



Biometric Identity Assurance Services (BIAS) SOAP Profile Version 2.0

Committee Specification Draft 01

21 April 2016

Specification URIs

This version:

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csd01/BIAS-v2.0-csd01.pdf> (Authoritative)
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csd01/BIAS-v2.0-csd01.html>
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csd01/BIAS-v2.0-csd01.docx>

Previous version:

N/A

Latest version:

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.pdf> (Authoritative)
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.html>
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.docx>

Technical Committee:

[OASIS Biometric Services \(BIOSERV\) TC](#)

Chair:

Kevin Mangold (kevin.mangold@nist.gov), NIST

Editors:

Kevin Mangold (kevin.mangold@nist.gov), NIST
Karen Marshall (karen.marshall@nist.gov), NIST

Additional artifacts:

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

- XML schemas: <http://docs.oasis-open.org/bioserv/BIAS/v2.0/csd01/schemas/>

Related work:

This specification is related to:

- ISO/IEC 30108-1:2015, Biometric Identity Assurance Services (BIAS). <http://www.iso.org>.

Declared XML namespaces:

- <http://docs.oasis-open.org/bioserv/ns/bias-2.0>
- <http://docs.oasis-open.org/bioserv/ns/biaspatronformat-2.0>

Abstract:

BIAS defines biometric services used for identity assurance that are invoked over a services-based framework. It is intended to provide a generic set of biometric and identity-related functions and associated data definitions to allow remote access to biometric services.

Status:

This document was last revised or approved by the OASIS Biometric Services (BIOSERV) 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=bioserv#technical.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "[Send A Comment](#)" button on the TC's web page at <https://www.oasis-open.org/committees/bioserv/>.

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 TC's web page (<https://www.oasis-open.org/committees/bioserv/ipr.php>).

Citation format:

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

[BIASPROFILE]

Biometric Identity Assurance Services (BIAS) SOAP Profile Version 2.0. Edited by Kevin Mangold and Karen Marshall. 21 April 2016. OASIS Committee Specification Draft 01. <http://docs.oasis-open.org/bioserv/BIAS/v2.0/csd01/BIAS-v2.0-csd01.html>. Latest version: <http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.html>.

Notices

Copyright © OASIS Open 2016. 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	9
1.1	Purpose/Scope	9
1.2	Overview	9
1.3	Background.....	9
1.4	Relationship to Other Standards	10
1.5	Terminology	10
1.6	References.....	11
1.6.1	Normative References.....	11
1.6.2	Non-Normative References.....	12
2	Design Concepts and Architecture (non-normative)	14
2.1	Philosophy	14
2.2	Context.....	14
2.3	Architecture.....	14
3	Data dictionary.....	17
3.1	Documentation Conventions.....	17
3.2	Common Elements	18
3.2.1	ApplicationIdentifier	18
3.2.2	ApplicationUserIdentifier	18
3.2.3	BaseBIRType	18
3.2.4	BIASBiometricDataType.....	18
3.2.5	BIASFaultCode.....	18
3.2.6	BIASFaultDetail	20
3.2.7	BIASIdentity	21
3.2.8	BIASIDType.....	21
3.2.9	BinaryBIR	21
3.2.10	BiographicDataItem Type.....	22
3.2.11	BiographicDataItemListType	22
3.2.12	BiographicDataListType	22
3.2.13	BiographicDataSetType	22
3.2.14	BiographicDataType	23
3.2.15	BiometricDataType	24
3.2.16	BiometricDataListType	24
3.2.17	CandidateListResultType	24
3.2.18	CandidateListType	25
3.2.19	CandidateType	25
3.2.20	CapabilityListType	25
3.2.21	CapabilityName	25
3.2.22	CapabilityType.....	31
3.2.23	CBEFF_BIR_ListType	31
3.2.24	CBEFF_BIR_Type.....	31
3.2.25	Classification	32
3.2.26	ClassificationAlgorithmType	32
3.2.27	ClassificationData.....	33

3.2.28	DocumentDataType	33
3.2.29	DocumentDataListType	35
3.2.30	EncounterCategoryType	35
3.2.31	EncounterListType	35
3.2.32	FusionDecision	35
3.2.33	FusionIdentityListType	36
3.2.34	FusionInformationListType	36
3.2.35	FusionInformationType	36
3.2.36	FusionResult	36
3.2.37	FusionScore	36
3.2.38	GenericRequestParameters	37
3.2.39	IdentifySubjectResultType	37
3.2.40	InformationType	37
3.2.41	ListFilterType	37
3.2.42	MatchType	38
3.2.43	OptionType	38
3.2.44	ProcessingOptionsType	38
3.2.45	ProductID	38
3.2.46	QualityData	38
3.2.47	ResponseStatus	39
3.2.48	ReturnCode	39
3.2.49	ScoreType	39
3.2.50	TokenResultType	40
3.2.51	TokenType	40
3.2.52	URI_BIR	40
3.2.53	VendorIdentifier	40
3.2.54	Version	40
3.2.55	VersionType	41
3.2.56	XML_BIR	41
4	BIAS Messages	42
4.1	Primitive Operations	42
4.1.1	AddSubjectToGallery	42
4.1.2	CheckQuality	43
4.1.3	ClassifyBiometricData	45
4.1.4	CreateEncounter	47
4.1.5	CreateSubject	49
4.1.6	DeleteBiographicData	50
4.1.7	DeleteBiometricData	51
4.1.8	DeleteDocumentData	53
4.1.9	DeleteEncounter	54
4.1.10	DeleteSubject	56
4.1.11	DeleteSubjectFromGallery	57
4.1.12	GetIdentifySubjectResults	58
4.1.13	IdentifySubject	60
4.1.14	ListBiographicData	63

4.1.15 ListBiometricData	65
4.1.16 ListDocumentData	70
4.1.17 PerformFusion	72
4.1.18 QueryCapabilities	74
4.1.19 RetrieveBiographicData	76
4.1.20 RetrieveBiometricData	78
4.1.21 RetrieveDocumentData	80
4.1.22 SetBiographicData	81
4.1.23 SetBiometricData	83
4.1.24 SetDocumentData	85
4.1.25 TransformBiometricData	87
4.1.26 UpdateBiographicData	88
4.1.27 UpdateBiometricData	90
4.1.28 UpdateDocumentData	91
4.1.29 VerifySubject	93
4.2 Aggregate Operations	95
4.2.1 Delete	95
4.2.2 Enroll	97
4.2.3 GetDeletionResults	99
4.2.4 GetEnrollResults	101
4.2.5 GetIdentifyResults	102
4.2.6 GetUpdateResults	104
4.2.7 GetVerifyResults	105
4.2.8 Identify	107
4.2.9 RetrieveData	109
4.2.10 Update	111
4.2.11 Verify	113
5 Message structure and rules	116
5.1 Purpose and constraints	116
5.2 Message requirements	117
5.3 Handling binary data	118
5.3.1 Base64 encoding	118
5.3.2 Use of XOP	119
5.4 Discovery	119
5.5 Identifying operations	119
5.5.1 Operation name element	119
5.5.2 WS-Addressing Action	120
5.6 Security	121
5.6.1 Use of SSL 3.0 or TLS 1.0	121
5.6.2 Data Origin Authentication	121
5.6.3 Message Integrity	121
5.6.4 Message Confidentiality	121
5.6.5 CBEFF BIR security features	122
5.6.6 Security Considerations	122
5.6.7 Security of Stored Data	122

5.6.8 Key Management	122
5.7 Use with other WS* standards.....	122
5.8 Tailoring	122
6 Error handling	124
6.1 BIAS operation return codes.....	124
6.2 SOAP fault codes	124
7 Conformance	125
Appendix A. XML Schema.....	126
Appendix B. BIAS Patron format specification	249
B.1 Patron	249
B.2 Patron identifier	249
B.3 Patron format name.....	249
B.4 Patron format identifier	249
B.5 ASN.1 object identifier for this patron format	249
B.6 Domain of use	249
B.7 Version identifier.....	249
B.8 CBEFF version	249
B.9 General.....	250
B.10 Specification	250
B.11 Element <BIR>	251
B.11.1 Syntax	251
B.11.2 Semantics	251
B.12 Element <Version>	252
B.12.1 Syntax	252
B.12.2 Semantics	252
B.13 Element <CBEFFVersion>	253
B.13.1 Syntax	253
B.13.2 Semantics	253
B.14 Element <BIRInfo>	253
B.14.1 Syntax	253
B.14.2 Semantics	254
B.15 Element <BDBInfo>.....	255
B.15.1 Syntax	255
B.15.2 Semantics	257
B.16 Element <SBInfo>	261
B.16.1 Syntax	261
B.16.2 Semantics	262
B.17 Representation of Integers.....	262
B.18 Representation of Octet Strings	262
B.19 Representation of Date and Time of the Day.....	263
B.20 Representation of Universally Unique Identifiers	264
B.21 Patron format conformance statement.....	264
B.21.1 Identifying information.....	264
B.21.2 ISO/IEC 19785-1:2006/Amd 1:2010 to Patron Format Mapping.....	264
B.22 XML schema of the BIAS patron format.....	266

B.23 Sample BIR encoding.....	269
Appendix C. Use Cases (non-normative)	271
C.1 Verification Use Case.....	271
C.2 Asynchronous Verification Use Case.....	272
C.3 Primitive Verification Use Case.....	273
C.4 Identification Use Case	274
C.5 Biometric Enrollment Use Case	275
C.6 Primitive Enrollment Use Case	276
Appendix D. Samples (non-normative).....	277
D.1 Create Subject Request/Response Example.....	277
D.2 Set Biographic Data Request/Response Example	279
D.3 Set Biometric Data Request/Response Example	280
Appendix E. Acknowledgements	283
Appendix F. Revision History	284

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 ISO/IEC 30108-1:2015, Information technology — Biometric Identity Assurance Services [ISO/IEC-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, ISO/IEC 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 Standards Organization and the International Electrotechnical Commission (ISO/IEC) 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.

1.4 Relationship to Other Standards

This OASIS BIAS profile comprises a companion standard to ISO/IEC 30108-1:2015, Information technology — Biometric Identity Assurance Services, which defines the BIAS requirements and taxonomy, specifying the identity assurance operations and the associated data elements. This OASIS BIAS profile specifies the design concepts and architecture, data model and data dictionary, message structure and rules, and error handling necessary to invoke SOAP-based services that implement BIAS operations.

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

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

1.5 Terminology

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

The following additional terms and definitions are used:

Note: The terms and definitions specified in ISO/IEC 30801-1 [ISO/IEC-BIAS] also apply to this Standard.

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

BIAS

Biometric Identity Assurance Services

BIR

Biometric Information Record

ESB

Enterprise Service Bus

HTTP

HyperText Transfer Protocol

HTTPS

HyperText Transfer Protocol over SSL or HTTP Secure

IRI

Internationalized Resource Identifier

SOA

Service-Oriented Architecture

SOAP

Simple Object Access Protocol

SSL

Secure Sockets Layer

TLS

Transport Layer Security

UDDI

Universal Description, Discovery, and Integration

URI

Uniform Resource Identifier

VPN

Virtual Private Network

WSDL

Web Services Description Language

WSS

Web Services Security

XML
eXtensible Markup Language

CBEFF
Common Biometric Exchange Formats Framework - data elements and BIR formats specified in ISO/IEC 19785-1

BIAS implementation
software entity that is capable of creating, processing, sending, and receiving BIAS messages

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

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

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

BIAS response message
BIAS message conveying a response to a prior BIAS requestmessage

1.6 References

1.6.1 Normative References

- [RFC2119]** S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, IETF RFC 2119, March 1997
<http://www.ietf.org/rfc/rfc2119.txt>
- [CBEFF]** ISO/IEC19785-1:2006, Information technology – Common Biometric Exchange Formats Framework – Part 1: Data element specification, with Amendment 1:2010
<http://www.iso.org>
- [DATE-TIME]** ISO 8601:2004, Data elements and interchange formats — Information interchange — Representation of dates and times
<http://www.iso.org>
- [INCITS-BIAS]** ANSI INCITS 442-2010, Biometric Identity Assurance Services (BIAS), July 2010
<http://www.incits.org>
- [IRI]** M. Duerst, et al, Internationalized Resource Identifiers, RFC3987, January 2005
<http://www.ietf.org/rfc/rfc3987.txt>
- [ISO/IEC-BIAS]** ISO/IEC 30108-1:2015, Information technology — Biometric Identity Assurance Services — Part 1: BIAS Services
<http://www.iso.org>
- [SOAP11]** Simple Object Access Protocol (SOAP) 1.1, 8 May 2000
<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

- [URI]** T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifiers (URI): Generic Syntax*, RFC 3986, MIT/LCS, U.C. Irvine, Xerox Corporation, January 2005.
<http://ietf.org/rfc/rfc3986>
- [UTF-8]** ISO/IEC 10646:2003, Information technology — Universal Multiple-Octet Coded Character Set (UCS)
<http://www.iso.org>
- [WS-Addr]** W3C Recommendation, Web Services Addressing 1.0 - Core, and Web Services Addressing 1.0 - SOAP Binding, 9 May 2006
<http://www.w3.org/2002/ws/addr/>
- [WS-I-Basic]** Basic Profile Version 1.1, 10 April 2006
<http://www.ws-i.org/Profiles/BasicProfile-1.1-2006-04-10.html>
- [WS-I-Bind]** Web Services-Interoperability Organization (WS-I) Simple SOAP Binding Profile Version 1.0, 24 August 2004
<http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0-2004-08-24.html>
- [WSDL11]** Web Services Description Language (WSDL) 1.1, 15 March 2001
<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>
- [XML 10]** Extensible Markup Language (XML) 1.0, 16 August 2006
<http://www.w3.org/TR/2006/REC-xml-20060816/>
- [XOP]** XML-binary Optimized Packaging, W3C Recommendation, 25 January 2005
<http://www.w3.org/TR/2005/REC-xop10-20050125/>

1.6.2 Non-Normative References

- [BioAPI]** ISO/IEC 19784-1:2006, Information technology – Biometric Application Programming Interface – Part 1: BioAPI Specification
<http://www.iso.org>
- [CBEFF-3]** ISO/IEC 19785-3:2007, Information technology – Common Biometric Exchange Formats Framework – Part 3: Patron format specifications, with Amendment 1:2010
<http://www.iso.org>
- [BIO SEC]** ISO 19092 Financial services -- Biometrics -- Security framework
<http://www.iso.org>
- [EBTS-DOD]** Department of Defense Electronic Biometric Transmission Specification, Version 2.0, 27 March 2009
<http://www.biometrics.dod.mil/CurrentInitiatives/Standards/dodebts.aspx>
- [EBTS-FBI]** IAFIS-DOC-01078-8.1, “Electronic Biometric Transmission Specification (EBTS)”, Version 8.1, November 19, 2008, Federal Bureau of Investigation, Criminal Justice Information Services Division
<https://www.fbibiospecs.org>
- [EFTS]** IAFIS-DOC-01078-7, “Electronic Fingerprint Transmission Specification (EFTS)”, Version 7.1, May 2, 2005, Federal Bureau of Investigation, Criminal Justice Information Services Division
<https://www.fbibiospecs.org>
- [HR-XML]** HR-XML Consortium Library, 2007 April 15
<http://www.hr-xml.org>

184 **[INT-I]** Interpol Implementation of ANSI/NIST ITL1-2000, Ver 4.22b, October 28, 2005, The Interpol
 185 AFIS Expert Group
 186 <http://www.interpol.int>

187 **[NIEM]** National Information Exchange Model (NIEM), Ver 2.0, June 2007, US DOJ/DHS
 188 <http://www.niem.gov>

189 **[RFC2246]** T. Dierks & C. Allen, *The TLS Protocol, Version 1.0*, January 1999
 190 <http://www.ietf.org/rfc/rfc2246.txt>

191 **[RFC2617]** J. Franks, et al, *HTTP Authentication: Basic and Digest Access Authentication*, June
 192 1999
 193 <http://www.ietf.org/rfc/rfc2617.txt>

194 **[RFC3280]** R. Housley, et al, Internet X.509 Public Key Infrastructure Certificate and Certificate
 195 Revocation List (CRL) Profile, April 2002
 196 <http://www.ietf.org/rfc/rfc3280.txt>

197 **[SAML]** Security Assertion Markup Language (SAML), Oasis Standard, March 2005
 198 <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>

199 **[SAML SEC]** Security and Privacy Considerations for the OASIS Security Assertion Markup
 200 Language (SAML) V2.0, Oasis Standard, 15 March 2005
 201 <http://docs.oasis-open.org/security/saml/v2.0/saml-sec-consider-2.0-os.pdf>

202 **[SSL3]** SSL 3.0 Specification
 203 <http://www.freesoft.org/CIE/Topics/ssl-draft/3-SPEC.HTM>

204 **[WSS]** Web Services Security: SOAP Message Security 1.1, (WS-Security 2004), OASIS Standard
 205 Specification, 1 February 2006
 206 [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)
 207 [SOAPMessageSecurity.pdf](http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf)

208 **[X509]** X.509: Information technology - Open Systems Interconnection - The Directory: Public-key
 209 and attribute certificate frameworks, ITU-T, August 2005
 210 <http://www.itu.int/rec/T-REC-X.509-200508-I>

211 **[xNAL]** Customer Information Quality Specifications Version 3.0: Name (xNL), Address (xAL), Name
 212 and Address (xNAL) and Party (xPIL), Committee Specification 02, 20 September 2008
 213 http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ciq

214

215

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.

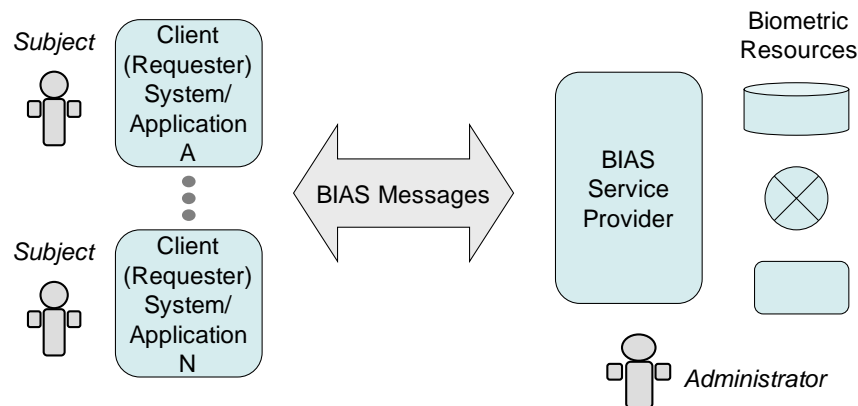


Figure 1. BIAS Context

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.

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

Note that services are represented as circles.

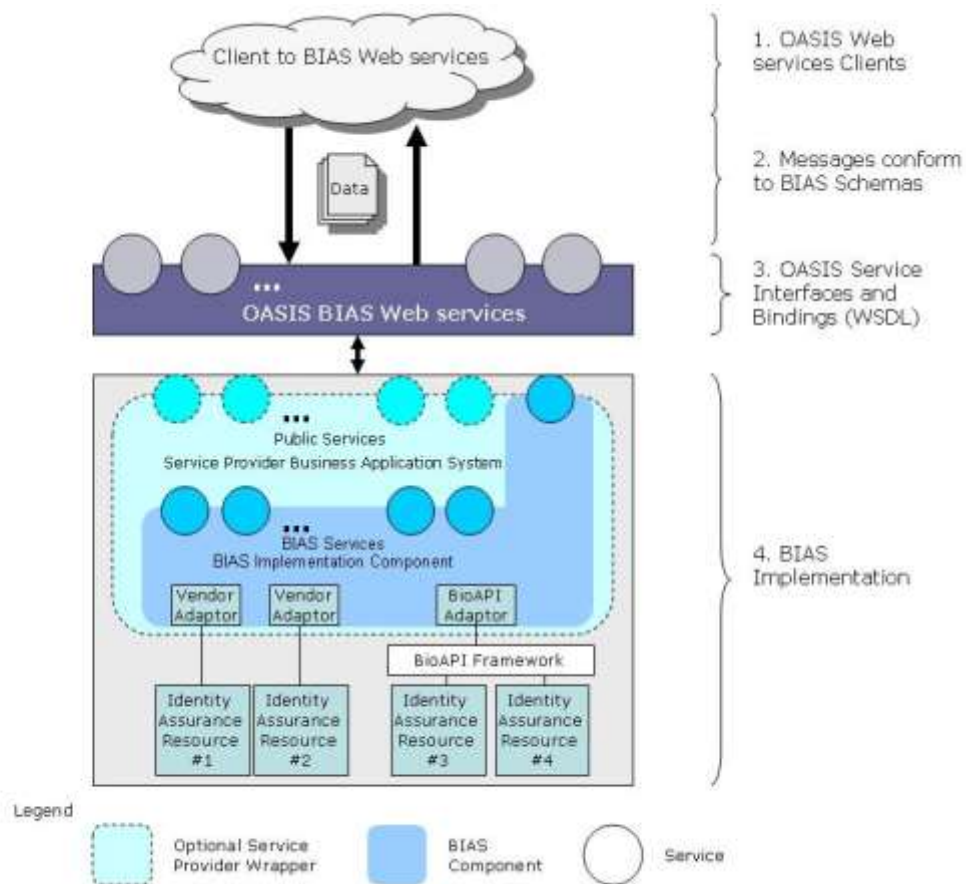


Figure 2. Representative BIAS Architecture

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

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

BIAS implementation

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

BIAS messages

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

BIAS endpoints

271 A BIAS endpoint is a runtime entity, uniquely identified and accessed by an endpoint URI/IRI [[URI](#)] [[IRI](#)],
272 capable of sending and receiving BIAS messages.

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

3 Data dictionary

This section describes the BIAS data elements used within BIAS messages (as defined in Clause 4). Common data elements are defined for use in one or more operations. These include common data types or return codes. BIAS data elements are defined in ISO/IEC 30108-1. The elements, complex types and simple types described for the BIAS messages belong to the following namespace: <http://docs.oasis-open.org/bias/ns/bias-1.0/>. See Annex A for the XML schema.

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

3.1 Documentation Conventions

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

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		

Figure 3. BIAS Message Input/Output Dictionary Table Headings

Fields Hierarchy Explained:

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

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

295 3.2 Common Elements

296 3.2.1 ApplicationIdentifier

Type:	string
Description:	Identifies an application.
Min Length:	1
Max Length:	255

297 3.2.2 ApplicationUserIdentifier

Type:	string
Description:	Identifies an application user or instance.
Min Length:	1
Max Length:	255

298 3.2.3 BaseBIRType

Type:	Schema complexType
Description:	Base type for all BIR subtypes; see BinaryBIR , URI_BIR , and XML_BIR for currently available types.

299 3.2.4 BIASBiometricDataType

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

300 3.2.5 BIASFaultCode

Type:	String
--------------	--------

Description:	Error code referenced in a SOAP fault.
---------------------	--

301

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.
UNKNOWN_DOCUMENT_CATEGORY	The data requested does not exist.
INVALID_TOKEN	The data requested does not exist.
TOKEN_EXPIRED	The data requested does not exist.
DUPLICATE_ENCOUNTER	The input encounter ID for a new encounter already exists for that subject.
IDENTIFICATION_RESULT_NOT_YET_AVAILABLE	The result of an asynchronous identification process is not yet available.

Value	Description
UNKNOWN_FORMAT	An unknown format was detected.
INVALID_LICENSE	An invalid license was found.
SERVICE_NOT_IMPLEMENTED	The requested service/function is not implemented.
INVALID_ENCOUNTER_TYPE	The input encounter type is not a recognized category.
INVALID_PROCESSING_OPTION	The supplied processing options are invalid.
CANNOT_STORE_DATA	Data cannot be stored due to an internal reason.
CANNOT_PROCESS_DATA	Data cannot be processed.
CANNOT_CREATE_TEMPLATE	A biometric template could not be created from the raw input data (either for purpose enroll or verify/identify).
CANNOT_DELETE_DATA	Data cannot be deleted.
CANNOT_RETRIEVE_DATA	Data cannot be retrieved (read).
CANNOT_INITIALIZE_INTERNAL_MODULES	Internal modules cannot be initialized and/or loaded.
CANNOT_VERIFY_DATA	Cannot perform a 1:N verification of the supplied and/or stored data.
CANNOT_IDENTIFY_DATA	Cannot perform a 1:N identification of the supplied and/or stored data.
INVALID_CONFIGURATION	The (central) system was configured improperly.
CANNOT_CHECK_QUALITY	The quality check cannot be performed due to an internal reason.
INTERNAL_DATABASE_ERROR	An internal error during the database connection occurred.
CANNOT_UPDATE_DATA	Data cannot be updated.
BIOMETRIC_TYPE_NOT_SUPPORTED	The biometric modality/type is not supported by the system.

NOTES:

- (1) See Clause 6 (Error handling) for an explanation of BIAS faults and return codes.
- (2) Service provider MAY define additional values specific to their service implementation.
- (3) See section 5.5 for additional information on BIAS security.

3.2.6 BIASFaultDetail

Field	Type	#	?	Meaning
BIASFaultDetail			Y	Defines the error information associated with a SOAP fault.
 BIASFaultType	BIASFaultCode	1	Y	References an error code.

Field	Type	#	?	Meaning
↑ BIASFaultMessage	string	1	Y	Provides a brief explanation of the fault.
↑ BIASFaultDescription	string	0..1	N	Provides detailed information about a BIAS fault, such as trace details.

307 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. The operations that use this type specify which elements are required.
↑ SubjectID	BIASIDType	0..1	C	A system unique identifier for a subject. Required as input to many operations.
↑ IdentityClaim	BIASIDType	0..1	N	An identifier by which a subject is known to a particular gallery or population group.
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
↑ EncounterList	EncounterListType	0..1	N	A list of encounters associated with a subject.
↑ BiographicData	BiographicDataType	0..1	N	An Identity's biographic data.
↑ BiometricData	BIASBiometricDataType	0..1	N	An Identity's biometric data.

308 3.2.8 BIASIDType

Type:	string
Description:	A BIAS Identifier.

309 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

3.2.10 BiographicDataItem Type

Field	Type	#	?	Meaning
BiographicDataItem Type			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.

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 BiographicDataItemList Type

Field	Type	#	?	Meaning
BiographicDataItemList Type			Y	Defines a list of biographic data elements.
↑ BiographicDataItem	BiographicDataItem Type	1..*	Y	Data structure containing information about a biographic record.

3.2.12 BiographicDataList Type

Field	Type	#	?	Meaning
BiographicDataList Type			Y	Defines a list of biographic data.
↑ BiographicData	BiographicData Type	0..*	N	Data structure containing information about a biographic record.

3.2.13 BiographicDataSet Type

Field	Type	#	?	Meaning
BiographicDataSet Type			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 Error! Reference source not found., 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).

Field	Type	#	?	Meaning
↑ 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.

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 ISO/IEC 30108 for further information.

3.2.14 BiographicDataType

Field	Type	#	?	Meaning
BiographicDataType			Y	Defines a set of biographic data elements, utilizing either the BiographicDataItemListType to represent a list of elements or the BiographicDataSetType 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.
↑ BiographicDataItemList	BiographicDataItemListType	0..1	N	A list of biographic data elements.
↑ BiographicDataItem	BiographicDataItem	1..*	Y	A single biographic data element.
↑ BiographicDataSet	BiographicDataSetType	0..1	N	A set of biographic data information.

NOTE: The implementer is given three choices for encoding biographic data:

- Encode only first and last name using the defined fields within BiographicDataType
- Define a list of biographic data elements using the BiographicDataItemListType
- Use a pre-defined set of biographic data (e.g., as specified in another standard) using the BiographicDataSetType.

See also ISO/IEC 30108-1, section 8.1 for further information.

328 3.2.15 BiometricDataType

Field	Type	#	?	Meaning
BiometricDataType			Y	Provides descriptive information about biometric data, such as the biometric type, subtype, and format, contained in the BDB of the CBEFF-BIR.
↑ BiometricType	<code>oasis_cbeff:MultipleTypesType</code>	1	Y	The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.
↑ BiometricTypeCount	<code>positiveInteger</code>	0..1	N	The number of biometric records having the biometric type recorded in the biometric type field.
↑ BiometricSubType	<code>oasis_cbeff:SubtypeType</code>	0..1	N	More specifically defines the type of biometric data stored in the biometric record, as defined by CBEFF.
↑ BDBFormatOwner	<code>positiveInteger</code>	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	<code>positiveInteger</code>	1	Y	Identifies the specific biometric data format specified by the CBEFF biometric organization recorded in the BDB Format Owner field.

329 3.2.16 BiometricDataListType

Field	Type	#	?	Meaning
BiometricDataListType			Y	A list of biometric data elements.
↑ BiometricDataRecord	BiometricDataType	0..*	N	Data structure containing information about a biometric record.

330 3.2.17 CandidateListResultType

Field	Type	#	?	Meaning
CandidateListResultType			Y	Defines a set of candidates, utilizing the CandidateType to represent each element in the set.
↑ CandidateList	CandidateListType	1	Y	The candidate list.

331 **3.2.18 CandidateListType**

Field	Type	#	?	Meaning
CandidateListType			Y	Defines a set of candidates, utilizing the CandidateType to represent each element in the set.
↑ Candidate	CandidateType	0..*	N	A single candidate.

332 **3.2.19 CandidateType**

Field	Type	#	?	Meaning
CandidateType			Y	Defines a single candidate as a possible match in response to a biometric identification request.
↑ ScoreList		0..1	N	A list of comparison(s) scores and optionally the type and subtype of the relating biometric
↑ Score	ScoreType	1	Y	The match score.
↑ Rank	integer	1	Y	The rank of the candidate in relation to other candidates for the same biometric identification operation.
↑ Identity				
↑ SubjectID	BIASIDType	0..1	Y	The identifier of the subject.
↑ BiographicData	BiographicDataType	0..1	N	Biographic data associated with the candidate match.
↑ BIRList	CBEFF_BIR_ListType	1	Y	Biometric data associated with the candidate match.

333 **3.2.20 CapabilityListType**

Field	Type	#	?	Meaning
CapabilityListType			Y	Defines a set of capabilities.
↑ Capability	CapabilityType	0..*	N	A single capability.

334 **3.2.21 CapabilityName**

Type:	string
Description:	A list of capability items.

335 **CapabilityName Enumeration Values**

Value	Description
-------	-------------

Value	Description
AggregateInputDataOptional	<p>A data element accepted as optional input by the implementing system for the aggregate services.</p> <p>The Capability Value should be set to the name of the data element that will be accepted by the aggregate services.</p> <p>The Capability Supporting Value that indicates the which services support the data element, using one or more of the following values, each separated by a comma:</p> <p>“Delete” “Enroll” “Identify” “Verify” “All”</p>
AggregateInputDataRequired	<p>A data element required as input by the implementing system for the aggregate services.</p> <p>The Capability Value should be set to the name of the data element that will be accepted by the aggregate services.</p> <p>The Capability Supporting Value that indicates the which services support the data element, using one or more of the following values, each separated by a comma:</p> <p>“Delete” “Enroll” “Identify” “Verify” “All”</p>

Value	Description
AggregateProcessingOption	<p>A processing option supported by the implementing system for the aggregate services.</p> <p>The Capability Value should be set to the option identifier, or “key” field, for the Processing Option parameter in the aggregate services.</p> <p>The Capability Supporting Value that indicates the option value, or “value” field, for the Processing Option parameter in the aggregate services, if applicable.</p> <p>The Capability Additional Info should indicate which aggregate services support the processing option, using one or more of the following values, each separated by a comma:</p> <p>“Delete” “Enroll” “Identify” “Verify” “Retrieve” “All”</p>
AggregateReturnData	<p>A data element returned by the implementing system for the aggregate services.</p> <p>The Capability Value should be set to the name of the data element that will be returned by the aggregate services.</p> <p>The Capability Supporting Value that indicates which services support the data element, using one or more of the following values, each separated by a comma:</p> <p>“Delete” “Enroll” “Identify” “Verify” “Retrieve” “All”</p>

Value	Description
AggregateServiceDescription	<p>Describes the processing logic of an aggregate service supported by the implementing system.</p> <p>The Capability Value should be set to the name of the data element that describes the aggregate services.</p> <p>The Capability Supporting Value that indicates the which services support the data element, using one or more of the following values, each separated by a comma:</p> <p>“Delete” “Enroll” “Identify” “Verify” “Retrieve”</p>
BiographicDataSet	<p>Identifies a biographic data set supported by the implementing system.</p> <p>The Capability Value should contain the name of the biographic data format supported by the implementing system(e.g. “EBTS” or “NIEM”).</p> <p>The Capability Supporting Value should contain the version of the biographic data format supported by the implementing system.</p> <p>The Capability Additional Info should contain the biographic data format type supported by the implementing system (e.g. ASCII or XML).</p>
CBEFFPatronFormat	<p>A patron format supported by the implementing system.</p> <p>The Capability Value should contain the format owner.</p> <p>The Capability Supporting Value should contain the format type.</p>
ClassificationAlgorithmType	<p>A classification algorithm type supported by the implementing system.</p> <p>The Capability Value should contain the name of classification alogorithm type supported by the implementing system.</p>

Value	Description
ConformanceClass	<p>Identifies the conformance class of the BIAS implementation.</p> <p>The Capability Value should be set to one of the following:</p> <ul style="list-style-type: none"> 1 – (Class 1 conformance) 2 – (Class 2 conformance) 3 – (Class 3 conformance) 4 – (Class 4 conformance) 5 – (Class 5 conformance) 6 – (Class 6 conformance) 7 – (Class 7 conformance)
Gallery	<p>A gallery or population group supported by the implementing system.</p> <p>The Capability Value should be the same as the value used for the Gallery ID parameter in the Add Subject to Gallery, Delete Biographic Data, Delete Biometric Data, Delete Subject From Gallery, Identify Subject, Retrieve Biographic Information, Retrieve Biometric Information, Retrieve Document Data, Set Biographic Data, Set Biometric Data, Set Document Data, and Verify Subject Services.</p>
IdentityModel	<p>Identifies whether the implementing system is person-centric or encounter-centric based.</p> <p>The Capability Value shall be set to one of the following:</p> <ul style="list-style-type: none"> “person” “encounter”
MatchAlgorithm	<p>A match algorithm vendor and algorithm vendor product ID supported by the implementing system.</p> <p>The Capability Value shall contain the algorithm vendor.</p> <p>The Capability Supporting Value shall contain the algorithm vendor product ID.</p> <p>The Capability Additional Info shall be set to the biometric type (defined by the XML Patron Format in ISO/IEC 19785-3) that corresponds to the match algorithm.</p> <p>The Capability Description shall contain the software version of the match algorithm.</p>

Value	Description
MatchScore	<p data-bbox="565 258 1414 285">Identifies the use of match scores returned by the implementing system.</p> <p data-bbox="565 342 1414 401">The Capability Value shall be set to the end-of-score-range that signifies a match.</p> <p data-bbox="565 457 1414 516">The Capability Supporting Value shall be set to the end-of-score-range that signifies a no-match.</p> <p data-bbox="565 573 1414 657">The Capability Additional Info shall be set to the biometric type (defined by the XML Patron Format in ISO/IEC 19785-3) that corresponds to the match score range.</p>
QualityAlgorithm	<p data-bbox="565 678 1414 737">A quality algorithm vendor and algorithm vendor product ID supported by the implementing system.</p> <p data-bbox="565 793 1227 821">The Capability Value shall contain the algorithm vendor.</p> <p data-bbox="565 877 1349 936">The Capability Supporting Value shall contain the algorithm vendor product ID.</p> <p data-bbox="565 993 1414 1077">The Capability Additional Info shall be set to the biometric type (defined by the XML Patron Format in ISO/IEC 19785-3) that corresponds to the quality algorithm.</p> <p data-bbox="565 1134 1349 1192">The Capability Description shall contain the software version of the quality algorithm.</p>
SupportedBiometric	<p data-bbox="565 1213 1224 1241">A biometric type supported by the implementing system.</p> <p data-bbox="565 1297 1414 1388">The Capability Value shall be set to the biometric type, as defined by the ZML Patron Format in ISO/IEC 19785-3 (for example, the biometric type for face is represented a “face”).</p> <p data-bbox="565 1444 1414 1528">The Capability Supporting Value shall indicate if the implementing system supports matching for the biometric type, using one of he following values:</p> <p data-bbox="565 1585 773 1612">“1” (identification)</p> <p data-bbox="565 1627 751 1654">“2” (verification)</p> <p data-bbox="565 1669 956 1696">“3” (identification and verification)</p> <p data-bbox="565 1711 924 1738">“4” (no comparison supported)</p>

Value	Description
TransformOperation	<p>A transform operation type supported by the implementing system.</p> <p>The Capability Value shall be equal to the value for the Transform Operation parameter in the Transform Biometric Data service.</p> <p>The Capability Supporting Value shall specify the value of the Transform Control parameter in the Transform Biometric Data service. The value returned may be either a single value or a range of values. If a range of values is returned, the Capability Description shall specify additional information for the value of the Transform Control parameter. If the Transform Operation does not support a Transform Control, the Capability Supporting value shall be set to "NotApplicable".</p>

336 3.2.22 CapabilityType

Field	Type	#	?	Meaning
CapabilityType			Y	Defines a single capability supported by an implementing system.
↗ CapabilityName	CapabilityName	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.

337 3.2.23 CBEFF_BIR_ListType

Field	Type	#	?	Meaning
CBEFF_BIR_ListType			Y	A list of CBEFF-BIR elements.
↗ BIR	CBEFF_BIR_Type	0..*	N	CBEFF structure containing information about a biometric sample.

338 3.2.24 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.

Field	Type	#	?	Meaning
↑ FormatType	positiveInteger	1	Y	Identifies the Patron format type.
↑ 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 .

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

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

The latter two alternatives can use any CBEFF Patron Format. The optional BIR_Information provides a mechanism for exposing metadata associated with a BIR format that is not easily decoded (i.e., a non-XML BIR). See section 5.3 for more information on handling of binary data within BIAS and ISO/IEC 30108, Clause 8.2, for more information on representing biometric data.

NOTE:

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

3.2.25 Classification

Type:	string
Description:	The result of a classification.

3.2.26 ClassificationAlgorithmType

Type:	string
Description:	Type of classification algorithm that was used to perform the classification.

354 **3.2.27 ClassificationData**

Field	Type	#	?	Meaning
ClassificationData			Y	Contains information on classification results and the algorithm used to determine the classification.
↑ Classification	Classification	1	Y	The result of the classification.
↑ ClassificationAlgorithmType	ClassificationAlgorithmType	1	Y	Identifies the type of classification algorithm that was used to perform the classification.

355 **3.2.28 DocumentDataType**

Field	Type	#	?	Meaning
DocumentDataType			Y	Defines a set of document data elements providing information about the presented identity document.
↑ DocumentCategory	string	1	Y	The type of identity document presented (e.g. passport).
↑ DocumentIDNumber	string	0..1	N	The number associated with the identity document (e.g. passport number)
↑ DocumentIssuanceCountryCode	iso_3166:CountryAlpha2CodeType	0..1	N	The ISO 2-character code for the country which issued the document or from within which it was issued.
↑ DocumentIssuingOrganization	string	0..1	N	The entity which issued the identity document.

Field	Type	#	?	Meaning
\uparrow DocumentIssuanceDate	datetime	0..1	N	The date upon which the identity document was issued.
\uparrow DocumentExpirationDate	datetime	0..1	N	The date upon which the identity document is no longer valid (expires).
\uparrow DocumentLastName	string	0..1	N	The family name of the person to whom the identity document was issued, as contained within the document itself.
\uparrow DocumentFirstName	string	0..1	N	The first given name of the person to whom the identity document was issued, as contained within the document itself.
\uparrow DocumentMiddleName	string	0..1	N	The second given name of the person to whom the identity document was issued, as contained within the document itself.
\uparrow DocumentValidity	boolean	0..1	N	The assessed validity of the identity document (e.g. as the result of local or online validity checks).
\uparrow DocumentValidityText	string	0..1	N	Details or remarks associated with the assessed validity (e.g. description of validity issue).

Field	Type	#	?	Meaning
↑ DocumentImage	xsd:base64binary	0..1	N	A scanned image of the subject document (e.g. passport picture page).

356 3.2.29 DocumentDataListType

Field	Type	#	?	Meaning
DocumentDataListType			Y	Provides a list of documents.
↑ DocumentData	DocumentDataType	0..*	Y	Data structure containing information about a document and optionally an image of that document.

357 3.2.30 EncounterCategoryType

Type:	String
Description:	Identifies the type of encounter (interaction) during which the identity (biographic, biometric, and/or document) data was collected from the subject as determined by the requester.

358 EncounterCategoryType Enumeration Values

Value	Description
Enrollment	The encounter is created during an enrolment interaction.
Recognition	The encounter is created during a recognition interaction.
Unspecified	The type of encounter is unknown.

359 3.2.31 EncounterListType

Field	Type	#	?	Meaning
EncounterListType			Y	Defines a set of encounters.
↑ EncounterID	BIASIDType	0..*	N	The identifier of an encounter.

360 3.2.32 FusionDecision

Type:	string
Description:	The match decision assigned by the matching algorithm

361 3.2.33 FusionIdentityListType

Field	Type	#	?	Meaning
FusionIdentityListType			Y	Contains fusion input elements for one or more identities, utilizing the FusionInformationListType to represent a single set of fusion information for each identity.
↑ FusionIdentity	FusionInformationListType	0..*	Y	A set of fusion information for a single identity.

362 3.2.34 FusionInformationListType

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

363 3.2.35 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.
↑ 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	FusionResult	0..1	C	Either FusionScore or a FusionDecision element MUST be used.

364 3.2.36 FusionResult

Type:	complexType
Description:	The base type for any resulting types which indicate the status of a Fusion operation

365 3.2.37 FusionScore

Type:	ScoreType
--------------	---------------------------

Description:	The similarity score assigned by the matching algorithm.
---------------------	--

3.2.38 GenericRequestParameters

Field	Type	#	?	Meaning
GenericRequestParameters			Y	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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.

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

3.2.39 IdentifySubjectResultType

Description:	A base type for all types that could be returned from the IdentifySubject operation
---------------------	---

3.2.40 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	

3.2.41 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.
↑ IncludeBiometricSubType	boolean	1	Y	A Boolean flag indicating if biometric subtype information should be returned.

372 **3.2.42 MatchType**

Type:	boolean
Description:	The result of a fusion method.

373 **3.2.43 OptionType**

Field	Type	#	?	Meaning
OptionType			Y	BIAS aggregate operations support the ability to include various processing options which direct and possibly control the business logic for that operation. Together with the ProcessingOptionsType, The OptionType provides a method to represent those options. Processing options SHOULD be defined by the implementing system.
↑ Key	string	1	Y	The identifier of an option supported by the implementing system.
↑ Value	string	0..1	N	The value for an option supported by the implementing system.

374 **3.2.44 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	OptionType	0..*	N	An option supported by the implementing system.

375 **3.2.45 ProductID**

Type:	string
Description:	The vendor's ID for a particular product.

376 **3.2.46 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.

Field	Type	#	?	Meaning
AlgorithmVendor	VendorIdentifier	1	Y	The vendor of the quality algorithm used to determine the quality score.
AlgorithmVendorProductID	ProductID	1	Y	The vendor's ID for the algorithm used to determine the quality.
AlgorithmVersion	VersionType	0..1	N	The version of the algorithm used to determine the quality.

377 3.2.47 ResponseStatus

Field	Type	#	?	Meaning
ResponseStatus			Y	
Return	ReturnCode	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.

378 3.2.48 ReturnCode

Type:	unsignedLong
Description:	Return value specifying success or other condition.

379 ReturnCode Enumeration Values

Value	Description
0	Success

380 3.2.49 ScoreType

Field	Type	#	?	Meaning
ScoreType			Y	
Value	float	1	Y	Defines a match result or quality score.
BiometricType	oasis_cbeff:MultipleTypesType	0..1	N	The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.
BiometricSubType	oasis_cbeff:SubTypesType	0..1	N	More specifically defines the type of biometric data stored in the biometric record.

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

3.2.50 TokenResultType

Field	Type	#	?	Meaning
TokenResultType			Y	Defines a token that is returned for asynchronous processing.
↑ TokenType	TokenType	1	Y	Defines a token that is returned for asynchronous processing.

3.2.51 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.

NOTE: Date/time format is defined in ISO/IEC 30108 and is consistent with the date format specified in Annex B and ISO 8601 [\[DATE-TIME\]](#). See also Annex A for schema definition.

3.2.52 URI_BIR

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

3.2.53 VendorIdentifier

Type:	string
Description:	Identifies a vendor.

