



# Biometric Identity Assurance Services (BIAS) SOAP Profile Version 1.0 **Plus** **Errata 02**

OASIS Standard **incorporating Approved Errata 02**

~~24 May 2012~~ **06 May 2014**

## Specification URIs

### This version:

<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/biasprofile-v1.0-errata02-os-complete.doc> (Authoritative)  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/biasprofile-v1.0-errata02-os-complete.html>  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/biasprofile-v1.0-errata02-os-complete.pdf>

### Previous version:

<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata01/os/biasprofile-v1.0-errata01-os-complete.doc> (Authoritative)  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata01/os/biasprofile-v1.0-errata01-os-complete.html>  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/errata01/os/biasprofile-v1.0-errata01-os-complete.pdf>

### Latest version:

<http://docs.oasis-open.org/bias/soap-profile/v1.0/biasprofile-1.0.doc> (Authoritative)  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/biasprofile-1.0.html>  
<http://docs.oasis-open.org/bias/soap-profile/v1.0/biasprofile-1.0.pdf>

## Technical Committee:

~~OASIS Biometric Identity Assurance Services (BIAS) Integration TC~~ OASIS Biometrics TC

## Chairs:

~~Cathy Tilton (cathy.tilton@daon.com), Daon~~  
Kevin Mangold ([kevin.mangold@nist.gov](mailto:kevin.mangold@nist.gov)), NIST

## Editors:

Kevin Mangold ([kevin.mangold@nist.gov](mailto:kevin.mangold@nist.gov)), NIST  
Matthew Swayze ([matthew.swayze@daon.com](mailto:matthew.swayze@daon.com)), Daon  
Cathy Tilton ([cathy.tilton@daon.com](mailto:cathy.tilton@daon.com)), Daon

## Additional artifacts:

This prose specification is one component of a Work Product which also includes:

- *Biometric Identity Assurance Services (BIAS) SOAP Profile Version 1.0 Errata 02*. Edited by Kevin Mangold and Cathy Tilton. 06 May 2014. OASIS Approved Errata. <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/biasprofile-v1.0-errata02-os.html>.
- XML schema: <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/cbeff.xsd>.
- WSDL: <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/BIAS.wsdl>.

**Related work:**

This specification is related to:

- ANSI INCITS 442-2010, Biometric Identity Assurance Services (BIAS)

**Declared XML namespaces:**

<http://docs.oasis-open.org/bias/ns/bias-1.0/>

<http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/>

**Abstract:**

This document specifies a SOAP profile that implements the BIAS abstract operations specified in INCITS 442 as SOAP messages [and incorporates Approved Errata](#).

**Status:**

This document was last revised or approved by the OASIS Biometrics TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at [https://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=biometrics#standards-track](https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=biometrics#standards-track).

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 <https://www.oasis-open.org/committees/biometrics/>.

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 (<https://www.oasis-open.org/committees/bias/ipr.php>).

**Citation format:**

When referencing this specification the following citation format should be used:

**[BIASPROFILE]**

*Biometric Identity Assurance Services (BIAS) SOAP Profile Version 1.0 Plus Errata 02*. Edited by Kevin Mangold, Matthew Swayze, and Cathy Tilton. 06 May 2014. OASIS Standard incorporating Approved Errata 02. <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/biasprofile-v1.0-errata02-os-complete.html>. Latest version: <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/biasprofile-v1.0-errata02-complete.html>.

---

## Notices

Copyright © OASIS Open 2014. All Rights Reserved.

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 <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

---

# Table of Contents

|        |  |    |
|--------|--|----|
| 1      | Introduction.....                                      | 8  |
| 1.1    | Purpose/Scope .....                                    | 8  |
| 1.2    | Overview .....   | 8  |
| 1.3    | Background.....  | 8  |
| 1.4    | Relationship to Other Standards .....                  | 9  |
| 1.5    | Terminology .....                                      | 9  |
| 1.6    | References.....  | 10 |
| 1.6.1  | Normative References.....                              | 10 |
| 1.6.2  | Non-Normative References.....                          | 11 |
| 2      | Design Concepts and Architecture (non-normative) ..... | 13 |
| 2.1    | Philosophy .....                                       | 13 |
| 2.2    | Context.....   | 13 |
| 2.3    | Architecture.....                                      | 13 |
| 3      | Data dictionary.....                                   | 16 |
| 3.1    | Documentation Conventions.....                         | 16 |
| 3.2    | Common Elements .....                                  | 17 |
| 3.2.1  | ApplicationIdentifier .....                            | 17 |
| 3.2.2  | ApplicationUserIdentifier .....                        | 17 |
| 3.2.3  | BaseBIRType .....                                      | 17 |
| 3.2.4  | BIASBiometricDataType.....                             | 17 |
| 3.2.5  | BIASFaultCode.....                                     | 18 |
| 3.2.6  | BIASFaultDetail .....                                  | 18 |
| 3.2.7  | BIASIdentity.....                                      | 19 |
| 3.2.8  | BIASIDType.....  | 19 |
| 3.2.9  | BinaryBIR .....  | 19 |
| 3.2.10 | BiographicDataItem Type.....                           | 20 |
| 3.2.11 | BiographicDataSetType .....                            | 20 |
| 3.2.12 | BiographicDataType .....                               | 21 |
| 3.2.13 | BiometricDataElementType.....                          | 21 |
| 3.2.14 | BiometricDataListType .....                            | 22 |
| 3.2.15 | CandidateListResultType .....                          | 22 |
| 3.2.16 | CandidateListType .....                                | 22 |
| 3.2.17 | CandidateType .....                                    | 23 |
| 3.2.18 | CapabilityListType .....                               | 23 |
| 3.2.19 | CapabilityName .....                                   | 23 |
| 3.2.20 | CapabilityType.....                                    | 24 |
| 3.2.21 | CBEFF_BIR_ListType.....                                | 24 |
| 3.2.22 | CBEFF_BIR_Type.....                                    | 24 |
| 3.2.23 | Classification .....                                   | 25 |
| 3.2.24 | ClassificationAlgorithmType .....                      | 25 |
| 3.2.25 | ClassificationData.....                                | 25 |
| 3.2.26 | EncounterListType .....                                | 26 |
| 3.2.27 | FusionDecision .....                                   | 26 |

|        |                                     |    |
|--------|-------------------------------------|----|
| 3.2.28 | FusionInformationListType .....     | 26 |
| 3.2.29 | FusionInformationType .....         | 26 |
| 3.2.30 | FusionResult .....                  | 27 |
| 3.2.31 | FusionScore .....                   | 27 |
| 3.2.32 | GenericRequestParameters .....      | 27 |
| 3.2.33 | IdentifySubjectResultType .....     | 27 |
| 3.2.34 | InformationType .....               | 27 |
| 3.2.35 | ListFilterType .....                | 28 |
| 3.2.36 | MatchType .....                     | 28 |
| 3.2.37 | ProcessingOptionsType .....         | 28 |
| 3.2.38 | ProductID .....                     | 28 |
| 3.2.39 | QualityData .....                   | 28 |
| 3.2.40 | ResponseStatus .....                | 29 |
| 3.2.41 | ReturnCode .....                    | 29 |
| 3.2.42 | Score .....                         | 29 |
| 3.2.43 | TokenResultType .....               | 29 |
| 3.2.44 | TokenType .....                     | 30 |
| 3.2.45 | URI_BIR .....                       | 30 |
| 3.2.46 | VendorIdentifier .....              | 30 |
| 3.2.47 | Version .....                       | 30 |
| 3.2.48 | VersionType .....                   | 30 |
| 3.2.49 | XML_BIR .....                       | 30 |
| 4      | BIAS Messages .....                 | 32 |
| 4.1    | Primitive Operations .....          | 32 |
| 4.1.1  | AddSubjectToGallery .....           | 32 |
| 4.1.2  | CheckQuality .....                  | 33 |
| 4.1.3  | ClassifyBiometricData .....         | 35 |
| 4.1.4  | CreateSubject .....                 | 37 |
| 4.1.5  | DeleteBiographicData .....          | 38 |
| 4.1.6  | DeleteBiometricData .....           | 39 |
| 4.1.7  | DeleteSubject .....                 | 40 |
| 4.1.8  | DeleteSubjectFromGallery .....      | 41 |
| 4.1.9  | GetIdentifySubjectResults .....     | 43 |
| 4.1.10 | IdentifySubject .....               | 45 |
| 4.1.11 | ListBiographicData .....            | 47 |
| 4.1.12 | ListBiometricData .....             | 50 |
| 4.1.13 | PerformFusion .....                 | 54 |
| 4.1.14 | QueryCapabilities .....             | 56 |
| 4.1.15 | RetrieveBiographicInformation ..... | 57 |
| 4.1.16 | RetrieveBiometricInformation .....  | 59 |
| 4.1.17 | SetBiographicData .....             | 61 |
| 4.1.18 | SetBiometricData .....              | 63 |
| 4.1.19 | TransformBiometricData .....        | 64 |
| 4.1.20 | UpdateBiographicData .....          | 66 |
| 4.1.21 | UpdateBiometricData .....           | 68 |

|          |  |     |
|----------|--|-----|
| 4.1.22   | VerifySubject .....                                  | 69  |
| 4.2      | Aggregate Operations.....                            | 71  |
| 4.2.1    | Enroll .....   | 71  |
| 4.2.2    | GetEnrollResults .....                               | 72  |
| 4.2.3    | GetIdentifyResults .....                             | 74  |
| 4.2.4    | GetVerifyResults .....                               | 75  |
| 4.2.5    | Identify .....                                       | 77  |
| 4.2.6    | RetrieveInformation .....                            | 78  |
| 4.2.7    | Verify .....   | 79  |
| 5        | Message structure and rules .....                    | 82  |
| 5.1      | Purpose and constraints .....                        | 82  |
| 5.2      | Message requirements .....                           | 83  |
| 5.3      | Handling binary data .....                           | 84  |
| 5.3.1    | Base64 encoding.....                                 | 84  |
| 5.3.2    | Use of XOP.....                                      | 84  |
| 5.4      | Discovery .....                                      | 85  |
| 5.5      | Identifying operations.....                          | 85  |
| 5.5.1    | Operation name element.....                          | 85  |
| 5.5.2    | WS-Addressing Action .....                           | 86  |
| 5.6      | Security .....                                       | 87  |
| 5.6.1    | Use of SSL 3.0 or TLS 1.0 .....                      | 87  |
| 5.6.2    | Data Origin Authentication .....                     | 87  |
| 5.6.3    | Message Integrity .....                              | 87  |
| 5.6.4    | Message Confidentiality .....                        | 87  |
| 5.6.5    | CBEFF BIR security features .....                    | 87  |
| 5.6.6    | Security Considerations .....                        | 88  |
| 5.6.7    | Security of Stored Data .....                        | 88  |
| 5.6.8    | Key Management .....                                 | 88  |
| 5.7      | Use with other WS* standards.....                    | 88  |
| 5.8      | Tailoring .....                                      | 88  |
| 6        | Error handling.....                                  | 90  |
| 6.1      | BIAS operation return codes.....                     | 90  |
| 6.2      | SOAP fault codes .....                               | 90  |
| 7        | Conformance .....                                    | 91  |
| Annex A. | XML Schema .....                                     | 92  |
| Annex B. | BIAS Patron format specification.....                | 173 |
| B.1      | Patron .....   | 173 |
| B.2      | Patron identifier .....                              | 173 |
| B.3      | Patron format name.....                              | 173 |
| B.4      | Patron format identifier .....                       | 173 |
| B.5      | ASN.1 object identifier for this patron format ..... | 173 |
| B.6      | Domain of use .....                                  | 173 |
| B.7      | Version identifier.....                              | 173 |
| B.8      | CBEFF version .....                                  | 173 |
| B.9      | General.....   | 174 |

|   |     |
|---|-----|
| B.10 Specification .....  | 174 |
| B.11 Element <BIR> .....  | 175 |
| B.11.1 Syntax .....   | 175 |
| B.11.2 Semantics .....  | 175 |
| B.12 Element <Version> .....  | 176 |
| B.12.1 Syntax .....   | 176 |
| B.12.2 Semantics .....  | 176 |
| B.13 Element <CBEFFVersion> .....                                     | 177 |
| B.13.1 Syntax .....   | 177 |
| B.13.2 Semantics .....  | 177 |
| B.14 Element <BIRInfo> .....  | 177 |
| B.14.1 Syntax .....   | 177 |
| B.14.2 Semantics .....  | 178 |
| B.15 Element <BDBInfo> .....  | 179 |
| B.15.1 Syntax .....   | 179 |
| B.15.2 Semantics .....  | 181 |
| B.16 Element <SBInfo> .....   | 185 |
| B.16.1 Syntax .....   | 185 |
| B.16.2 Semantics .....  | 186 |
| B.17 Representation of Integers .....                                 | 186 |
| B.18 Representation of Octet Strings .....                            | 186 |
| B.19 Representation of Date and Time of the Day .....                 | 187 |
| B.20 Representation of Universally Unique Identifiers .....           | 188 |
| B.21 Patron format conformance statement .....                        | 188 |
| B.21.1 Identifying information .....                                  | 188 |
| B.21.2 ISO/IEC 19785-1:2006/Amd 1:2010 to Patron Format Mapping ..... | 188 |
| B.22 XML schema of the BIAS patron format .....                       | 190 |
| B.23 Sample BIR encoding .....  | 193 |
| Annex C. Use Cases (non-normative) .....                              | 195 |
| C.1 Verification Use Case .....                                       | 195 |
| C.2 Asynchronous Verification Use Case .....                          | 196 |
| C.3 Primitive Verification Use Case .....                             | 197 |
| C.4 Identification Use Case .....                                     | 198 |
| C.5 Biometric Enrollment Use Case .....                               | 199 |
| C.6 Primitive Enrollment Use Case .....                               | 200 |
| Annex D. Samples (non-normative) .....                                | 201 |
| D.1 Create Subject Request/Response Example .....                     | 201 |
| D.2 Set Biographic Data Request/Response Example .....                | 203 |
| D.3 Set Biometric Data Request/Response Example .....                 | 204 |
| Annex E. Acknowledgements .....                                       | 207 |
| Annex F. Revision History .....                                       | 208 |

---

# 1 Introduction

## 1.1 Purpose/Scope

This Organization for the Advancement of Structured Information Standards (OASIS) Biometric Identity Assurance Services (BIAS) profile specifies how to use the eXtensible Markup Language (XML) [XML10] defined in ANSI INCITS 442-2010 – Biometric Identity Assurance Services [INCITS-BIAS] to invoke Simple Object Access Protocol (SOAP) -based services that implement BIAS operations. These SOAP-based services enable an application to invoke biometric identity assurance operations remotely in a Services Oriented Architecture (SOA) infrastructure.

Not included in the scope of BIAS is the incorporation of biometric authentication as an integral component of an authentication or security protocol. (However, BIAS services may be leveraged to implement biometric authentication in the future.)

## 1.2 Overview

In addition to this introduction, this standard includes the following:

- Clause 2 presents the design concepts and architecture for invoking SOAP-based services that implement BIAS operations.
- Clause 3 presents the namespaces necessary to implement this profile, INCITS BIAS data elements, and identifies relationships to external data definitions.
- Clause 4 specifies the content of the BIAS messages.
- Clause 5 presents the BIAS message structure, as well as rules and considerations for its application.
- Clause 6 presents information on error handling.
- Clause 7 specifies conformance requirements.
- Annexes include the OASIS BIAS XML schema/sample Web Service Definition Language (WSDL), BIAS CBEFF Patron Format, use cases, sample code, acknowledgements, and the revision history of this profile.

## 1.3 Background

In late 2005/early 2006, a gap was identified in the existing biometric standards portfolio with respect to biometric services. The Biometric Identity Assurance Services standard proposal was for a collaborative effort between government and private industry to provide a services-based framework for delivering identity assurance capabilities, allowing for platform and application independence. This standard proposal required the attention of two major technical disciplines: biometrics and service architectures. The expertise of both disciplines was required to ensure the standard was technically sound, market relevant, and achieved widespread adoption. The International Committee for Information Technology Standards (INCITS) M1 provided the standards leadership relevant to biometrics, defining the “taxonomy” of biometric operations and data elements. OASIS provided the standards leadership relevant to service architectures with an initial focus on web services, defining the schema and SOAP messaging.

The driving requirements of the BIAS standard proposal were to provide the ability to remotely invoke biometric operations across an SOA infrastructure; to provide business level operations without constraining the application/business logic that implements those operations; to be as generic as possible – technology, framework, & application domain independent; and to provide basic capabilities that can be used to construct higher level, aggregate/composite operations.



## 42 1.4 Relationship to Other Standards

43 This OASIS BIAS profile comprises a companion standard to ANSI INCITS 442-2010 – Biometric Identity  
44 Assurance Services, which defines the BIAS requirements and taxonomy, specifying the identity  
45 assurance operations and the associated data elements. This OASIS BIAS profile specifies the design  
46 concepts and architecture, data model and data dictionary, message structure and rules, and error  
47 handling necessary to invoke SOAP-based services that implement BIAS operations.

48 Together, the BIAS standard and the BIAS profile provide an open framework for deploying and remotely  
49 invoking biometric-based identity assurance capabilities that can be readily accessed across an SOA  
50 infrastructure.

51 This relationship allows the leveraging of the biometrics and web services expertise of the two standards  
52 development organizations. Existing standards are available in both domains and many of these  
53 standards will provide the foundation and underlying capabilities upon which the biometric services  
54 depend.

## 55 1.5 Terminology

56 The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD  
57 NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described  
58 in [RFC2119].

59 The following additional terms and definitions are used:

60 Note: The terms and definitions specified in INCITS (InterNational Committee for Information Technology  
61 Standards) (Project 1823-D) also apply to this Standard.

62  
63 BIAS operation and data element names are not defined here, but in their respective sections.

### 64 **BIAS**

65 Biometric Identity Assurance Services

### 66 **BIR**

67 Biometric Information Record

### 68 **ESB**

69 Enterprise Service Bus

### 70 **HTTP**

71 HyperText Transfer Protocol

### 72 **HTTPS**

73 HyperText Transfer Protocol over SSL or HTTP Secure

### 74 **IRI**

75 Internationalized Resource Identifier

### 76 **SOA**

77 Service-Oriented Architecture

### 78 **SOAP**

79 Simple Object Access Protocol

### 80 **SSL**

81 Secure Sockets Layer

### 82 **TLS**

83 Transport Layer Security

### 84 **UDDI**

85 Universal Description, Discovery, and Integration

### 86 **URI**

87 Uniform Resource Identifier

### 88 **VPN**

89 Virtual Private Network

### 90 **WSDL**

91 Web Services Description Language

### 92 **WSS**

93 Web Services Security

94

95 **XML**  
96 eXtensible Markup Language  
97  
98  
99 **CBEFF**  
100 Common Biometric Exchange Formats Framework - data elements and BIR formats specified in  
101 ISO/IEC 19785-1  
102 **BIAS implementation**  
103 software entity that is capable of creating, processing, sending, and receiving BIAS messages  
104 **BIAS endpoint**  
105 runtime entity, identified by an endpoint URI/IRI, capable of sending and receiving BIAS  
106 messages, and containing a running BIAS implementation  
107 **BIAS message**  
108 message that can be sent from a BIAS endpoint to another BIAS endpoint through a BIAS link  
109 channel  
110 **BIAS request message**  
111 BIAS message conveying a request for an action to be performed by the receiving BIAS endpoint  
112 **BIAS response message**  
113 BIAS message conveying a response to a prior BIAS requestmessage

## 114 1.6 References

### 115 1.6.1 Normative References

116 **[RFC2119]** S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, IETF RFC 2119,  
117 March 1997  
118 <http://www.ietf.org/rfc/rfc2119.txt>  
119  
120 **[CBEFF]** ISO/IEC19785-1:2006, Information technology – Common Biometric Exchange Formats  
121 Framework – Part 1: Data element specification, with Amendment 1:2010  
122 <http://www.iso.org>  
123  
124 **[DATE-TIME]** ISO 8601:2004, Data elements and interchange formats — Information interchange —  
125 Representation of dates and times  
126 <http://www.iso.org>  
127  
128 **[INCITS-BIAS]** ANSI INCITS 442-2010, Biometric Identity Assurance Services (BIAS), July 2010  
129 <http://www.incits.org>  
130  
131 **[IRI]** M. Duerst, et al, Internationalized Resource Identifiers, RFC3987, January 2005  
132 <http://www.ietf.org/rfc/rfc3987.txt>  
133 **[SOAP11]** Simple Object Access Protocol (SOAP) 1.1, 8 May 2000  
134 <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>  
135  
136 **[URI]** T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifiers (URI): Generic Syntax*,  
137 RFC 3986, MIT/LCS, U.C. Irvine, Xerox Corporation, January 2005.  
<http://ietf.org/rfc/rfc3986>  
138  
139 **[UTF-8]** ISO/IEC 10646:2003, Information technology — Universal Multiple-Octet Coded Character  
140 Set (UCS)  
<http://www.iso.org>

- 141 **[WS-Addr]** W3C Recommendation, Web Services Addressing 1.0 - Core, and Web Services  
142 Addressing 1.0 - SOAP Binding, 9 May 2006  
143 <http://www.w3.org/2002/ws/addr/>
- 144 **[WS-I-Basic]** Basic Profile Version 1.1, 10 April 2006  
145 <http://www.ws-i.org/Profiles/BasicProfile-1.1-2006-04-10.html>
- 146 **[WS-I-Bind]** Web Services-Interoperability Organization (WS-I) Simple SOAP Binding Profile Version  
147 1.0, 24 August 2004  
148 <http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0-2004-08-24.html>
- 149 **[WSDL11]** Web Services Description Language (WSDL) 1.1, 15 March 2001  
150 <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>
- 151 **[XML 10]** Extensible Markup Language (XML) 1.0, 16 August 2006  
152 <http://www.w3.org/TR/2006/REC-xml-20060816/>
- 153 **[XOP]** XML-binary Optimized Packaging, W3C Recommendation, 25 January 2005  
154 <http://www.w3.org/TR/2005/REC-xop10-20050125/>

## 155 **1.6.2 Non-Normative References**

- 156 **[BioAPI]** ISO/IEC 19784-1:2006, Information technology – Biometric Application Programming  
157 Interface – Part 1: BioAPI Specification  
158 <http://www.iso.org>
- 159 **[CBEFF-3]** ISO/IEC 19785-3:2007, Information technology – Common Biometric Exchange Formats  
160 Framework – Part 3: Patron format specifications, with Amendment 1:2010  
161 <http://www.iso.org>  
162
- 163 **[BIO SEC]** ISO 19092 Financial services -- Biometrics -- Security framework  
164 <http://www.iso.org>
- 165 **[EBTS-DOD]** Department of Defense Electronic Biometric Transmission Specification, Version 2.0,  
166 27 March 2009  
167 <http://www.biometrics.dod.mil/CurrentInitiatives/Standards/dodebts.aspx>
- 168 **[EBTS-FBI]** IAFIS-DOC-01078-8.1, “Electronic Biometric Transmission Specification (EBTS)”,  
169 Version 8.1, November 19, 2008, Federal Bureau of Investigation, Criminal Justice  
170 Information Services Division  
171 <https://www.fbibiospecs.org>
- 172 **[EFTS]** IAFIS-DOC-01078-7, “Electronic Fingerprint Transmission Specification (EFTS)”, Version  
173 7.1, May 2, 2005, Federal Bureau of Investigation, Criminal Justice Information Services  
174 Division  
175 <https://www.fbibiospecs.org>
- 176 **[HR-XML]** HR-XML Consortium Library, 2007 April 15  
177 <http://www.hr-xml.org>
- 178 **[INT-I]** Interpol Implementation of ANSI/NIST ITL1-2000, Ver 4.22b, October 28, 2005, The Interpol  
179 AFIS Expert Group  
180 <http://www.interpol.int>
- 181 **[NIEM]** National Information Exchange Model (NIEM), Ver 2.0, June 2007, US DOJ/DHS  
182 <http://www.niem.gov>
- 183 **[RFC2246]** T. Dierks & C. Allen, *The TLS Protocol, Version 1.0*, January 1999  
184 <http://www.ietf.org/rfc/rfc2246.txt>

- 185 **[RFC2617]** J. Franks, et al, *HTTP Authentication: Basic and Digest Access Authentication*, June  
186 1999  
187 <http://www.ietf.org/rfc/rfc2617.txt>
- 188 **[RFC3280]** R. Housley, et al, Internet X.509 Public Key Infrastructure Certificate and Certificate  
189 Revocation List (CRL) Profile, April 2002  
190 <http://www.ietf.org/rfc/rfc3280.txt>
- 191 **[SAML]** Security Assertion Markup Language (SAML), Oasis Standard, March 2005  
192 <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- 193 **[SAML SEC]** Security and Privacy Considerations for the OASIS Security Assertion Markup  
194 Language (SAML) V2.0, Oasis Standard, 15 March 2005  
195 <http://docs.oasis-open.org/security/saml/v2.0/saml-sec-consider-2.0-os.pdf>
- 196 **[SSL3]** SSL 3.0 Specification  
197 <http://www.freesoft.org/CIE/Topics/ssl-draft/3-SPEC.HTM>
- 198 **[WSS]** Web Services Security: SOAP Message Security 1.1, (WS-Security 2004), OASIS Standard  
199 Specification, 1 February 2006  
200 [http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-](http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf)  
201 [SOAPMessageSecurity.pdf](http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf)
- 202 **[X509]** X.509: Information technology - Open Systems Interconnection - The Directory: Public-key  
203 and attribute certificate frameworks, ITU-T, August 2005  
204 <http://www.itu.int/rec/T-REC-X.509-200508-I>
- 205 **[xNAL]** Customer Information Quality Specifications Version 3.0: Name (xNL), Address  
206 (xAL), Name and Address (xNAL) and Party (xPIL), Committee Specification 02,  
207 20 September 2008  
208 [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=ciq](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ciq)

## 2 Design Concepts and Architecture (non-normative)

### 2.1 Philosophy

Rather than define a totally new and unique messaging protocol for biometric services, this specification instead defines a method for using existing biometric and Web services standards to exchange biometric data and perform biometric operations.

### 2.2 Context

Today, biometric systems are being developed which collect, process, store and match biometric data for a variety of purposes. In many cases, data and/or capabilities need to be shared between systems or systems serve a number of different client stakeholders. As architectures move towards services-based frameworks, access to these biometric databases and services is via a Web services front-end. However, lack of standardization in this area has led implementers to develop customized services for each system/application.

BIAS is intended to provide a common, yet flexible, Web services interface that can be used within both closed and open SOA systems. Figure 1, below, depicts the context in which the BIAS messages will be implemented.

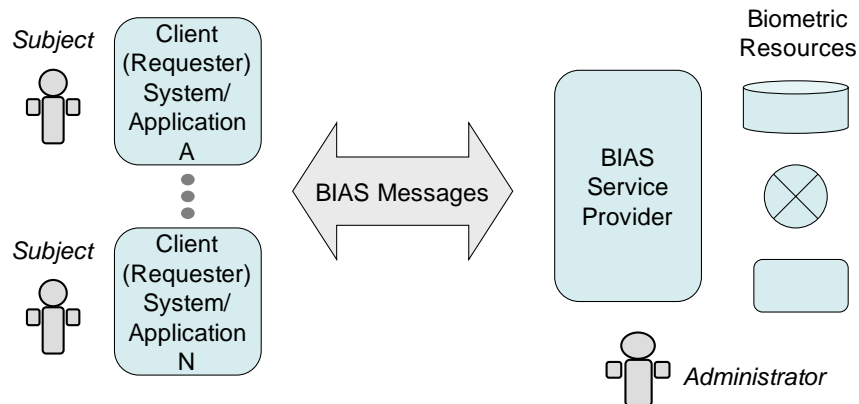


Figure 1. BIAS Context

The clients (requesters) may use standard discovery mechanisms (i.e., UDDI directories) to discover the BIAS service provider (implementation) or, particularly in closed systems, the URI/IRI and WSDL for the service provider may be known *a priori* by the client BIAS application developer.

### 2.3 Architecture

BIAS Web services are intended to be used within systems employing a services framework, such as a services-oriented architecture (SOA) (although implementations are not limited to this environment). As such, it is recognized that the clients may interact directly with the BIAS service provider or layers may exist between the client and the service provider, for example as an ESB or other application layer.

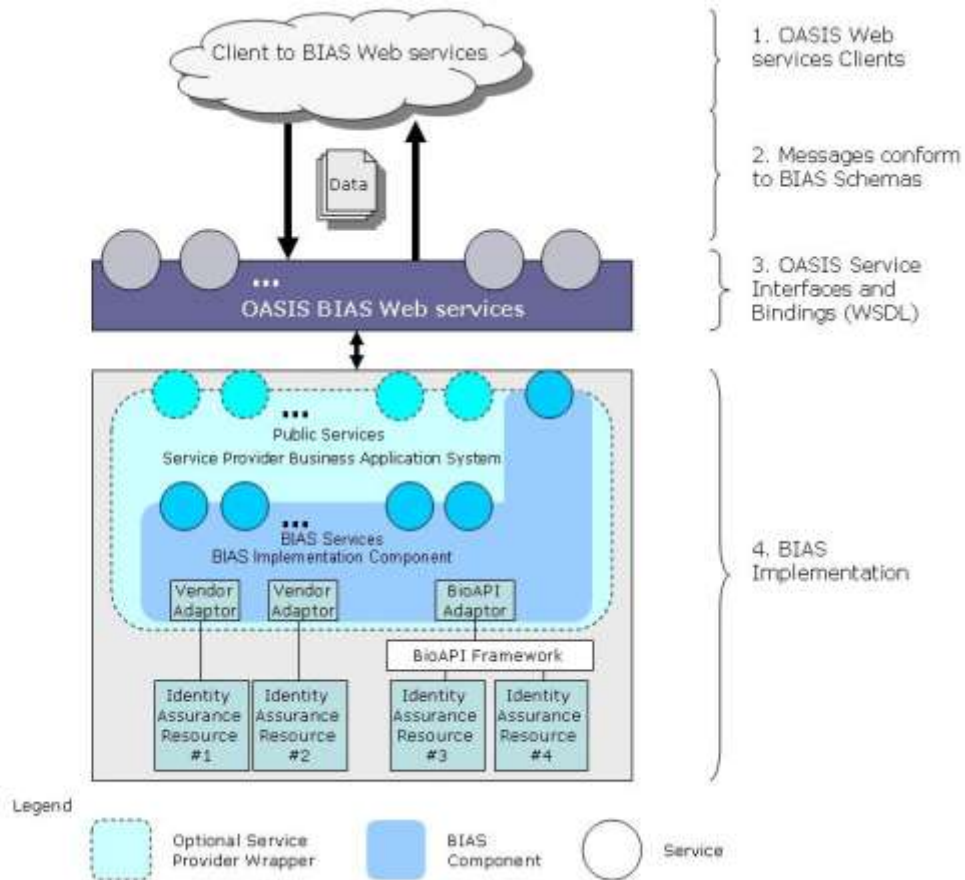
The BIAS Architecture as shown in Figure 2, in which:

- A Client request to the BIAS Web services may be triggered by a human interaction OR any proxy system such as an ESB.

- 241 • Client sends and receives SOAP messages that conform to the BIAS schemas
- 242 • Calls to the BIAS Implementation use OASIS Service Interfaces and Bindings (via WSDL)
- 243 • The BIAS implementation maps the service call to the appropriate internal API or set of APIs
- 244 and returns data according to the service interface.

245 Note that services are represented as circles.

246



247

248

249 **Figure 2. Representative BIAS Architecture**

250

251 NOTE: It is possible that BIAS may also be used between the service provider and the managed  
 252 resource (e.g., a biometric matcher).

253

254 At the heart of the BIAS SOAP Profile are the concepts of BIAS messages and endpoints.

255

256 **BIAS implementation**

257 A BIAS implementation is a software entity that is capable of creating, processing, sending, and receiving  
 258 BIAS messages. This standard does not define requirements for the BIAS implementation other than  
 259 defining the messages and protocols used by the endpoints.

260 **BIAS messages**

261 A BIAS message is a one that can be sent from a BIAS endpoint to another BIAS endpoint over a TCP/IP  
 262 link.

263 **BIAS endpoints**

264 A BIAS endpoint is a runtime entity, uniquely identified and accessed by an endpoint URI/IRI [URI] [IRI],  
265 capable of sending and receiving BIAS messages.

266 NOTE: When not publicly and directly exposed, the endpoints for purposes of this specification are  
267 the BIAS service provider exposing BIAS services and the component that directly interacts with that  
268 service provider, e.g., the business application or ESB, rather than the ultimate end client requester.

## 269 3 Data dictionary

270 This section describes the BIAS data elements used within BIAS messages (as defined in Clause 4).  
 271 Common data elements are defined for use in one or more operations. These include common data types  
 272 or return codes. BIAS data elements are defined in ANSI INCITS 442-2010. The elements, complex types  
 273 and simple types described for the BIAS messages belong to the following namespace: [http://docs.oasis-](http://docs.oasis-open.org/bias/ns/bias-1.0/)  
 274 [open.org/bias/ns/bias-1.0/](http://docs.oasis-open.org/bias/ns/bias-1.0/). See Annex A for the XML schema.

275 NOTE: Biographic and biometric data included in a native XML format MAY contain elements  
 276 referencing external namespaces (e.g., ansi-nist).

### 277 3.1 Documentation Conventions

278 Each common element has a section describing its content. Likewise, each operation has a section  
 279 describing the request and response messages and the associated input and output parameters. The  
 280 input and output of each message and the comment elements are detailed in a table as described in the  
 281 figure below. Each field that forms part of the message request/response is detailed in the table.

282

| Header Name | Description                                  | Values | Value Meaning   |
|-------------|--|--------|---|
| Field       | The name of the field.                       |        |   |
| Type        | The XML schema type of the field.            |        |   |
| #           | The cardinality of the field                 | 1      | One occurrence  |
|             |  | 0..1   | Zero or one occurrence  |
|             |  | 0..*   | Zero or more occurrences  |
|             |  | 1..*   | One or more occurrences   |
| ?           | Defines if the field must be present.        | Y      | Yes – is always required  |
|             |  | N      | No – is not always required, an optional field.                         |
|             |  | C      | Conditional – requirement is dependent on system or message conditions. |
| Meaning     | Gives a short description of the field's use |        |   |

283

**Figure 3. BIAS Message Input/Output Dictionary Table Headings**

#### 284 **Fields Hierarchy Explained:**

285 To denote the field hierarchy the symbol  $\uparrow$  is used to denote the child-of relationship.

286 All string types/elements MUST consist of ISO/IEC 10646 (Unicode) characters encoded in UTF-8 [UTF-  
 287 8] (see ISO/IEC 10646:2003, Annex D).



288 **3.2 Common Elements**

289 **3.2.1 ApplicationIdentifier**

|                     |                            |
|---------------------|----------------------------|
| <b>Type:</b>        | string                     |
| <b>Description:</b> | Identifies an application. |
| <b>Min Length:</b>  | 1                          |
| <b>Max Length:</b>  | 255                        |

290 **3.2.2 ApplicationUserIdentifier**

|                     |   |
|---------------------|---|
| <b>Type:</b>        | string                                      |
| <b>Description:</b> | Identifies an application user or instance. |
| <b>Min Length:</b>  | 1   |
| <b>Max Length:</b>  | 255   |

291 **3.2.3 BaseBIRType**

|                     |  |
|---------------------|--|
| <b>Type:</b>        | Schema complexType   |
| <b>Description:</b> | Base type for all BIR subtypes; see <a href="#">BinaryBIR</a> , <a href="#">URI_BIR</a> , and <a href="#">XML_BIR</a> for currently available types. |

292

293 **3.2.4 BIASBiometricDataType**

| Field                 | Type                                  | #    | ? | Meaning   |
|-----------------------|---------------------------------------|------|---|---|
| BIASBiometricDataType |                                       |      | Y | Wraps the various BIAS biometric types.<br>The operations that use this type specify which elements are required. |
| ↑ BIRList             | <a href="#">CBEFF_BIR_ListType</a>    | 0..1 | N | A list of CBEFF-BIR elements.   |
| ↑ BIR                 | <a href="#">CBEFF_BIR_Type</a>        | 0..1 | N | Contains biometric information in either a non-XML or an XML representation.                                      |
| ↑ InputBIR            | <a href="#">CBEFF_BIR_Type</a>        | 0..1 | N | Maps to specific INCITS BIAS elements as required by that specification.  |
| ↑ ReferenceBIR        | <a href="#">CBEFF_BIR_Type</a>        | 0..1 | N | Maps to specific INCITS BIAS elements as required by that specification.  |
| ↑ BiometricDataList   | <a href="#">BiometricDataListType</a> | 0..1 | N | A list of biometric data elements.  |

294 **3.2.5 BIASFaultCode**

|                     |  |
|---------------------|--|
| <b>Type:</b>        | String                                 |
| <b>Description:</b> | Error code referenced in a SOAP fault. |

295 **BIASFaultCode Enumeration Values**

| Value                     | Description  |
|---------------------------|--|
| UNKNOWN_ERROR             | The service failed for an unknown reason.                                |
| UNSUPPORTED_CAPABILITY    | A requested capability is not supported by the service implementation.   |
| INVALID_INPUT             | The data in a service input parameter is invalid.                        |
| BIR_QUALITY_ERROR         | Biometric sample quality is too poor for the service to succeed.         |
| INVALID_BIR               | The input BIR is empty or in an invalid or unrecognized format.          |
| BIR_SIGNATURE_FAILURE     | The service could not validate the signature, if used, on the input BIR. |
| BIR_DECRYPTION_FAILURE    | The service could not decrypt an encrypted input BIR.                    |
| INVALID_ENCOUNTER_ID      | The input encounter ID is empty or in an invalid format.                 |
| INVALID_SUBJECT_ID        | The input subject ID is empty or in an invalid format.                   |
| UNKNOWN_SUBJECT           | The subject referenced by the input subject ID does not exist.           |
| UNKNOWN_GALLERY           | The gallery referenced by the input gallery ID does not exist.           |
| UNKNOWN_ENCOUNTER         | The encounter referenced by the input encounter ID does not exist.       |
| UNKNOWN_BIOGRAPHIC_FORMAT | The biographic data format is not known or not supported.                |
| UNKNOWN_IDENTITY_CLAIM    | The identity referenced by the input identity claim does not exist.      |
| INVALID_IDENTITY_CLAIM    | The identity claim requested is already in use.                          |
| NONEXISTANT_DATA          | The data requested for deletion does not exist.                          |

296 NOTES:

297 (1) See Clause 6 (Error handling) for an explanation of BIAS faults and return codes.

298 (2) Service provider MAY define additional values specific to their service implementation.

299 (3) See section 5.5 for additional information on BIAS security.

300 **3.2.6 BIASFaultDetail**

| Field              | Type                          | # | ? | Meaning   |
|--------------------|-------------------------------|---|---|---|
| BIASFaultDetail    |                               |   | Y | Defines the error information associated with a SOAP fault. |
| ↑ BIASFaultType    | <a href="#">BIASFaultCode</a> | 1 | Y | References an error code.                                   |
| ↑ BIASFaultMessage | string                        | 1 | Y | Provides a brief explanation of the fault.                  |

| Field                      | Type   | #    | ? | Meaning  |
|----------------------------|--------|------|---|--|
| n<br>↑ BIASFaultDescriptio | string | 0..1 | N | Provides detailed information about a BIAS fault, such as trace details. |

### 301 3.2.7 BIASIdentity

| Field                        | Type                                  | #    | ? | Meaning  |
|------------------------------|---------------------------------------|------|---|--|
| BIASIdentity                 |                                       |      | Y | Defines a single element for encapsulating the data associated with an Identity. Includes the Identity's reference identifiers, biographic data, and biometric data.<br><br>The operations that use this type specify which elements are required. |
| ↑ SubjectID                  | <a href="#">BIASIDType</a>            | 0..1 | C | A system unique identifier for a subject.<br><br>Required as input to many operations.   |
| ↑ IdentityClaim              | <a href="#">BIASIDType</a>            | 0..1 | N | An identifier by which a subject is known to a particular gallery or population group.   |
| ↑ EncounterID                | <a href="#">BIASIDType</a>            | 0..1 | C | The identifier of an encounter associated with the subject.<br><br>Required for encounter-centric models.  |
| ↑ EncounterList              | <a href="#">EncounterListType</a>     | 0..1 | N | A list of encounters associated with a subject.  |
| ↑ BiographicData             | <a href="#">BiographicDataType</a>    | 0..1 | N | An Identity's biographic data.   |
| s<br>↑ BiographicDataElement | <a href="#">BiographicDataType</a>    | 0..1 | N | An Identity's biographic data elements that are stored in the implementing system.   |
| ↑ BiometricData              | <a href="#">BIASBiometricDataType</a> | 0..1 | N | An Identity's biometric data.  |

### 302 3.2.8 BIASIDType

|                     |                    |
|---------------------|--------------------|
| <b>Type:</b>        | string             |
| <b>Description:</b> | A BIAS Identifier. |

### 303 3.2.9 BinaryBIR

| Field     | Type        | # | ? | Meaning                      |
|-----------|-------------|---|---|------------------------------|
| BinaryBIR | BaseBIRType |   | Y | Defines a BIR type of Binary |

| Field    | Type         | # | ? | Meaning                                 |
|----------|--------------|---|---|---|
| ↑ Binary | base64Binary | 1 | Y | BIR information in base64 binary format |

304

### 3.2.10 BiographicDataItem

305

| Field              | Type   | #    | ? | Meaning   |
|--------------------|--------|------|---|---|
| BiographicDataItem |        |      | Y | Defines a single biographic data element.       |
| ↑ Name             | string | 1    | Y | The name of the biographic data item.           |
| ↑ Type             | string | 1    | Y | The data type for the biographic data item.     |
| ↑ Value            | string | 0..1 | N | The value assigned to the biographic data item. |

306

307

NOTE: This element can be used to transmit scanned identity documents or document information (e.g., passports, driver's license, birth certificates, utility bills, etc. required to establish an identity).

### 3.2.11 BiographicDataSet

308

| Field             | Type   | #    | ? | Meaning   |
|-------------------|--------|------|---|---|
| BiographicDataSet |        |      | Y | Defines a set of biographic data that is formatted according to the specified format.   |
| ↑ name            | string | 1    | Y | The name of the biographic data format. Use these names for common formats: FBI-EFTS [EFTS], FBI-EBTS [EBTS-FBI], DOD-EBTS [EBTS-DOD], INT-I [INT-I], NIEM [NIEM], xNAL [xNAL], HR-XML [HR-XML].  |
| ↑ version         | string | 0..1 | N | The version of the biographic data format (e.g., "7.1" for FBI-EFTS or "2.0" for NIEM).   |
| ↑ source          | string | 1    | Y | Reference to a URI/IRI describing the biographic data format. For example: (FBI-EFTS and FBI-EBTS) www.fbibiospecs.org, (DOD-EBTS) www.biometrics.dod.mil, (INT-I) www.interpol.int, (NIEM) www.niem.gov, (xNAL) www.oasis-open.org, (HR-XML) www.hr-xml.org. |
| ↑ type            | string | 1    | Y | The biographic data format type. Use these types for common formats: ASCII (e.g., for non-XML versions of FBI-EFTS, FBI-EBTS, DOD-EBTS, or INT-I), XML (e.g., for NIEM, xNAL, and HR-XML or future versions of FBI-EBTS).                                     |
| ↑ unspecified     | any    | 0..* | N | Biographic data formatted according to a specific format.   |

309

310

311

NOTE: Biographic data formats are not limited to those listed. The string value is not enumerated. If one of the common types are used, it MUST be indicated by the specified name values; however, the service provider MAY offer other formats. See INCITS 442 for further information.

312 **3.2.12 BiographicDataType**

| Field                 | Type                               | #    | ? | Meaning  |
|-----------------------|------------------------------------|------|---|--|
| BiographicDataType    |                                    |      | Y | Defines a set of biographic data elements, utilizing either the <a href="#">BiographicDataItem</a> to represent a list of elements or the <a href="#">BiographicDataSet</a> to represent a complete, formatted set of biographic information. One of the following elements must be present. |
| ↳ LastName            | string                             | 0..1 | N | The last name of a subject.  |
| ↳ FirstName           | string                             | 0..1 | N | The first name of a subject.   |
| ↳ BiographicDataItems | <a href="#">BiographicDataItem</a> | 0..1 | N | A list of biographic data elements.  |
| ↳ BiographicDataItem  | <a href="#">BiographicDataItem</a> | 1..* | N | A single biographic data element.  |
| ↳ BiographicDataSet   | <a href="#">BiographicDataSet</a>  | 0..1 | N | A set of biographic data information.  |

- 313 NOTE: The implementer is given three choices for encoding biographic data:
- 314
- 315 • Encode only first and last name using the defined fields within BiographicDataType
  - 316 • Define a list of biographic data elements using the BiographicDataItem
  - 317 • Use a pre-defined set of biographic data (e.g., as specified in another standard) using the BiographicDataSet.
- 318 See also INCITS 442, section 8.1 for further information.

319 **3.2.13 BiometricDataElement**

| Field                | Type                          | #    | ? | Meaning  |
|----------------------|-------------------------------|------|---|--|
| BiometricDataElement |                               |      | Y | Provides descriptive information about biometric data, such as the biometric type, subtype, and format, contained in the BDB of the CBEFF-BIR. |
| ↳ BiometricType      | oasis_cbeff:MultipleTypesType | 1    | Y | The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.   |
| ↳ BiometricTypeCount | positiveInteger               | 0..1 | N | The number of biometric records having the biometric type recorded in the biometric type field.  |

| Field              | Type                    | #    | ? | Meaning  |
|--------------------|-------------------------|------|---|--|
| ↑ BiometricSubType | oasis_cbeff:SubtypeType | 0..1 | N | More specifically defines the type of biometric data stored in the biometric record, as defined by CBEFF.  |
| ↑ BDBFormatOwner   | positiveInteger         | 1    | Y | Identifies the standards body, working group, industry consortium, or other CBEFF biometric organization that has defined the format for the biometric data. |
| ↑ BDBFormatType    | positiveInteger         | 1    | Y | Identifies the specific biometric data format specified by the CBEFF biometric organization recorded in the BDB Format Owner field.                          |

320

### 321 3.2.14 BiometricDataListType

| Field                  | Type                                   | #    | ? | Meaning   |
|------------------------|--|------|---|---|
| BiometricDataListType  |  |      | Y | A list of biometric data elements.                              |
| ↑ BiometricDataElement | <b>3.2.13 BiometricDataElementType</b> | 0..* | N | Data structure containing information about a biometric record. |

### 322 3.2.15 CandidateListResultType

| Field                   | Type                            | # | ? | Meaning  |
|-------------------------|---------------------------------|---|---|--|
| CandidateListResultType |                                 |   | Y | Defines a set of candidates, utilizing the <a href="#">CandidateType</a> to represent each element in the set. |
| ↑ CandidateList         | <b>3.2.16 CandidateListType</b> | 1 | Y | The candidate list.  |

323

### 324 3.2.16 CandidateListType

| Field             | Type                          | #    | ? | Meaning  |
|-------------------|-------------------------------|------|---|--|
| CandidateListType |                               |      | Y | Defines a set of candidates, utilizing the <a href="#">CandidateType</a> to represent each element in the set. |
| ↑ Candidate       | <a href="#">CandidateType</a> | 0..* | N | A single candidate.  |

325 **3.2.17 CandidateType**

| Field            | Type                               | #    | ? | Meaning  |
|------------------|------------------------------------|------|---|--|
| CandidateType    |                                    |      | Y | Defines a single candidate as a possible match in response to a biometric identification request.          |
| ↑ Score          | <a href="#">Score</a>              | 0..1 | N | The match score.   |
| ↑ Rank           | integer                            | 1    | Y | The rank of the candidate in relation to other candidates for the same biometric identification operation. |
| ↑ BiographicData | <a href="#">BiographicDataType</a> | 0..1 | N | Biographic data associated with the candidate match.   |
| ↑ BIRList        | <a href="#">CBEFF_BIR_ListType</a> | 1    | Y | Biometric data associated with the candidate match.  |

326 **3.2.18 CapabilityListType**

| Field              | Type                           | #    | ? | Meaning                        |
|--------------------|--------------------------------|------|---|--------------------------------|
| CapabilityListType |                                |      | Y | Defines a set of capabilities. |
| ↑ Capability       | <a href="#">CapabilityType</a> | 0..* | N | A single capability.           |

327 **3.2.19 CapabilityName**

|                     |                             |
|---------------------|-----------------------------|
| <b>Type:</b>        | string                      |
| <b>Description:</b> | A list of capability items. |

328 **CapabilityName Enumeration Values**

| Value                       | Description  |
|-----------------------------|--|
| AggregateInputDataOptional  | A data element accepted as optional input by the implementing system for the aggregate services. |
| AggregateInputDataRequired  | A data element required as input by the implementing system for the aggregate services.          |
| AggregateProcessingOption   | A processing option supported by the implementing system for the aggregate services.             |
| AggregateReturnData         | A data element returned by the implementing system for the aggregate services.                   |
| AggregateServiceDescription | Describes the processing logic of an aggregate service supported by the implementing system.     |
| BiographicDataSet           | Identifies a biographic data set supported by the implementing system.                           |
| CBEFFPatronFormat           | A patron format supported by the implementing system.  |
| ClassificationAlgorithmType | A classification algorithm type supported by the implementing system.                            |
| ConformanceClass            | Identifies the conformance class of the BIAS implementation.                                     |

| Value              | Description  |
|--------------------|--|
| Gallery            | A gallery or population group supported by the implementing system.                              |
| IdentityModel      | Identifies whether the implementing system is person-centric or encounter-centric based.         |
| MatchScore         | Identifies the use of match scores returned by the implementing system.                          |
| QualityAlgorithm   | A quality algorithm vendor and algorithm vendor product ID supported by the implementing system. |
| SupportedBiometric | A biometric type supported by the implementing system.   |
| TransformOperation | A transform operation type supported by the implementing system.                                 |

### 329 3.2.20 CapabilityType

| Field                       | Type                           | #    | ? | Meaning  |
|-----------------------------|--------------------------------|------|---|--|
| CapabilityType              |                                |      | Y | Defines a single capability supported by an implementing system.     |
| ↑ CapabilityName            | <a href="#">CapabilityName</a> | 1    | Y | The name of the capability.  |
| ↑ CapabilityID              | string                         | 0..1 | N | An identifier assigned to the capability by the implementing system. |
| ↑ CapabilityDescription     | string                         | 0..1 | N | A description of the capability.                                     |
| ↑ CapabilityValue           | string                         | 0..1 | N | A value assigned to the capability.                                  |
| ↑ CapabilitySupportingValue | string                         | 0..1 | N | A secondary value supporting the capability.                         |
| ↑ CapabilityAdditionalInfo  | string                         | 0..1 | N | Contains additional information for the supported capability.        |

### 330 3.2.21 CBEFF\_BIR\_ListType

| Field              | Type                           | #    | ? | Meaning  |
|--------------------|--------------------------------|------|---|--|
| CBEFF_BIR_ListType |                                |      | Y | A list of CBEFF-BIR elements.                                    |
| ↑ BIR              | <a href="#">CBEFF_BIR_Type</a> | 0..* | N | CBEFF structure containing information about a biometric sample. |

### 331 3.2.22 CBEFF\_BIR\_Type

| Field          | Type            | # | ? | Meaning  |
|----------------|-----------------|---|---|--|
| CBEFF_BIR_Type |                 |   | Y | Represents biometric information, with either a non-XML or XML representation. |
| ↑ FormatOwner  | positiveInteger | 1 | Y | Identifies the Patron format owner.  |
| ↑ FormatType   | positiveInteger | 1 | Y | Identifies the Patron format type.   |



| Field             | Type                    | #    | ? | Meaning  |
|-------------------|-------------------------|------|---|--|
| ↑ BIR_Information |                         | 0..1 | N | Describes what is contained in a BIR.  |
| ↑ BIR_Info        | oasis_cbeff:BIRInfoType | 0..1 | N | Contains information about the CBEFF-BIR.  |
| ↑ BDB_Info        | oasis_cbeff:BDBInfoType | 0..1 | N | Contains information about the BDB in a simple CBEFF-BIR.                          |
| ↑ SB_Info         | oasis_cbeff:SBInfoType  | 0..1 | N | Contains information about the security block, if used, in a simple CBEFF-BIR.     |
| ↑ BIR             | BaseBIRType             | 1    | Y | One of the following sub-elements must be present: BinaryBIR, URI_BIR, or XML_BIR. |

332 NOTE: The implementer is given three choices for encoding a BIR:

- 333
- As an XML BIR (following the XML Patron format as specified in Annex B)
  - 334 • As a reference to a URI (from which the receiver would retrieve the actual BIR)
  - 335 • As a complete Base64 encoded binary (non-XML) BIR.

