



Biometric Identity Assurance Services (BIAS) Soap Profile Version 2.0

**Committee Specification ~~Draft 02 /~~
~~Public Review Draft 01~~**

~~30 August 2016~~

11 July 2017

Specification URIs

This version:

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/cs01/BIAS-v2.0-cs01.pdf> (Authoritative)
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/cs01/BIAS-v2.0-cs01.html>
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/cs01/BIAS-v2.0-cs01.docx>

Previous version:

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csprd01/BIAS-v2.0-csprd01.pdf> (Authoritative)
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csprd01/BIAS-v2.0-csprd01.html>
<http://docs.oasis-open.org/bioserv/BIAS/v2.0/csprd01/BIAS-v2.0-csprd01.docx>

Previous version:

~~(Authoritative)~~

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 that also includes:

- XML schemas: <http://docs.oasis-open.org/bioserv/BIAS/v2.0/cs01/schemas/>

Related work:

This specification replaces or supersedes:

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

This specification is related to:

- ISO/IEC 30108-1:2015, Biometric Identity Assurance Services (BIAS).
http://www.iso.org/iso/catalogue_detail.htm?csnumber=53228

Declared XML namespace:

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

This Committee Specification is provided under the [RAND Mode of the OASIS IPR Policy](#), the mode chosen when the Technical Committee was established. 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>).

Note that any machine-readable content ([Computer Language Definitions](#)) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product’s prose narrative document(s), the content in the separate plain text file prevails.

Citation format:

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

[BIASPROFILEBIAS-Profile-v2.0]

Biometric Identity Assurance Services (BIAS) Soap Profile Version 2.0. Edited by Kevin Mangold and Karen Marshall. ~~30 August 2016~~ [11 July 2017](#). OASIS Committee Specification ~~Draft 02/~~ ~~Public Review Draft 01~~. <http://docs.oasis-open.org/bioserv/BIAS/v2.0/cs01/BIAS-v2.0-cs01.html>. Latest version: <http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.html>.

Notices

Copyright © OASIS Open 2016~~7~~⁷. All Rights Reserved.

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

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

Table of Contents

1	Introduction	8
1.0	IPR Policy	8
1.1	Purpose/Scope	8
1.2	Overview	8
1.3	Background.....	8
1.4	Relationship to Other Standards	9
1.5	Terminology	9
1.6	References.....	10
1.6.1	Normative References.....	10
1.6.2	Non-Normative References.....	11
2	Design Concepts and Architecture (non-normative)	13
2.1	Philosophy	13
2.2	Context.....	13
2.3	Architecture.....	13
3	Data Dictionary	16
3.1	Documentation Conventions.....	16
3.2	Common Elements	17
3.2.1	ApplicationIdentifier	17
3.2.2	ApplicationUserIdentifier	17
3.2.3	BaseBIRType	17
3.2.4	BIASBiometricDataType.....	17
3.2.5	BIASFaultCode.....	18
3.2.6	BIASFaultDetail	19
3.2.7	BIASIdentity.....	19
3.2.8	BIASIDType.....	20
3.2.9	BinaryBIR	20
3.2.10	BiographicDataItem Type	20
3.2.11	BiographicDataItemListType	20
3.2.12	BiographicDataListType	20
3.2.13	BiographicDataSetType	20
3.2.14	BiographicDataType	21
3.2.15	BiometricDataType	22
3.2.16	BiometricDataListType	22
3.2.17	CandidateListResultType	23
3.2.18	CandidateListType	23
3.2.19	CandidateType	23
3.2.20	CapabilityListType	23
3.2.21	CapabilityName	24
3.2.22	CapabilityType.....	29
3.2.23	CBEFF_BIR_ListType	29
3.2.24	CBEFF_BIR_Type.....	29
3.2.25	Classification	30
3.2.26	ClassificationAlgorithmType	30

3.2.27	ClassificationData.....	31
3.2.28	DocumentDataType	31
3.2.29	DocumentDataListType.....	32
3.2.30	EncounterCategoryType	33
3.2.31	EncounterListType	33
3.2.32	FusionDecision.....	33
3.2.33	FusionIdentityListType	33
3.2.34	FusionInformationListType	34
3.2.35	FusionInformationType.....	34
3.2.36	FusionResult.....	34
3.2.37	FusionScore	34
3.2.38	GenericRequestParameters.....	35
3.2.39	IdentifySubjectResultType.....	35
3.2.40	InformationType	35
3.2.41	ListFilterType	35
3.2.42	MatchType.....	36
3.2.43	OptionType	36
3.2.44	ProcessingOptionsType	36
3.2.45	ProductID.....	36
3.2.46	QualityData.....	36
3.2.47	ResponseStatus	37
3.2.48	ReturnCode	37
3.2.49	ScoreType	37
3.2.50	TokenResultType	38
3.2.51	TokenType.....	38
3.2.52	URI_BIR	38
3.2.53	VendorIdentifier	38
3.2.54	Version	38
3.2.55	VersionType	39
3.2.56	XML_BIR	39
4	BIAS Messages	40
4.1	Primitive Operations	40
4.1.1	AddSubjectToGallery	40
4.1.2	CheckQuality	41
4.1.3	ClassifyBiometricData	44
4.1.4	CreateEncounter	46
4.1.5	CreateSubject.....	47
4.1.6	DeleteBiographicData	48
4.1.7	DeleteBiometricData	50
4.1.8	DeleteDocumentData	51
4.1.9	DeleteEncounter.....	53
4.1.10	DeleteSubject	54
4.1.11	DeleteSubjectFromGallery	55
4.1.12	GetIdentifySubjectResults	57
4.1.13	IdentifySubject.....	59

4.1.14	ListBiographicData	61
4.1.15	ListBiometricData	64
4.1.16	ListDocumentData	69
4.1.17	PerformFusion	71
4.1.18	QueryCapabilities	73
4.1.19	RetrieveBiographicData	75
4.1.20	RetrieveBiometricData	77
4.1.21	RetrieveDocumentData	79
4.1.22	SetBiographicData	81
4.1.23	SetBiometricData	83
4.1.24	SetDocumentData	85
4.1.25	TransformBiometricData	86
4.1.26	UpdateBiographicData	88
4.1.27	UpdateBiometricData	90
4.1.28	UpdateDocumentData	92
4.1.29	VerifySubject	93
4.2	Aggregate Operations	95
4.2.1	Delete	95
4.2.2	Enrol	97
4.2.3	GetDeletionResults	99
4.2.4	GetEnrolResults	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	117
5.1	Purpose and constraints	117
5.2	Message requirements	118
5.3	Handling binary data	119
5.3.1	Base64 encoding	120
5.3.2	Use of XOP	120
5.4	Discovery	120
5.5	Identifying operations	120
5.5.1	Operation name element	120
5.5.2	WS-Addressing Action	121
5.6	Security	122
5.6.1	Use of SSL 3.0 or TLS 1.0	122
5.6.2	Data Origin Authentication	122
5.6.3	Message Integrity	122
5.6.4	Message Confidentiality	123
5.6.5	CBEFF BIR security features	123
5.6.6	Security Considerations	123

5.6.7 Security of Stored Data	123
5.6.8 Key Management	123
5.7 Use with other WS* standards	124
5.8 Tailoring	124
6 Error handling	125
6.1 BIAS operation return codes	125
6.2 SOAP fault codes	125
7 Conformance	126
Appendix A. XML Schema	129
Appendix B. Use Cases (non-normative)	255
B.1 Verification Use Case	255
B.2 Asynchronous Verification Use Case	256
B.3 Primitive Verification Use Case	257
B.4 Identification Use Case	258
B.5 Biometric Enrolment Use Case	259
B.6 Primitive Enrolment Use Case	260
Appendix C. Samples (non-normative)	261
C.1 Create Subject Request/Response Example	261
C.2 Set Biographic Data Request/Response Example	263
C.3 Set Biometric Data Request/Response Example	264
Appendix D. Acknowledgements	266
Appendix E. Revision History	267

1 Introduction

1.0 IPR Policy

This Committee Specification is provided under the RAND Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established.

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>).

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), 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

88	Simple Object Access Protocol
89	SSL
90	Secure Sockets Layer
91	TLS
92	Transport Layer Security
93	UDDI
94	Universal Description, Discovery, and Integration
95	URI
96	Uniform Resource Identifier
97	VPN
98	Virtual Private Network
99	WSDL
100	Web Services Description Language
101	WSS
102	Web Services Security
103	XML
104	eXtensible Markup Language
105	CBEFF
106	Common Biometric Exchange Formats Framework - data elements and BIR formats specified in
107	ISO/IEC 19785-1
108	BIAS implementation
109	software entity that is capable of creating, processing, sending, and receiving BIAS messages
110	BIAS endpoint
111	runtime entity, identified by an endpoint URI/IRI, capable of sending and receiving BIAS
112	messages, and containing a running BIAS implementation
113	BIAS message
114	message that can be sent from a BIAS endpoint to another BIAS endpoint through a BIAS link
115	channel
116	BIAS request message
117	BIAS message conveying a request for an action to be performed by the receiving BIAS endpoint
118	BIAS response message
119	BIAS message conveying a response to a prior BIAS requestmessage

120 1.6 References

121 1.6.1 Normative References

- 122 [RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, IETF RFC 2119,
123 March 1997
124 <http://www.ietf.org/rfc/rfc2119.txt>
125
- 126 [CBEFF] ISO/IEC19785-1:2006, Information technology – Common Biometric Exchange Formats
127 Framework – Part 1: Data element specification, with Amendment 1:2010
128 <http://www.iso.org>
129

- [CBEFF-3]** ISO/IEC19785-3:2015, Information technology – Common Biometric Exchange Formats Framework – Part 3: Patron format specifications
<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>

173 **[BIO SEC]** ISO 19092 Financial services -- Biometrics -- Security framework
174 <http://www.iso.org>

175 **[EBTS-DOD]** Department of Defense Electronic Biometric Transmission Specification, Version 2.0,
176 27 March 2009
177 <http://www.biometrics.dod.mil/CurrentInitiatives/Standards/dodebts.aspx>

178 **[EBTS-FBI]** IAFIS-DOC-01078-8.1, "Electronic Biometric Transmission Specification (EBTS)",
179 Version 8.1, November 19, 2008, Federal Bureau of Investigation, Criminal Justice
180 Information Services Division
181 <https://www.fbibiospecs.org>

182 **[EFTS]** IAFIS-DOC-01078-7, "Electronic Fingerprint Transmission Specification (EFTS)", Version
183 7.1, May 2, 2005, Federal Bureau of Investigation, Criminal Justice Information Services
184 Division
185 <https://www.fbibiospecs.org>

186 **[HR-XML]** HR-XML Consortium Library, 2007 April 15
187 <http://www.hr-xml.org>

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

191 **[NIEM]** National Information Exchange Model (NIEM), Ver 2.0, June 2007, US DOJ/DHS
192 [BIAS-v2.0-csprd01.docx](#)

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

195 **[RFC2617]** J. Franks, et al, *HTTP Authentication: Basic and Digest Access Authentication*, June
196 1999
197 <http://www.ietf.org/rfc/rfc2617.txt>

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

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

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

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

208 **[WSS]** Web Services Security: SOAP Message Security 1.1, (WS-Security 2004), OASIS Standard
209 Specification, 1 February 2006
210 [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)
211 [SOAPMessageSecurity.pdf](#)

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

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

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.

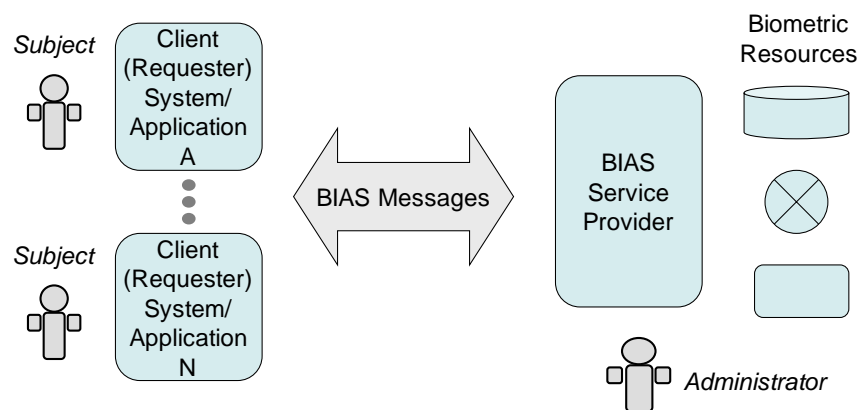


Figure 1. BIAS Context

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.

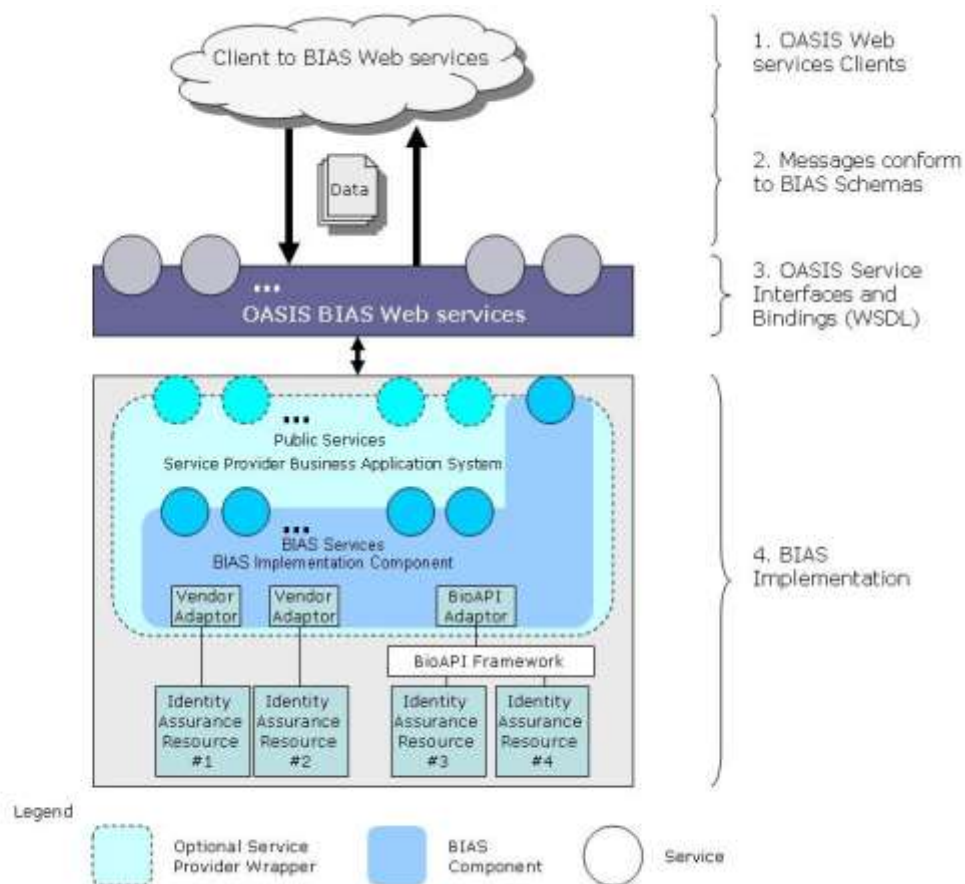


Figure 2. Representative BIAS Architecture

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

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

273 ***BIAS endpoints***

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

276 NOTE: When not publicly and directly exposed, the endpoints for purposes of this specification are
277 the BIAS service provider exposing BIAS services and the component that directly interacts with that
278 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-2.0/>. 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).

299 3.2 Common Elements

300 3.2.1 ApplicationIdentifier

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

301 3.2.2 ApplicationUserIdentifier

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

302 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.

303 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.

304 3.2.5 BIASFaultCode

Type:	String
Description:	Error code referenced in a SOAP fault.

305 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.

Value	Description
IDENTIFICATION_RESULT_NOT_YET_AVAILABLE	The result of an asynchronous identification process is not yet available.

306 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.
↗ 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.

3.2.8 BIASIDType

Type:	string
Description:	A BIAS Identifier.

3.2.9 BinaryBIR

Field	Type	#	?	Meaning
BinaryBIR	BaseBIRType		Y	Defines a BIR type of Binary
↑ Binary	base64Binary	1	Y	BIR information in base64 binary format

3.2.10 BiographicDataItem

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

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

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

3.2.12 BiographicDataList

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

3.2.13 BiographicDataSet

Field	Type	#	?	Meaning
BiographicDataSet			Y	Defines a set of biographic data that is formatted according to the specified format.

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

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.
↑ BiographicDataItem	BiographicDataItem	0..1	N	A list of biographic data elements.
↑ BiographicDataItem	BiographicDataItem	1..*	Y	A single biographic data element.
↑ BiographicDataSet	BiographicDataSet	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

- 323 • Use a pre-defined set of biographic data (e.g., as specified in another standard) using
 324 the BiographicDataSetType.
 325 See also ISO/IEC 30108-1, section 8.1 for further information.

326 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	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.

327 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.

328 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.

329 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.

330 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.

331 3.2.20 CapabilityListType

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

332 3.2.21 CapabilityName

Type:	string
Description:	A list of capability items.

333 CapabilityName Enumeration Values

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 Data, Retrieve Biometric Data, 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>Identifies the use of match scores returned by the implementing system.</p> <p>The Capability Value shall be set to the end-of-score-range that signifies a match.</p> <p>The Capability Supporting Value shall be set to the end-of-score-range that signifies a no-match.</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 score range.</p>
QualityAlgorithm	<p>A quality 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 quality algorithm.</p> <p>The Capability Description shall contain the software version of the quality algorithm.</p>
SupportedBiometric	<p>A biometric type supported by the implementing system.</p> <p>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>The Capability Supporting Value shall indicate if the implementing system supports matching for the biometric type, using one of he following values:</p> <p>"1" (identification)</p> <p>"2" (verification)</p> <p>"3" (identification and verification)</p> <p>"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>

334 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.

335 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.

336 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
\uparrow FormatType	positiveInteger	1	Y	Identifies the Patron format type.
\uparrow BIR_Information		0..1	N	Describes what is contained in a BIR.
\uparrow BIR_Info	oasis_cbeff:BIRInfoType	0..1	N	Contains information about the CBEFF-BIR.
\uparrow BDB_Info	oasis_cbeff:BDBInfoType	0..1	N	Contains information about the BDB in a simple CBEFF-BIR.
\uparrow SB_Info	oasis_cbeff:SBIInfoType	0..1	N	Contains information about the security block, if used, in a simple CBEFF-BIR.
\uparrow 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.

357 **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.

358 **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_1: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.
↑ DocumentIssuanceDate	datetime	0..1	N	The date upon which the identity document was issued.

Field	Type	#	?	Meaning
DocumentExpirationDate	datetime	0..1	N	The date upon which the identity document is no longer valid (expires).
DocumentLastName	string	0..1	N	The family name of the person to whom the identity document was issued, as contained within the document itself.
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.
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.
DocumentValidity	boolean	0..1	N	The assessed validity of the identity document (e.g. as the result of local or online validity checks).
DocumentValidityText	string	0..1	N	Details or remarks associated with the assessed validity (e.g. description of validity issue).
DocumentImage	xsd:base64binary	0..1	N	A scanned image of the subject document (e.g. passport picture page).

359 3.2.29 DocumentDataListType

Field	Type	#	?	Meaning
DocumentDataListType			Y	Provides a list of documents.

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

360 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.

361 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.

362 3.2.31 EncounterListType

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

363 3.2.32 FusionDecision

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

364 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.

365 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.

366 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.

367 3.2.36 FusionResult

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

368 3.2.37 FusionScore

Type:	ScoreType
Description:	The similarity score assigned by the matching algorithm.

369 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.

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

372 3.2.39 IdentifySubjectResultType

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

373 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	

374 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	<code>oasis_cbeff:MultipleTypesType</code>	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.

375 **3.2.42 MatchType**

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

376 **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.

377 **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.

378 **3.2.45 ProductID**

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

379 **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.

380

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.

3.2.48 ReturnCode

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

ReturnCode Enumeration Values

Value	Description
0	Success

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.

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

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 CBEFF_BIR_TYPE schema.
↑ major	nonNegativeInteger	1	Y	
↑ minor	nonNegativeInteger	1	Y	

399 **3.2.55 VersionType**

Type:	string
Description:	The version of a component.

400 **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

401

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".

Field	Type	#	?	Meaning
↑ 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.
↑ 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.

418 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.

419 4.1.2 CheckQuality

420 [CheckQualityRequest](#)

421 [CheckQualityResponse](#)

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

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

433 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.

Field	Type	#	?	Meaning
↑ 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 vender product ID.
↑ 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

434 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.

Field	Type	#	?	Meaning
↳ 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.

4.1.3 ClassifyBiometricData

[ClassifyBiometricDataRequest](#)

[ClassifyBiometricDataResponse](#)

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

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

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".

Field	Type	#	?	Meaning
↑ 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.

443 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.