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

3.2.54 Version

Field	Type	#	?	Meaning
Version			Y	For a description or definition of each data element, see the referenced CBEFF standards in the 3.2.24 CBEFF_BIR_Typeschema.
↑ major	nonNegativeInteger	1	Y	
↑ minor	nonNegativeInteger	1	Y	

393 **3.2.55 VersionType**

Type:	string
Description:	The version of a component.

394 **3.2.56 XML_BIR**

Field	Type	#	?	Meaning
XML_BIR	BaseBIRType		Y	Defines a BIR type of Binary
↑ XML	Oasis_cbeff:BIRType	1	Y	BIR information in XML format

395

4 BIAS Messages

This section describes the BIAS messages implementing BIAS operations as defined in ISO/IEC 30108-1:2015. The operations are listed alphabetically, with each operation containing a request and a response message. The tables follow the conventions described in section 3.1.

4.1 Primitive Operations

4.1.1 AddSubjectToGallery

[AddSubjectToGalleryRequest](#)

[AddSubjectToGalleryResponse](#)

The AddSubjectToGallery operation registers a subject to a given gallery or population group. As an OPTIONAL parameter, the value of the claim to identity by which the subject is known to the gallery MAY be specified. This claim to identity MUST be unique across the gallery. If no claim to identity is specified, the subject ID (assigned with the [CreateSubject](#) operation) will be used as the claim to identity. In the encounter-centric model, the encounter ID associated with the subject's biometrics that will be added to the gallery MUST be specified. Additionally, the service provider implementation is responsible for the creation and management of galleries. For this purpose, services are not exposed to the requester.

Request Message

Field	Type	#	?	Meaning
AddSubjectToGallery			Y	Register a subject to a given gallery or population group.
⬆ AddSubjectToGalleryRequest		1	Y	
⬆ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⬆ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⬆ ApplicationUser	ApplicationUserIdentifier	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	BIASIDType	1	Y	The identifier of the gallery or population group to which the subject will be added.
⬆ Identity	BIASIdentity	1	Y	The identity to add to the gallery.

Field	Type	#	?	Meaning
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ IdentityClaim	BIASIDType	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	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.

412 Response Message

Field	Type	#	?	Meaning
AddSubjectToGalleryResponse			Y	The response to an AddSubjectToGallery operation.
↑ AddSubjectToGalleryResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.

413 4.1.2 CheckQuality

414 [CheckQualityRequest](#)

415 [CheckQualityResponse](#)

416 The CheckQuality operation returns a quality score for a given biometric or a specified subject. Either a
417 biometric sample or a subject ID MUST be provided. The biometric input is provided in a CBEFF basic
418 structure or CBEFF record, which in this specification is called a CBEFF-BIR. The algorithm vendor and
419 algorithm vendor product ID MAY be optionally provided in order to request a particular algorithm's use in
420 calculating the biometric quality. If an algorithm vendor is provided, then the algorithm vendor product ID
421 is REQUIRED. If no algorithm vendor is provided, the implementing system will provide the algorithm
422 vendor and algorithm vendor product ID that were used to calculate the biometric quality as output
423 parameters.

424 Algorithm Vendors are registered with the ISO Biometric Registration Authority. They are assigned unique
425 identifiers as outlined in ISO/IEC 19785-2. Algorithm Product IDs are assigned by the registered algorithm
426 vendor.

Request Message

Field	Type	#	?	Meaning
CheckQuality			Y	Calculate a quality score for a given biometric.
↑ CheckQualityRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	BIASBiometricDataType	0..1	C	Data structure containing a single biometric sample for which a quality score is to be determined; required if no Subject ID is provided.
↑ BIR	CBEFF_BIR_Type	1	Y	The biometric sample.
↑ Identity	BIASIdentity	0..1	C	Required if no BIR is provided.
↑ SubjectID	BIASIDType	1	Y	Identifies the subject; required if no BIR is provided.
↑ Qualityinfo	QualityData	0..1	N	Specifies a particular algorithm vendor and vendor product ID.

Field	Type	#	?	Meaning
AlgorithmVendor	VendorIdentifier	1	Y	The vendor of the quality algorithm used to determine the quality score.
AlgorithmVendorProductID	ProductID	1	C	The vendor assigned ID for the algorithm used to determine the quality; required as input if algorithm vendor is provided

428 Response Message

Field	Type	#	?	Meaning
CheckQualityResponse			Y	The response to a CheckQuality operation.
CheckQualityResponsePackage		1	Y	
ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
Return	ReturnCode	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	QualityData	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	VendorIdentifier	1	Y	The vendor of the quality algorithm used to determine the quality score.
AlgorithmVendorProductID	ProductID	1	Y	The vendor's ID for the algorithm used to determine the quality.
AlgorithmVersion	VersionType	1	Y	The version of the algorithm used to determine the quality.

429 4.1.3 ClassifyBiometricData

430 [ClassifyBiometricDataRequest](#)

431 [ClassifyBiometricDataResponse](#)

432 The ClassifyBiometricData operation attempts to classify a biometric sample. For example, a fingerprint
 433 biometric sample may be classified as a whorl, loop, or arch (or other classification classes and sub-
 434 classes). If no classification algorithm is input, then the BIAS service provider will make the selection.

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

436 **Request Message**

Field	Type	#	?	Meaning
ClassifyBiometricData			Y	Classifies a biometric sample.
└ ClassifyBiometricDataRequest		1	Y	
└ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
└ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
└ ApplicationUser	ApplicationUserIdentifier	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	BIASBiometricDataType	1	Y	Data structure containing a single biometric sample for which the classification is to be determined.
└ BIR	CBEFF_BIR_Type	1	Y	The biometric sample.
└ ClassificationData		0..1	N	Contains the algorithm to be used to determine the classification.
└ ClassificationAlgorithmType	ClassificationAlgorithmType	1	Y	Identifies the type of classification algorithm to be used to perform the classification.

437 Response Message

Field	Type	#	?	Meaning
ClassifyBiometricDataResponse			Y	The response to a ClassifyBiometricData operation, containing the classification of a biometric sample.
↑ ClassifyBiometricDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	ClassificationData	1	Y	Information on the results and type of classification performed.
↑ Classification	Classification	1	Y	The result of the classification.
↑ ClassificationAlgorithmType	ClassificationAlgorithmType	1	Y	Identifies the type of classification algorithm that was used to perform the classification.

438 4.1.4 CreateEncounter

439 [CreateEncounterRequest](#)

440 [CreateEncounterResponse](#)

441 The CreateEncounter operation creates a new encounter record for a subject and associates an
 442 encounter ID to that record. If an encounter ID is not supplied by the requester, the service generates an
 443 encounter ID that uniquely identifies the encounter within the subject record. The CreateEncounter
 444 operation is performed prior to a [SetBiographicData](#), [SetBiometricData](#), or [SetDocumentData](#) operation.

445 In encounter mode, for match operations, the BIAS service provider will create the encounter and will set
 446 the encounter type to "recognition". Normally the BIAS implementation will assign the encounter ID.
 447 However, if a requester assigns the encounter ID, it should be used unless it duplicates an existing
 448 encounter ID. If that happens an error should be returned.

449 Request Message

Field	Type	#	?	Meaning
CreateEncounter			Y	Create a new encounter record for the specified subject and associate an encounter ID to that record.

Field	Type	#	?	Meaning
⌞ CreateEncounterRequest		1	Y	
⌞ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⌞ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⌞ ApplicationUser	ApplicationUserIdentifier	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: "CreateEncounter".
⌞ Identity	BIASIdentity	1	Y	
⌞ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
⌞ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject. Required for encounter-centric models.
⌞ EncounterType	EncounterCategoryType	1	Y	Identifies the type of encounter during which data was collected from the subject, as determined by the requester.

450 Response Message

Field	Type	#	?	Meaning
CreateEncounterResponse			Y	The response to a CreateEncounter operation, containing the new encounterID associated with the specified subject.
⌞ CreateEncounterResponsePackage		1	Y	
⌞ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
⌞ Return	ReturnCode	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.
↑ Identity	BIASIdentity	1	Y	
↑ EncounterID	BIASIDType	1	Y	The identifier of an encounter associated with the subject. Required for encounter-centric models.

4.1.5 CreateSubject

[CreateSubjectRequest](#)

[CreateSubjectResponse](#)

The CreateSubject operation creates a new subject record and associates a subject ID to that record. As an optional parameter, the subject ID MAY be specified by the caller. If no subject ID is specified, the CreateSubject operation will generate one. UUIDs should be used for Subject IDs when universal uniqueness is required.

Request Message

Field	Type	#	?	Meaning
CreateSubject			Y	
↑ CreateSubjectRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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".

Response Message

Field	Type	#	?	Meaning
CreateSubjectResponse			Y	The response to a CreateSubject operation, containing the subject ID of the new subject record.

Field	Type	#	?	Meaning
↑ CreateSubjectResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	1	Y	
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.

4.1.6 DeleteBiographicData

[DeleteBiographicDataRequest](#)

[DeleteBiographicDataResponse](#)

The DeleteBiographicData operation erases all of the biographic data associated with a given subject record. In the encounter-centric model the operation erases all of the biographic data associated with a given encounter, and therefore the encounter ID MUST be specified. If no encounter ID is specified, or it is null, biographic data will be removed from all encounters. If a gallery is specified, biographic data will be deleted from that gallery only.

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

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	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "DeleteBiographicData".
Identity	BIASIdentity	1	Y	
SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group from which the biographic information will be deleted.

472 Response Message

Field	Type	#	?	Meaning
DeleteBiographicDataResponse			Y	The response to a DeleteBiographicData operation.
DeleteBiographicDataResponsePackage		1	Y	
ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
Return	ReturnCode	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.

473 4.1.7 DeleteBiometricData

474 [DeleteBiometricDataRequest](#)

475 [DeleteBiometricDataResponse](#)

476 The DeleteBiometricData operation erases all of the biometric data associated with a given subject
477 record. In the encounter-centric model the operation erases all of the biometric data associated with a
478 given encounter, and therefore the encounter ID MUST be specified. If no encounter ID is specified, or it
479 is null, biometric data will be removed from all encounters. If a gallery is specified, biometric data will be
480 deleted from that gallery only. If a biometric type(s) is specified, then only biometric data of that type will
481 be deleted.

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

485 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	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↳ ApplicationUser	ApplicationUserIdentifier	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	BIASIdentity	1	Y	
↳ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↳ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
↳ BiometricType	oasis_cbeff:MultipleTypesType	0..1	N	The type of biological or behavioral data to delete, as defined by CBEFF.
↳ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group from which the biometric information will be deleted.

486 Response Message

Field	Type	#	?	Meaning
DeleteBiometricDataResponse			Y	The response to a DeleteBiometricData operation.
↳ DeleteBiometricDataResponsePackage		1	Y	
↳ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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.

487 4.1.8 DeleteDocumentData

488 [DeleteDocumentDataRequest](#)

489 [DeleteDocumentDataResponse](#)

490 The DeleteDocumentData operation erases all of the document data of the specified category(ies)
 491 associated with a given subject record. In the encounter-centric model the service erases all of the
 492 document data associated with a given encounter, and therefore the encounter ID MUST be specified. If
 493 no encounter ID is specified, or it is null, document data will be removed from all encounters. If no
 494 categories are specified, then all categories (for the specified encounters) will be deleted.

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

498 Request Message

Field	Type	#	?	Meaning
DeleteDocumentData			Y	Erase all of the document data associated with a given subject record or, in the encounter-centric model, with a given encounter.
↳ DeleteDocumentDataRequest		1	Y	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↳ ApplicationUser	ApplicationUserIdentifier	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: "DeleteDocumentData".
Identity	BIASIdentity	1	Y	
SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
DocumentData	DocumentDataType	0..1	N	Defines a set of document data elements providing information about the presented identity document.
DocumentCategory	string	1	Y	The category(ies) of the identity documents to be deleted.

499 Response Message

Field	Type	#	?	Meaning
DeleteDocumentDataResponse			Y	The response to a DeleteDocumentData operation.
DeleteDocumentDataResponsePackage		1	Y	
ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
Return	ReturnCode	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.

500 4.1.9 DeleteEncounter

501 [DeleteEncounterRequest](#)

502 [DeleteEncounterResponse](#)

503 The DeleteEncounter operation deletes an existing encounter record from the system. When deleting an
504 encounter, BIAS implementations MAY completely erase the information in order to prevent the ability to

505 reconstruct a record in whole or in part, or they MAY track and record the deleted information for auditing
506 and/or quality control purposes.

507 Request Message

Field	Type	#	?	Meaning
DeleteEncounter			Y	Delete an existing encounter record and, any associated encounter information.
↳ DeleteEncounterRequest		1	Y	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↳ ApplicationUser	ApplicationUserIdentifier	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: "DeleteEncounter".
↳ Identity	BIASIdentity	1	Y	
↳ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
↳ EncounterID	BIASIDType	1	Y	The identifier of an encounter associated with the subject.

508 Response Message

Field	Type	#	?	Meaning
DeleteEncounterResponse			Y	The response to a DeleteEncounter operation.
↳ DeleteEncounterResponsePackage		1	Y	
↳ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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.

4.1.10 DeleteSubject

[DeleteSubjectRequest](#)

[DeleteSubjectResponse](#)

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

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

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	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
└ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
└ ApplicationUser	ApplicationUserIdentifier	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	BIASIdentity	1	Y	The identity of the subject to delete.
└ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.

Response Message

Field	Type	#	?	Meaning
DeleteSubjectResponse			Y	The response to a DeleteSubject operation.
└ DeleteSubjectResponsePackage		1	Y	
└ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.

Field	Type	#	?	Meaning
↑ Return	ReturnCode	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.

4.1.11 DeleteSubjectFromGallery

[DeleteSubjectFromGalleryRequest](#)

[DeleteSubjectFromGalleryResponse](#)

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

Request Message

Field	Type	#	?	Meaning
DeleteSubjectFromGallery			Y	Remove the registration of a subject from a gallery or population group.
↑ DeleteSubjectFromGalleryRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	BIASIDType	1	Y	The identifier of the gallery or population group from which the subject will be deleted.
↑ Identity	BIASIdentity	1	Y	The identity to remove from the gallery.

Field	Type	#	?	Meaning
↑ SubjectID	BIASIDType	0..1	C	A system unique identifier for a subject. Required if an Identity Claim is not provided.
↑ IdentityClaim	BIASIDType	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.

527 Response Message

Field	Type	#	?	Meaning
DeleteSubjectFromGalleryResponse			Y	The response to a DeleteSubjectFromGallery operation.
↑ DeleteSubjectFromGalleryResponsePackage				
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.

528 4.1.12 GetIdentifySubjectResults

529 [GetIdentifyResultsRequest](#)

530 [GetIdentifySubjectResultsResponse](#)

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

535 Request Message

Field	Type	#	?	Meaning
GetIdentifySubjectResults			Y	Retrieve the identification results for a specified token, which was returned by the IdentifySubject operation.
↑ GetIdentifySubjectResultsRequest		1	Y	

Field	Type	#	?	Meaning
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "GetIdentifySubjectResults".
↑ Token	TokenType	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.

536 Response Message

Field	Type	#	?	Meaning
GetIdentifySubjectResultsResponse			Y	The response to a GetIdentifySubjectResults operation, which includes a candidate list.
↑ GetIdentifySubjectResultsResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.
↑ CandidateList	CandidateListType	1	Y	A rank-ordered list of candidates that have a likelihood of matching the input biometric sample.
↑ Candidate	CandidateType	0..*	N	A single candidate.
↑ Score	ScoreType	0..1	N	The match score.
↑ BiographicData	BiographicDataType	0..1	N	Biographic data associated with the candidate match.
↑ BIRList	CBEFF_BIR_ListType	1	Y	Biometric data associated with the candidate match.
↑ BIR	CBEFF_BIR_Type	0..*	N	CBEFF structure containing information about a biometric sample.

537 4.1.13 IdentifySubject

538 [IdentifySubjectRequest](#)

539 [IdentifySubjectResponse](#)

540 The IdentifySubject operation performs an identification search against a given gallery for a given
541 biometric, returning a rank-ordered candidate list of a given maximum size. Note that multiple
542 scores/candidates is already incorporated as a score comes with a CandidateType which is a member of
543 CandidateList.

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

549 Gallery ID must not be used in conjunction with Gallery parameter. Gallery must not be used in
550 conjunction with Gallery ID parameter. However, Gallery ID or Gallery MUST be present.

551

552 Request Message

Field	Type	#	?	Meaning
IdentifySubject			Y	Perform an identification search against a given gallery for a given biometric.
↑ IdentifySubjectRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.

Field	Type	#	?	Meaning
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	BIASIDType	0..1	C	The identifier of the gallery or population group which will be searched. Must not be used in conjunction with Gallery parameter.
↑ Gallery	CandidateListType	0..1	C	A list of BIRs that must be used instead of a stored gallery. Must not be used in conjunction with GalleryID parameter.
↑ Identity	BIASIdentity	1	Y	Contains the BIR, a data structure containing the biometric sample for the search.
↑ BiometricData	BIASBiometricDataType	1	Y	An Identity's biometric data.
↑ BIR	CBEFF_BIR_Type	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.

553 Response Message

Field	Type	#	?	Meaning
IdentifySubjectResponse			Y	The response to an IdentifySubject operation, returning a rank-ordered candidate list.
↑ IdentifySubjectResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.

Field	Type	#	?	Meaning
↑ Return	ReturnCode	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	CandidateListResultType (see IdentifySubjectResultType)	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). Rank ordering is from highest to lowest match score. Returned with successful synchronous request processing.
↑ Candidate	CandidateType	0..*	N	A single candidate.
↑ Score	ScoreType	0..1	N	The match score.
BiographicData ↑	BiographicDataType	0..1	N	Biographic data associated with the candidate match.
↑ BIRList	CBEFF_BIR_ListType	1	Y	Biometric data associated with the candidate match.
BIR ↑	CBEFF_BIR_Type	0..*	N	CBEFF structure containing information about a biometric sample.
↑ Token	TokenResultType (see IdentifySubjectResultType)	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.

- 555 (1) In the event that the number of candidates exceeding the threshold exceeds the
 556 MaxListSize, the system will determine which candidate is included in the last position of
 557 the rank ordered candidate list (i.e., in the event of a tie).
 558 (2) Requesters MAY NOT change the system thresholds.

559 4.1.14 ListBiographicData

560 [ListBiographicDataRequest](#)

561 [ListBiographicDataResponse](#)

562 The ListBiographicData operation lists the biographic data elements stored for a subject using the
 563 Biographic Data Elements output parameter. Note that no actual biographic data is returned by this
 564 operation (see the [RetrieveBiographicData](#) operation to obtain the biographic data). In the encounter-
 565 centric model, an encounter ID MAY be specified to indicate that only the biographic data elements stored
 566 for that encounter should be returned. If an encounter ID is not specified and encounter data exists for the
 567 subject, the operation returns the list of encounter IDs which contain biographic data using the Encounter
 568 List output parameter, and the Biographic Data Element List output parameter is empty.

569 Request Message

Field	Type	#	?	Meaning
ListBiographicData			Y	Lists the biographic data elements stored for a subject.
⌞ ListBiographicDataRequest		1	Y	
⌞ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⌞ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⌞ ApplicationUser	ApplicationUserIdentifier	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	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter.
⌞ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
⌞ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.

Field	Type	#	?	Meaning
↑ EncounterType	EncounterCategoryType	0..1	C	Identifies the category of encounter. If an encounter ID is not specified and encounter data exists for the subject, the operation returns the list of encounter IDs of a specific type which contain biographic data using the Encounter List output parameter, and the Biographic Data Elements output parameter is empty. Should not be used in conjunction with EncounterID.

570 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	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	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.
↑ BiographicData	BiographicDataType	0..1	C	An Identity's biographic data elements that are stored in the implementing system.
BiographicDataItemList ↑	BiographicDataItemListType	0..1	N	A list of biographic data elements.
BiographicDataItem ↑	BiographicDataItemType	1..*	Y	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	EncounterListType	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.
↑ EncounterID	BIASIDType	0..*	N	The identifier of an encounter.

571 4.1.15 ListBiometricData

572 [ListBiometricDataRequest](#)

573 [ListBiometricDataResponse](#)

574 The ListBiometricData operation lists the biometric data elements stored for a subject using the Biometric
575 Data List output parameter. Note that no actual biometric data is returned by this operation (see the
576 [RetrieveBiometricData](#) operation to obtain the biometric data). In the encounter-centric model, an

577 encounter ID MAY be specified to indicate that only the biometric data elements stored for that encounter
578 should be returned. If an encounter ID is not specified and encounter data exists for the subject, the
579 operation returns the list of encounter IDs which contain biometric data using the Encounter List output
580 parameter, and the Biometric Data List output parameter is empty.

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








586 Request Message

Field	Type	#	?	Meaning
ListBiometricData			Y	Lists the biometric data elements stored for a subject.
↑ ListBiometricDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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".
↑ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.

Field	Type	#	?	Meaning
↑ EncounterType	EncounterCategoryType	0..1	C	Identifies the category of encounter. If an encounter ID is not specified and encounter data exists for the subject, the operation may return the list of encounter IDs of a specific type which contain biometric data using the Encounter List output parameter, and the Biometric Data List output parameter is empty. Should not be used in conjunction with EncounterID.
↑ ListFilterType	ListFilterType	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.

587 **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	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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	BIASIdentity	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	BIASBiometricDataType	0..1	C	An Identity's biometric data.

Field	Type	#	?	Meaning
BiometricDataList 	BiometricDataListType	0..1	N	A list of biometric data elements.
BiometricDataRecord 	BiometricDataType	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	EncounterListType	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	BIASIDType	1..*	Y	The identifier of an encounter.

4.1.16 ListDocumentData

[ListDocumentDataRequest](#)

[ListDocumentDataResponse](#)

The ListDocumentData operation lists the document categories stored for a subject using the Document Data List output parameter. Note that no other document data is returned by this operation (see the [RetrieveDocumentData](#) operation to obtain document data by category.) In the encounter-centric model, an encounter ID may be specified to indicate that only the document data elements stored for that encounter should be returned. If an encounter ID is not specified and encounter data exists for the subject, the operation must return the list of encounter IDs which contain document data using the Encounter List Output parameter, and the Document Data List output parameter must be empty.

Request Message

Field	Type	#	?	Meaning
ListDocumentData			Y	
↑ ListDocumentDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "ListDocumentData".

Field	Type	#	?	Meaning
↑ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject.
↑ EncounterType	EncounterCategoryType	0..1	C	Identifies the category of encounter. If an encounter ID is not specified and encounter data exists for the subject, the operation must return the list of encounter IDs which contain document data using the Encounter List Output parameter, and the Document Data List output parameter must be empty. Should not be used in conjunction with EncounterID.

599 Response Message

Field	Type	#	?	Meaning
ListDocumentDataResponse			Y	The response to a ListDocumentData operation.
↑ ListDocumentDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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
↑ DocumentDataList	3.2.29 DocumentDataListType	0..1	C	A list of document categories associated with a subject or encounter; non-empty if the service was successful, document data exists, and either the person-centric model is being used or the encounter-centric model is being used and an encounter identifier was specified.
↑ DocumentData	3.2.28 DocumentDataType	1..*	Y	Defines a set of document data elements providing information about the presented identity document
DocumentCategory ↑	string	1	Y	The type of identity document presented (e.g. passport).
↑ Identity	BIASIdentity	0..1	C	
↑ EncounterList	EncounterListType	1	Y	A list of encounter IDs associated with a subject and which contain document data; non-empty if the service was successful, document data exists, the encounter-centric model is being used, and an encounter identifier was not specified.
↑ EncounterID	BIASIDType	1..*	Y	The identifier of an encounter.

4.1.17 PerformFusion

[PerformFusionRequest](#)

[PerformFusionResponse](#)

The PerformFusion operation accepts either match score or match decision information and creates a fused match result. The [FusionInformationListType](#), through the [FusionInformationType](#), provides specific elements for match score input and match decision input for a single identity, while the [FusionIdentityListType](#) provides the ability to submit multiple identities to the Perform Fusion operation. The fusion method and processes are left to the implementing system.

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	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⌞ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⌞ ApplicationUser	ApplicationUserIdentifier	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	FusionIdentityListType	1	Y	
⌞ FusionIdentity	FusionInformationListType	1	Y	Score or decision input information to the fusion method for each identity.
FusionElement ⌞	FusionInformationType	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.

Field	Type	#	?	Meaning
AlgorithmType	string	1	Y	The Algorithm Owner's identifier for the specific algorithm product and version used to determine the score or decision.
FusionResult	FusionResult	0..1	C	Either FusionScore or a FusionDecision element MUST be used.

609 Response Message

Field	Type	#	?	Meaning
PerformFusionResponse			Y	The response to the PerformFusion operation.
PerformFusionResponsePackage		1	Y	
ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
Return	ReturnCode	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	MatchType	1	Y	Indicates the result of the fusion method.

610 4.1.18 QueryCapabilities

611 [QueryCapabilitiesRequest](#)

612 [QueryCapabilitiesResponse](#)

613 The QueryCapabilities operation returns a list of the capabilities, options, galleries, etc. that are supported
614 by the BIAS implementation. Refer to Annex A in the ISO/IEC BIAS standard Error! Reference source not
615 ound. for conformance requirements regarding which capability names an implementation must use in the
616 QueryCapabilities operation. If the implementing system does not support a capability item, the Capability
617 Value can be set to null in the response.

618 Proprietary and additional information may be returned by returning capabilities that are not part of those
619 capabilities enumerated in the CapabilityNames section 3.2.20. When returning capabilities not
620 enumerated in section 3.2.20, the Capability Description should describe the capability. For each
621 capability enumerated in section 3.2.20, the Capability Name should be set to the name in the value
622 column.

623 **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	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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".

624 **Response Message**

Field	Type	#	?	Meaning
QueryCapabilitiesResponse			Y	The response to a QueryCapabilities operation.
↑ QueryCapabilitiesResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	CapabilityListType	1	Y	A list of capabilities supported by the BIAS implementation.
↑ Capability	CapabilityType	0..*	N	A single capability.
↑ CapabilityName	CapabilityName	1	Y	The name of the capability.

Field	Type	#	?	Meaning
↑ 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.

4.1.19 RetrieveBiographicData

[RetrieveBiographicDataRequest](#)

[RetrieveBiographicDataResponse](#)

The RetrieveBiographicData operation retrieves the biographic data associated with a subject ID. In the encounter-centric model, the encounter ID MAY be specified and the operation will return the set of biographic data associated with that encounter (the list contains a single set). If the encounter ID is not specified in the encounter-centric model, the operation returns the list of biographic information associated with the most recent encounter. If no gallery ID is specified, a list of biographic information from all galleries will be returned.

Request Message

Field	Type	#	?	Meaning
RetrieveBiographicData			Y	Retrieves the biographic data associated with a subject ID.
↑ RetrieveBiographicDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "RetrieveBiographic Data".
↑ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.
↑ EncounterType	EncounterCategoryType	0..1	N	Identifies the type of encounter during which data was collected from the subject, as determined by the requester.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group from which the biographic information will be retrieved.

635 Response Message

Field	Type	#	?	Meaning
RetrieveBiographicDataResponse			Y	The response to a RetrieveBiographicData operation.
↑ RetrieveBiographicDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.
↑ Identity	BIASIdentity	1	Y	Includes the set of biographic data associated with a subject.
↑ BiographicDataList	BiographicDataListType	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.
↑ BiographicDataItemList	BiographicDataItemType	0..*	C	A list of biographic data elements.
↑ BiographicDataSet	BiographicDataItemType	0..1	C	A set of biographic data information.

4.1.20 RetrieveBiometricData

[RetrieveBiometricDataRequest](#)

[RetrieveBiometricDataResponse](#)

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

Request Message

Field	Type	#	?	Meaning
RetrieveBiometricData			Y	Retrieves the biometric data associated with a subject ID.
↑ RetrieveBiometricDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.

Field	Type	#	?	Meaning
↑ ApplicationUser	ApplicationUserIdentifier	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: "RetrieveBiometricData".
↑ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.
↑ EncounterType	EncounterCategoryType	0..1	N	Identifies the type of encounter during which data was collected from the subject, as determined by the requester.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group from which the biometric information will be retrieved.
↑ BiometricType	oasis_cbeff:MultipleTypesType	0..1	N	The type of biological or behavioral data to retrieve.

646 Response Message

Field	Type	#	?	Meaning
RetrieveBiometricDataResponse			Y	The response to a RetrieveBiometricData operation.
↑ RetrieveBiometricDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.
↑ Identity	BIASIdentity	1	Y	Includes the biometric data associated with a subject.
↑ BiometricData	BIASBiometricDataType	1	Y	An Identity's biometric data.
↑ BIRList	CBEFF_BIR_ListType	1	Y	A list of CBEFF-BIR elements.
↑ BIR	CBEFF_BIR_Type	0..*	N	CBEFF structure containing information about a biometric sample.

4.1.21 RetrieveDocumentData

[RetrieveDocumentDataRequest](#)

[RetrieveDocumentDataResponse](#)

The RetrieveDocumentData operation retrieves the list of document data associated with a subject ID for the category(ies) specified. In the encounter-centric model, the encounter ID MAY be specified and the operation returns the list of document data associated with that encounter. If the encounter ID is not specified in the encounter-centric model, the operation returns the list of document information associated with the most recent encounter for which document data exist. If no gallery ID is specified, document data from all galleries must be returned. If no document category is specified, all documents associated with the subject (and encounter ID, if present) must be returned.

Request Message

Field	Type	#	?	Meaning
RetrieveDocumentData			Y	
↑ RetrieveDocumentDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "RetrieveDocumentData".
↑ Identity	BIASIdentity	1	Y	

Field	Type	#	?	Meaning
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
↑ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.
↑ EncounterType	EncounterCategoryType	0..1	C	Identifies the category of encounter.
↑ DocumentData	DocumentDataType	0..1	N	Defines a set of document data elements providing information about the requested identity document.
↑ DocumentCategory	<u>string</u>	0..1	Y	The category(ies) of the identity documents to be retrieved.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group from which the biographic information will be retrieved.

659 Response Message

Field	Type	#	?	Meaning
RetrieveDocumentDataResponse			Y	The response to a RetrieveDocumentData operation.
↑ RetrieveDocumentDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.
↑ DocumentDataList	DocumentDataListType	1	Y	A list of document data associated with a subject or encounter.

660 4.1.22 SetBiographicData

661 [SetBiographicDataRequest](#)

662 [SetBiographicDataResponse](#)

663 The SetBiographicData operation associates biographic data to a given subject record. The identity
664 model of the system determines whether the biographic information should replace any existing
665 biographic information (person-centric model) or if a new encounter should be created and associated

666 with the subject (encounter-centric model). For encounter-centric models, the encounter ID MAY be
 667 specified by the caller in order to link biographic with biometric and/or document information (assuming
 668 biometric and/or document information was previously associated using the [SetBiometricData](#) and/or
 669 [SetDocumentData](#) operations). If the encounter ID is omitted for the encounter-centric model, the
 670 operation returns a system-assigned encounter ID.

671 For encounter-based systems, the Create Encounter operation should be called prior to Set Biographic
 672 Data and/or Set Document Data. The Encounter ID assigned as a result should be used as input to this
 673 operation.

674 Request Message

Field	Type	#	?	Meaning
SetBiographicData			Y	Associates biographic data to a given subject record.
⤴ SetBiographicDataRequest		1	Y	
⤴ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⤴ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⤴ ApplicationUser	ApplicationUserIdentifier	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".
⤴ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to store.
⤴ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
⤴ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject.

Field	Type	#	?	Meaning
↑ BiographicData	BiographicDataType	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.
BiographicDataItemList ↑	BiographicDataItemType	0..*	C	A list of biographic data elements.
BiographicDataSet ↑	BiographicDataSetType	0..1	C	A set of biographic data information.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group to which the biographic will be added.

675 Response Message

Field	Type	#	?	Meaning
SetBiographicDataResponse			Y	The response to a SetBiographicData operation.
↑ SetBiographicDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	0..1	C	In an encounter-centric model, identifies the encounter ID assigned to a new encounter.
↑ EncounterID	BIASIDType	1	Y	The identifier of an encounter associated with the subject.

676 4.1.23 SetBiometricData

677 [SetBiometricDataRequest](#)

678 [SetBiometricDataResponse](#)

679 The SetBiometricData operation associates biometric data to a given subject record. The identity model
 680 of the system determines whether the biometric information should replace any existing biometric
 681 information (person-centric model) or if a new encounter should be created and associated with the
 682 subject (encounter-centric model). For encounter-centric models, the encounter ID MAY be specified by
 683 the caller in order to link biometric with biographic and/or document information (assuming biographic
 684 and/or document information was previously associated using the [SetBiographicData](#) and/or
 685 [SetDocumentData](#) operation). If the encounter ID is omitted for the encounter-centric model, the
 686 operation returns a system-assigned encounter ID.

687 For encounter-based systems, the Create Encounter operation should be called prior to Set Biometric
 688 Data. The Encounter ID assigned as a result should be used as input to this operation.

689

690 **Request Message**

Field	Type	#	?	Meaning
SetBiometricData			Y	Associates biometric data to a given subject record.
↑ SetBiometricDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to store.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject.
↑ BiometricData	BIASBiometricDataType	1	Y	An Identity's biometric data.
↑ BIRList	CBEFF_BIR_ListType	1	Y	A list of CBEFF-BIR elements.

Field	Type	#	?	Meaning
BIR ↑	CBEFF_BIR_Type	1..*	Y	CBEFF structure containing information about a biometric sample.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group to which the biometric will be added.

691 Response Message

Field	Type	#	?	Meaning
SetBiometricDataResponse			Y	The response to a SetBiometricData operation.
↑ SetBiometricDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.

692 4.1.24 SetDocumentData

693 [SetDocumentDataRequest](#)

694 [SetDocumentDataResponse](#)

695 The SetDocumentData operation associates identity document data to a given subject record. The
696 identity model of the system determines whether the document information should replace any existing
697 document information for the same document category (person-centric model) or if a new encounter
698 should be created and associated with the subject (encounter-centric model). For encounter-centric
699 models, the encounter ID MAY be specified by the caller in order to link document with biographic and/or
700 biometric information (assuming biographic and/or biometric information was previously associated using
701 the [SetBiographicData](#) and/or [SetBiometricData](#) operation). If the encounter ID is omitted for the
702 encounter-centric model, the operation returns a system-assigned encounter ID.

703 Request Message

Field	Type	#	?	Meaning
SetDocumentData			Y	
↑ SetDocumentDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.

Field	Type	#	?	Meaning
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "SetDocumentData".
↑ Identity	BIASIdentity	1	Y	
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject
↑ EncounterID	BIASIDType	0..1	N	The identifier of an encounter associated with the subject.
↑ DocumentDataList	DocumentDataListType	1	Y	A list of document data to associate with the subject or encounter.
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group to which the document information will be added.

704 Response Message

Field	Type	#	?	Meaning
SetDocumentDataResponse			Y	The response to a SetDocumentData operation.
↑ SetDocumentDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	1	Y	
↑ EncounterID	BIASIDType	1	Y	The identifier of an encounter associated with the subject.

4.1.25 TransformBiometricData

[TransformBiometricDataRequest](#)

[TransformBiometricDataResponse](#)

The TransformBiometricData operation transforms or processes a given biometric in one format into a 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	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
└┐ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
└┐ ApplicationUser	ApplicationUserIdentifier	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: "TransformBiometricData".
└┐ InputBIR	CBEFF_BIR_Type	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. Note: This could be a compression ratio, target data format, etc.

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

712 Response Message

Field	Type	#	?	Meaning
TransformBiometricDataResponse			Y	The response to a TransformBiometricData operation.
⌞ TransformBiometricDataResponsePackage		1	Y	
⌞ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
⌞ Return	ReturnCode	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	CBEFF_BIR_Type	1	Y	Data structure containing the new, transformed biometric information.

713 4.1.26 UpdateBiographicData

714 [UpdateBiographicDataRequest](#)

715 [UpdateBiographicDataResponse](#)

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

719 Request Message

Field	Type	#	?	Meaning
UpdateBiographicData			Y	Updates the biographic data for a given subject record.
⌞ UpdateBiographicDataRequest		1	Y	
⌞ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
⌞ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
⌞ ApplicationUser	ApplicationUserIdentifier	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: "UpdateBiographic Data".
⌞ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to update.
⌞ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
⌞ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
⌞ BiographicData	BiographicDataType	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.
⌞ BiographicDataItemList	BiographicDataItemType	0..*	C	A list of biographic data elements.
⌞ BiographicDataSet	BiographicDataSetType	0..1	C	A set of biographic data information.

720 Response Message

Field	Type	#	?	Meaning
UpdateBiographicDataResponse			Y	The response to an UpdateBiographicData operation.
⌞ UpdateBiographicDataResponsePackage		1	Y	
⌞ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.

Field	Type	#	?	Meaning
↑ Return	ReturnCode	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.

4.1.27 UpdateBiometricData

[UpdateBiometricDataRequest](#)

[UpdateBiometricDataResponse](#)

The UpdateBiometricData operation updates the biometric data for an existing subject record. The operation includes an OPTIONAL parameter indicating if the new biometric sample should be merged with the existing biometric sample. If this parameter is set to “False” or is not used in the request, the operation replaces the existing biometric sample with the new biometric sample. The “merge” process is determined by the implementation. It may be accomplished by adding the sample to a multi-sample record or by performing some level of biometric fusion (for example, feature or sample level fusion). In the encounter-centric model, the encounter ID MUST be specified.

Request Message

Field	Type	#	?	Meaning
UpdateBiometricData			Y	Updates a single biometric sample for a given subject record.
↑ UpdateBiometricDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to update.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.

Field	Type	#	?	Meaning
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
↑ BiometricData	BIASBiometricDataType	1	Y	An Identity's biometric data.
↑ BIR	CBEFF_BIR_Type	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.

732 Response Message

Field	Type	#	?	Meaning
UpdateBiometricDataResponse			Y	The response to an UpdateBiometricData operation.
↑ UpdateBiometricDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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.

733 4.1.28 UpdateDocumentData

734 [UpdateDocumentDataRequest](#)

735 [UpdateDocumentDataResponse](#)

736 The UpdateDocumentData operation updates the document data for an existing subject record. The
737 operation replaces any existing document data of the same category with the new document data. In the
738 encounter-centric model, the encounter ID MUST be specified.

Request Message

Field	Type	#	?	Meaning
UpdateDocumentData			Y	Updates the document data for a given subject record.
↑ UpdateDocumentDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "UpdateDocument Data".
↑ Identity	BIASIdentity	1	Y	Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the document data to update.
↑ SubjectID	BIASIDType	1	Y	A system unique identifier for a subject.
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required for encounter-centric models.
↑ DocumentDataList	DocumentDataListType	1	Y	A list of updated document data.

740 Response Message

Field	Type	#	?	Meaning
UpdateDocumentDataResponse			Y	The response to an UpdateDocumentData operation.
↳ UpdateDocumentDataResponsePackage		1	Y	
↳ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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.

741 4.1.29 VerifySubject

742 [VerifySubjectRequest](#)

743 [VerifySubjectResponse](#)

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

747 In the encounter-centric model, for match operations, it is not necessary to explicitly create an encounter.

748 The BIAS service provider will create the encounter and will set the encounter type to “recognition”.


749 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	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↳ ApplicationUser	ApplicationUserIdentifier	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: "VerifySubject".
 GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group of which the subject must be a member. Required if an Identity Claim is provided.
 Identity	BIASIdentity	1	Y	Includes the identifying information and/or input and reference biometric samples.
 IdentityClaim	BIASIDType	0..1	C	A unique identifier by which a subject is known to a particular gallery or population group. (e.g. Subject ID or account number) Required if a Reference BIR is not provided.
 BiometricData	BIASBiometricDataType	1	Y	An Identity's biometric data.
 InputBIR	CBEFF_BIR_Type	1	Y	Maps to specific ISO/IEC BIAS elements as required by that specification. When multiple samples are included as input (e.g. in a multimodal operation), a complex BIR is used.
ReferenceBIR 	CBEFF_BIR_Type	0..1	C	Maps to specific ISO/IEC BIAS elements as required by that specification. Required if an Identity Claim is not provided.

750

Response Message

Field	Type	#	?	Meaning
VerifySubjectResponse			Y	The response to a VerifySubject operation.
 VerifySubjectResponsePackage		1	Y	

Field	Type	#	?	Meaning
↳ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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	Y	Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR.
↳ Score	ScoreType	0..1	N	The score if the biometric information matched.

4.2 Aggregate Operations

4.2.1 Delete

[DeleteRequest](#)

[DeleteResponse](#)

The Delete operation deletes an existing subject or, in an encounter-centric model, an existing encounter from the system. This may be accomplished in a number of different ways according to system requirements and/or resources. If the Delete operation is implemented as a synchronous service, the implementing system immediately processes the request and returns the results in the Return Data parameter. If the Delete operation is implemented as an asynchronous service, the implementing system returns a token in the Token parameter, which is an indication that the request is being handled asynchronously. In this case, the [GetDeletionResults](#) operation is used to poll for the results of the Delete request.

Request Message

Field	Type	#	?	Meaning
Delete			Y	Deletes a subject or, in an encounter-centric model, an existing encounter from the system.
↳ DeleteRequest		1	Y	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.

Field	Type	#	?	Meaning
↑ ApplicationUser	ApplicationUserIdentifier	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: "Delete".
↑ ProcessingOptions	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.
↑ InputData	InformationType	0..1	N	Contains the input data for the operation, as required by the implementing system.
↑ Identity	BIASIdentity	0..1	N	The identifier for the subject, or in encounter-centric model the encounter to be deleted; required for encounter-centric models.
↑ SubjectID	BIASIDType	0..1	C	The identifier assigned to the subject.
↑ EncounterID	BIASIDType	0..1	C	The identifier for the encounter; required for encounter-centric models.

764 Response Message

Field	Type	#	?	Meaning
DeleteResponse			Y	The response to a Delete operation.
↑ DeleteResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	1	Y	
↑ SubjectID	BIASIDType	0..1	C	The identifier assigned to the subject.

Field	Type	#	?	Meaning
↑ EncounterID	BIASIDType	0..1	C	The identifier of the encounter; required for encounter-centric models.
↑ ReturnData	InformationType	0..1	N	Contains the output data for the response.
↑ Token	TokenType	0..1	C	A token used to retrieve the results of the Delete request; returned with asynchronous request processing. If set to zero, operation is processed synchronously. If set to a non-zero value, operation is processed asynchronously and Get Deletion Results must be used to retrieve the results.
TokenValue	string	1	Y	A value returned by the implementing system that is used by Get Deletion Results to retrieve the results 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.

4.2.2 Enroll

[EnrollRequest](#)

[EnrollResponse](#)

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

If the identity model is encounter-centric, the encounter ID may optionally be specified by the caller. If the encounter ID is omitted the operation returns a system-assigned encounter ID.

Request Message

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

Field	Type	#	?	Meaning
↑ EnrollRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.
↑ InputData	InformationType	1	Y	Contains the input data for the operation, as required by the implementing system.
↑ Identity	BIASIdentity	0..1	N	
↑ EncounterID	BIASIDType	0..1	N	The identifier for the encounter; required for encounter-centric models.

779 Response Message

Field	Type	#	?	Meaning
EnrollResponse			Y	The response to an Enroll operation.
↑ EnrollResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity	1	Y	

Field	Type	#	?	Meaning
↑ SubjectID	BIASIDType	0..1	C	The identifier assigned to the subject.
↑ EncounterID	BIASIDType	0..1	C	The identifier of the encounter; required for encounter-centric models.
↑ ReturnData	InformationType	0..1	N	Contains the output data for the response.
↑ Token	TokenType	0..1	C	A token used to retrieve the results of the Enroll request; returned with asynchronous request processing. If set to zero, operation is processed synchronously. If set to a non-zero value, operation is processed asynchronously and Get Enroll Results must be used to retrieve the results.
TokenValue	string	1	Y	A value returned by the implementing system that is used by Get Enroll Results to retrieve the results 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.

4.2.3 GetDeletionResults

[GetDeletionResultsRequest](#)

[GetDeletionResultsResponse](#)

The GetDeletionResults operation retrieves the deletion results for the specified token. This operation is used in conjunction with the [Delete](#) operation. If the Delete operation is implemented as an asynchronous service, the implementing system returns a token and the GetDeletionResults operation is used to poll for the results of the original Delete request.

If the service provider implements an asynchronous Delete operation, then it MUST also implement the Get Deletion Results operation.

Request Message

Field	Type	#	?	Meaning
GetDeletionResults			Y	Retrieves the deletion results for the specified token.
↑ GetDeletionResultsRequest		1	Y	

Field	Type	#	?	Meaning
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "GetDeletionResults".
↑ Token	TokenType	1	Y	A value used to retrieve the results of the Delete 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.