336 The latter two alternatives can use any CBEFF Patron Format. The optional BIR\_Information provides a  
 337 mechanism for exposing metadata associated with a BIR format that is not easily decoded (i.e., a non-  
 338 XML BIR). See section 5.3 for more information on handling of binary data within BIAS and INCITS 442,  
 339 Clause 8.2, for more information on representing biometric data.

340 NOTE:

- 341 (1) XML BIRs MUST conform to the XML patron format in Annex B; however, non-XML (binary)  
 342 and URI BIRs MAY implement any CBEFF patron format.
- 343 (2) It is RECOMMENDED that only registered CBEFF patron formats be used; however, in closed  
 344 systems, this may not be required.

### 345 3.2.23 Classification

|                     |                                 |
|---------------------|---------------------------------|
| <b>Type:</b>        | string                          |
| <b>Description:</b> | The result of a classification. |

### 346 3.2.24 ClassificationAlgorithmType

|                     |   |
|---------------------|---|
| <b>Type:</b>        | string  |
| <b>Description:</b> | Type of classification algorithm that was used to perform the classification. |

### 347 3.2.25 ClassificationData

| Field              | Type | # | ? | Meaning  |
|--------------------|------|---|---|--|
| ClassificationData |      |   | Y | Contains information on classification results and the algorithm used to determine the classification. |

| Field                           | Type  | # | ? | Meaning  |
|---------------------------------|---|---|---|--|
| ↑ Classification                | <a href="#">Classification</a>              | 1 | Y | The result of the classification.  |
| e ↑ ClassificationAlgorithmType | <a href="#">ClassificationAlgorithmType</a> | 1 | Y | Identifies the type of classification algorithm that was used to perform the classification. |

### 348 3.2.26 EncounterListType

| Field             | Type                       | #    | ? | Meaning                         |
|-------------------|----------------------------|------|---|---------------------------------|
| EncounterListType |                            |      | Y | Defines a set of encounters.    |
| ↑ EncounterID     | <a href="#">BIASIDType</a> | 0..* | N | The identifier of an encounter. |

### 349 3.2.27 FusionDecision

|                     |   |
|---------------------|---|
| <b>Type:</b>        | string  |
| <b>Description:</b> | The match decision assigned by the matching algorithm |

350

### 351 3.2.28 FusionInformationListType

| Field                     | Type                                  | #    | ? | Meaning   |
|---------------------------|---------------------------------------|------|---|---|
| FusionInformationListType |                                       |      | Y | Contains at a minimum two sets of fusion input elements, as input to the <a href="#">PerformFusion</a> operation. |
| ↑ FusionElement           | <a href="#">FusionInformationType</a> | 2..* | Y | A set of fusion information.  |

### 352 3.2.29 FusionInformationType

| Field                 | Type                          | #    | ? | Meaning  |
|-----------------------|-------------------------------|------|---|--|
| FusionInformationType |                               |      | Y | Represents the information necessary to perform a fusion operation.  |
| ↑ BiometricType       | oasis_cbeff:MultipleTypesType | 1    | Y | The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.                           |
| e ↑ BiometricSubType  | oasis_cbeff:SubtypeType       | 0..1 | N | More specifically defines the type of biometric data stored in the biometric record.                                     |
| ↑ AlgorithmOwner      | string                        | 1    | Y | The owner or vendor of the algorithm used to determine the score or decision.  |
| ↑ AlgorithmType       | string                        | 1    | Y | The Algorithm Owner's identifier for the specific algorithm product and version used to determine the score or decision. |

| Field          | Type                         | #    | ? | Meaning  |
|----------------|------------------------------|------|---|--|
| ↑ FusionResult | <a href="#">FusionResult</a> | 0..1 | C | Either <a href="#">FusionScore</a> or a <a href="#">FusionDecision</a> element MUST be used. |

### 353 3.2.30 FusionResult

|                     |   |
|---------------------|---|
| <b>Type:</b>        | complexType   |
| <b>Description:</b> | The base type for any resulting types which indicate the status of a Fusion operation |

### 354 3.2.31 FusionScore

|                     |  |
|---------------------|--|
| <b>Type:</b>        | <a href="#">Score</a>                                    |
| <b>Description:</b> | The similarity score assigned by the matching algorithm. |

355

### 356 3.2.32 GenericRequestParameters

| Field                    | Type                                      | #    | ? | Meaning   |
|--------------------------|---|------|---|---|
| GenericRequestParameters |   |      | Y | Common request parameters that can be used to identify the requester. |
| ↑ Application            | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                |
| ↑ ApplicationUser        | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.        |
| ↑ BIASOperationName      | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested.                |

357 NOTE: See section 5.4 for alternatives for identifying the requested BIAS operation in a BIAS  
358 SOAP message.

### 359 3.2.33 IdentifySubjectResultType

|                     |   |
|---------------------|---|
| <b>Description:</b> | A base type for all types that could be returned from the IdentifySubject operation |
|---------------------|---|

360

### 361 3.2.34 InformationType

| Field           | Type | #    | ? | Meaning  |
|-----------------|------|------|---|--|
| InformationType |      |      | Y | Allows for an unlimited number of data element types, and it does not specify nor require any particular data element. |
| ↑ unspecified   | any  | 0..* | N |  |

362 **3.2.35 ListFilterType**

| Field                          | Type                          | #    | ? | Meaning   |
|--------------------------------|-------------------------------|------|---|---|
| ListFilterType                 |                               |      | Y | Provides a method to filter the amount of information returned in a search of biometric data. |
| ↑ BiometricTypeFilters         |                               | 1    | Y |   |
| ↑ BiometricTypeFilter          | oasis_cbeff:MultipleTypesType | 1..* | Y | Limits the returned information to a specific type of biometric, as defined by CBEFF.         |
| e    ↑ IncludeBiometricSubType | boolean                       | 1    | Y | A Boolean flag indicating if biometric subtype information should be returned.                |

363 **3.2.36 MatchType**

|                     |                                |
|---------------------|--------------------------------|
| <b>Type:</b>        | boolean                        |
| <b>Description:</b> | The result of a fusion method. |

364 **3.2.37 ProcessingOptionsType**

| Field                 | Type   | #    | ? | Meaning   |
|-----------------------|--------|------|---|---|
| ProcessingOptionsType |        |      | Y | BIAS aggregate operations support the ability to include various processing options which direct and possibly control the business logic for that operation. The ProcessingOptionsType provides a method to represent those options. Processing options SHOULD be defined by the implementing system. |
| ↑ Option              | string | 0..* | N | An option supported by the implementing system.   |

365 **3.2.38 ProductID**

|                     |   |
|---------------------|---|
| <b>Type:</b>        | string                                    |
| <b>Description:</b> | The vendor's ID for a particular product. |

366 **3.2.39 QualityData**

| Field             | Type                             | #    | ? | Meaning  |
|-------------------|----------------------------------|------|---|--|
| QualityData       |                                  |      | Y | Contains information about a biometric sample's quality and the algorithm used to compute the quality. |
| ↑ QualityScore    | oasis_cbeff:QualityType          | 0..1 | N | The quality of a biometric sample.   |
| ↑ AlgorithmVendor | <a href="#">VendorIdentifier</a> | 1    | Y | The vendor of the quality algorithm used to determine the quality score.                               |

| Field                    | Type                        | #    | ? | Meaning  |
|--------------------------|-----------------------------|------|---|--|
| AlgorithmVendorProductID | <a href="#">ProductID</a>   | 1    | Y | The vendor's ID for the algorithm used to determine the quality. |
| AlgorithmVersion         | <a href="#">VersionType</a> | 0..1 | N | The version of the algorithm used to determine the quality.      |

### 367 3.2.40 ResponseStatus

| Field          | Type                       | #    | ? | Meaning   |
|----------------|----------------------------|------|---|---|
| ResponseStatus |                            |      | Y |   |
| Return         | <a href="#">ReturnCode</a> | 1    | Y | The return code indicates the return status of the operation. |
| Message        | string                     | 0..1 | N | A short message corresponding to the return code.             |

### 368 3.2.41 ReturnCode

|                     |   |
|---------------------|---|
| <b>Type:</b>        | unsignedLong  |
| <b>Description:</b> | Return value specifying success or other condition. |

369

### 370 ReturnCode Enumeration Values

| Value | Description |
|-------|-------------|
| 0     | Success     |

### 371 3.2.42 Score

|                     |                                |
|---------------------|--------------------------------|
| <b>Type:</b>        | float                          |
| <b>Description:</b> | Match result or quality score. |

372 NOTE: Matching scores MAY be in a standardized or proprietary form in terms of value range and  
373 interpretation. Quality scores, however, follow the definition found in Annex B.

### 374 3.2.43 TokenResultType

| Field           | Type                      | # | ? | Meaning   |
|-----------------|---------------------------|---|---|---|
| TokenResultType |                           |   | Y | Defines a token that is returned for asynchronous processing. |
| TokenType       | <a href="#">TokenType</a> | 1 | Y | Defines a token that is returned for asynchronous processing. |

375

376 **3.2.44 TokenType**

| Field        | Type   | # | ? | Meaning  |
|--------------|--------|---|---|--|
| TokenType    |        |   | Y | Defines a token that is returned for asynchronous processing.  |
| ↑ TokenValue | string | 1 | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time.    |
| ↑ Expiration | date   | 1 | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

377 NOTE: Date/time format is defined in INCITS 442 and is consistent with the date format specified  
 378 in Annex B and ISO 8601 [DATE-TIME]. See also Annex A for schema definition.

379 **3.2.45 URI\_BIR**

| Field   | Type        | # | ? | Meaning                      |
|---------|-------------|---|---|------------------------------|
| URI_BIR | BaseBIRType |   | Y | Defines a BIR type of Binary |
| ↑ URI   | anyURI      | 1 | Y | The URI of the BIR           |

380

381 **3.2.46 VendorIdentifier**

|                     |                      |
|---------------------|----------------------|
| <b>Type:</b>        | string               |
| <b>Description:</b> | Identifies a vendor. |

382 NOTE: Vendor identifiers are registered with IBIA as the CBEFF registration authority (see  
 383 ISO/IEC 19785-2). Registered biometric organizations are listed at:  
 384 [http://www.ibia.org/cbeff/\\_biometric\\_org.php](http://www.ibia.org/cbeff/_biometric_org.php).

385 **3.2.47 Version**

| Field   | Type               | # | ? | Meaning  |
|---------|--------------------|---|---|--|
| Version |                    |   | Y | For a description or definition of each data element, see the referenced CBEFF standards in the <b>3.2.22 CBEFF_BIR_Type</b> schema. |
| ↑ major | nonNegativeInteger | 1 | Y |  |
| ↑ minor | nonNegativeInteger | 1 | Y |  |

386 **3.2.48 VersionType**

|                     |                             |
|---------------------|-----------------------------|
| <b>Type:</b>        | string                      |
| <b>Description:</b> | The version of a component. |

387 **3.2.49 XML\_BIR**

| Field   | Type        | # | ? | Meaning                      |
|---------|-------------|---|---|------------------------------|
| XML_BIR | BaseBIRType |   | Y | Defines a BIR type of Binary |

| Field | Type               | # | ? | Meaning                       |
|-------|--------------------|---|---|-------------------------------|
| ↑ XML | Oasis_cbeff:BIType | 1 | Y | BIR information in XML format |

388

## 389 4 BIAS Messages

390 This section describes the BIAS messages implementing BIAS operations as defined in ANSI INCITS  
391 442-2010. The operations are listed alphabetically, with each operation containing a request and a  
392 response message. The tables follow the conventions described in section 3.1.

### 393 4.1 Primitive Operations

#### 394 4.1.1 AddSubjectToGallery

395 [AddSubjectToGalleryRequest](#)

396 [AddSubjectToGalleryResponse](#)

397 The AddSubjectToGallery operation registers a subject to a given gallery or population group. As an  
398 OPTIONAL parameter, the value of the claim to identity by which the subject is known to the gallery MAY  
399 be specified. This claim to identity MUST be unique across the gallery. If no claim to identity is specified,  
400 the subject ID (assigned with the [CreateSubject](#) operation) will be used as the claim to identity.  
401 Additionally, in the encounter-centric model, the encounter ID associated with the subject's biometrics  
402 that will be added to the gallery MUST be specified.

#### 403 Request Message

| Field                        | Type                                      | #    | ? | Meaning   |
|------------------------------|---|------|---|---|
| AddSubjectToGallery          |   |      | Y | Register a subject to a given gallery or population group.                            |
| ↑ AddSubjectToGalleryRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters   | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                 |
| ↑ Application                | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser            | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                        |
| ↑ BIASOperationName          | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "AddSubjectToGallery".         |
| ↑ GalleryID                  | <a href="#">BIASIDType</a>                | 1    | Y | The identifier of the gallery or population group to which the subject will be added. |
| ↑ Identity                   | <a href="#">BIASIdentity</a>              | 1    | Y | The identity to add to the gallery.   |
| ↑ SubjectID                  | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |



| Field           | Type                       | #    | ? | Meaning   |
|-----------------|----------------------------|------|---|---|
| ↑ IdentityClaim | <a href="#">BIASIDType</a> | 0..1 | N | An identifier by which a subject is known to a particular gallery or population group. (This could be a username or account number, for example.) |
| ↑ EncounterID   | <a href="#">BIASIDType</a> | 0..1 | C | The identifier of an encounter associated with the subject.<br>Required for encounter-centric models.   |

#### 404 **Response Message**

| Field                                | Type                           | #    | ? | Meaning   |
|--------------------------------------|--------------------------------|------|---|---|
| AddSubjectToGalleryResponse          |                                |      | Y | The response to an AddSubjectToGallery operation.             |
| ↑ AddSubjectToGalleryResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                     | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                             | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                            | string                         | 0..1 | N | A short message corresponding to the return code.             |

#### 405 **4.1.2 CheckQuality**

##### 406 [CheckQualityRequest](#)

##### 407 [CheckQualityResponse](#)

408 The CheckQuality operation returns a quality score for a given biometric. The biometric input is provided  
409 in a CBEFF basic structure or CBEFF record, which in this specification is called a CBEFF-BIR. The  
410 algorithm vendor and algorithm vendor product ID MAY be optionally provided in order to request a  
411 particular algorithm's use in calculating the biometric quality. If an algorithm vendor is provided then the  
412 algorithm vendor product ID is REQUIRED. If no algorithm vendor is provided, the implementing system  
413 will provide the algorithm vendor and algorithm vendor product ID that were used to calculate the  
414 biometric quality as output parameters.

#### 415 **Request Message**

| Field        | Type | # | ? | Meaning  |
|--------------|------|---|---|--|
| CheckQuality |      |   | Y | Calculate a quality score for a given biometric. |

| Field                      | Type                                      | #    | ? | Meaning  |
|----------------------------|---|------|---|--|
| ↑ CheckQualityRequest      |   | 1    | Y |  |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                              |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                                     |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "CheckQuality".                             |
| ↑ BiometricData            | <a href="#">BIASBiometricDataType</a>     | 1    | Y | Data structure containing a single biometric sample for which a quality score is to be determined. |
| ↑ BIR                      | <a href="#">CBEFF_BIR_Type</a>            | 1    | Y | The biometric sample.  |
| ↑ Quality                  | <a href="#">QualityData</a>               | 0..1 | N | Specifies a particular algorithm vendor and vendor product ID.                                     |
| ↑ AlgorithmVendor          | <a href="#">VendorIdentifier</a>          | 1    | Y | The vendor of the quality algorithm used to determine the quality score.                           |
| ↑ AlgorithmVendorProductID | <a href="#">ProductID</a>                 | 1    | Y | The vendor's ID for the algorithm used to determine the quality.                                   |

416 **Response Message**

| Field                         | Type                             | #    | ? | Meaning  |
|-------------------------------|----------------------------------|------|---|--|
| CheckQualityResponse          |                                  |      | Y | The response to a CheckQuality operation.                                |
| ↳ CheckQualityResponsePackage |                                  | 1    | Y |  |
| ↳ ResponseStatus              | <a href="#">ResponseStatus</a>   | 1    | Y | Returned status for the operation.                                       |
| ↳ Return                      | <a href="#">ReturnCode</a>       | 1    | Y | The return code indicates the return status of the operation.            |
| ↳ Message                     | string                           | 0..1 | N | A short message corresponding to the return code.                        |
| ↳ QualityInfo                 | <a href="#">QualityData</a>      | 1    | Y | Contains the quality information for the submitted biometric sample.     |
| ↳ QualityScore                | oasis_cbeff:QualityType          | 0..1 | N | The quality of a biometric sample.                                       |
| ↳ AlgorithmVendor             | <a href="#">VendorIdentifier</a> | 1    | Y | The vendor of the quality algorithm used to determine the quality score. |
| ↳ AlgorithmVendorProductID    | <a href="#">ProductID</a>        | 1    | Y | The vendor's ID for the algorithm used to determine the quality.         |
| ↳ AlgorithmVersion            | <a href="#">VersionType</a>      | 1    | Y | The version of the algorithm used to determine the quality.              |

417 **4.1.3 ClassifyBiometricData**

418 [ClassifyBiometricDataRequest](#)

419 [ClassifyBiometricDataResponse](#)

420 The ClassifyBiometricData operation attempts to classify a biometric sample. For example, a fingerprint  
 421 biometric sample may be classified as a whorl, loop, or arch (or other classification classes and sub-  
 422 classes).

423 To obtain the types of classification algorithms and classes, see the [QueryCapabilities](#) operation.

424 **Request Message**

| Field                          | Type | # | ? | Meaning                        |
|--------------------------------|------|---|---|--------------------------------|
| ClassifyBiometricData          |      |   | Y | Classifies a biometric sample. |
| ↳ ClassifyBiometricDataRequest |      | 1 | Y |                                |

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                                 |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "ClassifyBiometricData".                       |
| ↑ BiometricData            | <a href="#">BIASBiometricDataType</a>     | 1    | Y | Data structure containing a single biometric sample for which the classification is to be determined. |
| ↑ BIR                      | <a href="#">CBEFF_BIR_Type</a>            | 1    | Y | The biometric sample.   |

425 **Response Message**

| Field                                  | Type                               | #    | ? | Meaning   |
|--|------------------------------------|------|---|---|
| ClassifyBiometricDataResponse          |                                    |      | Y | The response to a ClassifyBiometricData operation, containing the classification of a biometric sample. |
| ↑ ClassifyBiometricDataResponsePackage |                                    | 1    | Y |   |
| ↑ ResponseStatus                       | <a href="#">ResponseStatus</a>     | 1    | Y | Returned status for the operation.  |
| ↑ Return                               | <a href="#">ReturnCode</a>         | 1    | Y | The return code indicates the return status of the operation.   |
| ↑ Message                              | string                             | 0..1 | N | A short message corresponding to the return code.   |
| ↑ ClassificationData                   | <a href="#">ClassificationData</a> | 1    | Y | Information on the results and type of classification performed.  |
| ↑ Classification                       | <a href="#">Classification</a>     | 1    | Y | The result of the classification.   |

| Field                         | Type  | # | ? | Meaning  |
|-------------------------------|---|---|---|--|
| ↑ ClassificationAlgorithmType | <a href="#">ClassificationAlgorithmType</a> | 1 | Y | Identifies the type of classification algorithm that was used to perform the classification. |

#### 426 4.1.4 CreateSubject

427 [CreateSubjectRequest](#)

428 [CreateSubjectResponse](#)

429 The CreateSubject operation creates a new subject record and associates a subject ID to that record. As  
 430 an optional parameter, the subject ID MAY be specified by the caller. If no subject ID is specified, the  
 431 CreateSubject operation will generate one.

#### 432 Request Message

| Field                        | Type                                      | #    | ? | Meaning   |
|------------------------------|---|------|---|---|
| CreateSubject                |   |      | Y |   |
| ↑ CreateSubjectRequest       |   | 1    | Y |   |
| s ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.   |
| ↑ Application                | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                  |
| ↑ ApplicationUser            | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.          |
| ↑ BIASOperationName          | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "CreateSubject". |

#### 433 Response Message

| Field                          | Type                           | #    | ? | Meaning   |
|--------------------------------|--------------------------------|------|---|---|
| CreateSubjectResponse          |                                |      | Y | The response to a CreateSubject operation, containing the subject ID of the new subject record. |
| ↑ CreateSubjectResponsePackage |                                | 1    | Y |   |
| s ↑ ResponseStatus             | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.  |
| ↑ Return                       | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation.                                   |
| ↑ Message                      | string                         | 0..1 | N | A short message corresponding to the return code.   |

| Field       | Type                         | # | ? | Meaning                                   |
|-------------|------------------------------|---|---|---|
| ↑ Identity  | <a href="#">BIASIdentity</a> | 1 | Y |   |
| ↑ SubjectID | <a href="#">BIASIDType</a>   | 1 | Y | A system unique identifier for a subject. |

#### 434 4.1.5 DeleteBiographicData

435 [DeleteBiographicDataRequest](#)

436 [DeleteBiographicDataResponse](#)

437 The DeleteBiographicData operation erases all of the biographic data associated with a given subject  
438 record. In the encounter-centric model the operation erases all of the biographic data associated with a  
439 given encounter, and therefore the encounter ID MUST be specified.

440 When deleting data, BIAS implementations MAY completely erase the information in order to prevent the  
441 ability to reconstruct a record in whole or in part, or they MAY track and record the deleted information for  
442 auditing and/or quality control purposes.

#### 443 Request Message

| Field                         | Type                                      | #    | ? | Meaning   |
|-------------------------------|---|------|---|---|
| DeleteBiographicData          |   |      | Y | Erase all of the biographic data associated with a given subject record or, in the encounter-centric model, with a given encounter. |
| ↑ DeleteBiographicDataRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters    | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.   |
| ↑ Application                 | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser             | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↑ BIASOperationName           | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested:<br>"DeleteBiographicData"  |
| ↑ Identity                    | <a href="#">BIASIdentity</a>              | 1    | Y |   |
| ↑ SubjectID                   | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |

| Field         | Type                       | #    | ? | Meaning   |
|---------------|----------------------------|------|---|---|
| ↑ EncounterID | <a href="#">BIASIDType</a> | 0..1 | C | The identifier of an encounter associated with the subject.<br>Required for encounter-centric models. |

#### 444 Response Message

| Field                                 | Type                           | #    | ? | Meaning   |
|---------------------------------------|--------------------------------|------|---|---|
| DeleteBiographicDataResponse          |                                |      | Y | The response to a DeleteBiographicData operation.             |
| ↑ DeleteBiographicDataResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                      | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                              | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                             | string                         | 0..1 | N | A short message corresponding to the return code.             |

#### 445 4.1.6 DeleteBiometricData

446 [DeleteBiometricDataRequest](#)

447 [DeleteBiometricDataResponse](#)

448 The DeleteBiometricData operation erases all of the biometric data associated with a given subject  
449 record. In the encounter-centric model the operation erases all of the biometric data associated with a  
450 given encounter, and therefore the encounter ID MUST be specified.

451 When deleting data, BIAS implementations MAY completely erase the information in order to prevent the  
452 ability to reconstruct a record in whole or in part, or they MAY track and record the deleted information for  
453 auditing and/or quality control purposes.

#### 454 Request Message

| Field                        | Type | # | ? | Meaning  |
|------------------------------|------|---|---|--|
| DeleteBiometricData          |      |   | Y | Erase all of the biometric data associated with a given subject record or, in the encounter-centric model, with a given encounter. |
| ↑ DeleteBiometricDataRequest |      | 1 | Y |  |

| Field                          | Type                                      | #    | ? | Meaning   |
|--------------------------------|---|------|---|---|
| ↑ GenericRequestParameter<br>s | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                                 |
| ↑ Application                  | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser              | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↑ BIASOperationName            | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "DeleteBiometricData".                         |
| ↑ Identity                     | <a href="#">BIASIdentity</a>              | 1    | Y |   |
| ↑ SubjectID                    | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |
| ↑ EncounterID                  | <a href="#">BIASIDType</a>                | 0..1 | C | The identifier of an encounter associated with the subject.<br>Required for encounter-centric models. |

## 455 Response Message

| Field                                | Type                           | #    | ? | Meaning   |
|--------------------------------------|--------------------------------|------|---|---|
| DeleteBiometricDataResponse          |                                |      | Y | The response to a DeleteBiometricData operation.              |
| ↑ DeleteBiometricDataResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                     | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                             | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                            | string                         | 0..1 | N | A short message corresponding to the return code.             |

## 456 4.1.7 DeleteSubject

### 457 [DeleteSubjectRequest](#)

### 458 [DeleteSubjectResponse](#)

459 The DeleteSubject operation deletes an existing subject record and, in an encounter-centric model, any  
460 associated encounter information from the system. This operation also removes the subject from any  
461 registered galleries.



462 When deleting a subject, BIAS implementations MAY completely erase the subject information in order to  
 463 prevent the ability to reconstruct a record or records in whole or in part, or they MAY track and record the  
 464 deleted information for auditing and/or quality control purposes.

465 **Request Message**

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| DeleteSubject              |   |      | Y | Delete an existing subject record and, in an encounter-centric model, any associated encounter information. |
| ↳ DeleteSubjectRequest     |   | 1    | Y |   |
| ↳ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                                       |
| ↳ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↳ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↳ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "DeleteSubject".                                     |
| ↳ Identity                 | <a href="#">BIASIdentity</a>              | 1    | Y | The identity of the subject to delete.  |
| ↳ SubjectID                | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |

466 **Response Message**

| Field                          | Type                           | #    | ? | Meaning   |
|--------------------------------|--------------------------------|------|---|---|
| DeleteSubjectResponse          |                                |      | Y | The response to a DeleteSubject operation.                    |
| ↳ DeleteSubjectResponsePackage |                                | 1    | Y |   |
| ↳ ResponseStatus               | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↳ Return                       | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↳ Message                      | string                         | 0..1 | N | A short message corresponding to the return code.             |

467 **4.1.8 DeleteSubjectFromGallery**

468 [DeleteSubjectFromGalleryRequest](#)

469 [DeleteSubjectFromGalleryResponse](#)

470 The DeleteSubjectFromGallery operation removes the registration of a subject from a gallery or  
 471 population group. The subject is identified by either the subject ID or the claim to identity that was  
 472 specified in the [AddSubjectToGallery](#) operation.

473 **Request Message**

| Field                             | Type                                      | #    | ? | Meaning  |
|-----------------------------------|---|------|---|--|
| DeleteSubjectFromGallery          |   |      | Y | Remove the registration of a subject from a gallery or population group.   |
| ↳ DeleteSubjectFromGalleryRequest |   | 1    | Y |  |
| ↳ GenericRequestParameters        | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↳ Application                     | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↳ ApplicationUser                 | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↳ BIASOperationName               | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "DeleteSubjectFromGallery".   |
| ↳ GalleryID                       | <a href="#">BIASIDType</a>                | 1    | Y | The identifier of the gallery or population group from which the subject will be deleted.  |
| ↳ Identity                        | <a href="#">BIASIdentity</a>              | 1    | Y | The identity to remove from the gallery.   |
| ↳ SubjectID                       | <a href="#">BIASIDType</a>                | 0..1 | C | A system unique identifier for a subject. Required if an Identity Claim is not provided.   |
| ↳ IdentityClaim                   | <a href="#">BIASIDType</a>                | 0..1 | C | An identifier by which a subject is known to a particular gallery or population group. Required if a Subject ID is not provided. |

474 **Response Message**

| Field                                     | Type                           | #    | ? | Meaning   |
|---|--------------------------------|------|---|---|
| DeleteSubjectFromGalleryResponse          |                                |      | Y | The response to a DeleteSubjectFromGallery operation.         |
| ↳ DeleteSubjectFromGalleryResponsePackage |                                |      |   |   |
| ↳ ResponseStatus                          | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↳ Return                                  | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↳ Message                                 | string                         | 0..1 | N | A short message corresponding to the return code.             |

475 **4.1.9 GetIdentifySubjectResults**

476 [GetIdentifyResultsRequest](#)

477 [GetIdentifySubjectResultsResponse](#)

478 The GetIdentifySubjectResults operation retrieves the identification results for the specified token. This  
 479 operation is used in conjunction with the [IdentifySubject](#) operation. If the IdentifySubject operation is  
 480 implemented as an asynchronous service, the implementing system returns a token and the  
 481 GetIdentifySubjectResults operation is used to poll for the results of the original IdentifySubject request.

482 **Request Message**

| Field                              | Type                                      | #    | ? | Meaning   |
|------------------------------------|---|------|---|---|
| GetIdentifySubjectResults          |   |      | Y | Retrieve the identification results for a specified token, which was returned by the <a href="#">IdentifySubject</a> operation. |
| ↳ GetIdentifySubjectResultsRequest |   | 1    | Y |   |
| ↳ GenericRequestParameters         | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.   |
| ↳ Application                      | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↳ ApplicationUser                  | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |

| Field               | Type                      | #    | ? | Meaning  |
|---------------------|---------------------------|------|---|--|
| ↑ BIASOperationName | string                    | 0..1 | N | Identifies the BIAS operation that is being requested: "GetIdentifySubjectResults".                                  |
| ↑ Token             | <a href="#">TokenType</a> | 1    | Y | A value used to retrieve the results of an IdentifySubject request.  |
| ↑ TokenValue        | string                    | 1    | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time.    |
| ↑ Expiration        | date                      | 1    | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

483 **Response Message**

| Field                                      | Type                               | #    | ? | Meaning  |
|--|------------------------------------|------|---|--|
| GetIdentifySubjectResultsResponse          |                                    |      | Y | The response to a GetIdentifySubjectResults operation, which includes a candidate list.          |
| ↑ GetIdentifySubjectResultsResponsePackage |                                    | 1    | Y |  |
| ↑ ResponseStatus                           | <a href="#">ResponseStatus</a>     | 1    | Y | Returned status for the operation.   |
| ↑ Return                                   | <a href="#">ReturnCode</a>         | 1    | Y | The return code indicates the return status of the operation.                                    |
| ↑ Message                                  | string                             | 0..1 | N | A short message corresponding to the return code.  |
| ↑ CandidateList                            | <a href="#">CandidateListType</a>  | 1    | Y | A rank-ordered list of candidates that have a likelihood of matching the input biometric sample. |
| ↑ Candidate                                | <a href="#">CandidateType</a>      | 0..* | N | A single candidate.  |
| ↑ Score                                    | <a href="#">Score</a>              | 0..1 | N | The match score.   |
| ↑ BiographicData                           | <a href="#">BiographicDataType</a> | 0..1 | N | Biographic data associated with the candidate match.   |

| Field     | Type                               | #    | ? | Meaning  |
|-----------|------------------------------------|------|---|--|
| ↑ BIRList | <a href="#">CBEFF_BIR_ListType</a> | 1    | Y | Biometric data associated with the candidate match.              |
| ↑ BIR     | <a href="#">CBEFF_BIR_Type</a>     | 0..* | N | CBEFF structure containing information about a biometric sample. |

#### 484 4.1.10 IdentifySubject

485 [IdentifySubjectRequest](#)

486 [IdentifySubjectResponse](#)

487 The IdentifySubject operation performs an identification search against a given gallery for a given  
488 biometric, returning a rank-ordered candidate list of a given maximum size.

489 If the IdentifySubject operation is implemented as a synchronous service, the implementing system  
490 immediately processes the request and returns the results in the candidate list. If the IdentifySubject  
491 operation is implemented as an asynchronous service, the implementing system returns a token, which is  
492 an indication that the request is being handled asynchronously. In this case, the  
493 [GetIdentifySubjectResults](#) operation is used to poll for the results of the IdentifySubject request.

#### 494 Request Message

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| IdentifySubject            |   |      | Y | Perform an identification search against a given gallery for a given biometric. |
| ↑ IdentifySubjectRequest   |   | 1    | Y |   |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.           |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                  |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "IdentifySubject".       |
| ↑ GalleryID                | <a href="#">BIASIDType</a>                | 1    | Y | The identifier of the gallery or population group which will be searched.       |

| Field           | Type                                  | # | ? | Meaning  |
|-----------------|---------------------------------------|---|---|--|
| ↑ Identity      | <a href="#">BIASIdentity</a>          | 1 | Y | Contains the BIR, a data structure containing the biometric sample for the search. |
| ↑ BiometricData | <a href="#">BIASBiometricDataType</a> | 1 | Y | An Identity's biometric data.  |
| ↑ BIR           | <a href="#">CBEFF_BIR_Type</a>        | 1 | Y | Contains biometric information in either a non-XML or an XML representation.       |
| ↑ MaxListSize   | positiveInteger                       | 1 | Y | The maximum size of the candidate list that should be returned.                    |

495 **Response Message**

| Field                            | Type  | #    | ? | Meaning  |
|----------------------------------|---|------|---|--|
| IdentifySubjectResponse          |   |      | Y | The response to an IdentifySubject operation, returning a rank-ordered candidate list.   |
| ↑ IdentifySubjectResponsePackage |   | 1    | Y |  |
| ↑ ResponseStatus                 | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.   |
| ↑ Return                         | <a href="#">ReturnCode</a>  | 1    | Y | The return code indicates the return status of the operation.  |
| ↑ Message                        | string  | 0..1 | N | A short message corresponding to the return code.  |
| ↑ CandidateList                  | <a href="#">CandidateListResultType</a><br>(see <a href="#">IdentifySubjectResultType</a> ) | 0..1 | C | A rank-ordered list of candidates that have a likelihood of matching the input biometric sample (i.e., exceed the system threshold).<br><br>Rank ordering is from highest to lowest match score.<br><br>Returned with successful synchronous request processing. |
| ↑ Candidate                      | <a href="#">CandidateType</a>   | 0..* | N | A single candidate.  |
| ↑ Score                          | string  | 0..1 | N | The match score.   |

| Field            | Type  | #    | ? | Meaning   |
|------------------|---|------|---|---|
| ↑ BiographicData | <a href="#">BiographicDataType</a>  | 0..1 | N | Biographic data associated with the candidate match.  |
| ↑ BIRList        | <a href="#">CBEFF_BIR_ListType</a>  | 1    | Y | Biometric data associated with the candidate match.   |
| ↑ BIR            | <a href="#">CBEFF_BIR_Type</a>  | 0..* | N | CBEFF structure containing information about a biometric sample.  |
| ↑ Token          | <a href="#">TokenResultType</a><br>(see <a href="#">IdentifySubjectResultType</a> ) | 0..1 | C | A token used to retrieve the results of the IdentifySubject operation. Returned with asynchronous request processing. |
| ↑ TokenValue     | string  | 1    | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time.     |
| ↑ Expiration     | date  | 1    | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available.  |

496 NOTES:

497 (1) In the event that the number of candidates exceeding the threshold exceeds the  
498 MaxListSize, the system will determine which candidate is included in the last position of  
499 the rank ordered candidate list (i.e., in the event of a tie).

500 (2) Requesters MAY NOT change the system thresholds.

#### 501 **4.1.11 ListBiographicData**

502 [ListBiographicDataRequest](#)

503 [ListBiographicDataResponse](#)

504 The ListBiographicData operation lists the biographic data elements stored for a subject using the  
505 Biographic Data Elements output parameter. Note that no actual biographic data is returned by this  
506 operation (see the [RetrieveBiographicInformation](#) operation to obtain the biographic data). In the  
507 encounter-centric model, an encounter ID MAY be specified to indicate that only the biographic data  
508 elements stored for that encounter should be returned. If an encounter ID is not specified and encounter  
509 data exists for the subject, the operation returns the list of encounter IDs which contain biographic data  
510 using the Encounter List output parameter, and the Biographic Data Elements output parameter is empty.

#### 511 **Request Message**

| Field              | Type | # | ? | Meaning  |
|--------------------|------|---|---|--|
| ListBiographicData |      |   | Y | Lists the biographic data elements stored for a subject. |

| Field                       | Type                                      | #    | ? | Meaning  |
|-----------------------------|---|------|---|--|
| ↑ ListBiographicDataRequest |   | 1    | Y |  |
| ↑ GenericRequestParameters  | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                  |
| ↑ Application               | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser           | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                         |
| ↑ BIASOperationName         | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "ListBiographicData".           |
| ↑ Identity                  | <a href="#">BIASIdentity</a>              | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| ↑ SubjectID                 | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.  |
| ↑ EncounterID               | <a href="#">BIASIDType</a>                | 0..1 | N | The identifier of an encounter associated with the subject.                            |

## 512 Response Message

| Field                               | Type | # | ? | Meaning   |
|-------------------------------------|------|---|---|---|
| ListBiographicDataResponse          |      |   | Y | The response to a ListBiographicData request, containing a list of biographic data elements stored for a subject. In the encounter-centric model, the biographic data elements for a specific encounter are returned. If an encounter ID is not specified and encounter data exists for the subject, the list of encounter IDs which contain biographic data is returned. |
| ↑ ListBiographicDataResponsePackage |      | 1 | Y |   |



| Field                    | Type                               | #    | ? | Meaning  |
|--------------------------|------------------------------------|------|---|--|
| ↑ ResponseStatus         | <a href="#">ResponseStatus</a>     | 1    | Y | Returned status for the operation.   |
| ↑ Return                 | <a href="#">ReturnCode</a>         | 1    | Y | The return code indicates the return status of the operation.  |
| ↑ Message                | string                             | 0..1 | N | A short message corresponding to the return code.  |
| ↑ Identity               | <a href="#">BIASIdentity</a>       | 1    | Y | Contains a list of biographic data elements associated with a subject or encounter; non-empty if the service was successful, biographic data exists, and either (a) the person-centric model is being used or (b) the encounter-centric model is being used and an encounter identifier was specified. |
| ↑ BiographicDataElements | <a href="#">BiographicDataType</a> | 0..1 | C | An Identity's biographic data elements that are stored in the implementing system.   |
| ↑ BiographicDataItem     | <a href="#">BiographicDataItem</a> | 0..* | N | A single biographic data element.  |
| ↑ Name                   | string                             | 1    | Y | The name of the biographic data item.  |
| ↑ Type                   | string                             | 1    | Y | The data type for the biographic data item.  |
| ↑ EncounterList          | <a href="#">EncounterListType</a>  | 0..1 | C | A list of encounter ID's associated with a subject and which contain biographic data; non-empty if the service was successful, biographic data exists, the encounter-centric model is being used, and an encounter identifier was not specified.   |

| Field         | Type                       | #    | ? | Meaning                         |
|---------------|----------------------------|------|---|---------------------------------|
| ↑ EncounterID | <a href="#">BIASIDType</a> | 0..* | N | The identifier of an encounter. |

## 513 4.1.12 ListBiometricData

514 [ListBiometricDataRequest](#)

515 [ListBiometricDataResponse](#)

516 The ListBiometricData operation lists the biometric data elements stored for a subject using the Biometric  
517 Data List output parameter. Note that no actual biometric data is returned by this operation (see the  
518 [RetrieveBiometricInformation](#) operation to obtain the biometric data). In the encounter-centric model, an  
519 encounter ID MAY be specified to indicate that only the biometric data elements stored for that encounter  
520 should be returned. If an encounter ID is not specified and encounter data exists for the subject, the  
521 operation returns the list of encounter IDs which contain biometric data using the Encounter List output  
522 parameter, and the Biometric Data List output parameter is empty.

523 An optional parameter MAY be used to indicate a filter on the list of returned data. Such a filter may  
524 indicate that only biometric types should be listed (e.g., face, finger, iris, etc.) or that only biometric  
525 subtypes for a particular biometric type should be listed (e.g., all fingerprints: left slap, right index, etc.). If  
526 a filter is not specified, all biometric type and biometric subtype information are listed (e.g., left index  
527 finger, right iris, face frontal, etc.).

## 528 Request Message

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| ListBiometricData          |   |      | Y | Lists the biometric data elements stored for a subject.                     |
| ↑ ListBiometricDataRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.       |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                      |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.              |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "ListBiometricData". |

| Field                     | Type                           | #    | ? | Meaning  |
|---------------------------|--------------------------------|------|---|--|
| ↑ Identity                | <a href="#">BIASIdentity</a>   | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| ↑ SubjectID               | <a href="#">BIASIDType</a>     | 1    | Y | A system unique identifier for a subject.  |
| ↑ EncounterID             | <a href="#">BIASIDType</a>     | 0..1 | N | The identifier of an encounter associated with the subject.                            |
| ↑ ListFilterType          | <a href="#">ListFilterType</a> | 0..1 | N | Indicates what biometric information should be returned.                               |
| ↑ BiometricTypeFilter     | oasis_cbeff:MultipleTypesType  | 1..* | Y | Limits the returned information to a specific type of biometric, as defined by CBEFF.  |
| ↑ IncludeBiometricSubType | boolean                        | 1    | Y | A Boolean flag indicating if biometric subtype information should be returned.         |

## Response Message

| Field                              | Type                                  | #    | ? | Meaning   |
|------------------------------------|---------------------------------------|------|---|---|
| ListBiometricDataResponse          |                                       |      | Y | The response to a ListBiometricData operation, containing a list of biometric data elements stored for a subject. In the encounter-centric model, the biometric data elements for a specific encounter are returned. If an encounter ID is not specified and encounter data exists for the subject, the list of encounter IDs which contain biometric data is returned. |
| ↳ ListBiometricDataResponsePackage |                                       | 1    | Y |   |
| ↳ ResponseStatus                   | <a href="#">ResponseStatus</a>        | 1    | Y | Returned status for the operation.  |
| ↳ Return                           | <a href="#">ReturnCode</a>            | 1    | Y | The return code indicates the return status of the operation.   |
| ↳ Message                          | string                                | 0..1 | N | A short message corresponding to the return code.   |
| ↳ Identity                         | <a href="#">BIASIdentity</a>          | 0..1 | N | Includes a list of biometric data elements associated with a subject or encounter or a list of encounter ID's associated with a subject and which contain biometric data.   |
| ↳ BiometricData                    | <a href="#">BIASBiometricDataType</a> | 0..1 | C | An Identity's biometric data.   |

| Field                  | Type                                     | #    | ? | Meaning  |
|------------------------|--|------|---|--|
| ↑ BiometricDataList    | <a href="#">BiometricDataListType</a>    | 0..1 | N | A list of biometric data elements.   |
| ↑ BiometricDataElement | <a href="#">BiometricDataElementType</a> | 1..* | Y | Data structure containing information about a biometric record.  |
| ↑ BiometricType        | oasis_cbeff:MultipleTypesType            | 1    | Y | The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.   |
| ↑ BiometricTypeCount   | positiveInteger                          | 0..1 | N | The number of biometric records having the biometric type recorded in the biometric type field.  |
| ↑ BiometricSubType     | oasis_cbeff:SubtypeType                  | 0..1 | N | More specifically defines the type of biometric data stored in the biometric record, as defined by CBEFF.  |
| ↑ BDBFormatOwner       | positiveInteger                          | 1    | Y | Identifies the standards body, working group, industry consortium, or other CBEFF biometric organization that has defined the format for the biometric data. |
| ↑ BDBFormatType        | positiveInteger                          | 1    | Y | Identifies the specific biometric data format specified by the CBEFF biometric organization recorded in the BDB Format Owner field.                          |

| Field           | Type                              | #    | ? | Meaning  |
|-----------------|-----------------------------------|------|---|--|
| ↑ EncounterList | <a href="#">EncounterListType</a> | 0..1 | C | A list of encounter ID's associated with a subject and which contain biometric data; non-empty if the service was successful, biometric data exists, the encounter-centric model is being used, and an encounter identifier was not specified. |
| ↑ EncounterID   | <a href="#">BIASIDType</a>        | 1..* | Y | The identifier of an encounter.  |

### 530 4.1.13 PerformFusion

531 [PerformFusionRequest](#)

532 [PerformFusionResponse](#)

533 The PerformFusion operation accepts either match score or match decision information and creates a  
534 fused match result. The [FusionInformationListType](#), through the [FusionInformationType](#), provides specific  
535 elements for match score input and match decision input. The fusion method and processes are left to the  
536 implementing system.

### 537 Request Message

| Field                      | Type                                     | #    | ? | Meaning  |
|----------------------------|--|------|---|--|
| PerformFusion              |  |      | Y | Accepts either match score or match decision information and creates a fused match result. |
| ↑ PerformFusionRequest     |  | 1    | Y |  |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a> | 0..1 | N | Common request parameters that can be used to identify the requester.                      |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>    | 0..1 | N | Identifies the requesting application.   |

| Field               | Type                                      | #    | ? | Meaning  |
|---------------------|---|------|---|--|
| ↑ ApplicationUser   | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↑ BIASOperationName | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "PerformFusion".  |
| ↑ FusionInput       | <a href="#">FusionInformationListType</a> | 1    | Y | Score or decision input information to the fusion method.  |
| ↑ FusionElement     | <a href="#">FusionInformationType</a>     | 2..* | Y | A set of fusion information.   |
| ↑ BiometricType     | oasis_cbeff:MultipleTypesType             | 1    | Y | The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.                           |
| ↑ BiometricSubType  | oasis_cbeff:SubtypeType                   | 0..1 | N | More specifically defines the type of biometric data stored in the biometric record.                                     |
| ↑ AlgorithmOwner    | string                                    | 1    | Y | The owner or vendor of the algorithm used to determine the score or decision.  |
| ↑ AlgorithmType     | string                                    | 1    | Y | The Algorithm Owner's identifier for the specific algorithm product and version used to determine the score or decision. |
| ↑ FusionResult      | <a href="#">FusionResult</a>              | 0..1 | C | Either <a href="#">FusionScore</a> or a <a href="#">FusionDecision</a> element MUST be used.                             |

## 538 Response Message

| Field                          | Type | # | ? | Meaning                                      |
|--------------------------------|------|---|---|--|
| PerformFusionResponse          |      |   | Y | The response to the PerformFusion operation. |
| ↑ PerformFusionResponsePackage |      | 1 | Y |  |

| Field            | Type                           | #    | ? | Meaning   |
|------------------|--------------------------------|------|---|---|
| ↳ ResponseStatus | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↳ Return         | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↳ Message        | string                         | 0..1 | N | A short message corresponding to the return code.             |
| ↳ Match          | <a href="#">MatchType</a>      | 1    | 1 | Indicates the result of the fusion method.                    |

#### 539 4.1.14 QueryCapabilities

540 [QueryCapabilitiesRequest](#)

541 [QueryCapabilitiesResponse](#)

542 The QueryCapabilities operation returns a list of the capabilities, options, galleries, etc. that are supported  
543 by the BIAS implementation. Refer to Annex A in the INCITS BIAS standard [[INCITS-BIAS](#)] for  
544 conformance requirements regarding which capability names an implementation must use in the  
545 QueryCapabilities operation.

