body shall be sent by the
master endpoint in response to the HTTP request.

The semantics of each of the XSD type definitions specified in this clause is defined by
interpreting an element of that type as the EXTENDED-XER encoding (see ITU-T Rec. X.693 Amd. 1 | ISO/IEC 8825-4 Amd.1) of an abstract value of an ASN.1 type that is specified in the referenced
subclause of clause 11 or Error! Reference source not found.. The provisions of the referenced
subclause indirectly apply to the creation and processing of elements of that XSD type through this
semantic mapping.
14 Security

14.1 Security considerations

There are three different ways of applying security techniques to a BIAS message when using this binding:

- point-to-point (carrier-level) security can be achieved by using HTTP over TLS (HTTPS) instead of regular HTTP over TCP/IP;
- standard encryption and integrity techniques can be applied to the entire XML message or one or more portions of it (such as a BIR or a BDB in a BIR contained in the message) by using XML encryption and XML signature standards such as W3C XMLENC and W3C XMLDS; or
- the CBEFF standard security provisions can be used to encrypt a BDB or to sign a BIR contained in a BIAS message.

<planned content> Discuss applicability/use of WS-Security, SAML, etc.

- How is access control handled?
- How is application level data protection handled?
15 Error Handling and Notification

<planned content> specifies error handling requirements.

- Business & system errors?
- Error notifications
- Reliable messaging?? Compensation? Transaction management?
A. Annex A Conformance

A.1 Clauses 4 to Error! Reference source not found. of this standard specify the syntax and semantics of BIAS messages and the behaviour of a BIAS implementation on reception of incoming BIAS method calls and BIAS messages.

A.2 This standard is implemented by a BIAS entity. A BIAS implementation has the ability to create, process, send, and receive BIAS messages. This entity may be an application which directly participates in BIAS interactions or an implementation that acts on behalf of an application. (BIAS implementations are not specified in this standard, but may be in the future.)

A.3 Conformance to this standard is organized into three role capability classes (master-role-capable, slave-role-capable, and dual-role-capable) as follows:

<table>
<thead>
<tr>
<th>Role capability class</th>
<th>Conformance level</th>
</tr>
</thead>
<tbody>
<tr>
<td>master-role-capable</td>
<td>BIAS implementation</td>
</tr>
<tr>
<td>slave-role-capable</td>
<td>BIAS implementation</td>
</tr>
<tr>
<td>dual-role-capable</td>
<td>BIAS implementation</td>
</tr>
</tbody>
</table>

A.5 A software entity may claim to be a master-role-capable BIAS entity if and only if it:

• exposes a BIAS client API to a local application;
• processes incoming BIAS client API calls from a local application as specified in clauses 4 to Error! Reference source not found.;
• processes incoming notification BIAS messages as specified in those clauses; and
• never produces BIAS messages except as specified in those clauses.

A.6 A software entity may claim to be a slave-role-capable BIAS implementation if and only if it:

• uses the BIAS server API;
• processes incoming request BIAS messages as specified in those clauses; and
• never produces BIAS messages except as specified in those clauses.

A.7 A software entity may claim to be a dual-role-capable BIAS implementation if and only if it satisfies the conditions for both a master-role-capable and a slave-role-capable BIAS implementation.

A.8 A software entity may claim to be a BIAS implementation (master-role-capable, slave-role-capable, or dual-role-capable) if and only if it is possible to determine, for any possible sequence (of arbitrary length) of incoming and outgoing BIAS messages, a congruous sequence of actions which could have been performed by an ideal BIAS implementation (of that role capability class) conceptually present within the software entity and that would have resulted in that sequence of BIAS messages.

A.9 As a corollary of the previous subclauses, the following interfaces are subject to conformance testing for each conformance class:
Table 2 – Conformance testing

<table>
<thead>
<tr>
<th>Conformance class</th>
<th>Messaging interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>master-role-capable implementation</td>
<td>BIAS</td>
</tr>
<tr>
<td>slave-role-capable implementation</td>
<td>BIAS</td>
</tr>
<tr>
<td>dual-role-capable implementation</td>
<td>BIAS</td>
</tr>
</tbody>
</table>

A.10 Conformance to the transport protocol binding specified in clauses .... is defined separately for each of those clauses. However, conformance testing of an implementation cannot be performed unless there is at least one binding that is supported both by the implementation and by the testing tool. In addition, conformance with respect to the correct processing of notification BIAS messages cannot be assessed unless the binding used in the test supports the transfer of notification BIAS messages.
**B. Use Cases**

The INCITS document will include a set of high level usage scenarios. This annex will provide more detailed operational sequence/flow charts that show how these scenarios would be implemented using the BIAS messaging protocol.
C. Acknowledgements

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

Participants:

[Participant Name, Affiliation | Individual Member]
[Participant Name, Affiliation | Individual Member]
D. Non-Normative Text
## E. Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2006-04-19</td>
<td>Alessandro Triglia</td>
<td>Initial draft</td>
</tr>
<tr>
<td>1</td>
<td>2006-08-16</td>
<td>Ash Parikh</td>
<td>Raining Data Corporation</td>
</tr>
<tr>
<td>2</td>
<td>2006-10-25</td>
<td>Ash Parikh</td>
<td>Raining Data Corporation</td>
</tr>
</tbody>
</table>
F. Open Issues

- We must identify and describe the fault messages associated with each operation in our web services. We have had discussions about 2 sets of fault messages, one set describing system faults and another set describing business problems. We should take the time to identify the structure and content of each fault message. Once we determine the structures, the fault definitions can be updated in the WSDL file.

- We must add the bindings for port types in the WSDL file. SOAP over HTTP/S will be one. Will there be others?
## E. Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Changes Made</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2006-04-19</td>
<td>Alessandro Triglia</td>
<td>Initial draft</td>
</tr>
<tr>
<td>1</td>
<td>2006-08-16</td>
<td>Ash Parikh</td>
<td>Raining Data Corporation</td>
</tr>
</tbody>
</table>