[](https://www.oasis-open.org/)

Biometric Identity Assurance Services (BIAS) SOAP Profile Version 2.0

Committee Specification Draft 01

21 April 2016

Specification URIs

This version:

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

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

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

Previous version:

N/A

Latest version:

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

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.html>

<http://docs.oasis-open.org/bioserv/BIAS/v2.0/BIAS-v2.0.docx>

Technical Committee:

[OASIS Biometric Services (BIOSERV) TC](https://www.oasis-open.org/committees/bioserv/)

Chair:

Kevin Mangold ([kevin.mangold@nist.gov](mailto:kevin.mangold@nist.gov)), [NIST](http://www.nist.gov/)

Editors:

Kevin Mangold ([kevin.mangold@nist.gov](mailto:kevin.mangold@nist.gov)), [NIST](http://www.nist.gov/)

Karen Marshall ([karen.marshall@nist.gov](mailto:karen.marshall@nist.gov)), [NIST](http://www.nist.gov/)

Additional artifacts:

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

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

Related work:

This specification is related to:

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

Declared XML namespaces:

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

Abstract:

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

Status:

This document was last revised or approved by the OASIS Biometric Services (BIOSERV) TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at <https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=bioserv#technical>.

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

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC’s web page (<https://www.oasis-open.org/committees/bioserv/ipr.php>).

Citation format:

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

[BIASPROFILE]

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

Notices

Copyright © OASIS Open 2016. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full [Policy](https://www.oasis-open.org/policies-guidelines/ipr) 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](https://www.oasis-open.org/), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

Table of Contents

[1 Introduction 9](#_Toc450646970)

[1.1 Purpose/Scope 9](#_Toc450646971)

[1.2 Overview 9](#_Toc450646972)

[1.3 Background 9](#_Toc450646973)

[1.4 Relationship to Other Standards 10](#_Toc450646974)

[1.5 Terminology 10](#_Toc450646975)

[1.6 References 11](#_Toc450646976)

[1.6.1 Normative References 11](#_Toc450646977)

[1.6.2 Non-Normative References 12](#_Toc450646978)

[2 Design Concepts and Architecture (non-normative) 14](#_Toc450646979)

[2.1 Philosophy 14](#_Toc450646980)

[2.2 Context 14](#_Toc450646981)

[2.3 Architecture 14](#_Toc450646982)

[3 Data dictionary 17](#_Toc450646983)

[3.1 Documentation Conventions 17](#_Toc450646984)

[3.2 Common Elements 18](#_Toc450646985)

[3.2.1 ApplicationIdentifier 18](#_Toc450646986)

[3.2.2 ApplicationUserIdentifier 18](#_Toc450646987)

[3.2.3 BaseBIRType 18](#_Toc450646988)

[3.2.4 BIASBiometricDataType 18](#_Toc450646989)

[3.2.5 BIASFaultCode 18](#_Toc450646990)

[3.2.6 BIASFaultDetail 20](#_Toc450646991)

[3.2.7 BIASIdentity 21](#_Toc450646992)

[3.2.8 BIASIDType 21](#_Toc450646993)

[3.2.9 BinaryBIR 21](#_Toc450646994)

[3.2.10 BiographicDataItemType 22](#_Toc450646995)

[3.2.11 BiographicDataItemListType 22](#_Toc450646996)

[3.2.12 BiographicDataListType 22](#_Toc450646997)

[3.2.13 BiographicDataSetType 22](#_Toc450646998)

[3.2.14 BiographicDataType 23](#_Toc450646999)

[3.2.15 BiometricDataType 24](#_Toc450647000)

[3.2.16 BiometricDataListType 24](#_Toc450647001)

[3.2.17 CandidateListResultType 24](#_Toc450647002)

[3.2.18 CandidateListType 25](#_Toc450647003)

[3.2.19 CandidateType 25](#_Toc450647004)

[3.2.20 CapabilityListType 25](#_Toc450647005)

[3.2.21 CapabilityName 25](#_Toc450647006)

[3.2.22 CapabilityType 31](#_Toc450647007)

[3.2.23 CBEFF\_BIR\_ListType 31](#_Toc450647008)

[3.2.24 CBEFF\_BIR\_Type 31](#_Toc450647009)

[3.2.25 Classification 32](#_Toc450647010)

[3.2.26 ClassificationAlgorithmType 32](#_Toc450647011)

[3.2.27 ClassificationData 33](#_Toc450647012)

[3.2.28 DocumentDataType 33](#_Toc450647013)

[3.2.29 DocumentDataListType 35](#_Toc450647014)

[3.2.30 EncounterCategoryType 35](#_Toc450647015)

[3.2.31 EncounterListType 35](#_Toc450647016)

[3.2.32 FusionDecision 35](#_Toc450647017)

[3.2.33 FusionIdentityListType 36](#_Toc450647018)

[3.2.34 FusionInformationListType 36](#_Toc450647019)

[3.2.35 FusionInformationType 36](#_Toc450647020)

[3.2.36 FusionResult 36](#_Toc450647021)

[3.2.37 FusionScore 36](#_Toc450647022)

[3.2.38 GenericRequestParameters 37](#_Toc450647023)

[3.2.39 IdentifySubjectResultType 37](#_Toc450647024)

[3.2.40 InformationType 37](#_Toc450647025)

[3.2.41 ListFilterType 37](#_Toc450647026)

[3.2.42 MatchType 38](#_Toc450647027)

[3.2.43 OptionType 38](#_Toc450647028)

[3.2.44 ProcessingOptionsType 38](#_Toc450647029)

[3.2.45 ProductID 38](#_Toc450647030)

[3.2.46 QualityData 38](#_Toc450647031)

[3.2.47 ResponseStatus 39](#_Toc450647032)

[3.2.48 ReturnCode 39](#_Toc450647033)

[3.2.49 ScoreType 39](#_Toc450647034)

[3.2.50 TokenResultType 40](#_Toc450647035)

[3.2.51 TokenType 40](#_Toc450647036)

[3.2.52 URI\_BIR 40](#_Toc450647037)

[3.2.53 VendorIdentifier 40](#_Toc450647038)

[3.2.54 Version 40](#_Toc450647039)