#### 546 Request Message

| Field                          | Type                                      | #    | ? | Meaning   |
|--------------------------------|---|------|---|---|
| QueryCapabilities              |   |      | Y | Returns a list of the capabilities, options, galleries, etc. that are supported by the BIAS implementation. |
| ↳ QueryCapabilitiesRequest     |   | 1    | Y |   |
| s<br>↳ GenericRequestParameter | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                                       |
| ↳ Application                  | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↳ ApplicationUser              | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↳ BIASOperationName            | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "QueryCapabilities".                                 |



547 **Response Message**

| Field                              | Type                               | #    | ? | Meaning  |
|------------------------------------|------------------------------------|------|---|--|
| QueryCapabilitiesResponse          |                                    |      | Y | The response to a QueryCapabilities operation.                       |
| ↑ QueryCapabilitiesResponsePackage |                                    | 1    | Y |  |
| ↑ ResponseStatus                   | <a href="#">ResponseStatus</a>     | 1    | Y | Returned status for the operation.                                   |
| ↑ Return                           | <a href="#">ReturnCode</a>         | 1    | Y | The return code indicates the return status of the operation.        |
| ↑ Message                          | string                             | 0..1 | N | A short message corresponding to the return code.                    |
| ↑ CapabilityList                   | <a href="#">CapabilityListType</a> | 1    | Y | A list of capabilities supported by the BIAS implementation.         |
| ↑ Capability                       | <a href="#">CapabilityType</a>     | 0..* | N | A single capability.   |
| ↑ CapabilityName                   | <a href="#">CapabilityName</a>     | 1    | Y | The name of the capability.  |
| ↑ CapabilityID                     | string                             | 0..1 | N | An identifier assigned to the capability by the implementing system. |
| ↑ CapabilityDescription            | string                             | 0..1 | N | A description of the capability.                                     |
| ↑ CapabilityValue                  | string                             | 0..1 | N | A value assigned to the capability.                                  |
| ↑ CapabilitySupportingValue        | string                             | 0..1 | N | A secondary value supporting the capability.                         |
| ↑ CapabilityAdditionalInfo         | string                             | 0..1 | N | Contains additional information for the supported capability.        |

548 **4.1.15 RetrieveBiographicInformation**

549 [RetrieveBiographicInformationRequest](#)

550 [RetrieveBiographicInformationResponse](#)

551 The RetrieveBiographicInformation operation retrieves the biographic data associated with a subject ID.

552 In the encounter-centric model, the encounter ID MAY be specified and the operation will return the

553 biographic data associated with that encounter. If the encounter ID is not specified in the encounter-

554 centric model, the operation returns the biographic information associated with the most recent  
 555 encounter.

556 **Request Message**

| Field                                  | Type                                      | #    | ? | Meaning   |
|--|---|------|---|---|
| RetrieveBiographicInformation          |   |      | Y | Retrieves the biographic data associated with a subject ID.                             |
| ↳ RetrieveBiographicInformationRequest |   | 1    | Y |   |
| ↳ GenericRequestParameters             | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                   |
| ↳ Application                          | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↳ ApplicationUser                      | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                          |
| ↳ BIASOperationName                    | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "RetrieveBiographicInformation". |
| ↳ Identity                             | <a href="#">BIASIdentity</a>              | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter.  |
| ↳ SubjectID                            | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |
| ↳ EncounterID                          | <a href="#">BIASIDType</a>                | 0..1 | N | The identifier of an encounter associated with the subject.                             |

557 **Response Message**

| Field  | Type | # | ? | Meaning  |
|--|------|---|---|--|
| RetrieveBiographicInformationResponse          |      |   | Y | The response to a RetrieveBiographicInformation operation. |
| ↳ RetrieveBiographicInformationResponsePackage |      | 1 | Y |  |

| Field                | Type                                   | #    | ? | Meaning  |
|----------------------|--|------|---|--|
| ↑ ResponseStatus     | <a href="#">ResponseStatus</a>         | 1    | Y | Returned status for the operation.   |
| ↑ Return             | <a href="#">ReturnCode</a>             | 1    | Y | The return code indicates the return status of the operation.                    |
| ↑ Message            | string                                 | 0..1 | N | A short message corresponding to the return code.                                |
| ↑ Identity           | <a href="#">BIASIdentity</a>           | 1    | Y | Includes the set of biographic data associated with a subject.                   |
| ↑ BiographicData     | <a href="#">BiographicDataType</a>     | 1    | Y | An Identity's biographic data.<br>One of the following elements MUST be present. |
| ↑ LastName           | string                                 | 0..1 | C | The last name of a subject.  |
| ↑ FirstName          | string                                 | 0..1 | C | The first name of a subject.   |
| ↑ BiographicDataItem | <a href="#">BiographicDataItemType</a> | 0..* | C | A single biographic data element.  |
| ↑ BiographicDataSet  | <a href="#">BiographicDataItemType</a> | 0..1 | C | A set of biographic data information.  |

#### 558 **4.1.16 RetrieveBiometricInformation**

559 [RetrieveBiometricInformationRequest](#)

560 [RetrieveBiometricInformationResponse](#)

561 The RetrieveBiometricInformation operation retrieves the biometric data associated with a subject ID. In  
562 the encounter-centric model, the encounter ID MAY be specified and the operation will return the biometric  
563 data associated with that encounter. If the encounter ID is not specified in the encounter-centric model,  
564 the operation returns the biometric information associated with the most recent encounter. The operation  
565 provides an OPTIONAL input parameter to specify that only biometric data of a certain type should be  
566 retrieved.

#### 567 **Request Message**

| Field                        | Type | # | ? | Meaning  |
|------------------------------|------|---|---|--|
| RetrieveBiometricInformation |      |   | Y | Retrieves the biometric data associated with a subject ID. |

| Field                                 | Type                                      | #    | ? | Meaning  |
|---------------------------------------|---|------|---|--|
| ↑ RetrieveBiometricInformationRequest |   | 1    | Y |  |
| ↑ GenericRequestParameters            | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.                  |
| ↑ Application                         | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser                     | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                         |
| ↑ BIASOperationName                   | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "RetrieveBiometricInformation". |
| ↑ Identity                            | <a href="#">BIASIdentity</a>              | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| ↑ SubjectID                           | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.  |
| ↑ EncounterID                         | <a href="#">BIASIDType</a>                | 0..1 | N | The identifier of an encounter associated with the subject.                            |
| ↑ BiometricType                       | oasis_cbeff:MultipleTypesType             | 0..1 | N | The type of biological or behavioral data to retrieve.                                 |

## 568 Response Message

| Field   | Type                           | #    | ? | Meaning   |
|---|--------------------------------|------|---|---|
| RetrieveBiometricInformationResponse          |                                |      | Y | The response to a RetrieveBiometricInformation operation.     |
| ↑ RetrieveBiometricInformationResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                              | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                                      | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                                     | string                         | 0..1 | N | A short message corresponding to the return code.             |

| Field           | Type                                  | #    | ? | Meaning  |
|-----------------|---------------------------------------|------|---|--|
| ↑ Identity      | <a href="#">BIASIdentity</a>          | 1    | Y | Includes the biometric data associated with a subject.           |
| ↑ BiometricData | <a href="#">BIASBiometricDataType</a> | 1    | Y | An Identity's biometric data.                                    |
| ↑ BIRList       | <a href="#">CBEFF_BIR_ListType</a>    | 1    | Y | A list of CBEFF-BIR elements.                                    |
| ↑ BIR           | <a href="#">CBEFF_BIR_Type</a>        | 0..* | N | CBEFF structure containing information about a biometric sample. |

#### 569 4.1.17 SetBiographicData

570 [SetBiographicDataRequest](#)

571 [SetBiometricDataResponse](#)

572 The SetBiographicData operation associates biographic data to a given subject record. The identity  
573 model of the system determines whether the biographic information should replace any existing  
574 biographic information (person-centric model) or if a new encounter should be created and associated  
575 with the subject (encounter-centric model). For encounter-centric models, the encounter ID MAY be  
576 specified by the caller in order to link biographic and biometric information (assuming biometric  
577 information was previously associated using the [SetBiometricData](#) operation). If the encounter ID is  
578 omitted for the encounter-centric model, the operation returns a system-assigned encounter ID.

#### 579 Request Message

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| SetBiographicData          |   |      | Y | Associates biographic data to a given subject record.                       |
| ↑ SetBiographicDataRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.       |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                      |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.              |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "SetBiographicData". |

| Field                | Type                                  | #    | ? | Meaning   |
|----------------------|---------------------------------------|------|---|---|
| ↑ Identity           | <a href="#">BIASIdentity</a>          | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to store. |
| ↑ SubjectID          | <a href="#">BIASIDType</a>            | 1    | Y | A system unique identifier for a subject.   |
| ↑ EncounterID        | <a href="#">BIASIDType</a>            | 0..1 | N | The identifier of an encounter associated with the subject.   |
| ↑ BiographicData     | <a href="#">BiographicDataType</a>    | 1    | Y | An Identity's biographic data. One of the following elements MUST be present.   |
| ↑ LastName           | string                                | 0..1 | C | The last name of a subject.   |
| ↑ FirstName          | string                                | 0..1 | C | The first name of a subject.  |
| ↑ BiographicDataItem | <a href="#">BiographicDataItem</a>    | 0..* | C | A single biographic data element.   |
| ↑ BiographicDataSet  | <a href="#">BiographicDataSetType</a> | 0..1 | C | A set of biographic data information.   |

## 580 Response Message

| Field                              | Type                           | #    | ? | Meaning   |
|------------------------------------|--------------------------------|------|---|---|
| SetBiographicDataResponse          |                                |      | Y | The response to a SetBiographicData operation.                |
| ↑ SetBiographicDataResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                   | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                           | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                          | string                         | 0..1 | N | A short message corresponding to the return code.             |

| Field         | Type                         | #    | ? | Meaning   |
|---------------|------------------------------|------|---|---|
| ↑ Identity    | <a href="#">BIASIdentity</a> | 0..1 | C | In an encounter-centric model, identifies the encounter ID assigned to a new encounter. |
| ↑ EncounterID | <a href="#">BIASIDType</a>   | 1    | Y | The identifier of an encounter associated with the subject.                             |

581 **4.1.18 SetBiometricData**

582 [SetBiometricDataRequest](#)

583 [SetBiometricDataResponse](#)

584 The SetBiometricData operation associates biometric data to a given subject record. The identity model  
585 of the system determines whether the biometric information should replace any existing biometric  
586 information (person-centric model) or if a new encounter should be created and associated with the  
587 subject (encounter-centric model). For encounter-centric models, the encounter ID MAY be specified by  
588 the caller in order to link biographic and biometric information (assuming biographic information was  
589 previously associated using the [SetBiographicData](#) operation). If the encounter ID is omitted for the  
590 encounter-centric model, the operation returns a system-assigned encounter ID.

591 **Request Message**

| Field                          | Type                                      | #    | ? | Meaning  |
|--------------------------------|---|------|---|--|
| SetBiometricData               |   |      | Y | Associates biometric data to a given subject record.   |
| ↑ SetBiometricDataRequest      |   | 1    | Y |  |
| ↑ GenericRequestParameter<br>s | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↑ Application                  | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser              | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↑ BIASOperationName            | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "SetBiometricData".   |
| ↑ Identity                     | <a href="#">BIASIdentity</a>              | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to store. |
| ↑ SubjectID                    | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.  |

| Field           | Type                                  | #    | ? | Meaning  |
|-----------------|---------------------------------------|------|---|--|
| ↑ EncounterID   | <a href="#">BIASIDType</a>            | 0..1 | N | The identifier of an encounter associated with the subject.      |
| ↑ BiometricData | <a href="#">BIASBiometricDataType</a> | 1    | Y | An Identity's biometric data.                                    |
| ↑ BIRList       | <a href="#">CBEFF_BIR_ListType</a>    | 1    | Y | A list of CBEFF-BIR elements.                                    |
| ↑ BIR           | <a href="#">CBEFF_BIR_Type</a>        | 1..* | Y | CBEFF structure containing information about a biometric sample. |

## 592 Response Message

| Field                             | Type                           | #    | ? | Meaning   |
|-----------------------------------|--------------------------------|------|---|---|
| SetBiometricDataResponse          |                                |      | Y | The response to a SetBiometricData operation.   |
| ↑ SetBiometricDataResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                  | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.  |
| ↑ Return                          | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation.                           |
| ↑ Message                         | string                         | 0..1 | N | A short message corresponding to the return code.                                       |
| ↑ Identity                        | <a href="#">BIASIdentity</a>   | 0..1 | C | In an encounter-centric model, identifies the encounter ID assigned to a new encounter. |
| ↑ EncounterID                     | <a href="#">BIASIDType</a>     | 1    | Y | The identifier of an encounter associated with the subject.                             |

## 593 4.1.19 TransformBiometricData

594 [TransformBiometricDataRequest](#)

595 [TransformBiometricDataResponse](#)

596 The TransformBiometricData operation transforms or processes a given biometric in one format into a  
597 new target format.



## Request Message

| Field                           | Type                                      | #    | ? | Meaning  |
|---------------------------------|---|------|---|--|
| TransformBiometricData          |   |      | Y | Transforms or processes a given biometric in one format into a new target format.  |
| ↑ TransformBiometricDataRequest |   | 1    | Y |  |
| ↑ GenericRequestParameters      | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↑ Application                   | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser               | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↑ BIASOperationName             | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested:<br>"TransformBiometricData".  |
| ↑ InputBIR                      | <a href="#">CBEFF_BIR_Type</a>            | 1    | Y | Data structure containing the biometric information to be transformed.   |
| ↑ TransformOperation            | unsignedLong                              | 1    | Y | Value indicating the type of transformation to perform.  |
| ↑ TransformControl              | string                                    | 0..1 | N | Specifies controls for the requested transform operation.<br><br>Note: This could be a compression ratio, target data format, etc. |

599 NOTE: The values for TransformOperation and TransformControl are implementation specific.

## 600 Response Message

| Field                                   | Type                           | # | ? | Meaning   |
|---|--------------------------------|---|---|---|
| TransformBiometricDataResponse          |                                |   | Y | The response to a TransformBiometricData operation. |
| ↑ TransformBiometricDataResponsePackage |                                | 1 | Y |   |
| ↑ ResponseStatus                        | <a href="#">ResponseStatus</a> | 1 | Y | Returned status for the operation.                  |

| Field       | Type                           | #    | ? | Meaning   |
|-------------|--------------------------------|------|---|---|
| ↑ Return    | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation.         |
| ↑ Message   | string                         | 0..1 | N | A short message corresponding to the return code.                     |
| ↑ OutputBIR | <a href="#">CBEFF_BIR_Type</a> | 0..1 | N | Data structure containing the new, transformed biometric information. |

601 **4.1.20 UpdateBiographicData**

602 [UpdateBiographicDataRequest](#)

603 [UpdateBiographicDataResponse](#)

604 The UpdateBiographicData operation updates the biographic data for an existing subject record. The  
605 operation replaces any existing biographic data with the new biographic data. In the encounter-centric  
606 model, the encounter ID MUST be specified.

607 **Request Message**

| Field                         | Type                                      | #    | ? | Meaning   |
|-------------------------------|---|------|---|---|
| UpdateBiographicData          |   |      | Y | Updates the biographic data for a given subject record.                         |
| ↑ UpdateBiographicDataRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters    | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.           |
| ↑ Application                 | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser             | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                  |
| ↑ BIASOperationName           | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "UpdateBiographic Data". |

| Field                | Type                                  | #    | ? | Meaning  |
|----------------------|---------------------------------------|------|---|--|
| ↑ Identity           | <a href="#">BIASIdentity</a>          | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to update. |
| ↑ SubjectID          | <a href="#">BIASIDType</a>            | 1    | Y | A system unique identifier for a subject.  |
| ↑ EncounterID        | <a href="#">BIASIDType</a>            | 0..1 | C | The identifier of an encounter associated with the subject.<br>Required for encounter-centric models.                              |
| ↑ BiographicData     | <a href="#">BiographicDataType</a>    | 1    | Y | An Identity's biographic data.<br>One of the following elements MUST be present.   |
| ↑ LastName           | string                                | 0..1 | C | The last name of a subject.  |
| ↑ FirstName          | string                                | 0..1 | C | The first name of a subject.   |
| ↑ BiographicDataItem | <a href="#">BiographicDataItem</a>    | 0..* | C | A single biographic data element.  |
| ↑ BiographicDataSet  | <a href="#">BiographicDataSetType</a> | 0..1 | C | A set of biographic data information.  |

## 608 Response Message

| Field                                 | Type                           | # | ? | Meaning   |
|---------------------------------------|--------------------------------|---|---|---|
| UpdateBiographicDataResponse          |                                |   | Y | The response to an UpdateBiographicData operation.            |
| ↑ UpdateBiographicDataResponsePackage |                                | 1 | Y |   |
| ↑ ResponseStatus                      | <a href="#">ResponseStatus</a> | 1 | Y | Returned status for the operation.                            |
| ↑ Return                              | <a href="#">ReturnCode</a>     | 1 | Y | The return code indicates the return status of the operation. |

| Field     | Type   | #    | ? | Meaning   |
|-----------|--------|------|---|---|
| ↑ Message | string | 0..1 | N | A short message corresponding to the return code. |

## 609 4.1.21 UpdateBiometricData

610 [UpdateBiometricDataRequest](#)

611 [UpdateBiometricDataResponse](#)

612 The UpdateBiometricData operation updates the biometric data for an existing subject record. The  
613 operation includes an OPTIONAL parameter indicating if the new biometric sample should be merged  
614 with the existing biometric sample. If this parameter is set to “False” or is not used in the request, the  
615 operation replaces the existing biometric sample with the new biometric sample. In the encounter-centric  
616 model, the encounter ID MUST be specified.

## 617 Request Message

| Field                        | Type                                      | #    | ? | Meaning   |
|------------------------------|---|------|---|---|
| UpdateBiometricData          |   |      | Y | Updates a single biometric sample for a given subject record.   |
| ↑ UpdateBiometricDataRequest |   | 1    | Y |   |
| ↑ GenericRequestParameters   | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.   |
| ↑ Application                | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↑ ApplicationUser            | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.  |
| ↑ BIASOperationName          | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: “UpdateBiometricData”.   |
| ↑ Identity                   | <a href="#">BIASIdentity</a>              | 1    | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to update. |
| ↑ SubjectID                  | <a href="#">BIASIDType</a>                | 1    | Y | A system unique identifier for a subject.   |
| ↑ EncounterID                | <a href="#">BIASIDType</a>                | 0..1 | C | The identifier of an encounter associated with the subject.<br>Required for encounter-centric models.                             |

| Field           | Type                                  | #    | ? | Meaning  |
|-----------------|---------------------------------------|------|---|--|
| ↑ BiometricData | <a href="#">BIASBiometricDataType</a> | 1    | Y | An Identity's biometric data.  |
| ↑ BIR           | <a href="#">CBEFF_BIR_Type</a>        | 1    | Y | Contains biometric information in either a non-XML or an XML representation.                             |
| ↑ Merge         | boolean                               | 0..1 | N | Value indicating if the input biometric sample should be merged with any existing biometric information. |

## 618 Response Message

| Field                                | Type                           | #    | ? | Meaning   |
|--------------------------------------|--------------------------------|------|---|---|
| UpdateBiometricDataResponse          |                                |      | Y | The response to an UpdateBiometricData operation.             |
| ↑ UpdateBiometricDataResponsePackage |                                | 1    | Y |   |
| ↑ ResponseStatus                     | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.                            |
| ↑ Return                             | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                            | string                         | 0..1 | N | A short message corresponding to the return code.             |

## 619 4.1.22 VerifySubject

620 [VerifySubjectRequest](#)

621 [VerifySubjectResponse](#)

622 The VerifySubject operation performs a 1:1 verification match between a given biometric and either a  
623 claim to identity in a given gallery or another given biometric. As such either the Identity Claim or  
624 Reference BIR input parameters are REQUIRED.

## 625 Request Message

| Field                  | Type | # | ? | Meaning   |
|------------------------|------|---|---|---|
| VerifySubject          |      |   | Y | Performs a 1:1 verification match between a given biometric and either a claim to identity in a given gallery or another given biometric. |
| ↑ VerifySubjectRequest |      | 1 | Y |   |

| Field                      | Type                                      | #    | ? | Meaning  |
|----------------------------|---|------|---|--|
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "VerifySubject".  |
| ↑ GalleryID                | <a href="#">BIASIDType</a>                | 0..1 | C | The identifier of the gallery or population group of which the subject must be a member.<br>Required if an Identity Claim is provided. |
| ↑ Identity                 | <a href="#">BIASIdentity</a>              | 1    | Y | Includes the identifying information and/or input and reference biometric samples.   |
| ↑ IdentityClaim            | <a href="#">BIASIDType</a>                | 0..1 | C | An identifier by which a subject is known to a particular gallery or population group.<br>Required if a Reference BIR is not provided. |
| ↑ BiometricData            | <a href="#">BIASBiometricDataType</a>     | 1    | Y | An Identity's biometric data.  |
| ↑ InputBIR                 | <a href="#">CBEFF_BIR_Type</a>            | 1    | Y | Maps to specific INCITS BIAS elements as required by that specification.   |
| ↑ ReferenceBIR             | <a href="#">CBEFF_BIR_Type</a>            | 0..1 | C | Maps to specific INCITS BIAS elements as required by that specification.<br>Required if an Identity Claim is not provided.             |

626 **Response Message**

| Field                          | Type                           | #    | ? | Meaning  |
|--------------------------------|--------------------------------|------|---|--|
| VerifySubjectResponse          |                                |      | Y | The response to a VerifySubject operation.   |
| ↳ VerifySubjectResponsePackage |                                | 1    | Y |  |
| ↳ ↳ ResponseStatus             | <a href="#">ResponseStatus</a> | 1    | Y | Returned status for the operation.   |
| ↳ ↳ Return                     | <a href="#">ReturnCode</a>     | 1    | Y | The return code indicates the return status of the operation.  |
| ↳ ↳ Message                    | string                         | 0..1 | N | A short message corresponding to the return code.  |
| ↳ ↳ Match                      | boolean                        | 0..1 | N | Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR. |
| ↳ ↳ Score                      | <a href="#">Score</a>          | 0..1 | N | The score if the biometric information matched.  |

627 **4.2 Aggregate Operations**

628 **4.2.1 Enroll**

629 [EnrollRequest](#)

630 [EnrollResponse](#)

631 The Enroll operation adds a new subject or, in an encounter-centric model, a new encounter to the  
 632 system. This may be accomplished in a number of different ways according to system requirements  
 633 and/or resources. If the Enroll operation is implemented as a synchronous service, the implementing  
 634 system immediately processes the request and returns the results in the Return Data parameter. If the  
 635 Enroll operation is implemented as an asynchronous service, the implementing system returns a token in  
 636 the Return Data parameter, which is an indication that the request is being handled asynchronously. In  
 637 this case, the [GetEnrollResults](#) operation is used to poll for the results of the Enroll request.

638 **Request Message**

| Field           | Type | # | ? | Meaning  |
|-----------------|------|---|---|--|
| Enroll          |      |   | Y | Adds a new subject or, in an encounter-centric model, a new encounter to the system. |
| ↳ EnrollRequest |      | 1 | Y |  |

| Field                          | Type                                      | #    | ? | Meaning  |
|--------------------------------|---|------|---|--|
| ↑ GenericRequestParameter<br>s | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.              |
| ↑ Application                  | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser              | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                     |
| ↑ BIASOperationName            | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "Enroll".                   |
| ↑ ProcessingOptions            | <a href="#">ProcessingOptionsType</a>     | 1    | Y | Options that guide how the aggregate service request is processed.                 |
| ↑ Option                       | string                                    | 0..* | N | An option supported by the implementing system.                                    |
| ↑ InputData                    | <a href="#">InformationType</a>           | 1    | Y | Contains the input data for the operation, as required by the implementing system. |

## 639 Response Message

| Field                   | Type                            | #    | ? | Meaning   |
|-------------------------|---------------------------------|------|---|---|
| EnrollResponse          |                                 |      | Y | The response to an Enroll operation.                          |
| ↑ EnrollResponsePackage |                                 | 1    | Y |   |
| ↑ ResponseStatus        | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.                            |
| ↑ Return                | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message               | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↑ ReturnData            | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

## 640 4.2.2 GetEnrollResults

641 [GetEnrollResultsRequest](#)

642 [GetEnrollResultsResponse](#)

643 The GetEnrollResults operation retrieves the enrollment results for the specified token. This operation is  
644 used in conjunction with the [Enroll](#) operation. If the Enroll operation is implemented as an asynchronous  
645 service, the implementing system returns a token and the GetEnrollResults operation is used to poll for  
646 the results of the original Enroll request.



647 If the service provider implements an asynchronous Enroll operation, then it MUST also implement the  
 648 GetEnrollResults operation.

649 **Request Message**

| Field                      | Type                                      | #    | ? | Meaning  |
|----------------------------|---|------|---|--|
| GetEnrollResults           |   |      | Y | Retrieves the enrollment results for the specified token.  |
| ↳ GetEnrollResultsRequest  |   | 1    | Y |  |
| ↳ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↳ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↳ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↳ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "GetEnrollResults".   |
| ↳ Token                    | <a href="#">TokenType</a>                 | 1    | Y | A value used to retrieve the results of the Enroll request.  |
| ↳ TokenValue               | string                                    | 1    | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time.    |
| ↳ Expiration               | date                                      | 1    | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

650 **Response Message**

| Field                             | Type                           | # | ? | Meaning                                       |
|-----------------------------------|--------------------------------|---|---|---|
| GetEnrollResultsResponse          |                                |   | Y | The response to a GetEnrollResults operation. |
| ↳ GetEnrollResultsResponsePackage |                                | 1 | Y |   |
| ↳ ResponseStatus                  | <a href="#">ResponseStatus</a> | 1 | Y | Returned status for the operation.            |

| Field        | Type                            | #    | ? | Meaning   |
|--------------|---------------------------------|------|---|---|
| ↑ Return     | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message    | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↑ ReturnData | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

651 **4.2.3 GetIdentifyResults**

652 [GetIdentifyResultsRequest](#)

653 [GetIdentifyResultsResponse](#)

654 The GetIdentifyResults operation retrieves the identification results for the specified token. This operation  
655 is used in conjunction with the [Identify](#) operation. If the Identify operation is implemented as an  
656 asynchronous service, the implementing system returns a token and the GetIdentifyResults operation is  
657 used to poll for the results of the original Identify request.

658 If the service provider implements an asynchronous Identify operation, then it MUST also implement the  
659 GetIdentifyResults operation.

660

661 **Request Message**

| Field                       | Type                                      | #    | ? | Meaning  |
|-----------------------------|---|------|---|--|
| GetIdentifyResults          |   |      | Y | Retrieves the identification results for the specified token                 |
| ↑ GetIdentifyResultsRequest |   | 1    | Y |  |
| ↑ GenericRequestParameters  | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.        |
| ↑ Application               | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                       |
| ↑ ApplicationUser           | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.               |
| ↑ BIASOperationName         | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "GetIdentifyResults". |
| ↑ Token                     | <a href="#">TokenType</a>                 | 1    | Y | A value used to retrieve the results of the Identify request.                |
| ↑ TokenValue                | string                                    | 1    | Y | A value returned by the implementing system that is used to                  |

| Field        | Type | # | ? | Meaning  |
|--------------|------|---|---|--|
|              |      |   |   | retrieve the results to an operation at a later time.  |
| ↑ Expiration | date | 1 | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

## 662 Response Message

| Field                               | Type                            | #    | ? | Meaning   |
|-------------------------------------|---------------------------------|------|---|---|
| GetIdentifyResultsResponse          |                                 |      | Y | The response to a GetIdentifyResults operation.               |
| ↑ GetIdentifyResultsResponsePackage |                                 | 1    | Y |   |
| ↑ ResponseStatus                    | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.                            |
| ↑ Return                            | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                           | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↑ ReturnData                        | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

## 663 4.2.4 GetVerifyResults

### 664 [GetVerifyResultsRequest](#)

### 665 [GetVerifyResultsResponse](#)

666 The GetVerifyResults operation retrieves the verification results for the specified token. This operation is  
667 used in conjunction with the [Verify](#) operation. If the Verify operation is implemented as an asynchronous  
668 service, the implementing system returns a token and the GetVerifyResults operation is used to poll for  
669 the results of the original Verify request.

670 If the service provider implements an asynchronous Verifyoperation, then it MUST also implement the  
671 GetVerifyResults operation.

## 672 Request Message

| Field                     | Type | # | ? | Meaning  |
|---------------------------|------|---|---|--|
| GetVerifyResults          |      |   | Y | Retrieves the verification results for the specified token |
| ↑ GetVerifyResultsRequest |      | 1 | Y |  |

| Field                      | Type                                      | #    | ? | Meaning  |
|----------------------------|---|------|---|--|
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "GetVerifyResults".   |
| ↑ Token                    | <a href="#">TokenType</a>                 | 1    | Y | A value used to retrieve the results of the Verify request.  |
| ↑ TokenValue               | string                                    | 1    | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time.    |
| ↑ Expiration               | date                                      | 1    | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

## 673 Response Message

| Field                             | Type                            | #    | ? | Meaning   |
|-----------------------------------|---------------------------------|------|---|---|
| GetVerifyResultsResponse          |                                 |      | Y | The response to a GetVerifyResults operation.                 |
| ↑ GetVerifyResultsResponsePackage |                                 | 1    | Y |   |
| ↑ ResponseStatus                  | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.                            |
| ↑ Return                          | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                         | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↑ ReturnData                      | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

| Field   | Type                  | #    | ? | Meaning  |
|---------|-----------------------|------|---|--|
| ↑ Match | boolean               | 0..1 | N | Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR. |
| ↑ Score | <a href="#">Score</a> | 0..1 | N | The score if the biometric information matched.  |

674 **4.2.5 Identify**

675 [IdentifyRequest](#)

676 [IdentifyResponse](#)

677 The Identify operation performs an identification function according to system requirements and/or  
678 resources.If the Identify operation is implemented as a synchronous service, the implementing system  
679 immediately processes the request and returns the results in the Return Data parameter. If the Identify  
680 operation is implemented as an asynchronous service, the implementing system returns a token in the  
681 Return Data parameter, which is an indication that the request is being handled asynchronously. In this  
682 case, the [GetIdentifyResults](#) operation is used to poll for the results of the Identify request.

683 **Request Message**

| Field                      | Type                                      | #    | ? | Meaning   |
|----------------------------|---|------|---|---|
| Identify                   |   |      | Y | Performs an identification function.                                  |
| IdentifyRequest            |   | 1    | Y |   |
| ↑ GenericRequestParameters | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester. |
| ↑ Application              | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.                                |
| ↑ ApplicationUser          | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.        |
| ↑ BIASOperationName        | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "Identify".    |
| ↑ ProcessingOptions        | <a href="#">ProcessingOptionsType</a>     | 1    | Y | Options that guide how the aggregate service request is processed.    |
| ↑ Option                   | string                                    | 0..* | N | An option supported by the implementing system.                       |
| ↑ InputData                | <a href="#">InformationType</a>           | 1    | Y | Contains the input data for the aggregate services.                   |

684 **Response Message**

| Field                     | Type                            | #    | ? | Meaning   |
|---------------------------|---------------------------------|------|---|---|
| IdentifyResponse          |                                 |      | Y | The response to an Identify operation.                        |
| ↳ IdentifyResponsePackage |                                 | 1    | Y |   |
| ↳ ResponseStatus          | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.                            |
| ↳ Return                  | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↳ Message                 | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↳ ReturnData              | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

685 **4.2.6 RetrieveInformation**

686 [RetrieveInformationRequest](#)

687 [RetrieveInformationResponse](#)

688 The RetrieveInformation operation retrieves requested information about a subject, or in an encounter-  
 689 centric model about an encounter. In a person-centric model, this operation can be used to retrieve both  
 690 biographic and biometric information for a subject record. In an encounter-centric model, this operation  
 691 can be used to retrieve biographic and/or biometric information for either a single encounter or all  
 692 encounters. Either a subject ID or encounter ID MUST be specified.

693 **Request Message**

| Field                        | Type                                      | #    | ? | Meaning   |
|------------------------------|---|------|---|---|
| RetrieveInformation          |   |      | Y | Retrieves requested information about a subject or encounter.                 |
| ↳ RetrieveInformationRequest |   | 1    | Y |   |
| ↳ GenericRequestParameters   | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.         |
| ↳ Application                | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.  |
| ↳ ApplicationUser            | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.                |
| ↳ BIASOperationName          | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "RetrieveInformation". |

| Field               | Type                                  | #    | ? | Meaning   |
|---------------------|---------------------------------------|------|---|---|
| ↑ ProcessingOptions | <a href="#">ProcessingOptionsType</a> | 1    | Y | Options that guide how the aggregate service request is processed, and MAY identify what type(s) of information should be returned. |
| ↑ Option            | string                                | 0..* | N | An option supported by the implementing system.   |
| ↑ Identity          | <a href="#">BIASIdentity</a>          | 1    | Y | Includes the identifier of the subject or encounter.  |
| ↑ SubjectID         | <a href="#">BIASIDType</a>            | 0..1 | C | A system unique identifier for a subject. Required if an Encounter ID is not provided.  |
| ↑ EncounterID       | <a href="#">BIASIDType</a>            | 0..1 | C | The identifier of an encounter associated with the subject. Required if a Subject ID is not provided.                               |

694 **Response Message**

| Field                                | Type                            | #    | ? | Meaning   |
|--------------------------------------|---------------------------------|------|---|---|
| RetrieveInformationResponse          |                                 |      | Y | Response to a RetrieveInformation operation.                  |
| ↑ RetrieveInformationResponsePackage |                                 | 1    | Y |   |
| ↑ ResponseStatus                     | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.                            |
| ↑ Return                             | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation. |
| ↑ Message                            | string                          | 0..1 | N | A short message corresponding to the return code.             |
| ↑ ReturnData                         | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.                    |

695 **4.2.7 Verify**

696 [VerifyRequest](#)

697 [VerifyResponse](#)

698 The Verify operation performs a 1:1 verification function according to system requirements and/or  
 699 resources. Either the Identity Claim or Reference BIR input parameters are REQUIRED. If the Verify  
 700 operation is implemented as a synchronous service, the implementing system immediately processes the  
 701 request and returns the results in the Return Data parameter. If the Verify operation is implemented as an  
 702 asynchronous service, the implementing system returns a token in the Return Data parameter, which is  
 703 an indication that the request is being handled asynchronously. In this case, the [GetVerifyResults](#)  
 704 operation is used to poll for the results of the Verify request.

705 **Request Message**

| Field                          | Type                                      | #    | ? | Meaning  |
|--------------------------------|---|------|---|--|
| Verify                         |   |      | Y | Performs a 1:1 verification function.  |
| ↳ VerifyRequest                |   | 1    | Y |  |
| ↳ GenericRequestParameter<br>s | <a href="#">GenericRequestParameters</a>  | 0..1 | N | Common request parameters that can be used to identify the requester.  |
| ↳ Application                  | <a href="#">ApplicationIdentifier</a>     | 0..1 | N | Identifies the requesting application.   |
| ↳ ApplicationUser              | <a href="#">ApplicationUserIdentifier</a> | 0..1 | N | Identifies the user or instance of the requesting application.   |
| ↳ BIASOperationName            | string                                    | 0..1 | N | Identifies the BIAS operation that is being requested: "Verify".   |
| ↳ ProcessingOptions            | <a href="#">ProcessingOptionsType</a>     | 1    | Y | Options that guide how the aggregate service request is processed.   |
| ↳ Option                       | string                                    | 0..* | N | An option supported by the implementing system.  |
| ↳ InputData                    | <a href="#">InformationType</a>           | 1    | Y | Contains the input data for the aggregate services.  |
| ↳ Identity                     | <a href="#">BIASIdentity</a>              | 1    | Y | Includes either the Identity Claim or Reference BIR.   |
| ↳ IdentityClaim                | <a href="#">BIASIDType</a>                | 0..1 | C | An identifier by which a subject is known to a particular gallery or population group.<br>Required if a Reference BIR is not provided. |
| ↳ BiometricData                | <a href="#">BIASBiometricDataType</a>     | 0..1 | N | An Identity's biometric data.  |



| Field          | Type                           | #    | ? | Meaning  |
|----------------|--------------------------------|------|---|--|
| ↑ ReferenceBIR | <a href="#">CBEFF_BIR_Type</a> | 0..1 | C | Maps to specific INCITS BIAS elements as required by that specification.<br>Required if an Identity Claim is not provided.             |
| ↑ GalleryID    | <a href="#">BIASIDType</a>     | 0..1 | C | The identifier of the gallery or population group of which the subject must be a member.<br>Required if an Identity Claim is provided. |

706 **Response Message**

| Field                   | Type                            | #    | ? | Meaning  |
|-------------------------|---------------------------------|------|---|--|
| VerifyResponse          |                                 |      | Y | The response to a Verify operation.  |
| ↑ VerifyResponsePackage |                                 | 1    | Y |  |
| ↑ ResponseStatus        | <a href="#">ResponseStatus</a>  | 1    | Y | Returned status for the operation.   |
| ↑ Return                | <a href="#">ReturnCode</a>      | 1    | Y | The return code indicates the return status of the operation.  |
| ↑ Message               | string                          | 0..1 | N | A short message corresponding to the return code.  |
| ↑ ReturnData            | <a href="#">InformationType</a> | 0..1 | N | Contains the output data for the response.   |
| ↑ Match                 | boolean                         | 0..1 | N | Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR. |
| ↑ Score                 | <a href="#">Score</a>           | 0..1 | N | The score if the biometric information matched.  |

707

708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727

## 5 Message structure and rules

BIAS operations and data elements are defined in XML in the INCITS 422 BIAS standard. This OASIS standard further specifies the full XML schema (see AnnexA) and specifies how this XML is packaged and exchanged as SOAP messages.

Annex A provides a WSDL of operations and structures aggregated from all the conformance classes, both synchronous and asynchronous. A specific implementation's WSDL must only expose its respective operations and structures. For example, for a Class 5-only conformant implementation, all of the primitive operations must not be exposed as operations (with the exception of QueryCapabilities) unless that functionality is supported. Additionally, the WSDL exposed by an implementation shall not contain instances of `xsd:any`, `xsd:anyType`, or `xsd:anyAttribute`; these instances must be replaced with explicit schema contents. An example is the XML complex type, `InformationType`, which has `xsd:any` as its only child. This type is used to represent implementation-specific input data and return data. The children of `InformationType` must be replaced with explicit content. Doing so removes the ability to transmit unexpected or arbitrary data. Also, it provides a clear definition of information that a client needs to provide to the server, or expect to receive, to optimally perform an operation.

SOAP 1.1 messages consist of three elements: an envelope, header data, and a message body. BIAS request-response elements MUST be enclosed within the SOAP message body. The general structure of the BIAS SOAP message is shown in Figure 4, below. The data model for BIAS is addressed in Section 3 and BIAS messages in Section 4.

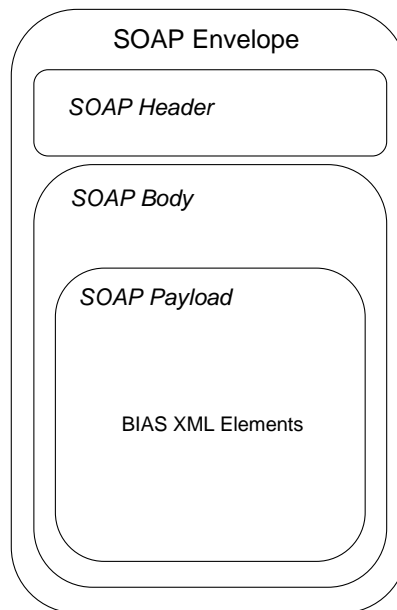


Figure 4. BIAS SOAP Structure

728  
729  
730  
731  
732  
733  
734  
735  
736  
737

Biometric data, regardless of native format, is carried as a binary structure. As such, options exist on how this data is carried within the SOAP structure. It can be carried as embedded Base-64 objects or [XOP] can be used – this standard allows for either method (See section 5.3).

### 5.1 Purpose and constraints

This document defines a SOAP profile describing how the XML elements defined in INCITS 442 are to be used as the payload of a SOAP message and the rules for structuring and exchanging such messages. Philosophical tenets include:

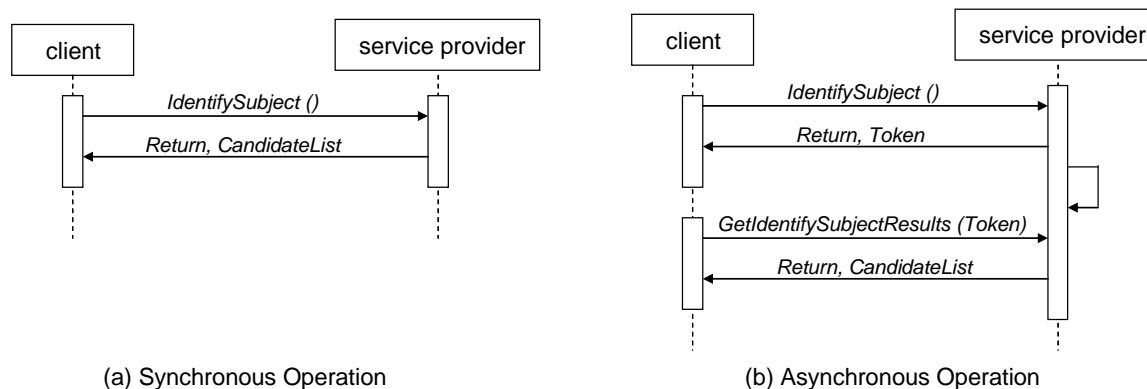
- 738 • SOAP messages will carry BIAS XML [XML 10] payloads.
- 739 • SOAP messages will follow WS-I and will deviate only when absolutely necessary.
- 740 • Message structures and interchanges will be kept as simple as possible – “nice to have”
- 741 features will be addressed in future revisions.
- 742 • XML schemas will be produced based on INCITS 442.
- 743 • BIAS will support a broad range of application domains.
- 744 • BIAS will allow for a variety of biometric and biographic data formats to be used
- 745 • Only the SOAP messaging will be defined – no message protocols or client/server agents
- 746 will be defined.
- 747 • Basic usage/formatting rules (beyond WS-I) will be defined.
- 748 • Existing biometric and Web services standards will be leveraged wherever possible.
- 749 • Sample WSDL and use cases will be provided as an aid in implementation.
- 750 • Use of basic SOAP will allow all other compatible WS\* standards (and discovery
- 751 mechanisms) to be used in conjunction with BIAS messaging.
- 752 • BIAS will support both secure (i.e., using existing security mechanisms such as WS-
- 753 Security, SAML, etc.) and non-secure implementations.
- 754 • Generic biometric operations will be defined – use of biometrics within a Web services
- 755 authentication protocol is not addressed.
- 756 • OASIS namespace rules will be followed, though some external schemas MAY also be
- 757 referenced.

## 758 5.2 Message requirements

759 BIAS SOAP messages MUST conform to [WS-I-Basic] and [WS-I-Bind]. A single BIAS SOAP message  
 760 MUST contain only one BIAS service request (or single BIAS service response). Binary components of  
 761 BIAS messages are already Base-64 encoded and therefore do not need to be conveyed as SOAP  
 762 attachments (though XOP MAY be utilized).

763 The system model used for BIAS conversations over SOAP is a simple request-response model. BIAS  
 764 comprises both synchronous and asynchronous operations, with the majority being of the former type.  
 765 Asynchronous operations are implemented through message pairs. That is, there are separate messages  
 766 to request the operation and to request the results of the operation. These have been defined for those  
 767 operations that are likely to take significant time to complete. For example, an identify operation can be  
 768 implemented as either a synchronous or asynchronous service as follows:

769



770

771

772

773

**Figure 5. Example of Synchronous and Asynchronous BIAS Operations**

774 The basic process for using SOAP for BIAS operations is:

- 775 1. A system entity acting as a BIAS requester transmits a BIAS request element within the body of a  
776 SOAP message to a system entity acting as a BIAS responder. The BIAS requester MUST NOT  
777 include more than one BIAS request per SOAP message or include any additional XML elements  
778 in the SOAP body.
- 779 2. The BIAS responder MUST return either a BIAS response element within the body of another  
780 SOAP message or generate a SOAP fault. The BIAS responder MUST NOT include more than  
781 one BIAS response per SOAP message or include any additional XML elements in the SOAP  
782 body. If a BIAS responder cannot, for some reason, process a BIAS request, it MUST generate a  
783 SOAP fault. (SOAP 1.1 faults and fault codes are discussed in [SOAP11] section 5.1.)
- 784 3. On receiving a BIAS response in a SOAP message, the BIAS requester MUST NOT send a fault  
785 code or other error messages to the BIAS responder. Since the format for the message  
786 interchange is a simple request-response pattern, adding additional items such as error  
787 conditions would needlessly complicate the protocol.

788 SOAP 1.1 also defines an optional data encoding system. This system is not used within the BIAS SOAP  
789 binding. This means that BIAS messages can be transported using SOAP without re-encoding from the  
790 “standard” BIAS schema to one based on the SOAP encoding.

791 NOTE: [SOAP11] references an early draft of the XML Schema specification including an  
792 obsolete namespace. BIAS requesters SHOULD generate SOAP documents referencing only the  
793 final XML schema namespace. BIAS responders MUST be able to process both the XML schema  
794 namespace used in [SOAP11] as well as the final XML schema namespace.

## 795 5.3 Handling binary data

796 BIAS messages frequently contain binary data (e.g., biometric data, scanned identity documents, etc.).  
797 Two methods are provided for dealing with this:

- 798 • Embedded Base64 encoding
- 799 • XOP [XOP]

800 Use of SOAP with Attachments (SWA) is deprecated.

### 801 5.3.1 Base64 encoding

802 This method is the default method for including binary data. Binary data is Base64 encoded and included  
803 between the tags in the XML SOAP body for the appropriate data elements. Data elements using this  
804 method are indicated as such in the schema.

805 As an example, the CBEFF\_BIR\_Type includes, as one of the BIR types, BinaryBIR of type  
806 base64binary.

```
807  
808 <xsd:element name="BinaryBIR" type="xs:base64Binary" />
```

809  
810 However, even an XML\_BIR as defined within [CBEFF3], contains a biometric data block (BDB) which  
811 may be entirely binary (most common),

```
812  
813 <xs:element name="bdb" type="xs:base64Binary" minOccurs="0"/>
```

814  
815 or contain an element which is binary (e.g., an image within an XML BDB).

### 816 5.3.2 Use of XOP

817 When XOP is used, the binary content is replaced with a reference (URI) to an attachment (i.e., MIME)  
818 which contains that “stripped” content via an xop:include. The advantage of this method is overall

819 message size during transmission since the overhead of the embedded Base64 is not present (since the  
820 MIME attachment contains the native binary format).

821 Use of XOP is generally transparent to the developer, other than in how they configure their toolset. Most  
822 frameworks support this; however, there is a possibility of mismatch if the transmitter supports and uses  
823 XOP but the receiver does not.

## 824 5.4 Discovery

825 BIAS implementers (service providers) MUST provide WSDL [[WSDL11](#)] to describe their  
826 implementations. This WSDL MAY or may not be made public via a standard discovery mechanism  
827 (such as UDDI) or other method.

828 In addition, it is REQUIRED that the BIAS implementation include the [QueryCapabilities](#) operation to  
829 provide dynamic information regarding BIAS capabilities, options, galleries, etc. that are supported.

## 830 5.5 Identifying operations

831 Receivers of BIAS SOAP messages require a method of easily identifying the operation being requested  
832 (or response being provided). This SHOULD be possible without the receiver needing to infer it from the  
833 sum of the elements provided within the body of the SOAP message. The BIAS SOAP profile allows for  
834 two methods of identifying BIAS operations:

- 835 • Explicit named element in body of the SOAP message
- 836 • Use of WS-Addressing Action element

### 837 5.5.1 Operation name element

838 The BIAS message sender (requester) will include within the body of the BIAS SOAP message an XML  
839 element <BIASOperationName>. The receiver (service provider) can search for this tag within a received  
840 BIAS SOAP message to determine what operation is being requested. There is no requirement related to  
841 the ordering of this element within the message, though it is RECOMMENDED that it be included early in  
842 the message to aid in human readability.

843 An example of this method for the CreateSubject operation is shown below:

```
844
845 POST /bias HTTP/1.1
846 Host: www.acme.com
847 Content-Type: application/soap+xml; charset="utf-8"
848 Content-Length: nnnn
849 SOAPAction: ""
850 <?xml version="1.0"?>
851 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
852   <soap:Body>
853     <tns:CreateSubjectRequest
854       xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
855       <tns:GenericRequestParameters>
856         <tns:Application>BIAS Application</tns:Application>
857         <tns:ApplicationUser>BIAS User</tns:ApplicationUser>
858         <tns:BIASOperationName>CreateSubject</tns:BIASOperationName
859 >
860       </tns:GenericRequestParameters>
861     <tns:Identity>
```

```
862         <tns:SubjectID>123456789</tns:SubjectID>
863     </tns:Identity>
864 </tns:CreateSubjectRequest>
865 </soap:Body>
866 </soap:Envelope>
```

## 867 5.5.2 WS-Addressing Action

868 WS-Addressing [WS-Addr] provides a mechanism for including action information inside any SOAP  
869 message. The information is in the SOAP Header. The WS-Addressing 'Action' element is used to  
870 indicate the intent of the message. The value is a URI/IRI identifying that intent; however, there are no  
871 restrictions on the format or specificity of the URI/IRI nor a requirement that it can be resolved. Adoption  
872 of this option also requires that the WS-Addressing 'To', 'ReplyTo', and 'MessageID' elements are  
873 supplied, as they are mandatory elements in a request-reply message pattern as used within BIAS.  
874 Response messages would also need to use WS-Addressing, requiring the 'To' (matching the 'ReplyTo'  
875 element in the request), 'RelatesTo' (matching the 'MessageID' element in the request), and  
876 'RelationshipType' (default value to "wsa:Reply") elements.