4.1.4 CreateEncounter

[CreateEncounterRequest](#)

[CreateEncounterResponse](#)

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

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

Request Message

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

Field	Type	#	?	Meaning
↑ EncounterType	EncounterCategoryType	1	Y	Identifies the type of encounter during which data was collected from the subject, as determined by the requester.

456 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.
↑ 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.

457 4.1.5 CreateSubject

458 [CreateSubjectRequest](#)

459 [CreateSubjectResponse](#)

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

464 Request Message

Field	Type	#	?	Meaning
CreateSubject			Y	

Field	Type	#	?	Meaning
↑ 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".

465 Response Message

Field	Type	#	?	Meaning
CreateSubjectResponse			Y	The response to a CreateSubject operation, containing the subject ID of the new subject record.
↑ CreateSubjectResponsePackage		1	Y	
↑ 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.

466 4.1.6 DeleteBiographicData

467 [DeleteBiographicDataRequest](#)

468 [DeleteBiographicDataResponse](#)

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

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

477 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.
⬆ 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.

478 Response Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.

4.1.7 DeleteBiometricData

[DeleteBiometricDataRequest](#)

[DeleteBiometricDataResponse](#)

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

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
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.

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: "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.

492 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.

493 4.1.8 DeleteDocumentData

494 [DeleteDocumentDataRequest](#)

495 [DeleteDocumentDataResponse](#)

496 The DeleteDocumentData operation erases all of the document data of the specified category(ies)
497 associated with a given subject record. In the encounter-centric model the service erases all of the

498 document data associated with a given encounter, and therefore the encounter ID MUST be specified. If
 499 no encounter ID is specified, or it is null, document data will be removed from all encounters. If no
 500 categories are specified, then all categories (for the specified encounters) will be deleted.

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

504 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.
⌞ 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.

Field	Type	#	?	Meaning
⤴ DocumentCategory	string	1	Y	The category(ies) of the identity documents to be deleted.

505 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.

506 4.1.9 DeleteEncounter

507 [DeleteEncounterRequest](#)

508 [DeleteEncounterResponse](#)

509 The DeleteEncounter operation deletes an existing encounter record from the system. When deleting an
510 encounter, BIAS implementations MAY completely erase the information in order to prevent the ability to
511 reconstruct a record in whole or in part, or they MAY track and record the deleted information for auditing
512 and/or quality control purposes.

513 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.

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: "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.

514 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.

515 4.1.10 DeleteSubject

516 [DeleteSubjectRequest](#)

517 [DeleteSubjectResponse](#)

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

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

524 Request Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.

525 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.
↳ 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.

526 4.1.11 DeleteSubjectFromGallery

527 [DeleteSubjectFromGalleryRequest](#)

528 [DeleteSubjectFromGalleryResponse](#)

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

532 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.
↳ 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.

533 Response Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.

4.1.12 GetIdentifySubjectResults

[GetIdentifySubjectResultsRequest](#)

[GetIdentifySubjectResultsResponse](#)

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




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	
↳ 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: "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.

542 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.
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.

Field	Type	#	?	Meaning
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.

4.1.13 IdentifySubject

[IdentifySubjectRequest](#)






[IdentifySubjectResponse](#)

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

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

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

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.
 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".

Field	Type	#	?	Meaning
↑ 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.

558 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.
↑ 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
↑ 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.

NOTES:

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

4.1.14 ListBiographicData

[ListBiographicDataRequest](#)

566 [ListBiographicDataResponse](#)

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

574 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.

575 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 ↑	BiographicDataItemLi stType	0..1	N	A list of biographic data elements.
BiographicDataItem ↑	BiographicDataItemTy pe	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.
↑ Value	string	0..1	N	The value assigned to 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.

576 4.1.15 ListBiometricData

577 [ListBiometricDataRequest](#)

[ListBiometricDataResponse](#)

The ListBiometricData operation lists the biometric data elements stored for a subject using the Biometric Data List output parameter. Note that no actual biometric data is returned by this operation (see the [RetrieveBiometricData](#) operation to obtain the biometric data). In the encounter-centric model, an encounter ID MAY be specified to indicate that only the biometric 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 returns the list of encounter IDs which contain biometric data using the Encounter List output parameter, and the Biometric Data List output parameter is empty.

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

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.

Field	Type	#	?	Meaning
↳ EncounterID	BIASIDType	0..1	N	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 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.

592 Response Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.
BiometricDataList ↳	BiometricDataListType	0..1	N	A list of biometric data elements.

Field	Type	#	?	Meaning
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.

Field	Type	#	?	Meaning
BIASOperationName	string	0..1	N	Identifies the BIAS operation that is being requested: "ListDocumentData".
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.

604 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.

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

613 **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.

Field	Type	#	?	Meaning
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.

614 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.

615 4.1.18 QueryCapabilities

616 [QueryCapabilitiesRequest](#)

617 [QueryCapabilitiesResponse](#)

618 The QueryCapabilities operation returns a list of the capabilities, options, galleries, etc. that are supported
619 by the BIAS implementation. Refer to Annex A in the ISO/IEC BIAS [\[ISO/IEC-BIAS\]](#) standard for
620 conformance requirements regarding which capability names an implementation must use in the
621 QueryCapabilities operation. If the implementing system does not support a capability item, the Capability
622 Value can be set to null in the response.

623 Proprietary and additional information may be returned by returning capabilities that are not part of those
624 capabilities enumerated in the [Capability Name](#) section 3.2.21. When returning capabilities not
625 enumerated in section 3.2.21, the Capability Description should describe the capability. For each

626 capability enumerated in section 3.2.21, the Capability Name should be set to the name in the value
 627 column.

628 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".

629 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.

Field	Type	#	?	Meaning
↑ Capability	CapabilityType	0..*	N	A single capability.
↑ 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.

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.

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: "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.

640 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.

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.
⌞ Identity	BIASIdentity	1	Y	Includes the set of biographic data associated with a subject.
⌞ BiographicDataList	BiographicDataListType	1	Y	A list of biographic data associated with the subject or encounter.
⌞ BiographicData	BiographicDataType	0..*	N	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 ⌞	BiographicDataItemListType	0..1	C	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.
⌞ Value	string	0..1	N	The value assigned to the biographic data item.
BiographicDataSet ⌞	BiographicDataSetType	0..1	C	A set of biographic data information.

641 4.1.20 RetrieveBiometricData

642 [RetrieveBiometricDataRequest](#)

643 [RetrieveBiometricDataResponse](#)

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

650 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.
⌞ 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.

Field	Type	#	?	Meaning
↑ 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.

651 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.
↑ 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.

652 4.1.21 RetrieveDocumentData

653 [RetrieveDocumentDataRequest](#)

654 [RetrieveDocumentDataResponse](#)

655 The RetrieveDocumentData operation retrieves the list of document data associated with a subject ID for
656 the category(ies) specified. In the encounter-centric model, the encounter ID MAY be specified and the
657 operation returns the list of document data associated with that encounter. If the encounter ID is not
658 specified in the encounter-centric model, the operation returns the list of document information associated
659 with the most recent encounter for which document data exist. If no gallery ID is specified, document data

660 from all galleries must be returned. If no document category is specified, all documents associated with
 661 the subject (and encounter ID, if present) must be returned.

662 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	
↳ 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.

663 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.

664 4.1.22 SetBiographicData

665 [SetBiographicDataRequest](#)

666 [SetBiographicDataResponse](#)

667 The SetBiographicData operation associates biographic data to a given subject record. The identity
668 model of the system determines whether the biographic information should replace any existing
669 biographic information (person-centric model) or if a new encounter should be created and associated
670 with the subject (encounter-centric model). For encounter-centric models, the encounter ID MAY be
671 specified by the caller in order to link biographic with biometric and/or document information (assuming
672 biometric and/or document information was previously associated using the [SetBiometricData](#) and/or
673 [SetDocumentData](#) operations). If the encounter ID is omitted for the encounter-centric model, the
674 operation returns a system-assigned encounter ID.

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

678 Request Message

Field	Type	#	?	Meaning
SetBiographicData			Y	Associates biographic data to a given subject record.
↑ SetBiographicDataRequest		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: "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.
↳ 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	BiographicDataItemListType	0..1	C	A list of biographic data elements.
↳ BiographicDataSet	BiographicDataSetType	0..1	C	A set of biographic data information.

Field	Type	#	?	Meaning
↑ GalleryID	BIASIDType	0..1	N	The identifier of the gallery or population group to which the biographic will be added.

679 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.

680 4.1.23 SetBiometricData

681 [SetBiometricDataRequest](#)

682 [SetBiometricDataResponse](#)

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

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

693 Request Message

Field	Type	#	?	Meaning
SetBiometricData			Y	Associates biometric data to a given subject record.

Field	Type	#	?	Meaning
↑ 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.
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.

694 Response Message

Field	Type	#	?	Meaning
SetBiometricDataResponse			Y	The response to a SetBiometricData operation.

Field	Type	#	?	Meaning
⌞ 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.

4.1.24 SetDocumentData

[SetDocumentDataRequest](#)

[SetDocumentDataResponse](#)

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

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.
⌞ 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

Field	Type	#	?	Meaning
↑ 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.

707 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.

708 4.1.25 TransformBiometricData

709 [TransformBiometricDataRequest](#)

710 [TransformBiometricDataResponse](#)

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

713 Request Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.

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

715 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.

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.
↰ OutputBIR	CBEFF_BIR_Type	1	Y	Data structure containing the new, transformed biometric information.

4.1.26 UpdateBiographicData

[UpdateBiographicDataRequest](#)

[UpdateBiographicDataResponse](#)

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

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.
↰ BIASOperationName	string	0..1	N	Identifies the BIAS operation that is being requested: "UpdateBiographic Data".

Field	Type	#	?	Meaning
↑ 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 ↑	BiographicDataItemListType	0..1	C	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.
↑ Value	string	0..1	N	The value assigned to the biographic data item.
BiographicDataSet ↑	BiographicDataSetType	0..1	C	A set of biographic data information.

723 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.
↑ 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.

724 4.1.27 UpdateBiometricData

725 [UpdateBiometricDataRequest](#)

726 [UpdateBiometricDataResponse](#)

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

734 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.

Field	Type	#	?	Meaning
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.
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.

735 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.

4.1.28 UpdateDocumentData

[UpdateDocumentDataRequest](#)

[UpdateDocumentDataResponse](#)

The UpdateDocumentData operation updates the document data for an existing subject record. The operation replaces any existing document data of the same category with the new document data. In the 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: "UpdateDocumentData".
⌞ 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.

Field	Type	#	?	Meaning
↑ 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.

743 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.

744 4.1.29 VerifySubject

745 [VerifySubjectRequest](#)

746 [VerifySubjectResponse](#)

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

750 In the encounter-centric model, for match operations, it is not necessary to explicitly create an encounter.
751 The BIAS service provider will create the encounter and will set the encounter type to “recognition”.

752 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.

Field	Type	#	?	Meaning
↑ 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.
↑ 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.

Field	Type	#	?	Meaning
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.

753 Response Message

Field	Type	#	?	Meaning
VerifySubjectResponse			Y	The response to a VerifySubject operation.
[↑] VerifySubjectResponsePackage		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	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.

754 4.2 Aggregate Operations

755 4.2.1 Delete

756 [DeleteRequest](#)

757 [DeleteResponse](#)

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

766 Request Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.
↳ 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.

767 Response Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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.
↳ 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

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.
⌞ 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.

Response Message

Field	Type	#	?	Meaning
-------	------	---	---	---------

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	
↳ 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.

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

793 **Request Message**

Field	Type	#	?	Meaning
GetDeletionResults			Y	Retrieves the deletion results for the specified token.
↳ GetDeletionResultsRequest		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: "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.

794 **Response Message**

Field	Type	#	?	Meaning
GetDeletionResultsResponse			Y	The response to a GetDeletionResults operation.
↳ GetDeletionResultsResponsePackage		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.
↑ ReturnData	InformationType	0..1	N	Contains the output data for the response.

4.2.4 GetEnrollResults

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".

Field	Type	#	?	Meaning
↑ 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.

806 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.

807 4.2.5 GetIdentifyResults

808 [GetIdentifyResultsRequest](#)

809 [GetIdentifyResultsResponse](#)

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

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

816 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.
↳ 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.

817 Response Message

Field	Type	#	?	Meaning
GetIdentifyResultsResponse			Y	The response to a GetIdentifyResults operation.
↳ GetIdentifyResultsResponsePackage		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.
↳ 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.
↳ ReturnData	InformationType	0..1	N	Contains the output data for the response.

818 4.2.6 GetUpdateResults

819 [GetUpdateResultsRequest](#)





820 [GetUpdateResultsResponse](#)

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






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

827 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.

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

828 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.

829 4.2.7 GetVerifyResults

830 [GetVerifyResultsRequest](#)

831 [GetVerifyResultsResponse](#)

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

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

838 **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.
↳ 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.

839 **Response Message**

Field	Type	#	?	Meaning
GetVerifyResultsResponse			Y	The response to a GetVerifyResults operation.
↳ GetVerifyResultsResponsePackage		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.
↑ 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.
↑ 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.

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: "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.
↑ 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.

850 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.

Field	Type	#	?	Meaning
↑ 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.
↑ 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.

Field	Type	#	?	Meaning
↑ 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.
↑ 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.

860 Response Message

Field	Type	#	?	Meaning
RetrieveDataResponse			Y	Response to a RetrieveData operation.
↑ RetrieveDataResponsePackage		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.
↳ ReturnData	InformationType	0..1	N	Contains the output data for the response.

4.2.10 Update

[UpdateRequest](#)

[UpdateResponse](#)

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

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.

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

870 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.

Field	Type	#	?	Meaning
↑ 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.
↑ 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	

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: "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.
↑ 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.

Field	Type	#	?	Meaning
↑ 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.

886 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.
↑ 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.

Field	Type	#	?	Meaning
↱ 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.

887
888
889
890
891
892
893
894

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.

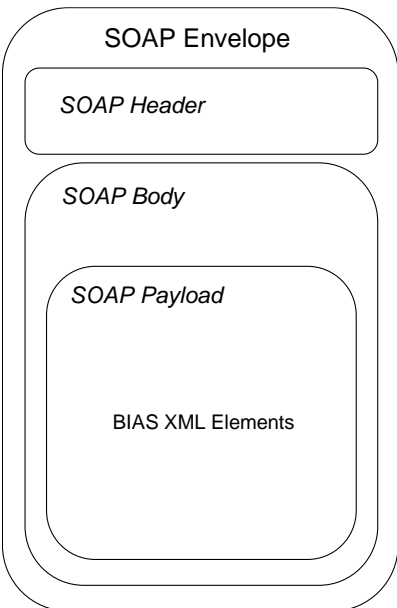


Figure 4. BIAS SOAP Structure

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:

- 925 • SOAP messages will carry BIAS XML [XML10] payloads.
- 926 • SOAP messages will follow WS-I and will deviate only when absolutely necessary.
- 927 • Message structures and interchanges will be kept as simple as possible – “nice to have”
- 928 features will be addressed in future revisions.
- 929 • XML schemas will be produced based on ISO/IEC 30108.
- 930 • BIAS will support a broad range of application domains.
- 931 • BIAS will allow for a variety of biometric and biographic data formats to be used
- 932 • Only the SOAP messaging will be defined – no message protocols or client/server agents
- 933 will be defined.
- 934 • Basic usage/formatting rules (beyond WS-I) will be defined.
- 935 • Existing biometric and Web services standards will be leveraged wherever possible.
- 936 • Sample WSDL and use cases will be provided as an aid in implementation.
- 937 • Use of basic SOAP will allow all other compatible WS* standards (and discovery
- 938 mechanisms) to be used in conjunction with BIAS messaging.
- 939 • BIAS will support both secure (i.e., using existing security mechanisms such as WS-
- 940 Security, SAML, etc,) and non-secure implementations.
- 941 • Generic biometric operations will be defined – use of biometrics within a Web services
- 942 authentication protocol is not addressed.
- 943 • OASIS namespace rules will be followed, though some external schemas MAY also be
- 944 referenced.

945 5.2 Message requirements

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

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

956

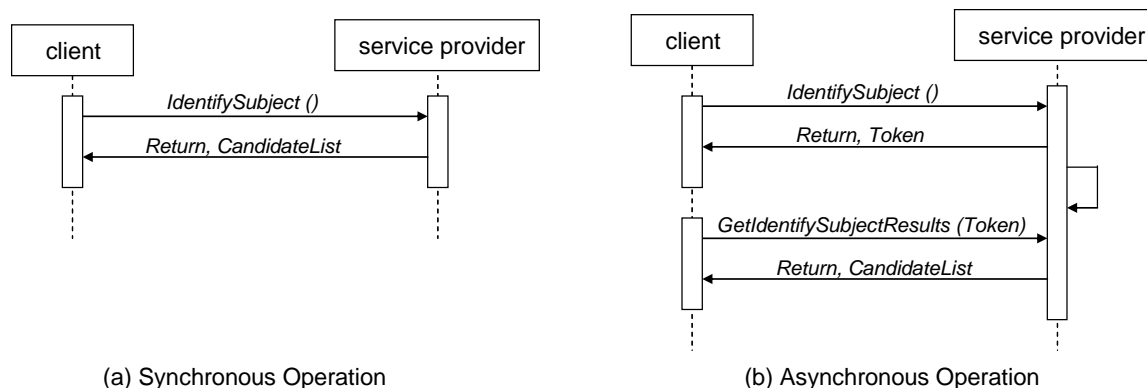


Figure 5. Example of Synchronous and Asynchronous BIAS Operations

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:


```

1031
1032 POST /bias HTTP/1.1
1033 Host: www.acme.com
1034 Content-Type: application/soap+xml; charset="utf-8"
1035 Content-Length: nnnn
1036 SOAPAction: ""
1037 <?xml version="1.0"?>
1038 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
1039     <soap:Body>
1040     <tns:CreateSubjectRequest                                xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
1041         <tns:GenericRequestParameters>
1042             <tns:Application>BIAS Application</tns:Application>
1043             <tns:ApplicationUser>BIAS User</tns:ApplicationUser>
1044             <tns:BIASOperationName>CreateSubject</tns:BIASOperationName>
1045         </tns:GenericRequestParameters>
1046         <tns:Identity>
1047             <tns:SubjectID>123456789</tns:SubjectID>
1048         </tns:Identity>
1049     </tns:CreateSubjectRequest>
1050 </soap:Body>
1051 </soap:Envelope>

```

1052 5.5.2 WS-Addressing Action

1053 WS-Addressing [WS-Addr] provides a mechanism for including action information inside any SOAP
1054 message. The information is in the SOAP Header. The WS-Addressing 'Action' element is used to
1055 indicate the intent of the message. The value is a URI/IRI identifying that intent; however, there are no
1056 restrictions on the format or specificity of the URI/IRI nor a requirement that it can be resolved. Adoption
1057 of this option also requires that the WS-Addressing 'To', 'ReplyTo', and 'MessageID' elements are
1058 supplied, as they are mandatory elements in a request-reply message pattern as used within BIAS.
1059 Response messages would also need to use WS-Addressing, requiring the 'To' (matching the 'ReplyTo'
1060 element in the request), 'RelatesTo' (matching the 'MessageID' element in the request), and
1061 'RelationshipType' (default value to "wsa:Reply") elements.

1062 Use of WS-Addressing is OPTIONAL in this profile as is this method of using the 'Action' field for this
1063 purpose. However, when BIAS is used within an environment using WS-Addressing, it is
1064 RECOMMENDED that this approach for use of the 'Action' field to carry the BIAS operation name is
1065 employed, either alone or in combination with the BIASOperationName approach described in section
1066 5.5.1.

1067 An example for a message request for the CreateSubject operation would look like the following:

```

1068
1069 POST /bias HTTP/1.1
1070 Host: www.acme.com
1071 Content-Type: application/soap+xml; charset="utf-8"
1072 Content-Length: nnnn
1073 SOAPAction: ""
1074 <?xml version="1.0"?>
1075 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">

```

```

1076 xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing">
1077   <soap:Header>
1078     <wsa:MessageID>some-ID</wsa:MessageID>
1079     <wsa:ReplyTo>
1080       <wsa:Address>response-URI</wsa:Address>
1081     </wsa:ReplyTo>
1082     <wsa:To>destination-URI</wsa:To>
1083     <wsa:Action>CreateSubject</wsa:Action>
1084   </soap:Header>
1085   <soap:Body>
1086     <tns:CreateSubjectRequest
1087       xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/" />
1088   </soap:Body>
1089 </soap:Envelope>
1090

```

1091 5.6 Security

1092 The end-points that exchange SOAP messages (or handle the contents of the BIAS operations) are
 1093 expected to be protected and trusted such that message-level security mechanisms may not be required.
 1094 The use of SSL (HTTPS) or VPN technology that provides end-point to end-point security is
 1095 RECOMMENDED and MAY be sufficient in some cases. Other mechanisms such as Signed XML or
 1096 WSS [WSS] could also be implemented.