[3.2.55 VersionType 41](#_Toc450647040)

[3.2.56 XML\_BIR 41](#_Toc450647041)

[4 BIAS Messages 42](#_Toc450647042)

[4.1 Primitive Operations 42](#_Toc450647043)

[4.1.1 AddSubjectToGallery 42](#_Toc450647044)

[4.1.2 CheckQuality 43](#_Toc450647045)

[4.1.3 ClassifyBiometricData 45](#_Toc450647046)

[4.1.4 CreateEncounter 47](#_Toc450647047)

[4.1.5 CreateSubject 49](#_Toc450647048)

[4.1.6 DeleteBiographicData 50](#_Toc450647049)

[4.1.7 DeleteBiometricData 51](#_Toc450647050)

[4.1.8 DeleteDocumentData 53](#_Toc450647051)

[4.1.9 DeleteEncounter 54](#_Toc450647052)

[4.1.10 DeleteSubject 56](#_Toc450647053)

[4.1.11 DeleteSubjectFromGallery 57](#_Toc450647054)

[4.1.12 GetIdentifySubjectResults 58](#_Toc450647055)

[4.1.13 IdentifySubject 60](#_Toc450647056)

[4.1.14 ListBiographicData 63](#_Toc450647057)

[4.1.15 ListBiometricData 65](#_Toc450647058)

[4.1.16 ListDocumentData 70](#_Toc450647059)

[4.1.17 PerformFusion 72](#_Toc450647060)

[4.1.18 QueryCapabilities 74](#_Toc450647061)

[4.1.19 RetrieveBiographicData 76](#_Toc450647062)

[4.1.20 RetrieveBiometricData 78](#_Toc450647063)

[4.1.21 RetrieveDocumentData 80](#_Toc450647064)

[4.1.22 SetBiographicData 81](#_Toc450647065)

[4.1.23 SetBiometricData 83](#_Toc450647066)

[4.1.24 SetDocumentData 85](#_Toc450647067)

[4.1.25 TransformBiometricData 87](#_Toc450647068)

[4.1.26 UpdateBiographicData 88](#_Toc450647069)

[4.1.27 UpdateBiometricData 90](#_Toc450647070)

[4.1.28 UpdateDocumentData 91](#_Toc450647071)

[4.1.29 VerifySubject 93](#_Toc450647072)

[4.2 Aggregate Operations 95](#_Toc450647073)

[4.2.1 Delete 95](#_Toc450647074)

[4.2.2 Enroll 97](#_Toc450647075)

[4.2.3 GetDeletionResults 99](#_Toc450647076)

[4.2.4 GetEnrollResults 101](#_Toc450647077)

[4.2.5 GetIdentifyResults 102](#_Toc450647078)

[4.2.6 GetUpdateResults 104](#_Toc450647079)

[4.2.7 GetVerifyResults 105](#_Toc450647080)

[4.2.8 Identify 107](#_Toc450647081)

[4.2.9 RetrieveData 109](#_Toc450647082)

[4.2.10 Update 111](#_Toc450647083)

[4.2.11 Verify 113](#_Toc450647084)

[5 Message structure and rules 116](#_Toc450647085)

[5.1 Purpose and constraints 116](#_Toc450647086)

[5.2 Message requirements 117](#_Toc450647087)

[5.3 Handling binary data 118](#_Toc450647088)

[5.3.1 Base64 encoding 118](#_Toc450647089)

[5.3.2 Use of XOP 119](#_Toc450647090)

[5.4 Discovery 119](#_Toc450647091)

[5.5 Identifying operations 119](#_Toc450647092)

[5.5.1 Operation name element 119](#_Toc450647093)

[5.5.2 WS-Addressing Action 120](#_Toc450647094)

[5.6 Security 121](#_Toc450647095)

[5.6.1 Use of SSL 3.0 or TLS 1.0 121](#_Toc450647096)

[5.6.2 Data Origin Authentication 121](#_Toc450647097)

[5.6.3 Message Integrity 121](#_Toc450647098)

[5.6.4 Message Confidentiality 121](#_Toc450647099)

[5.6.5 CBEFF BIR security features 122](#_Toc450647100)

[5.6.6 Security Considerations 122](#_Toc450647101)

[5.6.7 Security of Stored Data 122](#_Toc450647102)

[5.6.8 Key Management 122](#_Toc450647103)

[5.7 Use with other WS\* standards 122](#_Toc450647104)

[5.8 Tailoring 122](#_Toc450647105)

[6 Error handling 124](#_Toc450647106)

[6.1 BIAS operation return codes 124](#_Toc450647107)

[6.2 SOAP fault codes 124](#_Toc450647108)

[7 Conformance 125](#_Toc450647109)

[Appendix A. XML Schema 126](#_Toc450647110)

[Appendix B. BIAS Patron format specification 249](#_Toc450647111)

[B.1 Patron 249](#_Toc450647112)

[B.2 Patron identifier 249](#_Toc450647113)

[B.3 Patron format name 249](#_Toc450647114)

[B.4 Patron format identifier 249](#_Toc450647115)

[B.5 ASN.1 object identifier for this patron format 249](#_Toc450647116)

[B.6 Domain of use 249](#_Toc450647117)

[B.7 Version identifier 249](#_Toc450647118)

[B.8 CBEFF version 249](#_Toc450647119)

[B.9 General 250](#_Toc450647120)

[B.10 Specification 250](#_Toc450647121)

[B.11 Element <BIR> 251](#_Toc450647122)

[B.11.1 Syntax 251](#_Toc450647123)

[B.11.2 Semantics 251](#_Toc450647124)

[B.12 Element <Version> 252](#_Toc450647125)

[B.12.1 Syntax 252](#_Toc450647126)

[B.12.2 Semantics 252](#_Toc450647127)

[B.13 Element <CBEFFVersion> 253](#_Toc450647128)

[B.13.1 Syntax 253](#_Toc450647129)

[B.13.2 Semantics 253](#_Toc450647130)

[B.14 Element <BIRInfo> 253](#_Toc450647131)

[B.14.1 Syntax 253](#_Toc450647132)

[B.14.2 Semantics 254](#_Toc450647133)

[B.15 Element <BDBInfo> 255](#_Toc450647134)