877 Use of WS-Addressing is OPTIONAL in this profile as is this method of using the 'Action' field for this  
878 purpose. However, when BIAS is used within an environment using WS-Addressing, it is  
879 RECOMMENDED that this approach for use of the 'Action' field to carry the BIAS operation name is  
880 employed, either alone or in combination with the BIASOperationName approach described in section  
881 5.5.1.

882 An example for a message request for the CreateSubject operation would look like the following:

```
883
884     POST /bias HTTP/1.1
885     Host: www.acme.com
886     Content-Type: application/soap+xml; charset="utf-8"
887     Content-Length: nnnn
888     SOAPAction: ""
889     <?xml version="1.0"?>
890     <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
891         xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addr
892         essing">
893         <soap:Header>
894             <wsa:MessageID>some-ID</wsa:MessageID>
895             <wsa:ReplyTo>
896                 <wsa:Address>response-URI</wsa:Address>
897             </wsa:ReplyTo>
898             <wsa:To>destination-URI</wsa:To>
899             <wsa:Action>CreateSubject</wsa:Action>
900         </soap:Header>
901         <soap:Body>
902             <tns:CreateSubjectRequest
903
904             xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/" />
905         </soap:Body>
906     </soap:Envelope>
```

907

## 908 **5.6 Security**

909 The end-points that exchange SOAP messages (or handle the contents of the BIAS operations) are  
910 expected to be protected and trusted such that message-level security mechanisms may not be required.  
911 The use of SSL (HTTPS) or VPN technology that provides end-point to end-point security is  
912 RECOMMENDED and MAY be sufficient in some cases. Other mechanisms such as Signed XML or  
913 WSS [WSS] could also be implemented.

914 Unless stated otherwise, the following security statements apply to all BIAS bindings.

### 915 **5.6.1 Use of SSL 3.0 or TLS 1.0**

916 Unless otherwise specified, in any BIAS binding's use of SSL 3.0 [SSL3] or TLS1.0 [RFC2246], servers  
917 MUST authenticate clients using a X.509 v3 certificate [X509]. The client MUST establish server identity  
918 based on contents of the certificate (typically through examination of the certificate's subject DN field,  
919 subjectAltName attribute, etc.).

920 Use of transport level security in the form of SSL or TLS is OPTIONAL but highly RECOMMENDED. Use  
921 of these mechanisms alone may not be sufficient for end-to-end integrity and confidentiality, however  
922 (see 5.6.3 and 5.6.4 below).

### 923 **5.6.2 Data Origin Authentication**

924 Authentication of both the BIAS requester and the BIAS responder associated with a message is  
925 OPTIONAL and depends on the environment of use: Authentication mechanisms available at the SOAP  
926 message exchange layer or from the underlying substrate protocol (for example, in many bindings the  
927 SSL/TLS or HTTP protocol) MAY be utilized to provide data origin authentication.

928 Transport authentication will not meet end-to-end origin authentication requirements in bindings where  
929 the BIAS SOAP message passes through an intermediary – in this case, message authentication is  
930 RECOMMENDED.

931 Note that SAML [SAML] MAY be used as the mechanism for parties to authenticate to one another.

### 932 **5.6.3 Message Integrity**

933 Message integrity of both BIAS requests and BIAS responses is OPTIONAL and depends on the  
934 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP  
935 message exchange layer MAY be used to ensure message integrity.

936 Transport integrity will not meet end-to-end integrity requirements in bindings where the BIAS SOAP  
937 message passes through an intermediary – in this case, message integrity is RECOMMENDED.

### 938 **5.6.4 Message Confidentiality**

939 Message confidentiality of both BIAS requests and BIAS responses is OPTIONAL and depends on the  
940 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP  
941 message exchange layer MAY be used to ensure message confidentiality.

942 Transport confidentiality will not meet end-to-end confidentiality requirements in bindings where the BIAS  
943 SOAP message passes through an intermediary.

944 NOTE: Biometric and biographic data is likely to contain personal information the confidentiality of  
945 which SHOULD be protected accordingly. See INCITS 442, section 6.3 for further discussion.

### 946 **5.6.5 CBEFF BIR security features**

947 Within BIAS, biometric data is transferred within a CBEFF BIR (as defined in ISO/IEC 19785-1). CBEFF  
948 provides for the optional encryption of the Biometric Data Block (BDB) of the BIR and for the integrity of  
949 the entire BIR. If implemented, this is indicated in the BIR header. The BIR structure defines an optional

950 Security Block which MAY contain a digital signature (or message authentication code), encryption  
951 parameters (e.g., key name, algorithm, etc.), and/or other security related data. Such protections are  
952 associated with an individual BIR and are separate from any other protections provided at the message  
953 level.

## 954 **5.6.6 Security Considerations**

955 Before deployment, each combination of authentication, message integrity, and confidentiality  
956 mechanisms SHOULD be analyzed for vulnerability in the context of the specific protocol exchange and  
957 the deployment environment.

958 Special care should be given to the impact of possible caching on security.

959 IETF RFC 2617 [RFC2617] describes possible attacks in the HTTP environment when basic or message  
960 digest authentication schemes are used.

961 Many of the security considerations identified in [SAML SEC] MAY also apply.

962 ISO/IEC 19092 [BIO SEC] describes a security framework for biometric systems including a minimum set  
963 of security requirements addressing integrity, authenticity, and confidentiality of biometric information  
964 during transmission and storage. These SHOULD be considered as part of an overall risk management  
965 approach.

966 NOTE: The requirements of ISO/IEC 19092, though useful across many application domains, are  
967 required for most biometric system implementations in the financial services environment.  
968 Application of this standard would make the requirements of sections 5.5.3 through 5.5.5  
969 mandatory rather than optional. This is highly RECOMMENDED for any high security environment  
970 or where privacy concerns exist.

## 971 **5.6.7 Security of Stored Data**

972 This specification does not address security considerations for stored data. It is the purview of the BIAS  
973 service provider to implement security mechanisms and protect data at rest as per their own security  
974 policies.

## 975 **5.6.8 Key Management**

976 This specification does not address key management considerations with respect to implementation of  
977 cryptographic security mechanisms (e.g., for authenticity, integrity, or confidentiality).

## 978 **5.7 Use with other WS\* standards**

979 The intent of specifying SOAP bindings for BIAS messages is to enable the full range of existing Web  
980 services standards to be able to be applied. Some may be normative while others can be optionally  
981 applied (i.e., WS-Security, WS-Addressing). Still others may require additional profiling to be used in an  
982 interoperable manner (e.g., WS-Notification); this is left to a future revision. However, the intent is to avoid  
983 specifying anything in the first, base version that would preclude the use of such standards in the future.

## 984 **5.8 Tailoring**

985 This standard provides for a common method of implementing biometric Web services; however, it does  
986 not guarantee interoperability in a specific application. In some cases further tailoring or profiling of this  
987 standard may be required in order to further constrain the implementation options available.

988 NOTE: As an example, BIAS allows for a number of different biographic and biometric data formats  
989 to be used, whereas a given application/domain MAY wish to limit this to a small set or just one of  
990 each type. Other examples (not comprehensive) include:

- 991 • Identification of a subset of BIAS operations to be used
- 992 • Specification of security features to be implemented (e.g., SSL, CBEFF BIR encryption, etc.)
- 993 • Choice of operation name identification method



- 994 • Choice of BIR type to be used (XML, non-XML, or URI)
- 995 • Further definition of aggregate operations
- 996 • Use (or not) of the encounter model
- 997 • Use (or not) of asynchronous operations
- 998 • Process sequences
- 999 • Implementation specific values (e.g., Transform operations/controls)

---

## 1000 6 Error handling

1001 There are two levels of errors that can be returned in an error response: system and service errors.

- 1002 • System-level errors occur when the implementing system cannot service a request. They could
- 1003 result due to an internal logic error or because the implementing system does not support a
- 1004 particular request.
- 1005 • Service-level errors occur when there is a problem transmitting or representing the service
- 1006 request. They could result due to an invalid service request or because of a communications
- 1007 error.

1008 The INCITS BIAS standard defines the error condition codes for system-level errors.

- 1009 • If successful, a response message (containing a return code) will be generated.
- 1010 • If unsuccessful, a SOAP fault message (containing a fault code) will be generated.

### 1011 6.1 BIAS operation return codes

1012 If a BIAS operation is successful, a response (service output) will be sent to the requester by the service  
1013 provider. Each response message contains a response status (see section 3.2.37) and return code (see  
1014 section 3.2.38) along with any response data as defined for that operation, if any. A response code of '0'  
1015 indicates success.

### 1016 6.2 SOAP fault codes

1017 If a BIAS operation is unsuccessful, no BIAS response message is sent. Instead a SOAP fault message  
1018 is returned.

1019 Every Web service (operation) described in the BIAS WSDL may result in a fault message that will be  
1020 returned in the response by the service provider in the event of an error. The fault message contains a  
1021 FaultCode element as defined by the SOAP 1.1 specification (see section 3.2.5). The fault message  
1022 MUST contain a Detail element in a common format, as described by the BIASFault element (see section  
1023 3.2.6).

1024 The schema provided in Annex A defines "BIASFaultCode" and "BIASFaultDetail" types as well as  
1025 "BIASFault", "BIASFaultType", "BIASFaultMessage" and "BIASFaultDescription" elements.

1026 The list of defined BIAS fault codes is provided in section 3.2.5. Note that BIAS service providers MAY  
1027 define additional fault codes unique to their service.

1028 NOTE: See also section 5.2 for additional information on message returns and faults.

---

## 1029 7 Conformance

1030 Implementations claiming conformance to this standard, MUST implement, at a minimum, all mandatory  
1031 requirements and provisions set forth in Clauses 3, 4, 5 and 6. If such implementations claim  
1032 conformance to any OPTIONAL requirements and provisions stated in Clauses 3, 4, 5 and 6, these  
1033 requirements and provisions MUST be implemented as set forth in these Clauses.

1034 INCITS 442 [INCITS-BIAS] (Annex A) specifies five BIAS conformance classes. For each class, a set of  
1035 mandatory BIAS operations is identified in order for implementations (BIAS service providers) to claim  
1036 conformance. These categories are:

- 1037 • Class 1: Full Primitive Services Implementation
- 1038 • Class 2: Full Aggregate Services Implementation
- 1039 • Class 3: Limited Primitive Services Implementation
- 1040 • Class 4: Minimum Primitive Services Implementation
- 1041 • Class 5: Minimum Aggregate Services Implementation

1042 In addition, the minimum capability information to be returned in response to a Query Capabilities request  
1043 (the only mandatory BIAS operation across all 5 classes) is specified for each class.

1044 These conformance classes and their associated requirements apply to this BIAS SOAP Profile.

1045 There are no minimum set of operations required to be implemented by BIAS requesters; however, any  
1046 operations implemented must conform to the requirements of Clauses 3 and 4 and those requirements  
1047 within Clause 5 that are mandatory and are not specific to BIAS responders.

---

## 1048 Annex A. XML Schema

```
1049 <?xml version="1.0" encoding="UTF-8"?>
1050 <!--
1051     Document   : BIAS.xml
1052     Created on  : December 17, 2009, 9:31 AM
1053     Last Edit   : February 12, 2014 by Kevin Mangold, National Institute of Standards and Technology
1054     Description:
1055         This document represents the WSDL definitions of BIAS Profile.
1056 -->
1057 <wSDL:definitions
1058     xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/"
1059     xmlns:soap="http://schemas.xmlsoap.org/wSDL/soap/"
1060     xmlns:http="http://schemas.xmlsoap.org/wSDL/http/"
1061     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1062     xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
1063     xmlns:mime="http://schemas.xmlsoap.org/wSDL/mime/"
1064     xmlns:oasis_cbeff="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
1065     xmlns:tNS="http://docs.oasis-open.org/bias/ns/bias-1.0/"
1066     targetNamespace="http://docs.oasis-open.org/bias/ns/bias-1.0/">
1067
1068     <wSDL:types>
1069         <xsd:schema elementFormDefault="qualified" targetNamespace="http://docs.oasis-
1070 open.org/bias/ns/bias-1.0/">
1071             <xsd:import namespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
1072 schemaLocation="cbeff.xsd"/>
1073             <xsd:complexType name="AggregateRequestTemplate">
1074                 <xsd:annotation>
1075                     <xsd:documentation>Base template for BIAS aggregate service requests.</xsd:documentation>
1076                 </xsd:annotation>
1077                 <xsd:complexContent>
1078                     <xsd:extension base="tNS:RequestTemplate">
1079                         <xsd:sequence>
1080                             <xsd:element name="ProcessingOptions" type="tNS:ProcessingOptionsType" minOccurs="1"
1081 maxOccurs="1">
1082                                 <xsd:annotation>
1083                                     <xsd:documentation>Options that guide how the aggregate service request is
1084 processed.</xsd:documentation>
1085                                 </xsd:annotation>
1086                             </xsd:element>
1087                             <xsd:element name="InputData" type="tNS:InformationType" minOccurs="1" maxOccurs="1">
1088                                 <xsd:annotation>
```

```

1089         <xsd:documentation>Contains the input data for the aggregate service
1090 request.</xsd:documentation>
1091     </xsd:annotation>
1092 </xsd:element>
1093 </xsd:sequence>
1094 </xsd:extension>
1095 </xsd:complexContent>
1096 </xsd:complexType>
1097 <xsd:complexType name="AggregateResponseTemplate">
1098     <xsd:annotation>
1099         <xsd:documentation>Base template for BIAS aggregate service responses.</xsd:documentation>
1100     </xsd:annotation>
1101     <xsd:complexContent>
1102         <xsd:extension base="tns:ResponseTemplate">
1103             <xsd:sequence>
1104                 <xsd:element name="ReturnData" type="tns:InformationType" minOccurs="0" maxOccurs="1">
1105                     <xsd:annotation>
1106                         <xsd:documentation>Contains the output data for the aggregate service
1107 response.</xsd:documentation>
1108                     </xsd:annotation>
1109                 </xsd:element>
1110             </xsd:sequence>
1111         </xsd:extension>
1112     </xsd:complexContent>
1113 </xsd:complexType>
1114 <xsd:simpleType name="ApplicationIdentifier">
1115     <xsd:annotation>
1116         <xsd:documentation>Identifies an application.</xsd:documentation>
1117     </xsd:annotation>
1118     <xsd:restriction base="xsd:string">
1119         <xsd:minLength value="1"/>
1120         <xsd:maxLength value="255"/>
1121     </xsd:restriction>
1122 </xsd:simpleType>
1123 <xsd:simpleType name="ApplicationUserIdentifier">
1124     <xsd:annotation>
1125         <xsd:documentation>Identifies an application user or instance.</xsd:documentation>
1126     </xsd:annotation>
1127     <xsd:restriction base="xsd:string">
1128         <xsd:minLength value="1"/>
1129         <xsd:maxLength value="255"/>
1130     </xsd:restriction>
1131 </xsd:simpleType>

```

```

1132     <xsd:complexType name="BIASBiometricDataType">
1133         <xsd:annotation>
1134             <xsd:documentation>Wraps the various BIAS biometric types.</xsd:documentation>
1135         </xsd:annotation>
1136         <xsd:sequence>
1137             <xsd:element name="BIRList" type="tns:CBEFF_BIR_ListType" minOccurs="0" maxOccurs="1">
1138                 <xsd:annotation>
1139                     <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
1140                 </xsd:annotation>
1141             </xsd:element>
1142             <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
1143                 <xsd:annotation>
1144                     <xsd:documentation>Contains biometric information in either a non-XML and an XML
1145 representation.</xsd:documentation>
1146                 </xsd:annotation>
1147             </xsd:element>
1148             <xsd:element name="InputBIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
1149                 <xsd:annotation>
1150                     <xsd:documentation>Maps to specific INCITS BIAS elements as required by that
1151 specification.</xsd:documentation>
1152                 </xsd:annotation>
1153             </xsd:element>
1154             <xsd:element name="ReferenceBIR" type="tns:CBEFF_BIR_Type" minOccurs="0"
1155 maxOccurs="1">
1156                 <xsd:annotation>
1157                     <xsd:documentation>Maps to specific INCITS BIAS elements as required by that
1158 specification.</xsd:documentation>
1159                 </xsd:annotation>
1160             </xsd:element>
1161             <xsd:element name="BiometricDataList" type="tns:BiometricDataListType" minOccurs="0"
1162 maxOccurs="1">
1163                 <xsd:annotation>
1164                     <xsd:documentation>A list of biometric data elements.</xsd:documentation>
1165                 </xsd:annotation>
1166             </xsd:element>
1167         </xsd:sequence>
1168     </xsd:complexType>
1169     <xsd:simpleType name="BIASFaultCode">
1170         <xsd:annotation>
1171             <xsd:documentation> </xsd:documentation>
1172         </xsd:annotation>
1173         <xsd:restriction base="xsd:string">
1174             <xsd:enumeration value="UNKNOWN_ERROR">
1175                 <xsd:annotation>

```

```

1176     <xsd:documentation>The service failed for an unknown reason.</xsd:documentation>
1177     </xsd:annotation>
1178 </xsd:enumeration>
1179 <xsd:enumeration value="UNSUPPORTED_CAPABILITY">
1180     <xsd:annotation>
1181         <xsd:documentation>A requested capability is not supported by the service
1182 implementation.</xsd:documentation>
1183     </xsd:annotation>
1184 </xsd:enumeration>
1185 <xsd:enumeration value="INVALID_INPUT">
1186     <xsd:annotation>
1187         <xsd:documentation>The data in a service input parameter is invalid.</xsd:documentation>
1188     </xsd:annotation>
1189 </xsd:enumeration>
1190 <xsd:enumeration value="BIR_QUALITY_ERROR">
1191     <xsd:annotation>
1192         <xsd:documentation>Biometric sample quality is too poor for the service to
1193 succeed.</xsd:documentation>
1194     </xsd:annotation>
1195 </xsd:enumeration>
1196 <xsd:enumeration value="INVALID_BIR">
1197     <xsd:annotation>
1198         <xsd:documentation>The input BIR is empty or in an invalid or unrecognized
1199 format.</xsd:documentation>
1200     </xsd:annotation>
1201 </xsd:enumeration>
1202 <xsd:enumeration value="BIR_SIGNATURE_FAILURE">
1203     <xsd:annotation>
1204         <xsd:documentation>The service could not validate the signature, if used, on the input
1205 BIR.</xsd:documentation>
1206     </xsd:annotation>
1207 </xsd:enumeration>
1208 <xsd:enumeration value="BIR_DECRYPTION_FAILURE">
1209     <xsd:annotation>
1210         <xsd:documentation>The service could not decrypt an encrypted input
1211 BIR.</xsd:documentation>
1212     </xsd:annotation>
1213 </xsd:enumeration>
1214 <xsd:enumeration value="INVALID_ENCOUNTER_ID">
1215     <xsd:annotation>
1216         <xsd:documentation>The input encounter ID is empty or in an invalid
1217 format.</xsd:documentation>
1218     </xsd:annotation>
1219 </xsd:enumeration>

```

1220 <xsd:enumeration value="INVALID\_SUBJECT\_ID">  
1221 <xsd:annotation>  
1222 <xsd:documentation>The input subject ID is empty or in an invalid format.</xsd:documentation>  
1223 </xsd:annotation>  
1224 </xsd:enumeration>  
1225 <xsd:enumeration value="UNKNOWN\_SUBJECT">  
1226 <xsd:annotation>  
1227 <xsd:documentation>The subject referenced by the input subject ID does not  
1228 exist.</xsd:documentation>  
1229 </xsd:annotation>  
1230 </xsd:enumeration>  
1231 <xsd:enumeration value="UNKNOWN\_GALLERY">  
1232 <xsd:annotation>  
1233 <xsd:documentation>The gallery referenced by the input gallery ID does not  
1234 exist.</xsd:documentation>  
1235 </xsd:annotation>  
1236 </xsd:enumeration>  
1237 <xsd:enumeration value="UNKNOWN\_ENCOUNTER">  
1238 <xsd:annotation>  
1239 <xsd:documentation>The encounter referenced by the input encounter ID does not  
1240 exist.</xsd:documentation>  
1241 </xsd:annotation>  
1242 </xsd:enumeration>  
1243 <xsd:enumeration value="UNKNOWN\_BIOGRAPHIC\_FORMAT">  
1244 <xsd:annotation>  
1245 <xsd:documentation>The biographic data format is not known or not  
1246 supported.</xsd:documentation>  
1247 </xsd:annotation>  
1248 </xsd:enumeration>  
1249 <xsd:enumeration value="UNKNOWN\_IDENTITY\_CLAIM">  
1250 <xsd:annotation>  
1251 <xsd:documentation>The identity referenced by the input identity claim does not  
1252 exist.</xsd:documentation>  
1253 </xsd:annotation>  
1254 </xsd:enumeration>  
1255 <xsd:enumeration value="INVALID\_IDENTITY\_CLAIM">  
1256 <xsd:annotation>  
1257 <xsd:documentation>The identity claim requested is already in use.</xsd:documentation>  
1258 </xsd:annotation>  
1259 </xsd:enumeration>  
1260 <xsd:enumeration value="NONEXISTANT\_DATA">  
1261 <xsd:annotation>  
1262 <xsd:documentation>The data requested for deletion does not exist.</xsd:documentation>  
1263 </xsd:annotation>



```

1264     </xsd:enumeration>
1265     </xsd:restriction>
1266 </xsd:simpleType>
1267 <xsd:complexType name="BIASFaultDetail">
1268     <xsd:annotation>
1269         <xsd:documentation>Defines the error information associated with a SOAP
1270 fault.</xsd:documentation>
1271     </xsd:annotation>
1272     <xsd:sequence>
1273         <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultType" type="tns:BIASFaultCode">
1274             <xsd:annotation>
1275                 <xsd:documentation>References an error code.</xsd:documentation>
1276             </xsd:annotation>
1277         </xsd:element>
1278         <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultMessage" type="xsd:string">
1279             <xsd:annotation>
1280                 <xsd:documentation>Provides an explanation of the fault.</xsd:documentation>
1281             </xsd:annotation>
1282         </xsd:element>
1283         <xsd:element minOccurs="0" maxOccurs="1" name="BIASFaultDescription" type="xsd:string">
1284             <xsd:annotation>
1285                 <xsd:documentation>Provides detailed information about a BIAS fault, such as trace
1286 details.</xsd:documentation>
1287             </xsd:annotation>
1288         </xsd:element>
1289     </xsd:sequence>
1290 </xsd:complexType>
1291 <xsd:complexType name="BIASIdentity">
1292     <xsd:annotation>
1293         <xsd:documentation>
1294             Defines a single element for encapsulating the data associated
1295             with an Identity. Includes the Identity's reference identifiers,
1296             biographic data, and biometric data.
1297         </xsd:documentation>
1298     </xsd:annotation>
1299     <xsd:sequence>
1300         <xsd:element name="SubjectID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1301             <xsd:annotation>
1302                 <xsd:documentation>A system unique identifier for a subject.</xsd:documentation>
1303             </xsd:annotation>
1304         </xsd:element>
1305         <xsd:element name="IdentityClaim" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1306             <xsd:annotation>

```

```

1307         <xsd:documentation>An identifier by which a subject is known to a particular gallery or
1308 population group.</xsd:documentation>
1309     </xsd:annotation>
1310 </xsd:element>
1311 <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1312     <xsd:annotation>
1313         <xsd:documentation>The identifier of an encounter associated with the subject, required for
1314 encounter-centric models.</xsd:documentation>
1315     </xsd:annotation>
1316 </xsd:element>
1317 <xsd:element name="EncounterList" type="tns:EncounterListType" minOccurs="0"
1318 maxOccurs="1">
1319     <xsd:annotation>
1320         <xsd:documentation>A list of encounters associated with a subject.</xsd:documentation>
1321     </xsd:annotation>
1322 </xsd:element>
1323 <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0"
1324 maxOccurs="1">
1325     <xsd:annotation>
1326         <xsd:documentation>An Identity's biographic data.</xsd:documentation>
1327     </xsd:annotation>
1328 </xsd:element>
1329 <xsd:element name="BiographicDataElements" type="tns:BiographicDataType" minOccurs="0"
1330 maxOccurs="1">
1331     <xsd:annotation>
1332         <xsd:documentation>An Identity's biographic data elements that are stored in the implementing
1333 system.</xsd:documentation>
1334     </xsd:annotation>
1335 </xsd:element>
1336 <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="0"
1337 maxOccurs="1">
1338     <xsd:annotation>
1339         <xsd:documentation>An Identity's biometric data.</xsd:documentation>
1340     </xsd:annotation>
1341 </xsd:element>
1342 </xsd:sequence>
1343 </xsd:complexType>
1344 <xsd:simpleType name="BIASIDType">
1345     <xsd:annotation>
1346         <xsd:documentation>A BIAS identifier</xsd:documentation>
1347     </xsd:annotation>
1348     <xsd:restriction base="xsd:string"/>
1349 </xsd:simpleType>
1350 <xsd:complexType name="BiographicDataItem Type">

```

```

1351     <xsd:annotation>
1352         <xsd:documentation>Defines a single biographic data element.</xsd:documentation>
1353     </xsd:annotation>
1354     <xsd:sequence>
1355         <xsd:element name="Name" type="xsd:string" minOccurs="1" maxOccurs="1">
1356             <xsd:annotation>
1357                 <xsd:documentation>The name of the biographic data item.</xsd:documentation>
1358             </xsd:annotation>
1359         </xsd:element>
1360         <xsd:element name="Type" type="xsd:string" minOccurs="1" maxOccurs="1">
1361             <xsd:annotation>
1362                 <xsd:documentation>The data type for the biographic data item.</xsd:documentation>
1363             </xsd:annotation>
1364         </xsd:element>
1365         <xsd:element name="Value" type="xsd:string" minOccurs="0" maxOccurs="1">
1366             <xsd:annotation>
1367                 <xsd:documentation>The value assigned to the biographic data item.</xsd:documentation>
1368             </xsd:annotation>
1369         </xsd:element>
1370     </xsd:sequence>
1371 </xsd:complexType>
1372 <xsd:complexType name="BiographicDataSetType">
1373     <xsd:annotation>
1374         <xsd:documentation>Defines a set of biographic data that is formatted according to the specified
1375 format.</xsd:documentation>
1376     </xsd:annotation>
1377     <xsd:sequence>
1378         <xsd:element name="name" type="xsd:string" minOccurs="1" maxOccurs="1">
1379             <xsd:annotation>
1380                 <xsd:documentation>The name of the biographic data format. Use these names for common
1381 formats: FBI-EFTS, FBI-EBTS, DOD-EBTS, INT-I, NIEM, xNAL, HR-XML.</xsd:documentation>
1382             </xsd:annotation>
1383         </xsd:element>
1384         <xsd:element name="version" type="xsd:string" minOccurs="0" maxOccurs="1">
1385             <xsd:annotation>
1386                 <xsd:documentation>The version of the biographic data format (e.g., "7.1" for FBI-EFTS or "2.0"
1387 for NIEM).</xsd:documentation>
1388             </xsd:annotation>
1389         </xsd:element>
1390         <xsd:element name="source" type="xsd:string" minOccurs="1" maxOccurs="1">
1391             <xsd:annotation>
1392                 <xsd:documentation>Reference to a URI/IRI describing the biographic data format. For example:
1393 (FBI-EFTS) www.fbibiospecs.org, (DOD-EBTS) www.biometrics.dod.mil, (INT-I) www.interpol.int, (NIEM)
1394 www.niem.gov, (xNAL) www.oasis-open.org, (HR-XML) www.hr-xml.org.</xsd:documentation>

```

```

1395     </xsd:annotation>
1396 </xsd:element>
1397 <xsd:element name="type" type="xsd:string" minOccurs="1" maxOccurs="1">
1398     <xsd:annotation>
1399         <xsd:documentation>The biographic data format type. Use these types for common formats:
1400 ASCII (e.g., for non-XML versions of FBI-EFTS, FBI-EBTS, DOD-EFTS, or INT-I), XML (e.g., for NIEM,
1401 xNAL, and HR-XML or future version of FBI-EBTS).</xsd:documentation>
1402     </xsd:annotation>
1403 </xsd:element>
1404 <xsd:any namespace="##any">
1405     <xsd:annotation>
1406         <xsd:documentation>Biographic data formatted according to a specific
1407 format.</xsd:documentation>
1408     </xsd:annotation>
1409 </xsd:any>
1410 </xsd:sequence>
1411 </xsd:complexType>
1412 <xsd:complexType name="BiographicDataType">
1413     <xsd:annotation>
1414         <xsd:documentation>
1415             Defines a set of biographic data elements, utilizing either the
1416             BiographicDataItem type to represent a list of elements or the
1417             BiographicDataSet type to represent a complete, formatted set of
1418             biographic information.
1419         </xsd:documentation>
1420     </xsd:annotation>
1421     <xsd:sequence>
1422         <xsd:element name="LastName" type="xsd:string" minOccurs="0" maxOccurs="1">
1423             <xsd:annotation>
1424                 <xsd:documentation>The last name of a subject.</xsd:documentation>
1425             </xsd:annotation>
1426         </xsd:element>
1427         <xsd:element name="FirstName" type="xsd:string" minOccurs="0" maxOccurs="1">
1428             <xsd:annotation>
1429                 <xsd:documentation>The first name of a subject.</xsd:documentation>
1430             </xsd:annotation>
1431         </xsd:element>
1432         <xsd:element name="BiographicDataItems" minOccurs="0">
1433             <xsd:complexType>
1434                 <xsd:sequence>
1435                     <xsd:element name="BiographicDataItem" type="tns:BiographicDataItem"
1436 maxOccurs="unbounded">
1437                 <xsd:annotation>

```

```

1438         <xsd:documentation>A single biographic data element.</xsd:documentation>
1439     </xsd:annotation>
1440 </xsd:element>
1441 </xsd:sequence>
1442 </xsd:complexType>
1443 </xsd:element>
1444 <xsd:element name="BiographicDataSet" type="tns:BiographicDataSetType" minOccurs="0"
1445 maxOccurs="1">
1446     <xsd:annotation>
1447         <xsd:documentation>A set of biographic data information.</xsd:documentation>
1448     </xsd:annotation>
1449 </xsd:element>
1450 </xsd:sequence>
1451 </xsd:complexType>
1452 <xsd:complexType name="BiometricDataElementType">
1453     <xsd:annotation>
1454         <xsd:documentation>
1455             Provides descriptive information about biometric data, such as
1456             the biometric type, subtype, and format, contained in the BDB of
1457             the CBEFF-BIR.
1458         </xsd:documentation>
1459     </xsd:annotation>
1460 <xsd:sequence>
1461     <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="1"
1462 maxOccurs="1">
1463         <xsd:annotation>
1464             <xsd:documentation>The type of biological or behavioral data stored in the biometric record, as
1465             defined by CBEFF.</xsd:documentation>
1466         </xsd:annotation>
1467     </xsd:element>
1468     <xsd:element name="BiometricTypeCount" type="xsd:positiveInteger" minOccurs="0"
1469 maxOccurs="1">
1470         <xsd:annotation>
1471             <xsd:documentation>The number of biometric records having the biometric type recorded in the
1472             biometric type field.</xsd:documentation>
1473         </xsd:annotation>
1474     </xsd:element>
1475     <xsd:element name="BiometricSubType" type="oasis_cbeff:SubtypeType" minOccurs="0"
1476 maxOccurs="1">
1477         <xsd:annotation>
1478             <xsd:documentation>More specifically defines the type of biometric data stored in the biometric
1479             record, as defined by CBEFF.</xsd:documentation>
1480         </xsd:annotation>
1481     </xsd:element>

```

```

1482     <xsd:element name="BDBFormatOwner" type="xsd:positiveInteger" minOccurs="1"
1483     maxOccurs="1">
1484     <xsd:annotation>
1485     <xsd:documentation>Identifies the standards body, working group, industry consortium, or other
1486     CBEFF biometric organization that has defined the format for the biometric data.</xsd:documentation>
1487     </xsd:annotation>
1488     </xsd:element>
1489     <xsd:element name="BDBFormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
1490     <xsd:annotation>
1491     <xsd:documentation>Identifies the specific biometric data format specified by the CBEFF
1492     biometric organization recorded in the BDB Format Owner field.</xsd:documentation>
1493     </xsd:annotation>
1494     </xsd:element>
1495     </xsd:sequence>
1496     </xsd:complexType>
1497     <xsd:complexType name="BiometricDataListType">
1498     <xsd:annotation>
1499     <xsd:documentation>A list of biometric data elements.</xsd:documentation>
1500     </xsd:annotation>
1501     <xsd:sequence>
1502     <xsd:element name="BiometricDataElement" type="tns:BiometricDataElementType"
1503     minOccurs="0" maxOccurs="unbounded">
1504     <xsd:annotation>
1505     <xsd:documentation>Data structure containing information about a biometric
1506     record.</xsd:documentation>
1507     </xsd:annotation>
1508     </xsd:element>
1509     </xsd:sequence>
1510     </xsd:complexType>
1511     <xsd:complexType name="CandidateListType">
1512     <xsd:annotation>
1513     <xsd:documentation>
1514     Defines a set of candidates, utilizing the Candidate Type to
1515     represent each element in the set.
1516     </xsd:documentation>
1517     </xsd:annotation>
1518     <xsd:sequence>
1519     <xsd:element name="Candidate" type="tns:CandidateType" minOccurs="0"
1520     maxOccurs="unbounded">
1521     <xsd:annotation>
1522     <xsd:documentation>A single candidate.</xsd:documentation>
1523     </xsd:annotation>
1524     </xsd:element>
1525     </xsd:sequence>

```

```

1526     </xsd:complexType>
1527     <xsd:complexType name="CandidateType">
1528         <xsd:annotation>
1529             <xsd:documentation>
1530                 Defines a single candidate as a possible match in response to a
1531                 biometric identification request.
1532             </xsd:documentation>
1533         </xsd:annotation>
1534         <xsd:sequence>
1535             <xsd:element name="Score" type="tns:Score" minOccurs="0" maxOccurs="1">
1536                 <xsd:annotation>
1537                     <xsd:documentation>The match score.</xsd:documentation>
1538                 </xsd:annotation>
1539             </xsd:element>
1540             <xsd:element name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1">
1541                 <xsd:annotation>
1542                     <xsd:documentation>The rank of the candidate in relation to other candidates for the same
1543                     biometric identification operation.</xsd:documentation>
1544                 </xsd:annotation>
1545             </xsd:element>
1546             <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0"
1547             maxOccurs="1">
1548                 <xsd:annotation>
1549                     <xsd:documentation>Biographic data associated with the candidate
1550                     match.</xsd:documentation>
1551                 </xsd:annotation>
1552             </xsd:element>
1553             <xsd:element name="BIRList" type="tns:CBEFF_BIR_ListType" minOccurs="1" maxOccurs="1">
1554                 <xsd:annotation>
1555                     <xsd:documentation>Biometric data associated with the candidate match.</xsd:documentation>
1556                 </xsd:annotation>
1557             </xsd:element>
1558         </xsd:sequence>
1559     </xsd:complexType>
1560     <xsd:complexType name="CapabilityListType">
1561         <xsd:annotation>
1562             <xsd:documentation>Defines a set of capabilities.</xsd:documentation>
1563         </xsd:annotation>
1564         <xsd:sequence>
1565             <xsd:element name="Capability" type="tns:CapabilityType" minOccurs="0"
1566             maxOccurs="unbounded">
1567                 <xsd:annotation>
1568                     <xsd:documentation>A single capability.</xsd:documentation>

```

```

1569         </xsd:annotation>
1570     </xsd:element>
1571 </xsd:sequence>
1572 </xsd:complexType>
1573 <xsd:simpleType name="CapabilityName">
1574     <xsd:annotation>
1575         <xsd:documentation>A list of capability items.</xsd:documentation>
1576     </xsd:annotation>
1577     <xsd:restriction base="xsd:string">
1578         <xsd:enumeration value="AggregateInputDataOptional">
1579             <xsd:annotation>
1580                 <xsd:documentation>A data element accepted as optional input by the implementing system for
1581 the aggregate services.</xsd:documentation>
1582             </xsd:annotation>
1583         </xsd:enumeration>
1584         <xsd:enumeration value="AggregateInputDataRequired">
1585             <xsd:annotation>
1586                 <xsd:documentation>A data element required as input by the implementing system for the
1587 aggregate services.</xsd:documentation>
1588             </xsd:annotation>
1589         </xsd:enumeration>
1590         <xsd:enumeration value="AggregateProcessingOption">
1591             <xsd:annotation>
1592                 <xsd:documentation>A processing option supported by the implementing system for the
1593 aggregate services.</xsd:documentation>
1594             </xsd:annotation>
1595         </xsd:enumeration>
1596         <xsd:enumeration value="AggregateReturnData">
1597             <xsd:annotation>
1598                 <xsd:documentation>A data element returned by the implementing system for the aggregate
1599 services.</xsd:documentation>
1600             </xsd:annotation>
1601         </xsd:enumeration>
1602         <xsd:enumeration value="AggregateServiceDescription">
1603             <xsd:annotation>
1604                 <xsd:documentation>Describes the processing logic of an aggregate service supported by the
1605 implementing system.</xsd:documentation>
1606             </xsd:annotation>
1607         </xsd:enumeration>
1608         <xsd:enumeration value="BiographicDataSet">
1609             <xsd:annotation>
1610                 <xsd:documentation>Identifies a biographic data set supported by the implementing
1611 system.</xsd:documentation>
1612             </xsd:annotation>

```



```

1613     </xsd:enumeration>
1614     <xsd:enumeration value="CBEFFPatronFormat">
1615         <xsd:annotation>
1616             <xsd:documentation>A patron format supported by the implementing
1617 system.</xsd:documentation>
1618         </xsd:annotation>
1619     </xsd:enumeration>
1620     <xsd:enumeration value="ClassificationAlgorithmType">
1621         <xsd:annotation>
1622             <xsd:documentation>A classification algorithm type supported by the implementing
1623 system.</xsd:documentation>
1624         </xsd:annotation>
1625     </xsd:enumeration>
1626     <xsd:enumeration value="ConformanceClass">
1627         <xsd:annotation>
1628             <xsd:documentation>Identifies the conformance class of the BIAS
1629 implementation.</xsd:documentation>
1630         </xsd:annotation>
1631     </xsd:enumeration>
1632     <xsd:enumeration value="Gallery">
1633         <xsd:annotation>
1634             <xsd:documentation>A gallery or population group supported by the implementing
1635 system.</xsd:documentation>
1636         </xsd:annotation>
1637     </xsd:enumeration>
1638     <xsd:enumeration value="IdentityModel">
1639         <xsd:annotation>
1640             <xsd:documentation>Identifies whether the implementing system is person-centric or encounter-
1641 centric based.</xsd:documentation>
1642         </xsd:annotation>
1643     </xsd:enumeration>
1644     <xsd:enumeration value="MatchScore">
1645         <xsd:annotation>
1646             <xsd:documentation> Identifies the use of match scores returned by the implementing
1647 system.</xsd:documentation>
1648         </xsd:annotation>
1649     </xsd:enumeration>
1650     <xsd:enumeration value="QualityAlgorithm">
1651         <xsd:annotation>
1652             <xsd:documentation>A quality algorithm vendor and algorithm vendor product ID supported by
1653 the implementing system.</xsd:documentation>
1654         </xsd:annotation>
1655     </xsd:enumeration>
1656     <xsd:enumeration value="SupportedBiometric">

```

```

1657     <xsd:annotation>
1658         <xsd:documentation>A biometric type supported by the implementing
1659 system.</xsd:documentation>
1660     </xsd:annotation>
1661 </xsd:enumeration>
1662 <xsd:enumeration value="TransformOperation">
1663     <xsd:annotation>
1664         <xsd:documentation>A transform operation type supported by the implementing
1665 system.</xsd:documentation>
1666     </xsd:annotation>
1667 </xsd:enumeration>
1668 </xsd:restriction>
1669 </xsd:simpleType>
1670 <xsd:complexType name="CapabilityType">
1671     <xsd:annotation>
1672         <xsd:documentation>Defines a single capability supported by an implementing
1673 system.</xsd:documentation>
1674     </xsd:annotation>
1675     <xsd:sequence>
1676         <xsd:element name="CapabilityName" type="tns:CapabilityName" minOccurs="1" maxOccurs="1">
1677             <xsd:annotation>
1678                 <xsd:documentation>The name of the capability.</xsd:documentation>
1679             </xsd:annotation>
1680         </xsd:element>
1681         <xsd:element name="CapabilityID" type="xsd:string" minOccurs="0" maxOccurs="1">
1682             <xsd:annotation>
1683                 <xsd:documentation>An identifier assigned to the capability by the implementing
1684 system.</xsd:documentation>
1685             </xsd:annotation>
1686         </xsd:element>
1687         <xsd:element name="CapabilityDescription" type="xsd:string" minOccurs="0" maxOccurs="1">
1688             <xsd:annotation>
1689                 <xsd:documentation>A description of the capability.</xsd:documentation>
1690             </xsd:annotation>
1691         </xsd:element>
1692         <xsd:element name="CapabilityValue" type="xsd:string" minOccurs="0" maxOccurs="1">
1693             <xsd:annotation>
1694                 <xsd:documentation>A value assigned to the capability.</xsd:documentation>
1695             </xsd:annotation>
1696         </xsd:element>
1697         <xsd:element name="CapabilitySupportingValue" type="xsd:string" minOccurs="0"
1698 maxOccurs="1">
1699             <xsd:annotation>
1700                 <xsd:documentation>A secondary value supporting the capability.</xsd:documentation>

```

```

1701         </xsd:annotation>
1702     </xsd:element>
1703     <xsd:element name="CapabilityAdditionalInfo" type="xsd:string" minOccurs="0" maxOccurs="1">
1704         <xsd:annotation>
1705             <xsd:documentation> Contains additional information for the supported
1706 capability.</xsd:documentation>
1707         </xsd:annotation>
1708     </xsd:element>
1709 </xsd:sequence>
1710 </xsd:complexType>
1711 <xsd:complexType name="CBEFF_BIR_ListType">
1712     <xsd:annotation>
1713         <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
1714     </xsd:annotation>
1715     <xsd:sequence>
1716         <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0"
1717 maxOccurs="unbounded">
1718             <xsd:annotation>
1719                 <xsd:documentation>CBEFF structure containing information about a biometric
1720 sample.</xsd:documentation>
1721             </xsd:annotation>
1722         </xsd:element>
1723     </xsd:sequence>
1724 </xsd:complexType>
1725 <xsd:complexType name="CBEFF_BIR_Type">
1726     <xsd:annotation>
1727         <xsd:documentation>Represents biometric information, with either a non-XML or XML
1728 representation.</xsd:documentation>
1729     </xsd:annotation>
1730     <xsd:sequence>
1731         <xsd:element name="FormatOwner" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1" />
1732         <xsd:element name="FormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1" />
1733         <xsd:element name="BIR_Information" minOccurs="0" maxOccurs="1">
1734             <xsd:complexType>
1735                 <xsd:sequence>
1736                     <xsd:element name="BIR_Info" type="oasis_cbeff:BIRInfoType" minOccurs="0"
1737 maxOccurs="1"/>
1738                     <xsd:element name="BDB_Info" type="oasis_cbeff:BDBInfoType" minOccurs="0"
1739 maxOccurs="1"/>
1740                     <xsd:element name="SB_Info" type="oasis_cbeff:SBInfoType" minOccurs="0"
1741 maxOccurs="1"/>
1742                 </xsd:sequence>
1743             </xsd:complexType>
1744         </xsd:element>

```

```

1745     <xsd:element name="BIR" type="tns:BaseBIRType" />
1746   </xsd:sequence>
1747 </xsd:complexType>
1748 <xsd:complexType name="BaseBIRType">
1749 </xsd:complexType>
1750 <xsd:complexType name="BinaryBIR">
1751   <xsd:complexContent>
1752     <xsd:extension base="tns:BaseBIRType">
1753       <xsd:sequence>
1754         <xsd:element name="Binary" type="xsd:base64Binary" />
1755       </xsd:sequence>
1756     </xsd:extension>
1757   </xsd:complexContent>
1758 </xsd:complexType>
1759 <xsd:complexType name="URI_BIR">
1760   <xsd:complexContent>
1761     <xsd:extension base="tns:BaseBIRType">
1762       <xsd:sequence>
1763         <xsd:element name="URI" type="xsd:anyURI" />
1764       </xsd:sequence>
1765     </xsd:extension>
1766   </xsd:complexContent>
1767 </xsd:complexType>
1768 <xsd:complexType name="XML_BIR">
1769   <xsd:complexContent>
1770     <xsd:extension base="tns:BaseBIRType">
1771       <xsd:sequence>
1772         <xsd:element name="XML" type="oasis_cbeff:BIRType" />
1773       </xsd:sequence>
1774     </xsd:extension>
1775   </xsd:complexContent>
1776 </xsd:complexType>
1777 <xsd:simpleType name="Classification">
1778   <xsd:annotation>
1779     <xsd:documentation>The result of a classification.</xsd:documentation>
1780   </xsd:annotation>
1781   <xsd:restriction base="xsd:string"/>
1782 </xsd:simpleType>
1783 <xsd:simpleType name="ClassificationAlgorithmType">
1784   <xsd:annotation>
1785     <xsd:documentation>Type of classification algorithm that was used to perform the
1786     classification.</xsd:documentation>
1787   </xsd:annotation>

```

```

1788     <xsd:restriction base="xsd:string"/>
1789 </xsd:simpleType>
1790 <xsd:complexType name="ClassificationData">
1791     <xsd:annotation>
1792         <xsd:documentation>Contains information on classification results and the algorithm used to
1793 determine the classification.</xsd:documentation>
1794     </xsd:annotation>
1795     <xsd:sequence>
1796         <xsd:element name="Classification" type="tns:Classification" minOccurs="1" maxOccurs="1">
1797             <xsd:annotation>
1798                 <xsd:documentation>The result of the classification.</xsd:documentation>
1799             </xsd:annotation>
1800         </xsd:element>
1801         <xsd:element name="ClassificationAlgorithmType" type="tns:ClassificationAlgorithmType"
1802 minOccurs="1" maxOccurs="1">
1803             <xsd:annotation>
1804                 <xsd:documentation>Identifies the type of classification algorithm that was used to perform the
1805 classification.</xsd:documentation>
1806             </xsd:annotation>
1807         </xsd:element>
1808     </xsd:sequence>
1809 </xsd:complexType>
1810 <xsd:complexType name="EncounterListType">
1811     <xsd:annotation>
1812         <xsd:documentation>Defines a set of encounters.</xsd:documentation>
1813     </xsd:annotation>
1814     <xsd:sequence>
1815         <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0"
1816 maxOccurs="unbounded">
1817             <xsd:annotation>
1818                 <xsd:documentation>The identifier of an encounter.</xsd:documentation>
1819             </xsd:annotation>
1820         </xsd:element>
1821     </xsd:sequence>
1822 </xsd:complexType>
1823 <xsd:complexType name="FusionInformationListType">
1824     <xsd:annotation>
1825         <xsd:documentation>
1826         Contains at a minimum two sets of fusion input
1827         elements, as input to the PerformFusion request.
1828     </xsd:documentation>
1829     </xsd:annotation>
1830 </xsd:sequence>

```

```

1831     <xsd:element name="FusionElement" type="tns:FusionInformationType" minOccurs="2"
1832 maxOccurs="unbounded">
1833     <xsd:annotation>
1834     <xsd:documentation>A set of fusion information.</xsd:documentation>
1835     </xsd:annotation>
1836     </xsd:element>
1837 </xsd:sequence>
1838 </xsd:complexType>
1839 <xsd:complexType name="FusionInformationType">
1840     <xsd:annotation>
1841     <xsd:documentation>Represents the information necessary to perform a fusion
1842 operation.</xsd:documentation>
1843     </xsd:annotation>
1844     <xsd:sequence>
1845     <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="1"
1846 maxOccurs="1">
1847     <xsd:annotation>
1848     <xsd:documentation>The type of biological or behavioral data stored in the biometric record, as
1849 defined by CBEFF.</xsd:documentation>
1850     </xsd:annotation>
1851     </xsd:element>
1852     <xsd:element name="BiometricSubType" type="oasis_cbeff:SubtypeType" minOccurs="0"
1853 maxOccurs="1">
1854     <xsd:annotation>
1855     <xsd:documentation>More specifically defines the type of biometric data stored in the biometric
1856 record.</xsd:documentation>
1857     </xsd:annotation>
1858     </xsd:element>
1859     <xsd:element name="AlgorithmOwner" type="xsd:string" minOccurs="1" maxOccurs="1">
1860     <xsd:annotation>
1861     <xsd:documentation>The owner or vendor of the algorithm used to determine the score or
1862 decision.</xsd:documentation>
1863     </xsd:annotation>
1864     </xsd:element>
1865     <xsd:element name="AlgorithmType" type="xsd:string" minOccurs="1" maxOccurs="1">
1866     <xsd:annotation>
1867     <xsd:documentation>The Algorithm Owner's identifier for the specific algorithm product and
1868 version used to determine the score or decision.</xsd:documentation>
1869     </xsd:annotation>
1870     </xsd:element>
1871     <xsd:element name="FusionResult" type="tns:FusionResult" />
1872 </xsd:sequence>
1873 </xsd:complexType>
1874 <xsd:complexType name="FusionResult">
1875 <xsd:sequence />