1097 Unless stated otherwise, the following security statements apply to all BIAS bindings.

1098 5.6.1 Use of SSL 3.0 or TLS 1.0

1099 Unless otherwise specified, in any BIAS binding's use of SSL 3.0 [SSL3] or TLS1.0 [RFC2246], servers
 1100 MUST authenticate clients using a X.509 v3 certificate [X509]. The client MUST establish server identity
 1101 based on contents of the certificate (typically through examination of the certificate's subject DN field,
 1102 subjectAltName attribute, etc.).

1103 Use of transport level security in the form of SSL or TLS is OPTIONAL but highly RECOMMENDED. Use
 1104 of these mechanisms alone may not be sufficient for end-to-end integrity and confidentiality, however
 1105 (see 5.6.3 and 5.6.4 below).

1106 5.6.2 Data Origin Authentication

1107 Authentication of both the BIAS requester and the BIAS responder associated with a message is
 1108 OPTIONAL and depends on the environment of use: Authentication mechanisms available at the SOAP
 1109 message exchange layer or from the underlying substrate protocol (for example, in many bindings the
 1110 SSL/TLS or HTTP protocol) MAY be utilized to provide data origin authentication.

1111 Transport authentication will not meet end-to-end origin authentication requirements in bindings where
 1112 the BIAS SOAP message passes through an intermediary – in this case, message authentication is
 1113 RECOMMENDED.

1114 Note that SAML [SAML] MAY be used as the mechanism for parties to authenticate to one another.

1115 5.6.3 Message Integrity

1116 Message integrity of both BIAS requests and BIAS responses is OPTIONAL and depends on the
 1117 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP
 1118 message exchange layer MAY be used to ensure message integrity.

1119 Transport integrity will not meet end-to-end integrity requirements in bindings where the BIAS SOAP
1120 message passes through an intermediary – in this case, message integrity is RECOMMENDED.

1121 **5.6.4 Message Confidentiality**

1122 Message confidentiality of both BIAS requests and BIAS responses is OPTIONAL and depends on the
1123 environment of use. The security layer in the underlying substrate protocol or a mechanism at the SOAP
1124 message exchange layer MAY be used to ensure message confidentiality.

1125 Transport confidentiality will not meet end-to-end confidentiality requirements in bindings where the BIAS
1126 SOAP message passes through an intermediary.

1127 NOTE: Biometric and biographic data is likely to contain personal information the confidentiality of which
1128 SHOULD be protected accordingly. See ISO/IEC 30108, section 6.5 for further discussion.

1129 **5.6.5 CBEFF BIR security features**

1130 Within BIAS, biometric data is transferred within a CBEFF BIR (as defined in ISO/IEC 19785-1). CBEFF
1131 provides for the optional encryption of the Biometric Data Block (BDB) of the BIR and for the integrity of
1132 the entire BIR. If implemented, this is indicated in the BIR header. The BIR structure defines an optional
1133 Security Block which MAY contain a digital signature (or message authentication code), encryption
1134 parameters (e.g., key name, algorithm, etc.), and/or other security related data. Such protections are
1135 associated with an individual BIR and are separate from any other protections provided at the message
1136 level.

1137 **5.6.6 Security Considerations**

1138 Before deployment, each combination of authentication, message integrity, and confidentiality
1139 mechanisms SHOULD be analyzed for vulnerability in the context of the specific protocol exchange and
1140 the deployment environment.

1141 Special care should be given to the impact of possible caching on security.

1142 IETF RFC 2617 [RFC2617] describes possible attacks in the HTTP environment when basic or message
1143 digest authentication schemes are used.

1144 Many of the security considerations identified in [SAML SEC] MAY also apply.

1145 ISO/IEC 19092 [BIO SEC] describes a security framework for biometric systems including a minimum set
1146 of security requirements addressing integrity, authenticity, and confidentiality of biometric information
1147 during transmission and storage. These SHOULD be considered as part of an overall risk management
1148 approach.

1149 NOTE: The requirements of ISO/IEC 19092, though useful across many application domains, are
1150 required for most biometric system implementations in the financial services environment. Application of
1151 this standard would make the requirements of sections 5.5.3 through 5.5.5 mandatory rather than
1152 optional. This is highly RECOMMENDED for any high security environment or where privacy concerns
1153 exist.

1154 **5.6.7 Security of Stored Data**

1155 This specification does not address security considerations for stored data. It is the purview of the BIAS
1156 service provider to implement security mechanisms and protect data at rest as per their own security
1157 policies.

1158 **5.6.8 Key Management**

1159 This specification does not address key management considerations with respect to implementation of
1160 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:

- Identification of a subset of BIAS operations to be used
- Specification of security features to be implemented (e.g., SSL, CBEFF BIR encryption, etc.)
- Choice of operation name identification method
- Choice of BIR type to be used (XML, non-XML, or URI)
- Further definition of aggregate operations
- Use (or not) of the encounter model
- Use (or not) of asynchronous operations
- Process sequences
- 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 sections 3, 4, 5 and 6](#): [listed for the target conformance class\(es\)](#). If such implementations claim conformance to any OPTIONAL requirements and provisions stated in [Clauses sections 3, 4, 5 and 6](#), these requirements and provisions MUST be implemented as set forth in these [Clauses sections](#).

ISO/IEC 30108 [[ISO/IEC-BIAS](#)] (Annex A) specifies seven BIAS conformance classes. [In this document, these classes are outlined at a high level; more detailed information can be found in ISO/IEC 30108 \[ISO/IEC-BIAS\]](#). For 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

[Required operations and capabilities are described in Table 7.1 below.](#)

[Table 7.1 – BIAS conformance classes](#)

Service / Capability	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
Primitive Services							
Add Subject to Gallery	X		X				
Check Quality	X						
Classify Biometric Data	X						
Create Encounter	C**		C**	C**			
Create Subject	X		X	X		X	
Delete Biographic Data	X		X	X			
Delete Biometric Data	X		X	X		X	
Delete Document Data	X		X				
Delete Encounter	C**		C**	C**			
Delete Subject	X		X	X			
Delete Subject from Gallery	X		X				
Get Identify Subject Results	C*		C*			C*	
Identify Subject	X		X			X	

<u>Service / Capability</u>	<u>Class 1</u>	<u>Class 2</u>	<u>Class 3</u>	<u>Class 4</u>	<u>Class 5</u>	<u>Class 6</u>	<u>Class 7</u>
<u>List Biographic Data</u>	X		X	X			
<u>List Biometric Data</u>	X		X	X		X	
<u>List Document Data</u>	X		X				
<u>Perform Fusion</u>	X						
<u>Query Capabilities</u>	X	X	X	X	X	X	X
<u>Retrieve Biographic Data</u>	X		X	X			
<u>Retrieve Biometric Data</u>	X		X	X		X	
<u>Retrieve Document Data</u>	X		X				
<u>Set Biographic Data</u>	X		X	X			
<u>Set Biometric Data</u>	X		X	X		X	
<u>Set Document Data</u>	X		X				
<u>Transform Biometric Data</u>	X						
<u>Update Biographic Data</u>	X		X	X			
<u>Update Biometric Data</u>	X		X	X		X	
<u>Update Document Data</u>	X		X				
<u>Verify Subject</u>	X		X	X		X	
<u>Aggregate Services</u>							
<u>Delete</u>		X					X
<u>Enrol</u>		X			X		X
<u>Get Delete Results</u>		C*					C*
<u>Get Enrol Results</u>		C*			C*		C*
<u>Get Identify Results</u>		C*					C*
<u>Get Update Results</u>		C*			C*		C*
<u>Get Verify Results</u>		C*			C*		C*
<u>Identify</u>		X					X
<u>Retrieve Data</u>		X			X		
<u>Update</u>		X			X		X
<u>Verify</u>		X			X		X
<u>Capability Information Items</u>							
<u>AggregateInputDataOptional</u>		X			X		X
<u>AggregateInputDataRequired</u>		X			X		X
<u>AggregateProcessingOption</u>		X			X		X

<u>Service / Capability</u>	<u>Class 1</u>	<u>Class 2</u>	<u>Class 3</u>	<u>Class 4</u>	<u>Class 5</u>	<u>Class 6</u>	<u>Class 7</u>
<u>AggregateReturnData</u>		X			X		X
<u>AggregateServiceDescription</u>		X			X		X
<u>BiographicDataSet</u>	X	X	X	X	X		
<u>CBEFFPatronFormat</u>	X	X	X	X	X	X	X
<u>ClassificataionAlgorithmType</u>	X						
<u>ConformanceClass</u>	X	X	X	X	X	X	X
<u>Gallery</u>	X	X	X				
<u>IdentityModel</u>	X	X	X	X	X		
<u>ComparisonAlgorithm</u>	X	X	X	X	X	X	X
<u>ComparisonScore</u>	X	X	X	X	X	X	X
<u>QualityAlgorithm</u>	X						
<u>SupportedBiometric</u>	X	X	X	X	X	X	X
<u>TransformOperation</u>	X						
<u>X – Required</u> <u>C* – Conditionally required if associated implementation is asynchronous</u> <u>C** – Conditionally required if implementation uses encounter-centric model</u>							

1238

1239 In addition, the minimum capability information to be returned in response to a Query Capabilities request

1240 (the only mandatory BIAS operation across all 5 classes) is specified for each class.

1241 These conformance classes and their associated requirements apply to this BIAS SOAP Profile.

1242 There are no minimum set of operations required to be implemented by BIAS requesters; however, any

1243 operations implemented must conform to the requirements of Clausessections 3 and 4 and those

1244 requirements within Clausesection 5 that are mandatory and are not specific to BIAS responders.

1245

1246

Appendix A. XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
    Biometric Identity Assurance Services (BIAS) SOAP Profile Version 2.0
    05 August 201626 May 2017
    Copyright (c) OASIS Open 2016. All Rights Reserved.
-->
<!--
    Document      : BIAS.xml
    Created on    : December 17, 2009, 9:31 AM
    Last Edit: August 5, 2016 : May 26, 2017 by Kevin Mangold, National Institute of Standards and Technology
    Description:
        This document represents the WSDL definitions of BIAS Profile.
-->
<wsl:definitions xmlns:wsl="http://schemas.xmlsoap.org/wsl/" xmlns:soap="http://schemas.xmlsoap.org/wsl/soap/"
xmlns:http="http://schemas.xmlsoap.org/wsl/http/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:mime="http://schemas.xmlsoap.org/wsl/mime/"
xmlns:cbeff="http://standards.iso.org/iso-iec/19785/-3/ed-2/" xmlns:iso_3166_1="http://standards.iso.org/iso/3166/-1/ed-3/"
xmlns:tns="http://docs.oasis-open.org/bioserv/ns/bias-2.0"
-/ targetNamespace="http://docs.oasis-open.org/bioaserv/ns/bias-2.0">/>
    <wsl:types>
        <xsd:schema elementFormDefault="qualified" targetNamespace="http://docs.oasis-open.org/bioaserv/ns/bias-2.0">/>
            <xsd:import namespace="http://standards.iso.org/iso-iec/19785/-3/ed-2/" schemaLocation="cbeff_ed2.xsd"/>
            <xsd:import namespace="http://standards.iso.org/iso/3166/-1/ed-3/" schemaLocation="iso_country_codes.xsd"/>
            <xsd:complexType name="AggregateRequestTemplate">
                <xsd:annotation>
                    <xsd:documentation>Base template for BIAS aggregate service requests.</xsd:documentation>
                </xsd:annotation>
                <xsd:complexContent>
                    <xsd:extension base="tns:RequestTemplate">
                        <xsd:sequence>
```

```

minOccurs="1" maxOccurs="1">
    <xsd:element name="ProcessingOptions" type="tns:ProcessingOptionsType"
        <xsd:annotation>
            <xsd:documentation>Options that guide how the aggregate service
request is processed.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="InputData" type="tns:InformationType" minOccurs="1"
maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>Contains the input data for the aggregate
service request.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="AggregateResponseTemplate">
    <xsd:annotation>
        <xsd:documentation>Base template for BIAS aggregate service responses.</xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="tns:ResponseTemplate">
            <xsd:sequence>
                <xsd:element name="ReturnData" type="tns:InformationType" minOccurs="0"
maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Contains the output data for the aggregate
service response.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:simpleType name="ApplicationIdentifier">

```

```

        <xsd:annotation>
            <xsd:documentation>Identifies an application.</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:minLength value="1"/>
            <xsd:maxLength value="255"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType name="ApplicationUserIdentifier">
        <xsd:annotation>
            <xsd:documentation>Identifies an application user or instance.</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:minLength value="1"/>
            <xsd:maxLength value="255"/>
        </xsd:restriction>
    </xsd:simpleType>
    <xsd:complexType name="BIASBiometricDataType">
        <xsd:annotation>
            <xsd:documentation>Wraps the various BIAS biometric types.</xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="BIRList" type="tns:CBEFF_BIR_ListType" minOccurs="0" maxOccurs="1">
                <xsd:annotation>
                    <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
            <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
                <xsd:annotation>
                    <xsd:documentation>Contains biometric information in either a non-XML and an XML
representation.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
            <xsd:element name="InputBIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
                <xsd:annotation>

```

```

specification.</xsd:documentation>
        <xsd:documentation>Maps to specific INCITS BIAS elements as required by that
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="ReferenceBIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>Maps to specific INCITS BIAS elements as required by that
specification.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="BiometricDataList" type="tns:BiometricDataListType" minOccurs="0"
maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>A list of biometric data elements.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="BIASFaultCode">
    <xsd:annotation>
        <xsd:documentation>Error code referenced in a SOAP fault. </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="UNKNOWN_ERROR">
            <xsd:annotation>
                <xsd:documentation>The service failed for an unknown reason.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="UNSUPPORTED_CAPABILITY">
            <xsd:annotation>
                <xsd:documentation>A requested capability is not supported by the service
implementation.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="INVALID_INPUT">
            <xsd:annotation>

```

invalid.</xsd:documentation>	<xsd:documentation>The data in a service input parameter is
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="BIR_QUALITY_ERROR">
	<xsd:annotation>
succeed.</xsd:documentation>	<xsd:documentation>Biometric sample quality is too poor for the service to
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="INVALID_BIR">
	<xsd:annotation>
format.</xsd:documentation>	<xsd:documentation>The input BIR is empty or in an invalid or unrecognized
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="BIR_SIGNATURE_FAILURE">
	<xsd:annotation>
input BIR.</xsd:documentation>	<xsd:documentation>The service could not validate the signature, if used, on the
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="BIR_DECRYPTION_FAILURE">
	<xsd:annotation>
BIR.</xsd:documentation>	<xsd:documentation>The service could not decrypt an encrypted input
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="INVALID_ENCOUNTER_ID">
	<xsd:annotation>
format.</xsd:documentation>	<xsd:documentation>The input encounter ID is empty or in an invalid
	</xsd:annotation>
	</xsd:enumeration>
	<xsd:enumeration value="INVALID_SUBJECT_ID">
	<xsd:annotation>
format.</xsd:documentation>	<xsd:documentation>The input subject ID is empty or in an invalid

```

        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="UNKNOWN_SUBJECT">
        <xsd:annotation>
            <xsd:documentation>The subject referenced by the input subject ID does not
exist.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="UNKNOWN_GALLERY">
        <xsd:annotation>
            <xsd:documentation>The gallery referenced by the input gallery ID does not
exist.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="UNKNOWN_ENCOUNTER">
        <xsd:annotation>
            <xsd:documentation>The encounter referenced by the input encounter ID does not
exist.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="UNKNOWN_BIOGRAPHIC_FORMAT">
        <xsd:annotation>
            <xsd:documentation>The biographic data format is not known or not
supported.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="UNKNOWN_IDENTITY_CLAIM">
        <xsd:annotation>
            <xsd:documentation>The identity referenced by the input identity claim does not
exist.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="INVALID_IDENTITY_CLAIM">
        <xsd:annotation>
            <xsd:documentation>The identity claim requested is already in
use.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>

```

```

        </xsd:enumeration>
        <xsd:enumeration value="NONEEXISTANT_DATA">
            <xsd:annotation>
                <xsd:documentation>The data requested for deletion does not
exist.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
    </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="BIASFaultDetail">
    <xsd:annotation>
        <xsd:documentation>Defines the error information associated with a SOAP
fault.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultType" type="tns:BIASFaultCode">
            <xsd:annotation>
                <xsd:documentation>References an error code.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element minOccurs="1" maxOccurs="1" name="BIASFaultMessage" type="xsd:string">
            <xsd:annotation>
                <xsd:documentation>Provides an explanation of the fault.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element minOccurs="0" maxOccurs="1" name="BIASFaultDescription" type="xsd:string">
            <xsd:annotation>
                <xsd:documentation>Provides detailed information about a BIAS fault, such as trace
details.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BIASIdentity">
    <xsd:annotation>
        <xsd:documentation>
            Defines a single element for encapsulating the data associated

```

```

    with an Identity. Includes the Identity's reference identifiers,
    biographic data, and biometric data.
  </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="SubjectID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>A system unique identifier for a subject.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="IdentityClaim" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>An identifier by which a subject is known to a particular
gallery or population group.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>The identifier of an encounter associated with the subject,
required for encounter-centric models.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="EncounterList" type="tns:EncounterListType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>A list of encounters associated with a
subject.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>An Identity's biographic data.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <!--<xsd:element name="BiographicDataElements" type="tns:BiographicDataType" minOccurs="0" maxOccurs="1"-->
      <!--<xsd:annotation>
      <xsd:documentation>An Identity's biographic data elements that are stored in the implementing system.</xsd:documentation>

```



```

</xsd:annotation>
</xsd:element>
<xsd:element name="BiometricData" type="tns:BIASBiometricDataType" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
        <xsd:documentation>An Identity's biometric data.</xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="BIASIDType">
    <xsd:annotation>
        <xsd:documentation>A BIAS identifier</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:complexType name="BiographicDataListType">
    <xsd:annotation>
        <xsd:documentation>Defines a list of biographic data.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0"
maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation>Data structure containing information about a biographic
record.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BiographicDataItemType">
    <xsd:annotation>
        <xsd:documentation>Defines a single biographic data element.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="Name" type="xsd:string" minOccurs="1" maxOccurs="1">
            <xsd:annotation>

```

```

        <xsd:documentation>The name of the biographic data item.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="Type" type="xsd:string" minOccurs="1" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>The data type for the biographic data item.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="Value" type="xsd:string" minOccurs="0" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>The value assigned to the biographic data
item.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BiographicDataItemListType">
  <xsd:annotation>
    <xsd:documentation>Defines a list of biographic data elements.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="BiographicDataItem" type="tns:BiographicDataItemType" minOccurs="1"
maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation>Data structure containing information about a biographic
record.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BiographicDataSetType">
  <xsd:annotation>
    <xsd:documentation>Defines a set of biographic data that is formatted according to the specified
format.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string" minOccurs="1" maxOccurs="1">

```

```

        <xsd:annotation>
            <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-XML.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="version" type="xsd:string" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>The version of the biographic data format (e.g., "7.1" for FBI-
EFTS or "2.0" for NIEM).</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="source" type="xsd:string" minOccurs="1" maxOccurs="1">
        <xsd:annotation>
            <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) www.interpol.int, (NIEM) www.niem.gov, (xNAL)
www.oasis-open.org, (HR-XML) www.hr-xml.org.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="type" type="xsd:string" minOccurs="1" maxOccurs="1">
        <xsd:annotation>
            <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), XML (e.g., for NIEM, xNAL, and HR-XML or future
version of FBI-EBTS).</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:any namespace="##any">
        <xsd:annotation>
            <xsd:documentation>Biographic data formatted according to a specific
format.</xsd:documentation>
        </xsd:annotation>
    </xsd:any>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BiographicDataType">
    <xsd:annotation>
        <xsd:documentation>
            Defines a set of biographic data elements, utilizing either the
            BiographicDataItemType to represent a list of elements or the

```

BiographicDataSetType to represent a complete, formatted set of biographic information.

```

</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="LastName" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The last name of a subject.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="FirstName" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The first name of a subject.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="BiographicDataItemList" type="tns:BiographicDataItemListType" minOccurs="0"
maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>A list of biographic data elements.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="BiographicDataItem" type="tns:BiographicDataItem" minOccurs="1"
maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation>A single biographic data element.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>

```

```

        <xsd:complexType name="BiometricDataType">
            <xsd:annotation>
                <xsd:documentation>
                    Provides descriptive information about biometric data, such as
                    the biometric type, subtype, and format, contained in the BDB of
                    the CBEFF-BIR.
                </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="BiometricType" type="cbeff:MultipleTypesType" minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>The type of biological or behavioral data stored in the
                        biometric record, as defined by CBEFF.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="BiometricTypeCount" type="xsd:positiveInteger" minOccurs="0" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>The number of biometric records having the biometric type
                        recorded in the biometric type field.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="BiometricSubType" type="cbeff:SubtypeType" minOccurs="0" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>More specifically defines the type of biometric data stored in
                        the biometric record, as defined by CBEFF.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="BDBFormatOwner" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Identifies the standards body, working group, industry
                        consortium, or other CBEFF biometric organization that has defined the format for the biometric data.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="BDBFormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Identifies the specific biometric data format specified by the
                        CBEFF biometric organization recorded in the BDB Format Owner field.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>

```

```

        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="BiometricDataListType">
    <xsd:annotation>
      <xsd:documentation>A list of biometric data elements.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="BiometricData" type="tns:BiometricDataType" minOccurs="0"
maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation>Data structure containing information about a biometric
record.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="CandidateListType">
    <xsd:annotation>
      <xsd:documentation>
        Defines a set of candidates, utilizing the Candidate Type to
        represent each element in the set.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="Candidate" type="tns:CandidateType" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation>A single candidate.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="CandidateType">
    <xsd:annotation>
      <xsd:documentation>

```

Defines a single candidate as a possible match in response to a biometric identification request.