[B.15.1 Syntax 255](#_Toc450647135)

[B.15.2 Semantics 257](#_Toc450647136)

[B.16 Element <SBInfo> 261](#_Toc450647137)

[B.16.1 Syntax 261](#_Toc450647138)

[B.16.2 Semantics 262](#_Toc450647139)

[B.17 Representation of Integers 262](#_Toc450647140)

[B.18 Representation of Octet Strings 262](#_Toc450647141)

[B.19 Representation of Date and Time of the Day 263](#_Toc450647142)

[B.20 Representation of Universally Unique Identifiers 264](#_Toc450647143)

[B.21 Patron format conformance statement 264](#_Toc450647144)

[B.21.1 Identifying information 264](#_Toc450647145)

[B.21.2 ISO/IEC 19785-1:2006/Amd 1:2010 to Patron Format Mapping 264](#_Toc450647146)

[B.22 XML schema of the BIAS patron format 266](#_Toc450647147)

[B.23 Sample BIR encoding 269](#_Toc450647148)

[Appendix C. Use Cases (non-normative) 271](#_Toc450647149)

[C.1 Verification Use Case 271](#_Toc450647150)

[C.2 Asynchronous Verification Use Case 272](#_Toc450647151)

[C.3 Primitive Verification Use Case 273](#_Toc450647152)

[C.4 Identification Use Case 274](#_Toc450647153)

[C.5 Biometric Enrollment Use Case 275](#_Toc450647154)

[C.6 Primitive Enrollment Use Case 276](#_Toc450647155)

[Appendix D. Samples (non-normative) 277](#_Toc450647156)

[D.1 Create Subject Request/Response Example 277](#_Toc450647157)

[D.2 Set Biographic Data Request/Response Example 279](#_Toc450647158)

[D.3 Set Biometric Data Request/Response Example 280](#_Toc450647159)

[Appendix E. Acknowledgements 283](#_Toc450647160)

[Appendix F. Revision History 284](#_Toc450647161)

# Introduction

**1.1 Purpose/Scope**

This Organization for the Advancement of Structured Information Standards (OASIS) Biometric Identity Assurance Services (BIAS) profile specifies how to use the eXtensible Markup Language (XML) [[XML10](#XML10)] defined in ISO/IEC 30108-1:2015, Information technology — Biometric Identity Assurance Services [[ISO/IEC-BIAS](#ISOIECBIAS)] to invoke Simple Object Access Protocol (SOAP) -based services that implement BIAS operations. These SOAP-based services enable an application to invoke biometric identity assurance operations remotely in a Services Oriented Architecture (SOA) infrastructure.

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

**1.2 Overview**

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

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

**1.3 Background**

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

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

**1.4 Relationship to Other Standards**

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

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

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

**1.5 Terminology**

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

The following additional terms and definitions are used:

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

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

**BIAS**

Biometric Identity Assurance Services

**BIR**

Biometric Information Record

**ESB**

Enterprise Service Bus

**HTTP**

HyperText Transfer Protocol

**HTTPS**

HyperText Transfer Protocol over SSL or HTTP Secure

**IRI**

Internationalized Resource Identifier

**SOA**

Service-Oriented Architecture

**SOAP**

Simple Object Access Protocol

**SSL**

Secure Sockets Layer

**TLS**

Transport Layer Security

**UDDI**

Universal Description, Discovery, and Integration

**URI**

Uniform Resource Identifier

**VPN**

Virtual Private Network

**WSDL**

Web Services Description Language

**WSS**

Web Services Security

**XML**

eXtensible Markup Language

**CBEFF**

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

**BIAS implementation**

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

**BIAS endpoint**

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

**BIAS message**

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

**BIAS request message**

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

**BIAS response message**

BIAS message conveying a response to a prior BIAS requestmessage

**1.6 References**

**1.6.1 Normative References**

**[RFC2119]** S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, IETF RFC 2119, March 1997  
<http://www.ietf.org/rfc/rfc2119.txt>

**[CBEFF]** ISO/IEC19785-1:2006, Information technology – Common Biometric Exchange Formats Framework – Part 1: Data element specification, with Amendment 1:2010  
<http://www.iso.org>

**[DATE-TIME]** ISO 8601:2004, Data elements and interchange formats — Information interchange — Representation of dates and times  
<http://www.iso.org>

**[INCITS-BIAS]** ANSI INCITS 442-2010, Biometric Identity Assurance Services (BIAS), July 2010  
<http://www.incits.org>

**[IRI]** M. Duerst, et al, Internationalized Resouce Identifiers, RFC3987, January 2005  
<http://www.ietf.org/rfc/rfc3987.txt>

**[****ISO/IEC-BIAS]** ISO/IEC 30108-1:2015, Information technology — Biometric Identity Assurance Services — Part 1: BIAS Services   
<http://www.iso.org>

**[SOAP11]** Simple Object Access Protocol (SOAP) 1.1, 8 May 2000  
<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

**[URI]** T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifiers (URI): Generic Syntax*, RFC 3986, MIT/LCS, U.C. Irvine, Xerox Corporation, January 2005.  
<http://ietf.org/rfc/rfc3986>

**[UTF-8]** ISO/IEC 10646:2003, Information technology — Universal Multiple-Octet Coded Character Set (UCS)  
<http://www.iso.org>

**[WS-Addr]** W3C Recommendation,Web Services Addressing 1.0 - Core, and Web Services Addressing 1.0 - SOAP Binding, 9 May 2006  
<http://www.w3.org/2002/ws/addr/>

**[WS-I-Basic]** Basic Profile Version 1.1, 10 April 2006  
<http://www.ws-i.org/Profiles/BasicProfile-1.1-2006-04-10.html>

**[WS-I-Bind]** Web Services-Interoperability Organization (WS-I) Simple SOAP Binding Profile Version 1.0, 24 August 2004  
<http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0-2004-08-24.html>

**[WSDL11]** Web Services Description Language (WSDL) 1.1, 15 March 2001  
<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

**[XML 10]** Extensible Markup Language (XML) 1.0, 16 August 2006  
<http://www.w3.org/TR/2006/REC-xml-20060816/>

**[XOP]** XML-binary Optimized Packaging, W3C Recommendation, 25 January 2005  
<http://www.w3.org/TR/2005/REC-xop10-20050125/>