790 Response Message

Field	Type	#	?	Meaning
GetDeletionResultsResponse			Y	The response to a GetDeletionResults operation.
↑ GetDeletionResultsResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.

4.2.4 GetEnrollResults

[GetEnrollResultsRequest](#)

[GetEnrollResultsResponse](#)

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

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

Request Message

Field	Type	#	?	Meaning
GetEnrollResults			Y	Retrieves the enrollment results for the specified token.
└ GetEnrollResultsRequest		1	Y	
└ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
└ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
└ ApplicationUser	ApplicationUserIdentifier	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	TokenType	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.

801 Response Message

Field	Type	#	?	Meaning
GetEnrollResultsResponse			Y	The response to a GetEnrollResults operation.
↑ GetEnrollResultsResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.
↑ Identity	BIASIdentity	1	Y	
↑ SubjectID	BIASIDType	0..1	C	The identifier assigned to the subject.
↑ EncounterID	BIASIDType	0..1	C	The identifier of the encounter; if assigned.

802 4.2.5 GetIdentifyResults

803 [GetIdentifyResultsRequest](#)

804 [GetIdentifyResultsResponse](#)

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

809 If the service provider implements an asynchronous Identify operation, then it **MUST** also implement the
810 GetIdentifyResults operation.

811 Request Message

Field	Type	#	?	Meaning
GetIdentifyResults			Y	Retrieves the identification results for the specified token
↑ GetIdentifyResultsRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.

Field	Type	#	?	Meaning
↑ ApplicationUser	ApplicationUserIdentifier	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	TokenType	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 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.

812 Response Message

Field	Type	#	?	Meaning
GetIdentifyResultsResponse			Y	The response to a GetIdentifyResults operation.
↑ GetIdentifyResultsResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	BIASIdentity			
↑ EncounterID	BIASIDType	0..1	C	The identifier of the encounter, if assigned.
↑ CandidateList	CandidateListType	0..1	C	A rank-ordered list of candidates that have a likelihood of matching the input biometric sample.

Field	Type	#	?	Meaning
↑ ReturnData	InformationType	0..1	N	Contains the output data for the response.

4.2.6 GetUpdateResults

[GetUpdateResultsRequest](#)

[GetUpdateResultsResponse](#)

The GetUpdateResults operation retrieves the update results for the specified token. This operation is used in conjunction with the [Update](#) operation. If the Update operation is implemented as an asynchronous service, the implementing system returns a token and the GetUpdateResults operation is used to poll for the results of the original Update request.

If the service provider implements an asynchronous Update operation, then it MUST also implement the GetUpdateResults operation.

Request Message

Field	Type	#	?	Meaning
GetUpdateResults			Y	Retrieves the Update results for the specified token
↑ GetUpdateResultsRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "GetUpdateResults".
↑ Token	TokenType	1	Y	A value used to retrieve the results of the Update 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

Field	Type	#	?	Meaning
				no longer guaranteed to be available.

823 Response Message

Field	Type	#	?	Meaning
GetUpdateResultsResponse			Y	The response to a GetUpdateResults operation.
↳ GetUpdateResultsResponsePackage		1	Y	
↳ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↳ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.

824 4.2.7 GetVerifyResults

825 [GetVerifyResultsRequest](#)

826 [GetVerifyResultsResponse](#)

827 The GetVerifyResults operation retrieves the verification results for the specified token. This operation is
828 used in conjunction with the [Verify](#) operation. If the Verify operation is implemented as an asynchronous
829 service, the implementing system returns a non-zero token and the GetVerifyResults operation is used to
830 poll for the results of the original Verify request.

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

833 Request Message

Field	Type	#	?	Meaning
GetVerifyResults			Y	Retrieves the verification results for the specified token
↳ GetVerifyResultsRequest		1	Y	
↳ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↳ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.

Field	Type	#	?	Meaning
↑ ApplicationUser	ApplicationUserIdentifier	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	TokenType	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.

834 Response Message

Field	Type	#	?	Meaning
GetVerifyResultsResponse			Y	The response to a GetVerifyResults operation.
↑ GetVerifyResultsResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.
↑ Identity	BIASIdentity	0..1	C	
↑ EncounterID	BIASIDType	1	Y	The identifier of the encounter, if assigned.

Field	Type	#	?	Meaning
↑ Match	boolean	1	Y	Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR.
↑ Score	ScoreType	0..1	N	The score if the biometric information matched.

4.2.8 Identify

[IdentifyRequest](#)

[IdentifyResponse](#)

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

Request Message

Field	Type	#	?	Meaning
Identify			Y	Performs an identification function.
IdentifyRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.
↑ InputData	InformationType	1	Y	Contains the input data for the aggregate services.

Field	Type	#	?	Meaning
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group which will be searched; this parameter may also be used to identify an external system where the identification request should be forwarded, if this capability is supported by the implementing system.
↑ MaxListSize	positiveInteger	1	Y	The maximum size of the candidate list that should be returned.

845 Response Message

Field	Type	#	?	Meaning
IdentifyResponse			Y	The response to an Identify operation.
↑ IdentifyResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.
↑ Identity	BIASIdentity	0..1	C	
↑ EncounterID	BIASIDType	1	Y	The identifier of the encounter, if assigned.
↑ CandidateList	CandidateListType	0..1	C	A rank-ordered list of candidates that have a likelihood of matching the input biometric sample; returned with successful, synchronous processing.

Field	Type	#	?	Meaning
↑ Token	TokenType	0..1	C	A value used to retrieve the results of the Identify request; returned with asynchronous request processing. If set to zero, operation is processed synchronously and candidate list is returned. If set to a non-zero value, operation is processed asynchronously and Get Identify Results must be used to retrieve the results.
↑ 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.

4.2.9 RetrieveData

[RetrieveDataRequest](#)

[RetrieveDataResponse](#)

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

Request Message

Field	Type	#	?	Meaning
RetrieveData			Y	Retrieves requested information about a subject or encounter.
↑ RetrieveDataRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "RetrieveData".
↑ ProcessingOptions	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed, and MAY identify what type(s) of information should be returned.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.
↑ Identity	BIASIdentity	1	Y	Includes the identifier of the subject or encounter.
↑ SubjectID	BIASIDType	0..1	C	A system unique identifier for a subject. Required if an Encounter ID is not provided.
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required if a Subject ID is not provided.

855 Response Message

Field	Type	#	?	Meaning
RetrieveDataResponse			Y	Response to a RetrieveData operation.
↑ RetrieveDataResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.

856 4.2.10 Update

857 [UpdateRequest](#)

858 [UpdateResponse](#)

859 The Update operation updates specified information about a subject, or in an encounter-centric model
 860 about an encounter. In a person-centric model, this operation can be used to update both biographic,
 861 biometric and document information for a subject record. In an encounter-centric model, this operation
 862 can be used to update biographic, biometric and/or document information for either a single encounter or
 863 all encounters. Either a subject ID or encounter ID MUST be specified.

864 Request Message

Field	Type	#	?	Meaning
Update			Y	Updates requested information about a subject or encounter.
↑ UpdateRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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: "Update".
↑ ProcessingOptions	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed, and MAY identify what type(s) of information should be returned.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.
↑ InputData	InformationType	1	Y	Contains the subject data to update.
↑ Identity	BIASIdentity	1	Y	Includes the identifier of the subject or encounter.
↑ SubjectID	BIASIDType	0..1	C	The identifier of the subject; Required if an Encounter ID is not provided.

Field	Type	#	?	Meaning
↑ EncounterID	BIASIDType	0..1	C	The identifier of an encounter associated with the subject. Required if a Subject ID is not provided.

865 Response Message

Field	Type	#	?	Meaning
UpdateResponse			Y	Response to an Update operation.
↑ UpdateResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	0..1	N	Contains the output data for the response.
↑ Token	TokenType	0..1	C	A value used to retrieve the results of the Update request; returned with asynchronous request processing. If set to zero, operation is processed synchronously. If set to a non-zero value, operation is processed asynchronously and Get Update Results must be used to retrieve the results.
↑ 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.

Field	Type	#	?	Meaning
↑ 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.

4.2.11 Verify

[VerifyRequest](#)

[VerifyResponse](#)

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

In encounter mode, for match operations, the BIAS service provider will create the encounter and will set the encounter type to "recognition". Additionally the encounter ID may optionally be specified by the caller. If the encounter ID is omitted for the encounter-centric model, the service should return a system-assigned encounter ID.

Request Message

Field	Type	#	?	Meaning
Verify			Y	Performs a 1:1 verification function.
↑ VerifyRequest		1	Y	
↑ GenericRequestParameters	GenericRequestParameters	0..1	N	Common request parameters that can be used to identify the requester.
↑ Application	ApplicationIdentifier	0..1	N	Identifies the requesting application.
↑ ApplicationUser	ApplicationUserIdentifier	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	ProcessingOptionsType	1	Y	Options that guide how the aggregate service request is processed.
↑ Option	OptionType	0..*	N	An option supported by the implementing system.

Field	Type	#	?	Meaning
↑ InputData	InformationType	1	Y	Contains the input data for the aggregate services.
↑ Identity	BIASIdentity	1	Y	Includes either the Identity Claim or Reference BIR.
↑ IdentityClaim	BIASIDType	0..1	C	An identifier by which a subject is known to a particular gallery or population group. Required if a Reference BIR is not provided.
↑ BiometricData	BIASBiometricDataType	0..1	N	An Identity's biometric data.
ReferenceBIR ↑	CBEFF_BIR_Type	0..1	C	Maps to specific ISO/IEC BIAS elements as required by that specification. Required if an Identity Claim is not provided.
↑ GalleryID	BIASIDType	0..1	C	The identifier of the gallery or population group of which the subject must be a member. Required if an Identity Claim is provided.

881 Response Message

Field	Type	#	?	Meaning
VerifyResponse			Y	The response to a Verify operation.
↑ VerifyResponsePackage		1	Y	
↑ ResponseStatus	ResponseStatus	1	Y	Returned status for the operation.
↑ Return	ReturnCode	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	InformationType	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	ScoreType	0..1	N	The score if the biometric information matched.
↑ Identity	BIASIdentity	0..1	C	
↑ EncounterID	BIASIDType	1	Y	The identifier of the encounter, if assigned.
↑ Token	TokenType	0..1	C	A value used to retrieve the results of the Verify request; returned with asynchronous request processing. If set to zero, operation is processed synchronously. If set to a non-zero value, operation is processed asynchronously and Get Verify Results must be used to retrieve the results.
↑ 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.

5 Message structure and rules

BIAS operations and data elements are defined in XML in the ISO/IEC 30108 BIAS standard. This OASIS standard further specifies the full XML schema (see Annex A) 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.

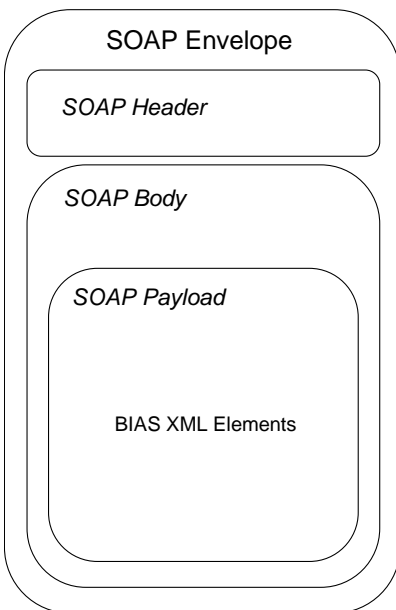


Figure 4. BIAS SOAP Structure

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 ISO/IEC 30108 are to be used as the payload of a SOAP message and the rules for structuring and exchanging such messages. Philosophical tenets include:

- SOAP messages will carry BIAS XML [XML 10] payloads.
- SOAP messages will follow WS-I and will deviate only when absolutely necessary.
- Message structures and interchanges will be kept as simple as possible – “nice to have” features will be addressed in future revisions.
- XML schemas will be produced based on ISO/IEC 30108.
- BIAS will support a broad range of application domains.
- BIAS will allow for a variety of biometric and biographic data formats to be used
- Only the SOAP messaging will be defined – no message protocols or client/server agents will be defined.
- Basic usage/formatting rules (beyond WS-I) will be defined.
- Existing biometric and Web services standards will be leveraged wherever possible.
- Sample WSDL and use cases will be provided as an aid in implementation.
- Use of basic SOAP will allow all other compatible WS* standards (and discovery mechanisms) to be used in conjunction with BIAS messaging.
- BIAS will support both secure (i.e., using existing security mechanisms such as WS-Security, SAML, etc.) and non-secure implementations.
- Generic biometric operations will be defined – use of biometrics within a Web services authentication protocol is not addressed.
- OASIS namespace rules will be followed, though some external schemas MAY also be referenced.

5.2 Message requirements

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

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

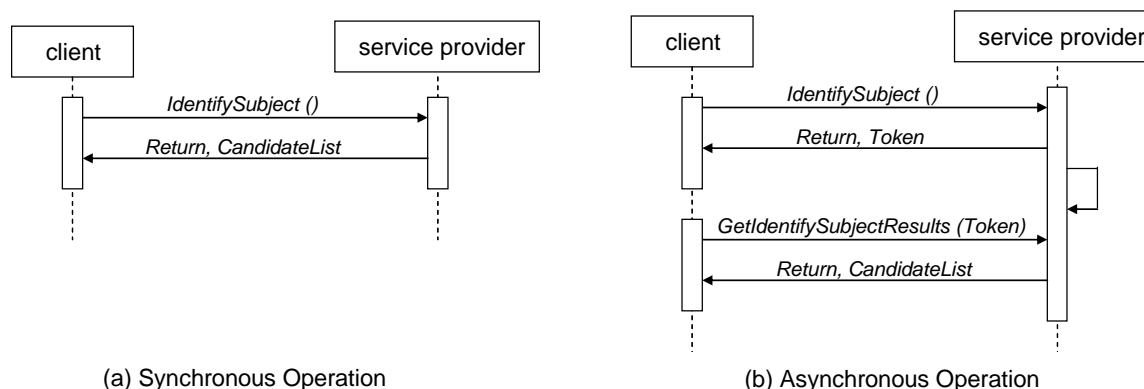


Figure 5. Example of Synchronous and Asynchronous BIAS Operations

The basic process for using SOAP for BIAS operations is:

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

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

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

5.3 Handling binary data

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

- Embedded Base64 encoding
- XOP [XOP]

Use of SOAP with Attachments (SWA) is deprecated.

5.3.1 Base64 encoding

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

As an example, the CBEFF_BIR_Type includes, as one of the BIR types, BinaryBIR of type base64binary.

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

However, even an XML_BIR as defined within [CBEFF3], contains a biometric data block (BDB) which may be entirely binary (most common),

```
<xsd
:element name="bdb" type="xsd:base64Binary" minOccurs="0"/>
```

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

5.3.2 Use of XOP

When XOP is used, the binary content is replaced with a reference (URI) to an attachment (i.e., MIME) which contains that “stripped” content via an xop:include. The advantage of this method is overall message size during transmission since the overhead of the embedded Base64 is not present (since the MIME attachment contains the native binary format).

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

5.4 Discovery

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

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

5.5 Identifying operations

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

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

5.5.1 Operation name element

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

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

```
POST /bias HTTP/1.1
Host: www.acme.com
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: ""

<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <tns:CreateSubjectRequest                                xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
      <tns:GenericRequestParameters>
        <tns:Application>BIAS Application</tns:Application>
        <tns:ApplicationUser>BIAS User</tns:ApplicationUser>
```

```

1034         <tns:BIASOperationName>CreateSubject</tns:BIASOperationName
1035     >
1036         </tns:GenericRequestParameters>
1037         <tns:Identity>
1038             <tns:SubjectID>123456789</tns:SubjectID>
1039         </tns:Identity>
1040     </tns:CreateSubjectRequest>
1041 </soap:Body>
1042 </soap:Envelope>

```

1043 5.5.2 WS-Addressing Action

1044 WS-Addressing [WS-Addr] provides a mechanism for including action information inside any SOAP
1045 message. The information is in the SOAP Header. The WS-Addressing 'Action' element is used to
1046 indicate the intent of the message. The value is a URI/IRI identifying that intent; however, there are no
1047 restrictions on the format or specificity of the URI/IRI nor a requirement that it can be resolved. Adoption
1048 of this option also requires that the WS-Addressing 'To', 'ReplyTo', and 'MessageID' elements are
1049 supplied, as they are mandatory elements in a request-reply message pattern as used within BIAS.
1050 Response messages would also need to use WS-Addressing, requiring the 'To' (matching the 'ReplyTo'
1051 element in the request), 'RelatesTo' (matching the 'MessageID' element in the request), and
1052 'RelationshipType' (default value to "wsa:Reply") elements.

1053 Use of WS-Addressing is OPTIONAL in this profile as is this method of using the 'Action' field for this
1054 purpose. However, when BIAS is used within an environment using WS-Addressing, it is
1055 RECOMMENDED that this approach for use of the 'Action' field to carry the BIAS operation name is
1056 employed, either alone or in combination with the BIASOperationName approach described in section
1057 5.5.1.

1058 An example for a message request for the CreateSubject operation would look like the following:

```

1059
1060     POST /bias HTTP/1.1
1061     Host: www.acme.com
1062     Content-Type: application/soap+xml; charset="utf-8"
1063     Content-Length: nnnn
1064     SOAPAction: ""
1065     <?xml version="1.0"?>
1066     <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
1067         xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addr
1068         essing">
1069         <soap:Header>
1070             <wsa:MessageID>some-ID</wsa:MessageID>
1071             <wsa:ReplyTo>
1072                 <wsa:Address>response-URI</wsa:Address>
1073             </wsa:ReplyTo>
1074             <wsa:To>destination-URI</wsa:To>
1075             <wsa:Action>CreateSubject</wsa:Action>
1076         </soap:Header>
1077         <soap:Body>
1078             <tns:CreateSubjectRequest

```



```

1079         xmlns:tns="http://docs.oasis-
1080         open.org/bias/bias-1.0/" />
1081         </soap:Body>
1082     </soap:Envelope>
1083

```

1084 5.6 Security

1085 The end-points that exchange SOAP messages (or handle the contents of the BIAS operations) are
1086 expected to be protected and trusted such that message-level security mechanisms may not be required.
1087 The use of SSL (HTTPS) or VPN technology that provides end-point to end-point security is
1088 RECOMMENDED and MAY be sufficient in some cases. Other mechanisms such as Signed XML or
1089 WSS [WSS] could also be implemented.

1090 Unless stated otherwise, the following security statements apply to all BIAS bindings.

1091 5.6.1 Use of SSL 3.0 or TLS 1.0

1092 Unless otherwise specified, in any BIAS binding's use of SSL 3.0 [SSL3] or TLS1.0 [RFC2246], servers
1093 MUST authenticate clients using a X.509 v3 certificate [X509]. The client MUST establish server identity
1094 based on contents of the certificate (typically through examination of the certificate's subject DN field,
1095 subjectAltName attribute, etc.).

1096 Use of transport level security in the form of SSL or TLS is OPTIONAL but highly RECOMMENDED. Use
1097 of these mechanisms alone may not be sufficient for end-to-end integrity and confidentiality, however
1098 (see 5.6.3 and 5.6.4 below).

1099 5.6.2 Data Origin Authentication

1100 Authentication of both the BIAS requester and the BIAS responder associated with a message is
1101 OPTIONAL and depends on the environment of use: Authentication mechanisms available at the SOAP
1102 message exchange layer or from the underlying substrate protocol (for example, in many bindings the
1103 SSL/TLS or HTTP protocol) MAY be utilized to provide data origin authentication.

1104 Transport authentication will not meet end-to-end origin authentication requirements in bindings where
1105 the BIAS SOAP message passes through an intermediary – in this case, message authentication is
1106 RECOMMENDED.

1107 Note that SAML [SAML] MAY be used as the mechanism for parties to authenticate to one another.

1108 5.6.3 Message Integrity

1109 Message integrity of both BIAS requests and BIAS responses is OPTIONAL and depends on the
1110 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP
1111 message exchange layer MAY be used to ensure message integrity.

1112 Transport integrity will not meet end-to-end integrity requirements in bindings where the BIAS SOAP
1113 message passes through an intermediary – in this case, message integrity is RECOMMENDED.

1114 5.6.4 Message Confidentiality

1115 Message confidentiality of both BIAS requests and BIAS responses is OPTIONAL and depends on the
1116 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP
1117 message exchange layer MAY be used to ensure message confidentiality.

1118 Transport confidentiality will not meet end-to-end confidentiality requirements in bindings where the BIAS
1119 SOAP message passes through an intermediary.

1120 NOTE: Biometric and biographic data is likely to contain personal information the confidentiality of
1121 which SHOULD be protected accordingly. See ISO/IEC 30108, section 6.5 for further discussion.

5.6.5 CBEFF BIR security features

Within BIAS, biometric data is transferred within a CBEFF BIR (as defined in ISO/IEC 19785-1). CBEFF provides for the optional encryption of the Biometric Data Block (BDB) of the BIR and for the integrity of the entire BIR. If implemented, this is indicated in the BIR header. The BIR structure defines an optional Security Block which MAY contain a digital signature (or message authentication code), encryption parameters (e.g., key name, algorithm, etc.), and/or other security related data. Such protections are associated with an individual BIR and are separate from any other protections provided at the message level.

5.6.6 Security Considerations

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

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

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

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

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

NOTE: The requirements of ISO/IEC 19092, though useful across many application domains, are required for most biometric system implementations in the financial services environment.

Application of this standard would make the requirements of sections 5.5.3 through 5.5.5 mandatory rather than optional. This is highly RECOMMENDED for any high security environment or where privacy concerns exist.

5.6.7 Security of Stored Data

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

5.6.8 Key Management

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

5.7 Use with other WS* standards

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

5.8 Tailoring

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

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

- 1167 • Identification of a subset of BIAS operations to be used
- 1168 • Specification of security features to be implemented (e.g., SSL, CBEFF BIR encryption, etc.)
- 1169 • Choice of operation name identification method
- 1170 • Choice of BIR type to be used (XML, non-XML, or URI)
- 1171 • Further definition of aggregate operations
- 1172 • Use (or not) of the encounter model
- 1173 • Use (or not) of asynchronous operations
- 1174 • Process sequences
- 1175 • Implementation specific values (e.g., Transform operations/controls)

6 Error handling

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

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

The ISO/IEC BIAS standard defines the error condition codes for system-level errors.

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

6.1 BIAS operation return codes

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

6.2 SOAP fault codes

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

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

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

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

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

7 Conformance

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

ISO/IEC 30108 Error! Reference source not found. (Annex A) specifies seven BIAS conformance classes. or each class, a set of mandatory BIAS operations is identified in order for implementations (BIAS service providers) to claim conformance. These categories are:

- Class 1: Full Primitive Services Implementation
- Class 2: Full Aggregate Services Implementation
- Class 3: Limited Primitive Services Implementation
- Class 4: Minimum Primitive Services Implementation
- Class 5: Minimum Aggregate Services Implementation
- Class 6: Matcher Primitive Services Implementation
- Class 7: Matcher Aggregate Services Implementation

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

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

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

1229 Appendix A. XML Schema

1230
1231 <?xml version="1.0" encoding="UTF-8"?>
1232 <!-- edited with XMLSpy v2016 (x64) (<http://www.altova.com>) by Karen Marshall (DoC/NIST) -->
1233 <!--
1234 Biometric Identity Assurance Services (BIAS) SOAP Profile Version 1.0
1235 OASIS Standard incorporating Approved Errata 02
1236 06 May 2014
1237 Copyright (c) OASIS Open 2014. All Rights Reserved.
1238 Source: <http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/>
1239 -->
1240 <!--
1241 Document : BIAS.xml
1242 Created on : December 17, 2009, 9:31 AM by Kevin Mangold, National Institute of Standards and Technology
1243 Last Edit : February 12, 2016 by Karen Marshall, National Institute of Standards and Technology
1244 Description:
1245 This document represents the WSDL definitions of BIAS Profile.
1246 -->
1247 <wsl:definitions xmlns:wsl="http://schemas.xmlsoap.org/wsl/"
1248 xmlns:soap="http://schemas.xmlsoap.org/wsl/soap/"
1249 xmlns:http="http://schemas.xmlsoap.org/wsl/http/"
1250 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
1251 xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
1252 xmlns:mime="http://schemas.xmlsoap.org/wsl/mime/"
1253 xmlns:oasis_cbeff="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
1254 xmlns:iso_3166="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
1255 xmlns:tns="http://docs.oasis-open.org/bias/ns/bias-1.0/"
1256 targetNamespace="http://docs.oasis-open.org/bias/ns/bias-1.0/">
1257
1258 <wsl:types>
1259 <xsd:schema elementFormDefault="qualified" targetNamespace="http://docs.oasis-open.org/bias/ns/bias-1.0/">

```

1260 <xsd:import namespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/" schemaLocation="cbeff.xsd"/>
1261 <xsd:import namespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/" schemaLocation="iso_country_codes.xsd"/>
1262 <xsd:complexType name="AggregateRequestTemplate">
1263   <xsd:annotation>
1264     <xsd:documentation>Base template for BIAS aggregate service requests.</xsd:documentation>
1265   </xsd:annotation>
1266   <xsd:complexContent>
1267     <xsd:extension base="tns:RequestTemplate">
1268       <xsd:sequence>
1269         <xsd:element name="ProcessingOptions" type="tns:ProcessingOptionsType" minOccurs="1" maxOccurs="1">
1270           <xsd:annotation>
1271             <xsd:documentation>Options that guide how the aggregate service request is processed.</xsd:documentation>
1272           </xsd:annotation>
1273         </xsd:element>
1274         <xsd:element name="InputData" type="tns:InformationType" minOccurs="1" maxOccurs="1">
1275           <xsd:annotation>
1276             <xsd:documentation>Contains the input data for the aggregate service request.</xsd:documentation>
1277           </xsd:annotation>
1278         </xsd:element>
1279       </xsd:sequence>
1280     </xsd:extension>
1281   </xsd:complexContent>
1282 </xsd:complexType>
1283 <xsd:complexType name="AggregateResponseTemplate">
1284   <xsd:annotation>
1285     <xsd:documentation>Base template for BIAS aggregate service responses.</xsd:documentation>
1286   </xsd:annotation>
1287   <xsd:complexContent>
1288     <xsd:extension base="tns:ResponseTemplate">
1289       <xsd:sequence>
1290         <xsd:element name="ReturnData" type="tns:InformationType" minOccurs="0" maxOccurs="1">
1291           <xsd:annotation>
1292             <xsd:documentation>Contains the output data for the aggregate service response.</xsd:documentation>
1293           </xsd:annotation>

```

```

1294     </xsd:element>
1295 </xsd:sequence>
1296 </xsd:extension>
1297 </xsd:complexContent>
1298 </xsd:complexType>
1299 <xsd:simpleType name="ApplicationIdentifier">
1300     <xsd:annotation>
1301         <xsd:documentation>Identifies an application.</xsd:documentation>
1302     </xsd:annotation>
1303     <xsd:restriction base="xsd:string">
1304         <xsd:minLength value="1"/>
1305         <xsd:maxLength value="255"/>
1306     </xsd:restriction>
1307 </xsd:simpleType>
1308 <xsd:simpleType name="ApplicationUserIdentifier">
1309     <xsd:annotation>
1310         <xsd:documentation>Identifies an application user or instance.</xsd:documentation>
1311     </xsd:annotation>
1312     <xsd:restriction base="xsd:string">
1313         <xsd:minLength value="1"/>
1314         <xsd:maxLength value="255"/>
1315     </xsd:restriction>
1316 </xsd:simpleType>
1317 <xsd:complexType name="BIASBiometricDataType">
1318     <xsd:annotation>
1319         <xsd:documentation>Wraps the various BIAS biometric types.</xsd:documentation>
1320     </xsd:annotation>
1321     <xsd:sequence>
1322         <xsd:element name="BIRList" type="tns:CBEFF_BIR_ListType" minOccurs="0" maxOccurs="1">
1323             <xsd:annotation>
1324                 <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
1325             </xsd:annotation>
1326         </xsd:element>
1327         <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">

```



```

1328     <xsd:annotation>
1329     <xsd:documentation>Contains biometric information in either a non-XML and an XML representation.</xsd:documentation>
1330     </xsd:annotation>
1331   </xsd:element>
1332   <xsd:element name="InputBIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
1333     <xsd:annotation>
1334     <xsd:documentation>Maps to specific INCITS BIAS elements as required by that specification.</xsd:documentation>
1335     </xsd:annotation>
1336   </xsd:element>
1337   <xsd:element name="ReferenceBIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
1338     <xsd:annotation>
1339     <xsd:documentation>Maps to specific INCITS BIAS elements as required by that specification.</xsd:documentation>
1340     </xsd:annotation>
1341   </xsd:element>
1342   <xsd:element name="BiometricDataList" type="tns:BiometricDataListType" minOccurs="0" maxOccurs="1">
1343     <xsd:annotation>
1344     <xsd:documentation>A list of biometric data elements.</xsd:documentation>
1345     </xsd:annotation>
1346   </xsd:element>
1347 </xsd:sequence>
1348 </xsd:complexType>
1349 <xsd:simpleType name="BIASFaultCode">
1350   <xsd:annotation>
1351   <xsd:documentation>Error code referenced in a SOAP fault. </xsd:documentation>
1352   </xsd:annotation>
1353   <xsd:restriction base="xsd:string">
1354     <xsd:enumeration value="UNKNOWN_ERROR">
1355       <xsd:annotation>
1356       <xsd:documentation>The service failed for an unknown reason.</xsd:documentation>
1357       </xsd:annotation>
1358     </xsd:enumeration>
1359     <xsd:enumeration value="UNSUPPORTED_CAPABILITY">
1360       <xsd:annotation>
1361       <xsd:documentation>A requested capability is not supported by the service implementation.</xsd:documentation>

```

1362 </xsd:annotation>
 1363 </xsd:enumeration>
 1364 <xsd:enumeration value="INVALID_INPUT">
 1365 <xsd:annotation>
 1366 <xsd:documentation>The data in a service input parameter is invalid.</xsd:documentation>
 1367 </xsd:annotation>
 1368 </xsd:enumeration>
 1369 <xsd:enumeration value="BIR_QUALITY_ERROR">
 1370 <xsd:annotation>
 1371 <xsd:documentation>Biometric sample quality is too poor for the service to succeed.</xsd:documentation>
 1372 </xsd:annotation>
 1373 </xsd:enumeration>
 1374 <xsd:enumeration value="INVALID_BIR">
 1375 <xsd:annotation>
 1376 <xsd:documentation>The input BIR is empty or in an invalid or unrecognized format.</xsd:documentation>
 1377 </xsd:annotation>
 1378 </xsd:enumeration>
 1379 <xsd:enumeration value="BIR_SIGNATURE_FAILURE">
 1380 <xsd:annotation>
 1381 <xsd:documentation>The service could not validate the signature, if used, on the input BIR.</xsd:documentation>
 1382 </xsd:annotation>
 1383 </xsd:enumeration>
 1384 <xsd:enumeration value="BIR_DECRYPTION_FAILURE">
 1385 <xsd:annotation>
 1386 <xsd:documentation>The service could not decrypt an encrypted input BIR.</xsd:documentation>
 1387 </xsd:annotation>
 1388 </xsd:enumeration>
 1389 <xsd:enumeration value="INVALID_ENCOUNTER_ID">
 1390 <xsd:annotation>
 1391 <xsd:documentation>The input encounter ID is empty or in an invalid format.</xsd:documentation>
 1392 </xsd:annotation>
 1393 </xsd:enumeration>
 1394 <xsd:enumeration value="INVALID_SUBJECT_ID">
 1395 <xsd:annotation>

1396 <xsd:documentation>The input subject ID is empty or in an invalid format.</xsd:documentation>
 1397 </xsd:annotation>
 1398 </xsd:enumeration>
 1399 <xsd:enumeration value="UNKNOWN_SUBJECT">
 1400 <xsd:annotation>
 1401 <xsd:documentation>The subject referenced by the input subject ID does not exist.</xsd:documentation>
 1402 </xsd:annotation>
 1403 </xsd:enumeration>
 1404 <xsd:enumeration value="UNKNOWN_GALLERY">
 1405 <xsd:annotation>
 1406 <xsd:documentation>The gallery referenced by the input gallery ID does not exist.</xsd:documentation>
 1407 </xsd:annotation>
 1408 </xsd:enumeration>
 1409 <xsd:enumeration value="UNKNOWN_ENCOUNTER">
 1410 <xsd:annotation>
 1411 <xsd:documentation>The encounter referenced by the input encounter ID does not exist.</xsd:documentation>
 1412 </xsd:annotation>
 1413 </xsd:enumeration>
 1414 <xsd:enumeration value="UNKNOWN_BIOGRAPHIC_FORMAT">
 1415 <xsd:annotation>
 1416 <xsd:documentation>The biographic data format is not known or not supported.</xsd:documentation>
 1417 </xsd:annotation>
 1418 </xsd:enumeration>
 1419 <xsd:enumeration value="UNKNOWN_IDENTITY_CLAIM">
 1420 <xsd:annotation>
 1421 <xsd:documentation>The identity referenced by the input identity claim does not exist.</xsd:documentation>
 1422 </xsd:annotation>
 1423 </xsd:enumeration>
 1424 <xsd:enumeration value="INVALID_IDENTITY_CLAIM">
 1425 <xsd:annotation>
 1426 <xsd:documentation>The identity claim requested is already in use.</xsd:documentation>
 1427 </xsd:annotation>
 1428 </xsd:enumeration>
 1429 <xsd:enumeration value="NONEXISTANT_DATA">

```

1430     <xsd:annotation>
1431     <xsd:documentation>The data requested for deletion does not exist.</xsd:documentation>
1432     </xsd:annotation>
1433   </xsd:enumeration>
1434 </xsd:restriction>
1435 </xsd:simpleType>
1436 <xsd:complexType name="BIASFaultDetail">
1437   <xsd:annotation>
1438   <xsd:documentation>Defines the error information associated with a SOAP fault.</xsd:documentation>
1439   </xsd:annotation>
1440   <xsd:sequence>
1441   <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultType" type="tns:BIASFaultCode">
1442     <xsd:annotation>
1443     <xsd:documentation>References an error code.</xsd:documentation>
1444     </xsd:annotation>
1445   </xsd:element>
1446   <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultMessage" type="xsd:string">
1447     <xsd:annotation>
1448     <xsd:documentation>Provides an explanation of the fault.</xsd:documentation>
1449     </xsd:annotation>
1450   </xsd:element>
1451   <xsd:element minOccurs="0" maxOccurs="1" name="BIASFaultDescription" type="xsd:string">
1452     <xsd:annotation>
1453     <xsd:documentation>Provides detailed information about a BIAS fault, such as trace details.</xsd:documentation>
1454     </xsd:annotation>
1455   </xsd:element>
1456   </xsd:sequence>
1457 </xsd:complexType>
1458 <xsd:complexType name="BIASIdentity">
1459   <xsd:annotation>
1460   <xsd:documentation>
1461     Defines a single element for encapsulating the data associated
1462     with an Identity. Includes the Identity's reference identifiers,
1463     biographic data, and biometric data.

```

```

1464         </xsd:documentation>
1465     </xsd:annotation>
1466 <xsd:sequence>
1467     <xsd:element name="SubjectID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1468         <xsd:annotation>
1469             <xsd:documentation>A system unique identifier for a subject.</xsd:documentation>
1470         </xsd:annotation>
1471     </xsd:element>
1472     <xsd:element name="IdentityClaim" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1473         <xsd:annotation>
1474             <xsd:documentation>An identifier by which a subject is known to a particular gallery or population group.</xsd:documentation>
1475         </xsd:annotation>
1476     </xsd:element>
1477     <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
1478         <xsd:annotation>
1479             <xsd:documentation>The identifier of an encounter associated with the subject, required for encounter-centric models.</xsd:documentation>
1480         </xsd:annotation>
1481     </xsd:element>
1482     <xsd:element name="EncounterList" type="tns:EncounterListType" minOccurs="0" maxOccurs="1">
1483         <xsd:annotation>
1484             <xsd:documentation>A list of encounters associated with a subject.</xsd:documentation>
1485         </xsd:annotation>
1486     </xsd:element>
1487     <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0" maxOccurs="1">
1488         <xsd:annotation>
1489             <xsd:documentation>An Identity's biographic data.</xsd:documentation>
1490         </xsd:annotation>
1491     </xsd:element>
1492     <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="0" maxOccurs="1">
1493         <xsd:annotation>
1494             <xsd:documentation>An Identity's biometric data.</xsd:documentation>
1495         </xsd:annotation>
1496     </xsd:element>
1497 </xsd:sequence>

```

```

1498 </xsd:complexType>
1499 <xsd:simpleType name="BIASIDType">
1500 <xsd:annotation>
1501 <xsd:documentation>A BIAS identifier</xsd:documentation>
1502 </xsd:annotation>
1503 <xsd:restriction base="xsd:string"/>
1504 </xsd:simpleType>
1505 <xsd:complexType name="BiographicDataListType">
1506 <xsd:annotation>
1507 <xsd:documentation>Defines a list of biographic data.</xsd:documentation>
1508 </xsd:annotation>
1509 <xsd:sequence>
1510 <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0" maxOccurs="unbounded">
1511 <xsd:annotation>
1512 <xsd:documentation>Data structure containing information about a biographic record.</xsd:documentation>
1513 </xsd:annotation>
1514 </xsd:element>
1515 </xsd:sequence>
1516 </xsd:complexType>
1517 <xsd:complexType name="BiographicDataItem Type">
1518 <xsd:annotation>
1519 <xsd:documentation>Defines a single biographic data element.</xsd:documentation>
1520 </xsd:annotation>
1521 <xsd:sequence>
1522 <xsd:element name="Name" type="xsd:string" minOccurs="1" maxOccurs="1">
1523 <xsd:annotation>
1524 <xsd:documentation>The name of the biographic data item.</xsd:documentation>
1525 </xsd:annotation>
1526 </xsd:element>
1527 <xsd:element name="Type" type="xsd:string" minOccurs="1" maxOccurs="1">
1528 <xsd:annotation>
1529 <xsd:documentation>The data type for the biographic data item.</xsd:documentation>
1530 </xsd:annotation>
1531 </xsd:element>

```

```

1532     <xsd:element name="Value" type="xsd:string" minOccurs="0" maxOccurs="1">
1533     <xsd:annotation>
1534     <xsd:documentation>The value assigned to the biographic data item.</xsd:documentation>
1535     </xsd:annotation>
1536     </xsd:element>
1537 </xsd:sequence>
1538 </xsd:complexType>
1539 <xsd:complexType name="BiographicDataItemListType">
1540 <xsd:annotation>
1541 <xsd:documentation>Defines a list of biographic data elements.</xsd:documentation>
1542 </xsd:annotation>
1543 <xsd:sequence>
1544 <xsd:element name="BiographicDataItem" type="tns:BiographicDataItem" minOccurs="1" maxOccurs="unbounded">
1545 <xsd:annotation>
1546 <xsd:documentation>Data structure containing information about a biographic record.</xsd:documentation>
1547 </xsd:annotation>
1548 </xsd:element>
1549 </xsd:sequence>
1550 </xsd:complexType>
1551 <xsd:complexType name="BiographicDataSetType">
1552 <xsd:annotation>
1553 <xsd:documentation>Defines a set of biographic data that is formatted according to the specified format.</xsd:documentation>
1554 </xsd:annotation>
1555 <xsd:sequence>
1556 <xsd:element name="name" type="xsd:string" minOccurs="1" maxOccurs="1">
1557 <xsd:annotation>
1558 <xsd:documentation>The name of the biographic data format. Use these names for common formats: FBI-EFTS, FBI-EBTS, DOD-EBTS, INT-I, NIEM, xNAL, HR-
1559 XML.</xsd:documentation>
1560 </xsd:annotation>
1561 </xsd:element>
1562 <xsd:element name="version" type="xsd:string" minOccurs="0" maxOccurs="1">
1563 <xsd:annotation>
1564 <xsd:documentation>The version of the biographic data format (e.g., "7.1" for FBI-EFTS or "2.0" for NIEM).</xsd:documentation>
1565 </xsd:annotation>

```

```

1566     </xsd:element>
1567     <xsd:element name="source" type="xsd:string" minOccurs="1" maxOccurs="1">
1568         <xsd:annotation>
1569             <xsd:documentation>Reference to a URI/IRI describing the biographic data format. For example: (FBI-EFTS) www.fbibiospecs.org, (DOD-EBTS) www.biometrics.dod.mil, (INT-I)
1570 www.interpol.int, (NIEM) www.niem.gov, (xNAL) www.oasis-open.org, (HR-XML) www.hr-xml.org.</xsd:documentation>
1571         </xsd:annotation>
1572     </xsd:element>
1573     <xsd:element name="type" type="xsd:string" minOccurs="1" maxOccurs="1">
1574         <xsd:annotation>
1575             <xsd:documentation>The biographic data format type. Use these types for common formats: ASCII (e.g., for non-XML versions of FBI-EFTS, FBI-EBTS, DOD-EFTS, or INT-I),
1576 XML (e.g., for NIEM, xNAL, and HR-XML or future version of FBI-EBTS).</xsd:documentation>
1577         </xsd:annotation>
1578     </xsd:element>
1579     <xsd:any namespace="##any">
1580         <xsd:annotation>
1581             <xsd:documentation>Biographic data formatted according to a specific format.</xsd:documentation>
1582         </xsd:annotation>
1583     </xsd:any>
1584 </xsd:sequence>
1585 </xsd:complexType>
1586 <xsd:complexType name="BiographicDataType">
1587     <xsd:annotation>
1588         <xsd:documentation>
1589             Defines a set of biographic data elements, utilizing either the
1590             BiographicDataItemType to represent a list of elements or the
1591             BiographicDataSetType to represent a complete, formatted set of
1592             biographic information.
1593         </xsd:documentation>
1594     </xsd:annotation>
1595 <xsd:sequence>
1596     <xsd:element name="LastName" type="xsd:string" minOccurs="0" maxOccurs="1">
1597         <xsd:annotation>
1598             <xsd:documentation>The last name of a subject.</xsd:documentation>
1599         </xsd:annotation>

```



```

1600 </xsd:element>
1601 <xsd:element name="FirstName" type="xsd:string" minOccurs="0" maxOccurs="1">
1602   <xsd:annotation>
1603     <xsd:documentation>The first name of a subject.</xsd:documentation>
1604   </xsd:annotation>
1605 </xsd:element>
1606 <xsd:element name="BiographicDataItemList" type="tns:BiographicDataItemListType" minOccurs="0" maxOccurs="1">
1607   <xsd:annotation>
1608     <xsd:documentation>A list of biographic data elements.</xsd:documentation>
1609   </xsd:annotation>
1610 </xsd:element>
1611 <xsd:element name="BiographicDataItem" type="tns:BiographicDataItemType" minOccurs="1" maxOccurs="unbounded">
1612   <xsd:annotation>
1613     <xsd:documentation>A single biographic data element.</xsd:documentation>
1614   </xsd:annotation>
1615 </xsd:element>
1616 <xsd:element name="BiographicDataSet" type="tns:BiographicDataSetType" minOccurs="0" maxOccurs="1">
1617   <xsd:annotation>
1618     <xsd:documentation>A set of biographic data information.</xsd:documentation>
1619   </xsd:annotation>
1620 </xsd:element>
1621 </xsd:sequence>
1622 </xsd:complexType>
1623 <xsd:complexType name="BiometricDataType">
1624   <xsd:annotation>
1625     <xsd:documentation>
1626       Provides descriptive information about biometric data, such as
1627       the biometric type, subtype, and format, contained in the BDB of
1628       the CBEFF-BIR.
1629     </xsd:documentation>
1630   </xsd:annotation>
1631 <xsd:sequence>
1632   <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="1" maxOccurs="1">
1633     <xsd:annotation>