```

</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="ScoreList" minOccurs="0"><u>"</u></u>
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="Score" type="tns:ScoreType"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="Identity" type="tns:BIASIdentity"/></u></u>
        <xsd:element name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1"><u>"</u></u>
            <xsd:annotation>
                <xsd:documentation>The rank of the candidate in relation to other candidates for
the same biometric identification operation.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="BiographicData" type="tns:BiographicDataType" minOccurs="0"><u>"</u></u>
            <xsd:annotation>
                <xsd:documentation>Biographic data associated with the candidate
match.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
    <!--<xsd:attribute name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1"></xsd:attribute>-->
</xsd:complexType>
<xsd:complexType name="CapabilityListType">
    <xsd:annotation>
        <xsd:documentation>Defines a set of capabilities.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="Capability" type="tns:CapabilityType" minOccurs="0" maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation>A single capability.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>

```

```

        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType name="CapabilityName">
    <xsd:annotation>
      <xsd:documentation>A list of capability items.</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="AggregateInputDataOptional">
        <xsd:annotation>
          <xsd:documentation>A data element accepted as optional input by the implementing
system for the aggregate services.</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="AggregateInputDataRequired">
        <xsd:annotation>
          <xsd:documentation>A data element required as input by the implementing system for
the aggregate services.</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="AggregateProcessingOption">
        <xsd:annotation>
          <xsd:documentation>A processing option supported by the implementing system for
the aggregate services.</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="AggregateReturnData">
        <xsd:annotation>
          <xsd:documentation>A data element returned by the implementing system for the
aggregate services.</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="AggregateServiceDescription">
        <xsd:annotation>
          <xsd:documentation>Describes the processing logic of an aggregate service
supported by the implementing system.</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
    </xsd:restriction>
  </xsd:simpleType>

```



```

        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="BiographicDataSet">
        <xsd:annotation>
            <xsd:documentation>Identifies a biographic data set supported by the implementing
system.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="CBEFFPatronFormat">
        <xsd:annotation>
            <xsd:documentation>A patron format supported by the implementing
system.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="ClassificationAlgorithmType">
        <xsd:annotation>
            <xsd:documentation>A classification algorithm type supported by the implementing
system.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="ConformanceClass">
        <xsd:annotation>
            <xsd:documentation>Identifies the conformance class of the BIAS
implementation.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="Gallery">
        <xsd:annotation>
            <xsd:documentation>A gallery or population group supported by the implementing
system.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="IdentityModel">
        <xsd:annotation>
            <xsd:documentation>Identifies whether the implementing system is person-centric or
encounter-centric based.</xsd:documentation>
        </xsd:annotation>
    </xsd:enumeration>

```

```

        </xsd:enumeration>
        <xsd:enumeration value="MatchAlgorithm">
            <xsd:annotation>
                <xsd:documentation>A match algorithm vendor and algorithm vendor product ID
supported by the implementing system.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="MatchScore">
            <xsd:annotation>
                <xsd:documentation> Identifies the use of match scores returned by the
implementing system.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="QualityAlgorithm">
            <xsd:annotation>
                <xsd:documentation>A quality algorithm vendor and algorithm vendor product ID
supported by the implementing system.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="SupportedBiometric">
            <xsd:annotation>
                <xsd:documentation>A biometric type supported by the implementing
system.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="TransformOperation">
            <xsd:annotation>
                <xsd:documentation>A transform operation type supported by the implementing
system.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
    </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="CapabilityType">
    <xsd:annotation>
        <xsd:documentation>Defines a single capability supported by an implementing
system.</xsd:documentation>
    </xsd:annotation>

```

```

</xsd:annotation>
<xsd:sequence>
  <xsd:element name="CapabilityName" type="tns:CapabilityName" minOccurs="1" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation>The name of the capability.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="CapabilityID" type="xsd:string" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation>An identifier assigned to the capability by the implementing
system.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="CapabilityDescription" type="xsd:string" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation>A description of the capability.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="CapabilityValue" type="xsd:string" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation>A value assigned to the capability.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="CapabilitySupportingValue" type="xsd:string" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation>A secondary value supporting the
capability.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="CapabilityAdditionalInfo" type="xsd:string" minOccurs="0" maxOccurs="1">
    <xsd:annotation>
      <xsd:documentation> Contains additional information for the supported
capability.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>

```

```

<xsd:complexType name="CBEFF_BIR_ListType">
  <xsd:annotation>
    <xsd:documentation>A list of CBEFF-BIR elements.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="BIR" type="tns:CBEFF_BIR_Type" minOccurs="0" maxOccurs="unbounded">
      <xsd:annotation>
        <xsd:documentation>CBEFF structure containing information about a biometric
sample.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CBEFF_BIR_Type">
  <xsd:annotation>
    <xsd:documentation>Represents biometric information, with either a non-XML or XML
representation.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="FormatOwner" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="FormatType" type="xsd:positiveInteger" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="BIR_Information" minOccurs="0" maxOccurs="1">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="BIR_Info" type="cbeff:BIRInfoType" minOccurs="0"
maxOccurs="1"/>
          <xsd:element name="BDB_Info" type="cbeff:BDBInfoType" minOccurs="0"
maxOccurs="1"/>
          <xsd:element name="SB_Info" type="cbeff:SBInfoType" minOccurs="0"
maxOccurs="1"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  <xsd:element name="BIR" type="tns:BaseBIRType"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BaseBIRType">

```

```

</xsd:complexType>
<xsd:complexType name="BinaryBIR">
  <xsd:complexContent>
    <xsd:extension base="tns:BaseBIRType">
      <xsd:sequence>
        <xsd:element name="Binary" type="xsd:base64Binary"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="URI_BIR">
  <xsd:complexContent>
    <xsd:extension base="tns:BaseBIRType">
      <xsd:sequence>
        <xsd:element name="URI" type="xsd:anyURI"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="XML_BIR">
  <xsd:complexContent>
    <xsd:extension base="tns:BaseBIRType">
      <xsd:sequence>
        <xsd:element name="XML" type="cbeff:BIRType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:simpleType name="Classification">
  <xsd:annotation>
    <xsd:documentation>The result of a classification.</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="ClassificationAlgorithmType">
  <xsd:annotation>

```

```

        <xsd:documentation>Type of classification algorithm that was used to perform the
classification.</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:complexType name="ClassificationData">
    <xsd:annotation>
        <xsd:documentation>Contains information on classification results and the algorithm used to
determine the classification.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="Classification" type="tns:Classification" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The result of the classification.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="ClassificationAlgorithmType" type="tns:ClassificationAlgorithmType"
minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>Identifies the type of classification algorithm that was used
to perform the classification.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DocumentDataType">
    <xsd:annotation>
        <xsd:documentation>Defines a set of document data elements providing information about the
presented identity document. </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="DocumentCategory" type="xsd:string" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The type of identity document presented (e.g.
passport).</xsd:documentation>
            </xsd:annotation>
        </xsd:element>

```

```

        <xsd:element name="DocumentIDNumber" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The number associated with the identity document (e.g. passport
number).</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentIssuanceCountryCode" type="iso_3166_1:CountryAlpha2CodeType"
minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The ISO 2-character code for the country which issued the
document or from within which it was issued.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentIssuingOrganization" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The entity which issued the identity
document.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentIssuanceDate" type="xsd:dateTime" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The date upon which the identity document was
issued.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentExpirationDate" type="xsd:dateTime" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The date upon which the identity document is no longer valid
(expires).</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentLastName" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The family name of the person to whom the identity document was
issued, as contained within the document itself.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="DocumentFirstName" type="xsd:string" minOccurs="0" maxOccurs="1">

```

```

        <xsd:annotation>
            <xsd:documentation>The first given name of the person to whom the identity
document was issued, as contained within the document itself.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="DocumentMiddleName" type="xsd:string" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>The second given name of the person to whom the identity
document was issued, as contained within the document itself.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="DocumentValidity" type="xsd:boolean" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>The assessed validity of the identity document (e.g. as the
result of local or online validity checks).</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="DocumentValidityText" type="xsd:string" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>Details or remarks associated with the assessed validity (e.g.
description of validity issue).</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="DocumentImage" type="xsd:base64Binary" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>A scanned image of the subject document (e.g. passport picture
page).</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DocumentDataListType">
    <xsd:annotation>
        <xsd:documentation>Defines a list of documents. </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="DocumentData" type="tns:DocumentDataType" minOccurs="0" maxOccurs="unbounded">

```



```

        <xsd:annotation>
            <xsd:documentation>Data structure containing information about a document and
optionally an image of that document.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="EncounterCategoryType">
    <xsd:annotation>
        <xsd:documentation>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.</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Enrolment">
            <xsd:annotation>
                <xsd:documentation>The encounter is created during an enrolment
interaction.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="Recognition">
            <xsd:annotation>
                <xsd:documentation>The encounter is created during a recognition
interaction.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:enumeration value="Unspecified">
            <xsd:annotation>
                <xsd:documentation>The type of encounter is unknown.</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
        <xsd:pattern value="([a-zA-Z0-9])+"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="EncounterListType">
    <xsd:annotation>
        <xsd:documentation>Defines a set of encounters.</xsd:documentation>
    </xsd:annotation>

```

```

        <xsd:sequence>
            <xsd:element name="EncounterID" type="tns:BIASIDType" minOccurs="0" maxOccurs="unbounded">
                <xsd:annotation>
                    <xsd:documentation>The identifier of an encounter.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="FusionIdentityListType">
        <xsd:annotation>
            <xsd:documentation>
                Contains fusion input elements for one or more identities, utilizing the FusionInformationListType to represent a single
                set of fusion information for each identity.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="FusionIdentity" type="tns:FusionInformationListType" minOccurs="0"
maxOccurs="unbounded">
                <xsd:annotation>
                    <xsd:documentation>A set of fusion information for a single
identity.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="FusionInformationListType">
        <xsd:annotation>
            <xsd:documentation>
                Contains at a minimum two sets of fusion input
                elements, as input to the PerformFusion request.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="FusionElement" type="tns:FusionInformationType" minOccurs="2"
maxOccurs="unbounded">
                <xsd:annotation>
                    <xsd:documentation>A set of fusion information.</xsd:documentation>

```

```

        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="FusionInformationType">
    <xsd:annotation>
        <xsd:documentation>Represents the information necessary to perform a fusion
operation.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="BiometricType" type="cbeff:MultipleTypesType" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The type of biological or behavioral data stored in the
biometric record, as defined by CBEFF.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="BiometricSubType" type="cbeff:SubtypeType" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>More specifically defines the type of biometric data stored in
the biometric record.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="AlgorithmOwner" type="xsd:string" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The owner or vendor of the algorithm used to determine the
score or decision.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="AlgorithmType" type="xsd:string" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>The Algorithm Owner's identifier for the specific algorithm
product and version used to determine the score or decision.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="FusionResult" type="tns:FusionResult"/>
    </xsd:sequence>
</xsd:complexType>

```

```

<xsd:complexType name="FusionResult">
  <xsd:sequence/>
</xsd:complexType>
<xsd:complexType name="FusionScore">
  <xsd:complexContent>
    <xsd:extension base="tns:FusionResult">
      <xsd:sequence>
        <xsd:element name="Score" type="tns:ScoreType">
          <xsd:annotation>
            <xsd:documentation>The similarity score assigned by the matching
algorithm.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="FusionDecision">
  <xsd:complexContent>
    <xsd:extension base="tns:FusionResult">
      <xsd:sequence>
        <xsd:element name="Decision" type="xsd:string">
          <xsd:annotation>
            <xsd:documentation>The match decision assigned by the matching
algorithm.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="GenericRequestParameters">
  <xsd:annotation>
    <xsd:documentation>Common request parameters that can be used to identify the
requester.</xsd:documentation>
  </xsd:annotation>
</xsd:complexType>
</xsd:sequence>

```

```

        <xsd:element name="Application" type="tns:ApplicationIdentifier" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>Identifies the requesting application.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="ApplicationUser" type="tns:ApplicationUserIdentifier" minOccurs="0"
maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>Identifiers the user or instance of the requesting
application.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="BIASOperationName" type="xsd:string" minOccurs="0" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>Identifiers the BIAS operation name that is being
requested.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="InformationType">
    <xsd:annotation>
        <xsd:documentation>
            Allows for an unlimited number of data element types, and it does
            not specify nor require any particular data element.
        </xsd:documentation>
    </xsd:annotation>
    <!-- THE FOLLOWING COMMENTED SECTION IS PROVIDED AS AN INFORMATIVE
        EXAMPLE OF WHAT 'InformationType'S CONTENT COULD BE -->
    <xsd:sequence>
        <xsd:element name="GUID" type="xsd:string" minOccurs="0" />
        <xsd:element name="GivenName" type="xsd:string" minOccurs="0" />
        <xsd:element name="FamilyName" type="xsd:string" minOccurs="0" />
        <xsd:element name="DateOfBirth" type="xsd:string" minOccurs="0" />
        <xsd:element name="Sex" type="xsd:string" minOccurs="0" />
        <xsd:element name="Citizenship" type="xsd:string" minOccurs="0" />
    </xsd:sequence>

```

```

        <xsd:element name="Images" minOccurs="0">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="Image" minOccurs="0" maxOccurs="unbounded">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element name="ContentType" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="FingerPositionEyebrowDistance"
type="xsd:string" minOccurs="0" />
                                <xsd:element name="HandPositionLEyebrowWidth"
type="xsd:string" minOccurs="0" />
                                <xsd:element name="REyebrowWidth" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="REyebrowHeight" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="LEyebrowHeight" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="EyeDistance" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="PupilDistance" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="BridgeWidth" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="NoseAlarWidth" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="NostrilWidth" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="UpperLipHeight" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="LowerLipHeight" type="xsd:string"
minOccurs="0" />
                                <xsd:element name="ImageData"
type="xsd:base64Binary" minOccurs="0" />
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>

```

```

        <xsd:element name="Identities" minOccurs="0">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="Identity" type="xsd:string" maxOccurs="unbounded" />
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence-->
</xsd:complexType>
<xsd:complexType name="Example BIR">
    <xsd:complexContent>
        <xsd:extension base="tns:BaseBIRType">
            <xsd:sequence>
                <xsd:element name="BiometricSample" type="xsd:string"/>
                <xsd:element name="BiometricSampleType" type="xsd:string"/>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ListFilterType">
    <xsd:annotation>
        <xsd:documentation>Provides a method to filter the amount of information returned in a search of
biometric data.
    </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="BiometricTypeFilters">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="BiometricTypeFilter" type="cbeff:MultipleTypesType"
minOccurs="1" maxOccurs="unbounded">
                        <xsd:annotation>
                            <xsd:documentation>Limits the returned information to a
specific type of biometric, as defined by CBEFF.</xsd:documentation>
                        </xsd:annotation>
                    </xsd:element>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>

```

```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="IncludeBiometricSubtype" type="xsd:boolean" minOccurs="1" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>A Boolean flag indicating if biometric subtype information
should be returned.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="MatchType">
  <xsd:annotation>
    <xsd:documentation>The result of a fusion method.</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:boolean"/>
</xsd:simpleType>
<xsd:complexType name="OptionType">
  <xsd:annotation>
    <xsd:documentation>
      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.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="Key" type="xsd:string"><u></u>
      <xsd:annotation>
        <xsd:documentation>The identifier of an option supported by the implementing
system.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="Value" type="xsd:string" minOccurs="0"><u></u>
      <xsd:annotation>
        <xsd:documentation>The value for an option supported by the implementing
system.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```



```

        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ProcessingOptionsType">
    <xsd:annotation>
        <xsd:documentation>
BIAS aggregate services support the ability to include various
processing options which direct and possibly control the business
logic for that service. The ProcessingOptionsType provides a
method to represent those options. Processing options should be
defined by the implementing system.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="Option" type="tns:OptionType" minOccurs="0" maxOccurs="unbounded">
            <xsd:annotation>
                <xsd:documentation>An option supported by the implementing
system.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="ProductID">
    <xsd:annotation>
        <xsd:documentation>The vendor's ID for a particular product.</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:complexType name="QualityData">
    <xsd:annotation>
        <xsd:documentation>Contains information about a biometric sample's quality and the algorithm used
to compute the quality.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="QualityScore" type="cbeff:QualityType" minOccurs="0" maxOccurs="1">
            <xsd:annotation>

```

```

        <xsd:documentation>The quality of a biometric sample.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="AlgorithmVendor" type="tns:VendorIdentifier" minOccurs="1" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>The vendor of the qualilty algorithm used to determine the
quality score.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="AlgorithmVendorProductID" type="tns:ProductID" minOccurs="1" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>The vendor's ID for the algorithm used to determine the
quality.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="AlgorithmVersion" type="tns:VersionType" minOccurs="0" maxOccurs="1">
      <xsd:annotation>
        <xsd:documentation>The version of the algorithm used to determine the
quality.</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RequestTemplate">
  <xsd:annotation>
    <xsd:documentation>Base template for BIAS primitive service requests.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="GenericRequestParameters" type="tns:GenericRequestParameters" minOccurs="0"
maxOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ResponseStatus">
  <xsd:sequence>
    <xsd:element name="Return" type="tns:ReturnCode" minOccurs="1" maxOccurs="1">
      <xsd:annotation>

```

```

        <xsd:documentation>The return code indicates the return status of the
operation.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="Message" type="xsd:string" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>A short message corresponding to the return
code.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ResponseTemplate">
    <xsd:annotation>
        <xsd:documentation>Base template for BIAS responses.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="ResponseStatus" type="tns:ResponseStatus" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <xsd:documentation>Returned status for the operation.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="ReturnCode">
    <xsd:annotation>
        <xsd:documentation>BIAS Operation Return Codes</xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:unsignedLong">
        <xsd:enumeration value="0">
            <xsd:annotation>
                <xsd:documentation>Success</xsd:documentation>
            </xsd:annotation>
        </xsd:enumeration>
    </xsd:restriction>
</xsd:simpleType>

```

```

<!--<
    <xsd:simpleType name="ScoreType">
        <xsd:annotation>
            <xsd:documentation>Match result or quality score.</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:float"/>
    </xsd:simpleType-->

<xsd:complexType name="ScoreType">
    <xsd:annotation>
        <xsd:documentation>Match result or quality score.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>

        <xsd:element name="Value" type="xsd:float"/>
        <xsd:element name="BiometricType" type="cbeff:MultipleTypesType" minOccurs="0"/>
        <xsd:element name="BiometricSubType" type="cbeff:SubtypeType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="TokenType">
    <xsd:annotation>
        <xsd:documentation>Defines a token that is returned for asynchronous
processing.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="TokenValue" type="xsd:string" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <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>
            </xsd:annotation>
        </xsd:element>
        <xsd:element name="Expiration" type="xsd:date" minOccurs="1" maxOccurs="1">
            <xsd:annotation>
                <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>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>

```

```

        </xsd:sequence>
    </xsd:complexType>
    <xsd:simpleType name="VendorIdentifier">
        <xsd:annotation>
            <xsd:documentation>Identifies a vendor.</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
    <xsd:complexType name="Version">
        <xsd:annotation>
            <xsd:documentation>
                For a description or definition of each data element, see the
                referenced CBEFF standards in the CBEFF_XML_BIR_Type schema.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="major" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
            <xsd:element name="minor" type="xsd:nonNegativeInteger" minOccurs="1" maxOccurs="1"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:simpleType name="VersionType">
        <xsd:annotation>
            <xsd:documentation>The version of a component.</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string"/>
    </xsd:simpleType>
    <xsd:element name="AddSubjectToGallery">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>
                    Register a subject to a given gallery or population group.
                </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="AddSubjectToGalleryRequest" nillable="true">
                    <xsd:complexType>

```

```

        <xsd:complexContent>
          <xsd:extension base="tns:RequestTemplate">
            <xsd:sequence>
              <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="1" maxOccurs="1">
                <xsd:annotation>
                  <xsd:documentation>The identifier of
the gallery or population group to which the subject will be added.</xsd:documentation>
                </xsd:annotation>
              </xsd:element>
              <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                <xsd:annotation>
                  <xsd:documentation>The identity to
add to the gallery.</xsd:documentation>
                </xsd:annotation>
              </xsd:element>
            </xsd:sequence>
          </xsd:extension>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="AddSubjectToGalleryResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to an AddSubjectToGallery request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="AddSubjectToGalleryResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>