**1.6.2 Non-Normative References**

**[BioAPI]** ISO/IEC 19784-1:2006, Information technology – Biometric Application Programming Interface – Part 1: BioAPI Specification  
<http://www.iso.org>

**[CBEFF-3]** ISO/IEC19785-3:2007, Information technology – Common Biometric Exchange Formats Framework – Part 3: Patron format specifications, with Amendment 1:2010  
<http://www.iso.org>

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

**[EBTS-DOD]** Department of DefenseElectronic Biometric TransmissionSpecification, Version 2.0, 27 March 2009  
<http://www.biometrics.dod.mil/CurrentInitiatives/Standards/dodebts.aspx>

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

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

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

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

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

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

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

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

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

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

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

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

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

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

# Design Concepts and Architecture (non-normative)

**2.1 Philosophy**

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

**2.2 Context**

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

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



**Figure 1. BIAS Context**

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

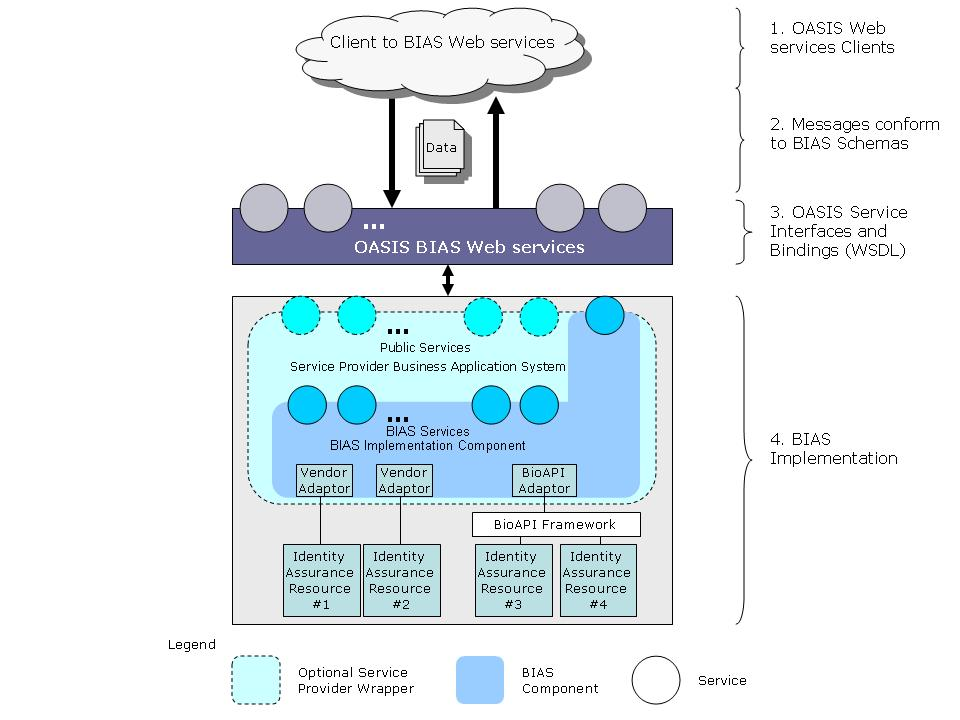
**2.3 Architecture**

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

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

* A Client request to the BIAS Web services may be triggered by a human interaction OR any proxy system such as an ESB.
* Client sends and receives SOAP messages that conform to the BIAS schemas
* Calls to the BIAS Implementation use OASIS Service Interfaces and Bindings (via WSDL)
* The BIAS implementation maps the service call to the appropriate internal API or set of APIs and returns data according to the service interface.

Note that services are represented as circles.



**Figure 2. Representative BIAS Architecture**

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

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

***BIAS implementation***

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

***BIAS messages***

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

***BIAS endpoints***

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

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

# Data dictionary

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

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

* 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  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](#UTF8)] (see ISO/IEC 10646:2003, Annex D).

**3.2 Common Elements**

### 3.2.1 ApplicationIdentifier

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

### 3.2.2 ApplicationUserIdentifier

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

### 3.2.3 BaseBIRType

|  |  |
| --- | --- |
| **Type:** | Schema complexType |
| **Description:** | Base type for all BIR subtypes; see [BinaryBIR](#_BinaryBIR), [URI\_BIR](#_URI_BIR), and [XML\_BIR](#_XML_BIR) for currently available types. |

### 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](#_CBEFF_BIR_ListType) | 0..1 | N | A list of CBEFF-BIR elements. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..1 | N | Contains biometric information in either a non-XML or an XML representation. |
| InputBIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..1 | N | Maps to specific ISO/IEC BIAS elements as required by that specification. |
| ReferenceBIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..1 | N | Maps to specific ISO/IEC BIAS elements as required by that specification. |
| BiometricDataList | [BiometricDataListType](#_BiometricDataListType) | 0..1 | N | A list of biometric data elements. |