```

```

1876     </xsd:complexType>
1877     <xsd:complexType name="FusionScore">
1878         <xsd:complexContent>
1879             <xsd:extension base="tns:FusionResult">
1880                 <xsd:sequence>
1881                     <xsd:element name="Score" type="tns:Score">
1882                         <xsd:annotation>
1883                             <xsd:documentation>The similarity score assigned by the matching
1884 algorithm.</xsd:documentation>
1885                         </xsd:annotation>
1886                     </xsd:element>
1887                 </xsd:sequence>
1888             </xsd:extension>
1889         </xsd:complexContent>
1890     </xsd:complexType>
1891     <xsd:complexType name="FusionDecision">
1892         <xsd:complexContent>
1893             <xsd:extension base="tns:FusionResult">
1894                 <xsd:sequence>
1895                     <xsd:element name="Decision" type="xsd:string">
1896                         <xsd:annotation>
1897                             <xsd:documentation>The match decision assigned by the matching
1898 algorithm.</xsd:documentation>
1899                         </xsd:annotation>
1900                     </xsd:element>
1901                 </xsd:sequence>
1902             </xsd:extension>
1903         </xsd:complexContent>
1904     </xsd:complexType>
1905     <xsd:complexType name="GenericRequestParameters">
1906         <xsd:annotation>
1907             <xsd:documentation>Common request paramters that can be used to identify the
1908 requester.</xsd:documentation>
1909         </xsd:annotation>
1910         <xsd:sequence>
1911             <xsd:element name="Application" type="tns:ApplicationIdentifier" minOccurs="0" maxOccurs="1">
1912                 <xsd:annotation>
1913                     <xsd:documentation>Identifies the requesting application.</xsd:documentation>
1914                 </xsd:annotation>
1915             </xsd:element>
1916             <xsd:element name="ApplicationUser" type="tns:ApplicationUserIdentifier" minOccurs="0"
1917 maxOccurs="1">
1918                 <xsd:annotation>

```

```

1919         <xsd:documentation>Identifers the user or instance of the requesting
1920 application.</xsd:documentation>
1921         </xsd:annotation>
1922     </xsd:element>
1923     <xsd:element name="BIASOperationName" type="xsd:string" minOccurs="0" maxOccurs="1">
1924         <xsd:annotation>
1925             <xsd:documentation>Identifers the BIAS operation name that is being
1926 requested.</xsd:documentation>
1927         </xsd:annotation>
1928     </xsd:element>
1929 </xsd:sequence>
1930 </xsd:complexType>
1931 <xsd:complexType name="InformationType">
1932     <xsd:annotation>
1933         <xsd:documentation>
1934             Allows for an unlimited number of data element types, and it does
1935             not specify nor require any particular data element.
1936         </xsd:documentation>
1937     </xsd:annotation>
1938     <!-- THE FOLLOWING COMMENTED SECTION IS PROVIDED AS AN INFORMATIVE
1939          EXAMPLE OF WHAT 'InformationType'S CONTENT COULD BE -->
1940
1941     <!-- xsd:sequence>
1942         <xsd:element name="GUID" type="xsd:string" minOccurs="0" />
1943         <xsd:element name="GivenName" type="xsd:string" minOccurs="0" />
1944         <xsd:element name="FamilyName" type="xsd:string" minOccurs="0" />
1945         <xsd:element name="DateOfBirth" type="xsd:string" minOccurs="0" />
1946         <xsd:element name="Sex" type="xsd:string" minOccurs="0" />
1947         <xsd:element name="Citizenship" type="xsd:string" minOccurs="0" />
1948         <xsd:element name="Images" minOccurs="0">
1949             <xsd:complexType>
1950                 <xsd:sequence>
1951                     <xsd:element name="Image" minOccurs="0" maxOccurs="unbounded">
1952                         <xsd:complexType>
1953                             <xsd:sequence>
1954                                 <xsd:element name="ContentType" type="xsd:string" minOccurs="0" />
1955                                 <xsd:element name="FingerPosition" type="xsd:string" minOccurs="0" />
1956                                 <xsd:element name="HandPosition" type="xsd:string" minOccurs="0" />
1957                                 <xsd:element name="ImageData" type="xsd:base64Binary" minOccurs="0" />
1958                             </xsd:sequence>
1959                         </xsd:complexType>
1960                     </xsd:element>
1961                 </xsd:sequence>

```



```

1962     </xsd:complexType>
1963 </xsd:element>
1964 <xsd:element name="Identities" minOccurs="0">
1965     <xsd:complexType>
1966         <xsd:sequence>
1967             <xsd:element name="Identity" type="xsd:string" maxOccurs="unbounded" />
1968         </xsd:sequence>
1969     </xsd:complexType>
1970 </xsd:element>
1971 </xsd:sequence-->
1972 </xsd:complexType>
1973 <xsd:complexType name="ListFilterType">
1974     <xsd:annotation>
1975         <xsd:documentation>
1976             Provides a method to filter the amount of information returned in
1977             a search of biometric data.
1978         </xsd:documentation>
1979     </xsd:annotation>
1980 <xsd:sequence>
1981     <xsd:element name="BiometricTypeFilters">
1982         <xsd:complexType>
1983             <xsd:sequence>
1984                 <xsd:element name="BiometricTypeFilter" type="oasis_cbeff:MultipleTypesType"
1985 minOccurs="1" maxOccurs="unbounded">
1986                 <xsd:annotation>
1987                     <xsd:documentation>Limits the returned information to a specific type of biometric, as
1988 defined by CBEFF.</xsd:documentation>
1989                 </xsd:annotation>
1990             </xsd:element>
1991         </xsd:sequence>
1992     </xsd:complexType>
1993 </xsd:element>
1994     <xsd:element name="IncludeBiometricSubtype" type="xsd:boolean" minOccurs="1"
1995 maxOccurs="1">
1996     <xsd:annotation>
1997         <xsd:documentation>A Boolean flag indicating if biometric subtype information should be
1998 returned.</xsd:documentation>
1999     </xsd:annotation>
2000 </xsd:element>
2001 </xsd:sequence>
2002 </xsd:complexType>
2003 <xsd:simpleType name="MatchType">
2004     <xsd:annotation>

```

```

2005     <xsd:documentation>The result of a fusion method.</xsd:documentation>
2006     </xsd:annotation>
2007     <xsd:restriction base="xsd:boolean"/>
2008 </xsd:simpleType>
2009 <xsd:complexType name="ProcessingOptionsType">
2010     <xsd:annotation>
2011     <xsd:documentation>
2012     BIAS aggregate services support the ability to include various
2013     processing options which direct and possibly control the business
2014     logic for that service. The ProcessingOptionsType provides a
2015     method to represent those options. Processing options should be
2016     defined by the implementing system.
2017     </xsd:documentation>
2018     </xsd:annotation>
2019     <xsd:sequence>
2020     <xsd:element name="Option" type="xsd:string" minOccurs="0" maxOccurs="unbounded">
2021     <xsd:annotation>
2022     <xsd:documentation>An option supported by the implementing system.</xsd:documentation>
2023     </xsd:annotation>
2024     </xsd:element>
2025     </xsd:sequence>
2026 </xsd:complexType>
2027 <xsd:simpleType name="ProductID">
2028     <xsd:annotation>
2029     <xsd:documentation>The vendor's ID for a particular product.</xsd:documentation>
2030     </xsd:annotation>
2031     <xsd:restriction base="xsd:string"/>
2032 </xsd:simpleType>
2033 <xsd:complexType name="QualityData">
2034     <xsd:annotation>
2035     <xsd:documentation>Contains information about a biometric sample's quality and the algorithm
2036     used to compute the quality.</xsd:documentation>
2037     </xsd:annotation>
2038     <xsd:sequence>
2039     <xsd:element name="QualityScore" type="oasis_cbeff:QualityType" minOccurs="0"
2040     maxOccurs="1">
2041     <xsd:annotation>
2042     <xsd:documentation>The quality of a biometric sample.</xsd:documentation>
2043     </xsd:annotation>
2044     </xsd:element>
2045     <xsd:element name="AlgorithmVendor" type="tns:VendorIdentifier" minOccurs="1"
2046     maxOccurs="1">
2047     <xsd:annotation>

```

```

2048         <xsd:documentation>The vendor of the quality algorithm used to determine the quality
2049 score.</xsd:documentation>
2050     </xsd:annotation>
2051 </xsd:element>
2052     <xsd:element name="AlgorithmVendorProductID" type="tns:ProductID" minOccurs="1"
2053 maxOccurs="1">
2054         <xsd:annotation>
2055             <xsd:documentation>The vendor's ID for the algorithm used to determine the
2056 quality.</xsd:documentation>
2057         </xsd:annotation>
2058     </xsd:element>
2059     <xsd:element name="AlgorithmVersion" type="tns:VersionType" minOccurs="0" maxOccurs="1">
2060         <xsd:annotation>
2061             <xsd:documentation>The version of the algorithm used to determine the
2062 quality.</xsd:documentation>
2063         </xsd:annotation>
2064     </xsd:element>
2065 </xsd:sequence>
2066 </xsd:complexType>
2067 <xsd:complexType name="RequestTemplate">
2068     <xsd:annotation>
2069         <xsd:documentation>Base template for BIAS primitive service requests.</xsd:documentation>
2070     </xsd:annotation>
2071     <xsd:sequence>
2072         <xsd:element name="GenericRequestParameters" type="tns:GenericRequestParameters"
2073 minOccurs="0" maxOccurs="1"/>
2074     </xsd:sequence>
2075 </xsd:complexType>
2076 <xsd:complexType name="ResponseStatus">
2077     <xsd:sequence>
2078         <xsd:element name="Return" type="tns:ReturnCode" minOccurs="1" maxOccurs="1">
2079             <xsd:annotation>
2080                 <xsd:documentation>The return code indicates the return status of the
2081 operation.</xsd:documentation>
2082             </xsd:annotation>
2083         </xsd:element>
2084         <xsd:element name="Message" type="xsd:string" minOccurs="0" maxOccurs="1">
2085             <xsd:annotation>
2086                 <xsd:documentation>A short message corresponding to the return code.</xsd:documentation>
2087             </xsd:annotation>
2088         </xsd:element>
2089     </xsd:sequence>
2090 </xsd:complexType>
2091 <xsd:complexType name="ResponseTemplate">

```

```

2092     <xsd:annotation>
2093         <xsd:documentation>Base template for BIAS responses.</xsd:documentation>
2094     </xsd:annotation>
2095     <xsd:sequence>
2096         <xsd:element name="ResponseStatus" type="tns:ResponseStatus" minOccurs="1"
2097     minOccurs="1">
2098         <xsd:annotation>
2099             <xsd:documentation>Returned status for the operation.</xsd:documentation>
2100         </xsd:annotation>
2101     </xsd:element>
2102 </xsd:sequence>
2103 </xsd:complexType>
2104 <xsd:simpleType name="ReturnCode">
2105     <xsd:annotation>
2106         <xsd:documentation>BIAS Operation Return Codes</xsd:documentation>
2107     </xsd:annotation>
2108     <xsd:restriction base="xsd:unsignedLong">
2109         <xsd:enumeration value="0">
2110             <xsd:annotation>
2111                 <xsd:documentation>Success</xsd:documentation>
2112             </xsd:annotation>
2113         </xsd:enumeration>
2114     </xsd:restriction>
2115 </xsd:simpleType>
2116 <xsd:simpleType name="Score">
2117     <xsd:annotation>
2118         <xsd:documentation>Match result or quality score.</xsd:documentation>
2119     </xsd:annotation>
2120     <xsd:restriction base="xsd:float"/>
2121 </xsd:simpleType>
2122 <xsd:complexType name="TokenType">
2123     <xsd:annotation>
2124         <xsd:documentation>Defines a token that is returned for asynchronous
2125     processing.</xsd:documentation>
2126     </xsd:annotation>
2127     <xsd:sequence>
2128         <xsd:element name="TokenValue" type="xsd:string" minOccurs="1" maxOccurs="1">
2129             <xsd:annotation>
2130                 <xsd:documentation>A value returned by the implementing system that is used to retrieve the
2131     results to a service at a later time.</xsd:documentation>
2132             </xsd:annotation>
2133         </xsd:element>
2134         <xsd:element name="Expiration" type="xsd:date" minOccurs="1" maxOccurs="1">

```

```

2135     <xsd:annotation>
2136         <xsd:documentation>A date and time at which point the token expires and the service results are
2137 no longer guaranteed to be available.</xsd:documentation>
2138     </xsd:annotation>
2139 </xsd:element>
2140 </xsd:sequence>
2141 </xsd:complexType>
2142 <xsd:simpleType name="VendorIdentifier">
2143     <xsd:annotation>
2144         <xsd:documentation>Identifies a vendor.</xsd:documentation>
2145     </xsd:annotation>
2146     <xsd:restriction base="xsd:string"/>
2147 </xsd:simpleType>
2148 <xsd:complexType name="Version">
2149     <xsd:annotation>
2150         <xsd:documentation>
2151             For a description or definition of each data element, see the
2152             referenced CBEFF standards in the CBEFF_XML_BIR_Type schema.
2153         </xsd:documentation>
2154     </xsd:annotation>
2155     <xsd:sequence>
2156         <xsd:element name="major" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
2157         <xsd:element name="minor" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
2158     </xsd:sequence>
2159 </xsd:complexType>
2160 <xsd:simpleType name="VersionType">
2161     <xsd:annotation>
2162         <xsd:documentation>The version of a component.</xsd:documentation>
2163     </xsd:annotation>
2164     <xsd:restriction base="xsd:string"/>
2165 </xsd:simpleType>
2166 <xsd:element name="AddSubjectToGallery">
2167     <xsd:complexType>
2168         <xsd:annotation>
2169             <xsd:documentation>
2170                 Register a subject to a given gallery or population group.
2171             </xsd:documentation>
2172         </xsd:annotation>
2173         <xsd:sequence>
2174             <xsd:element name="AddSubjectToGalleryRequest" nillable="true">
2175                 <xsd:complexType>
2176                     <xsd:complexContent>
2177                         <xsd:extension base="tns:RequestTemplate">

```

```

2178         <xsd:sequence>
2179             <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="1" maxOccurs="1">
2180                 <xsd:annotation>
2181                     <xsd:documentation>The identifier of the gallery or population group to which the subject
2182 will be added.</xsd:documentation>
2183                 </xsd:annotation>
2184             </xsd:element>
2185             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2186                 <xsd:annotation>
2187                     <xsd:documentation>The identity to add to the gallery.</xsd:documentation>
2188                 </xsd:annotation>
2189             </xsd:element>
2190         </xsd:sequence>
2191     </xsd:extension>
2192 </xsd:complexContent>
2193 </xsd:complexType>
2194 </xsd:element>
2195 </xsd:sequence>
2196 </xsd:complexType>
2197 </xsd:element>
2198 <xsd:element name="AddSubjectToGalleryResponse">
2199     <xsd:complexType>
2200         <xsd:annotation>
2201             <xsd:documentation>
2202                 The response to an AddSubjectToGallery request.
2203             </xsd:documentation>
2204         </xsd:annotation>
2205         <xsd:sequence>
2206             <xsd:element name="AddSubjectToGalleryResponsePackage" nillable="true">
2207                 <xsd:complexType>
2208                     <xsd:complexContent>
2209                         <xsd:extension base="tns:ResponseTemplate"/>
2210                     </xsd:complexContent>
2211                 </xsd:complexType>
2212             </xsd:element>
2213         </xsd:sequence>
2214     </xsd:complexType>
2215 </xsd:element>
2216 <xsd:element name="CheckQuality">
2217     <xsd:complexType>
2218         <xsd:annotation>
2219             <xsd:documentation>
2220                 Calculate a quality score for a given biometric.

```

```

2221     </xsd:documentation>
2222 </xsd:annotation>
2223 <xsd:sequence>
2224     <xsd:element name="CheckQualityRequest" nillable="true">
2225         <xsd:complexType>
2226             <xsd:complexContent>
2227                 <xsd:extension base="tns:RequestTemplate">
2228                     <xsd:sequence>
2229                         <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="1"
2230 maxOccurs="1">
2231                             <xsd:annotation>
2232                                 <xsd:documentation>Data structure containing a single biometric sample for which a
2233 quality score is to be determined.</xsd:documentation>
2234                             </xsd:annotation>
2235                         </xsd:element>
2236                         <xsd:element name="Quality" type="tns:QualityData" minOccurs="0" maxOccurs="1">
2237                             <xsd:annotation>
2238                                 <xsd:documentation>Specifies a particular algorithm vendor and vender product
2239 ID.</xsd:documentation>
2240                             </xsd:annotation>
2241                         </xsd:element>
2242                     </xsd:sequence>
2243                 </xsd:extension>
2244             </xsd:complexContent>
2245         </xsd:complexType>
2246     </xsd:element>
2247 </xsd:sequence>
2248 </xsd:complexType>
2249 </xsd:element>
2250 <xsd:element name="CheckQualityResponse">
2251     <xsd:complexType>
2252         <xsd:annotation>
2253             <xsd:documentation>
2254                 The response to a CheckQuality request.
2255             </xsd:documentation>
2256         </xsd:annotation>
2257     <xsd:sequence>
2258         <xsd:element name="CheckQualityResponsePackage" nillable="true">
2259             <xsd:complexType>
2260                 <xsd:complexContent>
2261                     <xsd:extension base="tns:ResponseTemplate">
2262                         <xsd:sequence>
2263                             <xsd:element name="QualityInfo" type="tns:QualityData" minOccurs="1" maxOccurs="1">

```

```

2264         <xsd:annotation>
2265             <xsd:documentation>Contains the quality information for the submitted biometric
2266 sample.</xsd:documentation>
2267         </xsd:annotation>
2268     </xsd:element>
2269 </xsd:sequence>
2270 </xsd:extension>
2271 </xsd:complexContent>
2272 </xsd:complexType>
2273 </xsd:element>
2274 </xsd:sequence>
2275 </xsd:complexType>
2276 </xsd:element>
2277 <xsd:element name="ClassifyBiometricData">
2278     <xsd:complexType>
2279         <xsd:annotation>
2280             <xsd:documentation>
2281                 Classifies a biometric sample.
2282             </xsd:documentation>
2283         </xsd:annotation>
2284     <xsd:sequence>
2285         <xsd:element name="ClassifyBiometricDataRequest" nillable="true">
2286             <xsd:complexType>
2287                 <xsd:complexContent>
2288                     <xsd:extension base="tns:RequestTemplate">
2289                         <xsd:sequence>
2290                             <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="1"
2291 maxOccurs="1">
2292                                 <xsd:annotation>
2293                                     <xsd:documentation>Data structure containing a single biometric sample for which the
2294 classification is to be determined.</xsd:documentation>
2295                                 </xsd:annotation>
2296                             </xsd:element>
2297                         </xsd:sequence>
2298                     </xsd:extension>
2299                 </xsd:complexContent>
2300             </xsd:complexType>
2301         </xsd:element>
2302     </xsd:sequence>
2303 </xsd:complexType>
2304 </xsd:element>
2305 <xsd:element name="ClassifyBiometricDataResponse">
2306     <xsd:complexType>

```



```

2307     <xsd:annotation>
2308     <xsd:documentation>
2309         The response to a ClassifyBiometricData request, containing
2310         the classification of a biometric sample.
2311     </xsd:documentation>
2312 </xsd:annotation>
2313 <xsd:sequence>
2314     <xsd:element name="ClassifyBiometricDataResponsePackage" nillable="true">
2315     <xsd:complexType>
2316     <xsd:complexContent>
2317         <xsd:extension base="tns:ResponseTemplate">
2318         <xsd:sequence>
2319             <xsd:element name="ClassificationData" type="tns:ClassificationData" minOccurs="1"
2320 maxOccurs="1">
2321                 <xsd:annotation>
2322                 <xsd:documentation>Information on the results and type of classification
2323 performed.</xsd:documentation>
2324                 </xsd:annotation>
2325                 </xsd:element>
2326                 </xsd:sequence>
2327                 </xsd:extension>
2328                 </xsd:complexContent>
2329                 </xsd:complexType>
2330                 </xsd:element>
2331                 </xsd:sequence>
2332                 </xsd:complexType>
2333             </xsd:element>
2334 <xsd:element name="CreateSubject">
2335 <xsd:complexType>
2336 <xsd:annotation>
2337 <xsd:documentation>
2338     Create a new subject record.
2339 </xsd:documentation>
2340 </xsd:annotation>
2341 <xsd:sequence>
2342 <xsd:element name="CreateSubjectRequest" nillable="true">
2343 <xsd:complexType>
2344 <xsd:complexContent>
2345 <xsd:extension base="tns:RequestTemplate"/>
2346 </xsd:complexContent>
2347 </xsd:complexType>
2348 </xsd:element>
2349 </xsd:sequence>

```

```

2350     </xsd:complexType>
2351 </xsd:element>
2352 <xsd:element name="CreateSubjectResponse">
2353     <xsd:complexType>
2354         <xsd:annotation>
2355             <xsd:documentation>
2356                 The response to a CreateSubject request, containing the subject
2357                 ID of the new subject record.
2358             </xsd:documentation>
2359         </xsd:annotation>
2360         <xsd:sequence>
2361             <xsd:element name="CreateSubjectResponsePackage" nillable="true">
2362                 <xsd:complexType>
2363                     <xsd:complexContent>
2364                         <xsd:extension base="tns:ResponseTemplate">
2365                             <xsd:sequence>
2366                                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2367                                     <xsd:annotation>
2368                                         <xsd:documentation>Contains the subject ID of the new subject
2369 record.</xsd:documentation>
2370                                     </xsd:annotation>
2371                                 </xsd:element>
2372                             </xsd:sequence>
2373                         </xsd:extension>
2374                     </xsd:complexContent>
2375                 </xsd:complexType>
2376             </xsd:element>
2377         </xsd:sequence>
2378     </xsd:complexType>
2379 </xsd:element>
2380 <xsd:element name="DeleteBiographicData">
2381     <xsd:complexType>
2382         <xsd:annotation>
2383             <xsd:documentation>
2384                 Erase all of the biographic data associated with a given
2385                 subject record or, in the encounter-centric model, with a
2386                 given encounter.
2387             </xsd:documentation>
2388         </xsd:annotation>
2389         <xsd:sequence>
2390             <xsd:element name="DeleteBiographicDataRequest" nillable="true">
2391                 <xsd:complexType>
2392                     <xsd:complexContent>

```

```

2393         <xsd:extension base="tns:RequestTemplate">
2394         <xsd:sequence>
2395             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2396                 <xsd:annotation>
2397                     <xsd:documentation>Contains either the subject ID or encounter ID
2398 reference.</xsd:documentation>
2399                 </xsd:annotation>
2400             </xsd:element>
2401         </xsd:sequence>
2402     </xsd:extension>
2403 </xsd:complexContent>
2404 </xsd:complexType>
2405 </xsd:element>
2406 </xsd:sequence>
2407 </xsd:complexType>
2408 </xsd:element>
2409 <xsd:element name="DeleteBiographicDataResponse">
2410     <xsd:complexType>
2411         <xsd:annotation>
2412             <xsd:documentation>
2413                 The response to a DeleteBiographicData request.
2414             </xsd:documentation>
2415         </xsd:annotation>
2416         <xsd:sequence>
2417             <xsd:element name="DeleteBiographicDataResponsePackage" nillable="true">
2418                 <xsd:complexType>
2419                     <xsd:complexContent>
2420                         <xsd:extension base="tns:ResponseTemplate"/>
2421                     </xsd:complexContent>
2422                 </xsd:complexType>
2423             </xsd:element>
2424         </xsd:sequence>
2425     </xsd:complexType>
2426 </xsd:element>
2427 <xsd:element name="DeleteBiometricData">
2428     <xsd:complexType>
2429         <xsd:annotation>
2430             <xsd:documentation>
2431                 Erase all of the biometric data associated with a given
2432                 subject record or, in the encounter-centric model, with a
2433                 given encounter.
2434             </xsd:documentation>
2435         </xsd:annotation>

```

```

2436     <xsd:sequence>
2437     <xsd:element name="DeleteBiometricDataRequest" nillable="true">
2438     <xsd:complexType>
2439     <xsd:complexContent>
2440     <xsd:extension base="tns:RequestTemplate">
2441     <xsd:sequence>
2442     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2443     <xsd:annotation>
2444     <xsd:documentation>Contains either the subject ID or encounter ID
2445 reference.</xsd:documentation>
2446     </xsd:annotation>
2447     </xsd:element>
2448     </xsd:sequence>
2449     </xsd:extension>
2450     </xsd:complexContent>
2451     </xsd:complexType>
2452     </xsd:element>
2453     </xsd:sequence>
2454     </xsd:complexType>
2455     </xsd:element>
2456     <xsd:element name="DeleteBiometricDataResponse">
2457     <xsd:complexType>
2458     <xsd:annotation>
2459     <xsd:documentation>
2460     The response to a DeleteBiometricData request.
2461     </xsd:documentation>
2462     </xsd:annotation>
2463     <xsd:sequence>
2464     <xsd:element name="DeleteBiometricDataResponsePackage" nillable="true">
2465     <xsd:complexType>
2466     <xsd:complexContent>
2467     <xsd:extension base="tns:ResponseTemplate"/>
2468     </xsd:complexContent>
2469     </xsd:complexType>
2470     </xsd:element>
2471     </xsd:sequence>
2472     </xsd:complexType>
2473     </xsd:element>
2474     <xsd:element name="DeleteSubject">
2475     <xsd:complexType>
2476     <xsd:annotation>
2477     <xsd:documentation>
2478     Delete an existing subject record and, in an encounter-centric

```

2479           model, any associated encounter information.

2480           </xsd:documentation>

2481           </xsd:annotation>

2482           <xsd:sequence>

2483           <xsd:element name="DeleteSubjectRequest" nillable="true">

2484            <xsd:complexType>

2485            <xsd:complexContent>

2486            <xsd:extension base="tns:RequestTemplate">

2487            <xsd:sequence>

2488            <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">

2489            <xsd:annotation>

2490            <xsd:documentation>Subject ID of the identity to delete.</xsd:documentation>

2491            </xsd:annotation>

2492            </xsd:element>

2493            </xsd:sequence>

2494            </xsd:extension>

2495            </xsd:complexContent>

2496            </xsd:complexType>

2497            </xsd:element>

2498            </xsd:sequence>

2499            </xsd:complexType>

2500           </xsd:element>

2501           <xsd:element name="DeleteSubjectResponse">

2502            <xsd:complexType>

2503            <xsd:annotation>

2504            <xsd:documentation>

2505            The response to a DeleteSubject request.

2506            </xsd:documentation>

2507            </xsd:annotation>

2508            <xsd:sequence>

2509            <xsd:element name="DeleteSubjectResponsePackage" nillable="true">

2510            <xsd:complexType>

2511            <xsd:complexContent>

2512            <xsd:extension base="tns:ResponseTemplate"/>

2513            </xsd:complexContent>

2514            </xsd:complexType>

2515            </xsd:element>

2516            </xsd:sequence>

2517            </xsd:complexType>

2518            </xsd:element>

2519            <xsd:element name="DeleteSubjectFromGallery">

2520            <xsd:complexType>

```

2521     <xsd:annotation>
2522     <xsd:documentation>
2523         Remove the registration of a subject from a gallery or
2524         population group.
2525     </xsd:documentation>
2526 </xsd:annotation>
2527 <xsd:sequence>
2528     <xsd:element name="DeleteSubjectFromGalleryRequest" nillable="true">
2529     <xsd:complexType>
2530     <xsd:complexContent>
2531     <xsd:extension base="tns:RequestTemplate">
2532     <xsd:sequence>
2533     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="1" maxOccurs="1">
2534     <xsd:annotation>
2535     <xsd:documentation>The identifier of the gallery or population group from which the
2536 subject will be deleted.</xsd:documentation>
2537     </xsd:annotation>
2538     </xsd:element>
2539     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2540     <xsd:annotation>
2541     <xsd:documentation>The identity to remove from the gallery.</xsd:documentation>
2542     </xsd:annotation>
2543     </xsd:element>
2544     </xsd:sequence>
2545     </xsd:extension>
2546     </xsd:complexContent>
2547     </xsd:complexType>
2548     </xsd:element>
2549     </xsd:sequence>
2550 </xsd:complexType>
2551 </xsd:element>
2552 <xsd:element name="DeleteSubjectFromGalleryResponse">
2553 <xsd:complexType>
2554 <xsd:annotation>
2555 <xsd:documentation>
2556     The response to a DeleteSubjectFromGallery request.
2557 </xsd:documentation>
2558 </xsd:annotation>
2559 <xsd:sequence>
2560 <xsd:element name="DeleteSubjectFromGalleryResponsePackage" nillable="true">
2561 <xsd:complexType>
2562 <xsd:complexContent>
2563 <xsd:extension base="tns:ResponseTemplate"/>

```

```

2564         </xsd:complexContent>
2565     </xsd:complexType>
2566 </xsd:element>
2567 </xsd:sequence>
2568 </xsd:complexType>
2569 </xsd:element>
2570 <xsd:element name="GetIdentifySubjectResults">
2571     <xsd:complexType>
2572         <xsd:annotation>
2573             <xsd:documentation>
2574                 Retrieve the identification results for a specified token,
2575                 which was returned by the Identify Subject service.
2576             </xsd:documentation>
2577         </xsd:annotation>
2578     <xsd:sequence>
2579         <xsd:element name="GetIdentifySubjectResultsRequest" nillable="true">
2580             <xsd:complexType>
2581                 <xsd:complexContent>
2582                     <xsd:extension base="tns:RequestTemplate">
2583                         <xsd:sequence>
2584                             <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
2585                                 <xsd:annotation>
2586                                     <xsd:documentation>A value used to retrieve the results of an IdentifySubject
2587 request.</xsd:documentation>
2588                                 </xsd:annotation>
2589                             </xsd:element>
2590                         </xsd:sequence>
2591                     </xsd:extension>
2592                 </xsd:complexContent>
2593             </xsd:complexType>
2594         </xsd:element>
2595     </xsd:sequence>
2596 </xsd:complexType>
2597 </xsd:element>
2598 <xsd:element name="GetIdentifySubjectResultsResponse">
2599     <xsd:complexType>
2600         <xsd:annotation>
2601             <xsd:documentation>
2602                 The response to a GetIdentifySubjectResults request, which includes a candidate list.
2603             </xsd:documentation>
2604         </xsd:annotation>
2605     <xsd:sequence>
2606         <xsd:element name="GetIdentifySubjectResultsResponsePackage" nillable="true">

```

```

2607     <xsd:complexType>
2608     <xsd:complexContent>
2609     <xsd:extension base="tns:ResponseTemplate">
2610     <xsd:sequence>
2611     <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="1"
2612     maxOccurs="1">
2613     <xsd:annotation>
2614     <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching
2615     the input biometric sample.</xsd:documentation>
2616     </xsd:annotation>
2617     </xsd:element>
2618     </xsd:sequence>
2619     </xsd:extension>
2620     </xsd:complexContent>
2621     </xsd:complexType>
2622     </xsd:element>
2623     </xsd:sequence>
2624     </xsd:complexType>
2625     </xsd:element>
2626     <xsd:element name="IdentifySubject">
2627     <xsd:complexType>
2628     <xsd:annotation>
2629     <xsd:documentation>
2630     Perform an identification search against a given gallery for
2631     a given biometric.
2632     </xsd:documentation>
2633     </xsd:annotation>
2634     <xsd:sequence>
2635     <xsd:element name="IdentifySubjectRequest" nillable="true">
2636     <xsd:complexType>
2637     <xsd:complexContent>
2638     <xsd:extension base="tns:RequestTemplate">
2639     <xsd:sequence>
2640     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="1" maxOccurs="1">
2641     <xsd:annotation>
2642     <xsd:documentation>The identifier of the gallery or population group which will be
2643     searched.</xsd:documentation>
2644     </xsd:annotation>
2645     </xsd:element>
2646     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2647     <xsd:annotation>
2648     <xsd:documentation>Contains the BIR, a data structure containing the biometric sample
2649     for the search.</xsd:documentation>

```



```

2650         </xsd:annotation>
2651     </xsd:element>
2652     <xsd:element name="MaxListSize" type="xsd:positiveInteger" minOccurs="1"
2653 maxOccurs="1">
2654         <xsd:annotation>
2655             <xsd:documentation>The maximum size of the candidate list that should be
2656 returned.</xsd:documentation>
2657         </xsd:annotation>
2658     </xsd:element>
2659 </xsd:sequence>
2660 </xsd:extension>
2661 </xsd:complexContent>
2662 </xsd:complexType>
2663 </xsd:element>
2664 </xsd:sequence>
2665 </xsd:complexType>
2666 </xsd:element>
2667 <xsd:element name="IdentifySubjectResponse">
2668     <xsd:complexType>
2669         <xsd:annotation>
2670             <xsd:documentation>
2671                 The response to an IdentifySubject request, returning a
2672                 rank-ordered candidate list.
2673             </xsd:documentation>
2674         </xsd:annotation>
2675         <xsd:sequence>
2676             <xsd:element name="IdentifySubjectResponsePackage" nillable="true">
2677                 <xsd:complexType>
2678                     <xsd:complexContent>
2679                         <xsd:extension base="tns:ResponseTemplate">
2680                             <xsd:sequence>
2681                                 <xsd:element name="IdentifySubjectResult" type="tns:IdentifySubjectResultType" />
2682                             </xsd:sequence>
2683                         </xsd:extension>
2684                     </xsd:complexContent>
2685                 </xsd:complexType>
2686             </xsd:element>
2687         </xsd:sequence>
2688     </xsd:complexType>
2689 </xsd:element>
2690 <xsd:complexType name="IdentifySubjectResultType">
2691     <xsd:sequence />
2692 </xsd:complexType>

```

```

2693 <xsd:complexType name="CandidateListResultType">
2694 <xsd:complexContent>
2695 <xsd:extension base="tns:IdentifySubjectResultType">
2696 <xsd:sequence>
2697 <xsd:element name="CandidateList" type="tns:CandidateListType">
2698 <xsd:annotation>
2699 <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching the
2700 input biometric sample; returned with successful synchronous request processing.</xsd:documentation>
2701 </xsd:annotation>
2702 </xsd:element>
2703 </xsd:sequence>
2704 </xsd:extension>
2705 </xsd:complexContent>
2706 </xsd:complexType>
2707 <xsd:complexType name="TokenResultType">
2708 <xsd:complexContent>
2709 <xsd:extension base="tns:IdentifySubjectResultType">
2710 <xsd:sequence>
2711 <xsd:element name="Token" type="tns:TokenType">
2712 <xsd:annotation>
2713 <xsd:documentation>A token used to retrieve the results of the IdentifySubject request;
2714 returned with asynchronous request processing.</xsd:documentation>
2715 </xsd:annotation>
2716 </xsd:element>
2717 </xsd:sequence>
2718 </xsd:extension>
2719 </xsd:complexContent>
2720 </xsd:complexType>
2721 <xsd:element name="ListBiographicData">
2722 <xsd:complexType>
2723 <xsd:annotation>
2724 <xsd:documentation>
2725 Lists the biographic data elements stored for a subject.
2726 </xsd:documentation>
2727 </xsd:annotation>
2728 <xsd:sequence>
2729 <xsd:element name="ListBiographicDataRequest" nillable="true">
2730 <xsd:complexType>
2731 <xsd:complexContent>
2732 <xsd:extension base="tns:RequestTemplate">
2733 <xsd:sequence>
2734 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2735 <xsd:annotation>

```

```

2736         <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
2737 and an encounter.</xsd:documentation>
2738     </xsd:annotation>
2739 </xsd:element>
2740 </xsd:sequence>
2741 </xsd:extension>
2742 </xsd:complexContent>
2743 </xsd:complexType>
2744 </xsd:element>
2745 </xsd:sequence>
2746 </xsd:complexType>
2747 </xsd:element>
2748 <xsd:element name="ListBiographicDataResponse">
2749 <xsd:complexType>
2750 <xsd:annotation>
2751 <xsd:documentation>
2752     The response to a ListBiographicData request, containing a list
2753     of biographic data elements stored for a subject. In the
2754     encounter-centric model, the biographic data elements for a
2755     specific encounter are returned. If an encounter ID is not
2756     specified and encounter data exists for the subject, the list
2757     of encounter IDs which contain biographic data is returned.
2758 </xsd:documentation>
2759 </xsd:annotation>
2760 <xsd:sequence>
2761 <xsd:element name="ListBiographicDataResponsePackage" nillable="true">
2762 <xsd:complexType>
2763 <xsd:complexContent>
2764 <xsd:extension base="tns:ResponseTemplate">
2765 <xsd:sequence>
2766 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
2767 <xsd:annotation>
2768 <xsd:documentation>
2769     Contains a list of biographic data elements associated with a
2770     subject or encounter; non-empty if the service was
2771     successful, biographic data exists, and either (a) the
2772     person-centric model is being used or (b) the
2773     encounter-centric model is being used and an encounter
2774     identifier was specified.
2775 </xsd:documentation>
2776 </xsd:annotation>
2777 </xsd:element>

```

```

2778         <xsd:element name="EncounterList" type="tns:EncounterListType" minOccurs="0"
2779 maxOccurs="1">
2780         <xsd:annotation>
2781         <xsd:documentation>
2782             A list of encounter ID's associated with a subject and
2783             which contain biographic data; non-empty if the service
2784             was successful, biographic data exists, the
2785             encounter-centric model is being used, and an encounter
2786             identifier was not specified.
2787         </xsd:documentation>
2788         </xsd:annotation>
2789         </xsd:element>
2790     </xsd:sequence>
2791 </xsd:extension>
2792 </xsd:complexContent>
2793 </xsd:complexType>
2794 </xsd:element>
2795 </xsd:sequence>
2796 </xsd:complexType>
2797 </xsd:element>
2798 <xsd:element name="ListBiometricData">
2799     <xsd:complexType>
2800     <xsd:annotation>
2801     <xsd:documentation>
2802         Lists the biometric data elements stored for a subject. Note
2803         that no actual biometric data is returned by this service (see
2804         the RetrieveBiometricInformation service to obtain the biometric
2805         data).
2806     </xsd:documentation>
2807 </xsd:annotation>
2808 <xsd:sequence>
2809     <xsd:element name="ListBiometricDataRequest" nillable="true">
2810     <xsd:complexType>
2811     <xsd:complexContent>
2812     <xsd:extension base="tns:RequestTemplate">
2813     <xsd:sequence>
2814     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2815     <xsd:annotation>
2816     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
2817 and an encounter.</xsd:documentation>
2818     </xsd:annotation>
2819     </xsd:element>
2820     <xsd:element name="ListFilter" type="tns:ListFilterType" minOccurs="0" maxOccurs="1">

```

```

2821         <xsd:annotation>
2822             <xsd:documentation>Indicates what biometric information should be
2823 returned.</xsd:documentation>
2824         </xsd:annotation>
2825     </xsd:element>
2826 </xsd:sequence>
2827 </xsd:extension>
2828 </xsd:complexContent>
2829 </xsd:complexType>
2830 </xsd:element>
2831 </xsd:sequence>
2832 </xsd:complexType>
2833 </xsd:element>
2834 <xsd:element name="ListBiometricDataResponse">
2835     <xsd:complexType>
2836         <xsd:annotation>
2837             <xsd:documentation>
2838                 The response to a ListBiometricData request, containing a list
2839                 of biometric data elements stored for a subject. In the
2840                 encounter-centric model, the biometric data elements for a
2841                 specific encounter are returned. If an encounter ID is not
2842                 specified and encounter data exists for the subject, the list
2843                 of encounter IDs which contain biometric data is returned.
2844             </xsd:documentation>
2845         </xsd:annotation>
2846         <xsd:sequence>
2847             <xsd:element name="ListBiometricDataResponsePackage" nillable="true">
2848                 <xsd:complexType>
2849                     <xsd:complexContent>
2850                         <xsd:extension base="tns:ResponseTemplate">
2851                             <xsd:sequence>
2852                                 <xsd:element name="Identity" type="tns:BIASIdentity">
2853                                     <xsd:annotation>
2854                                         <xsd:documentation>
2855                                             Includes a list of biometric data elements associated
2856                                             with a subject or encounter or a list of encounter ID's
2857                                             associated with a subject and which contain biometric
2858                                             data.
2859                                         </xsd:documentation>
2860                                     </xsd:annotation>
2861                                 </xsd:element>
2862                             </xsd:sequence>
2863                         </xsd:extension>

```

```

2864         </xsd:complexContent>
2865     </xsd:complexType>
2866 </xsd:element>
2867 </xsd:sequence>
2868 </xsd:complexType>
2869 </xsd:element>
2870 <xsd:element name="PerformFusion">
2871     <xsd:complexType>
2872         <xsd:annotation>
2873             <xsd:documentation>
2874                 Accepts either match score or match decision information and creates a fused match result.
2875             </xsd:documentation>
2876         </xsd:annotation>
2877     <xsd:sequence>
2878         <xsd:element name="PerformFusionRequest" nillable="true">
2879             <xsd:complexType>
2880                 <xsd:complexContent>
2881                     <xsd:extension base="tns:RequestTemplate">
2882                         <xsd:sequence>
2883                             <xsd:element name="FusionInput" type="tns:FusionInformationListType" minOccurs="1"
2884 maxOccurs="1">
2885                                 <xsd:annotation>
2886                                     <xsd:documentation>Score or decision input information to the fusion
2887 method.</xsd:documentation>
2888                                 </xsd:annotation>
2889                             </xsd:element>
2890                         </xsd:sequence>
2891                     </xsd:extension>
2892                 </xsd:complexContent>
2893             </xsd:complexType>
2894         </xsd:element>
2895     </xsd:sequence>
2896 </xsd:complexType>
2897 </xsd:element>
2898 <xsd:element name="PerformFusionResponse">
2899     <xsd:complexType>
2900         <xsd:annotation>
2901             <xsd:documentation>
2902                 The response to the PerformFusion request.
2903             </xsd:documentation>
2904         </xsd:annotation>
2905     <xsd:sequence>
2906         <xsd:element name="PerformFusionResponsePackage" nillable="true">

```

```

2907     <xsd:complexType>
2908     <xsd:complexContent>
2909         <xsd:extension base="tns:ResponseTemplate">
2910         <xsd:sequence>
2911             <xsd:element name="Match" type="tns:MatchType" minOccurs="1" maxOccurs="1">
2912                 <xsd:annotation>
2913                     <xsd:documentation>Indicates the result of the fusion method</xsd:documentation>
2914                 </xsd:annotation>
2915             </xsd:element>
2916         </xsd:sequence>
2917     </xsd:extension>
2918 </xsd:complexContent>
2919 </xsd:complexType>
2920 </xsd:element>
2921 </xsd:sequence>
2922 </xsd:complexType>
2923 </xsd:element>
2924 <xsd:element name="QueryCapabilities">
2925     <xsd:complexType>
2926     <xsd:annotation>
2927     <xsd:documentation>
2928         Returns a list of the capabilities, options, galleries, etc.
2929         that are supported by the BIAS implementation.
2930     </xsd:documentation>
2931 </xsd:annotation>
2932 <xsd:sequence>
2933     <xsd:element name="QueryCapabilitiesRequest" nillable="true">
2934     <xsd:complexType>
2935     <xsd:complexContent>
2936         <xsd:extension base="tns:RequestTemplate"/>
2937     </xsd:complexContent>
2938     </xsd:complexType>
2939 </xsd:element>
2940 </xsd:sequence>
2941 </xsd:complexType>
2942 </xsd:element>
2943 <xsd:element name="QueryCapabilitiesResponse">
2944     <xsd:complexType>
2945     <xsd:annotation>
2946     <xsd:documentation>
2947         The response to a QueryCapabilities request.
2948     </xsd:documentation>

```

```

2949     </xsd:annotation>
2950     <xsd:sequence>
2951       <xsd:element name="QueryCapabilitiesResponsePackage" nillable="true">
2952         <xsd:complexType>
2953           <xsd:complexContent>
2954             <xsd:extension base="tns:ResponseTemplate">
2955               <xsd:sequence>
2956                 <xsd:element name="CapabilityList" type="tns:CapabilityListType" minOccurs="1"
2957 maxOccurs="1">
2958                   <xsd:annotation>
2959                     <xsd:documentation>A list of capabilities supported by the BIAS
2960 implementation.</xsd:documentation>
2961                   </xsd:annotation>
2962                   </xsd:element>
2963                 </xsd:sequence>
2964             </xsd:extension>
2965           </xsd:complexContent>
2966         </xsd:complexType>
2967       </xsd:element>
2968     </xsd:sequence>
2969   </xsd:complexType>
2970 </xsd:element>
2971 <xsd:element name="RetrieveBiographicInformation">
2972   <xsd:complexType>
2973     <xsd:annotation>
2974       <xsd:documentation>
2975         Retrieves the biographic data associated with a subject ID.
2976       </xsd:documentation>
2977     </xsd:annotation>
2978   <xsd:sequence>
2979     <xsd:element name="RetrieveBiographicInformationRequest" nillable="true">
2980       <xsd:complexType>
2981         <xsd:complexContent>
2982           <xsd:extension base="tns:RequestTemplate">
2983             <xsd:sequence>
2984               <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2985                 <xsd:annotation>
2986                   <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
2987 and an encounter.</xsd:documentation>
2988                 </xsd:annotation>
2989               </xsd:element>
2990             </xsd:sequence>
2991           </xsd:extension>

```



```

2992         </xsd:complexContent>
2993     </xsd:complexType>
2994 </xsd:element>
2995 </xsd:sequence>
2996 </xsd:complexType>
2997 </xsd:element>
2998 <xsd:element name="RetrieveBiographicInformationResponse">
2999     <xsd:complexType>
3000         <xsd:annotation>
3001             <xsd:documentation>
3002                 The response to a RetrieveBiographicInformation request,
3003                 containing the biographic data associated with a subject ID. In
3004                 the encounter-centric model, the biographic data associated with
3005                 a specified encounter is returned. If the encounter ID is not
3006                 specified in the encounter-centric model, the biographic
3007                 information associated with the most recent encounter is returned.
3008             </xsd:documentation>
3009         </xsd:annotation>
3010     <xsd:sequence>
3011         <xsd:element name="RetrieveBiographicInformationResponsePackage" nillable="true">
3012             <xsd:complexType>
3013                 <xsd:complexContent>
3014                     <xsd:extension base="tns:ResponseTemplate">
3015                         <xsd:sequence>
3016                             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3017                                 <xsd:annotation>
3018                                     <xsd:documentation>Includes the set of biographic data associated with a
3019                                     subject.</xsd:documentation>
3020                                 </xsd:annotation>
3021                             </xsd:element>
3022                         </xsd:sequence>
3023                     </xsd:extension>
3024                 </xsd:complexContent>
3025             </xsd:complexType>
3026         </xsd:element>
3027     </xsd:sequence>
3028 </xsd:complexType>
3029 </xsd:element>
3030 <xsd:element name="RetrieveBiometricInformation">
3031     <xsd:complexType>
3032         <xsd:annotation>
3033             <xsd:documentation>
3034                 Retrieves the biometric data associated with a subject ID.