```

```

        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="CheckQuality">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Calculate a quality score for a given biometric.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="CheckQualityRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="BiometricData"
type="tns:BIASBiometricDataType" minOccurs="0" maxOccurs="1">
          <xsd:annotation>
            <xsd:documentation>Data structure
containing a single biometric sample for which a quality score is to be determined; required if no SubjectID is
provided.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
          <xsd:annotation>
            <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
provided.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element name="QualityInfo"
type="tns:QualityData" minOccurs="0" maxOccurs="1">
          <xsd:annotation>

```

```

particular algorithm vendor and vender product ID.</xsd:documentation>
<xsd:documentation>Specifies a
</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="CheckQualityResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a CheckQuality request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="CheckQualityResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="QualityInfo"
type="tns:QualityData" minOccurs="1" maxOccurs="1">
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:documentation>Contains the
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>

```



```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ClassifyBiometricData">
      <xsd:complexType>
        <xsd:annotation>
          <xsd:documentation>
            Classifies a biometric sample.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
          <xsd:element name="ClassifyBiometricDataRequest" nillable="true">
            <xsd:complexType>
              <xsd:complexContent>
                <xsd:extension base="tns:RequestTemplate">
                  <xsd:sequence>
                    <xsd:element name="BiometricData"
type="tns:BIASBiometricDataType" minOccurs="1" maxOccurs="1">
                      <xsd:annotation>
                        <xsd:documentation>Data structure
containing a single biometric sample for which the classification is to be determined.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:element>
                    <xsd:element name="ClassificationData"
type="tns:ClassificationData" minOccurs="0" maxOccurs="1">
                      <xsd:annotation>
                        <xsd:documentation>Contains the
algorithm to be used to determine the classification.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:element>
                  </xsd:sequence>
                </xsd:extension>
              </xsd:complexContent>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```

```

</xsd:element>
<xsd:element name="ClassifyBiometricDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a ClassifyBiometricData request, containing
        the classification of a biometric sample.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ClassifyBiometricDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="ClassificationData"
type="tns:ClassificationData" minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Information on the
results and type of classification performed.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="CreateEncounter">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Create a new encounter record.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:complexType>
</xsd:element>

```

```

        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="CreateEncounterRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate"/>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="CreateEncounterResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a CreateEncounter request, containing the encounter
                ID of the new encounter record.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="CreateEncounterResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:ResponseTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Contains the
encounter ID of the new encounter record.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>

```

```

        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="CreateSubject">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Create a new subject record.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="CreateSubjectRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate"/>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="CreateSubjectResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a CreateSubject request, containing the subject
                ID of the new subject record.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="CreateSubjectResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:ResponseTemplate">

```

```

minOccurs="1" maxOccurs="1">
    <xsd:sequence>
        <xsd:element name="Identity" type="tns:BIASIdentity"
            <xsd:annotation>
                <xsd:documentation>Contains the
subject ID of the new subject record.</xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteBiographicData">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Erase all of the biographic data associated with a given
                subject record or, in the encounter-centric model, with a
                given encounter.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="DeleteBiographicDataRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Contains either
the subject ID or encounter ID reference.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:sequence>
                            </xsd:extension>
                        </xsd:complexContent>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

```

```

minOccurs="0" maxOccurs="1">
    </xsd:element>
    <xsd:element name="GalleryID" type="tns:BIASIDType"
        <xsd:annotation>
            <xsd:documentation>The identifier of
the gallery or population group from which the biographic information will be deleted.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteBiographicDataResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a DeleteBiographicData request.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="DeleteBiographicDataResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:ResponseTemplate"/>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="DeleteBiometricData">
    <xsd:complexType>

```

```

        <xsd:annotation>
            <xsd:documentation>
                Erase all of the biometric data associated with a given
                subject record or, in the encounter-centric model, with a
                given encounter.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="DeleteBiometricDataRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Contains either
the subject ID or encounter ID reference.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="BiometricType"
type="cbeff:MultipleTypesType" minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>The type of
biological or behavioral data to delete, as defined by CBEFF.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>The identifier of
the gallery or population group from which the biometric information will be deleted.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:extension>
</xsd:complexContent>

```

```

        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="DeleteBiometricDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a DeleteBiometricData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteBiometricDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="DeleteDocumentData">
  <xsd:complexType>
    <xsd:annotation>
      <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>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteDocumentDataRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>

```



```

minOccurs="1" maxOccurs="1">

the subject ID or encounter ID reference.</xsd:documentation>

type="tns:DocumentDataType" minOccurs="0" maxOccurs="1">

document data elements providing information about the presented identity document. </xsd:documentation>

<xsd:element name="Identity" type="tns:BIASIdentity"
  <xsd:annotation>
    <xsd:documentation>Contains either
  </xsd:annotation>
</xsd:element>
  <xsd:element name="DocumentData"
    <xsd:annotation>
      <xsd:documentation>Defines a set of
    </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteDocumentDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a DeleteDocumentData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteDocumentDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>

```

```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="DeleteEncounter">
      <xsd:complexType>
        <xsd:annotation>
          <xsd:documentation>
            Delete an existing encounter record from the system.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
          <xsd:element name="DeleteEncounterRequest" nillable="true">
            <xsd:complexType>
              <xsd:complexContent>
                <xsd:extension base="tns:RequestTemplate">
                  <xsd:sequence>
                    <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                      <xsd:annotation>
                        <xsd:documentation>Subject ID and
encounter ID of the identity to delete.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:element>
                  </xsd:sequence>
                </xsd:extension>
              </xsd:complexContent>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="DeleteEncounterResponse">
      <xsd:complexType>
        <xsd:annotation>
          <xsd:documentation>
            The response to a DeleteEncounter request.

```

```

        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="DeleteEncounterResponsePackage" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:ResponseTemplate"/>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteSubject">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Delete an existing subject record and, in an encounter-centric
                model, any associated encounter information.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="DeleteSubjectRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Subject ID of the
identity to delete.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>

```

```

        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteSubjectResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a DeleteSubject request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteSubjectResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="DeleteSubjectFromGallery">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Remove the registration of a subject from a gallery or
        population group.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteSubjectFromGalleryRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>

```

```

        <xsd:extension base="tns:RequestTemplate">
            <xsd:sequence>
                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>The identifier of
the gallery or population group from which the subject will be deleted.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>The identity to
remove from the gallery.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteSubjectFromGalleryResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a DeleteSubjectFromGallery request.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="DeleteSubjectFromGalleryResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:ResponseTemplate"/>
                    </xsd:complexContent>

```

```

        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GetIdentifySubjectResults">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Retrieve the identification results for a specified token,
        which was returned by the Identify Subject service.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="GetIdentifySubjectResultsRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GetIdentifySubjectResultsResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>A value used to
retrieve the results of an IdentifySubject request.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

```

```

        <xsd:annotation>
            <xsd:documentation>
                The response to a GetIdentifySubjectResults request, which includes a candidate
list.

            </xsd:documentation>
        </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="GetIdentifySubjectResultsResponsePackage" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:ResponseTemplate">
                        <xsd:sequence>
                            <xsd:element name="CandidateList"
type="tns:CandidateListType" minOccurs="1" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>A rank-ordered
list of candidates that have a likelihood of matching the input biometric sample.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="IdentifySubject">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Perform an identification search against a given gallery for
                a given biometric.
            </xsd:documentation>
        </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="IdentifySubjectRequest" nillable="true">

```

```

        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <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>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="Gallery"
type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <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>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Contains the BIR,
a data structure containing the biometric sample for the search.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="MaxListSize"
type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>The maximum size
of the candidate list that should be returned.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>

```



```

        </xsd:complexType>
    </xsd:element>
    <xsd:element name="IdentifySubjectResponse">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>
                    The response to an IdentifySubject request, returning a
                    rank-ordered candidate list.
                </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="IdentifySubjectResponsePackage" nillable="true">
                    <xsd:complexType>
                        <xsd:complexContent>
                            <xsd:extension base="tns:ResponseTemplate">
                                <xsd:sequence>
                                    <xsd:element name="IdentifySubjectResult"
type="tns:IdentifySubjectResultType"/>
                                </xsd:sequence>
                            </xsd:extension>
                        </xsd:complexContent>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:complexType name="IdentifySubjectResultType">
        <xsd:sequence/>
    </xsd:complexType>
    <xsd:complexType name="CandidateListResultType">
        <xsd:complexContent>
            <xsd:extension base="tns:IdentifySubjectResultType">
                <xsd:sequence>
                    <xsd:element name="CandidateList" type="tns:CandidateListType" minOccurs="0"
maxOccurs="1">
                        <xsd:annotation>

```

```

                                <xsd:documentation>A rank-ordered list of candidates that have a
likelihood of matching the input biometric sample; returned with successful synchronous request processing.</xsd:documentation>
                                </xsd:annotation>
                                </xsd:element>
                                </xsd:sequence>
                                </xsd:extension>
                                </xsd:complexContent>
                                </xsd:complexType>
                                <xsd:complexType name="TokenResultType">
                                <xsd:complexContent>
                                <xsd:extension base="tns:IdentifySubjectResultType">
                                <xsd:sequence>
                                <xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                <xsd:documentation>A token used to retrieve the results of the
IdentifySubject request; returned with asynchronous request processing.</xsd:documentation>
                                </xsd:annotation>
                                </xsd:element>
                                </xsd:sequence>
                                </xsd:extension>
                                </xsd:complexContent>
                                </xsd:complexType>
                                <xsd:element name="ListBiographicData">
                                <xsd:complexType>
                                <xsd:annotation>
                                <xsd:documentation>
                                Lists the biographic data elements stored for a subject.
                                </xsd:documentation>
                                </xsd:annotation>
                                <xsd:sequence>
                                <xsd:element name="ListBiographicDataRequest" nillable="true">
                                <xsd:complexType>
                                <xsd:complexContent>
                                <xsd:extension base="tns:RequestTemplate">
                                <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">

```

```

        <xsd:annotation>
            <xsd:documentation>Identifies the
subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="EncounterType"
type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
            <xsd:documentation>Identifies the
category of encounter. If an encounter ID is not specified
the subject, the operation returns the list of encounter IDs of a specific
data using the Encounter List output parameter, and the Biographic
parameter is empty. Should not be used in conjunction with EncounterID.
and encounter data exists for
type which contain biographic
Data Elements output
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListBiographicDataResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                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.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:complexType>
</xsd:element>

```

```

        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="ListBiographicDataResponsePackage" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:ResponseTemplate">
                        <xsd:sequence>
                            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>
                                        Contains a list of biographic
                                        subject or encounter; non-
                                        successful, biographic data
                                        person-centric model is being
                                        encounter-centric model is
                                        identifier was specified.
                                    </xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListBiometricData">
    <xsd:complexType>
        <xsd:annotation>

```

data elements associated with a

empty if the service was

exists, and either (a) the

used or (b) the

being used and an encounter

```

        <xsd:documentation>
            Lists the biometric data elements stored for a subject. Note
            that no actual biometric data is returned by this service (see
            the RetrieveBiometricData service to obtain the biometric
            data).
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="ListBiometricDataRequest" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:RequestTemplate">
                        <xsd:sequence>
                            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>Identifies the
subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                            <xsd:element name="EncounterType"
type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>Identifies the
encounter data exists for the
type which contain biometric
List output parameter is
empty. Should not be used in conjunction with EncounterID.
                                </xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                            <xsd:element name="ListFilter"
type="tns:ListFilterType" minOccurs="0" maxOccurs="1">
                                <xsd:annotation>

```

```

biometric information should be returned.</xsd:documentation>
<xsd:documentation>Indicates what
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListBiometricDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a ListBiometricData request, 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.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ListBiometricDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity"
type="tns:BIASIdentity">
</xsd:annotation>
<xsd:documentation>
  Includes a list of biometric
data elements associated

```

or a list of encounter ID's

which contain biometric

~~data.~~

with a subject or encounter

associated with a subject and

~~data.~~

```
</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ListDocumentData">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Lists the document categories stored for a subject.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ListDocumentDataRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:annotation>
</xsd:documentation>Identifies the
subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
</xsd:annotation>
```

type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">	</xsd:element> <xsd:element name="EncounterType"
category of encounter. If an encounter ID is not specified and encounter data exists for the	<xsd:annotation>
the list of encounter IDs which contain document data using the Encounter List Output parameter,	<xsd:documentation>Identifies the
parameter shall be empty. Should not be used in conjunction with EncounterID.	subject, the operation shall return
	and the Document Data Elements output
	</xsd:documentation>
	</xsd:annotation>
	</xsd:element>
	</xsd:sequence>
	</xsd:extension>
	</xsd:complexContent>
	</xsd:complexType>
	</xsd:element>
	</xsd:sequence>
	</xsd:complexType>
	</xsd:element>
	<xsd:element name="ListDocumentDataResponse">
	<xsd:complexType>
	<xsd:annotation>
categories stored for a subject.	<xsd:documentation>
only document data categories	The response to a ListDocumentData request, containing a list of document
	In the encounter-centric model, an encounterID may be specified to indicate that
	stored for that encounter should be returned. If an encounter ID is not
	specified and encounter data exists for the subject, the list
	of encounter IDs which contain document data is returned.
	</xsd:documentation>
	</xsd:annotation>
	<xsd:sequence>
	<xsd:element name="ListDocumentDataResponsePackage" nillable="true">
	<xsd:complexType>


```

        <xsd:complexContent>
          <xsd:extension base="tns:ResponseTemplate">
            <xsd:sequence>
              <xsd:element name="DocumentDataList"
type="tns:DocumentDataListType" minOccurs="0" maxOccurs="1">
                <xsd:annotation>
                  <xsd:documentation>
                    A list of document categories
                    associated with a subject or encounter; non-empty if the service
                    exists, and either the person-centric model is being used or the
                    encounter-centric model is
                    being used and an encounter identifier was specified.
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:element>
              <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                <xsd:annotation>
                  <xsd:documentation>
                    A list of encounter IDs
                    associated with a subject and which contain document data; non-empty
                    successful, document data exists, the encounter-centric model is being used,
                    and an encounter identifier
                    was not specified..
                  </xsd:documentation>
                </xsd:annotation>
              </xsd:element>
            </xsd:sequence>
          </xsd:extension>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PerformFusion">

```

```

        <xsd:complexType>
          <xsd:annotation>
            <xsd:documentation>
              Accepts either match score or match decision information and creates a fused match
result.
            </xsd:documentation>
          </xsd:annotation>
          <xsd:sequence>
            <xsd:element name="PerformFusionRequest" nillable="true">
              <xsd:complexType>
                <xsd:complexContent>
                  <xsd:extension base="tns:RequestTemplate">
                    <xsd:sequence>
                      <xsd:element name="FusionInput"
type="tns:FusionIdentityListType" minOccurs="1" maxOccurs="1">
                        <xsd:annotation>
                          <xsd:documentation>Score or decision
input information to the fusion method for each identity.</xsd:documentation>
                        </xsd:annotation>
                      </xsd:element>
                    </xsd:sequence>
                  </xsd:extension>
                </xsd:complexContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="PerformFusionResponse">
        <xsd:complexType>
          <xsd:annotation>
            <xsd:documentation>
              The response to the PerformFusion request.
            </xsd:documentation>
          </xsd:annotation>
          <xsd:sequence>
            <xsd:element name="PerformFusionResponsePackage" nillable="true">

```

```

        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Match" type="tns:MatchType"
minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Indicates the
result of the fusion method</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="QueryCapabilities">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Returns a list of the capabilities, options, galleries, etc.
        that are supported by the BIAS implementation.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="QueryCapabilitiesRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

```

```

        </xsd:complexType>
    </xsd:element>
    <xsd:element name="QueryCapabilitiesResponse">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>
                    The response to a QueryCapabilities request.
                </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="QueryCapabilitiesResponsePackage" nillable="true">
                    <xsd:complexType>
                        <xsd:complexContent>
                            <xsd:extension base="tns:ResponseTemplate">
                                <xsd:sequence>
                                    <xsd:element name="CapabilityList"
type="tns:CapabilityListType" minOccurs="1" maxOccurs="1">
                                        <xsd:annotation>
                                            <xsd:documentation>A list of
capabilities supported by the BIAS implementation.</xsd:documentation>
                                        </xsd:annotation>
                                    </xsd:element>
                                </xsd:sequence>
                            </xsd:extension>
                        </xsd:complexContent>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="RetrieveBiographicData">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>
                    Retrieves the biographic data associated with a subject ID.
                </xsd:documentation>
            </xsd:annotation>
        </xsd:complexType>
    </xsd:element>

```

```

</xsd:annotation>
<xsd:sequence>
  <xsd:element name="RetrieveBiographicDataRequest" nillable="true">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="tns:RequestTemplate">
          <xsd:sequence>
            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>Identifies the
subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="EncounterType"
type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>Identifies the
type of encounter during which data was collected from the subject,
as determined by the requester.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:element>
          <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
            <xsd:annotation>
              <xsd:documentation>The identifier of
the gallery or population group from which the biographic information will be retrieved.</xsd:documentation>
            </xsd:annotation>
          </xsd:element>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

```

```

</xsd:element>
<xsd:element name="RetrieveBiographicDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a RetrieveBiographicData request,
        containing the biographic data associated with a subject ID. In
        the encounter-centric model, the biographic data associated with
        a specified encounter is returned. If the encounter ID is not
        specified in the encounter-centric model, the biographic
        information associated with the most recent encounter is returned.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="RetrieveBiographicDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Includes the set
of biographic data associated with a subject.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="RetrieveBiometricData">
  <xsd:complexType>

```

```

<xsd:annotation>
  <xsd:documentation>
    Retrieves the biometric data associated with a subject ID.
  </xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element name="RetrieveBiometricDataRequest" nillable="true">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="tns:RequestTemplate">
          <xsd:sequence>
            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>Identifies the
subject or, in the encounter-centric model, a subject and an encounter.</xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="EncounterType"
type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>Identifies the
type of encounter during which data was collected from the subject,
as determined by the requester.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>The identifier of
the gallery or population group from which the biometric information will be retrieved.</xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="BiometricType"
type="cbeff:MultipleTypesType" minOccurs="0" maxOccurs="1">
              <xsd:annotation>

```



```

        </xsd:sequence>
        </xsd:extension>
        </xsd:complexContent>
        </xsd:complexType>
        </xsd:element>
        </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="RetrieveDocumentData">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>Retrieves the list of document data associated with a subject ID for
the category(ies) specified. </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="RetrieveDocumentDataRequest" nillable="true">
                    <xsd:complexType>
                        <xsd:complexContent>
                            <xsd:extension base="tns:RequestTemplate">
                                <xsd:sequence>
                                    <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                        <xsd:annotation>
                                            <xsd:documentation>Contains either
the subject ID or encounter ID reference.</xsd:documentation>
                                        </xsd:annotation>
                                    </xsd:element>
                                    <xsd:element name="DocumentData"
type="tns:DocumentDataType" minOccurs="0" maxOccurs="1">
                                        <xsd:annotation>
                                            <xsd:documentation>Defines a set of
document data elements providing information about the requested identity document. </xsd:documentation>
                                        </xsd:annotation>
                                    </xsd:element>
                                    <xsd:element name="EncounterType"
type="tns:EncounterCategoryType" minOccurs="0" maxOccurs="1">
                                        <xsd:annotation>

```

```

category of encounter.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
</xsd:annotation>
<xsd:documentation>The identifier of
the gallery or population group from which the biographic information will be retrieved. </xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="RetrieveDocumentDataResponse">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>
The response to a RetrieveDocumentData request.
</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="RetrieveDocumentDataResponsePackage" nillable="true">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="tns:ResponseTemplate"/>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="DocumentDataList" type="tns:DocumentDataListType" minOccurs="1"
maxOccurs="1">
<xsd:annotation>

```

```

        <xsd:documentation>A list of document data associated with a subject or
        encounter</xsd:documentation>

        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="SetBiographicData">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Associates biographic data to a given subject record.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="SetBiographicDataRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <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>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>The identifier of
the gallery or population group to which the biographic will be added.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>

```

```

        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SetBiographicDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to a SetBiographicData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="SetBiographicDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>In an encounter-
centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SetBiometricData">
  <xsd:complexType>
    <xsd:annotation>

```

```

        <xsd:documentation>
            Associates biometric data to a given subject record.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="SetBiometricDataRequest" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:RequestTemplate">
                        <xsd:sequence>
                            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                <xsd:annotation>
                                    <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>
                                </xsd:annotation>
                            </xsd:element>
                            <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>The identifier of
the gallery or population group to which the biometric will be added.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="SetBiometricDataResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a SetBiometricData request.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:complexType>

```