### 3.2.5 BIASFaultCode

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

**BIASFaultCode Enumeration Values**

| **Value** | **Description** |
| --- | --- |
| UNKNOWN\_ERROR | The service failed for an unknown reason. |
| UNSUPPORTED\_CAPABILITY | A requested capability is not supported by the service implementation. |
| INVALID\_INPUT | The data in a service input parameter is invalid. |
| BIR\_QUALITY\_ERROR | Biometric sample quality is too poor for the service to succeed. |
| INVALID\_BIR | The input BIR is empty or in an invalid or unrecognized format. |
| BIR\_SIGNATURE\_FAILURE | The service could not validate the signature, if used, on the input BIR. |
| BIR\_DECRYPTION\_FAILURE | The service could not decrypt an encrypted input BIR. |
| INVALID\_ENCOUNTER\_ID | The input encounter ID is empty or in an invalid format. |
| INVALID\_SUBJECT\_ID | The input subject ID is empty or in an invalid format. |
| UNKNOWN\_SUBJECT | The subject referenced by the input subject ID does not exist. |
| UNKNOWN\_GALLERY | The gallery referenced by the input gallery ID does not exist. |
| UNKNOWN\_ENCOUNTER | The encounter referenced by the input encounter ID does not exist. |
| UNKNOWN\_BIOGRAPHIC\_FORMAT | The biographic data format is not known or not supported. |
| UNKNOWN\_IDENTITY\_CLAIM | The identity referenced by the input identity claim does not exist. |
| INVALID\_IDENTITY\_CLAIM | The identity claim requested is already in use. |
| NONEXISTANT\_DATA | The data requested for deletion does not exist. |
| UNKNOWN\_DOCUMENT\_CATEGORY | The data requested does not exist. |
| INVALID\_TOKEN | The data requested does not exist. |
| TOKEN\_EXPIRED | The data requested does not exist. |
| DUPLICATE\_ENCOUNTER | The input encounter ID for a new encounter already exists for that subject. |
| IDENTIFICATION\_RESULT\_NOT\_YET\_AVAILABLE | The result of an asynchronous identification process is not yet available. |
| UNKNOWN\_FORMAT | An unknown format was detected. |
| INVALID\_LICENSE | An invalid license was found. |
| SERVICE\_NOT\_IMPLEMENTED | The requested service/function is not implemented. |
| INVALID\_ENCOUNTER\_TYPE | The input encounter type is not a recognized category. |
| INVALID\_PROCESSING\_OPTION | The supplied processing options are invalid. |
| CANNOT\_STORE\_DATA | Data cannot be stored due to an internal reason. |
| CANNOT\_PROCESS\_DATA | Data cannot be processed. |
| CANNOT\_CREATE\_TEMPLATE | A biometric template could not be created from the raw input data (either for purpose enroll or verify/identify). |
| CANNOT\_DELETE\_DATA | Data cannot be deleted. |
| CANNOT\_RETRIEVE\_DATA | Data cannot be retrieved (read). |
| CANNOT\_INITIALIZE\_INTERNAL\_MODULES | Internal modules cannot be initialized and/or loaded. |
| CANNOT\_VERIFY\_DATA | Cannot perform a 1:N verification of the supplied and/or stored data. |
| CANNOT\_IDENTIFY\_DATA | Cannot perform a 1:N identification of the supplied and/or stored data. |
| INVALID\_CONFIGURATION | The (central) system was configured improperly. |
| CANNOT\_CHECK\_QUALITY | The quality check cannot be performed due to an internal reason. |
| INTERNAL\_DATABASE\_ERROR | An internal error during the database connection occurred. |
| CANNOT\_UPDATE\_DATA | Data cannot be updated. |
| BIOMETRIC\_TYPE\_NOT\_SUPPORTED | The biometric modality/type is not supported by the system. |

NOTES:

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

### 3.2.6 BIASFaultDetail

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BIASFaultDetail |  |  | Y | Defines the error information associated with a SOAP fault. |
| BIASFaultType | [BIASFaultCode](#_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. |

### 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](#_BIASIDType) | 0..1 | C | A system unique identifier for a subject.  Required as input to many operations. |
| IdentityClaim | [BIASIDType](#_BIASIDType) | 0..1 | N | An identifier by which a subject is known to a particular gallery or population group. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| EncounterList | [EncounterListType](#_EncounterListType) | 0..1 | N | A list of encounters associated with a subject. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 0..1 | N | An Identity’s biographic data. |
| BiometricData | [BIASBiometricDataType](#_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 BiographicDataItemType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BiographicDataItemType |  |  | 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 BiographicDataItemListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BiographicDataItemListType |  |  | Y | Defines a list of biographic data elements. |
| BiographicDataItem | [BiographicDataItemType](#_3.2.11_BiographicDataItemType) | 1..\* | Y | Data structure containing information about a biographic record. |

### 3.2.12 BiographicDataListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BiographicDataListType |  |  | Y | Defines a list of biographic data. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 0..\* | N | Data structure containing information about a biographic record. |

### 3.2.13 BiographicDataSetType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BiographicDataSetType |  |  | Y | Defines a set of biographic data that is formatted according to the specified format. |
| name | string | 1 | Y | The name of the biographic data format. Use these names for common formats: FBI-EFTS [EFTS], FBI-EBTS [EBTS-FBI], DOD-EBTS [EBTS-DOD], INT-I [INT-I], NIEM [NIEM], xNAL Error! Reference source not found., HR-XML [[HR-XML](#HRXML)]. |
| 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](#_BiographicDataItemListType) to represent a list of elements or the [BiographicDataSetType](#_BiographicDataSetType) to represent a complete, formatted set of biographic information.  One of the following elements must be present. |
| LastName | string | 0..1 | N | The last name of a subject. |
| FirstName | string | 0..1 | N | The first name of a subject. |
| BiographicDataItemList | [BiographicDataItemListType](#_BiographicDataItemListType) | 0..1 | N | A list of biographic data elements. |
| BiographicDataItem | [BiographicDataItemType](#_BiographicDataItemType) | 1..\* | Y | A single biographic data element. |
| BiographicDataSet | [BiographicDataSetType](#_BiographicDataSetType) | 0..1 | N | A set of biographic data information. |

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

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

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

### 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. |

### 3.2.16 BiometricDataListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| BiometricDataListType |  |  | Y | A list of biometric data elements. |
| BiometricDataRecord | [BiometricDataType](#_BiometricDataElementType) | 0..\* | N | Data structure containing information about a biometric record. |

### 3.2.17 CandidateListResultType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CandidateListResultType |  |  | Y | Defines a set of candidates, utilizing the [CandidateType](#_CandidateType) to represent each element in the set. |
| CandidateList | [CandidateListType](#_CandidateListType) | 1 | Y | The candidate list. |

### 3.2.18 CandidateListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CandidateListType |  |  | Y | Defines a set of candidates, utilizing the [CandidateType](#_CandidateType) to represent each element in the set. |
| Candidate | [CandidateType](#_CandidateType) | 0..\* | N | A single candidate. |

### 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](#_Score) | 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](#_BIASIDType) | 0..1 | Y | The identifier of the subject. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 0..1 | N | Biographic data associated with the candidate match. |
| BIRList | [CBEFF\_BIR\_ListType](#_CBEFF_BIR_ListType) | 1 | Y | Biometric data associated with the candidate match. |