```

```

3035     </xsd:documentation>
3036 </xsd:annotation>
3037 <xsd:sequence>
3038     <xsd:element name="RetrieveBiometricInformationRequest" nillable="true">
3039         <xsd:complexType>
3040             <xsd:complexContent>
3041                 <xsd:extension base="tns:RequestTemplate">
3042                     <xsd:sequence>
3043                         <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3044                             <xsd:annotation>
3045                                 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
3046 and an encounter.</xsd:documentation>
3047                             </xsd:annotation>
3048                         </xsd:element>
3049                         <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="0"
3050 maxOccurs="1">
3051                             <xsd:annotation>
3052                                 <xsd:documentation>The type of biological or behavioral data to
3053 retrieve.</xsd:documentation>
3054                             </xsd:annotation>
3055                         </xsd:element>
3056                     </xsd:sequence>
3057                 </xsd:extension>
3058             </xsd:complexContent>
3059         </xsd:complexType>
3060     </xsd:element>
3061 </xsd:sequence>
3062 </xsd:complexType>
3063 </xsd:element>
3064 <xsd:element name="RetrieveBiometricInformationResponse">
3065     <xsd:complexType>
3066         <xsd:annotation>
3067             <xsd:documentation>
3068                 The response to a RetrieveBiometricInformation request,
3069                 containing the biometric data associated with a subject ID. In
3070                 the encounter-centric model, the biometric data associated with
3071                 a specified encounter is returned. If the encounter ID is not
3072                 specified in the encounter-centric model, the biometric
3073                 information associated with the most recent encounter is returned.
3074             </xsd:documentation>
3075         </xsd:annotation>
3076     <xsd:sequence>
3077         <xsd:element name="RetrieveBiometricInformationResponsePackage" nillable="true">

```

```

3078     <xsd:complexType>
3079     <xsd:complexContent>
3080     <xsd:extension base="tns:ResponseTemplate">
3081     <xsd:sequence>
3082     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3083     <xsd:annotation>
3084     <xsd:documentation>Includes the biometric data associated with a
3085 subject.</xsd:documentation>
3086     </xsd:annotation>
3087     </xsd:element>
3088     </xsd:sequence>
3089     </xsd:extension>
3090     </xsd:complexContent>
3091     </xsd:complexType>
3092     </xsd:element>
3093     </xsd:sequence>
3094     </xsd:complexType>
3095     </xsd:element>
3096     <xsd:element name="SetBiographicData">
3097     <xsd:complexType>
3098     <xsd:annotation>
3099     <xsd:documentation>
3100 Associates biographic data to a given subject record.
3101     </xsd:documentation>
3102     </xsd:annotation>
3103     <xsd:sequence>
3104     <xsd:element name="SetBiographicDataRequest" nillable="true">
3105     <xsd:complexType>
3106     <xsd:complexContent>
3107     <xsd:extension base="tns:RequestTemplate">
3108     <xsd:sequence>
3109     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3110     <xsd:annotation>
3111     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
3112 and an encounter, and includes the biographic data to store.</xsd:documentation>
3113     </xsd:annotation>
3114     </xsd:element>
3115     </xsd:sequence>
3116     </xsd:extension>
3117     </xsd:complexContent>
3118     </xsd:complexType>
3119     </xsd:element>
3120     </xsd:sequence>

```

```

3121     </xsd:complexType>
3122 </xsd:element>
3123 <xsd:element name="SetBiographicDataResponse">
3124     <xsd:complexType>
3125         <xsd:annotation>
3126             <xsd:documentation>
3127                 The response to a SetBiographicData request.
3128             </xsd:documentation>
3129         </xsd:annotation>
3130         <xsd:sequence>
3131             <xsd:element name="SetBiographicDataResponsePackage" nillable="true">
3132                 <xsd:complexType>
3133                     <xsd:complexContent>
3134                         <xsd:extension base="tns:ResponseTemplate">
3135                             <xsd:sequence>
3136                                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3137                                     <xsd:annotation>
3138                                         <xsd:documentation>In an encounter-centric model, identifies the encounter ID assigned
3139 to a new encounter.</xsd:documentation>
3140                                     </xsd:annotation>
3141                                 </xsd:element>
3142                             </xsd:sequence>
3143                         </xsd:extension>
3144                     </xsd:complexContent>
3145                 </xsd:complexType>
3146             </xsd:element>
3147         </xsd:sequence>
3148     </xsd:complexType>
3149 </xsd:element>
3150 <xsd:element name="SetBiometricData">
3151     <xsd:complexType>
3152         <xsd:annotation>
3153             <xsd:documentation>
3154                 Associates biometric data to a given subject record.
3155             </xsd:documentation>
3156         </xsd:annotation>
3157         <xsd:sequence>
3158             <xsd:element name="SetBiometricDataRequest" nillable="true">
3159                 <xsd:complexType>
3160                     <xsd:complexContent>
3161                         <xsd:extension base="tns:RequestTemplate">
3162                             <xsd:sequence>
3163                                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">

```

```

3164         <xsd:annotation>
3165             <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
3166 and an encounter, and includes the biometric data to store.</xsd:documentation>
3167         </xsd:annotation>
3168     </xsd:element>
3169 </xsd:sequence>
3170 </xsd:extension>
3171 </xsd:complexContent>
3172 </xsd:complexType>
3173 </xsd:element>
3174 </xsd:sequence>
3175 </xsd:complexType>
3176 </xsd:element>
3177 <xsd:element name="SetBiometricDataResponse">
3178     <xsd:complexType>
3179         <xsd:annotation>
3180             <xsd:documentation>
3181                 The response to a SetBiometricData request.
3182             </xsd:documentation>
3183         </xsd:annotation>
3184     <xsd:sequence>
3185         <xsd:element name="SetBiometricDataResponsePackage" nillable="true">
3186             <xsd:complexType>
3187                 <xsd:complexContent>
3188                     <xsd:extension base="tns:ResponseTemplate">
3189                         <xsd:sequence>
3190                             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3191                                 <xsd:annotation>
3192                                     <xsd:documentation>In an encounter-centric model, identifies the encounter ID assigned
3193 to a new encounter.</xsd:documentation>
3194                                 </xsd:annotation>
3195                             </xsd:element>
3196                         </xsd:sequence>
3197                     </xsd:extension>
3198                 </xsd:complexContent>
3199             </xsd:complexType>
3200         </xsd:element>
3201     </xsd:sequence>
3202 </xsd:complexType>
3203 </xsd:element>
3204 <xsd:element name="TransformBiometricData">
3205     <xsd:complexType>
3206         <xsd:annotation>

```

```

3207     <xsd:documentation>
3208         Transforms or processes a given biometric in one format into a new target format.
3209     </xsd:documentation>
3210 </xsd:annotation>
3211 <xsd:sequence>
3212     <xsd:element name="TransformBiometricDataRequest" nillable="true">
3213         <xsd:complexType>
3214             <xsd:complexContent>
3215                 <xsd:extension base="tns:RequestTemplate">
3216                     <xsd:sequence>
3217                         <xsd:element name="InputBIR" type="tns:CBEFF_BIR_Type" minOccurs="1"
3218 maxOccurs="1">
3219                             <xsd:annotation>
3220                                 <xsd:documentation>Data structure containing the biometric information to be
3221 transformed.</xsd:documentation>
3222                             </xsd:annotation>
3223                         </xsd:element>
3224                         <xsd:element name="TransformOperation" type="xsd:unsignedLong" minOccurs="1"
3225 maxOccurs="1">
3226                             <xsd:annotation>
3227                                 <xsd:documentation>Value indicating the type of transformation to
3228 perform.</xsd:documentation>
3229                             </xsd:annotation>
3230                         </xsd:element>
3231                         <xsd:element name="TransformControl" type="xsd:string" minOccurs="0" maxOccurs="1">
3232                             <xsd:annotation>
3233                                 <xsd:documentation> Specifies controls for the requested transform
3234 operation.</xsd:documentation>
3235                             </xsd:annotation>
3236                         </xsd:element>
3237                     </xsd:sequence>
3238                 </xsd:extension>
3239             </xsd:complexContent>
3240         </xsd:complexType>
3241     </xsd:element>
3242 </xsd:sequence>
3243 </xsd:complexType>
3244 </xsd:element>
3245 <xsd:element name="TransformBiometricDataResponse">
3246     <xsd:complexType>
3247         <xsd:annotation>
3248             <xsd:documentation>
3249                 The response to a TransformBiometricData request.
3250             </xsd:documentation>

```

```

3251     </xsd:annotation>
3252     <xsd:sequence>
3253         <xsd:element name="TransformBiometricDataResponsePackage" nillable="true">
3254             <xsd:complexType>
3255                 <xsd:complexContent>
3256                     <xsd:extension base="tns:ResponseTemplate">
3257                         <xsd:sequence>
3258                             <xsd:element name="OutputBIR" type="tns:CBEFF_BIR_Type" minOccurs="0"
3259 maxOccurs="1">
3260                                 <xsd:annotation>
3261                                     <xsd:documentation>Data structure containing the new, transformed biometric
3262 information.</xsd:documentation>
3263                                 </xsd:annotation>
3264                                 </xsd:element>
3265                             </xsd:sequence>
3266                         </xsd:extension>
3267                     </xsd:complexContent>
3268                 </xsd:complexType>
3269             </xsd:element>
3270         </xsd:sequence>
3271     </xsd:complexType>
3272 </xsd:element>
3273 <xsd:element name="UpdateBiographicData">
3274     <xsd:complexType>
3275         <xsd:annotation>
3276             <xsd:documentation>
3277                 Updates the biographic data for a given subject record.
3278             </xsd:documentation>
3279         </xsd:annotation>
3280     <xsd:sequence>
3281         <xsd:element name="UpdateBiographicDataRequest" nillable="true">
3282             <xsd:complexType>
3283                 <xsd:complexContent>
3284                     <xsd:extension base="tns:RequestTemplate">
3285                         <xsd:sequence>
3286                             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3287                                 <xsd:annotation>
3288                                     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
3289 and an encounter, and includes the biographic data to update.</xsd:documentation>
3290                                 </xsd:annotation>
3291                             </xsd:element>
3292                         </xsd:sequence>
3293                     </xsd:extension>

```

```

3294         </xsd:complexContent>
3295     </xsd:complexType>
3296 </xsd:element>
3297 </xsd:sequence>
3298 </xsd:complexType>
3299 </xsd:element>
3300 <xsd:element name="UpdateBiographicDataResponse">
3301     <xsd:complexType>
3302         <xsd:annotation>
3303             <xsd:documentation>
3304                 The response to an UpdateBiographicData request.
3305             </xsd:documentation>
3306         </xsd:annotation>
3307         <xsd:sequence>
3308             <xsd:element name="UpdateBiographicDataResponsePackage" nillable="true">
3309                 <xsd:complexType>
3310                     <xsd:complexContent>
3311                         <xsd:extension base="tns:ResponseTemplate"/>
3312                     </xsd:complexContent>
3313                 </xsd:complexType>
3314             </xsd:element>
3315         </xsd:sequence>
3316     </xsd:complexType>
3317 </xsd:element>
3318 <xsd:element name="UpdateBiometricData">
3319     <xsd:complexType>
3320         <xsd:annotation>
3321             <xsd:documentation>
3322                 Updates a single biometric sample for a given subject record.
3323             </xsd:documentation>
3324         </xsd:annotation>
3325         <xsd:sequence>
3326             <xsd:element name="UpdateBiometricDataRequest" nillable="true">
3327                 <xsd:complexType>
3328                     <xsd:complexContent>
3329                         <xsd:extension base="tns:RequestTemplate">
3330                             <xsd:sequence>
3331                                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3332                                     <xsd:annotation>
3333                                         <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject
3334 and an encounter, and includes the biometric data to update.</xsd:documentation>
3335                                     </xsd:annotation>
3336                                 </xsd:element>

```



```

3337         <xsd:element name="Merge" type="xsd:boolean" minOccurs="0" maxOccurs="1">
3338             <xsd:annotation>
3339                 <xsd:documentation>Value indicating if the input biometric sample should be merged
3340 with any existing biometric information.</xsd:documentation>
3341             </xsd:annotation>
3342         </xsd:element>
3343     </xsd:sequence>
3344 </xsd:extension>
3345 </xsd:complexContent>
3346 </xsd:complexType>
3347 </xsd:element>
3348 </xsd:sequence>
3349 </xsd:complexType>
3350 </xsd:element>
3351 <xsd:element name="UpdateBiometricDataResponse">
3352     <xsd:complexType>
3353         <xsd:annotation>
3354             <xsd:documentation>
3355                 The response to an UpdateBiometricData request.
3356             </xsd:documentation>
3357         </xsd:annotation>
3358     <xsd:sequence>
3359         <xsd:element name="UpdateBiometricDataResponsePackage" nillable="true">
3360             <xsd:complexType>
3361                 <xsd:complexContent>
3362                     <xsd:extension base="tns:ResponseTemplate"/>
3363                 </xsd:complexContent>
3364             </xsd:complexType>
3365         </xsd:element>
3366     </xsd:sequence>
3367 </xsd:complexType>
3368 </xsd:element>
3369 <xsd:element name="VerifySubject">
3370     <xsd:complexType>
3371         <xsd:annotation>
3372             <xsd:documentation>
3373                 Performs a 1:1 verification match between a given biometric and
3374                 either a claim to identity in a given gallery or another given
3375                 biometric.
3376             </xsd:documentation>
3377         </xsd:annotation>
3378     <xsd:sequence>
3379         <xsd:element name="VerifySubjectRequest" nillable="true">

```

```

3380     <xsd:complexType>
3381     <xsd:complexContent>
3382     <xsd:extension base="tns:RequestTemplate">
3383     <xsd:sequence>
3384     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3385     <xsd:annotation>
3386     <xsd:documentation>The identifier of the gallery or population group of which the subject
3387 must be a member.</xsd:documentation>
3388     </xsd:annotation>
3389     </xsd:element>
3390     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3391     <xsd:annotation>
3392     <xsd:documentation>Includes the identifying information and/or input and reference
3393 biometric samples.</xsd:documentation>
3394     </xsd:annotation>
3395     </xsd:element>
3396     </xsd:sequence>
3397     </xsd:extension>
3398     </xsd:complexContent>
3399     </xsd:complexType>
3400     </xsd:element>
3401     </xsd:sequence>
3402     </xsd:complexType>
3403     </xsd:element>
3404     <xsd:element name="VerifySubjectResponse">
3405     <xsd:complexType>
3406     <xsd:annotation>
3407     <xsd:documentation>
3408     The response to a VerifySubject request.
3409     </xsd:documentation>
3410     </xsd:annotation>
3411     <xsd:sequence>
3412     <xsd:element name="VerifySubjectResponsePackage" nillable="true">
3413     <xsd:complexType>
3414     <xsd:complexContent>
3415     <xsd:extension base="tns:ResponseTemplate">
3416     <xsd:sequence>
3417     <xsd:element name="Match" type="xsd:boolean" minOccurs="0" maxOccurs="1">
3418     <xsd:annotation>
3419     <xsd:documentation>Indicates if the Input BIR matched either the biometric information
3420 associated with the Identity Claim or the Reference BIR.</xsd:documentation>
3421     </xsd:annotation>
3422     </xsd:element>

```

```

3423         <xsd:element name="Score" type="tns:Score" minOccurs="0" maxOccurs="1">
3424             <xsd:annotation>
3425                 <xsd:documentation>The score if the biometric information
3426 matched.</xsd:documentation>
3427             </xsd:annotation>
3428         </xsd:element>
3429     </xsd:sequence>
3430 </xsd:extension>
3431 </xsd:complexContent>
3432 </xsd:complexType>
3433 </xsd:element>
3434 </xsd:sequence>
3435 </xsd:complexType>
3436 </xsd:element>
3437 <xsd:element name="Enroll">
3438     <xsd:complexType>
3439         <xsd:annotation>
3440             <xsd:documentation>
3441                 The Enroll aggregate service adds a new subject or, in an
3442                 encounter-centric model, a new encounter to the system. This may
3443                 be accomplished in a number of different ways according to
3444                 system requirements and/or resources. If the Enroll aggregate
3445                 service is implemented as a synchronous service, the
3446                 implementing system immediately processes the request and
3447                 returns the results in the ReturnData parameter. If the Enroll
3448                 aggregate service is implemented as an asynchronous service, the
3449                 implementing system returns a token in the ReturnData
3450                 parameter, which is an indication that the request is being
3451                 handled asynchronously. In this case, the GetEnrollResults
3452                 service is used to poll for the results of the Enroll request.
3453             </xsd:documentation>
3454         </xsd:annotation>
3455         <xsd:sequence>
3456             <xsd:element name="EnrollRequest" nillable="true">
3457                 <xsd:complexType>
3458                     <xsd:complexContent>
3459                         <xsd:extension base="tns:AggregateRequestTemplate"/>
3460                     </xsd:complexContent>
3461                 </xsd:complexType>
3462             </xsd:element>
3463         </xsd:sequence>
3464     </xsd:complexType>
3465 </xsd:element>

```

```

3466 <xsd:element name="EnrollResponse">
3467   <xsd:complexType>
3468     <xsd:annotation>
3469       <xsd:documentation>The response to an Enroll request.</xsd:documentation>
3470     </xsd:annotation>
3471     <xsd:sequence>
3472       <xsd:element name="EnrollResponsePackage" nillable="true">
3473         <xsd:complexType>
3474           <xsd:complexContent>
3475             <xsd:extension base="tns:AggregateResponseTemplate"/>
3476           </xsd:complexContent>
3477         </xsd:complexType>
3478       </xsd:element>
3479     </xsd:sequence>
3480   </xsd:complexType>
3481 </xsd:element>
3482 <xsd:element name="GetEnrollResults">
3483   <xsd:complexType>
3484     <xsd:annotation>
3485       <xsd:documentation>
3486         The GetEnrollResults aggregate service retrieves the enrollment
3487         results for the specified token. This service is used in
3488         conjunction with the Enroll aggregate service. If the Enroll
3489         aggregate service is implemented as an asynchronous service, the
3490         implementing system returns a token, and the GetEnrollResults
3491         service is used to poll for the results of the original Enroll
3492         request.
3493       </xsd:documentation>
3494     </xsd:annotation>
3495     <xsd:sequence>
3496       <xsd:element name="GetEnrollResultsRequest" nillable="true">
3497         <xsd:complexType>
3498           <xsd:complexContent>
3499             <xsd:extension base="tns:RequestTemplate">
3500               <xsd:sequence>
3501                 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
3502                   <xsd:annotation>
3503                     <xsd:documentation>A value used to retrieve the results of the Enroll
3504                     request.</xsd:documentation>
3505                   </xsd:annotation>
3506                 </xsd:element>
3507               </xsd:sequence>
3508             </xsd:extension>

```

```

3509         </xsd:complexContent>
3510     </xsd:complexType>
3511 </xsd:element>
3512 </xsd:sequence>
3513 </xsd:complexType>
3514 </xsd:element>
3515 <xsd:element name="GetEnrollResultsResponse">
3516     <xsd:complexType>
3517         <xsd:annotation>
3518             <xsd:documentation>The response to a GetEnrollResults request.</xsd:documentation>
3519         </xsd:annotation>
3520         <xsd:sequence>
3521             <xsd:element name="GetEnrollResultsResponsePackage" nillable="true">
3522                 <xsd:complexType>
3523                     <xsd:complexContent>
3524                         <xsd:extension base="tns:AggregateResponseTemplate"/>
3525                     </xsd:complexContent>
3526                 </xsd:complexType>
3527             </xsd:element>
3528         </xsd:sequence>
3529     </xsd:complexType>
3530 </xsd:element>
3531 <xsd:element name="GetIdentifyResults">
3532     <xsd:complexType>
3533         <xsd:annotation>
3534             <xsd:documentation>
3535                 The GetIdentifyResults aggregate service retrieves the
3536                 identification results for the specified token. This service is
3537                 used in conjunction with the Identify aggregate service. If the
3538                 Identify aggregate service is implemented as an asynchronous
3539                 service, the implementing system returns a token, and the
3540                 GetIdentifyResults service is used to poll for the results of
3541                 the original Identify request.
3542             </xsd:documentation>
3543         </xsd:annotation>
3544         <xsd:sequence>
3545             <xsd:element name="GetIdentifyResultsRequest" nillable="true">
3546                 <xsd:complexType>
3547                     <xsd:complexContent>
3548                         <xsd:extension base="tns:RequestTemplate">
3549                             <xsd:sequence>
3550                                 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">

```

```

3551         <xsd:annotation>
3552             <xsd:documentation>A value used to retrieve the results of the Identify
3553 request.</xsd:documentation>
3554         </xsd:annotation>
3555     </xsd:element>
3556 </xsd:sequence>
3557 </xsd:extension>
3558 </xsd:complexContent>
3559 </xsd:complexType>
3560 </xsd:element>
3561 </xsd:sequence>
3562 </xsd:complexType>
3563 </xsd:element>
3564 <xsd:element name="GetIdentifyResultsResponse">
3565     <xsd:complexType>
3566         <xsd:annotation>
3567             <xsd:documentation>The response to a GetIdentifyResults request.</xsd:documentation>
3568         </xsd:annotation>
3569         <xsd:sequence>
3570             <xsd:element name="GetIdentifyResultsResponsePackage" nillable="true">
3571                 <xsd:complexType>
3572                     <xsd:complexContent>
3573                         <xsd:extension base="tns:AggregateResponseTemplate"/>
3574                     </xsd:complexContent>
3575                 </xsd:complexType>
3576             </xsd:element>
3577         </xsd:sequence>
3578     </xsd:complexType>
3579 </xsd:element>
3580 <xsd:element name="GetVerifyResults">
3581     <xsd:complexType>
3582         <xsd:annotation>
3583             <xsd:documentation>
3584                 The GetVerifyResults aggregate service retrieves the verification
3585                 results for the specified token. This service is used in
3586                 conjunction with the Verify aggregate service. If the Verify
3587                 aggregate service is implemented as an asynchronous service, the
3588                 implementing system returns a token, and the GetVerifyResults
3589                 service is used to poll for the results of the original Verify
3590                 request.
3591             </xsd:documentation>
3592         </xsd:annotation>
3593     <xsd:sequence>

```

```

3594     <xsd:element name="GetVerifyResultsRequest" nillable="true">
3595         <xsd:complexType>
3596             <xsd:complexContent>
3597                 <xsd:extension base="tns:RequestTemplate">
3598                     <xsd:sequence>
3599                         <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
3600                             <xsd:annotation>
3601                                 <xsd:documentation>A value used to retrieve the results of the Verify
3602 request.</xsd:documentation>
3603                             </xsd:annotation>
3604                         </xsd:element>
3605                     </xsd:sequence>
3606                 </xsd:extension>
3607             </xsd:complexContent>
3608         </xsd:complexType>
3609     </xsd:element>
3610 </xsd:sequence>
3611 </xsd:complexType>
3612 </xsd:element>
3613 <xsd:element name="GetVerifyResultsResponse">
3614     <xsd:complexType>
3615         <xsd:annotation>
3616             <xsd:documentation>The response to a GetVerifyResults request.</xsd:documentation>
3617         </xsd:annotation>
3618         <xsd:sequence>
3619             <xsd:element name="GetVerifyResultsResponsePackage" nillable="true">
3620                 <xsd:complexType>
3621                     <xsd:complexContent>
3622                         <xsd:extension base="tns:AggregateResponseTemplate">
3623                             <xsd:sequence>
3624                                 <xsd:element name="Match" type="xsd:boolean" minOccurs="0" maxOccurs="1">
3625                                     <xsd:annotation>
3626                                         <xsd:documentation>Indicates if the Input BIR matched either the biometric information
3627 associated with the Identity Claim or the Reference BIR.</xsd:documentation>
3628                                     </xsd:annotation>
3629                                 </xsd:element>
3630                                 <xsd:element name="Score" type="tns:Score" minOccurs="0" maxOccurs="1">
3631                                     <xsd:annotation>
3632                                         <xsd:documentation>The score if the biometric information
3633 matched.</xsd:documentation>
3634                                     </xsd:annotation>
3635                                 </xsd:element>
3636                             </xsd:sequence>

```

```

3637         </xsd:extension>
3638     </xsd:complexContent>
3639 </xsd:complexType>
3640 </xsd:element>
3641 </xsd:sequence>
3642 </xsd:complexType>
3643 </xsd:element>
3644 <xsd:element name="Identify">
3645     <xsd:complexType>
3646     <xsd:annotation>
3647     <xsd:documentation>
3648         The Identify aggregate service performs an identification
3649         function according to system requirements and/or resources. If
3650         the Identify aggregate service is implemented as a synchronous
3651         service, the implementing system immediately processes the
3652         request and returns the results in the ReturnData parameter. If
3653         the Identify aggregate service is implemented as an asynchronous
3654         service, the implementing system returns a token in the
3655         ReturnData parameter, which is an indication that the request is
3656         being handled asynchronously. In this case, the
3657         GetIdentifyResults service is used to poll for the results of
3658         the Identify request.
3659     </xsd:documentation>
3660 </xsd:annotation>
3661 <xsd:sequence>
3662     <xsd:element name="IdentifyRequest" nillable="true">
3663     <xsd:complexType>
3664     <xsd:complexContent>
3665     <xsd:extension base="tns:AggregateRequestTemplate"/>
3666     </xsd:complexContent>
3667     </xsd:complexType>
3668     </xsd:element>
3669 </xsd:sequence>
3670 </xsd:complexType>
3671 </xsd:element>
3672 <xsd:element name="IdentifyResponse">
3673     <xsd:complexType>
3674     <xsd:annotation>
3675     <xsd:documentation>The response to an Identify request.</xsd:documentation>
3676     </xsd:annotation>
3677 </xsd:sequence>
3678     <xsd:element name="IdentifyResponsePackage" nillable="true">

```



```

3679     <xsd:complexType>
3680     <xsd:complexContent>
3681     <xsd:extension base="tns:AggregateResponseTemplate"/>
3682     </xsd:complexContent>
3683     </xsd:complexType>
3684 </xsd:element>
3685 </xsd:sequence>
3686 </xsd:complexType>
3687 </xsd:element>
3688 <xsd:element name="RetrievalInformation">
3689     <xsd:complexType>
3690     <xsd:annotation>
3691     <xsd:documentation>
3692     The RetrievalInformation aggregate service retrieves requested
3693     information about a subject, or in an encounter-centric model
3694     about an encounter. In a person-centric model, this aggregate
3695     service may be used to retrieve both biographic and biometric
3696     information for a subject record. In an encounter-centric model,
3697     this aggregate service may be used to retrieve biographic and/or
3698     biometric information for either a single encounter or all
3699     encounters. Either a SubjectID or EncounterID must be specified
3700     in the Identify parameter.
3701     </xsd:documentation>
3702 </xsd:annotation>
3703 <xsd:sequence>
3704     <xsd:element name="RetrievalInformationRequest" nillable="true">
3705     <xsd:complexType>
3706     <xsd:complexContent>
3707     <xsd:extension base="tns:RequestTemplate">
3708     <xsd:sequence>
3709     <xsd:element name="ProcessingOptions" type="tns:ProcessingOptionsType"
3710 minOccurs="1" maxOccurs="1">
3711     <xsd:annotation>
3712     <xsd:documentation>Options that guide how the service request is processed, and may
3713 identify what type(s) of information should be returned.</xsd:documentation>
3714     </xsd:annotation>
3715     </xsd:element>
3716     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3717     <xsd:annotation>
3718     <xsd:documentation>Includes the identifier of the subject or
3719 encounter.</xsd:documentation>
3720     </xsd:annotation>
3721     </xsd:element>

```

```

3722         </xsd:sequence>
3723     </xsd:extension>
3724 </xsd:complexContent>
3725 </xsd:complexType>
3726 </xsd:element>
3727 </xsd:sequence>
3728 </xsd:complexType>
3729 </xsd:element>
3730 <xsd:element name="RetrieveInformationResponse">
3731     <xsd:complexType>
3732         <xsd:annotation>
3733             <xsd:documentation>The response to a RetrieveInformation request.</xsd:documentation>
3734         </xsd:annotation>
3735         <xsd:sequence>
3736             <xsd:element name="RetrieveInformationResponsePackage" nillable="true">
3737                 <xsd:complexType>
3738                     <xsd:complexContent>
3739                         <xsd:extension base="tns:AggregateResponseTemplate"/>
3740                     </xsd:complexContent>
3741                 </xsd:complexType>
3742             </xsd:element>
3743         </xsd:sequence>
3744     </xsd:complexType>
3745 </xsd:element>
3746 <xsd:element name="Verify">
3747     <xsd:complexType>
3748         <xsd:annotation>
3749             <xsd:documentation>
3750                 The Verify aggregate service performs a 1:1 verification
3751                 function according to system requirements and/or resources.
3752                 Either the IdentityClaim or ReferenceBIR input data elements in
3753                 the Identity parameter are required. If the Verify aggregate
3754                 service is implemented as a synchronous service, the
3755                 implementing system immediately processes the request and returns
3756                 the results in the ReturnData parameter. If the Verify aggregate
3757                 service is implemented as an asynchronous service, the
3758                 implementing system returns a token in the ReturnData parameter,
3759                 which is an indication that the request is being handled
3760                 asynchronously. In this case, the GetVerifyResults service is
3761                 used to poll for the results of the Verify request.
3762             </xsd:documentation>
3763         </xsd:annotation>

```

```

3764     <xsd:sequence>
3765     <xsd:element name="VerifyRequest" nillable="true">
3766     <xsd:complexType>
3767     <xsd:complexContent>
3768     <xsd:extension base="tns:AggregateRequestTemplate">
3769     <xsd:sequence>
3770     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3771     <xsd:annotation>
3772     <xsd:documentation>Includes either the IdentityClaim or
3773 ReferenceBIR.</xsd:documentation>
3774     </xsd:annotation>
3775     </xsd:element>
3776     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3777     <xsd:annotation>
3778     <xsd:documentation>The identifier of the gallery or population group of which the subject
3779 must be a member.</xsd:documentation>
3780     </xsd:annotation>
3781     </xsd:element>
3782     </xsd:sequence>
3783     </xsd:extension>
3784     </xsd:complexContent>
3785     </xsd:complexType>
3786     </xsd:element>
3787     </xsd:sequence>
3788     </xsd:complexType>
3789     </xsd:element>
3790     <xsd:element name="VerifyResponse">
3791     <xsd:complexType>
3792     <xsd:annotation>
3793     <xsd:documentation>The response to a Verify request.</xsd:documentation>
3794     </xsd:annotation>
3795     <xsd:sequence>
3796     <xsd:element name="VerifyResponsePackage" nillable="true">
3797     <xsd:complexType>
3798     <xsd:complexContent>
3799     <xsd:extension base="tns:AggregateResponseTemplate">
3800     <xsd:sequence>
3801     <xsd:element name="Match" type="xsd:boolean" minOccurs="0" maxOccurs="1">
3802     <xsd:annotation>
3803     <xsd:documentation>Indicates if the Input BIR matched either the biometric information
3804 associated with the Identity Claim or the Reference BIR.</xsd:documentation>
3805     </xsd:annotation>
3806     </xsd:element>

```

```

3807         <xsd:element name="Score" type="tns:Score" minOccurs="0" maxOccurs="1">
3808             <xsd:annotation>
3809                 <xsd:documentation>The score if the biometric information
3810 matched.</xsd:documentation>
3811             </xsd:annotation>
3812         </xsd:element>
3813     </xsd:sequence>
3814 </xsd:extension>
3815 </xsd:complexContent>
3816 </xsd:complexType>
3817 </xsd:element>
3818 </xsd:sequence>
3819 </xsd:complexType>
3820 </xsd:element>
3821 <xsd:element name="BIASFault" type="tns:BIASFaultDetail"/>
3822 </xsd:schema>
3823 </wsdl:types>
3824 <wsdl:message name="BIASFaultMessage">
3825     <wsdl:part name="parameters" element="tns:BIASFault"/>
3826 </wsdl:message>
3827 <wsdl:message name="AddSubjectToGalleryRequestMessage">
3828     <wsdl:part name="parameters" element="tns:AddSubjectToGallery"/>
3829 </wsdl:message>
3830 <wsdl:message name="AddSubjectToGalleryResponseMessage">
3831     <wsdl:part name="parameters" element="tns:AddSubjectToGalleryResponse"/>
3832 </wsdl:message>
3833 <wsdl:message name="CheckQualityRequestMessage">
3834     <wsdl:part name="parameters" element="tns:CheckQuality"/>
3835 </wsdl:message>
3836 <wsdl:message name="CheckQualityResponseMessage">
3837     <wsdl:part name="parameters" element="tns:CheckQualityResponse"/>
3838 </wsdl:message>
3839 <wsdl:message name="ClassifyBiometricDataRequestMessage">
3840     <wsdl:part name="parameters" element="tns:ClassifyBiometricData"/>
3841 </wsdl:message>
3842 <wsdl:message name="ClassifyBiometricDataResponseMessage">
3843     <wsdl:part name="parameters" element="tns:ClassifyBiometricDataResponse"/>
3844 </wsdl:message>
3845 <wsdl:message name="CreateSubjectRequestMessage">
3846     <wsdl:part name="parameters" element="tns:CreateSubject"/>
3847 </wsdl:message>
3848 <wsdl:message name="CreateSubjectResponseMessage">
3849     <wsdl:part name="parameters" element="tns:CreateSubjectResponse"/>

```

3850 </wsdl:message>  
3851 <wsdl:message name="DeleteBiographicDataRequestMessage">  
3852 <wsdl:part name="parameters" element="tns:DeleteBiographicData"/>  
3853 </wsdl:message>  
3854 <wsdl:message name="DeleteBiographicDataResponseMessage">  
3855 <wsdl:part name="parameters" element="tns:DeleteBiographicDataResponse"/>  
3856 </wsdl:message>  
3857 <wsdl:message name="DeleteBiometricDataRequestMessage">  
3858 <wsdl:part name="parameters" element="tns:DeleteBiometricData"/>  
3859 </wsdl:message>  
3860 <wsdl:message name="DeleteBiometricDataResponseMessage">  
3861 <wsdl:part name="parameters" element="tns:DeleteBiometricDataResponse"/>  
3862 </wsdl:message>  
3863 <wsdl:message name="DeleteSubjectRequestMessage">  
3864 <wsdl:part name="parameters" element="tns:DeleteSubject"/>  
3865 </wsdl:message>  
3866 <wsdl:message name="DeleteSubjectResponseMessage">  
3867 <wsdl:part name="parameters" element="tns:DeleteSubjectResponse"/>  
3868 </wsdl:message>  
3869 <wsdl:message name="DeleteSubjectFromGalleryRequestMessage">  
3870 <wsdl:part name="parameters" element="tns:DeleteSubjectFromGallery"/>  
3871 </wsdl:message>  
3872 <wsdl:message name="DeleteSubjectFromGalleryResponseMessage">  
3873 <wsdl:part name="parameters" element="tns:DeleteSubjectFromGalleryResponse"/>  
3874 </wsdl:message>  
3875 <wsdl:message name="GetIdentifySubjectResultsRequestMessage">  
3876 <wsdl:part name="parameters" element="tns:GetIdentifySubjectResults"/>  
3877 </wsdl:message>  
3878 <wsdl:message name="GetIdentifySubjectResultsResponseMessage">  
3879 <wsdl:part name="parameters" element="tns:GetIdentifySubjectResultsResponse"/>  
3880 </wsdl:message>  
3881 <wsdl:message name="IdentifySubjectRequestMessage">  
3882 <wsdl:part name="parameters" element="tns:IdentifySubject"/>  
3883 </wsdl:message>  
3884 <wsdl:message name="IdentifySubjectResponseMessage">  
3885 <wsdl:part name="parameters" element="tns:IdentifySubjectResponse"/>  
3886 </wsdl:message>  
3887 <wsdl:message name="ListBiographicDataRequestMessage">  
3888 <wsdl:part name="parameters" element="tns:ListBiographicData"/>  
3889 </wsdl:message>  
3890 <wsdl:message name="ListBiographicDataResponseMessage">  
3891 <wsdl:part name="parameters" element="tns:ListBiographicDataResponse"/>

```

3892 </wsdl:message>
3893 <wsdl:message name="ListBiometricDataRequestMessage">
3894   <wsdl:part name="parameters" element="tns:ListBiometricData"/>
3895 </wsdl:message>
3896 <wsdl:message name="ListBiometricDataResponseMessage">
3897   <wsdl:part name="parameters" element="tns:ListBiometricDataResponse"/>
3898 </wsdl:message>
3899 <wsdl:message name="PerformFusionRequestMessage">
3900   <wsdl:part name="parameters" element="tns:PerformFusion"/>
3901 </wsdl:message>
3902 <wsdl:message name="PerformFusionResponseMessage">
3903   <wsdl:part name="parameters" element="tns:PerformFusionResponse"/>
3904 </wsdl:message>
3905 <wsdl:message name="QueryCapabilitiesRequestMessage">
3906   <wsdl:part name="parameters" element="tns:QueryCapabilities"/>
3907 </wsdl:message>
3908 <wsdl:message name="QueryCapabilitiesResponseMessage">
3909   <wsdl:part name="parameters" element="tns:QueryCapabilitiesResponse"/>
3910 </wsdl:message>
3911 <wsdl:message name="RetrieveBiographicInformationRequestMessage">
3912   <wsdl:part name="parameters" element="tns:RetrieveBiographicInformation"/>
3913 </wsdl:message>
3914 <wsdl:message name="RetrieveBiographicInformationResponseMessage">
3915   <wsdl:part name="parameters" element="tns:RetrieveBiographicInformationResponse"/>
3916 </wsdl:message>
3917 <wsdl:message name="RetrieveBiometricInformationRequestMessage">
3918   <wsdl:part name="parameters" element="tns:RetrieveBiometricInformation"/>
3919 </wsdl:message>
3920 <wsdl:message name="RetrieveBiometricInformationResponseMessage">
3921   <wsdl:part name="parameters" element="tns:RetrieveBiometricInformationResponse"/>
3922 </wsdl:message>
3923 <wsdl:message name="SetBiographicDataRequestMessage">
3924   <wsdl:part name="parameters" element="tns:SetBiographicData"/>
3925 </wsdl:message>
3926 <wsdl:message name="SetBiographicDataResponseMessage">
3927   <wsdl:part name="parameters" element="tns:SetBiographicDataResponse"/>
3928 </wsdl:message>
3929 <wsdl:message name="SetBiometricDataRequestMessage">
3930   <wsdl:part name="parameters" element="tns:SetBiometricData"/>
3931 </wsdl:message>
3932 <wsdl:message name="SetBiometricDataResponseMessage">
3933   <wsdl:part name="parameters" element="tns:SetBiometricDataResponse"/>

```

3934 </wsdl:message>  
3935 <wsdl:message name="TransformBiometricDataRequestMessage">  
3936 <wsdl:part name="parameters" element="tns:TransformBiometricData"/>  
3937 </wsdl:message>  
3938 <wsdl:message name="TransformBiometricDataResponseMessage">  
3939 <wsdl:part name="parameters" element="tns:TransformBiometricDataResponse"/>  
3940 </wsdl:message>  
3941 <wsdl:message name="UpdateBiographicDataRequestMessage">  
3942 <wsdl:part name="parameters" element="tns:UpdateBiographicData"/>  
3943 </wsdl:message>  
3944 <wsdl:message name="UpdateBiographicDataResponseMessage">  
3945 <wsdl:part name="parameters" element="tns:UpdateBiographicDataResponse"/>  
3946 </wsdl:message>  
3947 <wsdl:message name="UpdateBiometricDataRequestMessage">  
3948 <wsdl:part name="parameters" element="tns:UpdateBiometricData"/>  
3949 </wsdl:message>  
3950 <wsdl:message name="UpdateBiometricDataResponseMessage">  
3951 <wsdl:part name="parameters" element="tns:UpdateBiometricDataResponse"/>  
3952 </wsdl:message>  
3953 <wsdl:message name="VerifySubjectRequestMessage">  
3954 <wsdl:part name="parameters" element="tns:VerifySubject"/>  
3955 </wsdl:message>  
3956 <wsdl:message name="VerifySubjectResponseMessage">  
3957 <wsdl:part name="parameters" element="tns:VerifySubjectResponse"/>  
3958 </wsdl:message>  
3959 <wsdl:message name="EnrollRequestMessage">  
3960 <wsdl:part name="parameters" element="tns:Enroll"/>  
3961 </wsdl:message>  
3962 <wsdl:message name="EnrollResponseMessage">  
3963 <wsdl:part name="parameters" element="tns:EnrollResponse"/>  
3964 </wsdl:message>  
3965 <wsdl:message name="GetEnrollResultsRequestMessage">  
3966 <wsdl:part name="parameters" element="tns:GetEnrollResults"/>  
3967 </wsdl:message>  
3968 <wsdl:message name="GetEnrollResultsResponseMessage">  
3969 <wsdl:part name="parameters" element="tns:GetEnrollResultsResponse"/>  
3970 </wsdl:message>  
3971 <wsdl:message name="GetIdentifyResultsRequestMessage">  
3972 <wsdl:part name="parameters" element="tns:GetIdentifyResults"/>  
3973 </wsdl:message>  
3974 <wsdl:message name="GetIdentifyResultsResponseMessage">  
3975 <wsdl:part name="parameters" element="tns:GetIdentifyResultsResponse"/>

```

3976 </wsdl:message>
3977 <wsdl:message name="GetVerifyResultsRequestMessage">
3978   <wsdl:part name="parameters" element="tns:GetVerifyResults"/>
3979 </wsdl:message>
3980 <wsdl:message name="GetVerifyResultsResponseMessage">
3981   <wsdl:part name="parameters" element="tns:GetVerifyResultsResponse"/>
3982 </wsdl:message>
3983 <wsdl:message name="IdentifyRequestMessage">
3984   <wsdl:part name="parameters" element="tns:Identify"/>
3985 </wsdl:message>
3986 <wsdl:message name="IdentifyResponseMessage">
3987   <wsdl:part name="parameters" element="tns:IdentifyResponse"/>
3988 </wsdl:message>
3989 <wsdl:message name="RetrieveInformationRequestMessage">
3990   <wsdl:part name="parameters" element="tns:RetrieveInformation"/>
3991 </wsdl:message>
3992 <wsdl:message name="RetrieveInformationResponseMessage">
3993   <wsdl:part name="parameters" element="tns:RetrieveInformationResponse"/>
3994 </wsdl:message>
3995 <wsdl:message name="VerifyRequestMessage">
3996   <wsdl:part name="parameters" element="tns:Verify"/>
3997 </wsdl:message>
3998 <wsdl:message name="VerifyResponseMessage">
3999   <wsdl:part name="parameters" element="tns:VerifyResponse"/>
4000 </wsdl:message>
4001 <wsdl:portType name="BIAS_v1">
4002   <wsdl:operation name="AddSubjectToGallery">
4003     <wsdl:input name="AddSubjectToGallery" message="tns:AddSubjectToGalleryRequestMessage"/>
4004     <wsdl:output name="AddSubjectToGalleryResponse"
4005 message="tns:AddSubjectToGalleryResponseMessage"/>
4006     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4007   </wsdl:operation>
4008   <wsdl:operation name="CheckQuality">
4009     <wsdl:input name="CheckQuality" message="tns:CheckQualityRequestMessage"/>
4010     <wsdl:output name="CheckQualityResponse" message="tns:CheckQualityResponseMessage"/>
4011     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4012   </wsdl:operation>
4013   <wsdl:operation name="ClassifyBiometricData">
4014     <wsdl:input name="ClassifyBiometricData" message="tns:ClassifyBiometricDataRequestMessage"/>
4015     <wsdl:output name="ClassifyBiometricDataResponse"
4016 message="tns:ClassifyBiometricDataResponseMessage"/>
4017     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4018   </wsdl:operation>

```



```

4019 <wsdl:operation name="CreateSubject">
4020   <wsdl:input name="CreateSubject" message="tns:CreateSubjectRequestMessage"/>
4021   <wsdl:output name="CreateSubjectResponse" message="tns:CreateSubjectResponseMessage"/>
4022   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4023 </wsdl:operation>
4024 <wsdl:operation name="DeleteBiographicData">
4025   <wsdl:input name="DeleteBiographicData" message="tns:DeleteBiographicDataRequestMessage"/>
4026   <wsdl:output name="DeleteBiographicDataResponse"
4027 message="tns:DeleteBiographicDataResponseMessage"/>
4028   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4029 </wsdl:operation>
4030 <wsdl:operation name="DeleteBiometricData">
4031   <wsdl:input name="DeleteBiometricData" message="tns:DeleteBiometricDataRequestMessage"/>
4032   <wsdl:output name="DeleteBiometricDataResponse"
4033 message="tns:DeleteBiometricDataResponseMessage"/>
4034   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4035 </wsdl:operation>
4036 <wsdl:operation name="DeleteSubject">
4037   <wsdl:input name="DeleteSubject" message="tns:DeleteSubjectRequestMessage"/>
4038   <wsdl:output name="DeleteSubjectResponse" message="tns:DeleteSubjectResponseMessage"/>
4039   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4040 </wsdl:operation>
4041 <wsdl:operation name="DeleteSubjectFromGallery">
4042   <wsdl:input name="DeleteSubjectFromGallery"
4043 message="tns:DeleteSubjectFromGalleryRequestMessage"/>
4044   <wsdl:output name="DeleteSubjectFromGalleryResponse"
4045 message="tns:DeleteSubjectFromGalleryResponseMessage"/>
4046   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4047 </wsdl:operation>
4048 <wsdl:operation name="GetIdentifySubjectResults">
4049   <wsdl:input name="GetIdentifySubjectResults"
4050 message="tns:GetIdentifySubjectResultsRequestMessage"/>
4051   <wsdl:output name="GetIdentifySubjectResultsResponse"
4052 message="tns:GetIdentifySubjectResultsResponseMessage"/>
4053   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4054 </wsdl:operation>
4055 <wsdl:operation name="IdentifySubject">
4056   <wsdl:input name="IdentifySubject" message="tns:IdentifySubjectRequestMessage"/>
4057   <wsdl:output name="IdentifySubjectResponse" message="tns:IdentifySubjectResponseMessage"/>
4058   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4059 </wsdl:operation>
4060 <wsdl:operation name="ListBiographicData">
4061   <wsdl:input name="ListBiographicData" message="tns:ListBiographicDataRequestMessage"/>

```

```

4062     <wsdl:output name="ListBiographicDataResponse"
4063 message="tns:ListBiographicDataResponseMessage"/>
4064     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4065 </wsdl:operation>
4066 <wsdl:operation name="ListBiometricData">
4067     <wsdl:input name="ListBiometricData" message="tns:ListBiometricDataRequestMessage"/>
4068     <wsdl:output name="ListBiometricDataResponse"
4069 message="tns:ListBiometricDataResponseMessage"/>
4070     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4071 </wsdl:operation>
4072 <wsdl:operation name="PerformFusion">
4073     <wsdl:input name="PerformFusion" message="tns:PerformFusionRequestMessage"/>
4074     <wsdl:output name="PerformFusionResponse" message="tns:PerformFusionResponseMessage"/>
4075     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4076 </wsdl:operation>
4077 <wsdl:operation name="QueryCapabilities">
4078     <wsdl:input name="QueryCapabilities" message="tns:QueryCapabilitiesRequestMessage"/>
4079     <wsdl:output name="QueryCapabilitiesResponse"
4080 message="tns:QueryCapabilitiesResponseMessage"/>
4081     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4082 </wsdl:operation>
4083 <wsdl:operation name="RetrieveBiographicInformation">
4084     <wsdl:input name="RetrieveBiographicInformation"
4085 message="tns:RetrieveBiographicInformationRequestMessage"/>
4086     <wsdl:output name="RetrieveBiographicInformationResponse"
4087 message="tns:RetrieveBiographicInformationResponseMessage"/>
4088     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4089 </wsdl:operation>
4090 <wsdl:operation name="RetrieveBiometricInformation">
4091     <wsdl:input name="RetrieveBiometricInformation"
4092 message="tns:RetrieveBiometricInformationRequestMessage"/>
4093     <wsdl:output name="RetrieveBiometricInformationResponse"
4094 message="tns:RetrieveBiometricInformationResponseMessage"/>
4095     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4096 </wsdl:operation>
4097 <wsdl:operation name="SetBiographicData">
4098     <wsdl:input name="SetBiographicData" message="tns:SetBiographicDataRequestMessage"/>
4099     <wsdl:output name="SetBiographicDataResponse"
4100 message="tns:SetBiographicDataResponseMessage"/>
4101     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4102 </wsdl:operation>
4103 <wsdl:operation name="SetBiometricData">
4104     <wsdl:input name="SetBiometricData" message="tns:SetBiometricDataRequestMessage"/>
4105     <wsdl:output name="SetBiometricDataResponse"
4106 message="tns:SetBiometricDataResponseMessage"/>

```

```

4107     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4108 </wsdl:operation>
4109 <wsdl:operation name="TransformBiometricData">
4110     <wsdl:input name="TransformBiometricData"
4111 message="tns:TransformBiometricDataRequestMessage"/>
4112     <wsdl:output name="TransformBiometricDataResponse"
4113 message="tns:TransformBiometricDataResponseMessage"/>
4114     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4115 </wsdl:operation>
4116 <wsdl:operation name="UpdateBiographicData">
4117     <wsdl:input name="UpdateBiographicData" message="tns:UpdateBiographicDataRequestMessage"/>
4118     <wsdl:output name="UpdateBiographicDataResponse"
4119 message="tns:UpdateBiographicDataResponseMessage"/>
4120     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4121 </wsdl:operation>
4122 <wsdl:operation name="UpdateBiometricData">
4123     <wsdl:input name="UpdateBiometricData" message="tns:UpdateBiometricDataRequestMessage"/>
4124     <wsdl:output name="UpdateBiometricDataResponse"
4125 message="tns:UpdateBiometricDataResponseMessage"/>
4126     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4127 </wsdl:operation>
4128 <wsdl:operation name="VerifySubject">
4129     <wsdl:input name="VerifySubject" message="tns:VerifySubjectRequestMessage"/>
4130     <wsdl:output name="VerifySubjectResponse" message="tns:VerifySubjectResponseMessage"/>
4131     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4132 </wsdl:operation>
4133 <wsdl:operation name="Enroll">
4134     <wsdl:input name="Enroll" message="tns:EnrollRequestMessage"/>
4135     <wsdl:output name="EnrollResponse" message="tns:EnrollResponseMessage"/>
4136     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4137 </wsdl:operation>
4138 <wsdl:operation name="GetEnrollResults">
4139     <wsdl:input name="GetEnrollResults" message="tns:GetEnrollResultsRequestMessage"/>
4140     <wsdl:output name="GetEnrollResultsResponse"
4141 message="tns:GetEnrollResultsResponseMessage"/>
4142     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4143 </wsdl:operation>
4144 <wsdl:operation name="GetIdentifyResults">
4145     <wsdl:input name="GetIdentifyResults" message="tns:GetIdentifyResultsRequestMessage"/>
4146     <wsdl:output name="GetIdentifyResultsResponse"
4147 message="tns:GetIdentifyResultsResponseMessage"/>
4148     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4149 </wsdl:operation>
4150 <wsdl:operation name="GetVerifyResults">