```

        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="SetBiometricDataResponsePackage" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:ResponseTemplate">
                        <xsd:sequence>
                            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>In an encounter-
centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="SetDocumentData">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Associates document data to a given subject record.
            </xsd:documentation>
        </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="SetDocumentDataRequest" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:RequestTemplate">
                        <xsd:sequence>

```

```

minOccurs="1" maxOccurs="1">
    <xsd:element name="Identity" type="tns:BIASIdentity"
        <xsd:annotation>
            <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>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="GalleryID" type="tns:BIASIDType"
        <xsd:annotation>
            <xsd:documentation>The identifier of
the gallery or population group to which the document data will be added.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    <xsd:element name="DocumentDataList"
        <xsd:annotation>
            <xsd:documentation>Defines a set of
document data elements to associate with the subject or encounter. </xsd:documentation>
        </xsd:annotation>
    </xsd:element>
    </xsd:sequence>
    </xsd:extension>
    </xsd:complexContent>
    </xsd:complexType>
    </xsd:element>
    </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="SetDocumentDataResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a SetDocumentData request.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:sequence>

```

```

        <xsd:element name="SetDocumentDataResponsePackage" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:ResponseTemplate">
                        <xsd:sequence>
                            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>In an encounter-
centric model, identifies the encounter ID assigned to a new encounter.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransformBiometricData">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Transforms or processes a given biometric in one format into a new target format.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="TransformBiometricDataRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="InputBIR"
type="tns:CBEFF_BIR_Type" minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>

```



```

containing the biometric information to be transformed.</xsd:documentation>
<xsd:documentation>Data structure
</xsd:annotation>
</xsd:element>
<xsd:element name="TransformOperation"
type="xsd:unsignedLong" minOccurs="1" maxOccurs="1">
<xsd:annotation>
<xsd:documentation>Value indicating
the type of transformation to perform.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="TransformControl"
type="xsd:string" minOccurs="0" maxOccurs="1">
<xsd:annotation>
<xsd:documentation> Specifies
controls for the requested transform operation.</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="TransformBiometricDataResponse">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>
The response to a TransformBiometricData request.
</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="TransformBiometricDataResponsePackage" nillable="true">
<xsd:complexType>
<xsd:complexContent>

```

```

        <xsd:extension base="tns:ResponseTemplate">
            <xsd:sequence>
                <xsd:element name="OutputBIR"
type="tns:CBEFF_BIR_Type" minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Data structure
containing the new, transformed biometric information.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UpdateBiographicData">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                Updates the biographic data for a given subject record.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="UpdateBiographicDataRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:RequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <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>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

```

```

        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UpdateBiographicDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to an UpdateBiographicData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="UpdateBiographicDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="UpdateBiometricData">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Updates a single biometric sample for a given subject record.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="UpdateBiometricDataRequest" nillable="true">

```

```

        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <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>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="Merge" type="xsd:boolean"
minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Value indicating
if the input biometric sample should be merged with any existing biometric information.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="UpdateBiometricDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to an UpdateBiometricData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="UpdateBiometricDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>

```

```

        <xsd:extension base="tns:ResponseTemplate"/>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="UpdateDocumentData">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Updates the document data for a given subject record.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="UpdateDocumentDataRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <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>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="DocumentDataList"
type="tns:DocumentDataListType" minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>Defines a set of
updated document data elements to associate with the subject or encounter. </xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>

```

```

        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="UpdateDocumentDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The response to an UpdateDocumentData request.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="UpdateDocumentDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:ResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="VerifySubject">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        Performs a 1:1 verification match between a given biometric and
        either a claim to identity in a given gallery or another given
        biometric.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="VerifySubjectRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>

```

```

        <xsd:extension base="tns:RequestTemplate">
            <xsd:sequence>
                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>The identifier of
the gallery or population group of which the subject must be a member.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Includes the
identifying information and/or input and reference biometric samples.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
                <xsd:element name="InputData"
type="tns:InformationType" minOccurs="0" maxOccurs="1">
                    <xsd:annotation>
                        <xsd:documentation>Contains the
output data for the response.</xsd:documentation>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="VerifySubjectResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The response to a VerifySubject request.
            </xsd:documentation>

```

```

</xsd:annotation>
<xsd:sequence>
  <xsd:element name="VerifySubjectResponsePackage" nillable="true">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="tns:ResponseTemplate">
          <xsd:sequence>
            <xsd:element name="Match" type="xsd:boolean"
minOccurs="1" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>Indicates if the
Input BIR matched either the biometric information associated with the Identity Claim or the Reference BIR.</xsd:documentation>
              </xsd:annotation>
            </xsd:element>
            <xsd:element name="Score" type="tns:ScoreType"
minOccurs="0" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>The score if the
biometric information matched.</xsd:documentation>
              </xsd:annotation>
            </xsd:element>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Delete">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
The Delete operation deletes an existing subject or, in an encounter-centric model, an
existing encounter from the
system.
    </xsd:documentation>
  </xsd:annotation>

```



```

</xsd:annotation>
<xsd:sequence>
  <xsd:element name="DeleteRequest" nillable="true">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="tns:AggregateRequestTemplate">
          <xsd:sequence>
            <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
              <xsd:annotation>
                <xsd:documentation>The identifier for
the subject, or in encounter-centric model the encounter to be
centric models.</xsd:documentation>
                deleted; required for encounter-
              </xsd:annotation>
            </xsd:element>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeleteResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>The response to an Delete request.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="DeleteResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">

```

```

the subject, or in encounter-centric model, the encounter being deleted;
models.</xsd:documentation>

minOccurs="0" maxOccurs="1">

retrieve the results of the Delete request.</xsd:documentation>

<xsd:annotation>
  <xsd:documentation>The identifier for
    required for encounter-centric
  </xsd:annotation>
</xsd:element>
<xsd:element name="Token" type="tns:TokenType"

  <xsd:annotation>
    <xsd:documentation>A value used to
  </xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Enroll">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The Enroll aggregate service adds a new subject or, in an
        encounter-centric model, a new encounter to the system.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="EnrollRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateRequestTemplate">
              <xsd:sequence>

```

```

minOccurs="0" maxOccurs="1">
    <xsd:element name="Identity" type="tns:BIASIdentity"
        <xsd:annotation>
            <xsd:documentation>The identifier for
the encounter; required for encounter-centric models.</xsd:documentation>
        </xsd:annotation>
    </xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="EnrollResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>The response to an Enroll request.</xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="EnrollResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateResponseTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
                                    <xsd:annotation>
                                        <xsd:documentation>The identifier for
the encounter; required for encounter-centric models.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="Token" type="tns:TokenType"
                                    <xsd:annotation>

```

```

        <xsd:documentation>A value used to
retrieve the results of the Enroll request.</xsd:documentation>

        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GetDeletionResults">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The GetDeletionResults operation retrieves the deletion results for the specified
token.

      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="GetDeletionResultsRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">

              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>

```

```

        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="GetDeletionResultsResponse">
    <xsd:complexType>
      <xsd:annotation>
        <xsd:documentation>The response to a GetDeletionResults request.</xsd:documentation>
      </xsd:annotation>
      <xsd:sequence>
        <xsd:element name="GetDeletionResultsResponsePackage" nillable="true">
          <xsd:complexType>
            <xsd:complexContent>
              <xsd:extension base="tns:ResponseTemplate">
                <xsd:sequence>
                  <xsd:element name="ReturnData"
type="tns:InformationType" minOccurs="0" maxOccurs="1">
                    <xsd:annotation>
                      <xsd:documentation>Contains the
output data for the response.</xsd:documentation>
                    </xsd:annotation>
                  </xsd:element>
                </xsd:sequence>
              </xsd:extension>
            </xsd:complexContent>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="GetEnrollResults">
    <xsd:complexType>
      <xsd:annotation>
        <xsd:documentation>
          The GetEnrollResults aggregate service retrieves the enrollment
          results for the specified token.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:complexType>
  </xsd:element>

```

```

        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="GetEnrollResultsRequest" nillable="true">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="tns:RequestTemplate">
                        <xsd:sequence>
                            <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">
                                <xsd:annotation>
                                    <xsd:documentation>A value used to
retrieve the results of the Enroll request.</xsd:documentation>
                                </xsd:annotation>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GetEnrollResultsResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>The response to a GetEnrollResults request.</xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="GetEnrollResultsResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateResponseTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">

```

```

the subject, and in encounter-centric models, the encounter.</xsd:documentation>
<xsd:annotation>
  <xsd:documentation>The identifier for
the subject, and in encounter-centric models, the encounter.</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GetIdentifyResults">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The GetIdentifyResults aggregate service retrieves the
        identification results for the specified token.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="GetIdentifyResultsRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
          <xsd:annotation>
            <xsd:documentation>A value used to
retrieve the results of the Identify request.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>

```

```

        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GetIdentifyResultsResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>The response to a GetIdentifyResults request.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="GetIdentifyResultsResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>The identifier of
the encounter, if assigned.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="CandidateList"
type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>A rank-ordered
list of candidates that have a likelihood of matching the
input biometric
sample.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```



```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="GetUpdateResults">
      <xsd:complexType>
        <xsd:annotation>
          <xsd:documentation>
            The GetUpdateResults operation retrieves the update results for the specified
token.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
          <xsd:element name="GetUpdateResultsRequest" nillable="true">
            <xsd:complexType>
              <xsd:complexContent>
                <xsd:extension base="tns:RequestTemplate">
                  <xsd:sequence>
                    <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">
                      <xsd:annotation>
                        <xsd:documentation>A value used to
retrieve the results of the Update request.</xsd:documentation>
                      </xsd:annotation>
                    </xsd:element>
                  </xsd:sequence>
                </xsd:extension>
              </xsd:complexContent>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="GetUpdateResultsResponse">
      <xsd:complexType>
        <xsd:annotation>
          <xsd:documentation>The response to a GetUpdateResults request.</xsd:documentation>
        </xsd:annotation>

```

```

        <xsd:sequence>
            <xsd:element name="GetUpdateResultsResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateResponseTemplate">
                            <xsd:sequence>
                                </xsd:sequence>
                            </xsd:extension>
                        </xsd:complexContent>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="GetVerifyResults">
        <xsd:complexType>
            <xsd:annotation>
                <xsd:documentation>
                    The GetVerifyResults aggregate service retrieves the verification
                    results for the specified token.
                </xsd:documentation>
            </xsd:annotation>
            <xsd:sequence>
                <xsd:element name="GetVerifyResultsRequest" nillable="true">
                    <xsd:complexType>
                        <xsd:complexContent>
                            <xsd:extension base="tns:RequestTemplate">
                                <xsd:sequence>
                                    <xsd:element name="Token" type="tns:TokenType"
minOccurs="1" maxOccurs="1">
                                        </xsd:sequence>
                                    <xsd:annotation>
                                        <xsd:documentation>A value used to
retrieve the results of the Verify request.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:sequence>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>

```

```

        </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GetVerifyResultsResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>The response to a GetVerifyResults request.</xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="GetVerifyResultsResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateResponseTemplate">
                            <xsd:sequence>
                                <xsd:element name="Match" type="xsd:boolean"
minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Indicates if the
Input BIR matched either the biometric information
the Reference BIR.</xsd:documentation>
                                        associated with the Identity Claim or
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="Score" type="tns:ScoreType"
minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>The score if the
biometric information matched.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="0" maxOccurs="1">
                                    <xsd:annotation>

```

```
<xsd:documentation>The identifier of
the encounter, if assigned.</xsd:documentation>

</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Identify">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The Identify aggregate service performs an identification
        function according to system requirements and/or resources.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="IdentifyRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateRequestTemplate">
              <xsd:sequence>
                <xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
                  <xsd:annotation>
                    <xsd:documentation>The identifier of
the gallery or population group which will be searched.</xsd:documentation>
                  </xsd:annotation>
                </xsd:element>
                <xsd:element name="MaxListSize"
type="xsd:positiveInteger" minOccurs="1" maxOccurs="1">
                  <xsd:annotation>
```

```

of the candidate list that should be returned.</xsd:documentation>
</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="IdentifyResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>The response to an Identify request.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="IdentifyResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateResponseTemplate">
              <xsd:sequence>
                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="40" maxOccurs="1">
              </xsd:sequence>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
type="tns:CandidateListType" minOccurs="0" maxOccurs="1">
  <xsd:annotation>
    <xsd:documentation>The identifier of
the encounter, if assigned.</xsd:documentation>
  </xsd:annotation>
</xsd:element>
<xsd:element name="CandidateList"
  <xsd:annotation>
    <xsd:documentation>A rank-ordered
list of candidates that have a likelihood of matching the input
successful, synchronous processing.</xsd:documentation>
    </xsd:annotation>
  </xsd:element>

```

```

minOccurs="0" maxOccurs="1">

retrieve the results of the Identify request.</xsd:documentation>

</xsd:annotation>
</xsd:element>
<xsd:element name="Token" type="tns:TokenType"

<xsd:annotation>
  <xsd:documentation>A value used to

</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="RetrieveData">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>
        The RetrieveData aggregate service retrieves requested
        information about a subject, or in an encounter-centric model
        about an encounter.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="RetrieveDataRequest" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:RequestTemplate">
              <xsd:sequence>
                <xsd:element name="ProcessingOptions"

type="tns:ProcessingOptionsType" minOccurs="1" maxOccurs="1">

<xsd:annotation>
  <xsd:documentation>Options that guide

how the service request is processed, and may identify what

```

```

returned.</xsd:documentation>

type(s) of information should be

</xsd:annotation>
</xsd:element>
<xsd:element name="Identity" type="tns:BIASIdentity"

minOccurs="1" maxOccurs="1">

<xsd:annotation>
<xsd:documentation>Includes the

identifier of the subject or encounter.</xsd:documentation>

</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="RetrieveDataResponse">
  <xsd:complexType>
    <xsd:annotation>
      <xsd:documentation>The response to a RetrieveData request.</xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="RetrieveDataResponsePackage" nillable="true">
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="tns:AggregateResponseTemplate"/>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Update">
  <xsd:complexType>

```

```

        <xsd:annotation>
            <xsd:documentation>
                The Update operation updates specified information about a subject, or in an
                encounter-centric model about an encounter.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="UpdateRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateRequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
minOccurs="1" maxOccurs="1">
                                    <xsd:annotation>
                                        <xsd:documentation>Includes the
identifier of the subject or encounter.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="UpdateResponse">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>The response to a Update request.</xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="UpdateResponsePackage" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>

```



```

minOccurs="0" maxOccurs="1">
    <xsd:extension base="tns:AggregateResponseTemplate">
        <xsd:sequence>
            <xsd:element name="Token" type="tns:TokenType"
                <xsd:annotation>
                    <xsd:documentation>A value used to
                        retrieve the results of the Update request.</xsd:documentation>
                </xsd:annotation>
            </xsd:element>
        </xsd:sequence>
    </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Verify">
    <xsd:complexType>
        <xsd:annotation>
            <xsd:documentation>
                The Verify aggregate service performs a 1:1 verification
                function according to system requirements and/or resources.
                Either the IdentityClaim or ReferenceBIR input data elements in
                the Identity parameter are required.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="VerifyRequest" nillable="true">
                <xsd:complexType>
                    <xsd:complexContent>
                        <xsd:extension base="tns:AggregateRequestTemplate">
                            <xsd:sequence>
                                <xsd:element name="Identity" type="tns:BIASIdentity"
                                    <xsd:annotation>

```

```

the IdentityClaim or ReferenceBIR.</xsd:documentation>
</xsd:annotation>
</xsd:element>
<xsd:element name="GalleryID" type="tns:BIASIDType"
minOccurs="0" maxOccurs="1">
</xsd:annotation>
</xsd:documentation>The identifier of
the gallery or population group of which the subject must be a member.</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="VerifyResponse">
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>The response to a Verify request.</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
<xsd:element name="VerifyResponsePackage" nillable="true">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="tns:AggregateResponseTemplate">
<xsd:sequence>
<xsd:element name="Match" type="xsd:boolean"
minOccurs="0" maxOccurs="1">
</xsd:annotation>
</xsd:documentation>Indicates if the
Input BIR matched either the biometric information
the Reference BIR.</xsd:documentation>
associated with the Identity Claim or

```

```

minOccurs="0" maxOccurs="1">

biometric information matched.</xsd:documentation>

minOccurs="0" maxOccurs="1">

the encounter, if assigned.</xsd:documentation>

minOccurs="0" maxOccurs="1">

retrieve the results of the Verify request; </xsd:documentation>

</xsd:annotation>
</xsd:element>
<xsd:element name="Score" type="tns:ScoreType"

<xsd:annotation>
<xsd:documentation>The score if the

</xsd:annotation>
</xsd:element>
<xsd:element name="Identity" type="tns:BIASIdentity"

<xsd:annotation>
<xsd:documentation>The identifier of

</xsd:annotation>
</xsd:element>
<xsd:element name="Token" type="tns:TokenType"

<xsd:annotation>
<xsd:documentation>A value used to

</xsd:annotation>
</xsd:element>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="BIASFault" type="tns:BIASFaultDetail"/>
</xsd:schema>
</wsdl:types>
<wsdl:message name="BIASFaultMessage">
<wsdl:part name="parameters" element="tns:BIASFault"/>
</wsdl:message>

```

```

<wsdl:message name="AddSubjectToGalleryRequestMessage">
    <wsdl:part name="parameters" element="tns:AddSubjectToGallery"/>
</wsdl:message>
<wsdl:message name="AddSubjectToGalleryResponseMessage">
    <wsdl:part name="parameters" element="tns:AddSubjectToGalleryResponse"/>
</wsdl:message>
<wsdl:message name="CheckQualityRequestMessage">
    <wsdl:part name="parameters" element="tns:CheckQuality"/>
</wsdl:message>
<wsdl:message name="CheckQualityResponseMessage">
    <wsdl:part name="parameters" element="tns:CheckQualityResponse"/>
</wsdl:message>
<wsdl:message name="ClassifyBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:ClassifyBiometricData"/>
</wsdl:message>
<wsdl:message name="ClassifyBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:ClassifyBiometricDataResponse"/>
</wsdl:message>
<wsdl:message name="CreateSubjectRequestMessage">
    <wsdl:part name="parameters" element="tns:CreateSubject"/>
</wsdl:message>
<wsdl:message name="CreateSubjectResponseMessage">
    <wsdl:part name="parameters" element="tns:CreateSubjectResponse"/>
</wsdl:message>
<wsdl:message name="DeleteBiographicDataRequestMessage">
    <wsdl:part name="parameters" element="tns>DeleteBiographicData"/>
</wsdl:message>
<wsdl:message name="DeleteBiographicDataResponseMessage">
    <wsdl:part name="parameters" element="tns>DeleteBiographicDataResponse"/>
</wsdl:message>
<wsdl:message name="DeleteBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns>DeleteBiometricData"/>
</wsdl:message>
<wsdl:message name="DeleteBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns>DeleteBiometricDataResponse"/>
</wsdl:message>

```

```

<wsdl:message name="DeleteSubjectRequestMessage">
    <wsdl:part name="parameters" element="tns:DeleteSubject"/>
</wsdl:message>
<wsdl:message name="DeleteSubjectResponseMessage">
    <wsdl:part name="parameters" element="tns:DeleteSubjectResponse"/>
</wsdl:message>
<wsdl:message name="DeleteSubjectFromGalleryRequestMessage">
    <wsdl:part name="parameters" element="tns:DeleteSubjectFromGallery"/>
</wsdl:message>
<wsdl:message name="DeleteSubjectFromGalleryResponseMessage">
    <wsdl:part name="parameters" element="tns:DeleteSubjectFromGalleryResponse"/>
</wsdl:message>
<wsdl:message name="GetIdentifySubjectResultsRequestMessage">
    <wsdl:part name="parameters" element="tns:GetIdentifySubjectResults"/>
</wsdl:message>
<wsdl:message name="GetIdentifySubjectResultsResponseMessage">
    <wsdl:part name="parameters" element="tns:GetIdentifySubjectResultsResponse"/>
</wsdl:message>
<wsdl:message name="IdentifySubjectRequestMessage">
    <wsdl:part name="parameters" element="tns:IdentifySubject"/>
</wsdl:message>
<wsdl:message name="IdentifySubjectResponseMessage">
    <wsdl:part name="parameters" element="tns:IdentifySubjectResponse"/>
</wsdl:message>
<wsdl:message name="ListBiographicDataRequestMessage">
    <wsdl:part name="parameters" element="tns:ListBiographicData"/>
</wsdl:message>
<wsdl:message name="ListBiographicDataResponseMessage">
    <wsdl:part name="parameters" element="tns:ListBiographicDataResponse"/>
</wsdl:message>
<wsdl:message name="ListBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:ListBiometricData"/>
</wsdl:message>
<wsdl:message name="ListBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:ListBiometricDataResponse"/>
</wsdl:message>