### 3.2.20 CapabilityListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CapabilityListType |  |  | Y | Defines a set of capabilities. |
| Capability | [CapabilityType](#_CapabilityType) | 0..\* | N | A single capability. |

### 3.2.21 CapabilityName

|  |  |
| --- | --- |
| **Type:** | string |
| **Description:** | A list of capability items. |

**CapabilityName Enumeration Values**

| **Value** | **Description** |
| --- | --- |
| AggregateInputDataOptional | A data element accepted as optional input by the implementing system for the aggregate services.  The Capability Value should be set to the name of the data element that will be accepted by the aggregate services.  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:  “Delete”  “Enroll”  “Identify”  “Verify”  “All” |
| AggregateInputDataRequired | A data element required as input by the implementing system for the aggregate services.  The Capability Value should be set to the name of the data element that will be accepted by the aggregate services.  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:  “Delete”  “Enroll”  “Identify”  “Verify”  “All” |
| AggregateProcessingOption | A processing option supported by the implementing system for the aggregate services.  The Capability Value should be set to the option identifier, or “key” field, for the Processing Option parameter in the aggregate services.  The Capability Supporting Value that indicates the option value, or “value” field, for the Processing Option parameter in the aggregate services, if applicable.  The Capability Additional Info should indicate which aggreagate services support the processing option, using one or more of the following values, each separated by a comma:  “Delete”  “Enroll”  “Identify”  “Verify”  “Retrieve”  “All” |
| AggregateReturnData | A data element returned by the implementing system for the aggregate services.  The Capability Value should be set to the name of the data element that will be returned by the aggregate services.  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:  “Delete”  “Enroll”  “Identify”  “Verify”  “Retrieve”  “All” |
| AggregateServiceDescription | Describes the processing logic of an aggregate service supported by the implementing system.  The Capability Value should be set to the name of the data element that describes the aggregate services.  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:  “Delete”  “Enroll”  “Identify”  “Verify”  “Retrieve” |
| BiographicDataSet | Identifies a biographic data set supported by the implementing system.  The Capability Value should contain the name of the biographic data format supported by the implementing system(e.g. “EBTS” or “NIEM”).  The Capability Supporting Value should contain the version of the biographic data format supported by the implementing system.  The Capability Additional Info should contain the biographic data format type supported by the implementing system (e.g. ASCII or XML). |
| CBEFFPatronFormat | A patron format supported by the implementing system.  The Capability Value should contain the format owner.  The Capability Supporting Value should contain the format type. |
| ClassificationAlgorithmType | A classification algorithm type supported by the implementing system.  The Capability Value should contain the name of classification alogorithm type supported by the implementing system. |
| ConformanceClass | Identifies the conformance class of the BIAS implementation.  The Capability Value should be set to one of the following:  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 | A gallery or population group supported by the implementing system.  The Capability Value should be the same as the value used for the Gallery ID parameter in the [Add Subject to Gallery](#_AddSubjectToGallery), [Delete Biographic Data](#_DeleteBiographicData), [Delete Biometric Data](#_DeleteBiometricData), [Delete Subject From Gallery](#_DeleteSubjectFromGallery), [Identify Subject](#_IdentifySubject), [Retrieve Biographic Information](#_RetrieveBiographicInformation), [Retrieve Biometric Information](#_RetrieveBiometricInformation), [Retrieve Document Data](#_RetrieveDocumentData), [Set Biographic Data](#_SetBiographicData_1), [Set Biometric Data](#_SetBiometricData), [Set Document Data,](#_SetDocumentData) and [Verify Subject](#_VerifySubject) Services. |
| IdentityModel | Identifies whether the implementing system is person-centric or encounter-centric based.  The Capability Value shall be set to one of the following:  “person”  “encounter” |
| MatchAlgorithm | A match algorithm vendor and algorithm vendor product ID supported by the implementing system.  The Capability Value shall contain the algorighm vendor.  The Capability Supporting Value shall contain the algorithm vendor product ID.  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.  The Capability Description shall contain the software version of the match algorithm. |
| MatchScore | Identifies the use of match scores returned by the implementing system.  The Capability Value shall be set to the end-of-score-range that signifies a match.  The Capability Supporting Value shall be set to the end-of-score-range that signifies a no-match.  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. |
| QualityAlgorithm | A quality algorithm vendor and algorithm vendor product ID supported by the implementing system.  The Capability Value shall contain the algorighm vendor.  The Capability Supporting Value shall contain the algorithm vendor product ID.  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.  The Capability Description shall contain the software version of the quality algorithm. |
| SupportedBiometric | A biometric type supported by the implementing system.  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”).  The Capability Supporting Value shall indicate if the implementing system supports matching for the biometric type, using one of he following values:  “1” (identification)  “2” (verification)  “3” (identification and verification)  “4” (no comparison supported) |
| TransformOperation | A transform operation type supported by the implementing system.  The Capability Value shall be equal to the value for the Transform Operation parameter in the Transform Biometric Data service.  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”. |

### 3.2.22 CapabilityType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CapabilityType |  |  | Y | Defines a single capability supported by an implementing system. |
| CapabilityName | [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. |

### 3.2.23 CBEFF\_BIR\_ListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CBEFF\_BIR\_ListType |  |  | Y | A list of CBEFF-BIR elements. |
| BIR | [CBEFF\_BIR\_Type](#_3.2.22_CBEFF_BIR_Type) | 0..\* | N | CBEFF structure containing information about a biometric sample. |

### 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. |
| FormatType | positiveInteger | 1 | Y | Identifies the Patron format type. |
| BIR\_Information |  | 0..1 | N | Describes what is contained in a BIR. |
| BIR\_Info | oasis\_cbeff:BIRInfoType | 0..1 | N | Contains information about the CBEFF-BIR. |
| BDB\_Info | oasis\_cbeff:BDBInfoType | 0..1 | N | Contains information about the BDB in a simple CBEFF-BIR. |
| SB\_Info | oasis\_cbeff:SBInfoType | 0..1 | N | Contains information about the security block, if used, in a simple CBEFF-BIR. |
| BIR | [BaseBIRType](#_BaseBIRType) | 1 | Y | One of the following sub-elements must be present: [BinaryBIR](#_BinaryBIR), [URI\_BIR](#_URI_BIR), or [XML\_BIR](#_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. |