```

```

1634     <xsd:documentation>The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.</xsd:documentation>
1635   </xsd:annotation>
1636 </xsd:element>
1637 <xsd:element name="BiometricTypeCount" type="xsd:positiveInteger" minOccurs="0" maxOccurs="1">
1638   <xsd:annotation>
1639     <xsd:documentation>The number of biometric records having the biometric type recorded in the biometric type field.</xsd:documentation>
1640   </xsd:annotation>
1641 </xsd:element>
1642 <xsd:element name="BiometricSubType" type="oasis_cbeff:SubtypeType" minOccurs="0" maxOccurs="1">
1643   <xsd:annotation>
1644     <xsd:documentation>More specifically defines the type of biometric data stored in the biometric record, as defined by CBEFF.</xsd:documentation>
1645   </xsd:annotation>
1646 </xsd:element>
1647 <xsd:element name="BDBFormatOwner" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
1648   <xsd:annotation>
1649     <xsd:documentation>Identifies the standards body, working group, industry consortium, or other CBEFF biometric organization that has defined the format for the biometric
1650 data.</xsd:documentation>
1651   </xsd:annotation>
1652 </xsd:element>
1653 <xsd:element name="BDBFormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
1654   <xsd:annotation>
1655     <xsd:documentation>Identifies the specific biometric data format specified by the CBEFF biometric organization recorded in the BDB Format Owner field.</xsd:documentation>
1656   </xsd:annotation>
1657 </xsd:element>
1658 </xsd:sequence>
1659 </xsd:complexType>
1660 <xsd:complexType name="BiometricDataListType">
1661   <xsd:annotation>
1662     <xsd:documentation>A list of biometric data elements.</xsd:documentation>
1663   </xsd:annotation>
1664   <xsd:sequence>
1665     <xsd:element name="BiometricData" type="tns:BiometricDataType" minOccurs="0" maxOccurs="unbounded">
1666       <xsd:annotation>
1667         <xsd:documentation>Data structure containing information about a biometric record.</xsd:documentation>

```

```

1668     </xsd:annotation>
1669 </xsd:element>
1670 </xsd:sequence>
1671 </xsd:complexType>
1672 <xsd:complexType name="CandidateListType">
1673   <xsd:annotation>
1674     <xsd:documentation>
1675       Defines a set of candidates, utilizing the Candidate Type to
1676       represent each element in the set.
1677     </xsd:documentation>
1678   </xsd:annotation>
1679   <xsd:sequence>
1680     <xsd:element name="Candidate" type="tns:CandidateType" minOccurs="0" maxOccurs="unbounded">
1681       <xsd:annotation>
1682         <xsd:documentation>A single candidate.</xsd:documentation>
1683       </xsd:annotation>
1684     </xsd:element>
1685   </xsd:sequence>
1686 </xsd:complexType>
1687 <xsd:complexType name="CandidateType">
1688   <xsd:annotation>
1689     <xsd:documentation>
1690       Defines a single candidate as a possible match in response to a
1691       biometric identification request.
1692     </xsd:documentation>
1693   </xsd:annotation>
1694   <xsd:sequence>
1695     <xsd:element name="ScoreList" minOccurs="0" >
1696       <xsd:complexType>
1697         <xsd:sequence>
1698           <xsd:element name="Score" type="tns:ScoreType"/>
1699         </xsd:sequence>
1700       </xsd:complexType>
1701     </xsd:element>

```

```

1702 <xsd:element name="Identity" type="tns:BIASIdentity" />
1703 <xsd:element name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1" >
1704 <xsd:annotation>
1705 <xsd:documentation>The rank of the candidate in relation to other candidates for the same biometric identification operation.</xsd:documentation>
1706 </xsd:annotation>
1707 </xsd:element>
1708 <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0" >
1709 <xsd:annotation>
1710 <xsd:documentation>Biographic data associated with the candidate match.</xsd:documentation>
1711 </xsd:annotation>
1712 </xsd:element>
1713 </xsd:sequence>
1714 <!--<xsd:attribute name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1" ></xsd:attribute>-->
1715 </xsd:complexType>
1716 <xsd:complexType name="CapabilityListType">
1717 <xsd:annotation>
1718 <xsd:documentation>Defines a set of capabilities.</xsd:documentation>
1719 </xsd:annotation>
1720 <xsd:sequence>
1721 <xsd:element name="Capability" type="tns:CapabilityType" minOccurs="0" maxOccurs="unbounded">
1722 <xsd:annotation>
1723 <xsd:documentation>A single capability.</xsd:documentation>
1724 </xsd:annotation>
1725 </xsd:element>
1726 </xsd:sequence>
1727 </xsd:complexType>
1728 <xsd:simpleType name="CapabilityName">
1729 <xsd:annotation>
1730 <xsd:documentation>A list of capability items.</xsd:documentation>
1731 </xsd:annotation>
1732 <xsd:restriction base="xsd:string">
1733 <xsd:enumeration value="AggregateInputDataOptional">
1734 <xsd:annotation>
1735 <xsd:documentation>A data element accepted as optional input by the implementing system for the aggregate services.</xsd:documentation>

```

1736 </xsd:annotation>
 1737 </xsd:enumeration>
 1738 <xsd:enumeration value="AggregateInputDataRequired">
 1739 <xsd:annotation>
 1740 <xsd:documentation>A data element required as input by the implementing system for the aggregate services.</xsd:documentation>
 1741 </xsd:annotation>
 1742 </xsd:enumeration>
 1743 <xsd:enumeration value="AggregateProcessingOption">
 1744 <xsd:annotation>
 1745 <xsd:documentation>A processing option supported by the implementing system for the aggregate services.</xsd:documentation>
 1746 </xsd:annotation>
 1747 </xsd:enumeration>
 1748 <xsd:enumeration value="AggregateReturnData">
 1749 <xsd:annotation>
 1750 <xsd:documentation>A data element returned by the implementing system for the aggregate services.</xsd:documentation>
 1751 </xsd:annotation>
 1752 </xsd:enumeration>
 1753 <xsd:enumeration value="AggregateServiceDescription">
 1754 <xsd:annotation>
 1755 <xsd:documentation>Describes the processing logic of an aggregate service supported by the implementing system.</xsd:documentation>
 1756 </xsd:annotation>
 1757 </xsd:enumeration>
 1758 <xsd:enumeration value="BiographicDataSet">
 1759 <xsd:annotation>
 1760 <xsd:documentation>Identifies a biographic data set supported by the implementing system.</xsd:documentation>
 1761 </xsd:annotation>
 1762 </xsd:enumeration>
 1763 <xsd:enumeration value="CBEFFPatronFormat">
 1764 <xsd:annotation>
 1765 <xsd:documentation>A patron format supported by the implementing system.</xsd:documentation>
 1766 </xsd:annotation>
 1767 </xsd:enumeration>
 1768 <xsd:enumeration value="ClassificationAlgorithmType">
 1769 <xsd:annotation>

1770 <xsd:documentation>A classification algorithm type supported by the implementing system.</xsd:documentation>
1771 </xsd:annotation>
1772 </xsd:enumeration>
1773 <xsd:enumeration value="ConformanceClass">
1774 <xsd:annotation>
1775 <xsd:documentation>Identifies the conformance class of the BIAS implementation.</xsd:documentation>
1776 </xsd:annotation>
1777 </xsd:enumeration>
1778 <xsd:enumeration value="Gallery">
1779 <xsd:annotation>
1780 <xsd:documentation>A gallery or population group supported by the implementing system.</xsd:documentation>
1781 </xsd:annotation>
1782 </xsd:enumeration>
1783 <xsd:enumeration value="IdentityModel">
1784 <xsd:annotation>
1785 <xsd:documentation>Identifies whether the implementing system is person-centric or encounter-centric based.</xsd:documentation>
1786 </xsd:annotation>
1787 </xsd:enumeration>
1788 <xsd:enumeration value="MatchAlgorithm">
1789 <xsd:annotation>
1790 <xsd:documentation>A match algorithm vendor and algorithm vendor product ID supported by the implementing system.</xsd:documentation>
1791 </xsd:annotation>
1792 </xsd:enumeration>
1793 <xsd:enumeration value="MatchScore">
1794 <xsd:annotation>
1795 <xsd:documentation> Identifies the use of match scores returned by the implementing system.</xsd:documentation>
1796 </xsd:annotation>
1797 </xsd:enumeration>
1798 <xsd:enumeration value="QualityAlgorithm">
1799 <xsd:annotation>
1800 <xsd:documentation>A quality algorithm vendor and algorithm vendor product ID supported by the implementing system.</xsd:documentation>
1801 </xsd:annotation>
1802 </xsd:enumeration>
1803 <xsd:enumeration value="SupportedBiometric">

```

1804     <xsd:annotation>
1805     <xsd:documentation>A biometric type supported by the implementing system.</xsd:documentation>
1806     </xsd:annotation>
1807 </xsd:enumeration>
1808 <xsd:enumeration value="TransformOperation">
1809     <xsd:annotation>
1810     <xsd:documentation>A transform operation type supported by the implementing system.</xsd:documentation>
1811     </xsd:annotation>
1812 </xsd:enumeration>
1813 </xsd:restriction>
1814 </xsd:simpleType>
1815 <xsd:complexType name="CapabilityType">
1816     <xsd:annotation>
1817     <xsd:documentation>Defines a single capability supported by an implementing system.</xsd:documentation>
1818     </xsd:annotation>
1819     <xsd:sequence>
1820     <xsd:element name="CapabilityName" type="tns:CapabilityName" minOccurs="1" maxOccurs="1">
1821         <xsd:annotation>
1822         <xsd:documentation>The name of the capability.</xsd:documentation>
1823         </xsd:annotation>
1824     </xsd:element>
1825     <xsd:element name="CapabilityID" type="xsd:string" minOccurs="0" maxOccurs="1">
1826         <xsd:annotation>
1827         <xsd:documentation>An identifier assigned to the capability by the implementing system.</xsd:documentation>
1828         </xsd:annotation>
1829     </xsd:element>
1830     <xsd:element name="CapabilityDescription" type="xsd:string" minOccurs="0" maxOccurs="1">
1831         <xsd:annotation>
1832         <xsd:documentation>A description of the capability.</xsd:documentation>
1833         </xsd:annotation>
1834     </xsd:element>
1835     <xsd:element name="CapabilityValue" type="xsd:string" minOccurs="0" maxOccurs="1">
1836         <xsd:annotation>
1837         <xsd:documentation>A value assigned to the capability.</xsd:documentation>

```

```

1838     </xsd:annotation>
1839 </xsd:element>
1840 <xsd:element name="CapabilitySupportingValue" type="xsd:string" minOccurs="0" maxOccurs="1">
1841     <xsd:annotation>
1842     <xsd:documentation>A secondary value supporting the capability.</xsd:documentation>
1843     </xsd:annotation>
1844 </xsd:element>
1845 <xsd:element name="CapabilityAdditionalInfo" type="xsd:string" minOccurs="0" maxOccurs="1">
1846     <xsd:annotation>
1847     <xsd:documentation> Contains additional information for the supported capability.</xsd:documentation>
1848     </xsd:annotation>
1849 </xsd:element>
1850 </xsd:sequence>
1851 </xsd:complexType>
1852 <xsd:complexType name="CBEFF_BIR_ListType">
1853     <xsd:annotation>
1854     <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
1855     </xsd:annotation>
1856 <xsd:sequence>
1857     <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="unbounded">
1858     <xsd:annotation>
1859     <xsd:documentation>CBEFF structure containing information about a biometric sample.</xsd:documentation>
1860     </xsd:annotation>
1861     </xsd:element>
1862 </xsd:sequence>
1863 </xsd:complexType>
1864 <xsd:complexType name="CBEFF_BIR_Type">
1865     <xsd:annotation>
1866     <xsd:documentation>Represents biometric information, with either a non-XML or XML representation.</xsd:documentation>
1867     </xsd:annotation>
1868 <xsd:sequence>
1869     <xsd:element name="FormatOwner" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1"/>
1870     <xsd:element name="FormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1"/>
1871     <xsd:element name="BIR_Information" minOccurs="0" maxOccurs="1">

```



```

1872 <xsd:complexType>
1873 <xsd:sequence>
1874 <xsd:element name="BIR_Info" type="oasis_cbeff:BIRInfoType" minOccurs="0" maxOccurs="1"/>
1875 <xsd:element name="BDB_Info" type="oasis_cbeff:BDBInfoType" minOccurs="0" maxOccurs="1"/>
1876 <xsd:element name="SB_Info" type="oasis_cbeff:SBInfoType" minOccurs="0" maxOccurs="1"/>
1877 </xsd:sequence>
1878 </xsd:complexType>
1879 </xsd:element>
1880 <xsd:element name="BIR" type="tns:BaseBIRType"/>
1881 </xsd:sequence>
1882 </xsd:complexType>
1883 <xsd:complexType name="BaseBIRType">
1884 </xsd:complexType>
1885 <xsd:complexType name="BinaryBIR">
1886 <xsd:complexContent>
1887 <xsd:extension base="tns:BaseBIRType">
1888 <xsd:sequence>
1889 <xsd:element name="Binary" type="xsd:base64Binary"/>
1890 </xsd:sequence>
1891 </xsd:extension>
1892 </xsd:complexContent>
1893 </xsd:complexType>
1894 <xsd:complexType name="URI_BIR">
1895 <xsd:complexContent>
1896 <xsd:extension base="tns:BaseBIRType">
1897 <xsd:sequence>
1898 <xsd:element name="URI" type="xsd:anyURI"/>
1899 </xsd:sequence>
1900 </xsd:extension>
1901 </xsd:complexContent>
1902 </xsd:complexType>
1903 <xsd:complexType name="XML_BIR">
1904 <xsd:complexContent>
1905 <xsd:extension base="tns:BaseBIRType">

```

```

1906     <xsd:sequence>
1907     <xsd:element name="XML" type="oasis_cbeff:BIRType"/>
1908     </xsd:sequence>
1909     </xsd:extension>
1910     </xsd:complexContent>
1911     </xsd:complexType>
1912     <xsd:simpleType name="Classification">
1913     <xsd:annotation>
1914     <xsd:documentation>The result of a classification.</xsd:documentation>
1915     </xsd:annotation>
1916     <xsd:restriction base="xsd:string"/>
1917     </xsd:simpleType>
1918     <xsd:simpleType name="ClassificationAlgorithmType">
1919     <xsd:annotation>
1920     <xsd:documentation>Type of classification algorithm that was used to perform the classification.</xsd:documentation>
1921     </xsd:annotation>
1922     <xsd:restriction base="xsd:string"/>
1923     </xsd:simpleType>
1924     <xsd:complexType name="ClassificationData">
1925     <xsd:annotation>
1926     <xsd:documentation>Contains information on classification results and the algorithm used to determine the classification.</xsd:documentation>
1927     </xsd:annotation>
1928     <xsd:sequence>
1929     <xsd:element name="Classification" type="tns:Classification" minOccurs="1" maxOccurs="1">
1930     <xsd:annotation>
1931     <xsd:documentation>The result of the classification.</xsd:documentation>
1932     </xsd:annotation>
1933     </xsd:element>
1934     <xsd:element name="ClassificationAlgorithmType" type="tns:ClassificationAlgorithmType" minOccurs="1" maxOccurs="1">
1935     <xsd:annotation>
1936     <xsd:documentation>Identifies the type of classification algorithm that was used to perform the classification.</xsd:documentation>
1937     </xsd:annotation>
1938     </xsd:element>
1939     </xsd:sequence>

```

1940 </xsd:complexType>

1941 <xsd:complexType name="DocumentDataType">

1942 <xsd:annotation>

1943 <xsd:documentation>Defines a set of document data elements providing information about the presented identity document. </xsd:documentation>

1944 </xsd:annotation>

1945 <xsd:sequence>

1946 <xsd:element name="DocumentCategory" type="xsd:string" minOccurs="1" maxOccurs="1">

1947 <xsd:annotation>

1948 <xsd:documentation>The type of identity document presented (e.g. passport).</xsd:documentation>

1949 </xsd:annotation>

1950 </xsd:element>

1951 <xsd:element name="DocumentIDNumber" type="xsd:string" minOccurs="0" maxOccurs="1">

1952 <xsd:annotation>

1953 <xsd:documentation>The number associated with the identity document (e.g. passport number).</xsd:documentation>

1954 </xsd:annotation>

1955 </xsd:element>

1956 <xsd:element name="DocumentIssuanceCountryCode" type="iso_3166:CountryAlpha2CodeType" minOccurs="0" maxOccurs="1">

1957 <xsd:annotation>

1958 <xsd:documentation>The ISO 2-character code for the country which issued the document or from within which it was issued.</xsd:documentation>

1959 </xsd:annotation>

1960 </xsd:element>

1961 <xsd:element name="DocumentIssuingOrganization" type="xsd:string" minOccurs="0" maxOccurs="1">

1962 <xsd:annotation>

1963 <xsd:documentation>The entity which issued the identity document.</xsd:documentation>

1964 </xsd:annotation>

1965 </xsd:element>

1966 <xsd:element name="DocumentIssuanceDate" type="xsd:dateTime" minOccurs="0" maxOccurs="1">

1967 <xsd:annotation>

1968 <xsd:documentation>The date upon which the identity document was issued.</xsd:documentation>

1969 </xsd:annotation>

1970 </xsd:element>

1971 <xsd:element name="DocumentExpirationDate" type="xsd:dateTime" minOccurs="0" maxOccurs="1">

1972 <xsd:annotation>

1973 <xsd:documentation>The date upon which the identity document is no longer valid (expires).</xsd:documentation>

1974 </xsd:annotation>
1975 </xsd:element>
1976 <xsd:element name="DocumentLastName" type="xsd:string" minOccurs="0" maxOccurs="1">
1977 <xsd:annotation>
1978 <xsd:documentation>The family name of the person to whom the identity document was issued, as contained within the document itself.</xsd:documentation>
1979 </xsd:annotation>
1980 </xsd:element>
1981 <xsd:element name="DocumentFirstName" type="xsd:string" minOccurs="0" maxOccurs="1">
1982 <xsd:annotation>
1983 <xsd:documentation>The first given name of the person to whom the identity document was issued, as contained within the document itself.</xsd:documentation>
1984 </xsd:annotation>
1985 </xsd:element>
1986 <xsd:element name="DocumentMiddleName" type="xsd:string" minOccurs="0" maxOccurs="1">
1987 <xsd:annotation>
1988 <xsd:documentation>The second given name of the person to whom the identity document was issued, as contained within the document itself.</xsd:documentation>
1989 </xsd:annotation>
1990 </xsd:element>
1991 <xsd:element name="DocumentValidity" type="xsd:boolean" minOccurs="0" maxOccurs="1">
1992 <xsd:annotation>
1993 <xsd:documentation>The assessed validity of the identity document (e.g. as the result of local or online validity checks).</xsd:documentation>
1994 </xsd:annotation>
1995 </xsd:element>
1996 <xsd:element name="DocumentValidityText" type="xsd:string" minOccurs="0" maxOccurs="1">
1997 <xsd:annotation>
1998 <xsd:documentation>Details or remarks associated with the assessed validity (e.g. description of validity issue).</xsd:documentation>
1999 </xsd:annotation>
2000 </xsd:element>
2001 <xsd:element name="DocumentImage" type="xsd:base64Binary" minOccurs="0" maxOccurs="1">
2002 <xsd:annotation>
2003 <xsd:documentation>A scanned image of the subject document (e.g. passport picture page).</xsd:documentation>
2004 </xsd:annotation>
2005 </xsd:element>
2006 </xsd:sequence>
2007 </xsd:complexType>

```

2008 <xsd:complexType name="DocumentDataListType">
2009 <xsd:annotation>
2010 <xsd:documentation>Defines a list of documents. </xsd:documentation>
2011 </xsd:annotation>
2012 <xsd:sequence>
2013 <xsd:element name="DocumentData" type="tns:DocumentDataType" minOccurs="0" maxOccurs="unbounded">
2014 <xsd:annotation>
2015 <xsd:documentation>Data structure containing information about a document and optionally an image of that document.</xsd:documentation>
2016 </xsd:annotation>
2017 </xsd:element>
2018 </xsd:sequence>
2019 </xsd:complexType>
2020 <xsd:simpleType name="EncounterCategoryType">
2021 <xsd:annotation>
2022 <xsd:documentation>Identifies the type of encounter (interaction) during which the identity (biographic, biometric, and/or document) data was collected from the subject as
2023 determined by the requester.</xsd:documentation>
2024 </xsd:annotation>
2025 <xsd:restriction base="xsd:string">
2026 <xsd:enumeration value="Enrolment">
2027 <xsd:annotation>
2028 <xsd:documentation>The encounter is created during an enrolment interaction.</xsd:documentation>
2029 </xsd:annotation>
2030 </xsd:enumeration>
2031 <xsd:enumeration value="Recognition">
2032 <xsd:annotation>
2033 <xsd:documentation>The encounter is created during a recognition interaction.</xsd:documentation>
2034 </xsd:annotation>
2035 </xsd:enumeration>
2036 <xsd:enumeration value="Unspecified">
2037 <xsd:annotation>
2038 <xsd:documentation>The type of encounter is unknown.</xsd:documentation>
2039 </xsd:annotation>
2040 </xsd:enumeration>
2041 <xsd:pattern value="([a-zA-Z0-9])+"/>

```

```

2042     </xsd:restriction>
2043 </xsd:simpleType>
2044 <xsd:complexType name="EncounterListType">
2045     <xsd:annotation>
2046         <xsd:documentation>Defines a set of encounters.</xsd:documentation>
2047     </xsd:annotation>
2048     <xsd:sequence>
2049         <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0" maxOccurs="unbounded">
2050             <xsd:annotation>
2051                 <xsd:documentation>The identifier of an encounter.</xsd:documentation>
2052             </xsd:annotation>
2053         </xsd:element>
2054     </xsd:sequence>
2055 </xsd:complexType>
2056 <xsd:complexType name="FusionIdentityListType">
2057     <xsd:annotation>
2058         <xsd:documentation>
2059             Contains fusion input elements for one or more identities, utilizing the FusionInformationListType to represent a single set of fusion information for each identity.
2060         </xsd:documentation>
2061     </xsd:annotation>
2062     <xsd:sequence>
2063         <xsd:element name="FusionIdentity" type="tns:FusionInformationListType" minOccurs="0" maxOccurs="unbounded">
2064             <xsd:annotation>
2065                 <xsd:documentation>A set of fusion information for a single identity.</xsd:documentation>
2066             </xsd:annotation>
2067         </xsd:element>
2068     </xsd:sequence>
2069 </xsd:complexType>
2070 <xsd:complexType name="FusionInformationListType">
2071     <xsd:annotation>
2072         <xsd:documentation>
2073             Contains at a minimum two sets of fusion input
2074             elements, as input to the PerformFusion request.
2075         </xsd:documentation>

```

```

2076 </xsd:annotation>
2077 <xsd:sequence>
2078 <xsd:element name="FusionElement" type="tns:FusionInformationType" minOccurs="2" maxOccurs="unbounded">
2079 <xsd:annotation>
2080 <xsd:documentation>A set of fusion information.</xsd:documentation>
2081 </xsd:annotation>
2082 </xsd:element>
2083 </xsd:sequence>
2084 </xsd:complexType>
2085 <xsd:complexType name="FusionInformationType">
2086 <xsd:annotation>
2087 <xsd:documentation>Represents the information necessary to perform a fusion operation.</xsd:documentation>
2088 </xsd:annotation>
2089 <xsd:sequence>
2090 <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="1" maxOccurs="1">
2091 <xsd:annotation>
2092 <xsd:documentation>The type of biological or behavioral data stored in the biometric record, as defined by CBEFF.</xsd:documentation>
2093 </xsd:annotation>
2094 </xsd:element>
2095 <xsd:element name="BiometricSubType" type="oasis_cbeff:SubtypeType" minOccurs="0" maxOccurs="1">
2096 <xsd:annotation>
2097 <xsd:documentation>More specifically defines the type of biometric data stored in the biometric record.</xsd:documentation>
2098 </xsd:annotation>
2099 </xsd:element>
2100 <xsd:element name="AlgorithmOwner" type="xsd:string" minOccurs="1" maxOccurs="1">
2101 <xsd:annotation>
2102 <xsd:documentation>The owner or vendor of the algorithm used to determine the score or decision.</xsd:documentation>
2103 </xsd:annotation>
2104 </xsd:element>
2105 <xsd:element name="AlgorithmType" type="xsd:string" minOccurs="1" maxOccurs="1">
2106 <xsd:annotation>
2107 <xsd:documentation>The Algorithm Owner's identifier for the specific algorithm product and version used to determine the score or decision.</xsd:documentation>
2108 </xsd:annotation>
2109 </xsd:element>

```

```

2110     <xsd:element name="FusionResult" type="tns:FusionResult"/>
2111 </xsd:sequence>
2112 </xsd:complexType>
2113 <xsd:complexType name="FusionResult">
2114     <xsd:sequence/>
2115 </xsd:complexType>
2116 <xsd:complexType name="FusionScore">
2117     <xsd:complexContent>
2118         <xsd:extension base="tns:FusionResult">
2119             <xsd:sequence>
2120                 <xsd:element name="Score" type="tns:ScoreType">
2121                     <xsd:annotation>
2122                         <xsd:documentation>The similarity score assigned by the matching algorithm.</xsd:documentation>
2123                     </xsd:annotation>
2124                 </xsd:element>
2125             </xsd:sequence>
2126         </xsd:extension>
2127     </xsd:complexContent>
2128 </xsd:complexType>
2129 <xsd:complexType name="FusionDecision">
2130     <xsd:complexContent>
2131         <xsd:extension base="tns:FusionResult">
2132             <xsd:sequence>
2133                 <xsd:element name="Decision" type="xsd:string">
2134                     <xsd:annotation>
2135                         <xsd:documentation>The match decision assigned by the matching algorithm.</xsd:documentation>
2136                     </xsd:annotation>
2137                 </xsd:element>
2138             </xsd:sequence>
2139         </xsd:extension>
2140     </xsd:complexContent>
2141 </xsd:complexType>
2142 <xsd:complexType name="GenericRequestParameters">
2143     <xsd:annotation>

```



```

2144     <xsd:documentation>Common request parameters that can be used to identify the requester.</xsd:documentation>
2145 </xsd:annotation>
2146 <xsd:sequence>
2147   <xsd:element name="Application" type="tns:ApplicationIdentifier" minOccurs="0" maxOccurs="1">
2148     <xsd:annotation>
2149       <xsd:documentation>Identifies the requesting application.</xsd:documentation>
2150     </xsd:annotation>
2151   </xsd:element>
2152   <xsd:element name="ApplicationUser" type="tns:ApplicationUserIdentifier" minOccurs="0" maxOccurs="1">
2153     <xsd:annotation>
2154       <xsd:documentation>Identifiers the user or instance of the requesting application.</xsd:documentation>
2155     </xsd:annotation>
2156   </xsd:element>
2157   <xsd:element name="BIASOperationName" type="xsd:string" minOccurs="0" maxOccurs="1">
2158     <xsd:annotation>
2159       <xsd:documentation>Identifiers the BIAS operation name that is being requested.</xsd:documentation>
2160     </xsd:annotation>
2161   </xsd:element>
2162 </xsd:sequence>
2163 </xsd:complexType>
2164 <xsd:complexType name="InformationType">
2165   <xsd:annotation>
2166     <xsd:documentation>
2167       Allows for an unlimited number of data element types, and it does
2168       not specify nor require any particular data element.
2169     </xsd:documentation>
2170   </xsd:annotation>
2171   <!-- THE FOLLOWING COMMENTED SECTION IS PROVIDED AS AN INFORMATIVE
2172     EXAMPLE OF WHAT 'InformationType'S CONTENT COULD BE -->
2173   <!-- xsd:sequence>
2174     <xsd:element name="GUID" type="xsd:string" minOccurs="0" />
2175     <xsd:element name="GivenName" type="xsd:string" minOccurs="0" />
2176     <xsd:element name="FamilyName" type="xsd:string" minOccurs="0" />
2177     <xsd:element name="DateOfBirth" type="xsd:string" minOccurs="0" />

```

```

2178 <xsd:element name="Sex" type="xsd:string" minOccurs="0" />
2179 <xsd:element name="Citizenship" type="xsd:string" minOccurs="0" />
2180 <xsd:element name="Images" minOccurs="0">
2181 <xsd:complexType>
2182 <xsd:sequence>
2183 <xsd:element name="Image" minOccurs="0" maxOccurs="unbounded">
2184 <xsd:complexType>
2185 <xsd:sequence>
2186 <xsd:element name="ContentType" type="xsd:string" minOccurs="0" />
2187 <xsd:element name="FingerPosition" type="xsd:string" minOccurs="0" />
2188 <xsd:element name="HandPosition" type="xsd:string" minOccurs="0" />
2189 <xsd:element name="ImageData" type="xsd:base64Binary" minOccurs="0" />
2190 </xsd:sequence>
2191 </xsd:complexType>
2192 </xsd:element>
2193 </xsd:sequence>
2194 </xsd:complexType>
2195 </xsd:element>
2196 <xsd:element name="Identities" minOccurs="0">
2197 <xsd:complexType>
2198 <xsd:sequence>
2199 <xsd:element name="Identity" type="xsd:string" maxOccurs="unbounded" />
2200 </xsd:sequence>
2201 </xsd:complexType>
2202 </xsd:element>
2203 </xsd:sequence-->
2204 </xsd:complexType>
2205 <xsd:complexType name="ListFilterType">
2206 <xsd:annotation>
2207 <xsd:documentation>Provides a method to filter the amount of information returned in a search of biometric data.
2208 </xsd:documentation>
2209 </xsd:annotation>
2210 <xsd:sequence>
2211 <xsd:element name="BiometricTypeFilters">

```

```

2212     <xsd:complexType>
2213     <xsd:sequence>
2214     <xsd:element name="BiometricTypeFilter" type="oasis_cbeff:MultipleTypesType" minOccurs="1" maxOccurs="unbounded">
2215     <xsd:annotation>
2216     <xsd:documentation>Limits the returned information to a specific type of biometric, as defined by CBEFF.</xsd:documentation>
2217     </xsd:annotation>
2218     </xsd:element>
2219     </xsd:sequence>
2220 </xsd:complexType>
2221 </xsd:element>
2222 <xsd:element name="IncludeBiometricSubtype" type="xsd:boolean" minOccurs="1" maxOccurs="1">
2223 <xsd:annotation>
2224 <xsd:documentation>A Boolean flag indicating if biometric subtype information should be returned.</xsd:documentation>
2225 </xsd:annotation>
2226 </xsd:element>
2227 </xsd:sequence>
2228 </xsd:complexType>
2229 <xsd:simpleType name="MatchType">
2230 <xsd:annotation>
2231 <xsd:documentation>The result of a fusion method.</xsd:documentation>
2232 </xsd:annotation>
2233 <xsd:restriction base="xsd:boolean"/>
2234 </xsd:simpleType>
2235 <xsd:complexType name="OptionType">
2236 <xsd:annotation>
2237 <xsd:documentation>
2238     BIAS aggregate operations support the ability to include various processing options which direct and possibly control the business logic for that operation. Together with the
2239     ProcessingOptionsType, The OptionType provides a method to represent those options. Processing options SHOULD be defined by the implementing system.
2240 </xsd:documentation>
2241 </xsd:annotation>
2242 <xsd:sequence>
2243 <xsd:element name="Key" type="xsd:string" >
2244 <xsd:annotation>
2245 <xsd:documentation>The identifier of an option supported by the implementing system.</xsd:documentation>

```

```

2246     </xsd:annotation>
2247   </xsd:element>
2248   <xsd:element name="Value" type="xsd:string" minOccurs="0" >
2249     <xsd:annotation>
2250       <xsd:documentation>The value for an option supported by the implementing system.</xsd:documentation>
2251     </xsd:annotation>
2252   </xsd:element>
2253 </xsd:sequence>
2254 </xsd:complexType>
2255
2256 <xsd:complexType name="ProcessingOptionsType">
2257   <xsd:annotation>
2258     <xsd:documentation>
2259       BIAS aggregate services support the ability to include various
2260       processing options which direct and possibly control the business
2261       logic for that service. The ProcessingOptionsType provides a
2262       method to represent those options. Processing options should be
2263       defined by the implementing system.
2264     </xsd:documentation>
2265   </xsd:annotation>
2266   <xsd:sequence>
2267     <xsd:element name="Option" type="tns:OptionType" minOccurs="0" maxOccurs="unbounded">
2268       <xsd:annotation>
2269         <xsd:documentation>An option supported by the implementing system.</xsd:documentation>
2270       </xsd:annotation>
2271     </xsd:element>
2272   </xsd:sequence>
2273 </xsd:complexType>
2274 <xsd:simpleType name="ProductID">
2275   <xsd:annotation>
2276     <xsd:documentation>The vendor's ID for a particular product.</xsd:documentation>
2277   </xsd:annotation>
2278   <xsd:restriction base="xsd:string"/>
2279 </xsd:simpleType>

```

```

2280 <xsd:complexType name="QualityData">
2281   <xsd:annotation>
2282     <xsd:documentation>Contains information about a biometric sample's quality and the algorithm used to compute the quality.</xsd:documentation>
2283   </xsd:annotation>
2284   <xsd:sequence>
2285     <xsd:element name="QualityScore" type="oasis_cbeff:QualityType" minOccurs="0" maxOccurs="1">
2286       <xsd:annotation>
2287         <xsd:documentation>The quality of a biometric sample.</xsd:documentation>
2288       </xsd:annotation>
2289     </xsd:element>
2290     <xsd:element name="AlgorithmVendor" type="tns:VendorIdentifier" minOccurs="1" maxOccurs="1">
2291       <xsd:annotation>
2292         <xsd:documentation>The vendor of the quality algorithm used to determine the quality score.</xsd:documentation>
2293       </xsd:annotation>
2294     </xsd:element>
2295     <xsd:element name="AlgorithmVendorProductID" type="tns:ProductID" minOccurs="1" maxOccurs="1">
2296       <xsd:annotation>
2297         <xsd:documentation>The vendor's ID for the algorithm used to determine the quality.</xsd:documentation>
2298       </xsd:annotation>
2299     </xsd:element>
2300     <xsd:element name="AlgorithmVersion" type="tns:VersionType" minOccurs="0" maxOccurs="1">
2301       <xsd:annotation>
2302         <xsd:documentation>The version of the algorithm used to determine the quality.</xsd:documentation>
2303       </xsd:annotation>
2304     </xsd:element>
2305   </xsd:sequence>
2306 </xsd:complexType>
2307 <xsd:complexType name="RequestTemplate">
2308   <xsd:annotation>
2309     <xsd:documentation>Base template for BIAS primitive service requests.</xsd:documentation>
2310   </xsd:annotation>
2311   <xsd:sequence>
2312     <xsd:element name="GenericRequestParameters" type="tns:GenericRequestParameters" minOccurs="0" maxOccurs="1"/>
2313   </xsd:sequence>

```

```

2314 </xsd:complexType>
2315 <xsd:complexType name="ResponseStatus">
2316   <xsd:sequence>
2317     <xsd:element name="Return" type="tns:ReturnCode" minOccurs="1" maxOccurs="1">
2318       <xsd:annotation>
2319         <xsd:documentation>The return code indicates the return status of the operation.</xsd:documentation>
2320       </xsd:annotation>
2321     </xsd:element>
2322     <xsd:element name="Message" type="xsd:string" minOccurs="0" maxOccurs="1">
2323       <xsd:annotation>
2324         <xsd:documentation>A short message corresponding to the return code.</xsd:documentation>
2325       </xsd:annotation>
2326     </xsd:element>
2327   </xsd:sequence>
2328 </xsd:complexType>
2329 <xsd:complexType name="ResponseTemplate">
2330   <xsd:annotation>
2331     <xsd:documentation>Base template for BIAS responses.</xsd:documentation>
2332   </xsd:annotation>
2333   <xsd:sequence>
2334     <xsd:element name="ResponseStatus" type="tns:ResponseStatus" minOccurs="1" maxOccurs="1">
2335       <xsd:annotation>
2336         <xsd:documentation>Returned status for the operation.</xsd:documentation>
2337       </xsd:annotation>
2338     </xsd:element>
2339   </xsd:sequence>
2340 </xsd:complexType>
2341 <xsd:simpleType name="ReturnCode">
2342   <xsd:annotation>
2343     <xsd:documentation>BIAS Operation Return Codes</xsd:documentation>
2344   </xsd:annotation>
2345   <xsd:restriction base="xsd:unsignedLong">
2346     <xsd:enumeration value="0">
2347     <xsd:annotation>

```

2348 <xsd:documentation>Success</xsd:documentation>
 2349 </xsd:annotation>
 2350 </xsd:enumeration>
 2351 </xsd:restriction>
 2352 </xsd:simpleType>
 2353 <xsd:complexType name="ScoreType">
 2354 <xsd:annotation>
 2355 <xsd:documentation>Match result or quality score.</xsd:documentation>
 2356 </xsd:annotation>
 2357 <xsd:sequence>
 2358 <xsd:element name="Value" type="xsd:float"/>
 2359 <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="0"/>
 2360 <xsd:element name="BiometricSubType" type="oasis_cbeff:SubtypeType" minOccurs="0"/>
 2361 </xsd:sequence>
 2362 </xsd:complexType>
 2363 <xsd:complexType name="TokenType">
 2364 <xsd:annotation>
 2365 <xsd:documentation>Defines a token that is returned for asynchronous processing.</xsd:documentation>
 2366 </xsd:annotation>
 2367 <xsd:sequence>
 2368 <xsd:element name="TokenValue" type="xsd:string" minOccurs="1" maxOccurs="1">
 2369 <xsd:annotation>
 2370 <xsd:documentation>A value returned by the implementing system that is used to retrieve the results to a service at a later time.</xsd:documentation>
 2371 </xsd:annotation>
 2372 </xsd:element>
 2373 <xsd:element name="Expiration" type="xsd:date" minOccurs="1" maxOccurs="1">
 2374 <xsd:annotation>
 2375 <xsd:documentation>A date and time at which point the token expires and the service results are no longer guaranteed to be available.</xsd:documentation>
 2376 </xsd:annotation>
 2377 </xsd:element>
 2378 </xsd:sequence>
 2379 </xsd:complexType>
 2380 <xsd:simpleType name="VendorIdentifier">
 2381 <xsd:annotation>

2382 <xsd:documentation>Identifies a vendor.</xsd:documentation>
 2383 </xsd:annotation>
 2384 <xsd:restriction base="xsd:string"/>
 2385 </xsd:simpleType>
 2386 <xsd:complexType name="Version">
 2387 <xsd:annotation>
 2388 <xsd:documentation>
 2389 For a description or definition of each data element, see the
 2390 referenced CBEFF standards in the CBEFF_XML_BIR_Type schema.
 2391 </xsd:documentation>
 2392 </xsd:annotation>
 2393 <xsd:sequence>
 2394 <xsd:element name="major" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
 2395 <xsd:element name="minor" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
 2396 </xsd:sequence>
 2397 </xsd:complexType>
 2398 <xsd:simpleType name="VersionType">
 2399 <xsd:annotation>
 2400 <xsd:documentation>The version of a component.</xsd:documentation>
 2401 </xsd:annotation>
 2402 <xsd:restriction base="xsd:string"/>
 2403 </xsd:simpleType>
 2404 <xsd:element name="AddSubjectToGallery">
 2405 <xsd:complexType>
 2406 <xsd:annotation>
 2407 <xsd:documentation>
 2408 Register a subject to a given gallery or population group.
 2409 </xsd:documentation>
 2410 </xsd:annotation>
 2411 <xsd:sequence>
 2412 <xsd:element name="AddSubjectToGalleryRequest" nillable="true">
 2413 <xsd:complexType>
 2414 <xsd:complexContent>
 2415 <xsd:extension base="tns:RequestTemplate">


```

2416     <xsd:sequence>
2417     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="1" maxOccurs="1">
2418     <xsd:annotation>
2419     <xsd:documentation>The identifier of the gallery or population group to which the subject will be added.</xsd:documentation>
2420     </xsd:annotation>
2421     </xsd:element>
2422     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2423     <xsd:annotation>
2424     <xsd:documentation>The identity to add to the gallery.</xsd:documentation>
2425     </xsd:annotation>
2426     </xsd:element>
2427     </xsd:sequence>
2428     </xsd:extension>
2429     </xsd:complexContent>
2430     </xsd:complexType>
2431     </xsd:element>
2432     </xsd:sequence>
2433     </xsd:complexType>
2434     </xsd:element>
2435     <xsd:element name="AddSubjectToGalleryResponse">
2436     <xsd:complexType>
2437     <xsd:annotation>
2438     <xsd:documentation>
2439     The response to an AddSubjectToGallery request.
2440     </xsd:documentation>
2441     </xsd:annotation>
2442     <xsd:sequence>
2443     <xsd:element name="AddSubjectToGalleryResponsePackage" nillable="true">
2444     <xsd:complexType>
2445     <xsd:complexContent>
2446     <xsd:extension base="tns:ResponseTemplate"/>
2447     </xsd:complexContent>
2448     </xsd:complexType>
2449     </xsd:element>

```

```

2450     </xsd:sequence>
2451 </xsd:complexType>
2452 </xsd:element>
2453 <xsd:element name="CheckQuality">
2454     <xsd:complexType>
2455         <xsd:annotation>
2456             <xsd:documentation>
2457                 Calculate a quality score for a given biometric.
2458             </xsd:documentation>
2459         </xsd:annotation>
2460     <xsd:sequence>
2461         <xsd:element name="CheckQualityRequest" nillable="true">
2462             <xsd:complexType>
2463                 <xsd:complexContent>
2464                     <xsd:extension base="tns:RequestTemplate">
2465                         <xsd:sequence>
2466                             <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="0" maxOccurs="1">
2467                                 <xsd:annotation>
2468                                     <xsd:documentation>Data structure containing a single biometric sample for which a quality score is to be determined; required if no SubjectID is
2469 provided.</xsd:documentation>
2470                                 </xsd:annotation>
2471                             </xsd:element>
2472                             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
2473                                 <xsd:annotation>
2474                                     <xsd:documentation>Data structure containing a Subject ID associated with a single biometric sample for which a quality score is to be determined; required if no BIR is
2475 provided.</xsd:documentation>
2476                                 </xsd:annotation>
2477                             </xsd:element>
2478                             <xsd:element name="QualityInfo" type="tns:QualityData" minOccurs="0" maxOccurs="1">
2479                                 <xsd:annotation>
2480                                     <xsd:documentation>Specifies a particular algorithm vendor and vender product ID.</xsd:documentation>
2481                                 </xsd:annotation>
2482                             </xsd:element>
2483                         </xsd:sequence>

```

```

2484     </xsd:extension>
2485     </xsd:complexContent>
2486   </xsd:complexType>
2487 </xsd:element>
2488 </xsd:sequence>
2489 </xsd:complexType>
2490 </xsd:element>
2491 <xsd:element name="CheckQualityResponse">
2492   <xsd:complexType>
2493     <xsd:annotation>
2494       <xsd:documentation>
2495         The response to a CheckQuality request.
2496       </xsd:documentation>
2497     </xsd:annotation>
2498     <xsd:sequence>
2499       <xsd:element name="CheckQualityResponsePackage" nillable="true">
2500         <xsd:complexType>
2501           <xsd:complexContent>
2502             <xsd:extension base="tns:ResponseTemplate">
2503               <xsd:sequence>
2504                 <xsd:element name="QualityInfo" type="tns:QualityData" minOccurs="1" maxOccurs="1">
2505                   <xsd:annotation>
2506                     <xsd:documentation>Contains the quality information for the submitted biometric sample.</xsd:documentation>
2507                   </xsd:annotation>
2508                 </xsd:element>
2509               </xsd:sequence>
2510             </xsd:extension>
2511           </xsd:complexContent>
2512         </xsd:complexType>
2513       </xsd:element>
2514     </xsd:sequence>
2515   </xsd:complexType>
2516 </xsd:element>
2517 <xsd:element name="ClassifyBiometricData">

```

```

2518 <xsd:complexType>
2519 <xsd:annotation>
2520 <xsd:documentation>
2521   Classifies a biometric sample.
2522 </xsd:documentation>
2523 </xsd:annotation>
2524 <xsd:sequence>
2525 <xsd:element name="ClassifyBiometricDataRequest" nillable="true">
2526 <xsd:complexType>
2527 <xsd:complexContent>
2528 <xsd:extension base="tns:RequestTemplate">
2529 <xsd:sequence>
2530 <xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="1" maxOccurs="1">
2531 <xsd:annotation>
2532 <xsd:documentation>Data structure containing a single biometric sample for which the classification is to be determined.</xsd:documentation>
2533 </xsd:annotation>
2534 </xsd:element>
2535 <xsd:element name="ClassificationData" type="tns:ClassificationData" minOccurs="0" maxOccurs="1">
2536 <xsd:annotation>
2537 <xsd:documentation>Contains the algorithm to be used to determine the classification.</xsd:documentation>
2538 </xsd:annotation>
2539 </xsd:element>
2540 </xsd:sequence>
2541 </xsd:extension>
2542 </xsd:complexContent>
2543 </xsd:complexType>
2544 </xsd:element>
2545 </xsd:sequence>
2546 </xsd:complexType>
2547 </xsd:element>
2548 <xsd:element name="ClassifyBiometricDataResponse">
2549 <xsd:complexType>
2550 <xsd:annotation>
2551 <xsd:documentation>