```

```

<wsdl:message name="PerformFusionRequestMessage">
    <wsdl:part name="parameters" element="tns:PerformFusion"/>
</wsdl:message>
<wsdl:message name="PerformFusionResponseMessage">
    <wsdl:part name="parameters" element="tns:PerformFusionResponse"/>
</wsdl:message>
<wsdl:message name="QueryCapabilitiesRequestMessage">
    <wsdl:part name="parameters" element="tns:QueryCapabilities"/>
</wsdl:message>
<wsdl:message name="QueryCapabilitiesResponseMessage">
    <wsdl:part name="parameters" element="tns:QueryCapabilitiesResponse"/>
</wsdl:message>
<wsdl:message name="RetrieveBiographicDataRequestMessage">
    <wsdl:part name="parameters" element="tns:RetrieveBiographicData"/>
</wsdl:message>
<wsdl:message name="RetrieveBiographicDataResponseMessage">
    <wsdl:part name="parameters" element="tns:RetrieveBiographicDataResponse"/>
</wsdl:message>
<wsdl:message name="RetrieveBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:RetrieveBiometricData"/>
</wsdl:message>
<wsdl:message name="RetrieveBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:RetrieveBiometricDataResponse"/>
</wsdl:message>
<wsdl:message name="SetBiographicDataRequestMessage">
    <wsdl:part name="parameters" element="tns:SetBiographicData"/>
</wsdl:message>
<wsdl:message name="SetBiographicDataResponseMessage">
    <wsdl:part name="parameters" element="tns:SetBiographicDataResponse"/>
</wsdl:message>
<wsdl:message name="SetBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:SetBiometricData"/>
</wsdl:message>
<wsdl:message name="SetBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:SetBiometricDataResponse"/>
</wsdl:message>

```

```

<wsdl:message name="TransformBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:TransformBiometricData"/>
</wsdl:message>
<wsdl:message name="TransformBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:TransformBiometricDataResponse"/>
</wsdl:message>
<wsdl:message name="UpdateBiographicDataRequestMessage">
    <wsdl:part name="parameters" element="tns:UpdateBiographicData"/>
</wsdl:message>
<wsdl:message name="UpdateBiographicDataResponseMessage">
    <wsdl:part name="parameters" element="tns:UpdateBiographicDataResponse"/>
</wsdl:message>
<wsdl:message name="UpdateBiometricDataRequestMessage">
    <wsdl:part name="parameters" element="tns:UpdateBiometricData"/>
</wsdl:message>
<wsdl:message name="UpdateBiometricDataResponseMessage">
    <wsdl:part name="parameters" element="tns:UpdateBiometricDataResponse"/>
</wsdl:message>
<wsdl:message name="VerifySubjectRequestMessage">
    <wsdl:part name="parameters" element="tns:VerifySubject"/>
</wsdl:message>
<wsdl:message name="VerifySubjectResponseMessage">
    <wsdl:part name="parameters" element="tns:VerifySubjectResponse"/>
</wsdl:message>
<wsdl:message name="EnrollRequestMessage">
    <wsdl:part name="parameters" element="tns:Enroll"/>
</wsdl:message>
<wsdl:message name="EnrollResponseMessage">
    <wsdl:part name="parameters" element="tns:EnrollResponse"/>
</wsdl:message>
<wsdl:message name="GetEnrollResultsRequestMessage">
    <wsdl:part name="parameters" element="tns:GetEnrollResults"/>
</wsdl:message>
<wsdl:message name="GetEnrollResultsResponseMessage">
    <wsdl:part name="parameters" element="tns:GetEnrollResultsResponse"/>

```

```

</wsdl:message>
<wsdl:message name="GetIdentifyResultsRequestMessage">
  <wsdl:part name="parameters" element="tns:GetIdentifyResults"/>
</wsdl:message>
<wsdl:message name="GetIdentifyResultsResponseMessage">
  <wsdl:part name="parameters" element="tns:GetIdentifyResultsResponse"/>
</wsdl:message>
<wsdl:message name="GetVerifyResultsRequestMessage">
  <wsdl:part name="parameters" element="tns:GetVerifyResults"/>
</wsdl:message>
<wsdl:message name="GetVerifyResultsResponseMessage">
  <wsdl:part name="parameters" element="tns:GetVerifyResultsResponse"/>
</wsdl:message>
<wsdl:message name="IdentifyRequestMessage">
  <wsdl:part name="parameters" element="tns:Identify"/>
</wsdl:message>
<wsdl:message name="IdentifyResponseMessage">
  <wsdl:part name="parameters" element="tns:IdentifyResponse"/>
</wsdl:message>
<wsdl:message name="RetrieveDataRequestMessage">
  <wsdl:part name="parameters" element="tns:RetrieveData"/>
</wsdl:message>
<wsdl:message name="RetrieveDataResponseMessage">
  <wsdl:part name="parameters" element="tns:RetrieveDataResponse"/>
</wsdl:message>
<wsdl:message name="VerifyRequestMessage">
  <wsdl:part name="parameters" element="tns:Verify"/>
</wsdl:message>
<wsdl:message name="VerifyResponseMessage">
  <wsdl:part name="parameters" element="tns:VerifyResponse"/>
</wsdl:message>
<wsdl:portType name="BIAS_v2">
  <wsdl:operation name="AddSubjectToGallery">
    <wsdl:input name="AddSubjectToGallery" message="tns:AddSubjectToGalleryRequestMessage"/>
    <wsdl:output name="AddSubjectToGalleryResponse" message="tns:AddSubjectToGalleryResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
  </wsdl:operation>

```



```

</wsdl:operation>
<wsdl:operation name="CheckQuality">
    <wsdl:input name="CheckQuality" message="tns:CheckQualityRequestMessage"/>
    <wsdl:output name="CheckQualityResponse" message="tns:CheckQualityResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="ClassifyBiometricData">
    <wsdl:input name="ClassifyBiometricData" message="tns:ClassifyBiometricDataRequestMessage"/>
    <wsdl:output name="ClassifyBiometricDataResponse" message="tns:ClassifyBiometricDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="CreateSubject">
    <wsdl:input name="CreateSubject" message="tns:CreateSubjectRequestMessage"/>
    <wsdl:output name="CreateSubjectResponse" message="tns:CreateSubjectResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="DeleteBiographicData">
    <wsdl:input name="DeleteBiographicData" message="tns:DeleteBiographicDataRequestMessage"/>
    <wsdl:output name="DeleteBiographicDataResponse" message="tns:DeleteBiographicDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="DeleteBiometricData">
    <wsdl:input name="DeleteBiometricData" message="tns:DeleteBiometricDataRequestMessage"/>
    <wsdl:output name="DeleteBiometricDataResponse" message="tns:DeleteBiometricDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="DeleteSubject">
    <wsdl:input name="DeleteSubject" message="tns:DeleteSubjectRequestMessage"/>
    <wsdl:output name="DeleteSubjectResponse" message="tns:DeleteSubjectResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="DeleteSubjectFromGallery">
    <wsdl:input name="DeleteSubjectFromGallery" message="tns:DeleteSubjectFromGalleryRequestMessage"/>
    <wsdl:output name="DeleteSubjectFromGalleryResponse" message="tns:DeleteSubjectFromGalleryResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>

```

```

<wsdl:operation name="GetIdentifySubjectResults">
    <wsdl:input name="GetIdentifySubjectResults" message="tns:GetIdentifySubjectResultsRequestMessage"/>
    <wsdl:output name="GetIdentifySubjectResultsResponse" message="tns:GetIdentifySubjectResultsResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="IdentifySubject">
    <wsdl:input name="IdentifySubject" message="tns:IdentifySubjectRequestMessage"/>
    <wsdl:output name="IdentifySubjectResponse" message="tns:IdentifySubjectResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="ListBiographicData">
    <wsdl:input name="ListBiographicData" message="tns:ListBiographicDataRequestMessage"/>
    <wsdl:output name="ListBiographicDataResponse" message="tns:ListBiographicDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="ListBiometricData">
    <wsdl:input name="ListBiometricData" message="tns:ListBiometricDataRequestMessage"/>
    <wsdl:output name="ListBiometricDataResponse" message="tns:ListBiometricDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="PerformFusion">
    <wsdl:input name="PerformFusion" message="tns:PerformFusionRequestMessage"/>
    <wsdl:output name="PerformFusionResponse" message="tns:PerformFusionResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="QueryCapabilities">
    <wsdl:input name="QueryCapabilities" message="tns:QueryCapabilitiesRequestMessage"/>
    <wsdl:output name="QueryCapabilitiesResponse" message="tns:QueryCapabilitiesResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="RetrieveBiographicData">
    <wsdl:input name="RetrieveBiographicData" message="tns:RetrieveBiographicDataRequestMessage"/>
    <wsdl:output name="RetrieveBiographicDataResponse" message="tns:RetrieveBiographicDataResponseMessage"/>
    <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
</wsdl:operation>
<wsdl:operation name="RetrieveBiometricData">

```

```

        <wsdl:input name="RetrieveBiometricData" message="tns:RetrieveBiometricDataRequestMessage"/>
        <wsdl:output name="RetrieveBiometricDataResponse" message="tns:RetrieveBiometricDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetBiographicData">
        <wsdl:input name="SetBiographicData" message="tns:SetBiographicDataRequestMessage"/>
        <wsdl:output name="SetBiographicDataResponse" message="tns:SetBiographicDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SetBiometricData">
        <wsdl:input name="SetBiometricData" message="tns:SetBiometricDataRequestMessage"/>
        <wsdl:output name="SetBiometricDataResponse" message="tns:SetBiometricDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="TransformBiometricData">
        <wsdl:input name="TransformBiometricData" message="tns:TransformBiometricDataRequestMessage"/>
        <wsdl:output name="TransformBiometricDataResponse" message="tns:TransformBiometricDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="UpdateBiographicData">
        <wsdl:input name="UpdateBiographicData" message="tns:UpdateBiographicDataRequestMessage"/>
        <wsdl:output name="UpdateBiographicDataResponse" message="tns:UpdateBiographicDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="UpdateBiometricData">
        <wsdl:input name="UpdateBiometricData" message="tns:UpdateBiometricDataRequestMessage"/>
        <wsdl:output name="UpdateBiometricDataResponse" message="tns:UpdateBiometricDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="VerifySubject">
        <wsdl:input name="VerifySubject" message="tns:VerifySubjectRequestMessage"/>
        <wsdl:output name="VerifySubjectResponse" message="tns:VerifySubjectResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Enroll">
        <wsdl:input name="Enroll" message="tns:EnrollRequestMessage"/>

```

```

        <wsdl:output name="EnrollResponse" message="tns:EnrollResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetEnrollResults">
        <wsdl:input name="GetEnrollResults" message="tns:GetEnrollResultsRequestMessage"/>
        <wsdl:output name="GetEnrollResultsResponse" message="tns:GetEnrollResultsResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetIdentifyResults">
        <wsdl:input name="GetIdentifyResults" message="tns:GetIdentifyResultsRequestMessage"/>
        <wsdl:output name="GetIdentifyResultsResponse" message="tns:GetIdentifyResultsResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetVerifyResults">
        <wsdl:input name="GetVerifyResults" message="tns:GetVerifyResultsRequestMessage"/>
        <wsdl:output name="GetVerifyResultsResponse" message="tns:GetVerifyResultsResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Identify">
        <wsdl:input name="Identify" message="tns:IdentifyRequestMessage"/>
        <wsdl:output name="IdentifyResponse" message="tns:IdentifyResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="RetrieveData">
        <wsdl:input name="RetrieveData" message="tns:RetrieveDataRequestMessage"/>
        <wsdl:output name="RetrieveDataResponse" message="tns:RetrieveDataResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Verify">
        <wsdl:input name="Verify" message="tns:VerifyRequestMessage"/>
        <wsdl:output name="VerifyResponse" message="tns:VerifyResponseMessage"/>
        <wsdl:fault name="BIASFault" message="tns:BIASFaultMessage"/>
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="BIAS_v2HttpBinding" type="tns:BIAS_v2">

```

```

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="AddSubjectToGallery">
  <soap:operation soapAction="AddSubjectToGallery"/>
  <wsdl:input name="AddSubjectToGallery">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="AddSubjectToGalleryResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="CheckQuality">
  <soap:operation soapAction="CheckQuality"/>
  <wsdl:input name="CheckQuality">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="CheckQualityResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="ClassifyBiometricData">
  <soap:operation soapAction="ClassifyBiometricData"/>
  <wsdl:input name="ClassifyBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="ClassifyBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="CreateSubject">
  <soap:operation soapAction="CreateSubject"/>
  <wsdl:input name="CreateSubject">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="CreateSubjectResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="DeleteBiographicData">
  <soap:operation soapAction="DeleteBiographicData"/>
  <wsdl:input name="DeleteBiographicData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="DeleteBiographicDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="DeleteBiometricData">
  <soap:operation soapAction="DeleteBiometricData"/>
  <wsdl:input name="DeleteBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="DeleteBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="DeleteSubject">
  <soap:operation soapAction="DeleteSubject"/>
  <wsdl:input name="DeleteSubject">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="DeleteSubjectResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="DeleteSubjectFromGallery">
  <soap:operation soapAction="DeleteSubjectFromGallery"/>
  <wsdl:input name="DeleteSubjectFromGallery">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="DeleteSubjectFromGalleryResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="GetIdentifySubjectResults">
  <soap:operation soapAction="GetIdentifySubjectResults"/>
  <wsdl:input name="GetIdentifySubjectResults">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="GetIdentifySubjectResultsResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="IdentifySubject">
  <soap:operation soapAction="IdentifySubject"/>
  <wsdl:input name="IdentifySubject">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="IdentifySubjectResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="ListBiographicData">
  <soap:operation soapAction="ListBiographicData"/>
  <wsdl:input name="ListBiographicData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="ListBiographicDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="ListBiometricData">
  <soap:operation soapAction="ListBiometricData"/>
  <wsdl:input name="ListBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="ListBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```



```

</wsdl:operation>
<wsdl:operation name="PerformFusion">
  <soap:operation soapAction="PerformFusion"/>
  <wsdl:input name="PerformFusion">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="PerformFusionResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="QueryCapabilities">
  <soap:operation soapAction="QueryCapabilities"/>
  <wsdl:input name="QueryCapabilities">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="QueryCapabilitiesResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="RetrieveBiographicData">
  <soap:operation soapAction="RetrieveBiographicData"/>
  <wsdl:input name="RetrieveBiographicData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="RetrieveBiographicDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="RetrieveBiometricData">
  <soap:operation soapAction="RetrieveBiometricData"/>
  <wsdl:input name="RetrieveBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="RetrieveBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="SetBiographicData">
  <soap:operation soapAction="SetBiographicData"/>
  <wsdl:input name="SetBiographicData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="SetBiographicDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="SetBiometricData">
  <soap:operation soapAction="SetBiometricData"/>
  <wsdl:input name="SetBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="SetBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="TransformBiometricData">
  <soap:operation soapAction="TransformBiometricData"/>
  <wsdl:input name="TransformBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="TransformBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="UpdateBiographicData">
  <soap:operation soapAction="UpdateBiographicData"/>
  <wsdl:input name="UpdateBiographicData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="UpdateBiographicDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="UpdateBiometricData">
  <soap:operation soapAction="UpdateBiometricData"/>
  <wsdl:input name="UpdateBiometricData">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="UpdateBiometricDataResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>

```

```

</wsdl:operation>
<wsdl:operation name="VerifySubject">
  <soap:operation soapAction="VerifySubject"/>
  <wsdl:input name="VerifySubject">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="VerifySubjectResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="Enroll">
  <soap:operation soapAction="Enroll"/>
  <wsdl:input name="Enroll">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="EnrollResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="GetEnrollResults">
  <soap:operation soapAction="GetEnrollResults"/>
  <wsdl:input name="GetEnrollResults">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="GetEnrollResultsResponse">
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="BIASFault">
    <soap:fault name="BIASFault" use="literal"/>
  </wsdl:fault>
</wsdl:operation>

```

```

        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="GetIdentifyResults">
        <soap:operation soapAction="GetIdentifyResults"/>
        <wsdl:input name="GetIdentifyResults">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="GetIdentifyResultsResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="BIASFault">
            <soap:fault name="BIASFault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="GetVerifyResults">
        <soap:operation soapAction="GetVerifyResults"/>
        <wsdl:input name="GetVerifyResults">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="GetVerifyResultsResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="BIASFault">
            <soap:fault name="BIASFault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="Identify">
        <soap:operation soapAction="Identify"/>
        <wsdl:input name="Identify">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="IdentifyResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="BIASFault">
            <soap:fault name="BIASFault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>

```

```

        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="RetrieveData">
        <soap:operation soapAction="RetrieveData"/>
        <wsdl:input name="RetrieveData">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="RetrieveDataResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="BIASFault">
            <soap:fault name="BIASFault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="Verify">
        <soap:operation soapAction="Verify"/>
        <wsdl:input name="Verify">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="VerifyResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="BIASFault">
            <soap:fault name="BIASFault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
</wsdl:binding>
<wsdl:service name="BIAS_v2">
    <wsdl:port name="BIAS_v2HttpBindingPort" binding="tns:BIAS_v2HttpBinding">
        <soap:address location="http://your_URI/IRI_to_your_BIAS_implementation"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

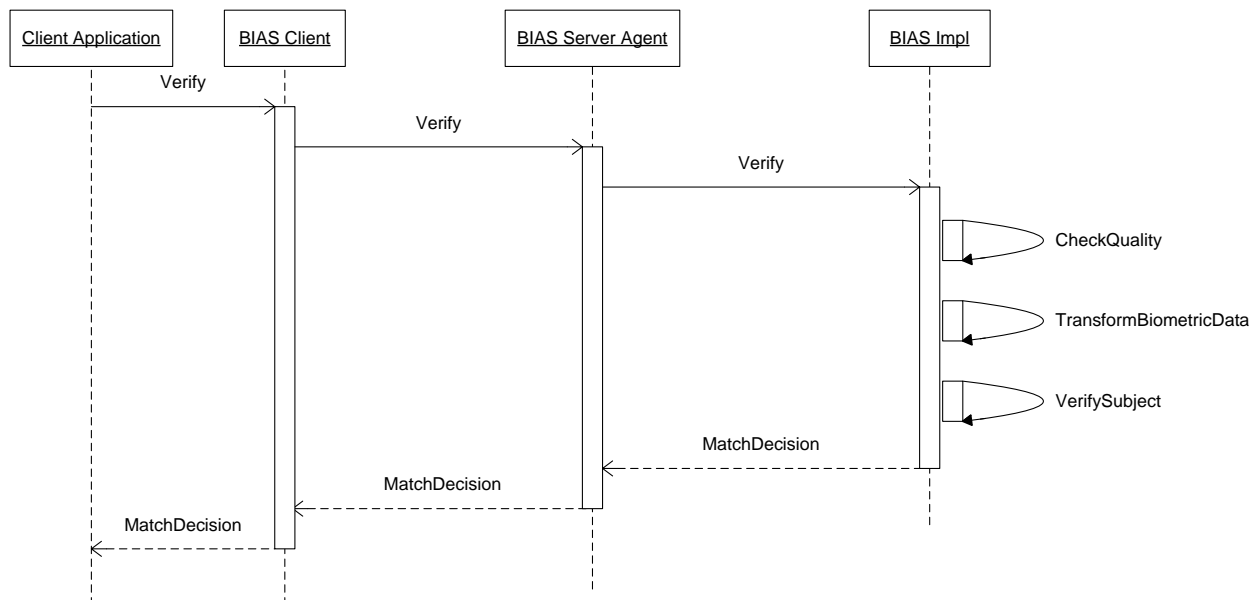
Appendix B. Use Cases (non-normative)

The intent of this annex is to provide operational sequence diagrams / flow charts that show how the higher level usage scenarios within [ISO/IEC-BIAS] could be implemented using the BIAS SOAP profile. The following use cases are given:

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

B.1 Verification Use Case

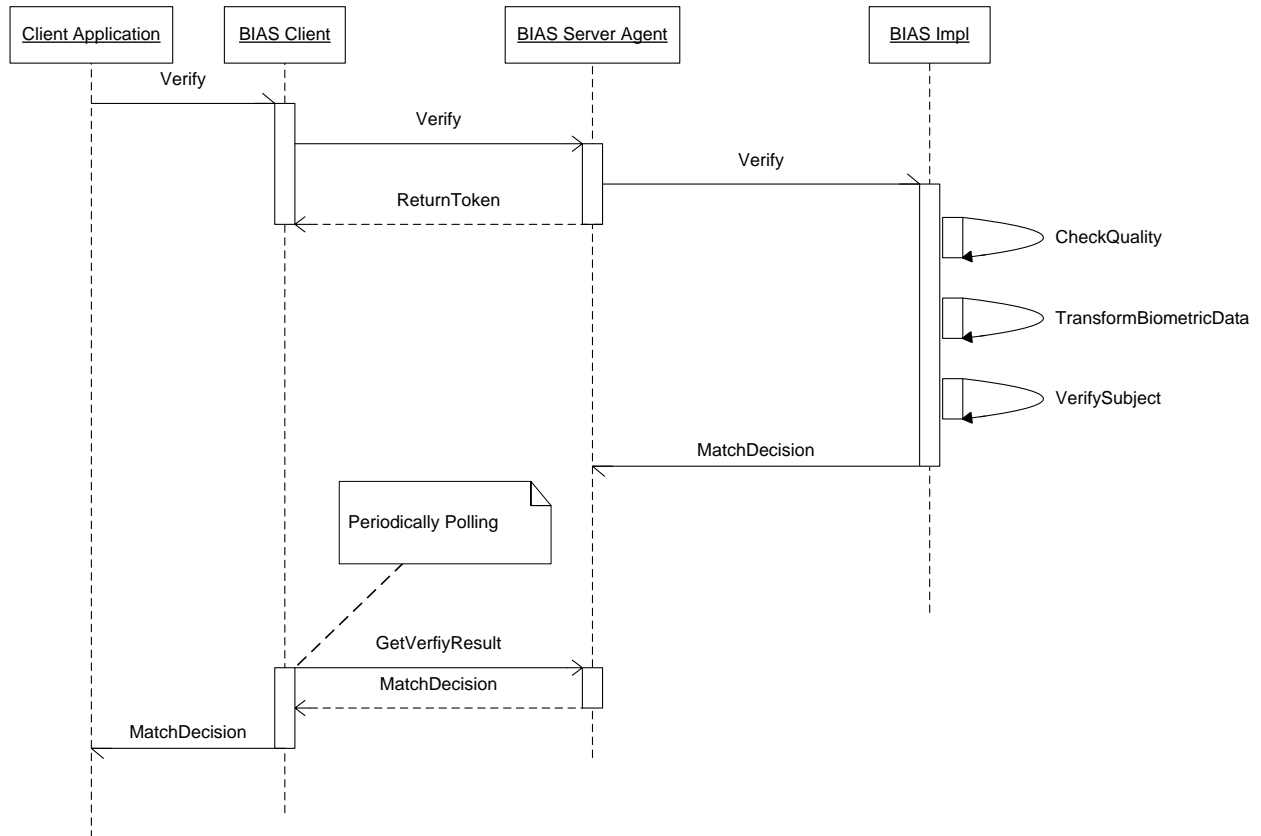
This use case uses the aggregate Verify operation in which a single request results in some set of operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service provider.