### 3.2.27 ClassificationData

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ClassificationData |  |  | Y | Contains information on classification results and the algorithm used to determine the classification. |
| Classification | [Classification](#_Classification) | 1 | Y | The result of the classification. |
| ClassificationAlgorithmType | [ClassificationAlgorithmType](#_ClassificationAlgorithmType) | 1 | Y | Identifies the type of classification algorithm that was used to perform the classification. |

### 3.2.28 DocumentDataType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DocumentDataType |  |  | Y | Defines a set of document data elements providing information about the presented identity document. |
| DocumentCategory | string | 1 | Y | The type of identity document presented (e.g. passport). |
| DocumentIDNumber | string | 0..1 | N | The number associated with the identity document (e.g. passport number) |
| DocumentIssuanceCountryCode | iso\_3166:CountryAlpha2CodeType | 0..1 | N | The ISO 2-character code for the country which issued the document or from within which it was issued. |
| DocumentIssuingOrganization | string | 0..1 | N | The entity which issued the identity document. |
| DocumentIssuanceDate | datetime | 0..1 | N | The date upon which the identity document was issued. |
| 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). |

### 3.2.29 DocumentDataListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DocumentDataListType |  |  | Y | Provides a list of documents. |
| DocumentData | [DocumentDataType](#_DocumentDataType) | 0..\* | Y | Data structure containing information about a document and optionally an image of that document. |

### 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. |

**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. |

### 3.2.31 EncounterListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| EncounterListType |  |  | Y | Defines a set of encounters. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..\* | N | The identifier of an encounter. |

### 3.2.32 FusionDecision

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

### 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](#_FusionInformationListType) | 0..\* | Y | A set of fusion information for a single identity. |

### 3.2.34 FusionInformationListType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| FusionInformationListType |  |  | Y | Contains at a minimum two sets of fusion input elements, as input to the [PerformFusion](#_PerformFusion) operation. |
| FusionElement | [FusionInformationType](#_FusionInformationType) | 2..\* | Y | A set of fusion information. |

### 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](#_FusionResult) | 0..1 | C | Either [FusionScore](#_FusionScore) or a [FusionDecision](#_FusionDecision) element MUST be used. |

### 3.2.36 FusionResult

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

### 3.2.37 FusionScore

|  |  |
| --- | --- |
| **Type:** | [ScoreType](#_Score) |
| **Description:** | The similarity score assigned by the matching algorithm. |

### 3.2.38 GenericRequestParameters

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GenericRequestParameters |  |  | Y | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested. |

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

### 3.2.39 IdentifySubjectResultType

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

### 3.2.40 InformationType

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

### 3.2.41 ListFilterType

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

### 3.2.42 MatchType

|  |  |
| --- | --- |
| **Type:** | boolean |
| **Description:** | The result of a fusion method. |

### 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. |

### 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](#_3.2.42_OptionType) | 0..\* | N | An option supported by the implementing system. |

### 3.2.45 ProductID

|  |  |
| --- | --- |
| **Type:** | string |
| **Description:** | The vendor’s ID for a particular product. |

### 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. |
| AlgorithmVendor | [VendorIdentifier](#_VendorIdentifier) | 1 | Y | The vendor of the quality algorithm used to determine the quality score. |
| AlgorithmVendorProductID | [ProductID](#_ProductID) | 1 | Y | The vendor’s ID for the algorithm used to determine the quality. |
| AlgorithmVersion | [VersionType](#_VersionType) | 0..1 | N | The version of the algorithm used to determine the quality. |

### 3.2.47 ResponseStatus

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ResponseStatus |  |  | Y |  |
| Return | [ReturnCode](#_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 B.

### 3.2.50 TokenResultType

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| TokenResultType |  |  | Y | Defines a token that is returned for asynchronous processing. |
| TokenType | [TokenType](#_TokenType) | 1 | Y | Defines a token that is returned for asynchronous processing. |

### 3.2.51 TokenType

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

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

### 3.2.52 URI\_BIR

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

### 3.2.53 VendorIdentifier

|  |  |
| --- | --- |
| **Type:** | string |
| **Description:** | Identifies a vendor. |

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

### 3.2.54 Version

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| Version |  |  | Y | For a description or definition of each data element, see the referenced CBEFF standards in the 3.2.24 CBEFF\_BIR\_Typeschema. |
| major | nonNegativeInteger | 1 | Y |
| minor | nonNegativeInteger | 1 | Y |

### 3.2.55 VersionType

|  |  |
| --- | --- |
| **Type:** | string |
| **Description:** | The version of a component. |

### 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 |

# 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.

## Primitive Operations

### AddSubjectToGallery

[AddSubjectToGalleryRequest](#AddSubjectToGalleryRequest)

[AddSubjectToGalleryResponse](#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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “AddSubjectToGallery”. |
| GalleryID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of the gallery or population group to which the subject will be added. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | The identity to add to the gallery. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| IdentityClaim | [BIASIDType](#_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](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| AddSubjectToGalleryResponse |  |  | Y | The response to an AddSubjectToGallery operation. |
| AddSubjectToGalleryResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### CheckQuality

[CheckQualityRequest](#CheckQualityRequest)

[CheckQualityResponse](#CheckQualityResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CheckQuality |  |  | Y | Calculate a quality score for a given biometric. |
| CheckQualityRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASBiometricDataType) | 0..1 | C | Data structure containing a single biometric sample for which a quality score is to be determined; required if no Subject ID is provided. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 1 | Y | The biometric sample. |
| Identity | [BIASIdentity](#_BIASIdentity) | 0..1 | C | Required if no BIR is provided. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | Identifies the subject; required if no BIR is provided. |
| Qualityinfo | [QualityData](#_QualityData) | 0..1 | N | Specifies a particular algorithm vendor and vender product ID. |
| AlgorithmVendor | [VendorIdentifier](#_VendorIdentifier) | 1 | Y | The vendor of the quality algorithm used to determine the quality score. |
| AlgorithmVendorProductID | [ProductID](#_ProductID) | 1 | C | The vendor assigned ID for the algorithm used to determine the quality; required as input if algorithm vendor is provided |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CheckQualityResponse |  |  | Y | The response to a CheckQuality operation. |
| CheckQualityResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_ReturnCode) | 1 | Y | The return code indicates the return status of the operation. |
| Message | string | 0..1 | N | A short message corresponding to the return code. |
| QualityInfo | [QualityData](#_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](#_VendorIdentifier) | 1 | Y | The vendor of the quality algorithm used to determine the quality score. |
| AlgorithmVendorProductID | [ProductID](#_ProductID) | 1 | Y | The vendor’s ID for the algorithm used to determine the quality. |
| AlgorithmVersion | [VersionType](#_VersionType) | 1 | Y | The version of the algorithm used to determine the quality. |

### ClassifyBiometricData

[ClassifyBiometricDataRequest](#ClassifyBiometricDataRequest)

[ClassifyBiometricDataResponse](#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](#_QueryCapabilities) operation.