```

2552 The response to a ClassifyBiometricData request, containing
2553 the classification of a biometric sample.
2554 </xsd:documentation>
2555 </xsd:annotation>
2556 <xsd:sequence>
2557 <xsd:element name="ClassifyBiometricDataResponsePackage" nillable="true">
2558 <xsd:complexType>
2559 <xsd:complexContent>
2560 <xsd:extension base="tns:ResponseTemplate">
2561 <xsd:sequence>
2562 <xsd:element name="ClassificationData" type="tns:ClassificationData" minOccurs="1" maxOccurs="1">
2563 <xsd:annotation>
2564 <xsd:documentation>Information on the results and type of classification performed.</xsd:documentation>
2565 </xsd:annotation>
2566 </xsd:element>
2567 </xsd:sequence>
2568 </xsd:extension>
2569 </xsd:complexContent>
2570 </xsd:complexType>
2571 </xsd:element>
2572 </xsd:sequence>
2573 </xsd:complexType>
2574 </xsd:element>
2575 <xsd:element name="CreateEncounter">
2576 <xsd:complexType>
2577 <xsd:annotation>
2578 <xsd:documentation>
2579 Create a new encounter record.
2580 </xsd:documentation>
2581 </xsd:annotation>
2582 <xsd:sequence>
2583 <xsd:element name="CreateEncounterRequest" nillable="true">
2584 <xsd:complexType>
2585 <xsd:complexContent>

```

2586     <xsd:extension base="tns:RequestTemplate"/>
2587   </xsd:complexContent>
2588 </xsd:complexType>
2589 </xsd:element>
2590 </xsd:sequence>
2591 </xsd:complexType>
2592 </xsd:element>
2593 <xsd:element name="CreateEncounterResponse">
2594   <xsd:complexType>
2595     <xsd:annotation>
2596       <xsd:documentation>
2597         The response to a CreateEncounter request, containing the encounter
2598         ID of the new encounter record.
2599       </xsd:documentation>
2600     </xsd:annotation>
2601     <xsd:sequence>
2602       <xsd:element name="CreateEncounterResponsePackage" nillable="true">
2603         <xsd:complexType>
2604           <xsd:complexContent>
2605             <xsd:extension base="tns:ResponseTemplate">
2606               <xsd:sequence>
2607                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2608                   <xsd:annotation>
2609                     <xsd:documentation>Contains the encounter ID of the new encounter record.</xsd:documentation>
2610                   </xsd:annotation>
2611                 </xsd:element>
2612               </xsd:sequence>
2613             </xsd:extension>
2614           </xsd:complexContent>
2615         </xsd:complexType>
2616       </xsd:element>
2617     </xsd:sequence>
2618   </xsd:complexType>
2619 </xsd:element>

```

```

2620 <xsd:element name="CreateSubject">
2621   <xsd:complexType>
2622     <xsd:annotation>
2623       <xsd:documentation>
2624         Create a new subject record.
2625       </xsd:documentation>
2626     </xsd:annotation>
2627     <xsd:sequence>
2628       <xsd:element name="CreateSubjectRequest" nillable="true">
2629         <xsd:complexType>
2630           <xsd:complexContent>
2631             <xsd:extension base="tns:RequestTemplate"/>
2632           </xsd:complexContent>
2633         </xsd:complexType>
2634       </xsd:element>
2635     </xsd:sequence>
2636   </xsd:complexType>
2637 </xsd:element>
2638 <xsd:element name="CreateSubjectResponse">
2639   <xsd:complexType>
2640     <xsd:annotation>
2641       <xsd:documentation>
2642         The response to a CreateSubject request, containing the subject
2643         ID of the new subject record.
2644       </xsd:documentation>
2645     </xsd:annotation>
2646     <xsd:sequence>
2647       <xsd:element name="CreateSubjectResponsePackage" nillable="true">
2648         <xsd:complexType>
2649           <xsd:complexContent>
2650             <xsd:extension base="tns:ResponseTemplate">
2651               <xsd:sequence>
2652                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2653                   <xsd:annotation>

```

```

2654      <xsd:documentation>Contains the subject ID of the new subject record.</xsd:documentation>
2655      </xsd:annotation>
2656    </xsd:element>
2657  </xsd:sequence>
2658 </xsd:extension>
2659 </xsd:complexContent>
2660 </xsd:complexType>
2661 </xsd:element>
2662 </xsd:sequence>
2663 </xsd:complexType>
2664 </xsd:element>
2665 <xsd:element name="DeleteBiographicData">
2666   <xsd:complexType>
2667     <xsd:annotation>
2668       <xsd:documentation>
2669         Erase all of the biographic data associated with a given
2670         subject record or, in the encounter-centric model, with a
2671         given encounter.
2672       </xsd:documentation>
2673     </xsd:annotation>
2674     <xsd:sequence>
2675       <xsd:element name="DeleteBiographicDataRequest" nillable="true">
2676         <xsd:complexType>
2677           <xsd:complexContent>
2678             <xsd:extension base="tns:RequestTemplate">
2679               <xsd:sequence>
2680                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2681                   <xsd:annotation>
2682                     <xsd:documentation>Contains either the subject ID or encounter ID reference.</xsd:documentation>
2683                   </xsd:annotation>
2684                 </xsd:element>
2685                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
2686                   <xsd:annotation>
2687                     <xsd:documentation>The identifier of the gallery or population group from which the biographic information will be deleted.</xsd:documentation>

```



```

2688     </xsd:annotation>
2689     </xsd:element>
2690   </xsd:sequence>
2691 </xsd:extension>
2692 </xsd:complexContent>
2693 </xsd:complexType>
2694 </xsd:element>
2695 </xsd:sequence>
2696 </xsd:complexType>
2697 </xsd:element>
2698 <xsd:element name="DeleteBiographicDataResponse">
2699   <xsd:complexType>
2700     <xsd:annotation>
2701       <xsd:documentation>
2702         The response to a DeleteBiographicData request.
2703       </xsd:documentation>
2704     </xsd:annotation>
2705     <xsd:sequence>
2706       <xsd:element name="DeleteBiographicDataResponsePackage" nillable="true">
2707         <xsd:complexType>
2708           <xsd:complexContent>
2709             <xsd:extension base="tns:ResponseTemplate"/>
2710           </xsd:complexContent>
2711         </xsd:complexType>
2712       </xsd:element>
2713     </xsd:sequence>
2714   </xsd:complexType>
2715 </xsd:element>
2716 <xsd:element name="DeleteBiometricData">
2717   <xsd:complexType>
2718     <xsd:annotation>
2719       <xsd:documentation>
2720         Erase all of the biometric data associated with a given
2721         subject record or, in the encounter-centric model, with a

```

```

2722     given encounter.
2723   </xsd:documentation>
2724 </xsd:annotation>
2725 <xsd:sequence>
2726   <xsd:element name="DeleteBiometricDataRequest" nillable="true">
2727     <xsd:complexType>
2728       <xsd:complexContent>
2729         <xsd:extension base="tns:RequestTemplate">
2730           <xsd:sequence>
2731             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2732               <xsd:annotation>
2733                 <xsd:documentation>Contains either the subject ID or encounter ID reference.</xsd:documentation>
2734               </xsd:annotation>
2735             </xsd:element>
2736             <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="0" maxOccurs="1">
2737               <xsd:annotation>
2738                 <xsd:documentation>The type of biological or behavioral data to delete, as defined by CBEFF.</xsd:documentation>
2739               </xsd:annotation>
2740             </xsd:element>
2741             <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
2742               <xsd:annotation>
2743                 <xsd:documentation>The identifier of the gallery or population group from which the biometric information will be deleted.</xsd:documentation>
2744               </xsd:annotation>
2745             </xsd:element>
2746           </xsd:sequence>
2747         </xsd:extension>
2748       </xsd:complexContent>
2749     </xsd:complexType>
2750   </xsd:element>
2751 </xsd:sequence>
2752 </xsd:complexType>
2753 </xsd:element>
2754 <xsd:element name="DeleteBiometricDataResponse">
2755   <xsd:complexType>

```

```

2756 <xsd:annotation>
2757 <xsd:documentation>
2758   The response to a DeleteBiometricData request.
2759 </xsd:documentation>
2760 </xsd:annotation>
2761 <xsd:sequence>
2762 <xsd:element name="DeleteBiometricDataResponsePackage" nillable="true">
2763 <xsd:complexType>
2764 <xsd:complexContent>
2765 <xsd:extension base="tns:ResponseTemplate"/>
2766 </xsd:complexContent>
2767 </xsd:complexType>
2768 </xsd:element>
2769 </xsd:sequence>
2770 </xsd:complexType>
2771 </xsd:element>
2772 <xsd:element name="DeleteDocumentData">
2773 <xsd:complexType>
2774 <xsd:annotation>
2775 <xsd:documentation>Erase all of the document data associated with a given subject record or, in the encounter-centric model, with a given encounter.</xsd:documentation>
2776 </xsd:annotation>
2777 <xsd:sequence>
2778 <xsd:element name="DeleteDocumentDataRequest" nillable="true">
2779 <xsd:complexType>
2780 <xsd:complexContent>
2781 <xsd:extension base="tns:RequestTemplate">
2782 <xsd:sequence>
2783 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2784 <xsd:annotation>
2785 <xsd:documentation>Contains either the subject ID or encounter ID reference.</xsd:documentation>
2786 </xsd:annotation>
2787 </xsd:element>
2788 <xsd:element name="DocumentData" type="tns:DocumentDataType" minOccurs="0" maxOccurs="1">
2789 <xsd:annotation>

```

2790 <xsd:documentation>Defines a set of document data elements providing information about the presented identity document. </xsd:documentation>
 2791 </xsd:annotation>
 2792 </xsd:element>
 2793 </xsd:sequence>
 2794 </xsd:extension>
 2795 </xsd:complexContent>
 2796 </xsd:complexType>
 2797 </xsd:element>
 2798 </xsd:sequence>
 2799 </xsd:complexType>
 2800 </xsd:element>
 2801 <xsd:element name="DeleteDocumentDataResponse">
 2802 <xsd:complexType>
 2803 <xsd:annotation>
 2804 <xsd:documentation>
 2805 The response to a DeleteDocumentData request.
 2806 </xsd:documentation>
 2807 </xsd:annotation>
 2808 <xsd:sequence>
 2809 <xsd:element name="DeleteDocumentDataResponsePackage" nillable="true">
 2810 <xsd:complexType>
 2811 <xsd:complexContent>
 2812 <xsd:extension base="tns:ResponseTemplate"/>
 2813 </xsd:complexContent>
 2814 </xsd:complexType>
 2815 </xsd:element>
 2816 </xsd:sequence>
 2817 </xsd:complexType>
 2818 </xsd:element>
 2819 <xsd:element name="DeleteEncounter">
 2820 <xsd:complexType>
 2821 <xsd:annotation>
 2822 <xsd:documentation>
 2823 Delete an existing encounter record from the system.

```

2824     </xsd:documentation>
2825 </xsd:annotation>
2826 <xsd:sequence>
2827   <xsd:element name="DeleteEncounterRequest" nillable="true">
2828     <xsd:complexType>
2829       <xsd:complexContent>
2830         <xsd:extension base="tns:RequestTemplate">
2831           <xsd:sequence>
2832             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2833               <xsd:annotation>
2834                 <xsd:documentation>Subject ID and encounter ID of the identity to delete.</xsd:documentation>
2835               </xsd:annotation>
2836             </xsd:element>
2837           </xsd:sequence>
2838         </xsd:extension>
2839       </xsd:complexContent>
2840     </xsd:complexType>
2841   </xsd:element>
2842 </xsd:sequence>
2843 </xsd:complexType>
2844 </xsd:element>
2845 <xsd:element name="DeleteEncounterResponse">
2846   <xsd:complexType>
2847     <xsd:annotation>
2848       <xsd:documentation>
2849         The response to a DeleteEncounter request.
2850       </xsd:documentation>
2851     </xsd:annotation>
2852     <xsd:sequence>
2853       <xsd:element name="DeleteEncounterResponsePackage" nillable="true">
2854         <xsd:complexType>
2855           <xsd:complexContent>
2856             <xsd:extension base="tns:ResponseTemplate"/>
2857           </xsd:complexContent>

```

```

2858     </xsd:complexType>
2859 </xsd:element>
2860 </xsd:sequence>
2861 </xsd:complexType>
2862 </xsd:element>
2863 <xsd:element name="DeleteSubject">
2864   <xsd:complexType>
2865     <xsd:annotation>
2866       <xsd:documentation>
2867         Delete an existing subject record and, in an encounter-centric
2868         model, any associated encounter information.
2869       </xsd:documentation>
2870     </xsd:annotation>
2871     <xsd:sequence>
2872       <xsd:element name="DeleteSubjectRequest" nillable="true">
2873         <xsd:complexType>
2874           <xsd:complexContent>
2875             <xsd:extension base="tns:RequestTemplate">
2876               <xsd:sequence>
2877                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2878                   <xsd:annotation>
2879                     <xsd:documentation>Subject ID of the identity to delete.</xsd:documentation>
2880                   </xsd:annotation>
2881                 </xsd:element>
2882               </xsd:sequence>
2883             </xsd:extension>
2884           </xsd:complexContent>
2885         </xsd:complexType>
2886       </xsd:element>
2887     </xsd:sequence>
2888   </xsd:complexType>
2889 </xsd:element>
2890 <xsd:element name="DeleteSubjectResponse">
2891   <xsd:complexType>

```

```

2892 <xsd:annotation>
2893 <xsd:documentation>
2894   The response to a DeleteSubject request.
2895 </xsd:documentation>
2896 </xsd:annotation>
2897 <xsd:sequence>
2898 <xsd:element name="DeleteSubjectResponsePackage" nillable="true">
2899   <xsd:complexType>
2900     <xsd:complexContent>
2901       <xsd:extension base="tns:ResponseTemplate"/>
2902     </xsd:complexContent>
2903   </xsd:complexType>
2904 </xsd:element>
2905 </xsd:sequence>
2906 </xsd:complexType>
2907 </xsd:element>
2908 <xsd:element name="DeleteSubjectFromGallery">
2909   <xsd:complexType>
2910     <xsd:annotation>
2911       <xsd:documentation>
2912         Remove the registration of a subject from a gallery or
2913         population group.
2914       </xsd:documentation>
2915     </xsd:annotation>
2916     <xsd:sequence>
2917       <xsd:element name="DeleteSubjectFromGalleryRequest" nillable="true">
2918         <xsd:complexType>
2919           <xsd:complexContent>
2920             <xsd:extension base="tns:RequestTemplate">
2921               <xsd:sequence>
2922                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="1" maxOccurs="1">
2923                   <xsd:annotation>
2924                     <xsd:documentation>The identifier of the gallery or population group from which the subject will be deleted.</xsd:documentation>
2925                   </xsd:annotation>

```

```

2926     </xsd:element>
2927     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
2928         <xsd:annotation>
2929             <xsd:documentation>The identity to remove from the gallery.</xsd:documentation>
2930         </xsd:annotation>
2931     </xsd:element>
2932 </xsd:sequence>
2933 </xsd:extension>
2934 </xsd:complexContent>
2935 </xsd:complexType>
2936 </xsd:element>
2937 </xsd:sequence>
2938 </xsd:complexType>
2939 </xsd:element>
2940 <xsd:element name="DeleteSubjectFromGalleryResponse">
2941     <xsd:complexType>
2942         <xsd:annotation>
2943             <xsd:documentation>
2944                 The response to a DeleteSubjectFromGallery request.
2945             </xsd:documentation>
2946         </xsd:annotation>
2947         <xsd:sequence>
2948             <xsd:element name="DeleteSubjectFromGalleryResponsePackage" nillable="true">
2949                 <xsd:complexType>
2950                     <xsd:complexContent>
2951                         <xsd:extension base="tns:ResponseTemplate"/>
2952                     </xsd:complexContent>
2953                 </xsd:complexType>
2954             </xsd:element>
2955         </xsd:sequence>
2956     </xsd:complexType>
2957 </xsd:element>
2958 <xsd:element name="GetIdentifySubjectResults">
2959     <xsd:complexType>

```



```

2960 <xsd:annotation>
2961 <xsd:documentation>
2962 Retrieve the identification results for a specified token,
2963 which was returned by the Identify Subject service.
2964 </xsd:documentation>
2965 </xsd:annotation>
2966 <xsd:sequence>
2967 <xsd:element name="GetIdentifySubjectResultsRequest" nillable="true">
2968 <xsd:complexType>
2969 <xsd:complexContent>
2970 <xsd:extension base="tns:RequestTemplate">
2971 <xsd:sequence>
2972 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
2973 <xsd:annotation>
2974 <xsd:documentation>A value used to retrieve the results of an IdentifySubject request.</xsd:documentation>
2975 </xsd:annotation>
2976 </xsd:element>
2977 </xsd:sequence>
2978 </xsd:extension>
2979 </xsd:complexContent>
2980 </xsd:complexType>
2981 </xsd:element>
2982 </xsd:sequence>
2983 </xsd:complexType>
2984 </xsd:element>
2985 <xsd:element name="GetIdentifySubjectResultsResponse">
2986 <xsd:complexType>
2987 <xsd:annotation>
2988 <xsd:documentation>
2989 The response to a GetIdentifySubjectResults request, which includes a candidate list.
2990 </xsd:documentation>
2991 </xsd:annotation>
2992 <xsd:sequence>
2993 <xsd:element name="GetIdentifySubjectResultsResponsePackage" nillable="true">

```

```

2994 <xsd:complexType>
2995 <xsd:complexContent>
2996 <xsd:extension base="tns:ResponseTemplate">
2997 <xsd:sequence>
2998 <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="1" maxOccurs="1">
2999 <xsd:annotation>
3000 <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching the input biometric sample.</xsd:documentation>
3001 </xsd:annotation>
3002 </xsd:element>
3003 </xsd:sequence>
3004 </xsd:extension>
3005 </xsd:complexContent>
3006 </xsd:complexType>
3007 </xsd:element>
3008 </xsd:sequence>
3009 </xsd:complexType>
3010 </xsd:element>
3011 <xsd:element name="IdentifySubject">
3012 <xsd:complexType>
3013 <xsd:annotation>
3014 <xsd:documentation>
3015 Perform an identification search against a given gallery for
3016 a given biometric.
3017 </xsd:documentation>
3018 </xsd:annotation>
3019 <xsd:sequence>
3020 <xsd:element name="IdentifySubjectRequest" nillable="true">
3021 <xsd:complexType>
3022 <xsd:complexContent>
3023 <xsd:extension base="tns:RequestTemplate">
3024 <xsd:sequence>
3025 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3026 <xsd:annotation>
3027 <xsd:documentation>The identifier of the gallery or population group which will be searched. Must not be used in conjunction with Gallery parameter</xsd:documentation>

```

```

3028     </xsd:annotation>
3029 </xsd:element>
3030 <xsd:element name="Gallery" type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
3031     <xsd:annotation>
3032         <xsd:documentation>A list of BIRs to be used instead of a stored gallery. Must not be used in conjunction with GalleryID parameter.</xsd:documentation>
3033     </xsd:annotation>
3034 </xsd:element>
3035 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3036     <xsd:annotation>
3037         <xsd:documentation>Contains the BIR, a data structure containing the biometric sample for the search.</xsd:documentation>
3038     </xsd:annotation>
3039 </xsd:element>
3040 <xsd:element name="MaxListSize" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
3041     <xsd:annotation>
3042         <xsd:documentation>The maximum size of the candidate list that should be returned.</xsd:documentation>
3043     </xsd:annotation>
3044 </xsd:element>
3045 </xsd:sequence>
3046 </xsd:extension>
3047 </xsd:complexContent>
3048 </xsd:complexType>
3049 </xsd:element>
3050 </xsd:sequence>
3051 </xsd:complexType>
3052 </xsd:element>
3053 <xsd:element name="IdentifySubjectResponse">
3054     <xsd:complexType>
3055         <xsd:annotation>
3056             <xsd:documentation>
3057                 The response to an IdentifySubject request, returning a
3058                 rank-ordered candidate list.
3059             </xsd:documentation>
3060         </xsd:annotation>
3061     <xsd:sequence>

```

```

3062 <xsd:element name="IdentifySubjectResponsePackage" nillable="true">
3063 <xsd:complexType>
3064 <xsd:complexContent>
3065 <xsd:extension base="tns:ResponseTemplate">
3066 <xsd:sequence>
3067 <xsd:element name="IdentifySubjectResult" type="tns:IdentifySubjectResultType"/>
3068 </xsd:sequence>
3069 </xsd:extension>
3070 </xsd:complexContent>
3071 </xsd:complexType>
3072 </xsd:element>
3073 </xsd:sequence>
3074 </xsd:complexType>
3075 </xsd:element>
3076 <xsd:complexType name="IdentifySubjectResultType">
3077 <xsd:sequence/>
3078 </xsd:complexType>
3079 <xsd:complexType name="CandidateListResultType">
3080 <xsd:complexContent>
3081 <xsd:extension base="tns:IdentifySubjectResultType">
3082 <xsd:sequence>
3083 <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
3084 <xsd:annotation>
3085 <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching the input biometric sample; returned with successful synchronous request
3086 processing.</xsd:documentation>
3087 </xsd:annotation>
3088 </xsd:element>
3089 </xsd:sequence>
3090 </xsd:extension>
3091 </xsd:complexContent>
3092 </xsd:complexType>
3093 <xsd:complexType name="TokenResultType">
3094 <xsd:complexContent>
3095 <xsd:extension base="tns:IdentifySubjectResultType">

```

```

3096 <xsd:sequence>
3097 <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
3098 <xsd:annotation>
3099 <xsd:documentation>A token used to retrieve the results of the IdentifySubject request; returned with asynchronous request processing.</xsd:documentation>
3100 </xsd:annotation>
3101 </xsd:element>
3102 </xsd:sequence>
3103 </xsd:extension>
3104 </xsd:complexContent>
3105 </xsd:complexType>
3106 <xsd:element name="ListBiographicData">
3107 <xsd:complexType>
3108 <xsd:annotation>
3109 <xsd:documentation>
3110 Lists the biographic data elements stored for a subject.
3111 </xsd:documentation>
3112 </xsd:annotation>
3113 <xsd:sequence>
3114 <xsd:element name="ListBiographicDataRequest" nillable="true">
3115 <xsd:complexType>
3116 <xsd:complexContent>
3117 <xsd:extension base="tns:RequestTemplate">
3118 <xsd:sequence>
3119 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3120 <xsd:annotation>
3121 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
3122 </xsd:annotation>
3123 </xsd:element>
3124 <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3125 <xsd:annotation>
3126 <xsd:documentation>Identifies the category of encounter. If an encounter ID is not specified
3127 and encounter data exists for the subject, the operation returns the list of encounter IDs of a specific
3128 type which contain biographic data using the Encounter List output parameter, and the Biographic
3129 Data Elements output parameter is empty. Should not be used in conjunction with EncounterID.

```

```

3130         </xsd:documentation>
3131     </xsd:annotation>
3132 </xsd:element>
3133 </xsd:sequence>
3134 </xsd:extension>
3135 </xsd:complexContent>
3136 </xsd:complexType>
3137 </xsd:element>
3138 </xsd:sequence>
3139 </xsd:complexType>
3140 </xsd:element>
3141 <xsd:element name="ListBiographicDataResponse">
3142     <xsd:complexType>
3143     <xsd:annotation>
3144     <xsd:documentation>
3145         The response to a ListBiographicData request, containing a list
3146         of biographic data elements stored for a subject. In the
3147         encounter-centric model, the biographic data elements for a
3148         specific encounter are returned. If an encounter ID is not
3149         specified and encounter data exists for the subject, the list
3150         of encounter IDs which contain biographic data is returned.
3151     </xsd:documentation>
3152     </xsd:annotation>
3153     <xsd:sequence>
3154     <xsd:element name="ListBiographicDataResponsePackage" nillable="true">
3155     <xsd:complexType>
3156     <xsd:complexContent>
3157     <xsd:extension base="tns:ResponseTemplate">
3158     <xsd:sequence>
3159     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3160     <xsd:annotation>
3161     <xsd:documentation>
3162         Contains a list of biographic data elements associated with a
3163         subject or encounter; non-empty if the service was

```

3164 successful, biographic data exists, and either (a) the
 3165 person-centric model is being used or (b) the
 3166 encounter-centric model is being used and an encounter
 3167 identifier was specified.
 3168 </xsd:documentation>
 3169 </xsd:annotation>
 3170 </xsd:element>
 3171 </xsd:sequence>
 3172 </xsd:extension>
 3173 </xsd:complexContent>
 3174 </xsd:complexType>
 3175 </xsd:element>
 3176 </xsd:sequence>
 3177 </xsd:complexType>
 3178 </xsd:element>
 3179 <xsd:element name="ListBiometricData">
 3180 <xsd:complexType>
 3181 <xsd:annotation>
 3182 <xsd:documentation>
 3183 Lists the biometric data elements stored for a subject. Note
 3184 that no actual biometric data is returned by this service (see
 3185 the RetrieveBiometricData service to obtain the biometric
 3186 data).
 3187 </xsd:documentation>
 3188 </xsd:annotation>
 3189 <xsd:sequence>
 3190 <xsd:element name="ListBiometricDataRequest" nillable="true">
 3191 <xsd:complexType>
 3192 <xsd:complexContent>
 3193 <xsd:extension base="tns:RequestTemplate">
 3194 <xsd:sequence>
 3195 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
 3196 <xsd:annotation>
 3197 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>

```

3198     </xsd:annotation>
3199   </xsd:element>
3200   <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3201     <xsd:annotation>
3202       <xsd:documentation>Identifies the category of encounter. If an encounter ID is not specified and
3203       encounter data exists for the subject, the operation may return the list of encounter IDs of a specific
3204       type which contain biometric data using the Encounter List output parameter, and the Biometric Data
3205       List output parameter is empty. Should not be used in conjunction with EncounterID.
3206     </xsd:documentation>
3207   </xsd:annotation>
3208 </xsd:element>
3209   <xsd:element name="ListFilter" type="tns:ListFilterType" minOccurs="0" maxOccurs="1">
3210     <xsd:annotation>
3211       <xsd:documentation>Indicates what biometric information should be returned.</xsd:documentation>
3212     </xsd:annotation>
3213   </xsd:element>
3214 </xsd:sequence>
3215 </xsd:extension>
3216 </xsd:complexContent>
3217 </xsd:complexType>
3218 </xsd:element>
3219 </xsd:sequence>
3220 </xsd:complexType>
3221 </xsd:element>
3222 <xsd:element name="ListBiometricDataResponse">
3223   <xsd:complexType>
3224     <xsd:annotation>
3225       <xsd:documentation>
3226         The response to a ListBiometricData request, containing a list
3227         of biometric data elements stored for a subject. In the
3228         encounter-centric model, the biometric data elements for a
3229         specific encounter are returned. If an encounter ID is not
3230         specified and encounter data exists for the subject, the list
3231         of encounter IDs which contain biometric data is returned.

```



```

3232     </xsd:documentation>
3233 </xsd:annotation>
3234 <xsd:sequence>
3235   <xsd:element name="ListBiometricDataResponsePackage" nillable="true">
3236     <xsd:complexType>
3237       <xsd:complexContent>
3238         <xsd:extension base="tns:ResponseTemplate">
3239           <xsd:sequence>
3240             <xsd:element name="Identity" type="tns:BIASIdentity">
3241               <xsd:annotation>
3242                 <xsd:documentation>
3243                   Includes a list of biometric data elements associated
3244                   with a subject or encounter or a list of encounter ID's
3245                   associated with a subject and which contain biometric
3246                   data.
3247                 </xsd:documentation>
3248               </xsd:annotation>
3249             </xsd:element>
3250           </xsd:sequence>
3251         </xsd:extension>
3252       </xsd:complexContent>
3253     </xsd:complexType>
3254   </xsd:element>
3255 </xsd:sequence>
3256 </xsd:complexType>
3257 </xsd:element>
3258 <xsd:element name="ListDocumentData">
3259   <xsd:complexType>
3260     <xsd:annotation>
3261       <xsd:documentation>
3262         Lists the document categories stored for a subject.
3263       </xsd:documentation>
3264     </xsd:annotation>
3265   <xsd:sequence>

```

```

3266 <xsd:element name="ListDocumentDataRequest" nillable="true">
3267 <xsd:complexType>
3268 <xsd:complexContent>
3269 <xsd:extension base="tns:RequestTemplate">
3270 <xsd:sequence>
3271 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3272 <xsd:annotation>
3273 <xsd:documentation>
3274     Identifies the subject or, in the encounter-centric model, a subject and an encounter.
3275 </xsd:documentation>
3276 </xsd:annotation>
3277 </xsd:element>
3278 <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3279 <xsd:annotation>
3280 <xsd:documentation>Identifies the category of encounter. If an encounter ID is not specified and encounter data exists for
3281     the subject, the operation shall return the list of encounter IDs which contain document data using the Encounter List
3282     Output parameter, and the Document Data Elements output parameter shall be empty. Should not be used in conjunction
3283     with EncounterID.
3284 </xsd:documentation>
3285 </xsd:annotation>
3286 </xsd:element>
3287 </xsd:sequence>
3288 </xsd:extension>
3289 </xsd:complexContent>
3290 </xsd:complexType>
3291 </xsd:element>
3292 </xsd:sequence>
3293 </xsd:complexType>
3294 </xsd:element>
3295 <xsd:element name="ListDocumentDataResponse">
3296 <xsd:complexType>
3297 <xsd:annotation>
3298 <xsd:documentation>
3299     The response to a ListDocumentData request, containing a list of document categories stored for a subject.

```

3300 In the encounter-centric model, an encounterID may be specified to indicate that only document data categories
3301 stored for that encounter should be returned. If an encounter ID is not
3302 specified and encounter data exists for the subject, the list
3303 of encounter IDs which contain document data is returned.
3304 </xsd:documentation>
3305 </xsd:annotation>
3306 <xsd:sequence>
3307 <xsd:element name="ListDocumentDataResponsePackage" nillable="true">
3308 <xsd:complexType>
3309 <xsd:complexContent>
3310 <xsd:extension base="tns:ResponseTemplate">
3311 <xsd:sequence>
3312 <xsd:element name="DocumentDataList" type="tns:DocumentDataListType" minOccurs="0" maxOccurs="1">
3313 <xsd:annotation>
3314 <xsd:documentation>
3315 A list of document categories associated with a subject or encounter; non-empty if the service
3316 was successful, document data exists, and either the person-centric model is being used or the
3317 encounter-centric model is being used and an encounter identifier was specified.
3318 </xsd:documentation>
3319 </xsd:annotation>
3320 </xsd:element>
3321 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3322 <xsd:annotation>
3323 <xsd:documentation>
3324 A list of encounter IDs associated with a subject and which contain document data; non-empty
3325 if the service was successful, document data exists, the encounter-centric model is being used,
3326 and an encounter identifier was not specified..
3327 </xsd:documentation>
3328 </xsd:annotation>
3329 </xsd:element>
3330 </xsd:sequence>
3331 </xsd:extension>
3332 </xsd:complexContent>
3333 </xsd:complexType>

```

3334     </xsd:element>
3335 </xsd:sequence>
3336 </xsd:complexType>
3337 </xsd:element>
3338 <xsd:element name="PerformFusion">
3339   <xsd:complexType>
3340     <xsd:annotation>
3341       <xsd:documentation>
3342         Accepts either match score or match decision information and creates a fused match result.
3343       </xsd:documentation>
3344     </xsd:annotation>
3345     <xsd:sequence>
3346       <xsd:element name="PerformFusionRequest" nillable="true">
3347         <xsd:complexType>
3348           <xsd:complexContent>
3349             <xsd:extension base="tns:RequestTemplate">
3350               <xsd:sequence>
3351                 <xsd:element name="FusionInput" type="tns:FusionIdentityListType" minOccurs="1" maxOccurs="1">
3352                   <xsd:annotation>
3353                     <xsd:documentation>Score or decision input information to the fusion method for each identity.</xsd:documentation>
3354                   </xsd:annotation>
3355                 </xsd:element>
3356               </xsd:sequence>
3357             </xsd:extension>
3358           </xsd:complexContent>
3359         </xsd:complexType>
3360       </xsd:element>
3361     </xsd:sequence>
3362   </xsd:complexType>
3363 </xsd:element>
3364 <xsd:element name="PerformFusionResponse">
3365   <xsd:complexType>
3366     <xsd:annotation>
3367       <xsd:documentation>

```

3368 The response to the PerformFusion request.
 3369 </xsd:documentation>
 3370 </xsd:annotation>
 3371 <xsd:sequence>
 3372 <xsd:element name="PerformFusionResponsePackage" nillable="true">
 3373 <xsd:complexType>
 3374 <xsd:complexContent>
 3375 <xsd:extension base="tns:ResponseTemplate">
 3376 <xsd:sequence>
 3377 <xsd:element name="Match" type="tns:MatchType" minOccurs="1" maxOccurs="1">
 3378 <xsd:annotation>
 3379 <xsd:documentation>Indicates the result of the fusion method</xsd:documentation>
 3380 </xsd:annotation>
 3381 </xsd:element>
 3382 </xsd:sequence>
 3383 </xsd:extension>
 3384 </xsd:complexContent>
 3385 </xsd:complexType>
 3386 </xsd:element>
 3387 </xsd:sequence>
 3388 </xsd:complexType>
 3389 </xsd:element>
 3390 <xsd:element name="QueryCapabilities">
 3391 <xsd:complexType>
 3392 <xsd:annotation>
 3393 <xsd:documentation>
 3394 Returns a list of the capabilities, options, galleries, etc.
 3395 that are supported by the BIAS implementation.
 3396 </xsd:documentation>
 3397 </xsd:annotation>
 3398 <xsd:sequence>
 3399 <xsd:element name="QueryCapabilitiesRequest" nillable="true">
 3400 <xsd:complexType>
 3401 <xsd:complexContent>

```

3402     <xsd:extension base="tns:RequestTemplate"/>
3403   </xsd:complexContent>
3404 </xsd:complexType>
3405 </xsd:element>
3406 </xsd:sequence>
3407 </xsd:complexType>
3408 </xsd:element>
3409 <xsd:element name="QueryCapabilitiesResponse">
3410   <xsd:complexType>
3411     <xsd:annotation>
3412       <xsd:documentation>
3413         The response to a QueryCapabilities request.
3414       </xsd:documentation>
3415     </xsd:annotation>
3416     <xsd:sequence>
3417       <xsd:element name="QueryCapabilitiesResponsePackage" nillable="true">
3418         <xsd:complexType>
3419           <xsd:complexContent>
3420             <xsd:extension base="tns:ResponseTemplate">
3421               <xsd:sequence>
3422                 <xsd:element name="CapabilityList" type="tns:CapabilityListType" minOccurs="1" maxOccurs="1">
3423                   <xsd:annotation>
3424                     <xsd:documentation>A list of capabilities supported by the BIAS implementation.</xsd:documentation>
3425                   </xsd:annotation>
3426                 </xsd:element>
3427               </xsd:sequence>
3428             </xsd:extension>
3429           </xsd:complexContent>
3430         </xsd:complexType>
3431       </xsd:element>
3432     </xsd:sequence>
3433   </xsd:complexType>
3434 </xsd:element>
3435 <xsd:element name="RetrieveBiographicData">

```

```

3436 <xsd:complexType>
3437 <xsd:annotation>
3438 <xsd:documentation>
3439     Retrieves the biographic data associated with a subject ID.
3440 </xsd:documentation>
3441 </xsd:annotation>
3442 <xsd:sequence>
3443 <xsd:element name="RetrieveBiographicDataRequest" nillable="true">
3444 <xsd:complexType>
3445 <xsd:complexContent>
3446 <xsd:extension base="tns:RequestTemplate">
3447 <xsd:sequence>
3448 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3449 <xsd:annotation>
3450 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
3451 </xsd:annotation>
3452 </xsd:element>
3453 <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3454 <xsd:annotation>
3455 <xsd:documentation>
3456     Identifies the type of encounter during which data was collected from the subject,
3457     as determined by the requester.
3458 </xsd:documentation>
3459 </xsd:annotation>
3460 </xsd:element>
3461 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3462 <xsd:annotation>
3463 <xsd:documentation>The identifier of the gallery or population group from which the biographic information will be retrieved.</xsd:documentation>
3464 </xsd:annotation>
3465 </xsd:element>
3466 </xsd:sequence>
3467 </xsd:extension>
3468 </xsd:complexContent>
3469 </xsd:complexType>

```

```

3470     </xsd:element>
3471 </xsd:sequence>
3472 </xsd:complexType>
3473 </xsd:element>
3474 <xsd:element name="RetrieveBiographicDataResponse">
3475   <xsd:complexType>
3476     <xsd:annotation>
3477       <xsd:documentation>
3478         The response to a RetrieveBiographicData request,
3479         containing the biographic data associated with a subject ID. In
3480         the encounter-centric model, the biographic data associated with
3481         a specified encounter is returned. If the encounter ID is not
3482         specified in the encounter-centric model, the biographic
3483         information associated with the most recent encounter is returned.
3484       </xsd:documentation>
3485     </xsd:annotation>
3486     <xsd:sequence>
3487       <xsd:element name="RetrieveBiographicDataResponsePackage" nillable="true">
3488         <xsd:complexType>
3489           <xsd:complexContent>
3490             <xsd:extension base="tns:ResponseTemplate">
3491               <xsd:sequence>
3492                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3493                   <xsd:annotation>
3494                     <xsd:documentation>Includes the set of biographic data associated with a subject.</xsd:documentation>
3495                   </xsd:annotation>
3496                 </xsd:element>
3497               </xsd:sequence>
3498             </xsd:extension>
3499           </xsd:complexContent>
3500         </xsd:complexType>
3501       </xsd:element>
3502     </xsd:sequence>
3503   </xsd:complexType>

```



```

3504 </xsd:element>
3505 <xsd:element name="RetrieveBiometricData">
3506   <xsd:complexType>
3507     <xsd:annotation>
3508       <xsd:documentation>
3509         Retrieves the biometric data associated with a subject ID.
3510       </xsd:documentation>
3511     </xsd:annotation>
3512     <xsd:sequence>
3513       <xsd:element name="RetrieveBiometricDataRequest" nillable="true">
3514         <xsd:complexType>
3515           <xsd:complexContent>
3516             <xsd:extension base="tns:RequestTemplate">
3517               <xsd:sequence>
3518                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3519                   <xsd:annotation>
3520                     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
3521                   </xsd:annotation>
3522                 </xsd:element>
3523                 <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3524                   <xsd:annotation>
3525                     <xsd:documentation>Identifies the type of encounter during which data was collected from the subject,
3526                     as determined by the requester.
3527                   </xsd:documentation>
3528                 </xsd:annotation>
3529                 </xsd:element>
3530                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3531                   <xsd:annotation>
3532                     <xsd:documentation>The identifier of the gallery or population group from which the biometric information will be retrieved.</xsd:documentation>
3533                   </xsd:annotation>
3534                 </xsd:element>
3535                 <xsd:element name="BiometricType" type="oasis_cbeff:MultipleTypesType" minOccurs="0" maxOccurs="1">
3536                   <xsd:annotation>
3537                     <xsd:documentation>The type of biological or behavioral data to retrieve.</xsd:documentation>

```

```

3538     </xsd:annotation>
3539   </xsd:element>
3540 </xsd:sequence>
3541 </xsd:extension>
3542 </xsd:complexContent>
3543 </xsd:complexType>
3544 </xsd:element>
3545 </xsd:sequence>
3546 </xsd:complexType>
3547 </xsd:element>
3548 <xsd:element name="RetrieveBiometricDataResponse">
3549   <xsd:complexType>
3550     <xsd:annotation>
3551       <xsd:documentation>
3552         The response to a RetrieveBiometricData request,
3553         containing the biometric data associated with a subject ID. In
3554         the encounter-centric model, the biometric data associated with
3555         a specified encounter is returned. If the encounter ID is not
3556         specified in the encounter-centric model, the biometric
3557         information associated with the most recent encounter is returned.
3558       </xsd:documentation>
3559     </xsd:annotation>
3560     <xsd:sequence>
3561       <xsd:element name="RetrieveBiometricDataResponsePackage" nillable="true">
3562         <xsd:complexType>
3563           <xsd:complexContent>
3564             <xsd:extension base="tns:ResponseTemplate">
3565               <xsd:sequence>
3566                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3567                   <xsd:annotation>
3568                     <xsd:documentation>Includes the biometric data associated with a subject.</xsd:documentation>
3569                   </xsd:annotation>
3570                 </xsd:element>
3571               </xsd:sequence>

```

```

3572     </xsd:extension>
3573     </xsd:complexContent>
3574   </xsd:complexType>
3575 </xsd:element>
3576 </xsd:sequence>
3577 </xsd:complexType>
3578 </xsd:element>
3579 <xsd:element name="RetrieveDocumentData">
3580   <xsd:complexType>
3581     <xsd:annotation>
3582       <xsd:documentation>Retrieves the list of document data associated with a subject ID for the category(ies) specified. </xsd:documentation>
3583     </xsd:annotation>
3584     <xsd:sequence>
3585       <xsd:element name="RetrieveDocumentDataRequest" nillable="true">
3586         <xsd:complexType>
3587           <xsd:complexContent>
3588             <xsd:extension base="tns:RequestTemplate">
3589               <xsd:sequence>
3590                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3591                   <xsd:annotation>
3592                     <xsd:documentation>Contains either the subject ID or encounter ID reference.</xsd:documentation>
3593                   </xsd:annotation>
3594                 </xsd:element>
3595                 <xsd:element name="DocumentData" type="tns:DocumentDataType" minOccurs="0" maxOccurs="1">
3596                   <xsd:annotation>
3597                     <xsd:documentation>Defines a set of document data elements providing information about the requested identity document. </xsd:documentation>
3598                   </xsd:annotation>
3599                 </xsd:element>
3600                 <xsd:element name="EncounterType" type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
3601                   <xsd:annotation>
3602                     <xsd:documentation>Identifies the category of encounter.</xsd:documentation>
3603                   </xsd:annotation>
3604                 </xsd:element>
3605                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">

```

```

3606     <xsd:annotation>
3607     <xsd:documentation>The identifier of the gallery or population group from which the biographic information will be retrieved. </xsd:documentation>
3608     </xsd:annotation>
3609     </xsd:element>
3610 </xsd:sequence>
3611 </xsd:extension>
3612 </xsd:complexContent>
3613 </xsd:complexType>
3614 </xsd:element>
3615 </xsd:sequence>
3616 </xsd:complexType>
3617 </xsd:element>
3618 <xsd:element name="RetrieveDocumentDataResponse">
3619 <xsd:complexType>
3620 <xsd:annotation>
3621 <xsd:documentation>
3622     The response to a RetrieveDocumentData request.
3623 </xsd:documentation>
3624 </xsd:annotation>
3625 <xsd:sequence>
3626 <xsd:element name="RetrieveDocumentDataResponsePackage" nillable="true">
3627 <xsd:complexType>
3628 <xsd:complexContent>
3629     <xsd:extension base="tns:ResponseTemplate"/>
3630 </xsd:complexContent>
3631 </xsd:complexType>
3632 </xsd:element>
3633 <xsd:element name="DocumentDataList" type="tns:DocumentDataListType" minOccurs="1" maxOccurs="1">
3634 <xsd:annotation>
3635 <xsd:documentation>A list of document data associated with a subject or encounter</xsd:documentation>
3636 </xsd:annotation>
3637 </xsd:element>
3638 </xsd:sequence>
3639 </xsd:complexType>

```

```

3640 </xsd:element>
3641 <xsd:element name="SetBiographicData">
3642   <xsd:complexType>
3643     <xsd:annotation>
3644       <xsd:documentation>
3645         Associates biographic data to a given subject record.
3646       </xsd:documentation>
3647     </xsd:annotation>
3648     <xsd:sequence>
3649       <xsd:element name="SetBiographicDataRequest" nillable="true">
3650         <xsd:complexType>
3651           <xsd:complexContent>
3652             <xsd:extension base="tns:RequestTemplate">
3653               <xsd:sequence>
3654                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3655                   <xsd:annotation>
3656                     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to store.</xsd:documentation>
3657                   </xsd:annotation>
3658                 </xsd:element>
3659                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3660                   <xsd:annotation>
3661                     <xsd:documentation>The identifier of the gallery or population group to which the biographic will be added.</xsd:documentation>
3662                   </xsd:annotation>
3663                 </xsd:element>
3664               </xsd:sequence>
3665             </xsd:extension>
3666           </xsd:complexContent>
3667         </xsd:complexType>
3668       </xsd:element>
3669     </xsd:sequence>
3670   </xsd:complexType>
3671 </xsd:element>
3672 <xsd:element name="SetBiographicDataResponse">
3673   <xsd:complexType>