```

```

4151     <wsdl:input name="GetVerifyResults" message="tns:GetVerifyResultsRequestMessage"/>
4152     <wsdl:output name="GetVerifyResultsResponse"
4153 message="tns:GetVerifyResultsResponseMessage"/>
4154     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4155 </wsdl:operation>
4156 <wsdl:operation name="Identify">
4157     <wsdl:input name="Identify" message="tns:IdentifyRequestMessage"/>
4158     <wsdl:output name="IdentifyResponse" message="tns:IdentifyResponseMessage"/>
4159     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4160 </wsdl:operation>
4161 <wsdl:operation name="RetrieveInformation">
4162     <wsdl:input name="RetrieveInformation" message="tns:RetrieveInformationRequestMessage"/>
4163     <wsdl:output name="RetrieveInformationResponse"
4164 message="tns:RetrieveInformationResponseMessage"/>
4165     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4166 </wsdl:operation>
4167 <wsdl:operation name="Verify">
4168     <wsdl:input name="Verify" message="tns:VerifyRequestMessage"/>
4169     <wsdl:output name="VerifyResponse" message="tns:VerifyResponseMessage"/>
4170     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4171 </wsdl:operation>
4172 </wsdl:portType>
4173 <wsdl:binding name="BIAS_v1HttpBinding" type="tns:BIAS_v1">
4174     <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
4175     <wsdl:operation name="AddSubjectToGallery">
4176         <soap:operation soapAction="AddSubjectToGallery"/>
4177         <wsdl:input name="AddSubjectToGallery">
4178             <soap:body use="literal"/>
4179         </wsdl:input>
4180         <wsdl:output name="AddSubjectToGalleryResponse">
4181             <soap:body use="literal"/>
4182         </wsdl:output>
4183         <wsdl:fault name="BIASFault">
4184             <soap:fault name="BIASFault" use="literal"/>
4185         </wsdl:fault>
4186     </wsdl:operation>
4187     <wsdl:operation name="CheckQuality">
4188         <soap:operation soapAction="CheckQuality"/>
4189         <wsdl:input name="CheckQuality">
4190             <soap:body use="literal"/>
4191         </wsdl:input>
4192         <wsdl:output name="CheckQualityResponse">
4193             <soap:body use="literal"/>

```

```

4194     </wsdl:output>
4195     <wsdl:fault name="BIASFault">
4196         <soap:fault name="BIASFault" use="literal"/>
4197     </wsdl:fault>
4198 </wsdl:operation>
4199 <wsdl:operation name="ClassifyBiometricData">
4200     <soap:operation soapAction="ClassifyBiometricData"/>
4201     <wsdl:input name="ClassifyBiometricData">
4202         <soap:body use="literal"/>
4203     </wsdl:input>
4204     <wsdl:output name="ClassifyBiometricDataResponse">
4205         <soap:body use="literal"/>
4206     </wsdl:output>
4207     <wsdl:fault name="BIASFault">
4208         <soap:fault name="BIASFault" use="literal"/>
4209     </wsdl:fault>
4210 </wsdl:operation>
4211 <wsdl:operation name="CreateSubject">
4212     <soap:operation soapAction="CreateSubject"/>
4213     <wsdl:input name="CreateSubject">
4214         <soap:body use="literal"/>
4215     </wsdl:input>
4216     <wsdl:output name="CreateSubjectResponse">
4217         <soap:body use="literal"/>
4218     </wsdl:output>
4219     <wsdl:fault name="BIASFault">
4220         <soap:fault name="BIASFault" use="literal"/>
4221     </wsdl:fault>
4222 </wsdl:operation>
4223 <wsdl:operation name="DeleteBiographicData">
4224     <soap:operation soapAction="DeleteBiographicData"/>
4225     <wsdl:input name="DeleteBiographicData">
4226         <soap:body use="literal"/>
4227     </wsdl:input>
4228     <wsdl:output name="DeleteBiographicDataResponse">
4229         <soap:body use="literal"/>
4230     </wsdl:output>
4231     <wsdl:fault name="BIASFault">
4232         <soap:fault name="BIASFault" use="literal"/>
4233     </wsdl:fault>
4234 </wsdl:operation>
4235 <wsdl:operation name="DeleteBiometricData">

```

```

4236     <soap:operation soapAction="DeleteBiometricData"/>
4237     <wsdl:input name="DeleteBiometricData">
4238         <soap:body use="literal"/>
4239     </wsdl:input>
4240     <wsdl:output name="DeleteBiometricDataResponse">
4241         <soap:body use="literal"/>
4242     </wsdl:output>
4243     <wsdl:fault name="BIASFault">
4244         <soap:fault name="BIASFault" use="literal"/>
4245     </wsdl:fault>
4246 </wsdl:operation>
4247 <wsdl:operation name="DeleteSubject">
4248     <soap:operation soapAction="DeleteSubject"/>
4249     <wsdl:input name="DeleteSubject">
4250         <soap:body use="literal"/>
4251     </wsdl:input>
4252     <wsdl:output name="DeleteSubjectResponse">
4253         <soap:body use="literal"/>
4254     </wsdl:output>
4255     <wsdl:fault name="BIASFault">
4256         <soap:fault name="BIASFault" use="literal"/>
4257     </wsdl:fault>
4258 </wsdl:operation>
4259 <wsdl:operation name="DeleteSubjectFromGallery">
4260     <soap:operation soapAction="DeleteSubjectFromGallery"/>
4261     <wsdl:input name="DeleteSubjectFromGallery">
4262         <soap:body use="literal"/>
4263     </wsdl:input>
4264     <wsdl:output name="DeleteSubjectFromGalleryResponse">
4265         <soap:body use="literal"/>
4266     </wsdl:output>
4267     <wsdl:fault name="BIASFault">
4268         <soap:fault name="BIASFault" use="literal"/>
4269     </wsdl:fault>
4270 </wsdl:operation>
4271 <wsdl:operation name="GetIdentifySubjectResults">
4272     <soap:operation soapAction="GetIdentifySubjectResults"/>
4273     <wsdl:input name="GetIdentifySubjectResults">
4274         <soap:body use="literal"/>
4275     </wsdl:input>
4276     <wsdl:output name="GetIdentifySubjectResultsResponse">
4277         <soap:body use="literal"/>

```

```

4278     </wsdl:output>
4279     <wsdl:fault name="BIASFault">
4280         <soap:fault name="BIASFault" use="literal"/>
4281     </wsdl:fault>
4282 </wsdl:operation>
4283 <wsdl:operation name="IdentifySubject">
4284     <soap:operation soapAction="IdentifySubject"/>
4285     <wsdl:input name="IdentifySubject">
4286         <soap:body use="literal"/>
4287     </wsdl:input>
4288     <wsdl:output name="IdentifySubjectResponse">
4289         <soap:body use="literal"/>
4290     </wsdl:output>
4291     <wsdl:fault name="BIASFault">
4292         <soap:fault name="BIASFault" use="literal"/>
4293     </wsdl:fault>
4294 </wsdl:operation>
4295 <wsdl:operation name="ListBiographicData">
4296     <soap:operation soapAction="ListBiographicData"/>
4297     <wsdl:input name="ListBiographicData">
4298         <soap:body use="literal"/>
4299     </wsdl:input>
4300     <wsdl:output name="ListBiographicDataResponse">
4301         <soap:body use="literal"/>
4302     </wsdl:output>
4303     <wsdl:fault name="BIASFault">
4304         <soap:fault name="BIASFault" use="literal"/>
4305     </wsdl:fault>
4306 </wsdl:operation>
4307 <wsdl:operation name="ListBiometricData">
4308     <soap:operation soapAction="ListBiometricData"/>
4309     <wsdl:input name="ListBiometricData">
4310         <soap:body use="literal"/>
4311     </wsdl:input>
4312     <wsdl:output name="ListBiometricDataResponse">
4313         <soap:body use="literal"/>
4314     </wsdl:output>
4315     <wsdl:fault name="BIASFault">
4316         <soap:fault name="BIASFault" use="literal"/>
4317     </wsdl:fault>
4318 </wsdl:operation>
4319 <wsdl:operation name="PerformFusion">

```

```

4320     <soap:operation soapAction="PerformFusion"/>
4321     <wsdl:input name="PerformFusion">
4322         <soap:body use="literal"/>
4323     </wsdl:input>
4324     <wsdl:output name="PerformFusionResponse">
4325         <soap:body use="literal"/>
4326     </wsdl:output>
4327     <wsdl:fault name="BIASFault">
4328         <soap:fault name="BIASFault" use="literal"/>
4329     </wsdl:fault>
4330 </wsdl:operation>
4331 <wsdl:operation name="QueryCapabilities">
4332     <soap:operation soapAction="QueryCapabilities"/>
4333     <wsdl:input name="QueryCapabilities">
4334         <soap:body use="literal"/>
4335     </wsdl:input>
4336     <wsdl:output name="QueryCapabilitiesResponse">
4337         <soap:body use="literal"/>
4338     </wsdl:output>
4339     <wsdl:fault name="BIASFault">
4340         <soap:fault name="BIASFault" use="literal"/>
4341     </wsdl:fault>
4342 </wsdl:operation>
4343 <wsdl:operation name="RetrieveBiographicInformation">
4344     <soap:operation soapAction="RetrieveBiographicInformation"/>
4345     <wsdl:input name="RetrieveBiographicInformation">
4346         <soap:body use="literal"/>
4347     </wsdl:input>
4348     <wsdl:output name="RetrieveBiographicInformationResponse">
4349         <soap:body use="literal"/>
4350     </wsdl:output>
4351     <wsdl:fault name="BIASFault">
4352         <soap:fault name="BIASFault" use="literal"/>
4353     </wsdl:fault>
4354 </wsdl:operation>
4355 <wsdl:operation name="RetrieveBiometricInformation">
4356     <soap:operation soapAction="RetrieveBiometricInformation"/>
4357     <wsdl:input name="RetrieveBiometricInformation">
4358         <soap:body use="literal"/>
4359     </wsdl:input>
4360     <wsdl:output name="RetrieveBiometricInformationResponse">
4361         <soap:body use="literal"/>

```



```

4362     </wsdl:output>
4363     <wsdl:fault name="BIASFault">
4364         <soap:fault name="BIASFault" use="literal"/>
4365     </wsdl:fault>
4366 </wsdl:operation>
4367 <wsdl:operation name="SetBiographicData">
4368     <soap:operation soapAction="SetBiographicData"/>
4369     <wsdl:input name="SetBiographicData">
4370         <soap:body use="literal"/>
4371     </wsdl:input>
4372     <wsdl:output name="SetBiographicDataResponse">
4373         <soap:body use="literal"/>
4374     </wsdl:output>
4375     <wsdl:fault name="BIASFault">
4376         <soap:fault name="BIASFault" use="literal"/>
4377     </wsdl:fault>
4378 </wsdl:operation>
4379 <wsdl:operation name="SetBiometricData">
4380     <soap:operation soapAction="SetBiometricData"/>
4381     <wsdl:input name="SetBiometricData">
4382         <soap:body use="literal"/>
4383     </wsdl:input>
4384     <wsdl:output name="SetBiometricDataResponse">
4385         <soap:body use="literal"/>
4386     </wsdl:output>
4387     <wsdl:fault name="BIASFault">
4388         <soap:fault name="BIASFault" use="literal"/>
4389     </wsdl:fault>
4390 </wsdl:operation>
4391 <wsdl:operation name="TransformBiometricData">
4392     <soap:operation soapAction="TransformBiometricData"/>
4393     <wsdl:input name="TransformBiometricData">
4394         <soap:body use="literal"/>
4395     </wsdl:input>
4396     <wsdl:output name="TransformBiometricDataResponse">
4397         <soap:body use="literal"/>
4398     </wsdl:output>
4399     <wsdl:fault name="BIASFault">
4400         <soap:fault name="BIASFault" use="literal"/>
4401     </wsdl:fault>
4402 </wsdl:operation>
4403 <wsdl:operation name="UpdateBiographicData">

```

```

4404     <soap:operation soapAction="UpdateBiographicData"/>
4405     <wsdl:input name="UpdateBiographicData">
4406         <soap:body use="literal"/>
4407     </wsdl:input>
4408     <wsdl:output name="UpdateBiographicDataResponse">
4409         <soap:body use="literal"/>
4410     </wsdl:output>
4411     <wsdl:fault name="BIASFault">
4412         <soap:fault name="BIASFault" use="literal"/>
4413     </wsdl:fault>
4414 </wsdl:operation>
4415 <wsdl:operation name="UpdateBiometricData">
4416     <soap:operation soapAction="UpdateBiometricData"/>
4417     <wsdl:input name="UpdateBiometricData">
4418         <soap:body use="literal"/>
4419     </wsdl:input>
4420     <wsdl:output name="UpdateBiometricDataResponse">
4421         <soap:body use="literal"/>
4422     </wsdl:output>
4423     <wsdl:fault name="BIASFault">
4424         <soap:fault name="BIASFault" use="literal"/>
4425     </wsdl:fault>
4426 </wsdl:operation>
4427 <wsdl:operation name="VerifySubject">
4428     <soap:operation soapAction="VerifySubject"/>
4429     <wsdl:input name="VerifySubject">
4430         <soap:body use="literal"/>
4431     </wsdl:input>
4432     <wsdl:output name="VerifySubjectResponse">
4433         <soap:body use="literal"/>
4434     </wsdl:output>
4435     <wsdl:fault name="BIASFault">
4436         <soap:fault name="BIASFault" use="literal"/>
4437     </wsdl:fault>
4438 </wsdl:operation>
4439 <wsdl:operation name="Enroll">
4440     <soap:operation soapAction="Enroll"/>
4441     <wsdl:input name="Enroll">
4442         <soap:body use="literal"/>
4443     </wsdl:input>
4444     <wsdl:output name="EnrollResponse">
4445         <soap:body use="literal"/>

```

```
4446     </wsdl:output>
4447     <wsdl:fault name="BIASFault">
4448         <soap:fault name="BIASFault" use="literal"/>
4449     </wsdl:fault>
4450 </wsdl:operation>
4451 <wsdl:operation name="GetEnrollResults">
4452     <soap:operation soapAction="GetEnrollResults"/>
4453     <wsdl:input name="GetEnrollResults">
4454         <soap:body use="literal"/>
4455     </wsdl:input>
4456     <wsdl:output name="GetEnrollResultsResponse">
4457         <soap:body use="literal"/>
4458     </wsdl:output>
4459     <wsdl:fault name="BIASFault">
4460         <soap:fault name="BIASFault" use="literal"/>
4461     </wsdl:fault>
4462 </wsdl:operation>
4463 <wsdl:operation name="GetIdentifyResults">
4464     <soap:operation soapAction="GetIdentifyResults"/>
4465     <wsdl:input name="GetIdentifyResults">
4466         <soap:body use="literal"/>
4467     </wsdl:input>
4468     <wsdl:output name="GetIdentifyResultsResponse">
4469         <soap:body use="literal"/>
4470     </wsdl:output>
4471     <wsdl:fault name="BIASFault">
4472         <soap:fault name="BIASFault" use="literal"/>
4473     </wsdl:fault>
4474 </wsdl:operation>
4475 <wsdl:operation name="GetVerifyResults">
4476     <soap:operation soapAction="GetVerifyResults"/>
4477     <wsdl:input name="GetVerifyResults">
4478         <soap:body use="literal"/>
4479     </wsdl:input>
4480     <wsdl:output name="GetVerifyResultsResponse">
4481         <soap:body use="literal"/>
4482     </wsdl:output>
4483     <wsdl:fault name="BIASFault">
4484         <soap:fault name="BIASFault" use="literal"/>
4485     </wsdl:fault>
4486 </wsdl:operation>
4487 <wsdl:operation name="Identify">
```

```

4488     <soap:operation soapAction="Identify"/>
4489     <wsdl:input name="Identify">
4490         <soap:body use="literal"/>
4491     </wsdl:input>
4492     <wsdl:output name="IdentifyResponse">
4493         <soap:body use="literal"/>
4494     </wsdl:output>
4495     <wsdl:fault name="BIASFault">
4496         <soap:fault name="BIASFault" use="literal"/>
4497     </wsdl:fault>
4498 </wsdl:operation>
4499 <wsdl:operation name="RetrieveInformation">
4500     <soap:operation soapAction="RetrieveInformation"/>
4501     <wsdl:input name="RetrieveInformation">
4502         <soap:body use="literal"/>
4503     </wsdl:input>
4504     <wsdl:output name="RetrieveInformationResponse">
4505         <soap:body use="literal"/>
4506     </wsdl:output>
4507     <wsdl:fault name="BIASFault">
4508         <soap:fault name="BIASFault" use="literal"/>
4509     </wsdl:fault>
4510 </wsdl:operation>
4511 <wsdl:operation name="Verify">
4512     <soap:operation soapAction="Verify"/>
4513     <wsdl:input name="Verify">
4514         <soap:body use="literal"/>
4515     </wsdl:input>
4516     <wsdl:output name="VerifyResponse">
4517         <soap:body use="literal"/>
4518     </wsdl:output>
4519     <wsdl:fault name="BIASFault">
4520         <soap:fault name="BIASFault" use="literal"/>
4521     </wsdl:fault>
4522 </wsdl:operation>
4523 </wsdl:binding>
4524 <wsdl:service name="BIAS_v1">
4525     <wsdl:port name="BIAS_v1HttpBindingPort" binding="tns:BIAS_v1HttpBinding">
4526         <soap:address location="http://your_URI/IRI_to_your_BIAS_implementation"/>
4527     </wsdl:port>
4528 </wsdl:service>
4529 </wsdl:definitions>

```

---

## 4530 **Annex B. BIAS Patron format specification**

4531 The BIAS SOAP Profile defines an XML CBEFF Patron Format based on, but tailored from, Clause 13/15  
4532 of ISO/IEC 19785-3 [CBEFF3] as specified below.

4533

### 4534 **B.1 Patron**

4535 Organization for the Advancement of Structured Information Standards (OASIS)

4536

### 4537 **B.2 Patron identifier**

4538 `82 (0052 Hex)`.

4539

4540 This has been allocated by the Registration Authority for ISO/IEC 19785-2.

4541

### 4542 **B.3 Patron format name**

4543 `OASIS BIAS CBEFF XML Patron Format`

4544

### 4545 **B.4 Patron format identifier**

4546 `01 (0001 Hex)`.

4547

4548 This has been registered in accordance with ISO/IEC 19785-2.

4549

### 4550 **B.5 ASN.1 object identifier for this patron format**

4551 `No ASN.1 object identifiers are assigned to this patron format`

4552

### 4553 **B.6 Domain of use**

4554 This clause specifies a patron format based on XML that is designed to be friendly with code generation  
4555 tools. It defines a CBEFF structure that allows for the creation of simple, complex, and multi-modal BIRS  
4556 for use within BIAS transactions.

4557

### 4558 **B.7 Version identifier**

4559 This patron format specification has a version identifier of (major 1, minor 0).

4560

### 4561 **B.8 CBEFF version**

4562 This specification conforms to CBEFF version (major 2, minor 0).

4563

4564 **B.9 General**

4565 **B.9.1** This patron format is based on W3C XML 1.0. It supports all the mandatory and optional data  
4566 elements specified in ISO/IEC 19785-1. It can support either a simple BIR or a complex BIR structure  
4567 where each intermediate node or leaf of the structure is itself a BIR (called a "child BIR").

4568 **B.9.2** Most fields in this patron format are optional. Some mandatory and optional fields are  
4569 represented by XML elements, others are represented by attributes of XML elements. The presence of  
4570 an optional field in a BIR is signaled by simply including the corresponding element or attribute, and its  
4571 absence is signaled by simply omitting the corresponding element or attribute.

4572 **B.9.3** Special encodings are specified for integers (see B.17), octet strings (see B.18), and date and  
4573 time-of-the-day abstract values (see B.19).

4574 **B.9.4** An instance of a BIR or child BIR contains either a BDB or one or more BIR children, but never  
4575 contains both.

4576 **B.9.5** An extension mechanism is specified, which enables the inclusion of application-specific data (not  
4577 standardized) within a BIR or child BIR (see B.11.1.6).

4578

4579 **B.10 Specification**

4580 **B.10.1** In the rest of this clause, the terms "element" and "attribute" are used with the meaning of "XML  
4581 element" and "XML attribute", respectively.

4582 **B.10.2** The namespace with the name "`http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/`"  
4583 is called the patron format namespace of this patron format.

4584 **B.10.3** All elements defined in this patron format have the patron format namespace name. All attribute  
4585 names are unqualified.

4586 **B.10.4** An instance of a BIR shall be represented as a `<BIR>` element (see B.11).

4587 **B.10.5** The `<BIR>` element may be the root of an XML document, but this is not required.

4588 **B.10.6** The portion of the XML document consisting of the `<BIR>` element and its whole content shall be  
4589 valid according to the XML schema provided in B.22.

4590 NOTE 1 – Validity according to that XML schema does not imply that the `<BIR>` element  
4591 satisfies all the requirements in the normative text of this specification, as there are some  
4592 requirements that cannot be (or are not) formally expressed in the XML schema.

4593 NOTE 2 – When the `<BIR>` element is the root of an XML document, the UTF-8  
4594 character encoding is recommended for the XML document, because it will usually  
4595 produce a smaller encoding.

4596 **B.10.7** The abstract value NO VALUE AVAILABLE, for any CBEFF data element that supports this  
4597 abstract value, shall be encoded as the omission of the corresponding element or attribute both in the  
4598 `<BIR>` element and in all of its ancestor `<BIR>` elements.

4599 NOTE – The inheritance mechanism specified in B.14.2.1, B.15.2.1 and B.16.2.1 causes  
4600 a data element of a BIR to inherit an abstract value (different from NO VALUE  
4601 AVAILABLE) from its closest ancestor `<BIR>` element that contains that element or  
4602 attribute when the `<BIR>` element in question does not contain it. If any `<BIR>` element  
4603 in a hierarchy of `<BIR>` elements specifies an abstract value for a given data element,  
4604 that abstract value can be overridden by a different abstract value in any of its  
4605 descendant `<BIR>` elements, but the overriding abstract value can never be NO VALUE  
4606 AVAILABLE.

## 4607 **B.11 Element <BIR>**

### 4608 **B.11.1 Syntax**

4609 **B.11.1.1** This element shall have no attributes, and shall have a content consisting of the following  
4610 (in order):

- 4611 a) an optional <Version> element (see B.12);
- 4612 b) an optional <CBEFFVersion> element (see B.13);
- 4613 c) zero or more application-specific elements;
- 4614 d) a mandatory <BIRInfo> element (see B.14);
- 4615 e) an optional <BDBInfo> element (see B.15);
- 4616 f) an optional <SBInfo> element (see B.16);
- 4617 g) zero or more <BIR> elements (see B.11);
- 4618 h) either an optional <BDB> element that shall contain a valid representation of an octet string (see  
4619 B.18), or an optional <bdbx> element that shall contain a valid XML string;
- 4620 i) an optional <SB> element – the content of this element shall be a valid representation of an octet  
4621 string.

4622 **B.11.1.2** The <BDB> or <bdbx> element shall not be present if one or more child <BIR> elements  
4623 are present, and shall be present if no child <BIR> elements are present.

4624 **B.11.1.3** The <SB> element shall be absent unless its presence is required by F.14.2.2 or  
4625 permitted by F.15.2.3.

4626 **B.11.1.4** If the <BDB> or <bdbx> element is present, then the <BDBInfo> element shall also be  
4627 present.

4628 **B.11.1.5** If the <SB> element is present, then the <SBInfo> element shall also be present.

4629 **B.11.1.6** The number of application-specific elements and their name, namespace name,  
4630 attributes, and content are not defined in this patron format specification. However, the namespace name  
4631 of those elements shall be different from the patron format namespace name (see B.10.2).

4632

### 4633 **B.11.2 Semantics**

4634 **B.11.2.1** This element is either a complex or a simple BIR, depending on which child elements are  
4635 present. If a child <BDB> or <bdbx> element is present, this element is a simple BIR. If one or more  
4636 child <BIR> elements are present, this element is a complex BIR.

4637 **B.11.2.2** The elements <Version>, <CBEFFVersion>, <BIRInfo>, <BDBInfo>, and  
4638 <SBInfo> and their content form the standard biometric header of the BIR.

4639 **B.11.2.3** The <Version> element (if present) carries the major and minor version number of this  
4640 patron format.

4641 **B.11.2.4** The <CBEFFVersion> element (if present) carries the major and minor version number  
4642 of the CBEFF standard.

4643 **B.11.2.5** Each <BIR> element is a whole BIR (of the same patron format) that is a child BIR of the  
4644 BIR.

4645 **B.11.2.6** The <BDB> or <bdbX> element (if present) carries the biometric data block (BDB) of the  
4646 BIR.

4647 NOTE – A <BDB> or <bdbX> element and a <BIR> element cannot coexist as children of  
4648 the same <BIR> element (see B.11.1.2).

4649 **B.11.2.7** The <SB> element (if present) carries the security block (SB) of the BIR.

4650 NOTE – A <SB> element can coexist with either a <BIR> element or a <BDB> or  
4651 <bdbX> element that is a child of the same <BIR> element.

4652 **B.11.2.8** The <BIRInfo> element carries information about both the BIR and (possibly) about its  
4653 descendant BIRs (if the <BIR> element has one or more child <BIR> elements), as specified in B.14.2.1.

4654 **B.11.2.9** The <BDBInfo> element (if present) carries information about either the BDB of the BIR  
4655 (if the <BIR> element has a child <BDB> or <bdbX> element) or about the BDBs of the descendant BIRs  
4656 that have a child <BDB> or <bdbX> element (if the <BIR> element has one or more child <BIR>  
4657 elements), as specified in B.15.2.1.

4658 **B.11.2.10** The <SBInfo> element (if present) carries information about either the SB of the BIR (if  
4659 the <BIR> element has a child <SB> element) or about the SBs of the descendant BIRs that have a child  
4660 <SB> element (if the <BIR> element has one or more child <BIR> elements but no child <SB> element),  
4661 as specified in B.16.2.1.

4662

## 4663 **B.12 Element <Version>**

### 4664 **B.12.1 Syntax**

4665 This element shall have contents consisting of the following (in order):

- 4666 a) a required <Major> element – the value of this element shall be a valid representation of a non-  
4667 negative integer.
- 4668 b) a required <Minor> element – the value of this element shall be a valid representation of a non-  
4669 negative integer.

### 4670 **B.12.2 Semantics**

4671 **B.12.2.1** This element represents the data element CBEFF\_patron\_header\_version, and carries  
4672 the (major and minor) version number of the patron format. The number assigned to this version of the  
4673 patron format is major 1, minor 0.

4674 **B.12.2.2** The <Major> element represents the major version number (1 in this version).

4675 **B.12.2.3** The <Minor> element represents the minor version number (0 in this version).

4676 **B.12.2.4** If this element is not present, the values Major="1" Minor="0" are implied.

4677 **B.12.2.5** A child <BIR> element shall have the same (major and minor) version number as its  
4678 parent <BIR> element.

4679 NOTE – This implies that the <Version> element, if present in a child <BIR> element,  
4680 has to carry the same values as the <Version> element in the parent <BIR> element.  
4681 This is equivalent to omitting the <Version> element. Therefore, this element is  
4682 normally omitted in child <BIR> elements.

4683



## 4684 **B.13 Element <CBEFFVersion>**

### 4685 **B.13.1 Syntax**

4686 This element shall have content consisting of the following (in order):

- 4687 a) a required **<Major>** element – the value of this element shall be a valid representation of a non-  
4688 negative integer (see B.17);
- 4689 b) a required **<Minor>** element – the value of this element shall be a valid representation of a non-  
4690 negative integer.

### 4691 **B.13.2 Semantics**

4692 **B.13.2.1** This element represents the data element CBEFF\_version, and carries the version  
4693 number of the CBEFF standard supported by this patron format. The number assigned to the version of  
4694 CBEFF supported by this patron format is Major=2, Minor=0.

4695 **B.13.2.2** The **<Major>** element represents the major version number (2 in this version).

4696 **B.13.2.3** The **<Minor>** element represents the minor version number (0 in this version).

4697 **B.13.2.4** If this element is not present, the values **Major="2"** **Minor="0"** are implied.

4698 **B.13.2.5** A child **<BIR>** element shall have the same CBEFF version number (major and minor) as  
4699 its parent **<BIR>** element.

4700 NOTE – Thus, the **<CBEFFVersion>** element is normally omitted from all child **<BIR>**  
4701 elements, as it would be redundant.

4702

## 4703 **B.14 Element <BIRInfo>**

### 4704 **B.14.1 Syntax**

4705 **B.14.1.1** This element shall have a content consisting of the following (in order):

- 4706 a) an optional **<Creator>** element – the content of this element shall be a string of ISO/IEC 10646  
4707 characters;
- 4708 b) an optional **<Index>** element – the content of this element shall be a valid representation of a  
4709 universally unique identifier (see B.20), and shall not inherit its value from any other level BIR;
- 4710 c) an optional **<Payload>** element – the content of this element shall be a valid representation of an  
4711 octet string, and shall not inherit its value from any other level BIR.
- 4712 d) a required **<Integrity>** element – the value of this element shall be one of the character strings in  
4713 the third cell of the corresponding row of Table B.1;
- 4714 e) an optional **<CreationDate>** element – the value of this element shall be a valid representation of a  
4715 date and time of the day (see B.19);
- 4716 f) an optional **<NotValidBefore>** element – the value of this element shall be a valid representation  
4717 of a date and time of the day;
- 4718 g) an optional **<NotValidAfter>** element – the value of this element shall be a valid representation of  
4719 a date and time of the day.

4720 **B.14.2 Semantics**

4721 **B.14.2.1** The <BIRInfo> element carries information about the BIR. In addition, if the BIR has  
 4722 one or more child BIRs (the <BIR> element has one or more child <BIR> elements), the information  
 4723 carried by the attributes and child elements of the <BIRInfo> element is inherited by those child BIRs  
 4724 except where overridden by a corresponding attribute or child element of the <BIRInfo> element of a  
 4725 child BIR. The information inherited by a BIR applies to that BIR, and (if the BIR has itself child BIRs) is  
 4726 further inherited by its child BIRs in the same way (and so on recursively).

4727 NOTE – Since the Integrity element is required and the <BIRInfo> element is  
 4728 mandatory in all <BIR> elements, inheritance of the Integrity element can never  
 4729 occur.

4730 **B.14.2.2** The Integrity element indicates whether integrity information about this BIR is  
 4731 provided within the security block (SB) of the BIR (the child <SB> element of the parent <BIR> element of  
 4732 this <BIRInfo> element).

4733 NOTE – This information may consist of a digital signature or MAC, a reference to a key  
 4734 or certificate, an encrypted key (with or without a reference to the key used to encrypt  
 4735 that key), or other parameters of the digital signing (or MAC) process.

4736 **B.14.2.3** If the value of the <Integrity> element is "true", then the parent <BIR> element of  
 4737 this <BIRInfo> element shall have a child <SB> element.

4738 **B.14.2.4** Table B.1 specifies the correspondence between the attributes and child elements of this  
 4739 element and CBEFF data elements, and specifies the supported abstract values and their encodings (see  
 4740 also B.10.7).

4741 NOTE - This element represents all CBEFF data elements whose name begins with  
 4742 "CBEFF\_BIR\_".

4743 **Table B.1 – BIR information**

| CBEFF data element name     | XML element | Supported abstract values and encodings   | Reference |
|-----------------------------|-------------|---|-----------|
| CBEFF_BIR_creator           | <Creator>   | All ISO/IEC 10646 character strings are supported.<br>The character string shall be encoded as the string itself.                                     |           |
| CBEFF_BIR_index             | <Index>     | All well-formed UUIDs are supported.<br>The UUIDs shall be encoded as specified in B.20.<br>Shall not inherit its value from any other BIR level.     |           |
| CBEFF_BIR_payload           | <Payload>   | All octet strings are supported.<br>The octet strings shall be encoded as specified in B.18.<br>Shall not inherit its value from any other BIR level. |           |
| CBEFF_BIR_integrity_options | <Integrity> | The following abstract values are supported.<br>The abstract values shall be encoded as shown below.<br>NO INTEGRITY:                                 |           |

| CBEFF data element name               | XML element      | Supported abstract values and encodings  | Reference |
|---------------------------------------|------------------|--|-----------|
|                                       |                  | "false"<br>INTEGRITY:<br>"true"  |           |
| CBEFF_BIR_creation_date               | <CreationDate>   | All date and time-of-the-day abstract values permitted by CBEFF are supported.<br>The abstract values shall be encoded as specified in B.19. |           |
| CBEFF_BIR_validity_period (lower end) | <NotValidBefore> | All date and time-of-the-day abstract values permitted by CBEFF are supported.<br>The abstract values shall be encoded as specified in B.19. |           |
| CBEFF_BIR_validity_period (upper end) | <NotValidAfter>  | All date and time-of-the-day abstract values permitted by CBEFF are supported.<br>The abstract values shall be encoded as specified in B.19. |           |

4744

## 4745 B.15 Element <BDBInfo>

### 4746 B.15.1 Syntax

4747 **B.15.1.1** This element shall have a content consisting of the following (in order):

- 4748 a) an optional <ChallengeResponse> element – the content of this element shall be a valid  
4749 representation of an octet string (see B.18);
- 4750 b) an optional <Index> element – the content of this element shall be a valid representation of a  
4751 universally unique identifier (see B.20).
- 4752 c) an optional <FormatOwner> element – the value of this element shall be a valid representation  
4753 of an integer in the range 1 to 65535 (see B.17);
- 4754 d) an optional <FormatType> element – the value of this element shall be a valid representation of  
4755 an integer in the range 1 to 65535;
- 4756 e) an optional <Encryption> element – the value of this element shall be one of the character  
4757 strings in the third cell of the corresponding row of Table B.2;
- 4758 f) an optional <CreationDate> element – the value of this element shall be a valid  
4759 representation of a date and time of the day (see B.19);
- 4760 g) an optional <NotValidBefore> element – the value of this element shall be a valid  
4761 representation of a date and time of the day;
- 4762 h) an optional <NotValidAfter> element – the value of this element shall be a valid  
4763 representation of a date and time of the day;

- 4764 i) an optional **<Type>** element – the value of this element shall be one of the character strings in  
4765 the third cell of the corresponding row of Table B.2;
- 4766 j) an optional **<Subtype>** element – the value of this element shall be one of the character strings  
4767 in the third cell of the corresponding row of Table B.2;
- 4768 k) an optional **<Level1>** element – the value of this element shall be one of the character strings in  
4769 the third cell of the corresponding row of Table B.2;
- 4770 l) an optional **<ProductOwner>** element – the value of this element shall be a valid  
4771 representation of an integer in the range 1..65535 (see B.17);
- 4772 m) an optional **<ProductType>** element – the value of this element shall be a valid representation  
4773 of an integer in the range 1..65535;
- 4774 n) an optional **<CaptureDeviceOwner>** element – the value of this element shall be a valid  
4775 representation of an integer in the range 1..65535 (see B.17);
- 4776 o) an optional **<CaptureDeviceType>** element – the value of this element shall be a valid  
4777 representation of an integer in the range 1..65535;
- 4778 p) an optional **<FeatureExtractionAlgorithmOwner>** element – the value of this element  
4779 shall be a valid representation of an integer in the range 1..65535 (see B.17);
- 4780 q) an optional **<FeatureExtractionAlgorithmType>** element – the value of this element shall  
4781 be a valid representation of an integer in the range 1..65535;
- 4782 r) an optional **<ComparisonAlgorithmOwner>** element – the value of this element shall be a  
4783 valid representation of an integer in the range 1..65535 (see B.17);
- 4784 s) an optional **<ComparisonAlgorithmType>** element – the value of this element shall be a valid  
4785 representation of an integer in the range 1..65535;
- 4786 t) an optional **<QualityAlgorithmOwner>** element – the value of this element shall be a valid  
4787 representation of an integer in the range 1..65535 (see B.17);
- 4788 u) an optional **<QualityAlgorithmType>** element – the value of this element shall be a valid  
4789 representation of an integer in the range 1..65535;
- 4790 v) an optional **<CompressionAlgorithmOwner>** element – the value of this element shall be a  
4791 valid representation of an integer in the range 1..65535 (see B.17);
- 4792 w) an optional **<CompressionAlgorithmType>** element – the value of this element shall be a  
4793 valid representation of an integer in the range 1..65535;
- 4794 x) an optional **<Purpose>** element – the value of this element shall be one of the character strings  
4795 in the third cell of the corresponding row of Table B.2;
- 4796 y) an optional **<Quality>** element – the value of this element shall be a valid representation of an  
4797 integer in the range –2..100 (see B.17), as specified in the third cell of the corresponding row of  
4798 Table B.2.

4799 **B.15.1.3** If the parent **<BIR>** element has a child **<BDB>** element, then the **<Encryption>**  
4800 element shall be present in this **<BDBInfo>** element unless it is present in the child **<BDBInfo>** element  
4801 of an ancestor **<BIR>** element (see also B.11.1.4).

4802 **B.15.1.4** If the parent <BIR> element has a child <BDB> element, then the <FormatOwner>  
 4803 element shall be present in this <BDBInfo> element unless it is present in the child <BDBInfo> element  
 4804 of an ancestor <BIR> element (see also B.11.1.4).

4805 **B.15.1.5** If the parent <BIR> element has a child <BDB> element, then the <FormatType>  
 4806 element shall be present in this <BDBInfo> element unless it is present in the child <BDBInfo> element  
 4807 of an ancestor <BIR> element (see also B.11.1.4).

4808 NOTE – The ancestor <BIR> elements mentioned in the last three subclauses above  
 4809 need not be the same.

4810

4811 **B.15.2 Semantics**

4812 **B.15.2.1** If the BIR has a BDB (the <BIR> element has a child <BDB> element), then the  
 4813 <BDBInfo> element carries information about that BDB. Otherwise, the information carried by the  
 4814 attributes and child elements of the <BDBInfo> element is inherited by all the BIRs that are children of  
 4815 the BIR except where overridden by a corresponding attribute or child element of the <BDBInfo>  
 4816 element of a child BIR. The information inherited by a BIR with a BDB applies to that BDB, whereas the  
 4817 information inherited by a BIR that has itself child BIRs is further inherited by all the BIRs that are children  
 4818 of the BIR in the same way (and so on recursively).

4819 **B.15.2.2** If the BIR has a BDB and encryption is applied to that BDB (either by including the  
 4820 **encryption** attribute with the value "true" in the <BDBInfo> element or by having the BIR inherit that  
 4821 attribute value from its parent BIR), then the BDB in the <BDB> element shall be encrypted.

4822 **B.15.2.3** If the BDB of a BIR is encrypted, information about the encryption process may be  
 4823 provided within the security block (SB) of that BIR (the child <SB> element of the parent <BIR> element  
 4824 of this <BIRInfo> element).

4825 NOTE – This information may consist of a reference to an encryption key, an encrypted  
 4826 key (with or without a reference to the key used to encrypt that key), or other parameters  
 4827 of the encryption process.

4828 **B.15.2.4** Table B.2 specifies the correspondence between the attributes and child elements of this  
 4829 element and CBEFF data elements, and specifies the supported abstract values and their encodings (see  
 4830 also F.10.7).

4831 NOTE – This element represents all CBEFF data elements whose name begins with  
 4832 "CBEFF\_BDB\_".

4833

**Table B.2 – BDB information**

| CBEFF data element name      | XML element   | Supported abstract values and encodings  | Reference |
|------------------------------|---------------|--|-----------|
| CBEFF_BDB_format_owner       | <FormatOwner> | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17.             |           |
| CBEFF_BDB_format_type        | <FormatType>  | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17.             |           |
| CBEFF_BDB_encryption_options | <Encryption>  | The following abstract values are supported.<br>The abstract values shall be encoded as shown below.<br>NO ENCRYPTION: |           |

| CBEFF data element name               | XML element         | Supported abstract values and encodings   | Reference |
|---------------------------------------|---------------------|---|-----------|
|                                       |                     | <p>"false"</p> <p>ENCRYPTION:</p> <p>"true"</p>   |           |
| CBEFF_BDB_creation_date               | <CreationDate>      | <p>All date and time-of-the-day abstract values permitted by CBEFF are supported.</p> <p>The abstract values shall be encoded as specified in B.19.</p>   |           |
| CBEFF_BDB_validity_period (lower end) | <NotValidBefore>    | <p>All date and time-of-the-day abstract values permitted by CBEFF are supported.</p> <p>The abstract values shall be encoded as specified in B.19.</p>   |           |
| CBEFF_BDB_challenge_response          | <ChallengeResponse> | <p>All octet strings are supported.</p> <p>The octet strings shall be encoded as specified in B.18.</p> <p>Shall appear only in BIRs that have a BDB.</p>   |           |
| CBEFF_BDB_index                       | <Index>             | <p>All well-formed UUIDs are supported.</p> <p>The UUIDs shall be encoded as specified in B.20</p> <p>Shall appear only in BIRs that have a BDB.</p>  |           |
| CBEFF_BDB_validity_period (upper end) | <NotValidAfter>     | <p>All date and time-of-the-day abstract values permitted by CBEFF are supported.</p> <p>The abstract values shall be encoded as specified in B.19.</p>   |           |
| CBEFF_BDB_biometric_type              | <Type>              | <p>The following abstract values and all their unordered combinations are supported.</p> <p>A single abstract value shall be encoded as the corresponding string shown below. A combination of two or more abstract values shall be encoded as the concatenation of the corresponding strings, using a single space as separator.</p> <p>SCENT:<br/>"Scent"</p> <p>DNA:<br/>"DNA"</p> <p>EAR:<br/>"Ear"</p> <p>FACE:<br/>"Face"</p> <p>FINGER:<br/>"Finger"</p> <p>FOOT:<br/>"Foot"</p> |           |

| CBEFF data element name     | XML element | Supported abstract values and encodings  | Reference |
|-----------------------------|-------------|--|-----------|
|                             |             | VEIN:<br><b>"Vein"</b><br>HAND GEOMETRY:<br><b>"HandGeometry"</b><br>IRIS:<br><b>"Iris"</b><br>RETINA:<br><b>"Retina"</b><br>VOICE:<br><b>"Voice"</b><br>GAIT:<br><b>"Gait"</b><br>KEYSTROKE:<br><b>"Keystroke"</b><br>LIP MOVEMENT:<br><b>"LipMovement"</b><br>SIGNATURE OR SIGN:<br><b>"SignatureSign"</b>   |           |
| CBEFF_BDB_biometric_subtype | <Subtype>   | The following abstract values are supported.<br>The abstract values shall be encoded as shown below. A combination of two or more abstract values shall be encoded as the concatenation of the corresponding strings, using a single space as separator.<br>LEFT:<br><b>"Left"</b><br>RIGHT:<br><b>"Right"</b><br>THUMB:<br><b>"Thumb"</b><br>INDEX FINGER:<br><b>"IndexFinger"</b><br>MIDDLE FINGER:<br><b>"MiddleFinger"</b><br>RING FINGER:<br><b>"RingFinger"</b><br>LITTLE FINGER:<br><b>"LittleFinger"</b> |           |
| CBEFF_BDB_processed_level   | <Level>     | The following abstract values are supported.<br>The abstract values shall be encoded as shown below.<br>RAW:<br><b>"Raw"</b><br>INTERMEDIATE:<br><b>"Intermediate"</b><br>PROCESSED:   |           |

| CBEFF data element name                      | XML element                       | Supported abstract values and encodings  | Reference |
|--|-----------------------------------|--|-----------|
|  |                                   | "Processed"  |           |
| CBEFF_BDB_product_owner                      | <ProductOwner>                    | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_product_type                       | <ProductType>                     | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_capture_device_owner               | <CaptureDeviceOwner>              | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_capture_device_type                | <CaptureDeviceType>               | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_feature_extraction_algorithm_owner | <FeatureExtractionAlgorithmOwner> | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_feature_extraction_algorithm_type  | <FeatureExtractionAlgorithmType>  | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_comparison_algorithm_owner         | <ComparisonAlgorithmOwner>        | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_comparison_algorithm_type          | <ComparisonAlgorithmType>         | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_quality_algorithm_owner            | <QualityAlgorithmOwner>           | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_quality_algorithm_type             | <QualityAlgorithmType>            | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_compression_algorithm_owner        | <CompressionAlgorithmOwner>       | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_BDB_compression_algorithm_type         | <CompressionAlgorithmType>        | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |



| CBEFF data element name | XML element | Supported abstract values and encodings   | Reference |
|-------------------------|-------------|---|-----------|
| CBEFF_BDB_purpose       | <Purpose>   | <p>The following abstract values are supported.</p> <p>The abstract values shall be encoded as shown below.</p> <p>VERIFY:<br/>"Verify"</p> <p>IDENTIFY:<br/>"Identify"</p> <p>ENROLL:<br/>"Enroll"</p> <p>ENROLL FOR VERIFICATION ONLY:<br/>"EnrollVerify"</p> <p>ENROLL FOR IDENTIFICATION ONLY:<br/>"EnrollIdentify"</p> <p>AUDIT:<br/>"Audit"</p> |           |
| CBEFF_BDB_quality       | <Quality>   | <p>The following abstract values are supported.</p> <p>The integers shall be encoded as specified in B.17. The other abstract values shall be encoded as shown below.</p> <p>INTEGER</p> <p>QUALITY NOT SUPPORTED BY BDB CREATOR:<br/>"-2"</p> <p>QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET:<br/>"-1"</p>  |           |

4834

## 4835 B.16 Element <SBInfo>

### 4836 B.16.1 Syntax

4837 **B.16.1.1** This element shall have content consisting of the following (in order):

4838 a) an optional <FormatOwner> element – the value of this element shall be a valid representation of an  
4839 integer in the range 1 to 65535 (see B.17);

4840 b) an optional <FormatType> element – the value of this element shall be a valid representation of an  
4841 integer in the range 1 to 65535

4842 **B.16.1.2** If the parent <BIR> element has a child <SB> element, then the <FormatOwner>  
4843 element shall be present in this <SBInfo> element unless it is present in the child <SBInfo> element of  
4844 an ancestor <BIR> element (see also B.11.1.5).

4845 **B.16.1.3** If the parent <BIR> element has a child <SB> element, then the <FormatType> element  
4846 shall be present in this <SBInfo> element unless it is present in the child <SBInfo> element of an  
4847 ancestor <BIR> element (see also B.11.1.5).

4848 NOTE 1 – The ancestor <BIR> elements mentioned in the last two subclauses above  
4849 need not be the same.

4850 NOTE 2 – When the parent <BIR> element has a child <SB> element and one omits both  
4851 children of the <SBInfo> element, the <SBInfo> element will have no attributes and an  
4852 empty content. Omission of the <SBInfo> element is not allowed in this case (see  
4853 B.11.1.5).

4854

## 4855 B.16.2 Semantics

4856 **B.16.2.1** If the BIR has an SB (the <BIR> element has a child <SB> element), then the <SBInfo>  
4857 element carries information about that SB. In addition, if the BIR has one or more child BIRs (the <BIR>  
4858 element has one or more child <BIR> elements), the information carried by the child element of the  
4859 <SBInfo> element is inherited by those child BIRs except where overridden by a corresponding child  
4860 element of the <SBInfo> element of a child BIR. The information inherited by a BIR with an SB applies  
4861 to that SB, and (if the BIR has itself child BIRs) is further inherited by its child BIRs in the same way (and  
4862 so on recursively).

4863 **B.16.2.2** Table B.3 specifies the correspondence between the attributes and child elements of this  
4864 element and CBEFF data elements, and specifies the supported abstract values and their encodings (see  
4865 also B.10.7).

4866 NOTE – This element represents all CBEFF data elements whose name begins with  
4867 "CBEFF\_SB\_".

4868

4869

**Table B.3 – SB information**

| CBEFF data element name | XML element   | Supported abstract values and encodings  | Reference |
|-------------------------|---------------|--|-----------|
| CBEFF_SB_format_owner   | <FormatOwner> | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |
| CBEFF_SB_format_type    | <FormatType>  | All integers in the range 1 to 65535 are supported.<br>The integers shall be encoded as specified in B.17. |           |

## 4870 B.17 Representation of Integers

4871 **B.17.1** A non-negative integer shall be represented as a string of one or more ISO/IEC 10646 characters  
4872 in the range DIGIT ZERO to DIGIT NINE ("0" to "9") in decimal notation.

4873 **B.17.2** A negative integer shall be represented as the corresponding positive integer, preceded by a  
4874 HYPHEN-MINUS character ("-").

4875 **B.17.3** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the  
4876 encoding.

4877

## 4878 B.18 Representation of Octet Strings

4879 **B.18.1** An octet string shall be represented as a string of the following ISO/IEC 10646 characters:

4880 a) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER Z;

4881 b) LATIN SMALL LETTER A to LATIN SMALL LETTER Z;

4882 c) DIGIT ZERO to DIGIT NINE;

4883 d) PLUS SIGN;

4884 e) SOLIDUS;

4885 f) EQUALS SIGN.

4886 forming the Base64 encoding of the octet string (see IETF RFC 2045), with all whitespace removed.

4887 **B.18.2** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the  
4888 encoding.

4889

## 4890 **B.19 Representation of Date and Time of the Day**

4891 **B.19.1** A date and time of the day shall be represented as a string of ISO/IEC 10646 characters in the  
4892 following format, which conforms to ISO 8601.

4893 **B.19.2** The encoding shall be the concatenation of all the following components (in order):

4894 a) the "year" component, consisting of the year encoded in four digits ("2000" to "2999");

4895 b) the hyphen character "-";

4896 c) the "month" component, consisting of the month encoded in two digits ("01" to "12");

4897 d) the hyphen character "-";

4898 e) the "day" component, consisting of the day encoded in two digits ("01" to "31");

4899 f) the letter "T";

4900 g) the "hour" component, consisting of the hour encoded in two digits ("00" to "23");

4901 h) the colon character ":";

4902 i) the "minute" component, consisting of the minute encoded in two digits ("00" to "59");

4903 j) the colon character ":";

4904 k) the "second" component, consisting of the second encoded in two digits ("00" to "59");

4905 l) the letter "z".

4906 **B.19.3** The "year", "month", "day", "hour", "minute", and "second" components shall be present.

4907 **B.19.4** The letter "T" shall be present.

4908 **B.19.5** The letter "z" shall be present whether or not the "hour" component is present.

4909 NOTE This letter indicates that the date and time of the day are UTC.

4910 **B.19.6** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the  
4911 encoding.