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

B.2 Asynchronous Verification Use Case

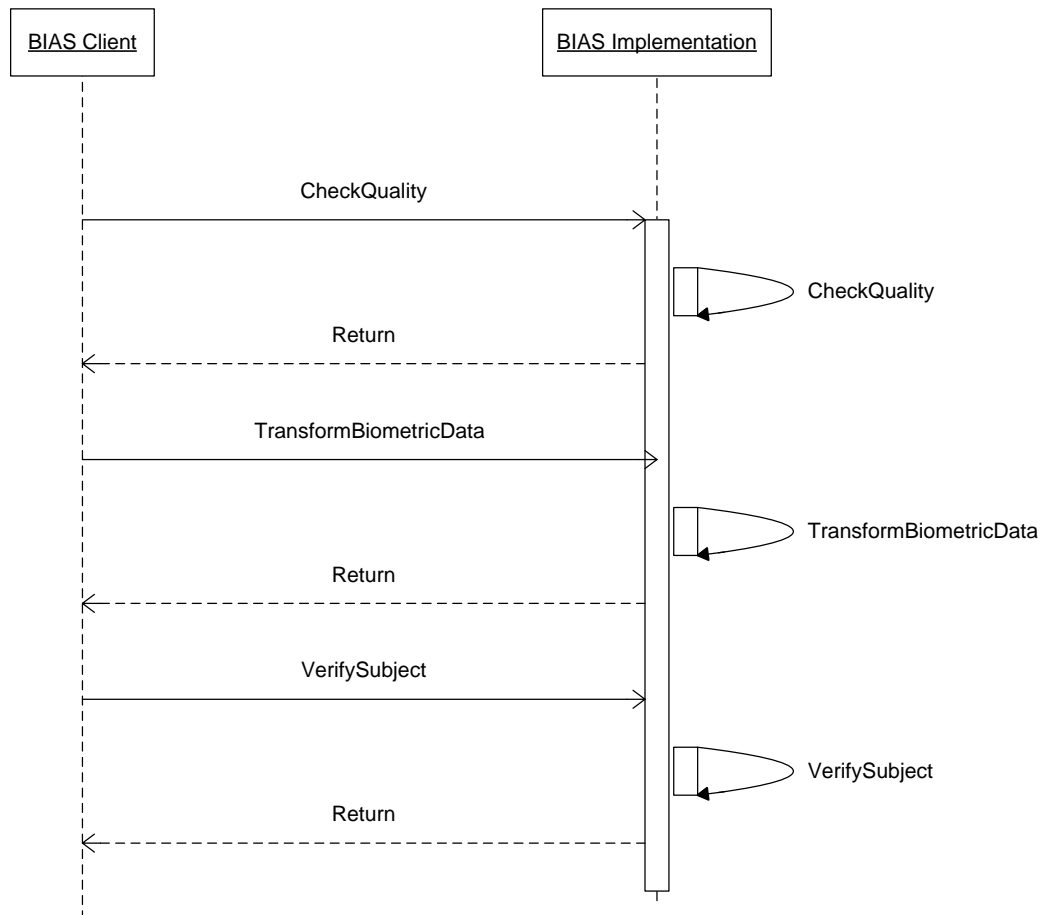
In this use case, the requester issues two requests – the BIAS Verify request to initiate the operation followed by a BIAS GetVerifyResult request to retrieve the results of that operation.



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

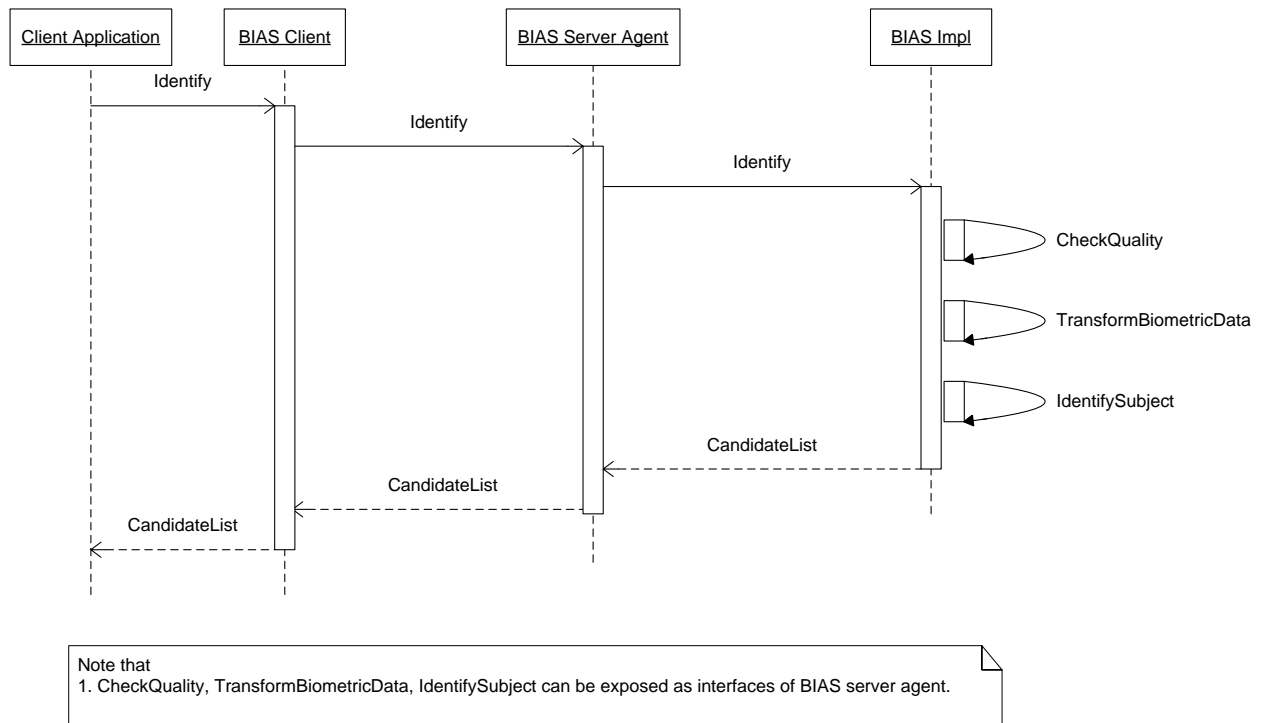
B.3 Primitive Verification Use Case

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.



B.4 Identification Use Case

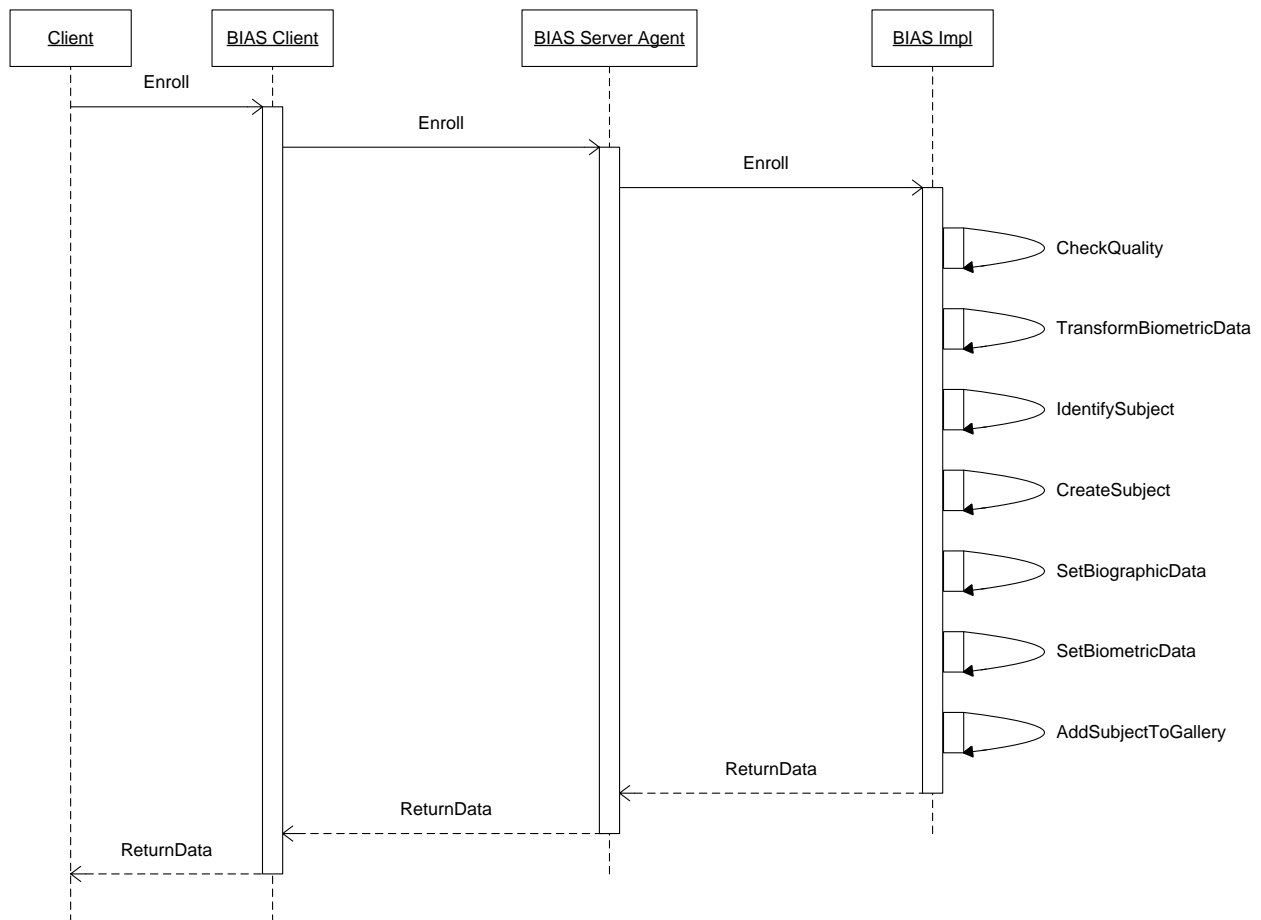
This use case uses the aggregate Identify operation in which a single request results in some set of operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service provider.



B.5 Biometric Enrollment Use Case

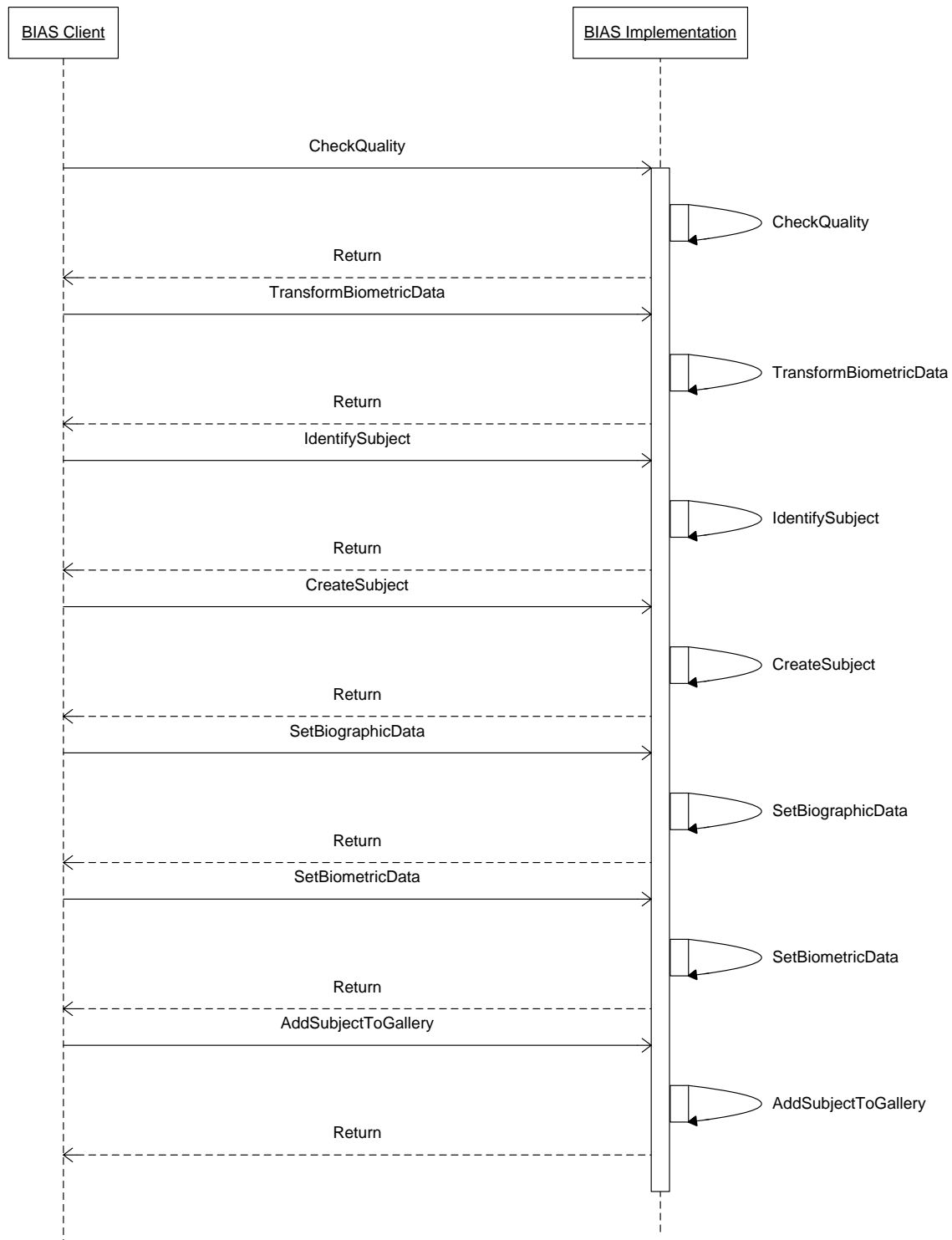
This use case uses the aggregate Enroll operation in which a single request results in some set of operations (in this case, a series of primitive BIAS operations) being performed by the BIAS service provider.

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.).



B.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 C. Samples (non-normative)

C.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>

</soap:Envelope>

Create Subject Request with Optional OASIS BIAS Content:

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:GenericRequestParameters>
 <tns:Application>BIAS Application</tns:Application>
 <tns:ApplicationUser>BIAS User</tns:ApplicationUser>
 </tns:GenericRequestParameters>
 <tns:Identity>
 <tns:SubjectID>123456789</tns:SubjectID>
 </tns:Identity>
 </tns:CreateSubject>
 </soap:Body>
</soap:Envelope>

Simple Create Subject Response:

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
 <soap:Body>
 <tns:CreateSubjectResponse xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
 <tns:ResponseStatus>
 <tns:Return>0</tns:Return>
 </tns:ResponseStatus>
 <tns:Identity>
 <tns:SubjectID>123456789</tns:SubjectID>
 </tns:Identity>
 </tns:CreateSubjectResponse>
 </soap:Body>
</soap:Envelope>

Create Subject Response with Optional OASIS BIAS Content:

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
 <soap:Body>
 <tns:CreateSubjectResponse xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">

```

        <tns:ResponseStatus>
            <tns:Return>0</tns:Return>
            <tns:Message>Subject ID 123456789 successfully
created.</tns:Message>
        </tns:ResponseStatus>
        <tns:Identity>
            <tns:SubjectID>123456789</tns:SubjectID>
        </tns:Identity>
    </tns:CreateSubjectResponse>
</soap:Body>
</soap:Envelope>

```

C.2 Set Biographic Data Request/Response Example

ISO/IEC BIAS Specification

```

<interface name="SetBiographicData">
    <parameter name="SubjectID" type="xs:string" direction="in" />
    <parameter name="IdentityModel" type="IdentityModelType" direction="in" />
    <parameter name="EncounterID" type="xs:string" direction="inout" use="optional" />
    <parameter name="BiographicData" type="BiographicDataType" direction="in" />
    <parameter name="Return" type="xs:unsignedLong" direction="out" />
</interface>

```

OASIS BIAS Examples

Set Biographic Data Request:

POST /bias HTTP/1.1

Host: www.acme.com

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

Content-Length: nnnn

SOAPAction: "SetBiographicData"

<?xml version="1.0"?>

```

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
    <soap:Body>
        <tns:SetBiographicData
            xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
            <tns:Identity>
                <tns:SubjectID>123456789</tns:SubjectID>
                <tns:BiographicData>
                    <tns:BiographicDataItem>
                        <tns:Name>Last</tns:Name>
                        <tns:Type>string</tns:Type>
                        <tns:Value>Doe</tns:Value>
                    </tns:BiographicDataItem>
                </tns:BiographicData>
            </tns:SetBiographicData>
        </tns:Body>
    </soap:Envelope>

```

```

        </tns:Identity>
        <tns:IdentityModel>person</tns:IdentityModel>
    </tns:SetBiographicData>
</soap:Body>
</soap:Envelope>
Set Biographic Data Response:
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
    <soap:Body>
        <tns:SetBiographicDataResponse
            xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
            <tns:ResponseStatus>
                <tns:Return>0</tns:Return>
            </tns:ResponseStatus>
        </tns:SetBiographicDataResponse>
    </soap:Body>
</soap:Envelope>

```

C.3 Set Biometric Data Request/Response Example

ISO/IEC BIAS Specification

```

<interface name="SetBiometricData">
    <parameter name="SubjectID" type="xs:string" direction="in" />
    <parameter name="IdentityModel" type="IdentityModelType" direction="in" />
    <parameter name="EncounterID" type="xs:string" direction="inout" use="optional" />
    <parameter name="BIRList" type="CBEFF_BIR_ListType" direction="in" />
    <parameter name="Return" type="xs:unsignedLong" direction="out" />
</interface>

```

OASIS BIAS Examples

Set Biometric Data Request:

```

POST /bias HTTP/1.1
Host: www.acme.com
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "SetBiometricData"
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
    <soap:Body>
        <tns:SetBiometricData

```



```

        xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
      <tns:Identity>
        <tns:SubjectID>123456789</tns:SubjectID>
        <tns:BiometricData>
          <tns:BIRList>
            <tns:BIR>biometric data</tns:BIR>
          </tns:BIRList>
        </tns:BiometricData>
      </tns:Identity>
      <tns:IdentityModel>person</tns:IdentityModel>
    </tns:SetBiometricData>
  </soap:Body>
</soap:Envelope>
Set Biometric Data Response:
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <tns:SetBiometricDataResponse
      xmlns:tns="http://docs.oasis-open.org/bias/bias-1.0/">
      <tns:ResponseStatus>
        <tns:Return>0</tns:Return>
      </tns:ResponseStatus>
    </tns:SetBiometricDataResponse>
  </soap:Body>
</soap:Envelope>

```

Appendix D. 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 E. Revision History

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