```

```

3674 <xsd:annotation>
3675 <xsd:documentation>
3676   The response to a SetBiographicData request.
3677 </xsd:documentation>
3678 </xsd:annotation>
3679 <xsd:sequence>
3680 <xsd:element name="SetBiographicDataResponsePackage" nillable="true">
3681 <xsd:complexType>
3682 <xsd:complexContent>
3683 <xsd:extension base="tns:ResponseTemplate">
3684 <xsd:sequence>
3685 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3686 <xsd:annotation>
3687 <xsd:documentation>In an encounter-centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
3688 </xsd:annotation>
3689 </xsd:element>
3690 </xsd:sequence>
3691 </xsd:extension>
3692 </xsd:complexContent>
3693 </xsd:complexType>
3694 </xsd:element>
3695 </xsd:sequence>
3696 </xsd:complexType>
3697 </xsd:element>
3698 <xsd:element name="SetBiometricData">
3699 <xsd:complexType>
3700 <xsd:annotation>
3701 <xsd:documentation>
3702   Associates biometric data to a given subject record.
3703 </xsd:documentation>
3704 </xsd:annotation>
3705 <xsd:sequence>
3706 <xsd:element name="SetBiometricDataRequest" nillable="true">
3707 <xsd:complexType>

```

```

3708     <xsd:complexContent>
3709     <xsd:extension base="tns:RequestTemplate">
3710     <xsd:sequence>
3711     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3712     <xsd:annotation>
3713     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to store.</xsd:documentation>
3714     </xsd:annotation>
3715     </xsd:element>
3716     <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3717     <xsd:annotation>
3718     <xsd:documentation>The identifier of the gallery or population group to which the biometric will be added.</xsd:documentation>
3719     </xsd:annotation>
3720     </xsd:element>
3721     </xsd:sequence>
3722     </xsd:extension>
3723     </xsd:complexContent>
3724     </xsd:complexType>
3725     </xsd:element>
3726     </xsd:sequence>
3727     </xsd:complexType>
3728     </xsd:element>
3729     <xsd:element name="SetBiometricDataResponse">
3730     <xsd:complexType>
3731     <xsd:annotation>
3732     <xsd:documentation>
3733     The response to a SetBiometricData request.
3734     </xsd:documentation>
3735     </xsd:annotation>
3736     <xsd:sequence>
3737     <xsd:element name="SetBiometricDataResponsePackage" nillable="true">
3738     <xsd:complexType>
3739     <xsd:complexContent>
3740     <xsd:extension base="tns:ResponseTemplate">
3741     <xsd:sequence>

```

```

3742     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3743         <xsd:annotation>
3744             <xsd:documentation>In an encounter-centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
3745         </xsd:annotation>
3746     </xsd:element>
3747 </xsd:sequence>
3748 </xsd:extension>
3749 </xsd:complexContent>
3750 </xsd:complexType>
3751 </xsd:element>
3752 </xsd:sequence>
3753 </xsd:complexType>
3754 </xsd:element>
3755 <xsd:element name="SetDocumentData">
3756     <xsd:complexType>
3757         <xsd:annotation>
3758             <xsd:documentation>
3759                 Associates document data to a given subject record.
3760             </xsd:documentation>
3761         </xsd:annotation>
3762     <xsd:sequence>
3763         <xsd:element name="SetDocumentDataRequest" nillable="true">
3764             <xsd:complexType>
3765                 <xsd:complexContent>
3766                     <xsd:extension base="tns:RequestTemplate">
3767                         <xsd:sequence>
3768                             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3769                                 <xsd:annotation>
3770                                     <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the document data to store.</xsd:documentation>
3771                                 </xsd:annotation>
3772                             </xsd:element>
3773                             <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
3774                                 <xsd:annotation>
3775                                     <xsd:documentation>The identifier of the gallery or population group to which the document data will be added.</xsd:documentation>

```



```

3776     </xsd:annotation>
3777 </xsd:element>
3778 <xsd:element name="DocumentDataList" type="tns:DocumentDataListType" minOccurs="1" maxOccurs="1">
3779     <xsd:annotation>
3780     <xsd:documentation>Defines a set of document data elements to associate with the subject or encounter. </xsd:documentation>
3781     </xsd:annotation>
3782 </xsd:element>
3783 </xsd:sequence>
3784 </xsd:extension>
3785 </xsd:complexContent>
3786 </xsd:complexType>
3787 </xsd:element>
3788 </xsd:sequence>
3789 </xsd:complexType>
3790 </xsd:element>
3791 <xsd:element name="SetDocumentDataResponse">
3792 <xsd:complexType>
3793 <xsd:annotation>
3794 <xsd:documentation>
3795     The response to a SetDocumentData request.
3796 </xsd:documentation>
3797 </xsd:annotation>
3798 <xsd:sequence>
3799 <xsd:element name="SetDocumentDataResponsePackage" nillable="true">
3800 <xsd:complexType>
3801 <xsd:complexContent>
3802 <xsd:extension base="tns:ResponseTemplate">
3803 <xsd:sequence>
3804 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
3805 <xsd:annotation>
3806 <xsd:documentation>In an encounter-centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
3807 </xsd:annotation>
3808 </xsd:element>
3809 </xsd:sequence>

```

```

3810     </xsd:extension>
3811   </xsd:complexContent>
3812 </xsd:complexType>
3813 </xsd:element>
3814 </xsd:sequence>
3815 </xsd:complexType>
3816 </xsd:element>
3817 <xsd:element name="TransformBiometricData">
3818   <xsd:complexType>
3819     <xsd:annotation>
3820       <xsd:documentation>
3821         Transforms or processes a given biometric in one format into a new target format.
3822       </xsd:documentation>
3823     </xsd:annotation>
3824     <xsd:sequence>
3825       <xsd:element name="TransformBiometricDataRequest" nillable="true">
3826         <xsd:complexType>
3827           <xsd:complexContent>
3828             <xsd:extension base="tns:RequestTemplate">
3829               <xsd:sequence>
3830                 <xsd:element name="InputBIR" type="tns:CBEFF_BIR_Type" minOccurs="1" maxOccurs="1">
3831                   <xsd:annotation>
3832                     <xsd:documentation>Data structure containing the biometric information to be transformed.</xsd:documentation>
3833                   </xsd:annotation>
3834                 </xsd:element>
3835                 <xsd:element name="TransformOperation" type="xsd:unsignedLong" minOccurs="1" maxOccurs="1">
3836                   <xsd:annotation>
3837                     <xsd:documentation>Value indicating the type of transformation to perform.</xsd:documentation>
3838                   </xsd:annotation>
3839                 </xsd:element>
3840                 <xsd:element name="TransformControl" type="xsd:string" minOccurs="0" maxOccurs="1">
3841                   <xsd:annotation>
3842                     <xsd:documentation> Specifies controls for the requested transform operation.</xsd:documentation>
3843                   </xsd:annotation>

```

```

3844     </xsd:element>
3845   </xsd:sequence>
3846 </xsd:extension>
3847 </xsd:complexContent>
3848 </xsd:complexType>
3849 </xsd:element>
3850 </xsd:sequence>
3851 </xsd:complexType>
3852 </xsd:element>
3853 <xsd:element name="TransformBiometricDataResponse">
3854   <xsd:complexType>
3855     <xsd:annotation>
3856       <xsd:documentation>
3857         The response to a TransformBiometricData request.
3858       </xsd:documentation>
3859     </xsd:annotation>
3860     <xsd:sequence>
3861       <xsd:element name="TransformBiometricDataResponsePackage" nillable="true">
3862         <xsd:complexType>
3863           <xsd:complexContent>
3864             <xsd:extension base="tns:ResponseTemplate">
3865               <xsd:sequence>
3866                 <xsd:element name="OutputBIR" type="tns:CBEFF_BIR_Type" minOccurs="1" maxOccurs="1">
3867                   <xsd:annotation>
3868                     <xsd:documentation>Data structure containing the new, transformed biometric information.</xsd:documentation>
3869                   </xsd:annotation>
3870                 </xsd:element>
3871               </xsd:sequence>
3872             </xsd:extension>
3873           </xsd:complexContent>
3874         </xsd:complexType>
3875       </xsd:element>
3876     </xsd:sequence>
3877   </xsd:complexType>

```

```

3878 </xsd:element>
3879 <xsd:element name="UpdateBiographicData">
3880 <xsd:complexType>
3881 <xsd:annotation>
3882 <xsd:documentation>
3883     Updates the biographic data for a given subject record.
3884 </xsd:documentation>
3885 </xsd:annotation>
3886 <xsd:sequence>
3887 <xsd:element name="UpdateBiographicDataRequest" nillable="true">
3888 <xsd:complexType>
3889 <xsd:complexContent>
3890 <xsd:extension base="tns:RequestTemplate">
3891 <xsd:sequence>
3892 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3893 <xsd:annotation>
3894 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biographic data to update.</xsd:documentation>
3895 </xsd:annotation>
3896 </xsd:element>
3897 </xsd:sequence>
3898 </xsd:extension>
3899 </xsd:complexContent>
3900 </xsd:complexType>
3901 </xsd:element>
3902 </xsd:sequence>
3903 </xsd:complexType>
3904 </xsd:element>
3905 <xsd:element name="UpdateBiographicDataResponse">
3906 <xsd:complexType>
3907 <xsd:annotation>
3908 <xsd:documentation>
3909     The response to an UpdateBiographicData request.
3910 </xsd:documentation>
3911 </xsd:annotation>

```

```

3912 <xsd:sequence>
3913 <xsd:element name="UpdateBiographicDataResponsePackage" nillable="true">
3914 <xsd:complexType>
3915 <xsd:complexContent>
3916 <xsd:extension base="tns:ResponseTemplate"/>
3917 </xsd:complexContent>
3918 </xsd:complexType>
3919 </xsd:element>
3920 </xsd:sequence>
3921 </xsd:complexType>
3922 </xsd:element>
3923 <xsd:element name="UpdateBiometricData">
3924 <xsd:complexType>
3925 <xsd:annotation>
3926 <xsd:documentation>
3927 Updates a single biometric sample for a given subject record.
3928 </xsd:documentation>
3929 </xsd:annotation>
3930 <xsd:sequence>
3931 <xsd:element name="UpdateBiometricDataRequest" nillable="true">
3932 <xsd:complexType>
3933 <xsd:complexContent>
3934 <xsd:extension base="tns:RequestTemplate">
3935 <xsd:sequence>
3936 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3937 <xsd:annotation>
3938 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the biometric data to update.</xsd:documentation>
3939 </xsd:annotation>
3940 </xsd:element>
3941 <xsd:element name="Merge" type="xsd:boolean" minOccurs="0" maxOccurs="1">
3942 <xsd:annotation>
3943 <xsd:documentation>Value indicating if the input biometric sample should be merged with any existing biometric information.</xsd:documentation>
3944 </xsd:annotation>
3945 </xsd:element>

```

```

3946     </xsd:sequence>
3947   </xsd:extension>
3948 </xsd:complexContent>
3949 </xsd:complexType>
3950 </xsd:element>
3951 </xsd:sequence>
3952 </xsd:complexType>
3953 </xsd:element>
3954 <xsd:element name="UpdateBiometricDataResponse">
3955   <xsd:complexType>
3956     <xsd:annotation>
3957       <xsd:documentation>
3958         The response to an UpdateBiometricData request.
3959       </xsd:documentation>
3960     </xsd:annotation>
3961     <xsd:sequence>
3962       <xsd:element name="UpdateBiometricDataResponsePackage" nillable="true">
3963         <xsd:complexType>
3964           <xsd:complexContent>
3965             <xsd:extension base="tns:ResponseTemplate"/>
3966           </xsd:complexContent>
3967         </xsd:complexType>
3968       </xsd:element>
3969     </xsd:sequence>
3970   </xsd:complexType>
3971 </xsd:element>
3972 <xsd:element name="UpdateDocumentData">
3973   <xsd:complexType>
3974     <xsd:annotation>
3975       <xsd:documentation>
3976         Updates the document data for a given subject record.
3977       </xsd:documentation>
3978     </xsd:annotation>
3979   <xsd:sequence>

```

```

3980 <xsd:element name="UpdateDocumentDataRequest" nillable="true">
3981 <xsd:complexType>
3982 <xsd:complexContent>
3983 <xsd:extension base="tns:RequestTemplate">
3984 <xsd:sequence>
3985 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
3986 <xsd:annotation>
3987 <xsd:documentation>Identifies the subject or, in the encounter-centric model, a subject and an encounter, and includes the document data to update.</xsd:documentation>
3988 </xsd:annotation>
3989 </xsd:element>
3990 <xsd:element name="DocumentDataList" type="tns:DocumentDataListType" minOccurs="1" maxOccurs="1">
3991 <xsd:annotation>
3992 <xsd:documentation>Defines a set of updated document data elements to associate with the subject or encounter. </xsd:documentation>
3993 </xsd:annotation>
3994 </xsd:element>
3995 </xsd:sequence>
3996 </xsd:extension>
3997 </xsd:complexContent>
3998 </xsd:complexType>
3999 </xsd:element>
4000 </xsd:sequence>
4001 </xsd:complexType>
4002 </xsd:element>
4003 <xsd:element name="UpdateDocumentDataResponse">
4004 <xsd:complexType>
4005 <xsd:annotation>
4006 <xsd:documentation>
4007 The response to an UpdateDocumentData request.
4008 </xsd:documentation>
4009 </xsd:annotation>
4010 <xsd:sequence>
4011 <xsd:element name="UpdateDocumentDataResponsePackage" nillable="true">
4012 <xsd:complexType>
4013 <xsd:complexContent>

```

```

4014     <xsd:extension base="tns:ResponseTemplate"/>
4015     </xsd:complexContent>
4016 </xsd:complexType>
4017 </xsd:element>
4018 </xsd:sequence>
4019 </xsd:complexType>
4020 </xsd:element>
4021 <xsd:element name="VerifySubject">
4022   <xsd:complexType>
4023     <xsd:annotation>
4024       <xsd:documentation>
4025         Performs a 1:1 verification match between a given biometric and
4026         either a claim to identity in a given gallery or another given
4027         biometric.
4028       </xsd:documentation>
4029     </xsd:annotation>
4030     <xsd:sequence>
4031       <xsd:element name="VerifySubjectRequest" nillable="true">
4032         <xsd:complexType>
4033           <xsd:complexContent>
4034             <xsd:extension base="tns:RequestTemplate">
4035               <xsd:sequence>
4036                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
4037                   <xsd:annotation>
4038                     <xsd:documentation>The identifier of the gallery or population group of which the subject must be a member.</xsd:documentation>
4039                   </xsd:annotation>
4040                 </xsd:element>
4041                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4042                   <xsd:annotation>
4043                     <xsd:documentation>Includes the identifying information and/or input and reference biometric samples.</xsd:documentation>
4044                   </xsd:annotation>
4045                 </xsd:element>
4046               </xsd:sequence>
4047             </xsd:extension>

```



```

4048     </xsd:complexContent>
4049     </xsd:complexType>
4050   </xsd:element>
4051 </xsd:sequence>
4052 </xsd:complexType>
4053 </xsd:element>
4054 <xsd:element name="VerifySubjectResponse">
4055   <xsd:complexType>
4056     <xsd:annotation>
4057       <xsd:documentation>
4058         The response to a VerifySubject request.
4059       </xsd:documentation>
4060     </xsd:annotation>
4061     <xsd:sequence>
4062       <xsd:element name="VerifySubjectResponsePackage" nillable="true">
4063         <xsd:complexType>
4064           <xsd:complexContent>
4065             <xsd:extension base="tns:ResponseTemplate">
4066               <xsd:sequence>
4067                 <xsd:element name="Match" type="xsd:boolean" minOccurs="1" maxOccurs="1">
4068                   <xsd:annotation>
4069                     <xsd:documentation>Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR.</xsd:documentation>
4070                   </xsd:annotation>
4071                 </xsd:element>
4072                 <xsd:element name="Score" type="tns:ScoreType" minOccurs="0" maxOccurs="1">
4073                   <xsd:annotation>
4074                     <xsd:documentation>The score if the biometric information matched.</xsd:documentation>
4075                   </xsd:annotation>
4076                 </xsd:element>
4077               </xsd:sequence>
4078             </xsd:extension>
4079           </xsd:complexContent>
4080         </xsd:complexType>
4081       </xsd:element>

```

```

4082     </xsd:sequence>
4083 </xsd:complexType>
4084 </xsd:element>
4085 <xsd:element name="Delete">
4086   <xsd:complexType>
4087     <xsd:annotation>
4088       <xsd:documentation>
4089         The Delete operation deletes an existing subject or, in an encounter-centric model, an existing encounter from the
4090         system.
4091       </xsd:documentation>
4092     </xsd:annotation>
4093     <xsd:sequence>
4094       <xsd:element name="DeleteRequest" nillable="true">
4095         <xsd:complexType>
4096           <xsd:complexContent>
4097             <xsd:extension base="tns:AggregateRequestTemplate">
4098               <xsd:sequence>
4099                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
4100                   <xsd:annotation>
4101                     <xsd:documentation>The identifier for the subject, or in encounter-centric model the encounter to be deleted; required for encounter-centric models.</xsd:documentation>
4102                   </xsd:annotation>
4103                 </xsd:element>
4104               </xsd:sequence>
4105             </xsd:extension>
4106           </xsd:complexContent>
4107         </xsd:complexType>
4108       </xsd:element>
4109     </xsd:sequence>
4110   </xsd:complexType>
4111 </xsd:element>
4112 <xsd:element name="DeleteResponse">
4113   <xsd:complexType>
4114     <xsd:annotation>
4115       <xsd:documentation>The response to an Delete request.</xsd:documentation>

```

```

4116     </xsd:annotation>
4117     <xsd:sequence>
4118     <xsd:element name="DeleteResponsePackage" nillable="true">
4119     <xsd:complexType>
4120     <xsd:complexContent>
4121     <xsd:extension base="tns:AggregateResponseTemplate">
4122     <xsd:sequence>
4123     <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4124     <xsd:annotation>
4125     <xsd:documentation>The identifier for the subject, or in encounter-centric model, the encounter being deleted; required for encounter-centric models.</xsd:documentation>
4126     </xsd:annotation>
4127     </xsd:element>
4128     <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
4129     <xsd:annotation>
4130     <xsd:documentation>A value used to retrieve the results of the Delete request.</xsd:documentation>
4131     </xsd:annotation>
4132     </xsd:element>
4133     </xsd:sequence>
4134     </xsd:extension>
4135     </xsd:complexContent>
4136     </xsd:complexType>
4137     </xsd:element>
4138     </xsd:sequence>
4139     </xsd:complexType>
4140     </xsd:element>
4141     <xsd:element name="Enroll">
4142     <xsd:complexType>
4143     <xsd:annotation>
4144     <xsd:documentation>
4145     The Enroll aggregate service adds a new subject or, in an
4146     encounter-centric model, a new encounter to the system.
4147     </xsd:documentation>
4148     </xsd:annotation>
4149     <xsd:sequence>

```

```

4150 <xsd:element name="EnrollRequest" nillable="true">
4151 <xsd:complexType>
4152 <xsd:complexContent>
4153 <xsd:extension base="tns:AggregateRequestTemplate">
4154 <xsd:sequence>
4155 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
4156 <xsd:annotation>
4157 <xsd:documentation>The identifier for the encounter; required for encounter-centric models.</xsd:documentation>
4158 </xsd:annotation>
4159 </xsd:element>
4160 </xsd:sequence>
4161 </xsd:extension>
4162 </xsd:complexContent>
4163 </xsd:complexType>
4164 </xsd:element>
4165 </xsd:sequence>
4166 </xsd:complexType>
4167 </xsd:element>
4168 <xsd:element name="EnrollResponse">
4169 <xsd:complexType>
4170 <xsd:annotation>
4171 <xsd:documentation>The response to an Enroll request.</xsd:documentation>
4172 </xsd:annotation>
4173 <xsd:sequence>
4174 <xsd:element name="EnrollResponsePackage" nillable="true">
4175 <xsd:complexType>
4176 <xsd:complexContent>
4177 <xsd:extension base="tns:AggregateResponseTemplate">
4178 <xsd:sequence>
4179 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4180 <xsd:annotation>
4181 <xsd:documentation>The identifier for the encounter; required for encounter-centric models.</xsd:documentation>
4182 </xsd:annotation>
4183 </xsd:element>

```

```

4184     <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
4185         <xsd:annotation>
4186             <xsd:documentation>A value used to retrieve the results of the Enroll request.</xsd:documentation>
4187         </xsd:annotation>
4188     </xsd:element>
4189 </xsd:sequence>
4190 </xsd:extension>
4191 </xsd:complexContent>
4192 </xsd:complexType>
4193 </xsd:element>
4194 </xsd:sequence>
4195 </xsd:complexType>
4196 </xsd:element>
4197 <xsd:element name="GetDeletionResults">
4198     <xsd:complexType>
4199         <xsd:annotation>
4200             <xsd:documentation>
4201                 The GetDeletionResults operation retrieves the deletion results for the specified token.
4202             </xsd:documentation>
4203         </xsd:annotation>
4204     <xsd:sequence>
4205         <xsd:element name="GetDeletionResultsRequest" nillable="true">
4206             <xsd:complexType>
4207                 <xsd:complexContent>
4208                     <xsd:extension base="tns:RequestTemplate">
4209                         <xsd:sequence>
4210                             <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
4211                                 <xsd:annotation>
4212                                     <xsd:documentation>A value used to retrieve the results of the Deletion request.</xsd:documentation>
4213                                 </xsd:annotation>
4214                             </xsd:element>
4215                         </xsd:sequence>
4216                     </xsd:extension>
4217                 </xsd:complexContent>

```

```

4218     </xsd:complexType>
4219 </xsd:element>
4220 </xsd:sequence>
4221 </xsd:complexType>
4222 </xsd:element>
4223 <xsd:element name="GetDeletionResultsResponse">
4224 <xsd:complexType>
4225 <xsd:annotation>
4226 <xsd:documentation>The response to a GetDeletionResults request.</xsd:documentation>
4227 </xsd:annotation>
4228 <xsd:sequence>
4229 <xsd:element name="GetDeletionResultsResponsePackage" nillable="true">
4230 <xsd:complexType>
4231 <xsd:complexContent>
4232 <xsd:extension base="tns:ResponseTemplate">
4233 <xsd:sequence>
4234 <xsd:element name="ReturnData" type="tns:InformationType" minOccurs="0" maxOccurs="1">
4235 <xsd:annotation>
4236 <xsd:documentation>Contains the output data for the response.</xsd:documentation>
4237 </xsd:annotation>
4238 </xsd:element>
4239 </xsd:sequence>
4240 </xsd:extension>
4241 </xsd:complexContent>
4242 </xsd:complexType>
4243 </xsd:element>
4244 </xsd:sequence>
4245 </xsd:complexType>
4246 </xsd:element>
4247 <xsd:element name="GetEnrollResults">
4248 <xsd:complexType>
4249 <xsd:annotation>
4250 <xsd:documentation>
4251 The GetEnrollResults aggregate service retrieves the enrollment

```

4252 results for the specified token.
 4253 </xsd:documentation>
 4254 </xsd:annotation>
 4255 <xsd:sequence>
 4256 <xsd:element name="GetEnrollResultsRequest" nillable="true">
 4257 <xsd:complexType>
 4258 <xsd:complexContent>
 4259 <xsd:extension base="tns:RequestTemplate">
 4260 <xsd:sequence>
 4261 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
 4262 <xsd:annotation>
 4263 <xsd:documentation>A value used to retrieve the results of the Enroll request.</xsd:documentation>
 4264 </xsd:annotation>
 4265 </xsd:element>
 4266 </xsd:sequence>
 4267 </xsd:extension>
 4268 </xsd:complexContent>
 4269 </xsd:complexType>
 4270 </xsd:element>
 4271 </xsd:sequence>
 4272 </xsd:complexType>
 4273 </xsd:element>
 4274 <xsd:element name="GetEnrollResultsResponse">
 4275 <xsd:complexType>
 4276 <xsd:annotation>
 4277 <xsd:documentation>The response to a GetEnrollResults request.</xsd:documentation>
 4278 </xsd:annotation>
 4279 <xsd:sequence>
 4280 <xsd:element name="GetEnrollResultsResponsePackage" nillable="true">
 4281 <xsd:complexType>
 4282 <xsd:complexContent>
 4283 <xsd:extension base="tns:AggregateResponseTemplate">
 4284 <xsd:sequence>
 4285 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">

```

4286     <xsd:annotation>
4287     <xsd:documentation>The identifier for the subject, and in encounter-centric models, the encounter.</xsd:documentation>
4288     </xsd:annotation>
4289   </xsd:element>
4290 </xsd:sequence>
4291 </xsd:extension>
4292 </xsd:complexContent>
4293 </xsd:complexType>
4294 </xsd:element>
4295 </xsd:sequence>
4296 </xsd:complexType>
4297 </xsd:element>
4298 <xsd:element name="GetIdentifyResults">
4299   <xsd:complexType>
4300     <xsd:annotation>
4301     <xsd:documentation>
4302       The GetIdentifyResults aggregate service retrieves the
4303       identification results for the specified token.
4304     </xsd:documentation>
4305   </xsd:annotation>
4306   <xsd:sequence>
4307     <xsd:element name="GetIdentifyResultsRequest" nillable="true">
4308       <xsd:complexType>
4309         <xsd:complexContent>
4310           <xsd:extension base="tns:RequestTemplate">
4311             <xsd:sequence>
4312               <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
4313                 <xsd:annotation>
4314                 <xsd:documentation>A value used to retrieve the results of the Identify request.</xsd:documentation>
4315                 </xsd:annotation>
4316               </xsd:element>
4317             </xsd:sequence>
4318           </xsd:extension>
4319         </xsd:complexContent>

```



```

4320     </xsd:complexType>
4321 </xsd:element>
4322 </xsd:sequence>
4323 </xsd:complexType>
4324 </xsd:element>
4325 <xsd:element name="GetIdentifyResultsResponse">
4326 <xsd:complexType>
4327 <xsd:annotation>
4328 <xsd:documentation>The response to a GetIdentifyResults request.</xsd:documentation>
4329 </xsd:annotation>
4330 <xsd:sequence>
4331 <xsd:element name="GetIdentifyResultsResponsePackage" nillable="true">
4332 <xsd:complexType>
4333 <xsd:complexContent>
4334 <xsd:extension base="tns:AggregateResponseTemplate">
4335 <xsd:sequence>
4336 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
4337 <xsd:annotation>
4338 <xsd:documentation>The identifier of the encounter, if assigned.</xsd:documentation>
4339 </xsd:annotation>
4340 </xsd:element>
4341 <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
4342 <xsd:annotation>
4343 <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching the input biometric sample.</xsd:documentation>
4344 </xsd:annotation>
4345 </xsd:element>
4346 </xsd:sequence>
4347 </xsd:extension>
4348 </xsd:complexContent>
4349 </xsd:complexType>
4350 </xsd:element>
4351 </xsd:sequence>
4352 </xsd:complexType>
4353 </xsd:element>

```

```

4354 <xsd:element name="GetUpdateResults">
4355   <xsd:complexType>
4356     <xsd:annotation>
4357       <xsd:documentation>
4358         The GetUpdateResults operation retrieves the update results for the specified token.
4359       </xsd:documentation>
4360     </xsd:annotation>
4361     <xsd:sequence>
4362       <xsd:element name="GetUpdateResultsRequest" nillable="true">
4363         <xsd:complexType>
4364           <xsd:complexContent>
4365             <xsd:extension base="tns:RequestTemplate">
4366               <xsd:sequence>
4367                 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
4368                   <xsd:annotation>
4369                     <xsd:documentation>A value used to retrieve the results of the Update request.</xsd:documentation>
4370                   </xsd:annotation>
4371                 </xsd:element>
4372               </xsd:sequence>
4373             </xsd:extension>
4374           </xsd:complexContent>
4375         </xsd:complexType>
4376       </xsd:element>
4377     </xsd:sequence>
4378   </xsd:complexType>
4379 </xsd:element>
4380 <xsd:element name="GetUpdateResultsResponse">
4381   <xsd:complexType>
4382     <xsd:annotation>
4383       <xsd:documentation>The response to a GetUpdateResults request.</xsd:documentation>
4384     </xsd:annotation>
4385     <xsd:sequence>
4386       <xsd:element name="GetUpdateResultsResponsePackage" nillable="true">
4387         <xsd:complexType>

```

```

4388     <xsd:complexContent>
4389     <xsd:extension base="tns:AggregateResponseTemplate">
4390     <xsd:sequence>
4391     </xsd:sequence>
4392     </xsd:extension>
4393     </xsd:complexContent>
4394     </xsd:complexType>
4395   </xsd:element>
4396 </xsd:sequence>
4397 </xsd:complexType>
4398 </xsd:element>
4399 <xsd:element name="GetVerifyResults">
4400 <xsd:complexType>
4401 <xsd:annotation>
4402 <xsd:documentation>
4403   The GetVerifyResults aggregate service retrieves the verification results for the specified token.
4404 </xsd:documentation>
4405 </xsd:annotation>
4406 <xsd:sequence>
4407 <xsd:element name="GetVerifyResultsRequest" nillable="true">
4408 <xsd:complexType>
4409 <xsd:complexContent>
4410 <xsd:extension base="tns:RequestTemplate">
4411 <xsd:sequence>
4412 <xsd:element name="Token" type="tns:TokenType" minOccurs="1" maxOccurs="1">
4413 <xsd:annotation>
4414 <xsd:documentation>A value used to retrieve the results of the Verify request.</xsd:documentation>
4415 </xsd:annotation>
4416 </xsd:element>
4417 </xsd:sequence>
4418 </xsd:extension>
4419 </xsd:complexContent>
4420 </xsd:complexType>
4421 </xsd:element>

```

```

4422     </xsd:sequence>
4423 </xsd:complexType>
4424 </xsd:element>
4425 <xsd:element name="GetVerifyResultsResponse">
4426 <xsd:complexType>
4427 <xsd:annotation>
4428 <xsd:documentation>The response to a GetVerifyResults request.</xsd:documentation>
4429 </xsd:annotation>
4430 <xsd:sequence>
4431 <xsd:element name="GetVerifyResultsResponsePackage" nillable="true">
4432 <xsd:complexType>
4433 <xsd:complexContent>
4434 <xsd:extension base="tns:AggregateResponseTemplate">
4435 <xsd:sequence>
4436 <xsd:element name="Match" type="xsd:boolean" minOccurs="0" maxOccurs="1">
4437 <xsd:annotation>
4438 <xsd:documentation>Indicates if the Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR.</xsd:documentation>
4439 </xsd:annotation>
4440 </xsd:element>
4441 <xsd:element name="Score" type="tns:ScoreType" minOccurs="0" maxOccurs="1">
4442 <xsd:annotation>
4443 <xsd:documentation>The score if the biometric information matched.</xsd:documentation>
4444 </xsd:annotation>
4445 </xsd:element>
4446 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
4447 <xsd:annotation>
4448 <xsd:documentation>The identifier of the encounter, if assigned.</xsd:documentation>
4449 </xsd:annotation>
4450 </xsd:element>
4451 </xsd:sequence>
4452 </xsd:extension>
4453 </xsd:complexContent>
4454 </xsd:complexType>
4455 </xsd:element>

```

```

4456     </xsd:sequence>
4457 </xsd:complexType>
4458 </xsd:element>
4459 <xsd:element name="Identify">
4460   <xsd:complexType>
4461     <xsd:annotation>
4462       <xsd:documentation>
4463         The Identify aggregate service performs an identification function according to system requirements and/or resources.
4464       </xsd:documentation>
4465     </xsd:annotation>
4466     <xsd:sequence>
4467       <xsd:element name="IdentifyRequest" nillable="true">
4468         <xsd:complexType>
4469           <xsd:complexContent>
4470             <xsd:extension base="tns:AggregateRequestTemplate">
4471               <xsd:sequence>
4472                 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
4473                   <xsd:annotation>
4474                     <xsd:documentation>The identifier of the gallery or population group which will be searched.</xsd:documentation>
4475                   </xsd:annotation>
4476                 </xsd:element>
4477                 <xsd:element name="MaxListSize" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
4478                   <xsd:annotation>
4479                     <xsd:documentation>The maximum size of the candidate list that should be returned.</xsd:documentation>
4480                   </xsd:annotation>
4481                 </xsd:element>
4482               </xsd:sequence>
4483             </xsd:extension>
4484           </xsd:complexContent>
4485         </xsd:complexType>
4486       </xsd:element>
4487     </xsd:sequence>
4488   </xsd:complexType>
4489 </xsd:element>

```

```

4490 <xsd:element name="IdentifyResponse">
4491   <xsd:complexType>
4492     <xsd:annotation>
4493       <xsd:documentation>The response to an Identify request.</xsd:documentation>
4494     </xsd:annotation>
4495     <xsd:sequence>
4496       <xsd:element name="IdentifyResponsePackage" nillable="true">
4497         <xsd:complexType>
4498           <xsd:complexContent>
4499             <xsd:extension base="tns:AggregateResponseTemplate">
4500               <xsd:sequence>
4501                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4502                   <xsd:annotation>
4503                     <xsd:documentation>The identifier of the encounter, if assigned.</xsd:documentation>
4504                   </xsd:annotation>
4505                 </xsd:element>
4506                 <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
4507                   <xsd:annotation>
4508                     <xsd:documentation>A rank-ordered list of candidates that have a likelihood of matching the input biometric sample;
4509                       returned with successful, synchronous processing.
4510                   </xsd:documentation>
4511                 </xsd:annotation>
4512                 </xsd:element>
4513                 <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
4514                   <xsd:annotation>
4515                     <xsd:documentation>A value used to retrieve the results of the Identify request.</xsd:documentation>
4516                   </xsd:annotation>
4517                 </xsd:element>
4518               </xsd:sequence>
4519             </xsd:extension>
4520           </xsd:complexContent>
4521         </xsd:complexType>
4522       </xsd:element>
4523     </xsd:sequence>

```

```

4524     </xsd:complexType>
4525 </xsd:element>
4526 <xsd:element name="RetrieveData">
4527     <xsd:complexType>
4528     <xsd:annotation>
4529     <xsd:documentation>
4530         The RetrieveData aggregate service retrieves requested information about a subject, or in an encounter-centric model
4531         about an encounter
4532     </xsd:documentation>
4533 </xsd:annotation>
4534     <xsd:sequence>
4535     <xsd:element name="RetrieveDataRequest" nillable="true">
4536     <xsd:complexType>
4537     <xsd:complexContent>
4538     <xsd:extension base="tns:RequestTemplate">
4539     <xsd:sequence>
4540     <xsd:element name="ProcessingOptions" type="tns:ProcessingOptionsType" minOccurs="1" maxOccurs="1">
4541     <xsd:annotation>
4542     <xsd:documentation>Options that guide how the service request is processed, and may identify what type(s) of information should be returned.</xsd:documentation>
4543 </xsd:annotation>
4544 </xsd:element>
4545 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4546 <xsd:annotation>
4547 <xsd:documentation>Includes the identifier of the subject or encounter.</xsd:documentation>
4548 </xsd:annotation>
4549 </xsd:element>
4550 </xsd:sequence>
4551 </xsd:extension>
4552 </xsd:complexContent>
4553 </xsd:complexType>
4554 </xsd:element>
4555 </xsd:sequence>
4556 </xsd:complexType>
4557 </xsd:element>

```

```

4558 <xsd:element name="RetrieveDataResponse">
4559   <xsd:complexType>
4560     <xsd:annotation>
4561       <xsd:documentation>The response to a RetrieveData request.</xsd:documentation>
4562     </xsd:annotation>
4563     <xsd:sequence>
4564       <xsd:element name="RetrieveDataResponsePackage" nillable="true">
4565         <xsd:complexType>
4566           <xsd:complexContent>
4567             <xsd:extension base="tns:AggregateResponseTemplate"/>
4568           </xsd:complexContent>
4569         </xsd:complexType>
4570       </xsd:element>
4571     </xsd:sequence>
4572   </xsd:complexType>
4573 </xsd:element>
4574 <xsd:element name="Update">
4575   <xsd:complexType>
4576     <xsd:annotation>
4577       <xsd:documentation>
4578         The Update operation updates specified information about a subject, or in an encounter-centric model about an encounter.
4579       </xsd:documentation>
4580     </xsd:annotation>
4581     <xsd:sequence>
4582       <xsd:element name="UpdateRequest" nillable="true">
4583         <xsd:complexType>
4584           <xsd:complexContent>
4585             <xsd:extension base="tns:AggregateRequestTemplate">
4586               <xsd:sequence>
4587                 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4588                   <xsd:annotation>
4589                     <xsd:documentation>Includes the identifier of the subject or encounter.</xsd:documentation>
4590                   </xsd:annotation>
4591                 </xsd:element>

```



```

4592     </xsd:sequence>
4593     </xsd:extension>
4594     </xsd:complexContent>
4595     </xsd:complexType>
4596   </xsd:element>
4597 </xsd:sequence>
4598 </xsd:complexType>
4599 </xsd:element>
4600 <xsd:element name="UpdateResponse">
4601   <xsd:complexType>
4602     <xsd:annotation>
4603       <xsd:documentation>The response to a Update request.</xsd:documentation>
4604     </xsd:annotation>
4605     <xsd:sequence>
4606       <xsd:element name="UpdateResponsePackage" nillable="true">
4607         <xsd:complexType>
4608           <xsd:complexContent>
4609             <xsd:extension base="tns:AggregateResponseTemplate">
4610               <xsd:sequence>
4611                 <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
4612                   <xsd:annotation>
4613                     <xsd:documentation>A value used to retrieve the results of the Update request.</xsd:documentation>
4614                   </xsd:annotation>
4615                 </xsd:element>
4616               </xsd:sequence>
4617             </xsd:extension>
4618           </xsd:complexContent>
4619         </xsd:complexType>
4620       </xsd:element>
4621     </xsd:sequence>
4622   </xsd:complexType>
4623 </xsd:element>
4624 <xsd:element name="Verify">
4625   <xsd:complexType>

```

```

4626 <xsd:annotation>
4627 <xsd:documentation>
4628     The Verify aggregate service performs a 1:1 verification
4629     function according to system requirements and/or resources.
4630     Either the IdentityClaim or ReferenceBIR input data elements in the Identity parameter are required.
4631 </xsd:documentation>
4632 </xsd:annotation>
4633 <xsd:sequence>
4634 <xsd:element name="VerifyRequest" nillable="true">
4635 <xsd:complexType>
4636 <xsd:complexContent>
4637 <xsd:extension base="tns:AggregateRequestTemplate">
4638 <xsd:sequence>
4639 <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="1" maxOccurs="1">
4640 <xsd:annotation>
4641 <xsd:documentation>Includes either the IdentityClaim or ReferenceBIR.</xsd:documentation>
4642 </xsd:annotation>
4643 </xsd:element>
4644 <xsd:element name="GalleryID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
4645 <xsd:annotation>
4646 <xsd:documentation>The identifier of the gallery or population group of which the subject must be a member.</xsd:documentation>
4647 </xsd:annotation>
4648 </xsd:element>
4649 </xsd:sequence>
4650 </xsd:extension>
4651 </xsd:complexContent>
4652 </xsd:complexType>
4653 </xsd:element>
4654 </xsd:sequence>
4655 </xsd:complexType>
4656 </xsd:element>
4657 <xsd:element name="VerifyResponse">
4658 <xsd:complexType>
4659 <xsd:annotation>

```

```

4660     <xsd:documentation>The response to a Verify request.</xsd:documentation>
4661 </xsd:annotation>
4662 <xsd:sequence>
4663   <xsd:element name="VerifyResponsePackage" nillable="true">
4664     <xsd:complexType>
4665       <xsd:complexContent>
4666         <xsd:extension base="tns:AggregateResponseTemplate">
4667           <xsd:sequence>
4668             <xsd:element name="Match" type="xsd:boolean" minOccurs="0" maxOccurs="1">
4669               <xsd:annotation>
4670                 <xsd:documentation>Indicates if the Input BIR matched either the Identity Claim or the Reference BIR.</xsd:documentation>
4671               </xsd:annotation>
4672             </xsd:element>
4673             <xsd:element name="Score" type="tns:ScoreType" minOccurs="0" maxOccurs="1">
4674               <xsd:annotation>
4675                 <xsd:documentation>The score if the biometric information matched.</xsd:documentation>
4676               </xsd:annotation>
4677             </xsd:element>
4678             <xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">
4679               <xsd:annotation>
4680                 <xsd:documentation>The identifier of the encounter, if assigned.</xsd:documentation>
4681               </xsd:annotation>
4682             </xsd:element>
4683             <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
4684               <xsd:annotation>
4685                 <xsd:documentation>A value used to retrieve the results of the Verify request; </xsd:documentation>
4686               </xsd:annotation>
4687             </xsd:element>
4688           </xsd:sequence>
4689         </xsd:extension>
4690       </xsd:complexContent>
4691     </xsd:complexType>
4692   </xsd:element>
4693 </xsd:sequence>

```

```

4694     </xsd:complexType>
4695 </xsd:element>
4696 <xsd:element name="BIASFault" type="tns:BIASFaultDetail"/>
4697 </xsd:schema>
4698 </wsdl:types>
4699 <wsdl:message name="BIASFaultMessage">
4700 <wsdl:part name="parameters" element="tns:BIASFault"/>
4701 </wsdl:message>
4702 <wsdl:message name="AddSubjectToGalleryRequestMessage">
4703 <wsdl:part name="parameters" element="tns:AddSubjectToGallery"/>
4704 </wsdl:message>
4705 <wsdl:message name="AddSubjectToGalleryResponseMessage">
4706 <wsdl:part name="parameters" element="tns:AddSubjectToGalleryResponse"/>
4707 </wsdl:message>
4708 <wsdl:message name="CheckQualityRequestMessage">
4709 <wsdl:part name="parameters" element="tns:CheckQuality"/>
4710 </wsdl:message>
4711 <wsdl:message name="CheckQualityResponseMessage">
4712 <wsdl:part name="parameters" element="tns:CheckQualityResponse"/>
4713 </wsdl:message>
4714 <wsdl:message name="ClassifyBiometricDataRequestMessage">
4715 <wsdl:part name="parameters" element="tns:ClassifyBiometricData"/>
4716 </wsdl:message>
4717 <wsdl:message name="ClassifyBiometricDataResponseMessage">
4718 <wsdl:part name="parameters" element="tns:ClassifyBiometricDataResponse"/>
4719 </wsdl:message>
4720 <wsdl:message name="CreateSubjectRequestMessage">
4721 <wsdl:part name="parameters" element="tns:CreateSubject"/>
4722 </wsdl:message>
4723 <wsdl:message name="CreateSubjectResponseMessage">
4724 <wsdl:part name="parameters" element="tns:CreateSubjectResponse"/>
4725 </wsdl:message>
4726 <wsdl:message name="DeleteBiographicDataRequestMessage">
4727 <wsdl:part name="parameters" element="tns:DeleteBiographicData"/>

```

```

4728 </wsdl:message>
4729 <wsdl:message name="DeleteBiographicDataResponseMessage">
4730 <wsdl:part name="parameters" element="tns:DeleteBiographicDataResponse"/>
4731 </wsdl:message>
4732 <wsdl:message name="DeleteBiometricDataRequestMessage">
4733 <wsdl:part name="parameters" element="tns:DeleteBiometricData"/>
4734 </wsdl:message>
4735 <wsdl:message name="DeleteBiometricDataResponseMessage">
4736 <wsdl:part name="parameters" element="tns:DeleteBiometricDataResponse"/>
4737 </wsdl:message>
4738 <wsdl:message name="DeleteSubjectRequestMessage">
4739 <wsdl:part name="parameters" element="tns:DeleteSubject"/>
4740 </wsdl:message>
4741 <wsdl:message name="DeleteSubjectResponseMessage">
4742 <wsdl:part name="parameters" element="tns:DeleteSubjectResponse"/>
4743 </wsdl:message>
4744 <wsdl:message name="DeleteSubjectFromGalleryRequestMessage">
4745 <wsdl:part name="parameters" element="tns:DeleteSubjectFromGallery"/>
4746 </wsdl:message>
4747 <wsdl:message name="DeleteSubjectFromGalleryResponseMessage">
4748 <wsdl:part name="parameters" element="tns:DeleteSubjectFromGalleryResponse"/>
4749 </wsdl:message>
4750 <wsdl:message name="GetIdentifySubjectResultsRequestMessage">
4751 <wsdl:part name="parameters" element="tns:GetIdentifySubjectResults"/>
4752 </wsdl:message>
4753 <wsdl:message name="GetIdentifySubjectResultsResponseMessage">
4754 <wsdl:part name="parameters" element="tns:GetIdentifySubjectResultsResponse"/>
4755 </wsdl:message>
4756 <wsdl:message name="IdentifySubjectRequestMessage">
4757 <wsdl:part name="parameters" element="tns:IdentifySubject"/>
4758 </wsdl:message>
4759 <wsdl:message name="IdentifySubjectResponseMessage">
4760 <wsdl:part name="parameters" element="tns:IdentifySubjectResponse"/>
4761 </wsdl:message>