4912

4913 **B.20 Representation of Universally Unique Identifiers**

4914 NOTE: The following subclauses describe the same representation of a UUID as is  
 4915 specified in ISO/IEC 9834-8, clause 8. An example of such a representation is: f81d4fae-  
 4916 7dec-11d0-a765-00a0c91e6bf6

4917 **B.20.1** A universally unique identifier (UUID) shall be represented as a string of ISO/IEC 10646  
 4918 characters. Each string shall contain exactly 36 characters from the union of the following sets:

- 4919 a) DIGIT ZERO to DIGIT NINE ("0" to "9"), each representing a hexadecimal digit 0 through 9;
- 4920 b) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER F ("A" to "F"), each representing a  
 4921 hexadecimal digit A through F;
- 4922 c) LATIN SMALL LETTER A to LATIN SMALL LETTER F ("a" to "f"), each representing a hexadecimal  
 4923 digit A through F; and
- 4924 d) HYPHEN-MINUS ("-").

4925 **B.20.2** Each of the positions 9, 14, 19, and 24 of an encoding shall contain a character from set (d).  
 4926 The other 32 positions shall contain characters from sets (a) through (c).

4927 **B.20.3** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the  
 4928 encoding.

4929

4930 **B.21 Patron format conformance statement**

4931 **B.21.1 Identifying information**

| Required Information                  | Patron format reference |
|---------------------------------------|-------------------------|
| Patron name                           | See B.1                 |
| Patron identifier                     | See B.2                 |
| Patron format name                    | See B.3                 |
| Patron format identifier              | See B.4                 |
| Patron format ASN.1 object identifier | See B.5                 |
| Domain of use description             | See B.6                 |
| Patron format version                 | See B.7                 |
| CBEFF version                         | See B.8                 |

4932

4933 **B.21.2 ISO/IEC 19785-1:2006/Amd 1:2010 to Patron Format Mapping**

| CBEFF data element name | Mandatory/<br>optional   | Patron format field name            | Abstract<br>values<br>specified? | Encodings<br>specified? |
|-------------------------|--|-------------------------------------|----------------------------------|-------------------------|
| CBEFF_BDB_format_owner  | Mandatory<br>(specified<br>or inherited)<br>if a BDB is<br>present | <FormatOwner> child of<br><BDBInfo> | Yes                              | Yes                     |
| CBEFF_BDB_format_type   | Mandatory<br>(specified)   | <FormatType> child of               | Yes                              | Yes                     |

| CBEFF data element name                      | Mandatory/ optional                                    | Patron format field name  | Abstract values specified? | Encodings specified? |
|--|--|---|----------------------------|----------------------|
|  | or inherited) if a BDB is present                      | <BDBInfo>   |                            |                      |
| CBEFF_BDB_encryption_options                 | Mandatory (specified or inherited) if a BDB is present | <Encryption> child of <BDBInfo>                                 | Yes                        | Yes                  |
| CBEFF_BIR_integrity_options                  | Mandatory  | <Integrity> child of <BIRInfo>                                  | Yes                        | Yes                  |
| CBEFF_BDB_subheader_count                    | Mandatory  | implied in the number of occurrences of the child <BIR> element | No (implied)               | No (implied)         |
| CBEFF_BDB_biometric_type                     | Optional   | <Type> child of <BDBInfo>                                       | Yes                        | Yes                  |
| CBEFF_BDB_biometric_subtype                  | Optional   | <Subtype> child of <BDBInfo>                                    | Yes                        | Yes                  |
| CBEFF_BDB_challenge_response                 | Optional   | <ChallengeResponse> child of <BDBInfo>                          | Yes                        | Yes                  |
| CBEFF_BDB_creation_date                      | Optional   | <CreationDate> child of <BDBInfo>                               | Yes                        | Yes                  |
| CBEFF_BDB_index                              | Optional   | <Index> child of <BDBInfo>                                      | Yes                        | Yes                  |
| CBEFF_BDB_product_owner                      | Optional   | <ProductOwner> child of <BDBInfo>                               | Yes                        | Yes                  |
| CBEFF_BDB_product_type                       | Optional   | <ProductType> child of <BDBInfo>                                | Yes                        | Yes                  |
| CBEFF_BDB_capture_device_owner               | Optional   | <CaptureDeviceOwner> child of <BDBInfo>                         | Yes                        | Yes                  |
| CBEFF_BDB_capture_device_type                | Optional   | <CaptureDeviceType> child of <BDBInfo>                          | Yes                        | Yes                  |
| CBEFF_BDB_feature_extraction_algorithm_owner | Optional   | <FeatureExtractionAlgorithmOwner> child of <BDBInfo>            | Yes                        | Yes                  |
| CBEFF_BDB_feature_extraction_algorithm_type  | Optional   | <FeatureExtractionAlgorithmType> child of <BDBInfo>             | Yes                        | Yes                  |
| CBEFF_BDB_comparison_algorithm_owner         | Optional   | <ComparisonAlgorithmOwner> child of <BDBInfo>                   | Yes                        | Yes                  |
| CBEFF_BDB_comparison_algorithm_type          | Optional   | <ComparisonAlgorithmType> child of <BDBInfo>                    | Yes                        | Yes                  |
| CBEFF_BDB_quality_algorithm_owner            | Optional   | <QualityAlgorithmOwner> child of <BDBInfo>                      | Yes                        | Yes                  |
| CBEFF_BDB_quality_algorithm_type             | Optional   | <QualityAlgorithmType> child of <BDBInfo>                       | Yes                        | Yes                  |
| CBEFF_BDB_compression_algorithm_owner        | Optional   | <CompressionAlgorithmOwner> child of <BDBInfo>                  | Yes                        | Yes                  |
| CBEFF_BDB_compression_algorithm_type         | Optional   | <CompressionAlgorithmType> child of <BDBInfo>                   | Yes                        | Yes                  |
| CBEFF_BDB_processed_level                    | Optional   | <Level> child of <BDBInfo>                                      | Yes                        | Yes                  |

| CBEFF data element name       | Mandatory/ optional | Patron format field name                                     | Abstract values specified? | Encodings specified? |
|-------------------------------|---------------------|--|----------------------------|----------------------|
| CBEFF_BDB_purpose             | Optional            | <Purpose> child of <BDBInfo>                                 | Yes                        | Yes                  |
| CBEFF_BDB_quality             | Optional            | <Quality> child of <BDBInfo>                                 | Yes                        | Yes                  |
| CBEFF_BDB_validity_period     | Optional            | <NotValidBefore> and <NotValidAfter> children of <BDBInfo>   | Yes                        | Yes                  |
| CBEFF_BIR_creation_date       | Optional            | <CreationDate> child of <BIRInfo>                            | Yes                        | Yes                  |
| CBEFF_BIR_creator             | Optional            | <Creator> child of <BIRInfo>                                 | Yes                        | Yes                  |
| CBEFF_BIR_index               | Optional            | <Index> child of <BDBInfo>                                   | Yes                        | Yes                  |
| CBEFF_BIR_patron_format_owner | N/A                 |  | No                         | No                   |
| CBEFF_BIR_patron_format_type  | N/A                 |  | No                         | No                   |
| CBEFF_BIR_payload             | Optional            | <Payload> child of <BDBInfo>                                 | Yes                        | Yes                  |
| CBEFF_SB_format_owner         | Optional            | <FormatOwner> child of <SBInfo>                              | Yes                        | Yes                  |
| CBEFF_SB_format_type          | Optional            | <FormatType> child of <sb-info>                              | Yes                        | Yes                  |
| CBEFF_BIR_validity_period     | Optional            | <NotValidBefore> and <NotValidAfter> attributes of <BIRInfo> | Yes                        | Yes                  |
| patron_header_version         | Optional            | <Major> and <Minor> children of <Version>                    | Yes                        | Yes                  |
| CBEFF_version                 | Optional            | <Major> and <Minor> children of <CBEFFVersion>               | Yes                        | Yes                  |
| BDB                           | Optional            | <BDB>  | Yes                        | Yes                  |
| SB                            | Optional            | <SB>   | Yes                        | Yes                  |

4934

## 4935 B.22 XML schema of the BIAS patron format

```

4936 <?xml version='1.0' encoding="utf-8"?>
4937 <xs:schema
4938   xmlns:xs="http://www.w3.org/2001/XMLSchema"
4939   xmlns="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
4940   targetNamespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
4941   elementFormDefault="qualified">
4942
4943   <xs:element name="BIR" type="BIRType"/>
4944
4945   <xs:complexType name="BIRType">
4946     <xs:sequence>
4947       <xs:element name="Version" type="VersionType" minOccurs="0"/>
4948       <xs:element name="CBEFFVersion" type="VersionType" minOccurs="0"/>
4949       <xs:any namespace="##other" processContents="skip" minOccurs="0" maxOccurs="unbounded"/>
4950       <xs:element name="BIRInfo" type="BIRInfoType"/>
4951       <xs:element name="BDBInfo" type="BDBInfoType" minOccurs="0"/>
4952       <xs:element name="SBInfo" type="SBInfoType" minOccurs="0"/>
4953       <xs:element name="BIR" type="BIRType" minOccurs="0" maxOccurs="unbounded"/>
4954       <xs:element name="BDB" type="xs:base64Binary" minOccurs="0"/>
4955       <xs:element name="SB" type="xs:base64Binary" minOccurs="0"/>
4956     </xs:sequence>

```

```

4957 </xs:complexType>
4958
4959 <xs:complexType name="VersionType">
4960 <xs:sequence>
4961 <xs:element name="Major" type="xs:nonNegativeInteger" />
4962 <xs:element name="Minor" type="xs:nonNegativeInteger" />
4963 </xs:sequence>
4964 </xs:complexType>
4965
4966 <xs:complexType name="BIRInfoType">
4967 <xs:sequence>
4968 <xs:element name="Creator" type="xs:string" minOccurs="0"/>
4969 <xs:element name="Index" type="UUIDType" minOccurs="0"/>
4970 <xs:element name="Payload" type="xs:base64Binary" minOccurs="0"/>
4971 <xs:element name="Integrity" type="xs:boolean" />
4972 <xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />
4973 <xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />
4974 <xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />
4975 </xs:sequence>
4976 </xs:complexType>
4977
4978 <xs:complexType name="BDBInfoType">
4979 <xs:sequence>
4980 <xs:element name="ChallengeResponse" type="xs:base64Binary" minOccurs="0"/>
4981 <xs:element name="Index" type="UUIDType" minOccurs="0"/>
4982 <xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />
4983 <xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />
4984 <xs:element name="Encryption" type="xs:boolean" minOccurs="0" />
4985 <xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />
4986 <xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />
4987 <xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />
4988 <xs:element name="Type" type="MultipleTypesType" minOccurs="0" />
4989 <xs:element name="Subtype" type="SubtypeType" minOccurs="0" />
4990 <xs:element name="Level" type="ProcessedLevelType" minOccurs="0" />
4991 <xs:element name="ProductOwner" type="xs:positiveInteger" minOccurs="0" />
4992 <xs:element name="ProductType" type="xs:positiveInteger" minOccurs="0" />
4993 <xs:element name="CaptureDeviceOwner" type="xs:positiveInteger" minOccurs="0" />
4994 <xs:element name="CaptureDeviceType" type="xs:positiveInteger" minOccurs="0" />
4995 <xs:element name="FeatureAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
4996 <xs:element name="FeatureAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
4997 <xs:element name="ComparisonAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
4998 <xs:element name="ComparisonAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
4999 <xs:element name="QualityAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5000 <xs:element name="QualityAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5001 <xs:element name="CompressionAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5002 <xs:element name="CompressionAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5003 <xs:element name="Purpose" type="PurposeType" minOccurs="0" />
5004 <xs:element name="Quality" type="QualityType" minOccurs="0" />
5005 </xs:sequence>
5006 </xs:complexType>
5007
5008 <xs:complexType name="SBInfoType">
5009 <xs:sequence>
5010 <xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />
5011 <xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />
5012 </xs:sequence>
5013 </xs:complexType>
5014
5015 <xs:simpleType name="QualityType">
5016 <xs:restriction base="xs:integer">
5017 <xs:minInclusive value="-2"/>
5018 <xs:maxInclusive value="100"/>
5019 </xs:restriction>
5020 </xs:simpleType>
5021
5022 <xs:simpleType name="SingleTypeType">
5023 <xs:restriction base="xs:string">
5024 <xs:enumeration value="Scent"/>
5025 <xs:enumeration value="DNA"/>
5026 <xs:enumeration value="Ear" />
5027 <xs:enumeration value="Face"/>

```

```

5028     <xs:enumeration value="Finger"/>
5029     <xs:enumeration value="Foot"/>
5030     <xs:enumeration value="HandGeometry"/>
5031     <xs:enumeration value="Vein"/>
5032     <xs:enumeration value="Iris"/>
5033     <xs:enumeration value="Retina"/>
5034     <xs:enumeration value="Voice"/>
5035     <xs:enumeration value="Gait"/>
5036     <xs:enumeration value="Keystroke"/>
5037     <xs:enumeration value="LipMovement"/>
5038     <xs:enumeration value="SignatureSign"/>
5039   </xs:restriction>
5040 </xs:simpleType>
5041
5042 <xs:simpleType name="MultipleTypesType">
5043   <xs:list itemType="SingleTypeType"/>
5044 </xs:simpleType>
5045
5046 <xs:simpleType name="SingleAnySubtypeType">
5047   <xs:restriction base="xs:string">
5048     <xs:enumeration value="Left"/>
5049     <xs:enumeration value="Right"/>
5050     <xs:enumeration value="Thumb"/>
5051     <xs:enumeration value="IndexFinger"/>
5052     <xs:enumeration value="MiddleFinger"/>
5053     <xs:enumeration value="RingFinger"/>
5054     <xs:enumeration value="LittleFinger"/>
5055   </xs:restriction>
5056 </xs:simpleType>
5057
5058 <xs:simpleType name="SingleVeinOnlySubtypeType">
5059   <xs:restriction base="xs:string">
5060     <xs:enumeration value="LeftVein"/>
5061     <xs:enumeration value="RightVein"/>
5062     <xs:enumeration value="Palm"/>
5063     <xs:enumeration value="BackOfHand"/>
5064     <xs:enumeration value="Wrist"/>
5065     <xs:enumeration value="Reserved1"/>
5066     <xs:enumeration value="Reserved2"/>
5067   </xs:restriction>
5068 </xs:simpleType>
5069
5070 <xs:simpleType name="MultipleAnySubtypesType">
5071   <xs:list itemType="SingleAnySubtypeType"/>
5072 </xs:simpleType>
5073
5074 <xs:simpleType name="MultipleVeinOnlySubtypesType">
5075   <xs:list itemType="SingleVeinOnlySubtypeType"/>
5076 </xs:simpleType>
5077
5078 <xs:simpleType name="SubtypeType">
5079   <xs:union memberTypes="MultipleAnySubtypesType MultipleVeinOnlySubtypesType"/>
5080 </xs:simpleType>
5081
5082 <xs:simpleType name="ProcessedLevelType">
5083   <xs:restriction base="xs:string">
5084     <xs:enumeration value="Raw"/>
5085     <xs:enumeration value="Intermediate"/>
5086     <xs:enumeration value="Processed"/>
5087   </xs:restriction>
5088 </xs:simpleType>
5089
5090 <xs:simpleType name="PurposeType">
5091   <xs:restriction base="xs:string">
5092     <xs:enumeration value="Verify"/>
5093     <xs:enumeration value="Identify"/>
5094     <xs:enumeration value="Enroll"/>
5095     <xs:enumeration value="EnrollVerify"/>
5096     <xs:enumeration value="EnrollIdentify"/>
5097     <xs:enumeration value="Audit"/>
5098   </xs:restriction>

```



```

5099     </xs:simpleType>
5100
5101     <xs:simpleType name="UIDType">
5102       <xs:restriction base="xs:string">
5103         <xs:pattern value="[a-fA-F0-9]{8}\-([a-fA-F0-9]{4}\-){3}[a-fA-F0-9]{12}"/>
5104       </xs:restriction>
5105     </xs:simpleType>
5106   </xs:schema>
5107

```

5108 NOTE NO VALUE AVAILABLE is encoded by the absence of optional fields in the XML  
5109 encoding. There is little value in, for example, having the following string appear in a  
5110 record: <level> no value available </level>.

5111

## 5112 B.23 Sample BIR encoding

5113 An example of a simple BIR in XML encoding (complying with the XSD schema and the normative textual  
5114 description) follows.

```

5115
5116     <?xml version="1.0" encoding="utf-8"?>
5117     <BIR xmlns=" http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/">
5118       <Version>
5119         <Major>1</Major>
5120         <Minor>0</Minor>
5121       </Version>
5122       <CBEFFVersion>
5123         <Major>2</Major>
5124         <Minor>0</Minor>
5125       </CBEFFVersion>
5126       <BIRInfo>
5127         <Creator>ABCDE</Creator>
5128         <Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
5129         <Payload>a2V2aW4ubWFFuZ29sZEBuaXN0Lmdvdg==</Payload>
5130         <Integrity>true</Integrity>
5131         <CreationDate>2004-03-02T15:03:15Z</CreationDate>
5132         <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
5133         <NotValidAfter>2004-03-03T15:00:00Z</NotValidAfter>
5134       </BIRInfo>
5135       <BDBInfo>
5136
5137         <ChallengeResponse>VmlzaXQgaHR0cDovL2J3cy5uaXN0LmdvdiBmb3lgc29tZSBhd2Vzb21lIGJpb21ldH
5138         JpY3Mvd2ViIHNicnZpY2UgcHJvamVjdHMh</ChallengeResponse>
5139         <Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
5140         <FormatOwner>51</FormatOwner>
5141         <FormatType>88</FormatType>
5142         <Encryption>true</Encryption>
5143         <CreationDate>2004-03-02T15:00:00Z</CreationDate>
5144         <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
5145         <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
5146         <Type>Iris</Type>
5147         <Subtype>Left</Subtype>
5148         <Level>Processed</Level>
5149         <ProductOwner>16</ProductOwner>

```

5150                   <ProductType>2</ProductType>  
5151                   <Purpose>Verify</Purpose>  
5152                   <Quality>100</Quality>  
5153                   </BDBInfo>  
5154                   <SBInfo>  
5155                    <FormatOwner>51</FormatOwner>  
5156                    <FormatType>99</FormatType>  
5157                   </SBInfo>  
5158                   <BDB>a2V2aW4ubWfuZ29sZEBuaXN0Lmdvdg==</BDB>  
5159                   <SB>TmF0aW9uYWwgSW5zdGI0dXRlIG9mIFN0YW5kYXJkcyBhbmQgVG9jaG5vbG9neQ==</SB>  
5160                   </BIR>

5161

## Annex C. Use Cases (non-normative)

5162 The intent of this annex is to provide operational sequence diagrams / flow charts that show how the  
5163 higher level usage scenarios within [INCITS-BIAS] could be implemented using the BIAS SOAP profile.

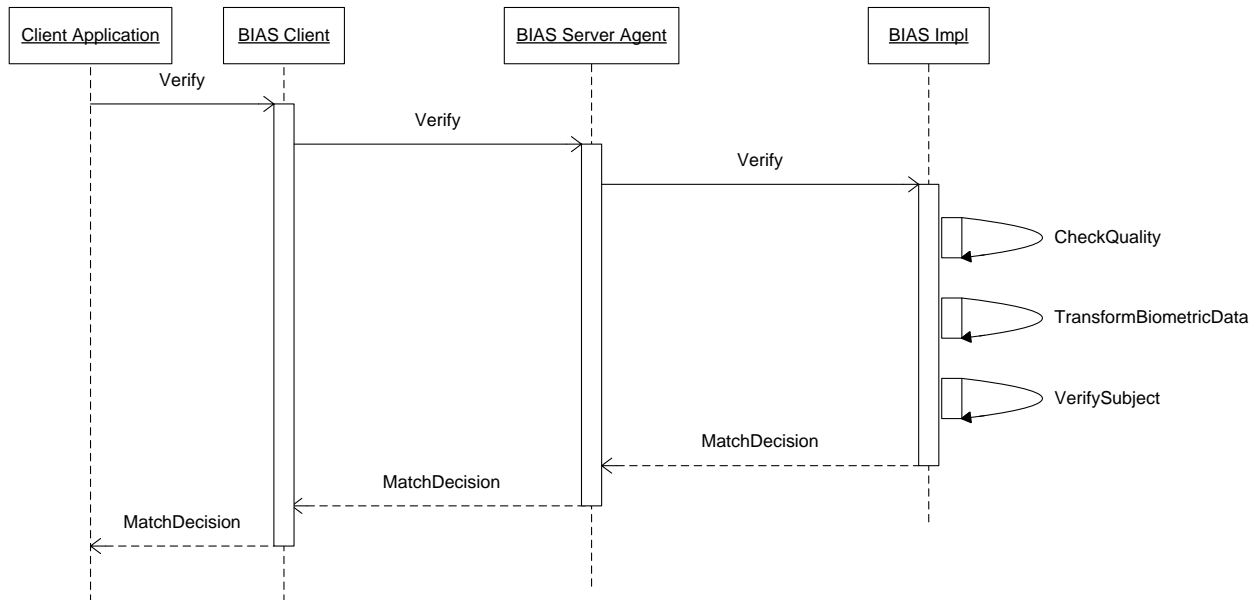
5164 The following use cases are given:

- 5165 • Verification (synchronous/aggregate)
- 5166 • Verification (asynchronous/aggregate)
- 5167 • Verification (primitive)
- 5168 • Identification (primitive)
- 5169 • Enrollment (aggregate)
- 5170 • Enrollment (primitive)

### 5171 C.1 Verification Use Case

5172 This use case uses the aggregate Verify operation in which a single request results in some set of  
5173 operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service  
5174 provider.

5175



Note that  
 1. CheckQuality, TransformBiometricData, VerifySubject can be exposed as interfaces of BIAS server agent.

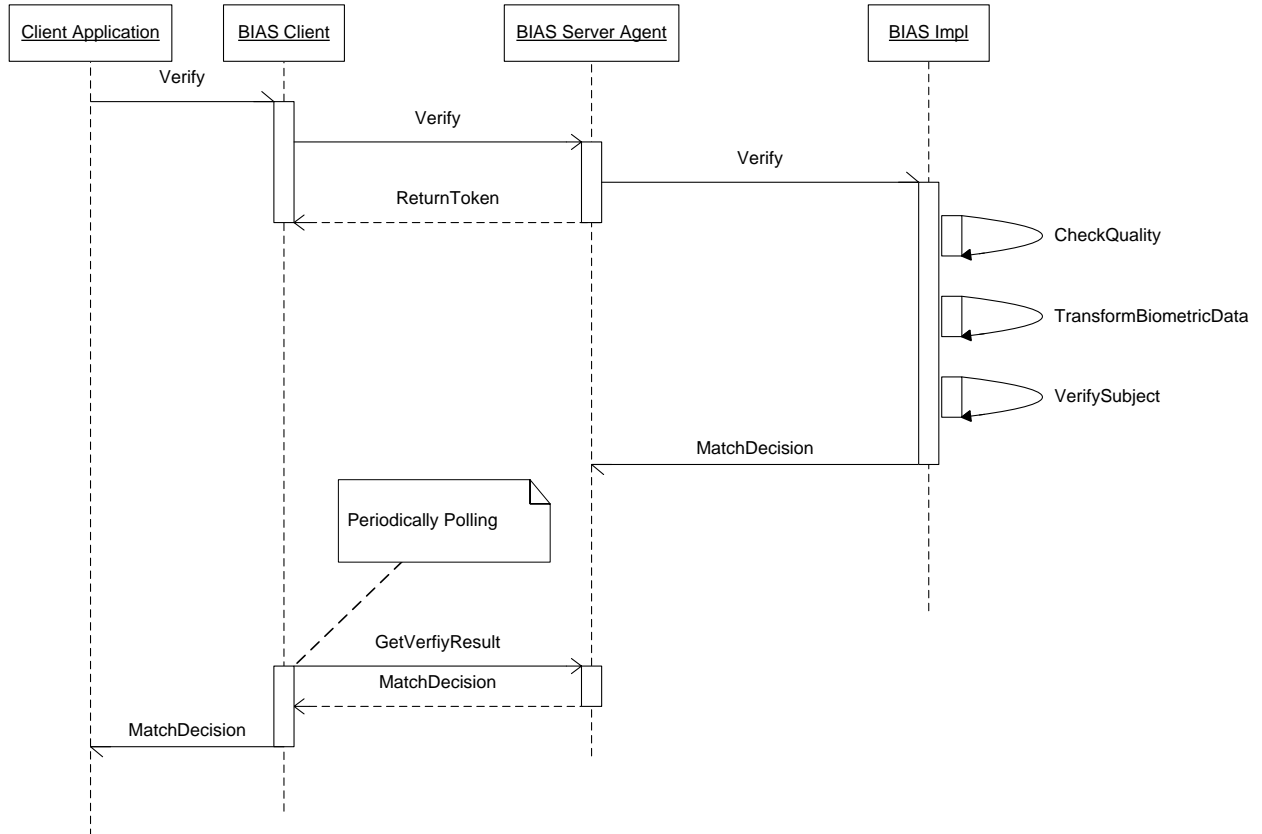
5176

5177

5178 **C.2 Asynchronous Verification Use Case**

5179 In this use case, the requester issues two requests – the BIAS Verify request to initiate the operation  
 5180 followed by a BIAS GetVerifyResult request to retrieve the results of that operation.

5181



Note that  
 1. CheckQuality, TransformBiometricData, VerifySubject can be exposed as interfaces of BIAS server agent.

5182

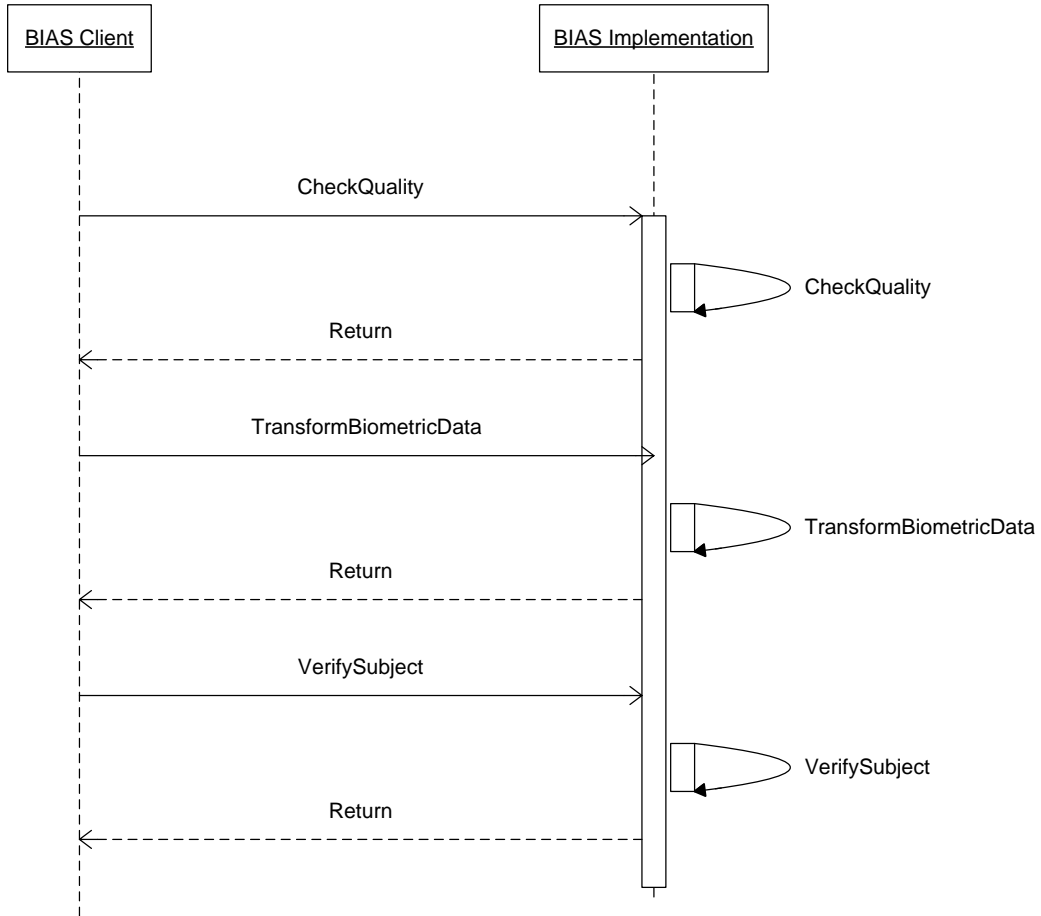
5183

5184

5185 **C.3 Primitive Verification Use Case**

5186 In this use case, the verification operation is performed as a series of requests using the BIAS primitive  
5187 operations. In this case, the client rather than the service provider controls the workflow of the higher  
5188 level operation.

5189



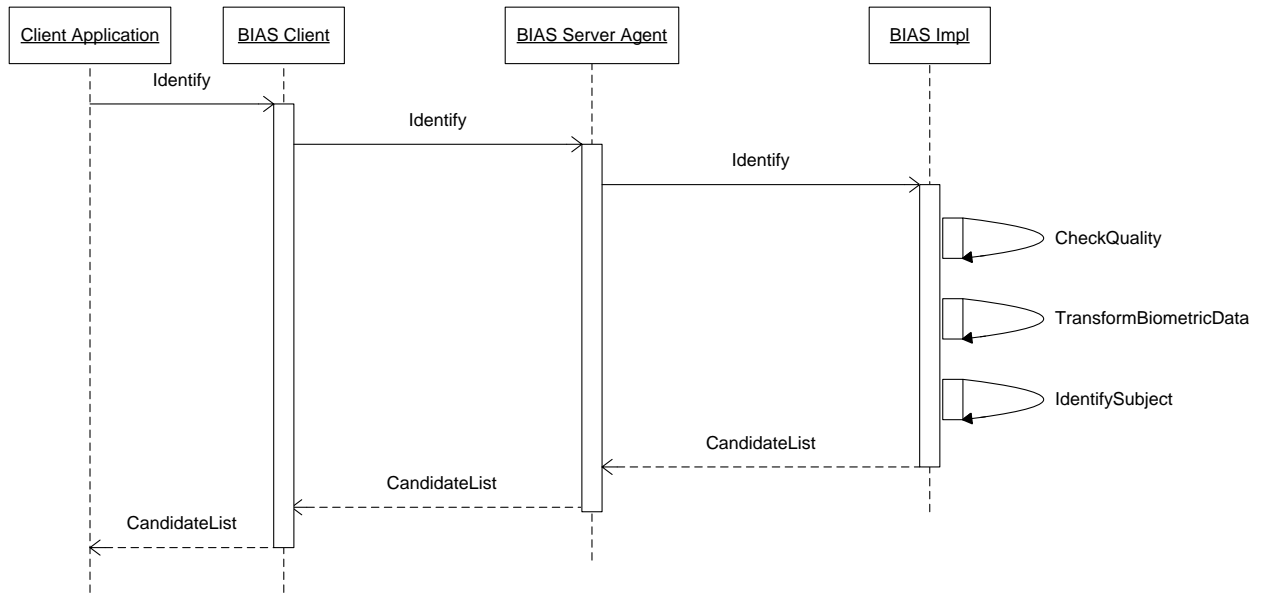
5190

5191

5192 **C.4 Identification Use Case**

5193 This use case uses the aggregate Identify operation in which a single request results in some set of  
5194 operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service  
5195 provider.

5196  
5197  
5198



Note that  
1. CheckQuality, TransformBiometricData, IdentifySubject can be exposed as interfaces of BIAS server agent.

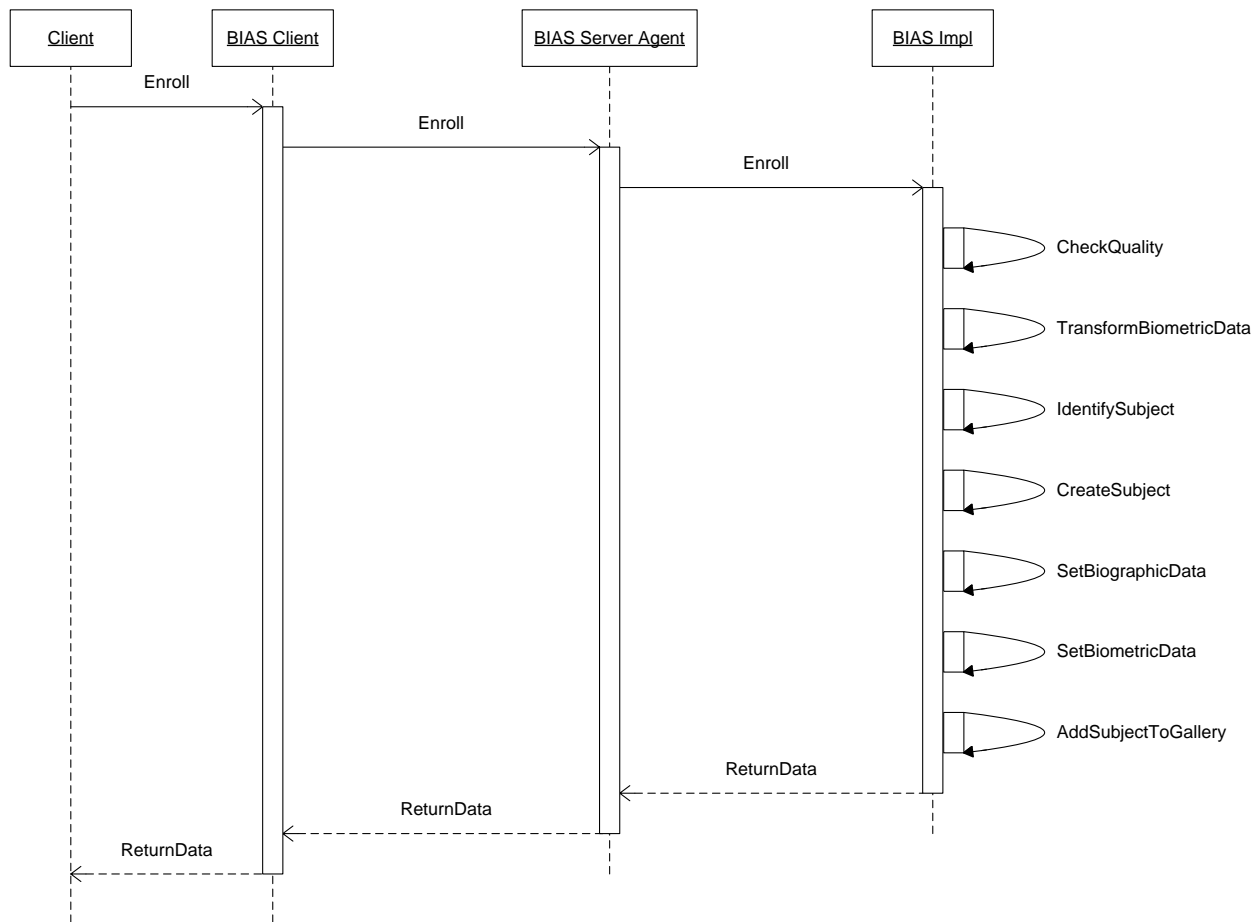
5199  
5200

5201 **C.5 Biometric Enrollment Use Case**

5202 This use case uses the aggregate Enroll operation in which a single request results in some set of  
5203 operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service  
5204 provider.

5205 Here, if the result of the IdentifySubject is no matches found, then the subject is added to the gallery. If a  
5206 match had been found then other logic may have been applied (e.g., return candidate list, add encounter  
5207 for existing subject, etc.).

5208



5209

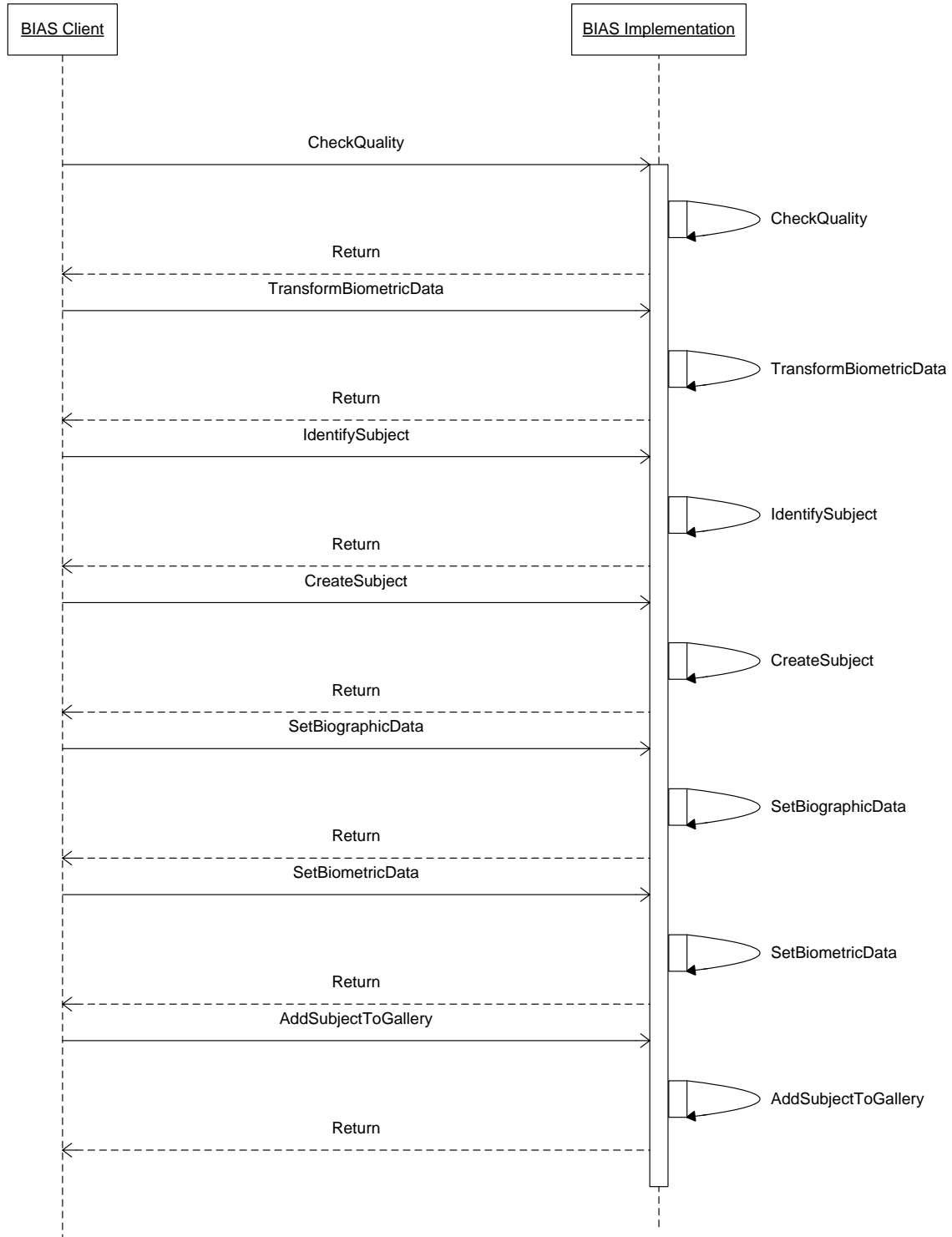
5210

5211

5212 **C.6 Primitive Enrollment Use Case**

5213 In this use case, the enrollment operation is performed as a series of requests using the BIAS primitive  
5214 operations. In this case, the client rather than the service provider controls the workflow of the higher  
5215 level operation.

5216



5217



---

## 5218 **Annex D. Samples (non-normative)**

### 5219 **D.1 Create Subject Request/Response Example**

#### 5220 *INCITS BIAS Specification*

```
5221 <interface name="CreateSubject">
5222     <parameter name="SubjectID" type="xs:string"
5223     direction="inout" use="optional" />
5224     <parameter name="Return" type="xs:unsignedLong" direction="out" />
5225 </interface>
```

#### 5226 *OASIS BIAS Examples*

##### 5227 **Simple Create Subject Request:**

```
5228 POST /bias HTTP/1.1
5229 Host: www.acme.com
5230 Content-Type: application/soap+xml; charset="utf-8"
5231 Content-Length: nnnn
5232 SOAPAction: "CreateSubject"
5233 <?xml version="1.0"?>
5234 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
5235     <soap:Body>
5236         <tns:CreateSubject
5237
5238         xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/" />
5239     </soap:Body>
5240 </soap:Envelope>
```

##### 5241 **Create Subject Request with SubjectID Parameter:**

```
5242 POST /bias HTTP/1.1
5243 Host: www.acme.com
5244 Content-Type: application/soap+xml; charset="utf-8"
5245 Content-Length: nnnn
5246 SOAPAction: "CreateSubject"
5247 <?xml version="1.0"?>
5248 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
5249     <soap:Body>
5250         <tns:CreateSubject
5251         xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
5252             <tns:Identity>
5253                 <tns:SubjectID>123456789</tns:SubjectID>
5254             </tns:Identity>
5255         </tns:CreateSubject>
5256     </soap:Body>
```

5257 </soap:Envelope>

5258 **Create Subject Request with Optional OASIS BIAS Content:**

5259 POST /bias HTTP/1.1

5260 Host: [www.acme.com](http://www.acme.com)

5261 Content-Type: application/soap+xml; charset="utf-8"

5262 Content-Length: nnnn

5263 SOAPAction: "CreateSubject"

5264 <?xml version="1.0"?>

5265 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">

5266 <soap:Body>

5267 <tns:CreateSubject

5268 xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">

5269 <tns:GenericRequestParameters>

5270 <tns:Application>BIAS Application</tns:Application>

5271 <tns:ApplicationUser>BIAS User</tns:ApplicationUser>

5272 </tns:GenericRequestParameters>

5273 <tns:Identity>

5274 <tns:SubjectID>123456789</tns:SubjectID>

5275 </tns:Identity>

5276 </tns:CreateSubject>

5277 </soap:Body>

5278 </soap:Envelope>

5279 **Simple Create Subject Response:**

5280 HTTP/1.1 200 OK

5281 Content-Type: application/soap+xml; charset="utf-8"

5282 Content-Length: nnnn

5283 <?xml version="1.0"?>

5284 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">

5285 <soap:Body>

5286 <tns:CreateSubjectResponse

5287 xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">

5288 <tns:ResponseStatus>

5289 <tns:Return>0</tns:Return>

5290 </tns:ResponseStatus>

5291 <tns:Identity>

5292 <tns:SubjectID>123456789</tns:SubjectID>

5293 </tns:Identity>

5294 </tns:CreateSubjectResponse>

5295 </soap:Body>

5296 </soap:Envelope>

5297 **Create Subject Response with Optional OASIS BIAS Content:**

5298 HTTP/1.1 200 OK  
5299 Content-Type: application/soap+xml; charset="utf-8"  
5300 Content-Length: nnnn  
5301 <?xml version="1.0"?>  
5302 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">  
5303     <soap:Body>  
5304         <tns:CreateSubjectResponse  
5305             xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">  
5306             <tns:ResponseStatus>  
5307                 <tns:Return>0</tns:Return>  
5308                 <tns:Message>Subject ID 123456789 successfully  
5309             created.</tns:Message>  
5310             </tns:ResponseStatus>  
5311             <tns:Identity>  
5312                 <tns:SubjectID>123456789</tns:SubjectID>  
5313             </tns:Identity>  
5314         </tns:CreateSubjectResponse>  
5315     </soap:Body>  
5316 </soap:Envelope>  
5317

## 5318 **D.2 Set Biographic Data Request/Response Example**

### 5319 *INCITS BIAS Specification*

```
5320 <interface name="SetBiographicData">  
5321     <parameter name="SubjectID" type="xs:string" direction="in" />  
5322     <parameter name="IdentityModel" type="IdentityModelType" direction="in"  
5323 />  
5324     <parameter name="EncounterID" type="xs:string" direction="inout"  
5325 use="optional" />  
5326     <parameter name="BiographicData"  
5327 type="BiographicDataType" direction="in" />  
5328     <parameter name="Return" type="xs:unsignedLong" direction="out" />  
5329 </interface>
```

### 5330 *OASIS BIAS Examples*

#### 5331 **Set Biographic Data Request:**

5332 POST /bias HTTP/1.1  
5333 Host: [www.acme.com](http://www.acme.com)  
5334 Content-Type: application/soap+xml; charset="utf-8"  
5335 Content-Length: nnnn  
5336 SOAPAction: "SetBiographicData"  
5337 <?xml version="1.0"?>  
5338 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">  
5339     <soap:Body>

```

5340         <tns:SetBiographicData
5341
5342     xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
5343         <tns:Identity>
5344             <tns:SubjectID>123456789</tns:SubjectID>
5345             <tns:BiographicData>
5346                 <tns:BiographicDataItem>
5347                     <tns:Name>Last</tns:Name>
5348                     <tns:Type>string</tns:Type>
5349                     <tns:Value>Doe</tns:Value>
5350                 </tns:BiographicDataItem>
5351             </tns:BiographicData>
5352         </tns:Identity>
5353         <tns:IdentityModel>person</tns:IdentityModel>
5354     </tns:SetBiographicData>
5355 </soap:Body>
5356 </soap:Envelope>

```

### 5357 **Set Biographic Data Response:**

```

5358 HTTP/1.1 200 OK
5359 Content-Type: application/soap+xml; charset="utf-8"
5360 Content-Length: nnnn
5361 <?xml version="1.0"?>
5362 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
5363     <soap:Body>
5364         <tns:SetBiographicDataResponse
5365
5366     xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
5367         <tns:ResponseStatus>
5368             <tns:Return>0</tns:Return>
5369         </tns:ResponseStatus>
5370     </tns:SetBiographicDataResponse>
5371 </soap:Body>
5372 </soap:Envelope>
5373

```

## 5374 **D.3 Set Biometric Data Request/Response Example**

### 5375 *INCITS BIAS Specification*

```

5376 <interface name="SetBiometricData">
5377     <parameter name="SubjectID" type="xs:string" direction="in" />
5378     <parameter name="IdentityModel" type="IdentityModelType" direction="in"
5379 />
5380     <parameter name="EncounterID" type="xs:string" direction="inout"
5381 use="optional" />

```

```
5382     <parameter name="BIRList" type="CBEFF_BIR_ListType" direction="in" />
5383     <parameter name="Return" type="xs:unsignedLong" direction="out" />
5384 </interface>
```

### 5385 *OASIS BIAS Examples*

#### 5386 **Set Biometric Data Request:**

```
5387 POST /bias HTTP/1.1
5388 Host: www.acme.com
5389 Content-Type: application/soap+xml; charset="utf-8"
5390 Content-Length: nnnn
5391 SOAPAction: "SetBiometricData"
5392 <?xml version="1.0"?>
5393 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
5394   <soap:Body>
5395     <tns:SetBiometricData
5396
5397       xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
5398       <tns:Identity>
5399         <tns:SubjectID>123456789</tns:SubjectID>
5400         <tns:BiometricData>
5401           <tns:BIRList>
5402             <tns:BIR>biometric data</tns:BIR>
5403           </tns:BIRList>
5404         </tns:BiometricData>
5405       </tns:Identity>
5406       <tns:IdentityModel>person</tns:IdentityModel>
5407     </tns:SetBiometricData>
5408   </soap:Body>
5409 </soap:Envelope>
```

#### 5410 **Set Biometric Data Response:**

```
5411 HTTP/1.1 200 OK
5412 Content-Type: application/soap+xml; charset="utf-8"
5413 Content-Length: nnnn
5414 <?xml version="1.0"?>
5415 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
5416   <soap:Body>
5417     <tns:SetBiometricDataResponse
5418
5419       xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
5420       <tns:ResponseStatus>
5421         <tns:Return>0</tns:Return>
5422       </tns:ResponseStatus>
5423     </tns:SetBiometricDataResponse>
```

5424 </soap:Body>

5425 </soap:Envelope>

5426

## Annex E. Acknowledgements

5427 The following individuals have participated in the creation of this specification and are gratefully  
5428 acknowledged:

5429

5430 **Participants:**

5431

| Name                    | Affiliation                        |
|-------------------------|------------------------------------|
| Mr. Young Bang          | Booz Allen Hamilton                |
| Mr. Ed. Clay            | Sun                                |
| Mr. Murty Gurajada *    | Raining Data Corporation           |
| Mr. Dale Hapeman        | US Department of Defense           |
| Dr. Charles Li          | Raytheon                           |
| Mr. Kevin Mangold       | NIST                               |
| Mr. John Mayer-Splain   | US Department of Homeland Security |
| Dr. Ross Michaels       | NIST                               |
| Mr. Ramesh Nagappan     | Sun                                |
| Mr. Ash Parikh *        | Raining Data Corporation           |
| Mr. Matthew Swayze      | Daon                               |
| Mr. Guy Swope*          | Raytheon                           |
| Mrs. Catherine Tilton   | Daon                               |
| Mr. Alessandro Triglia* | OSS Nokalva                        |
| Mr. Matthew Young       | US Department of Defense           |
| Mr. Brad Wing           | NIST (formerly DHS)                |
| Mr. Michael Wittman*    | Raytheon                           |
| Mr. Gregory Zektser     | Booz Allen Hamilton                |

5432

5433 \* Though no longer members of the BIAS TC at time of publication, these individuals contributed in the  
5434 early stages of the development of this standard.

5435 In addition, the inputs from the INCITS technical committee M1 are also gratefully appreciated.

5436

---

## Annex F. Revision History

5437

| Revision | Date       | Editor        | Changes Made  |
|----------|------------|---------------|---|
| WD 1     | 2014-04-03 | Kevin Mangold | Updated BIAS Patron Format to replace normative XML/WSDL with informative |

5438