```

```

4762 <wsdl:message name="ListBiographicDataRequestMessage">
4763   <wsdl:part name="parameters" element="tns:ListBiographicData"/>
4764 </wsdl:message>
4765 <wsdl:message name="ListBiographicDataResponseMessage">
4766   <wsdl:part name="parameters" element="tns:ListBiographicDataResponse"/>
4767 </wsdl:message>
4768 <wsdl:message name="ListBiometricDataRequestMessage">
4769   <wsdl:part name="parameters" element="tns:ListBiometricData"/>
4770 </wsdl:message>
4771 <wsdl:message name="ListBiometricDataResponseMessage">
4772   <wsdl:part name="parameters" element="tns:ListBiometricDataResponse"/>
4773 </wsdl:message>
4774 <wsdl:message name="PerformFusionRequestMessage">
4775   <wsdl:part name="parameters" element="tns:PerformFusion"/>
4776 </wsdl:message>
4777 <wsdl:message name="PerformFusionResponseMessage">
4778   <wsdl:part name="parameters" element="tns:PerformFusionResponse"/>
4779 </wsdl:message>
4780 <wsdl:message name="QueryCapabilitiesRequestMessage">
4781   <wsdl:part name="parameters" element="tns:QueryCapabilities"/>
4782 </wsdl:message>
4783 <wsdl:message name="QueryCapabilitiesResponseMessage">
4784   <wsdl:part name="parameters" element="tns:QueryCapabilitiesResponse"/>
4785 </wsdl:message>
4786 <wsdl:message name="RetrieveBiographicDataRequestMessage">
4787   <wsdl:part name="parameters" element="tns:RetrieveBiographicData"/>
4788 </wsdl:message>
4789 <wsdl:message name="RetrieveBiographicDataResponseMessage">
4790   <wsdl:part name="parameters" element="tns:RetrieveBiographicDataResponse"/>
4791 </wsdl:message>
4792 <wsdl:message name="RetrieveBiometricDataRequestMessage">
4793   <wsdl:part name="parameters" element="tns:RetrieveBiometricData"/>
4794 </wsdl:message>
4795 <wsdl:message name="RetrieveBiometricDataResponseMessage">

```

```

4796     <wsdl:part name="parameters" element="tns:RetrieveBiometricDataResponse"/>
4797 </wsdl:message>
4798 <wsdl:message name="SetBiographicDataRequestMessage">
4799     <wsdl:part name="parameters" element="tns:SetBiographicData"/>
4800 </wsdl:message>
4801 <wsdl:message name="SetBiographicDataResponseMessage">
4802     <wsdl:part name="parameters" element="tns:SetBiographicDataResponse"/>
4803 </wsdl:message>
4804 <wsdl:message name="SetBiometricDataRequestMessage">
4805     <wsdl:part name="parameters" element="tns:SetBiometricData"/>
4806 </wsdl:message>
4807 <wsdl:message name="SetBiometricDataResponseMessage">
4808     <wsdl:part name="parameters" element="tns:SetBiometricDataResponse"/>
4809 </wsdl:message>
4810 <wsdl:message name="TransformBiometricDataRequestMessage">
4811     <wsdl:part name="parameters" element="tns:TransformBiometricData"/>
4812 </wsdl:message>
4813 <wsdl:message name="TransformBiometricDataResponseMessage">
4814     <wsdl:part name="parameters" element="tns:TransformBiometricDataResponse"/>
4815 </wsdl:message>
4816 <wsdl:message name="UpdateBiographicDataRequestMessage">
4817     <wsdl:part name="parameters" element="tns:UpdateBiographicData"/>
4818 </wsdl:message>
4819 <wsdl:message name="UpdateBiographicDataResponseMessage">
4820     <wsdl:part name="parameters" element="tns:UpdateBiographicDataResponse"/>
4821 </wsdl:message>
4822 <wsdl:message name="UpdateBiometricDataRequestMessage">
4823     <wsdl:part name="parameters" element="tns:UpdateBiometricData"/>
4824 </wsdl:message>
4825 <wsdl:message name="UpdateBiometricDataResponseMessage">
4826     <wsdl:part name="parameters" element="tns:UpdateBiometricDataResponse"/>
4827 </wsdl:message>
4828 <wsdl:message name="VerifySubjectRequestMessage">
4829     <wsdl:part name="parameters" element="tns:VerifySubject"/>

```

```

4830 </wsdl:message>
4831 <wsdl:message name="VerifySubjectResponseMessage">
4832   <wsdl:part name="parameters" element="tns:VerifySubjectResponse"/>
4833 </wsdl:message>
4834 <wsdl:message name="EnrollRequestMessage">
4835   <wsdl:part name="parameters" element="tns:Enroll"/>
4836 </wsdl:message>
4837 <wsdl:message name="EnrollResponseMessage">
4838   <wsdl:part name="parameters" element="tns:EnrollResponse"/>
4839 </wsdl:message>
4840 <wsdl:message name="GetEnrollResultsRequestMessage">
4841   <wsdl:part name="parameters" element="tns:GetEnrollResults"/>
4842 </wsdl:message>
4843 <wsdl:message name="GetEnrollResultsResponseMessage">
4844   <wsdl:part name="parameters" element="tns:GetEnrollResultsResponse"/>
4845 </wsdl:message>
4846 <wsdl:message name="GetIdentifyResultsRequestMessage">
4847   <wsdl:part name="parameters" element="tns:GetIdentifyResults"/>
4848 </wsdl:message>
4849 <wsdl:message name="GetIdentifyResultsResponseMessage">
4850   <wsdl:part name="parameters" element="tns:GetIdentifyResultsResponse"/>
4851 </wsdl:message>
4852 <wsdl:message name="GetVerifyResultsRequestMessage">
4853   <wsdl:part name="parameters" element="tns:GetVerifyResults"/>
4854 </wsdl:message>
4855 <wsdl:message name="GetVerifyResultsResponseMessage">
4856   <wsdl:part name="parameters" element="tns:GetVerifyResultsResponse"/>
4857 </wsdl:message>
4858 <wsdl:message name="IdentifyRequestMessage">
4859   <wsdl:part name="parameters" element="tns:Identify"/>
4860 </wsdl:message>
4861 <wsdl:message name="IdentifyResponseMessage">
4862   <wsdl:part name="parameters" element="tns:IdentifyResponse"/>
4863 </wsdl:message>

```



```

4864 <wsdl:message name="RetrieveDataRequestMessage">
4865   <wsdl:part name="parameters" element="tns:RetrieveData"/>
4866 </wsdl:message>
4867 <wsdl:message name="RetrieveDataResponseMessage">
4868   <wsdl:part name="parameters" element="tns:RetrieveDataResponse"/>
4869 </wsdl:message>
4870 <wsdl:message name="VerifyRequestMessage">
4871   <wsdl:part name="parameters" element="tns:Verify"/>
4872 </wsdl:message>
4873 <wsdl:message name="VerifyResponseMessage">
4874   <wsdl:part name="parameters" element="tns:VerifyResponse"/>
4875 </wsdl:message>
4876 <wsdl:portType name="BIAS_v1">
4877   <wsdl:operation name="AddSubjectToGallery">
4878     <wsdl:input name="AddSubjectToGallery" message="tns:AddSubjectToGalleryRequestMessage"/>
4879     <wsdl:output name="AddSubjectToGalleryResponse" message="tns:AddSubjectToGalleryResponseMessage"/>
4880     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4881   </wsdl:operation>
4882   <wsdl:operation name="CheckQuality">
4883     <wsdl:input name="CheckQuality" message="tns:CheckQualityRequestMessage"/>
4884     <wsdl:output name="CheckQualityResponse" message="tns:CheckQualityResponseMessage"/>
4885     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4886   </wsdl:operation>
4887   <wsdl:operation name="ClassifyBiometricData">
4888     <wsdl:input name="ClassifyBiometricData" message="tns:ClassifyBiometricDataRequestMessage"/>
4889     <wsdl:output name="ClassifyBiometricDataResponse" message="tns:ClassifyBiometricDataResponseMessage"/>
4890     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4891   </wsdl:operation>
4892   <wsdl:operation name="CreateSubject">
4893     <wsdl:input name="CreateSubject" message="tns:CreateSubjectRequestMessage"/>
4894     <wsdl:output name="CreateSubjectResponse" message="tns:CreateSubjectResponseMessage"/>
4895     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4896   </wsdl:operation>
4897   <wsdl:operation name="DeleteBiographicData">

```

```

4898 <wsdl:input name="DeleteBiographicData" message="tns:DeleteBiographicDataRequestMessage"/>
4899 <wsdl:output name="DeleteBiographicDataResponse" message="tns:DeleteBiographicDataResponseMessage"/>
4900 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4901 </wsdl:operation>
4902 <wsdl:operation name="DeleteBiometricData">
4903 <wsdl:input name="DeleteBiometricData" message="tns:DeleteBiometricDataRequestMessage"/>
4904 <wsdl:output name="DeleteBiometricDataResponse" message="tns:DeleteBiometricDataResponseMessage"/>
4905 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4906 </wsdl:operation>
4907 <wsdl:operation name="DeleteSubject">
4908 <wsdl:input name="DeleteSubject" message="tns:DeleteSubjectRequestMessage"/>
4909 <wsdl:output name="DeleteSubjectResponse" message="tns:DeleteSubjectResponseMessage"/>
4910 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4911 </wsdl:operation>
4912 <wsdl:operation name="DeleteSubjectFromGallery">
4913 <wsdl:input name="DeleteSubjectFromGallery" message="tns:DeleteSubjectFromGalleryRequestMessage"/>
4914 <wsdl:output name="DeleteSubjectFromGalleryResponse" message="tns:DeleteSubjectFromGalleryResponseMessage"/>
4915 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4916 </wsdl:operation>
4917 <wsdl:operation name="GetIdentifySubjectResults">
4918 <wsdl:input name="GetIdentifySubjectResults" message="tns:GetIdentifySubjectResultsRequestMessage"/>
4919 <wsdl:output name="GetIdentifySubjectResultsResponse" message="tns:GetIdentifySubjectResultsResponseMessage"/>
4920 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4921 </wsdl:operation>
4922 <wsdl:operation name="IdentifySubject">
4923 <wsdl:input name="IdentifySubject" message="tns:IdentifySubjectRequestMessage"/>
4924 <wsdl:output name="IdentifySubjectResponse" message="tns:IdentifySubjectResponseMessage"/>
4925 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4926 </wsdl:operation>
4927 <wsdl:operation name="ListBiographicData">
4928 <wsdl:input name="ListBiographicData" message="tns:ListBiographicDataRequestMessage"/>
4929 <wsdl:output name="ListBiographicDataResponse" message="tns:ListBiographicDataResponseMessage"/>
4930 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4931 </wsdl:operation>

```

```

4932 <wsdl:operation name="ListBiometricData">
4933 <wsdl:input name="ListBiometricData" message="tns:ListBiometricDataRequestMessage"/>
4934 <wsdl:output name="ListBiometricDataResponse" message="tns:ListBiometricDataResponseMessage"/>
4935 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4936 </wsdl:operation>
4937 <wsdl:operation name="PerformFusion">
4938 <wsdl:input name="PerformFusion" message="tns:PerformFusionRequestMessage"/>
4939 <wsdl:output name="PerformFusionResponse" message="tns:PerformFusionResponseMessage"/>
4940 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4941 </wsdl:operation>
4942 <wsdl:operation name="QueryCapabilities">
4943 <wsdl:input name="QueryCapabilities" message="tns:QueryCapabilitiesRequestMessage"/>
4944 <wsdl:output name="QueryCapabilitiesResponse" message="tns:QueryCapabilitiesResponseMessage"/>
4945 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4946 </wsdl:operation>
4947 <wsdl:operation name="RetrieveBiographicData">
4948 <wsdl:input name="RetrieveBiographicData" message="tns:RetrieveBiographicDataRequestMessage"/>
4949 <wsdl:output name="RetrieveBiographicDataResponse" message="tns:RetrieveBiographicDataResponseMessage"/>
4950 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4951 </wsdl:operation>
4952 <wsdl:operation name="RetrieveBiometricData">
4953 <wsdl:input name="RetrieveBiometricData" message="tns:RetrieveBiometricDataRequestMessage"/>
4954 <wsdl:output name="RetrieveBiometricDataResponse" message="tns:RetrieveBiometricDataResponseMessage"/>
4955 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4956 </wsdl:operation>
4957 <wsdl:operation name="SetBiographicData">
4958 <wsdl:input name="SetBiographicData" message="tns:SetBiographicDataRequestMessage"/>
4959 <wsdl:output name="SetBiographicDataResponse" message="tns:SetBiographicDataResponseMessage"/>
4960 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4961 </wsdl:operation>
4962 <wsdl:operation name="SetBiometricData">
4963 <wsdl:input name="SetBiometricData" message="tns:SetBiometricDataRequestMessage"/>
4964 <wsdl:output name="SetBiometricDataResponse" message="tns:SetBiometricDataResponseMessage"/>
4965 <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>

```

```

4966 </wsdl:operation>
4967 <wsdl:operation name="TransformBiometricData">
4968   <wsdl:input name="TransformBiometricData" message="tns:TransformBiometricDataRequestMessage"/>
4969   <wsdl:output name="TransformBiometricDataResponse" message="tns:TransformBiometricDataResponseMessage"/>
4970   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4971 </wsdl:operation>
4972 <wsdl:operation name="UpdateBiographicData">
4973   <wsdl:input name="UpdateBiographicData" message="tns:UpdateBiographicDataRequestMessage"/>
4974   <wsdl:output name="UpdateBiographicDataResponse" message="tns:UpdateBiographicDataResponseMessage"/>
4975   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4976 </wsdl:operation>
4977 <wsdl:operation name="UpdateBiometricData">
4978   <wsdl:input name="UpdateBiometricData" message="tns:UpdateBiometricDataRequestMessage"/>
4979   <wsdl:output name="UpdateBiometricDataResponse" message="tns:UpdateBiometricDataResponseMessage"/>
4980   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4981 </wsdl:operation>
4982 <wsdl:operation name="VerifySubject">
4983   <wsdl:input name="VerifySubject" message="tns:VerifySubjectRequestMessage"/>
4984   <wsdl:output name="VerifySubjectResponse" message="tns:VerifySubjectResponseMessage"/>
4985   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4986 </wsdl:operation>
4987 <wsdl:operation name="Enroll">
4988   <wsdl:input name="Enroll" message="tns:EnrollRequestMessage"/>
4989   <wsdl:output name="EnrollResponse" message="tns:EnrollResponseMessage"/>
4990   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4991 </wsdl:operation>
4992 <wsdl:operation name="GetEnrollResults">
4993   <wsdl:input name="GetEnrollResults" message="tns:GetEnrollResultsRequestMessage"/>
4994   <wsdl:output name="GetEnrollResultsResponse" message="tns:GetEnrollResultsResponseMessage"/>
4995   <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
4996 </wsdl:operation>
4997 <wsdl:operation name="GetIdentifyResults">
4998   <wsdl:input name="GetIdentifyResults" message="tns:GetIdentifyResultsRequestMessage"/>
4999   <wsdl:output name="GetIdentifyResultsResponse" message="tns:GetIdentifyResultsResponseMessage"/>

```

```

5000     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
5001 </wsdl:operation>
5002 <wsdl:operation name="GetVerifyResults">
5003     <wsdl:input name="GetVerifyResults" message="tns:GetVerifyResultsRequestMessage"/>
5004     <wsdl:output name="GetVerifyResultsResponse" message="tns:GetVerifyResultsResponseMessage"/>
5005     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
5006 </wsdl:operation>
5007 <wsdl:operation name="Identify">
5008     <wsdl:input name="Identify" message="tns:IdentifyRequestMessage"/>
5009     <wsdl:output name="IdentifyResponse" message="tns:IdentifyResponseMessage"/>
5010     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
5011 </wsdl:operation>
5012 <wsdl:operation name="RetrieveData">
5013     <wsdl:input name="RetrieveData" message="tns:RetrieveDataRequestMessage"/>
5014     <wsdl:output name="RetrieveDataResponse" message="tns:RetrieveDataResponseMessage"/>
5015     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
5016 </wsdl:operation>
5017 <wsdl:operation name="Verify">
5018     <wsdl:input name="Verify" message="tns:VerifyRequestMessage"/>
5019     <wsdl:output name="VerifyResponse" message="tns:VerifyResponseMessage"/>
5020     <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
5021 </wsdl:operation>
5022 </wsdl:portType>
5023 <wsdl:binding name="BIAS_v1HttpBinding" type="tns:BIAS_v1">
5024     <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
5025     <wsdl:operation name="AddSubjectToGallery">
5026         <soap:operation soapAction="AddSubjectToGallery"/>
5027         <wsdl:input name="AddSubjectToGallery">
5028             <soap:body use="literal"/>
5029         </wsdl:input>
5030         <wsdl:output name="AddSubjectToGalleryResponse">
5031             <soap:body use="literal"/>
5032         </wsdl:output>
5033     <wsdl:fault name="BIASFault">

```

```

5034     <soap:fault name="BIASFault" use="literal"/>
5035 </wsdl:fault>
5036 </wsdl:operation>
5037 <wsdl:operation name="CheckQuality">
5038   <soap:operation soapAction="CheckQuality"/>
5039   <wsdl:input name="CheckQuality">
5040     <soap:body use="literal"/>
5041   </wsdl:input>
5042   <wsdl:output name="CheckQualityResponse">
5043     <soap:body use="literal"/>
5044   </wsdl:output>
5045   <wsdl:fault name="BIASFault">
5046     <soap:fault name="BIASFault" use="literal"/>
5047   </wsdl:fault>
5048 </wsdl:operation>
5049 <wsdl:operation name="ClassifyBiometricData">
5050   <soap:operation soapAction="ClassifyBiometricData"/>
5051   <wsdl:input name="ClassifyBiometricData">
5052     <soap:body use="literal"/>
5053   </wsdl:input>
5054   <wsdl:output name="ClassifyBiometricDataResponse">
5055     <soap:body use="literal"/>
5056   </wsdl:output>
5057   <wsdl:fault name="BIASFault">
5058     <soap:fault name="BIASFault" use="literal"/>
5059   </wsdl:fault>
5060 </wsdl:operation>
5061 <wsdl:operation name="CreateSubject">
5062   <soap:operation soapAction="CreateSubject"/>
5063   <wsdl:input name="CreateSubject">
5064     <soap:body use="literal"/>
5065   </wsdl:input>
5066   <wsdl:output name="CreateSubjectResponse">
5067     <soap:body use="literal"/>

```

```

5068     </wsdl:output>
5069     <wsdl:fault name="BIASFault">
5070         <soap:fault name="BIASFault" use="literal"/>
5071     </wsdl:fault>
5072 </wsdl:operation>
5073 <wsdl:operation name="DeleteBiographicData">
5074     <soap:operation soapAction="DeleteBiographicData"/>
5075     <wsdl:input name="DeleteBiographicData">
5076         <soap:body use="literal"/>
5077     </wsdl:input>
5078     <wsdl:output name="DeleteBiographicDataResponse">
5079         <soap:body use="literal"/>
5080     </wsdl:output>
5081     <wsdl:fault name="BIASFault">
5082         <soap:fault name="BIASFault" use="literal"/>
5083     </wsdl:fault>
5084 </wsdl:operation>
5085 <wsdl:operation name="DeleteBiometricData">
5086     <soap:operation soapAction="DeleteBiometricData"/>
5087     <wsdl:input name="DeleteBiometricData">
5088         <soap:body use="literal"/>
5089     </wsdl:input>
5090     <wsdl:output name="DeleteBiometricDataResponse">
5091         <soap:body use="literal"/>
5092     </wsdl:output>
5093     <wsdl:fault name="BIASFault">
5094         <soap:fault name="BIASFault" use="literal"/>
5095     </wsdl:fault>
5096 </wsdl:operation>
5097 <wsdl:operation name="DeleteSubject">
5098     <soap:operation soapAction="DeleteSubject"/>
5099     <wsdl:input name="DeleteSubject">
5100         <soap:body use="literal"/>
5101     </wsdl:input>

```

```

5102 <wsdl:output name="DeleteSubjectResponse">
5103   <soap:body use="literal"/>
5104 </wsdl:output>
5105 <wsdl:fault name="BIASFault">
5106   <soap:fault name="BIASFault" use="literal"/>
5107 </wsdl:fault>
5108 </wsdl:operation>
5109 <wsdl:operation name="DeleteSubjectFromGallery">
5110   <soap:operation soapAction="DeleteSubjectFromGallery"/>
5111   <wsdl:input name="DeleteSubjectFromGallery">
5112     <soap:body use="literal"/>
5113   </wsdl:input>
5114   <wsdl:output name="DeleteSubjectFromGalleryResponse">
5115     <soap:body use="literal"/>
5116   </wsdl:output>
5117   <wsdl:fault name="BIASFault">
5118     <soap:fault name="BIASFault" use="literal"/>
5119   </wsdl:fault>
5120   </wsdl:operation>
5121 <wsdl:operation name="GetIdentifySubjectResults">
5122   <soap:operation soapAction="GetIdentifySubjectResults"/>
5123   <wsdl:input name="GetIdentifySubjectResults">
5124     <soap:body use="literal"/>
5125   </wsdl:input>
5126   <wsdl:output name="GetIdentifySubjectResultsResponse">
5127     <soap:body use="literal"/>
5128   </wsdl:output>
5129   <wsdl:fault name="BIASFault">
5130     <soap:fault name="BIASFault" use="literal"/>
5131   </wsdl:fault>
5132   </wsdl:operation>
5133 <wsdl:operation name="IdentifySubject">
5134   <soap:operation soapAction="IdentifySubject"/>
5135   <wsdl:input name="IdentifySubject">

```



```

5136     <soap:body use="literal"/>
5137 </wsdl:input>
5138 <wsdl:output name="IdentifySubjectResponse">
5139     <soap:body use="literal"/>
5140 </wsdl:output>
5141 <wsdl:fault name="BIASFault">
5142     <soap:fault name="BIASFault" use="literal"/>
5143 </wsdl:fault>
5144 </wsdl:operation>
5145 <wsdl:operation name="ListBiographicData">
5146     <soap:operation soapAction="ListBiographicData"/>
5147     <wsdl:input name="ListBiographicData">
5148         <soap:body use="literal"/>
5149     </wsdl:input>
5150     <wsdl:output name="ListBiographicDataResponse">
5151         <soap:body use="literal"/>
5152     </wsdl:output>
5153     <wsdl:fault name="BIASFault">
5154         <soap:fault name="BIASFault" use="literal"/>
5155     </wsdl:fault>
5156 </wsdl:operation>
5157 <wsdl:operation name="ListBiometricData">
5158     <soap:operation soapAction="ListBiometricData"/>
5159     <wsdl:input name="ListBiometricData">
5160         <soap:body use="literal"/>
5161     </wsdl:input>
5162     <wsdl:output name="ListBiometricDataResponse">
5163         <soap:body use="literal"/>
5164     </wsdl:output>
5165     <wsdl:fault name="BIASFault">
5166         <soap:fault name="BIASFault" use="literal"/>
5167     </wsdl:fault>
5168 </wsdl:operation>
5169 <wsdl:operation name="PerformFusion">

```

```

5170 <soap:operation soapAction="PerformFusion"/>
5171 <wsdl:input name="PerformFusion">
5172   <soap:body use="literal"/>
5173 </wsdl:input>
5174 <wsdl:output name="PerformFusionResponse">
5175   <soap:body use="literal"/>
5176 </wsdl:output>
5177 <wsdl:fault name="BIASFault">
5178   <soap:fault name="BIASFault" use="literal"/>
5179 </wsdl:fault>
5180 </wsdl:operation>
5181 <wsdl:operation name="QueryCapabilities">
5182   <soap:operation soapAction="QueryCapabilities"/>
5183   <wsdl:input name="QueryCapabilities">
5184     <soap:body use="literal"/>
5185   </wsdl:input>
5186   <wsdl:output name="QueryCapabilitiesResponse">
5187     <soap:body use="literal"/>
5188   </wsdl:output>
5189   <wsdl:fault name="BIASFault">
5190     <soap:fault name="BIASFault" use="literal"/>
5191   </wsdl:fault>
5192 </wsdl:operation>
5193 <wsdl:operation name="RetrieveBiographicData">
5194   <soap:operation soapAction="RetrieveBiographicData"/>
5195   <wsdl:input name="RetrieveBiographicData">
5196     <soap:body use="literal"/>
5197   </wsdl:input>
5198   <wsdl:output name="RetrieveBiographicDataResponse">
5199     <soap:body use="literal"/>
5200   </wsdl:output>
5201   <wsdl:fault name="BIASFault">
5202     <soap:fault name="BIASFault" use="literal"/>
5203   </wsdl:fault>

```

```

5204 </wsdl:operation>
5205 <wsdl:operation name="RetrieveBiometricData">
5206   <soap:operation soapAction="RetrieveBiometricData"/>
5207   <wsdl:input name="RetrieveBiometricData">
5208     <soap:body use="literal"/>
5209   </wsdl:input>
5210   <wsdl:output name="RetrieveBiometricDataResponse">
5211     <soap:body use="literal"/>
5212   </wsdl:output>
5213   <wsdl:fault name="BIASFault">
5214     <soap:fault name="BIASFault" use="literal"/>
5215   </wsdl:fault>
5216 </wsdl:operation>
5217 <wsdl:operation name="SetBiographicData">
5218   <soap:operation soapAction="SetBiographicData"/>
5219   <wsdl:input name="SetBiographicData">
5220     <soap:body use="literal"/>
5221   </wsdl:input>
5222   <wsdl:output name="SetBiographicDataResponse">
5223     <soap:body use="literal"/>
5224   </wsdl:output>
5225   <wsdl:fault name="BIASFault">
5226     <soap:fault name="BIASFault" use="literal"/>
5227   </wsdl:fault>
5228 </wsdl:operation>
5229 <wsdl:operation name="SetBiometricData">
5230   <soap:operation soapAction="SetBiometricData"/>
5231   <wsdl:input name="SetBiometricData">
5232     <soap:body use="literal"/>
5233   </wsdl:input>
5234   <wsdl:output name="SetBiometricDataResponse">
5235     <soap:body use="literal"/>
5236   </wsdl:output>
5237   <wsdl:fault name="BIASFault">

```

```

5238     <soap:fault name="BIASFault" use="literal"/>
5239   </wsdl:fault>
5240 </wsdl:operation>
5241 <wsdl:operation name="TransformBiometricData">
5242   <soap:operation soapAction="TransformBiometricData"/>
5243   <wsdl:input name="TransformBiometricData">
5244     <soap:body use="literal"/>
5245   </wsdl:input>
5246   <wsdl:output name="TransformBiometricDataResponse">
5247     <soap:body use="literal"/>
5248   </wsdl:output>
5249   <wsdl:fault name="BIASFault">
5250     <soap:fault name="BIASFault" use="literal"/>
5251   </wsdl:fault>
5252 </wsdl:operation>
5253 <wsdl:operation name="UpdateBiographicData">
5254   <soap:operation soapAction="UpdateBiographicData"/>
5255   <wsdl:input name="UpdateBiographicData">
5256     <soap:body use="literal"/>
5257   </wsdl:input>
5258   <wsdl:output name="UpdateBiographicDataResponse">
5259     <soap:body use="literal"/>
5260   </wsdl:output>
5261   <wsdl:fault name="BIASFault">
5262     <soap:fault name="BIASFault" use="literal"/>
5263   </wsdl:fault>
5264 </wsdl:operation>
5265 <wsdl:operation name="UpdateBiometricData">
5266   <soap:operation soapAction="UpdateBiometricData"/>
5267   <wsdl:input name="UpdateBiometricData">
5268     <soap:body use="literal"/>
5269   </wsdl:input>
5270   <wsdl:output name="UpdateBiometricDataResponse">
5271     <soap:body use="literal"/>

```

```

5272     </wsdl:output>
5273     <wsdl:fault name="BIASFault">
5274         <soap:fault name="BIASFault" use="literal"/>
5275     </wsdl:fault>
5276 </wsdl:operation>
5277 <wsdl:operation name="VerifySubject">
5278     <soap:operation soapAction="VerifySubject"/>
5279     <wsdl:input name="VerifySubject">
5280         <soap:body use="literal"/>
5281     </wsdl:input>
5282     <wsdl:output name="VerifySubjectResponse">
5283         <soap:body use="literal"/>
5284     </wsdl:output>
5285     <wsdl:fault name="BIASFault">
5286         <soap:fault name="BIASFault" use="literal"/>
5287     </wsdl:fault>
5288 </wsdl:operation>
5289 <wsdl:operation name="Enroll">
5290     <soap:operation soapAction="Enroll"/>
5291     <wsdl:input name="Enroll">
5292         <soap:body use="literal"/>
5293     </wsdl:input>
5294     <wsdl:output name="EnrollResponse">
5295         <soap:body use="literal"/>
5296     </wsdl:output>
5297     <wsdl:fault name="BIASFault">
5298         <soap:fault name="BIASFault" use="literal"/>
5299     </wsdl:fault>
5300 </wsdl:operation>
5301 <wsdl:operation name="GetEnrollResults">
5302     <soap:operation soapAction="GetEnrollResults"/>
5303     <wsdl:input name="GetEnrollResults">
5304         <soap:body use="literal"/>
5305     </wsdl:input>

```

```

5306 <wsdl:output name="GetEnrollResultsResponse">
5307   <soap:body use="literal"/>
5308 </wsdl:output>
5309 <wsdl:fault name="BIASFault">
5310   <soap:fault name="BIASFault" use="literal"/>
5311 </wsdl:fault>
5312 </wsdl:operation>
5313 <wsdl:operation name="GetIdentifyResults">
5314   <soap:operation soapAction="GetIdentifyResults"/>
5315   <wsdl:input name="GetIdentifyResults">
5316     <soap:body use="literal"/>
5317   </wsdl:input>
5318   <wsdl:output name="GetIdentifyResultsResponse">
5319     <soap:body use="literal"/>
5320   </wsdl:output>
5321   <wsdl:fault name="BIASFault">
5322     <soap:fault name="BIASFault" use="literal"/>
5323   </wsdl:fault>
5324   </wsdl:operation>
5325   <wsdl:operation name="GetVerifyResults">
5326     <soap:operation soapAction="GetVerifyResults"/>
5327     <wsdl:input name="GetVerifyResults">
5328       <soap:body use="literal"/>
5329     </wsdl:input>
5330     <wsdl:output name="GetVerifyResultsResponse">
5331       <soap:body use="literal"/>
5332     </wsdl:output>
5333     <wsdl:fault name="BIASFault">
5334       <soap:fault name="BIASFault" use="literal"/>
5335     </wsdl:fault>
5336     </wsdl:operation>
5337   <wsdl:operation name="Identify">
5338     <soap:operation soapAction="Identify"/>
5339     <wsdl:input name="Identify">

```

```

5340     <soap:body use="literal"/>
5341 </wsdl:input>
5342 <wsdl:output name="IdentifyResponse">
5343     <soap:body use="literal"/>
5344 </wsdl:output>
5345 <wsdl:fault name="BIASFault">
5346     <soap:fault name="BIASFault" use="literal"/>
5347 </wsdl:fault>
5348 </wsdl:operation>
5349 <wsdl:operation name="RetrieveData">
5350     <soap:operation soapAction="RetrieveData"/>
5351 <wsdl:input name="RetrieveData">
5352     <soap:body use="literal"/>
5353 </wsdl:input>
5354 <wsdl:output name="RetrieveDataResponse">
5355     <soap:body use="literal"/>
5356 </wsdl:output>
5357 <wsdl:fault name="BIASFault">
5358     <soap:fault name="BIASFault" use="literal"/>
5359 </wsdl:fault>
5360 </wsdl:operation>
5361 <wsdl:operation name="Verify">
5362     <soap:operation soapAction="Verify"/>
5363 <wsdl:input name="Verify">
5364     <soap:body use="literal"/>
5365 </wsdl:input>
5366 <wsdl:output name="VerifyResponse">
5367     <soap:body use="literal"/>
5368 </wsdl:output>
5369 <wsdl:fault name="BIASFault">
5370     <soap:fault name="BIASFault" use="literal"/>
5371 </wsdl:fault>
5372 </wsdl:operation>
5373 </wsdl:binding>

```

5374 <wsdl:service name="BIAS_v1">
5375 <wsdl:port name="BIAS_v1HttpBindingPort" binding="tns:BIAS_v1HttpBinding">
5376 <soap:address location="http://your_URI/IRI_to_your_BIAS_implementation"/>
5377 </wsdl:port>
5378 </wsdl:service>
5379 </wsdl:definitions>
5380

5381
5382
5383
5384
5385
5386
5387
5388
5389
5390
5391
5392
5393
5394
5395
5396
5397
5398

Appendix B. BIAS Patron format specification

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

B.1 Patron

Organization for the Advancement of Structured Information Standards (OASIS)

B.2 Patron identifier

82 (0052 Hex).

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

B.3 Patron format name

OASIS BIAS CBEFF XML Patron Format

B.4 Patron format identifier

01 (0001 Hex).

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

B.5 ASN.1 object identifier for this patron format

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

B.6 Domain of use

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

B.7 Version identifier

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

B.8 CBEFF version

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

B.9 General

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

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

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

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

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

B.10 Specification

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

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

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

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

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

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

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

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

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

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

B.11 Element <BIR>

B.11.1 Syntax

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

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

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

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

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

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

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

B.11.2 Semantics

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

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

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

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

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

5514 **B.11.2.6** The <BDB> or <bdbX> element (if present) carries the biometric data block (BDB) of the
5515 BIR.

5516 NOTE – A <BDB> or <bdbX> element and a <BIR> element cannot coexist as children of
5517 the same <BIR> element (see B.11.1.2).

5518 **B.11.2.7** The <SB> element (if present) carries the security block (SB) of the BIR.

5519 NOTE – A <SB> element can coexist with either a <BIR> element or a <BDB> or
5520 <bdbX> element that is a child of the same <BIR> element.

5521 **B.11.2.8** The <BIRInfo> element carries information about both the BIR and (possibly) about its
5522 descendant BIRs (if the <BIR> element has one or more child <BIR> elements), as specified in B.14.2.1.

5523 **B.11.2.9** The <BDBInfo> element (if present) carries information about either the BDB of the BIR
5524 (if the <BIR> element has a child <BDB> or <bdbX> element) or about the BDBs of the descendant BIRs
5525 that have a child <BDB> or <bdbX> element (if the <BIR> element has one or more child <BIR>
5526 elements), as specified in B.15.2.1.

5527 **B.11.2.10** The <SBInfo> element (if present) carries information about either the SB of the BIR (if
5528 the <BIR> element has a child <SB> element) or about the SBs of the descendant BIRs that have a child
5529 <SB> element (if the <BIR> element has one or more child <BIR> elements but no child <SB> element),
5530 as specified in B.16.2.1.

5531

5532 **B.12 Element <Version>**

5533 **B.12.1 Syntax**

5534 This element shall have contents consisting of the following (in order):

5535 a) a required <Major> element – the value of this element shall be a valid representation of a non-
5536 negative integer.

5537 b) a required <Minor> element – the value of this element shall be a valid representation of a non-
5538 negative integer.

5539 **B.12.2 Semantics**

5540 **B.12.2.1** This element represents the data element CBEFF_patron_header_version, and carries
5541 the (major and minor) version number of the patron format. The number assigned to this version of the
5542 patron format is major 1, minor 0.

5543 **B.12.2.2** The <Major> element represents the major version number (1 in this version).

5544 **B.12.2.3** The <Minor> element represents the minor version number (0 in this version).

5545 **B.12.2.4** If this element is not present, the values Major="1" Minor="0" are implied.

5546 **B.12.2.5** A child <BIR> element shall have the same (major and minor) version number as its
5547 parent <BIR> element.

5548 NOTE – This implies that the <Version> element, if present in a child <BIR> element,
5549 has to carry the same values as the <Version> element in the parent <BIR> element.
5550 This is equivalent to omitting the <Version> element. Therefore, this element is
5551 normally omitted in child <BIR> elements.

5552

5553 **B.13 Element <CBEFFVersion>**

5554 **B.13.1 Syntax**

5555 This element shall have content consisting of the following (in order):

- 5556 a) a required **<Major>** element – the value of this element shall be a valid representation of a non-
5557 negative integer (see B.17);
- 5558 b) a required **<Minor>** element – the value of this element shall be a valid representation of a non-
5559 negative integer.

5560 **B.13.2 Semantics**

5561 **B.13.2.1** This element represents the data element CBEFF_version, and carries the version
5562 number of the CBEFF standard supported by this patron format. The number assigned to the version of
5563 CBEFF supported by this patron format is Major=2, Minor=0.

5564 **B.13.2.2** The **<Major>** element represents the major version number (2 in this version).

5565 **B.13.2.3** The **<Minor>** element represents the minor version number (0 in this version).

5566 **B.13.2.4** If this element is not present, the values **Major="2"** **Minor="0"** are implied.

5567 **B.13.2.5** A child **<BIR>** element shall have the same CBEFF version number (major and minor) as
5568 its parent **<BIR>** element.

5569 NOTE – Thus, the **<CBEFFVersion>** element is normally omitted from all child **<BIR>**
5570 elements, as it would be redundant.

5571

5572 **B.14 Element <BIRInfo>**

5573 **B.14.1 Syntax**

5574 **B.14.1.1** This element shall have a content consisting of the following (in order):

- 5575 a) an optional **<Creator>** element – the content of this element shall be a string of ISO/IEC 10646
5576 characters;
- 5577 b) an optional **<Index>** element – the content of this element shall be a valid representation of a
5578 universally unique identifier (see B.20), and shall not inherit its value from any other level BIR;
- 5579 c) an optional **<Payload>** element – the content of this element shall be a valid representation of an
5580 octet string, and shall not inherit its value from any other level BIR.
- 5581 d) a required **<Integrity>** element – the value of this element shall be one of the character strings in
5582 the third cell of the corresponding row of Table B.1;
- 5583 e) an optional **<CreationDate>** element – the value of this element shall be a valid representation of a
5584 date and time of the day (see B.19);
- 5585 f) an optional **<NotValidBefore>** element – the value of this element shall be a valid representation
5586 of a date and time of the day;
- 5587 g) an optional **<NotValidAfter>** element – the value of this element shall be a valid representation of
5588 a date and time of the day.

B.14.2 Semantics

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

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

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

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

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

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

NOTE - This element represents all CBEFF data elements whose name begins with "CBEFF_BIR_".

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. The character string shall be encoded as the string itself.	
CBEFF_BIR_index	<Index>	All well-formed UUIDs are supported. The UUIDs shall be encoded as specified in B.20. Shall not inherit its value from any other BIR level.	
CBEFF_BIR_payload	<Payload>	All octet strings are supported. The octet strings shall be encoded as specified in B.18. Shall not inherit its value from any other BIR level.	
CBEFF_BIR_integrity_options	<Integrity>	The following abstract values are supported. The abstract values shall be encoded as shown below. NO INTEGRITY:	

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

5613

5614 B.15 Element <BDBInfo>

5615 B.15.1 Syntax

5616 **B.15.1.1** This element shall have a content consisting of the following (in order):

- 5617 a) an optional <ChallengeResponse> element – the content of this element shall be a valid
5618 representation of an octet string (see B.18);
- 5619 b) an optional <Index> element – the content of this element shall be a valid representation of a
5620 universally unique identifier (see B.20).
- 5621 c) an optional <FormatOwner> element – the value of this element shall be a valid representation
5622 of an integer in the range 1 to 65535 (see B.17);
- 5623 d) an optional <FormatType> element – the value of this element shall be a valid representation of
5624 an integer in the range 1 to 65535;
- 5625 e) an optional <Encryption> element – the value of this element shall be one of the character
5626 strings in the third cell of the corresponding row of Table B.2;
- 5627 f) an optional <CreationDate> element – the value of this element shall be a valid
5628 representation of a date and time of the day (see B.19);
- 5629 g) an optional <NotValidBefore> element – the value of this element shall be a valid
5630 representation of a date and time of the day;
- 5631 h) an optional <NotValidAfter> element – the value of this element shall be a valid
5632 representation of a date and time of the day;

- 5633 i) an optional **<Type>** element – the value of this element shall be one of the character strings in
5634 the third cell of the corresponding row of Table B.2;
- 5635 j) an optional **<Subtype>** element – the value of this element shall be one of the character strings
5636 in the third cell of the corresponding row of Table B.2;
- 5637 k) an optional **<Level1>** element – the value of this element shall be one of the character strings in
5638 the third cell of the corresponding row of Table B.2;
- 5639 l) an optional **<ProductOwner>** element – the value of this element shall be a valid
5640 representation of an integer in the range 1..65535 (see B.17);
- 5641 m) an optional **<ProductType>** element – the value of this element shall be a valid representation
5642 of an integer in the range 1..65535;
- 5643 n) an optional **<CaptureDeviceOwner>** element – the value of this element shall be a valid
5644 representation of an integer in the range 1..65535 (see B.17);
- 5645 o) an optional **<CaptureDeviceType>** element – the value of this element shall be a valid
5646 representation of an integer in the range 1..65535;
- 5647 p) an optional **<FeatureExtractionAlgorithmOwner>** element – the value of this element
5648 shall be a valid representation of an integer in the range 1..65535 (see B.17);
- 5649 q) an optional **<FeatureExtractionAlgorithmType>** element – the value of this element shall
5650 be a valid representation of an integer in the range 1..65535;
- 5651 r) an optional **<ComparisonAlgorithmOwner>** element – the value of this element shall be a
5652 valid representation of an integer in the range 1..65535 (see B.17);
- 5653 s) an optional **<ComparisonAlgorithmType>** element – the value of this element shall be a valid
5654 representation of an integer in the range 1..65535;
- 5655 t) an optional **<QualityAlgorithmOwner>** element – the value of this element shall be a valid
5656 representation of an integer in the range 1..65535 (see B.17);
- 5657 u) an optional **<QualityAlgorithmType>** element – the value of this element shall be a valid
5658 representation of an integer in the range 1..65535;
- 5659 v) an optional **<CompressionAlgorithmOwner>** element – the value of this element shall be a
5660 valid representation of an integer in the range 1..65535 (see B.17);
- 5661 w) an optional **<CompressionAlgorithmType>** element – the value of this element shall be a
5662 valid representation of an integer in the range 1..65535;
- 5663 x) an optional **<Purpose>** element – the value of this element shall be one of the character strings
5664 in the third cell of the corresponding row of Table B.2;
- 5665 y) an optional **<Quality>** element – the value of this element shall be a valid representation of an
5666 integer in the range –2..100 (see B.17), as specified in the third cell of the corresponding row of
5667 Table B.2.

5668 **B.15.1.3** If the parent **<BIR>** element has a child **<BDB>** element, then the **<Encryption>**
5669 element shall be present in this **<BDBInfo>** element unless it is present in the child **<BDBInfo>** element
5670 of an ancestor **<BIR>** element (see also B.11.1.4).

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

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

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

B.15.2 Semantics

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

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

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

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

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

NOTE – This element represents all CBEFF data elements whose name begins with "CBEFF_BDB_".

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. 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. The integers shall be encoded as specified in B.17.	
CBEFF_BDB_encryption_options	<Encryption>	The following abstract values are supported. The abstract values shall be encoded as shown below. NO ENCRYPTION:	

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

CBEFF data element name	XML element	Supported abstract values and encodings	Reference
		VEIN: "Vein" HAND GEOMETRY: "HandGeometry" IRIS: "Iris" RETINA: "Retina" VOICE: "Voice" GAIT: "Gait" KEYSTROKE: "Keystroke" LIP MOVEMENT: "LipMovement" SIGNATURE OR SIGN: "SignatureSign"	
CBEFF_BDB_biometric_subtype	<Subtype>	The following abstract values are supported. 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. LEFT: "Left" RIGHT: "Right" THUMB: "Thumb" INDEX FINGER: "IndexFinger" MIDDLE FINGER: "MiddleFinger" RING FINGER: "RingFinger" LITTLE FINGER: "LittleFinger"	
CBEFF_BDB_processed_level	<Level>	The following abstract values are supported. The abstract values shall be encoded as shown below. RAW: "Raw" INTERMEDIATE: "Intermediate" 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. 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. 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. 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. 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. 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. 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. 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. 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. 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. 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. 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. 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: "Verify"</p> <p>IDENTIFY: "Identify"</p> <p>ENROLL: "Enroll"</p> <p>ENROLL FOR VERIFICATION ONLY: "EnrollVerify"</p> <p>ENROLL FOR IDENTIFICATION ONLY: "EnrollIdentify"</p> <p>AUDIT: "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: "-2"</p> <p>QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET: "-1"</p>	

5703

5704 B.16 Element <SBInfo>

5705 B.16.1 Syntax

5706 **B.16.1.1** This element shall have content consisting of the following (in order):

5707 a) an optional **<FormatOwner>** element – the value of this element shall be a valid representation of an
5708 integer in the range 1 to 65535 (see B.17);

5709 b) an optional **<FormatType>** element – the value of this element shall be a valid representation of an
5710 integer in the range 1 to 65535

5711 **B.16.1.2** If the parent **<BIR>** element has a child **<SB>** element, then the **<FormatOwner>**
5712 element shall be present in this **<SBInfo>** element unless it is present in the child **<SBInfo>** element of
5713 an ancestor **<BIR>** element (see also B.11.1.5).

5714 **B.16.1.3** If the parent **<BIR>** element has a child **<SB>** element, then the **<FormatType>** element
5715 shall be present in this **<SBInfo>** element unless it is present in the child **<SBInfo>** element of an
5716 ancestor **<BIR>** element (see also B.11.1.5).

5717 NOTE 1 – The ancestor <BIR> elements mentioned in the last two subclauses above
5718 need not be the same.

5719 NOTE 2 – When the parent <BIR> element has a child <SB> element and one omits both
5720 children of the <SBInfo> element, the <SBInfo> element will have no attributes and an
5721 empty content. Omission of the <SBInfo> element is not allowed in this case (see
5722 B.11.1.5).

5723

5724 **B.16.2 Semantics**

5725 **B.16.2.1** If the BIR has an SB (the <BIR> element has a child <SB> element), then the <SBInfo>
5726 element carries information about that SB. In addition, if the BIR has one or more child BIRs (the <BIR>
5727 element has one or more child <BIR> elements), the information carried by the child element of the
5728 <SBInfo> element is inherited by those child BIRs except where overridden by a corresponding child
5729 element of the <SBInfo> element of a child BIR. The information inherited by a BIR with an SB applies
5730 to that SB, and (if the BIR has itself child BIRs) is further inherited by its child BIRs in the same way (and
5731 so on recursively).

5732 **B.16.2.2** Table B.3 specifies the correspondence between the attributes and child elements of this
5733 element and CBEFF data elements, and specifies the supported abstract values and their encodings (see
5734 also B.10.7).

5735 NOTE – This element represents all CBEFF data elements whose name begins with
5736 "CBEFF_SB_".

5737

5738 **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. 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. The integers shall be encoded as specified in B.17.	

5739 **B.17 Representation of Integers**

5740 **B.17.1** A non-negative integer shall be represented as a string of one or more ISO/IEC 10646 characters
5741 in the range DIGIT ZERO to DIGIT NINE ("0" to "9") in decimal notation.

5742 **B.17.2** A negative integer shall be represented as the corresponding positive integer, preceded by a
5743 HYPHEN-MINUS character ("-").

5744 **B.17.3** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the
5745 encoding.

5746

5747 **B.18 Representation of Octet Strings**

5748 **B.18.1** An octet string shall be represented as a string of the following ISO/IEC 10646 characters:
5749 a) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER Z;

5750 b) LATIN SMALL LETTER A to LATIN SMALL LETTER Z;

5751 c) DIGIT ZERO to DIGIT NINE;

5752 d) PLUS SIGN;

5753 e) SOLIDUS;

5754 f) EQUALS SIGN.

5755 forming the Base64 encoding of the octet string (see IETF RFC 2045), with all whitespace removed.

5756 **B.18.2** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the
5757 encoding.

5758

5759 **B.19 Representation of Date and Time of the Day**

5760 **B.19.1** A date and time of the day shall be represented as a string of ISO/IEC 10646 characters in the
5761 following format, which conforms to ISO 8601.

5762 **B.19.2** The encoding shall be the concatenation of all the following components (in order):

5763 a) the "year" component, consisting of the year encoded in four digits ("2000" to "2999") ;

5764 b) the hyphen character "-";

5765 c) the "month" component, consisting of the month encoded in two digits ("01" to "12");

5766 d) the hyphen character "-";

5767 e) the "day" component, consisting of the day encoded in two digits ("01" to "31");

5768 f) the letter "T";

5769 g) the "hour" component, consisting of the hour encoded in two digits ("00" to "23");

5770 h) the colon character ":";

5771 i) the "minute" component, consisting of the minute encoded in two digits ("00" to "59");

5772 j) the colon character ":";

5773 k) the "second" component, consisting of the second encoded in two digits ("00" to "59");

5774 l) the letter "Z".

5775 **B.19.3** The "year", "month", "day", "hour", "minute", and "second" components shall be present.

5776 **B.19.4** The letter "T" shall be present.

5777 **B.19.5** The letter "Z" shall be present whether or not the "hour" component is present.

5778 NOTE This letter indicates that the date and time of the day are UTC.

5779 **B.19.6** Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the
5780 encoding.

5781

B.20 Representation of Universally Unique Identifiers

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

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

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

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

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

B.21 Patron format conformance statement

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

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

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

CBEFF data element name	Mandatory/ optional	Patron format field name	Abstract values specified?	Encodings specified?
	is present			
CBEFF_BDB_format_type	Mandatory (specified or inherited) if a BDB is present	<FormatType> child of <BDBInfo>	Yes	Yes
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	<FeatureExtractionAlgorithm Owner> child of <BDBInfo>	Yes	Yes
CBEFF_BDB_feature_extraction_algorithm_type	Optional	<FeatureExtractionAlgorithm Type> 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 data element name	Mandatory/ optional	Patron format field name	Abstract values specified?	Encodings specified?
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_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

5803

5804 B.22 XML schema of the BIAS patron format

5805 <?xml version='1.0' encoding='utf-8'?>

```

5806 <xs:schema
5807   xmlns:xs="http://www.w3.org/2001/XMLSchema"
5808   xmlns="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
5809   targetNamespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"
5810   elementFormDefault="qualified">
5811
5812   <xs:element name="BIR" type="BIRType"/>
5813
5814   <xs:complexType name="BIRType">
5815     <xs:sequence>
5816       <xs:element name="Version" type="VersionType" minOccurs="0"/>
5817       <xs:element name="CBEFFVersion" type="VersionType" minOccurs="0"/>
5818       <xs:any namespace="##other" processContents="skip" minOccurs="0" maxOccurs="unbounded"/>
5819       <xs:element name="BIRInfo" type="BIRInfoType"/>
5820       <xs:element name="BDBInfo" type="BDBInfoType" minOccurs="0"/>
5821       <xs:element name="SBInfo" type="SBInfoType" minOccurs="0"/>
5822       <xs:element name="BIR" type="BIRType" minOccurs="0" maxOccurs="unbounded"/>
5823       <xs:element name="BDB" type="xs:base64Binary" minOccurs="0"/>
5824       <xs:element name="SB" type="xs:base64Binary" minOccurs="0"/>
5825     </xs:sequence>
5826   </xs:complexType>
5827
5828   <xs:complexType name="VersionType">
5829     <xs:sequence>
5830       <xs:element name="Major" type="xs:nonNegativeInteger" />
5831       <xs:element name="Minor" type="xs:nonNegativeInteger" />
5832     </xs:sequence>
5833   </xs:complexType>
5834
5835   <xs:complexType name="BIRInfoType">
5836     <xs:sequence>
5837       <xs:element name="Creator" type="xs:string" minOccurs="0"/>
5838       <xs:element name="Index" type="UUIDType" minOccurs="0"/>
5839       <xs:element name="Payload" type="xs:base64Binary" minOccurs="0"/>
5840       <xs:element name="Integrity" type="xs:boolean" />
5841       <xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />
5842       <xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />
5843       <xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />
5844     </xs:sequence>
5845   </xs:complexType>
5846
5847   <xs:complexType name="BDBInfoType">
5848     <xs:sequence>
5849       <xs:element name="ChallengeResponse" type="xs:base64Binary" minOccurs="0"/>
5850       <xs:element name="Index" type="UUIDType" minOccurs="0"/>
5851       <xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />
5852       <xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />
5853       <xs:element name="Encryption" type="xs:boolean" minOccurs="0" />
5854       <xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />
5855       <xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />
5856       <xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />
5857       <xs:element name="Type" type="MultipleTypesType" minOccurs="0" />
5858       <xs:element name="Subtype" type="SubtypeType" minOccurs="0" />
5859       <xs:element name="Level" type="ProcessedLevelType" minOccurs="0" />
5860       <xs:element name="ProductOwner" type="xs:positiveInteger" minOccurs="0" />
5861       <xs:element name="ProductType" type="xs:positiveInteger" minOccurs="0" />
5862       <xs:element name="CaptureDeviceOwner" type="xs:positiveInteger" minOccurs="0" />
5863       <xs:element name="CaptureDeviceType" type="xs:positiveInteger" minOccurs="0" />
5864       <xs:element name="FeatureAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5865       <xs:element name="FeatureAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5866       <xs:element name="ComparisonAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5867       <xs:element name="ComparisonAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5868       <xs:element name="QualityAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5869       <xs:element name="QualityAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5870       <xs:element name="CompressionAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />
5871       <xs:element name="CompressionAlgorithmType" type="xs:positiveInteger" minOccurs="0" />
5872       <xs:element name="Purpose" type="PurposeType" minOccurs="0" />
5873       <xs:element name="Quality" type="QualityType" minOccurs="0" />
5874     </xs:sequence>
5875   </xs:complexType>
5876

```

```

5877 <xs:complexType name="SBInfoType">
5878 <xs:sequence>
5879 <xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />
5880 <xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />
5881 </xs:sequence>
5882 </xs:complexType>
5883
5884 <xs:simpleType name="QualityType">
5885 <xs:restriction base="xs:integer">
5886 <xs:minInclusive value="-2"/>
5887 <xs:maxInclusive value="100"/>
5888 </xs:restriction>
5889 </xs:simpleType>
5890
5891 <xs:simpleType name="SingleTypeType">
5892 <xs:restriction base="xs:string">
5893 <xs:enumeration value="Scent"/>
5894 <xs:enumeration value="DNA"/>
5895 <xs:enumeration value="Ear"/>
5896 <xs:enumeration value="Face"/>
5897 <xs:enumeration value="Finger"/>
5898 <xs:enumeration value="Foot"/>
5899 <xs:enumeration value="HandGeometry"/>
5900 <xs:enumeration value="Vein"/>
5901 <xs:enumeration value="Iris"/>
5902 <xs:enumeration value="Retina"/>
5903 <xs:enumeration value="Voice"/>
5904 <xs:enumeration value="Gait"/>
5905 <xs:enumeration value="Keystroke"/>
5906 <xs:enumeration value="LipMovement"/>
5907 <xs:enumeration value="SignatureSign"/>
5908 </xs:restriction>
5909 </xs:simpleType>
5910
5911 <xs:simpleType name="MultipleTypesType">
5912 <xs:list itemType="SingleTypeType"/>
5913 </xs:simpleType>
5914
5915 <xs:simpleType name="SingleAnySubtypeType">
5916 <xs:restriction base="xs:string">
5917 <xs:enumeration value="Left"/>
5918 <xs:enumeration value="Right"/>
5919 <xs:enumeration value="Thumb"/>
5920 <xs:enumeration value="IndexFinger"/>
5921 <xs:enumeration value="MiddleFinger"/>
5922 <xs:enumeration value="RingFinger"/>
5923 <xs:enumeration value="LittleFinger"/>
5924 </xs:restriction>
5925 </xs:simpleType>
5926
5927 <xs:simpleType name="SingleVeinOnlySubtypeType">
5928 <xs:restriction base="xs:string">
5929 <xs:enumeration value="LeftVein"/>
5930 <xs:enumeration value="RightVein"/>
5931 <xs:enumeration value="Palm"/>
5932 <xs:enumeration value="BackOfHand"/>
5933 <xs:enumeration value="Wrist"/>
5934 <xs:enumeration value="Reserved1"/>
5935 <xs:enumeration value="Reserved2"/>
5936 </xs:restriction>
5937 </xs:simpleType>
5938
5939 <xs:simpleType name="MultipleAnySubtypesType">
5940 <xs:list itemType="SingleAnySubtypeType"/>
5941 </xs:simpleType>
5942
5943 <xs:simpleType name="MultipleVeinOnlySubtypesType">
5944 <xs:list itemType="SingleVeinOnlySubtypeType"/>
5945 </xs:simpleType>
5946
5947 <xs:simpleType name="SubtypeType">

```

```

5948     <xs:union memberTypes="MultipleAnySubtypesType MultipleVeinOnlySubtypesType"/>
5949 </xs:simpleType>
5950
5951 <xs:simpleType name="ProcessedLevelType">
5952   <xs:restriction base="xs:string">
5953     <xs:enumeration value="Raw"/>
5954     <xs:enumeration value="Intermediate"/>
5955     <xs:enumeration value="Processed"/>
5956   </xs:restriction>
5957 </xs:simpleType>
5958
5959 <xs:simpleType name="PurposeType">
5960   <xs:restriction base="xs:string">
5961     <xs:enumeration value="Verify"/>
5962     <xs:enumeration value="Identify"/>
5963     <xs:enumeration value="Enroll"/>
5964     <xs:enumeration value="EnrollVerify"/>
5965     <xs:enumeration value="EnrollIdentify"/>
5966     <xs:enumeration value="Audit"/>
5967   </xs:restriction>
5968 </xs:simpleType>
5969
5970 <xs:simpleType name="UUIDType">
5971   <xs:restriction base="xs:string">
5972     <xs:pattern value="[a-fA-F0-9]{8}\-([a-fA-F0-9]{4}\-){3}[a-fA-F0-9]{12}"/>
5973   </xs:restriction>
5974 </xs:simpleType>
5975 </xs:schema>
5976

```

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

B.23 Sample BIR encoding

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

```

5982 <?xml version="1.0" encoding="utf-8"?>
5983 <BIR xmlns="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/">
5984   <Version>
5985     <Major>1</Major>
5986     <Minor>0</Minor>
5987   </Version>
5988   <CBEFFVersion>
5989     <Major>2</Major>
5990     <Minor>0</Minor>
5991   </CBEFFVersion>
5992   <BIRInfo>
5993     <Creator>ABCDE</Creator>
5994     <Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
5995     <Payload>a2V2aW4ubWFiZ29sZEBuaXN0Lmdvdg==</Payload>
5996     <Integrity>true</Integrity>
5997     <CreationDate>2004-03-02T15:03:15Z</CreationDate>
5998     <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
5999     <NotValidAfter>2004-03-03T15:00:00Z</NotValidAfter>
6000   </BIRInfo>
6001   <BDBInfo>

```

```

6005
6006      <ChallengeResponse>VmlzaXQgaHR0cDovL2J3cy5uaXN0LmdvdiBmb3lgc29tZSBhd2Vzb21lIGJpb21ldH
6007      JpY3Mvd2VilHNicnZpY2UgcHJvamVjdHMh</ChallengeResponse>
6008          <Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
6009          <FormatOwner>51</FormatOwner>
6010          <FormatType>88</FormatType>
6011          <Encryption>true</Encryption>
6012          <CreationDate>2004-03-02T15:00:00Z</CreationDate>
6013          <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
6014          <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
6015          <Type>Iris</Type>
6016          <Subtype>Left</Subtype>
6017          <Level>Processed</Level>
6018          <ProductOwner>16</ProductOwner>
6019          <ProductType>2</ProductType>
6020          <Purpose>Verify</Purpose>
6021          <Quality>100</Quality>
6022      </BDBInfo>
6023      <SBInfo>
6024          <FormatOwner>51</FormatOwner>
6025          <FormatType>99</FormatType>
6026      </SBInfo>
6027      <BDB>a2V2aW4ubWFWZ29sZEBuaXN0Lmdvdg==</BDB>
6028      <SB>TmF0aW9uYWwgSW5zdGI0dXRlIG9mIFN0YW5kYXJkcyBhbmQgVG5vbG9neQ==</SB>
6029  </BIR>

```

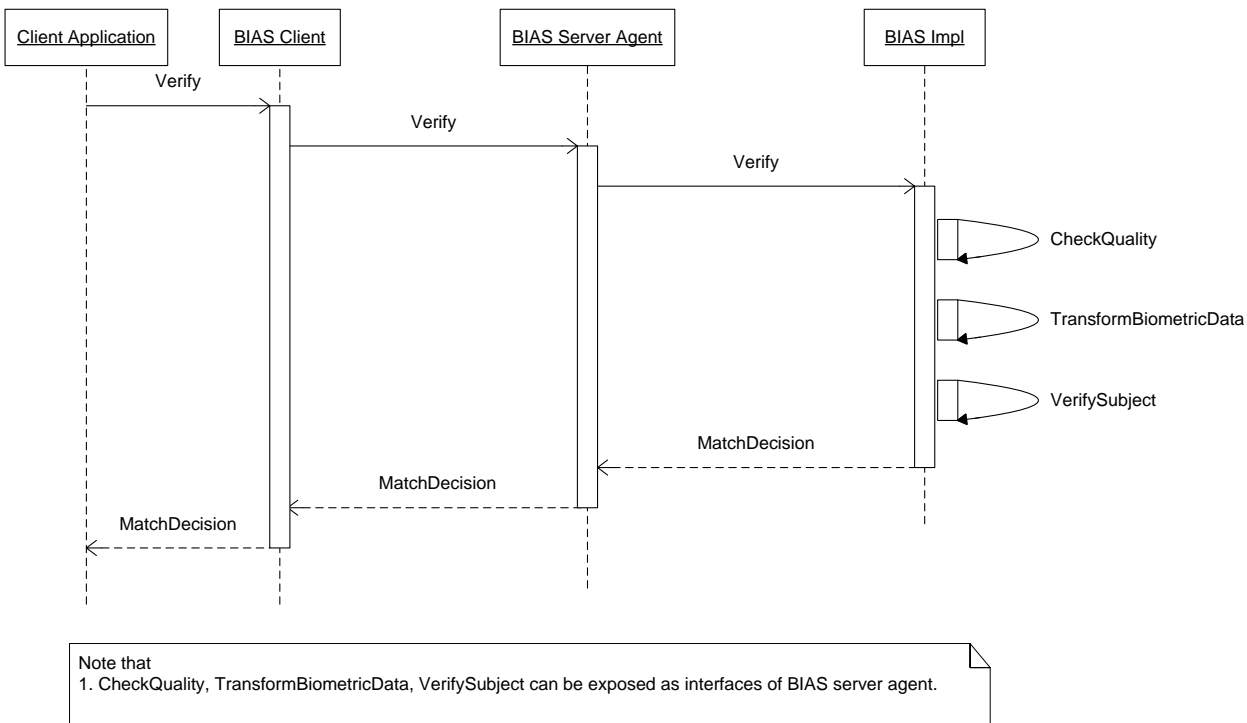
Appendix C. Use Cases (non-normative)

The intent of this annex is to provide operational sequence diagrams / flow charts that show how the higher level usage scenarios within Error! Reference source not found. could be implemented using the IAS SOAP profile. The following use cases are given:

- Verification (synchronous/aggregate)
- Verification (asynchronous/aggregate)
- Verification (primitive)
- Identification (primitive)
- Enrollment (aggregate)
- Enrollment (primitive)

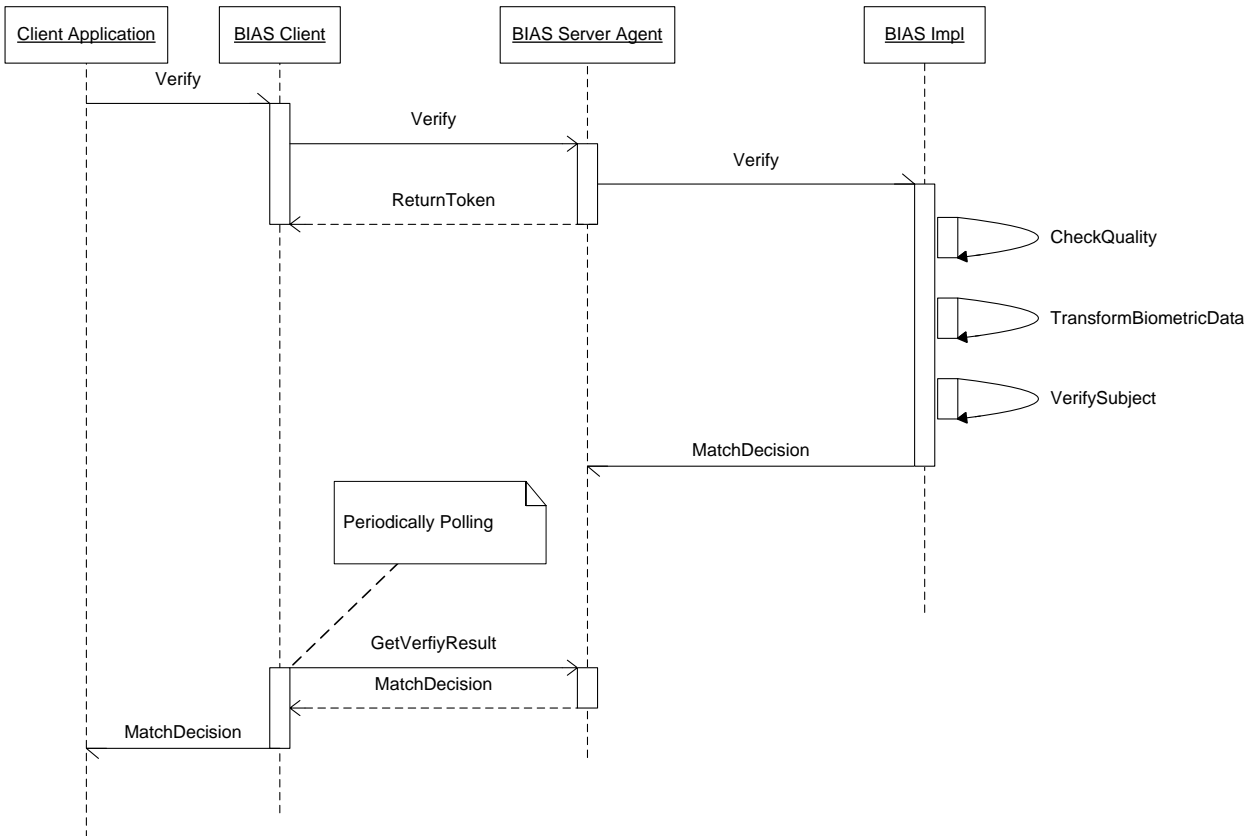
C.1 Verification Use Case

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



C.2 Asynchronous Verification Use Case

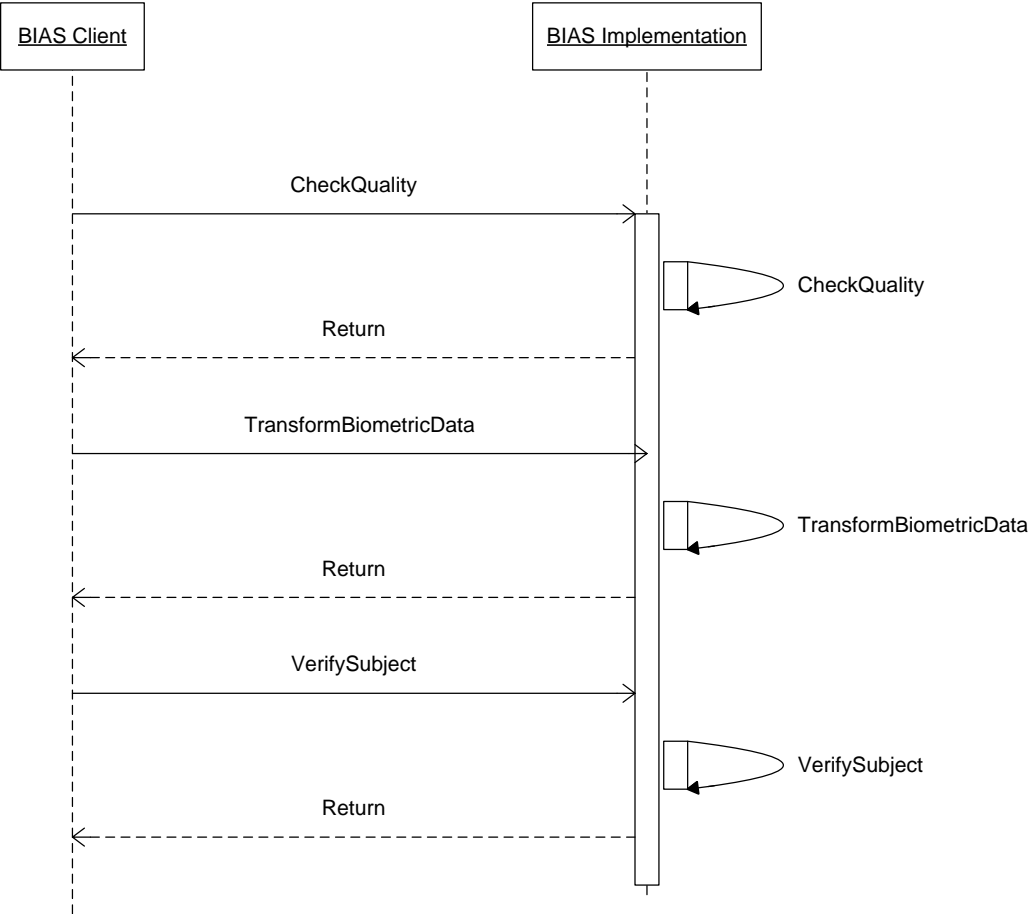
In this use case, the requester issues two requests – the BIAS Verify request to initiate the operation followed by a BIAS GetVerifyResult request to retrieve the results of that operation.



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

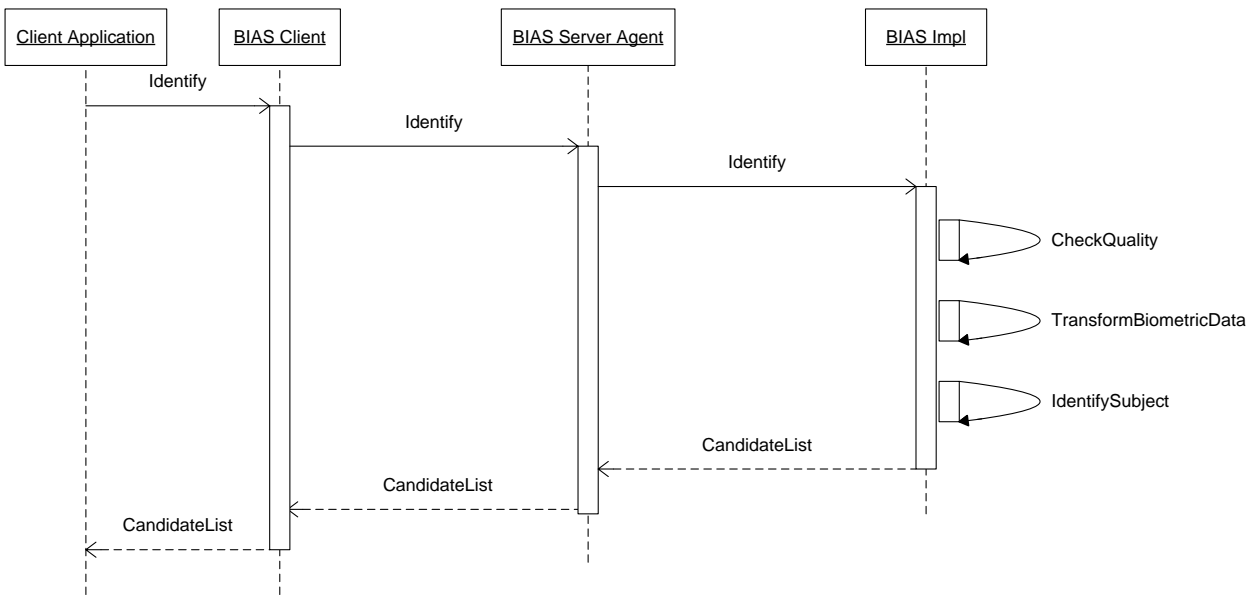
C.3 Primitive Verification Use Case

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



C.4 Identification Use Case

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

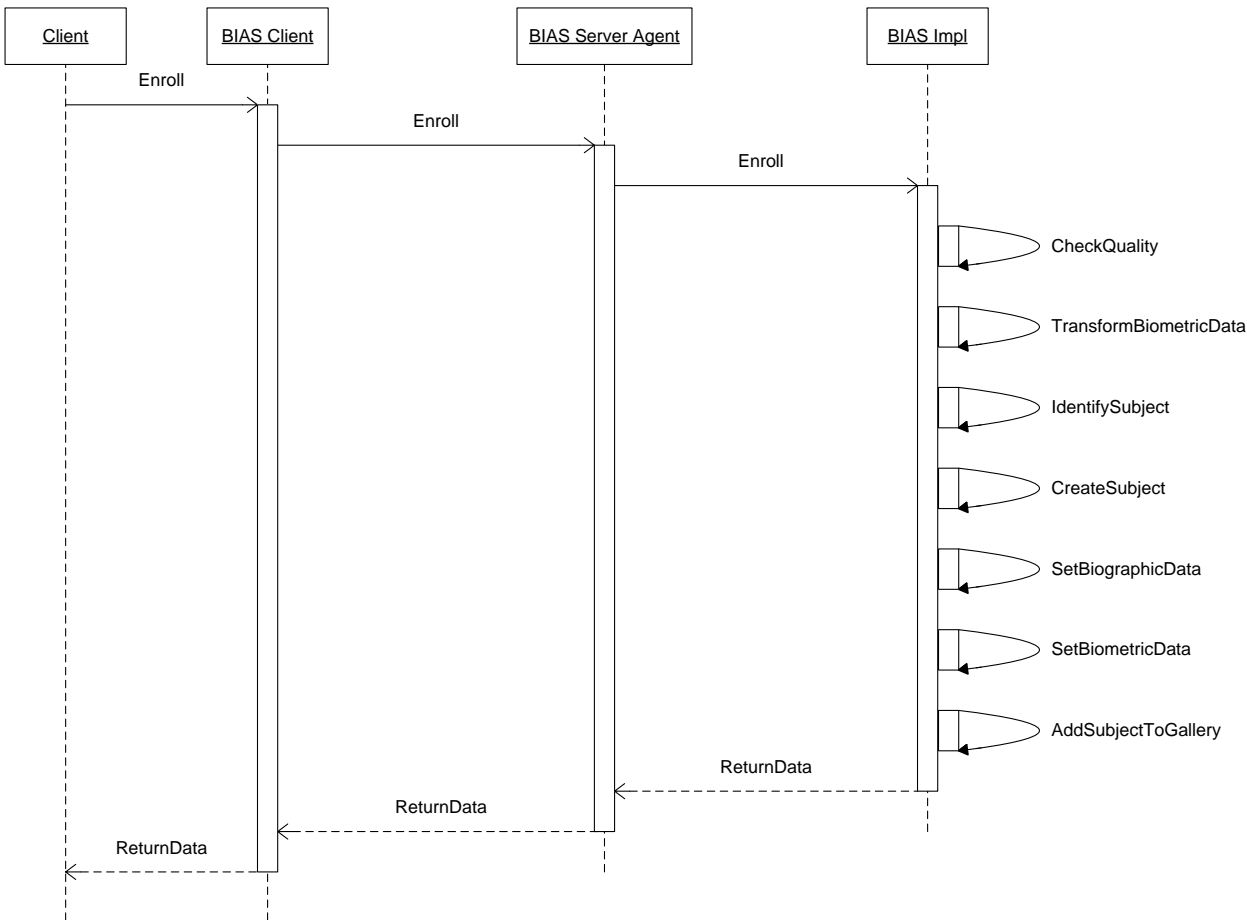


Note that
1. CheckQuality, TransformBiometricData, IdentifySubject can be exposed as interfaces of BIAS server agent.

C.5 Biometric Enrollment Use Case

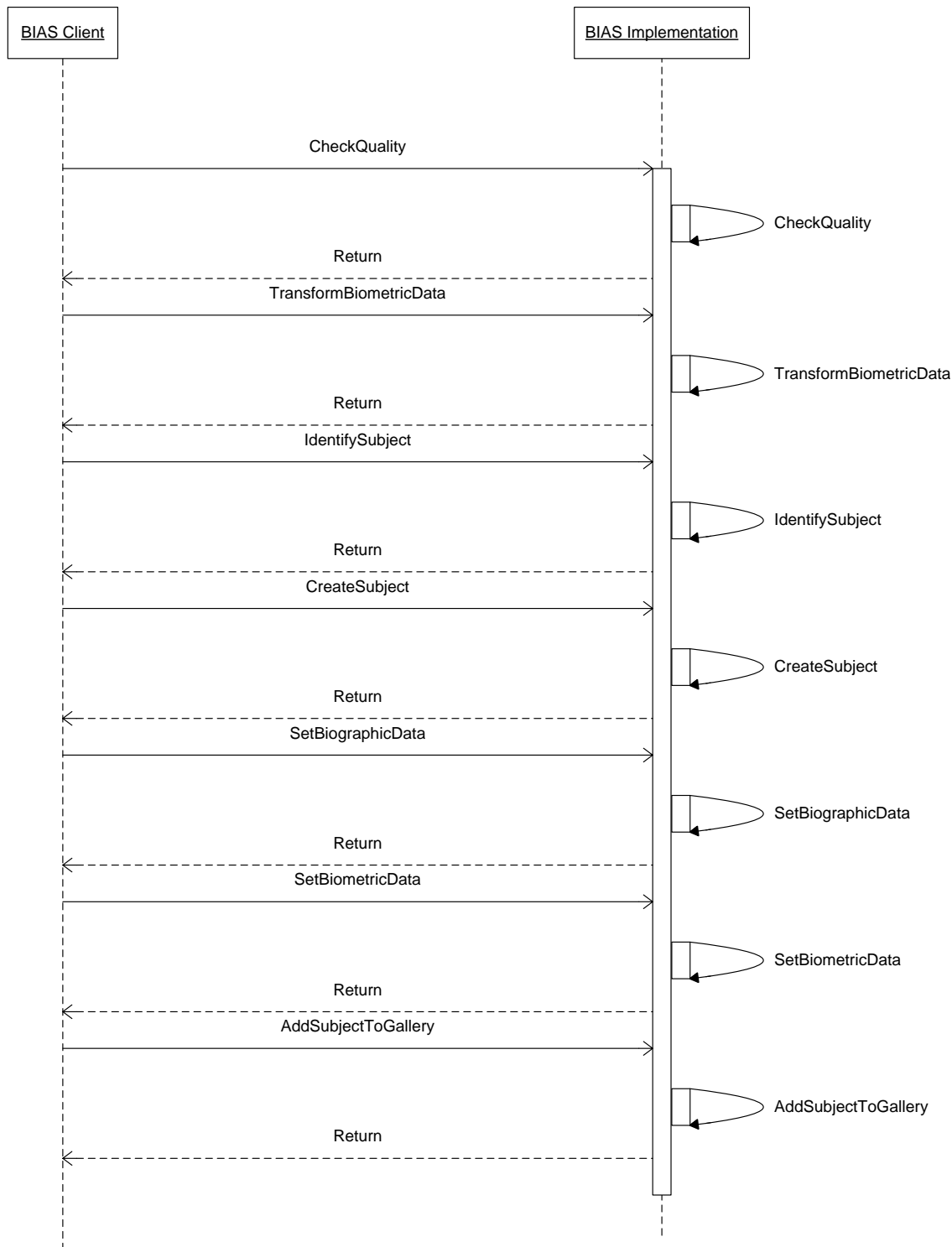
This use case uses the aggregate Enroll operation in which a single request results in some set of operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service provider.

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



C.6 Primitive Enrollment Use Case

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



Appendix D. Samples (non-normative)

D.1 Create Subject Request/Response Example

ISO/IEC BIAS Specification

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

OASIS BIAS Examples

Simple Create Subject Request:

```
POST /bias HTTP/1.1
Host: www.acme.com
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "CreateSubject"

<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <tns:CreateSubject
                                xmlns:tns="http://docs.oasis-open.org/bias/bias-
1.0/" />
  </soap:Body>
</soap:Envelope>
```

Create Subject Request with SubjectID Parameter:

```
POST /bias HTTP/1.1
Host: www.acme.com
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "CreateSubject"

<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <tns:CreateSubject xmlns:tns="http://docs.oasis-
open.org/bias/bias-1.0/">
      <tns:Identity>
        <tns:SubjectID>123456789</tns:SubjectID>
      </tns:Identity>
    </tns:CreateSubject>
  </soap:Body>
```

```

6126 </soap:Envelope>

6127 Create Subject Request with Optional OASIS BIAS Content:
6128 POST /bias HTTP/1.1
6129 Host: www.acme.com
6130 Content-Type: application/soap+xml; charset="utf-8"
6131 Content-Length: nnnn
6132 SOAPAction: "CreateSubject"
6133 <?xml version="1.0"?>
6134 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6135     <soap:Body>
6136         <tns:CreateSubject xmlns:tns="http://docs.oasis-
6137 open.org/bias/bias-1.0/">
6138             <tns:GenericRequestParameters>
6139                 <tns:Application>BIAS Application</tns:Application>
6140                 <tns:ApplicationUser>BIAS User</tns:ApplicationUser>
6141             </tns:GenericRequestParameters>
6142             <tns:Identity>
6143                 <tns:SubjectID>123456789</tns:SubjectID>
6144             </tns:Identity>
6145         </tns:CreateSubject>
6146     </soap:Body>
6147 </soap:Envelope>

6148 Simple Create Subject Response:
6149 HTTP/1.1 200 OK
6150 Content-Type: application/soap+xml; charset="utf-8"
6151 Content-Length: nnnn
6152 <?xml version="1.0"?>
6153 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6154     <soap:Body>
6155         <tns:CreateSubjectResponse xmlns:tns="http://docs.oasis-
6156 open.org/bias/bias-1.0/">
6157             <tns:ResponseStatus>
6158                 <tns:Return>0</tns:Return>
6159             </tns:ResponseStatus>
6160             <tns:Identity>
6161                 <tns:SubjectID>123456789</tns:SubjectID>
6162             </tns:Identity>
6163         </tns:CreateSubjectResponse>
6164     </soap:Body>
6165 </soap:Envelope>

6166 Create Subject Response with Optional OASIS BIAS Content:

```

```

6167 HTTP/1.1 200 OK
6168 Content-Type: application/soap+xml; charset="utf-8"
6169 Content-Length: nnnn
6170 <?xml version="1.0"?>
6171 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6172     <soap:Body>
6173         <tns:CreateSubjectResponse xmlns:tns="http://docs.oasis-
6174 open.org/bias/bias-1.0/">
6175             <tns:ResponseStatus>
6176                 <tns:Return>0</tns:Return>
6177                 <tns:Message>Subject ID 123456789 successfully
6178 created.</tns:Message>
6179             </tns:ResponseStatus>
6180             <tns:Identity>
6181                 <tns:SubjectID>123456789</tns:SubjectID>
6182             </tns:Identity>
6183         </tns:CreateSubjectResponse>
6184     </soap:Body>
6185 </soap:Envelope>
6186

```

D.2 Set Biographic Data Request/Response Example

ISO/IEC BIAS Specification

```

6189 <interface name="SetBiographicData">
6190     <parameter name="SubjectID" type="xs:string" direction="in" />
6191     <parameter name="IdentityModel" type="IdentityModelType" direction="in"
6192 />
6193     <parameter name="EncounterID" type="xs:string" direction="inout"
6194 use="optional" />
6195     <parameter name="BiographicData"
6196 type="BiographicDataType" direction="in" />
6197     <parameter name="Return" type="xs:unsignedLong" direction="out" />
6198 </interface>

```

OASIS BIAS Examples

Set Biographic Data Request:

```

6201 POST /bias HTTP/1.1
6202 Host: www.acme.com
6203 Content-Type: application/soap+xml; charset="utf-8"
6204 Content-Length: nnnn
6205 SOAPAction: "SetBiographicData"
6206 <?xml version="1.0"?>
6207 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6208     <soap:Body>

```

```

6209         <tns:SetBiographicData
6210             xmlns:tns="http://docs.oasis-open.org/bias/bias-
6211 1.0/">
6212             <tns:Identity>
6213                 <tns:SubjectID>123456789</tns:SubjectID>
6214                 <tns:BiographicData>
6215                     <tns:BiographicDataItem>
6216                         <tns:Name>Last</tns:Name>
6217                         <tns:Type>string</tns:Type>
6218                         <tns:Value>Doe</tns:Value>
6219                     </tns:BiographicDataItem>
6220                 </tns:BiographicData>
6221             </tns:Identity>
6222             <tns:IdentityModel>person</tns:IdentityModel>
6223         </tns:SetBiographicData>
6224     </soap:Body>
6225 </soap:Envelope>

```

6226 **Set Biographic Data Response:**

```

6227 HTTP/1.1 200 OK
6228 Content-Type: application/soap+xml; charset="utf-8"
6229 Content-Length: nnnn
6230 <?xml version="1.0"?>
6231 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6232     <soap:Body>
6233         <tns:SetBiographicDataResponse
6234             xmlns:tns="http://docs.oasis-open.org/bias/bias-
6235 1.0/">
6236             <tns:ResponseStatus>
6237                 <tns:Return>0</tns:Return>
6238             </tns:ResponseStatus>
6239         </tns:SetBiographicDataResponse>
6240     </soap:Body>
6241 </soap:Envelope>
6242

```

6243 **D.3 Set Biometric Data Request/Response Example**

6244 *ISO/IEC BIAS Specification*

```

6245 <interface name="SetBiometricData">
6246     <parameter name="SubjectID" type="xs:string" direction="in" />
6247     <parameter name="IdentityModel" type="IdentityModelType" direction="in"
6248 />
6249     <parameter name="EncounterID" type="xs:string" direction="inout"
6250 use="optional" />

```



```

6251         <parameter name="BIRList" type="CBEFF_BIR_ListType" direction="in" />
6252         <parameter name="Return" type="xs:unsignedLong" direction="out" />
6253     </interface>

6254     OASIS BIAS Examples

6255     Set Biometric Data Request:

6256     POST /bias HTTP/1.1
6257     Host: www.acme.com
6258     Content-Type: application/soap+xml; charset="utf-8"
6259     Content-Length: nnnn
6260     SOAPAction: "SetBiometricData"

6261     <?xml version="1.0"?>
6262     <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6263         <soap:Body>
6264             <tns:SetBiometricData
6265
6266                 xmlns:tns="http://docs.oasis-open.org/bias/bias-
6267
6268                     <tns:Identity>
6269                         <tns:SubjectID>123456789</tns:SubjectID>
6270                         <tns:BiometricData>
6271                             <tns:BIRList>
6272                                 <tns:BIR>biometric data</tns:BIR>
6273                             </tns:BIRList>
6274                         </tns:BiometricData>
6275                     </tns:Identity>
6276                     <tns:IdentityModel>person</tns:IdentityModel>
6277                 </tns:SetBiometricData>
6278             </soap:Body>
6279         </soap:Envelope>

6279     Set Biometric Data Response:

6280     HTTP/1.1 200 OK
6281     Content-Type: application/soap+xml; charset="utf-8"
6282     Content-Length: nnnn
6283     <?xml version="1.0"?>
6284     <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
6285         <soap:Body>
6286             <tns:SetBiometricDataResponse
6287
6288                 xmlns:tns="http://docs.oasis-open.org/bias/bias-
6289
6290                     <tns:ResponseStatus>
6291                         <tns:Return>0</tns:Return>
6292                     </tns:ResponseStatus>
6293             </tns:SetBiometricDataResponse>

```

6293 </soap:Body>

6294 </soap:Envelope>

Appendix E. Acknowledgements

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

Participants:

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

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

In addition, the inputs from the ISO/IEC technical committee are also gratefully appreciated.

Appendix F. Revision History

Revision	Date	Editor	Changes Made
WD 1	2015-XX-XX	Kevin Mangold	