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ClassifyBiometricData |  |  | Y | Classifies a biometric sample. |
| * ClassifyBiometricDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “ClassifyBiometricData”. |
| BiometricData | [BIASBiometricDataType](#_BIASBiometricDataType) | 1 | Y | Data structure containing a single biometric sample for which the classification is to be determined. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 1 | Y | The biometric sample. |
| ClassificationData |  | 0..1 | N | Contains the algorithm to be used to determine the classification. |
| ClassificationAlgorithmType | [ClassificationAlgorithmType](#_ClassificationAlgorithmType) | 1 | Y | Identifies the type of classification algorithm to be used to perform the classification. |

**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](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_ClassificationData) | 1 | Y | Information on the results and type of classification performed. |
| Classification | [Classification](#_Classification) | 1 | Y | The result of the classification. |
| ClassificationAlgorithmType | [ClassificationAlgorithmType](#_ClassificationAlgorithmType) | 1 | Y | Identifies the type of classification algorithm that was used to perform the classification. |

### CreateEncounter

[CreateEncounterRequest](#CreateEncounterRequest)

[CreateEncounterResponse](#CreateEncounterResponse)

The CreateEncounteroperation 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](#_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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| EncounterType | [EncounterCategoryType](#_3.2.29_EncounterCategoryType) | 1 | Y | Identifies the type of encounter during which data was collected from the subject, as determined by the requester. |

**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](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |

### CreateSubject

[CreateSubjectRequest](#CreateSubjectRequest)

[CreateSubjectResponse](#CreateSubjectResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CreateSubject |  |  | Y |  |
| CreateSubjectRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “CreateSubject”. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| CreateSubjectResponse |  |  | Y | The response to a CreateSubject operation, containing the subject ID of the new subject record. |
| CreateSubjectResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |

### DeleteBiographicData

[DeleteBiographicDataRequest](#DeleteBiographicDataRequest)

[DeleteBiographicDataResponse](#DeleteBiographicDataResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteBiographicData |  |  | Y | Erase all of the biographic data associated with a given subject record or, in the encounter-centric model, with a given encounter. |
| DeleteBiographicDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group from which the biographic information will be deleted. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteBiographicDataResponse |  |  | Y | The response to a DeleteBiographicData operation. |
| * DeleteBiographicDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### DeleteBiometricData

[DeleteBiometricDataRequest](#DeleteBiometricDataRequest)

[DeleteBiometricDataResponse](#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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_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](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group from which the biometric information will be deleted. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteBiometricDataResponse |  |  | Y | The response to a DeleteBiometricData operation. |
| DeleteBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### DeleteDocumentData

[DeleteDocumentDataRequest](#DeleteDocumentDataRequest)

[DeleteDocumentDataResponse](#DeleteDocumentDataResponse)

The DeleteDocumentData operation erases all of the document data of the specified category(ies) associated with a given subject record. In the encounter-centric model the service erases all of the document data associated with a given encounter, and therefore the encounter ID MUST be specified. If no encounter ID is specified, or it is null, document data will be removed from all encounters. If no categories are specified, then all categories (for the specified encounters) 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** |
| --- | --- | --- | --- | --- |
| 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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| DocumentData | [DocumentDataType](file:///C:\Users\karenm\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.Outlook\Z4GLQVP7\BIAS-v2.0-wd01_WithChanges10142015.docx#DocumentDataType) | 0..1 | N | Defines a set of document data elements providing information about the presented identity document. |
| DocumentCategory | string | 1 | Y | The category(ies) of the identity documents to be deleted. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteDocumentDataResponse |  |  | Y | The response to a DeleteDocumentData operation. |
| DeleteDocumentDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### DeleteEncounter

[DeleteEncounterRequest](#DeleteEncounterRequest)

[DeleteEncounterResponse](#DeleteEncounterResponse)

The DeleteEncounter operation deletes an existing encounter record from the system. When deleting an encounter, 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** |
| --- | --- | --- | --- | --- |
| DeleteEncounter |  |  | Y | Delete an existing encounter record and, any associated encounter information. |
| * DeleteEncounterRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of an encounter associated with the subject. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteEncounterResponse |  |  | Y | The response to a DeleteEncounter operation. |
| DeleteEncounterResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### DeleteSubject

[DeleteSubjectRequest](#DeleteSubjectRequest)

[DeleteSubjectResponse](#DeleteSubjectResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteSubject |  |  | Y | Delete an existing subject record and, in an encounter-centric model, any associated encounter information. |
| * DeleteSubjectRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y | The identity of the subject to delete. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteSubjectResponse |  |  | Y | The response to a DeleteSubject operation. |
| DeleteSubjectResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### DeleteSubjectFromGallery

[DeleteSubjectFromGalleryRequest](#DeleteSubjectFromGalleryRequest)

[DeleteSubjectFromGalleryResponse](#DeleteSubjectFromGalleryResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteSubjectFromGallery |  |  | Y | Remove the registration of a subject from a gallery or population group. |
| DeleteSubjectFromGalleryRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIDType) | 1 | Y | The identifier of the gallery or population group from which the subject will be deleted. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | The identity to remove from the gallery. |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | A system unique identifier for a subject.  Required if an Identity Claim is not provided. |
| IdentityClaim | [BIASIDType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteSubjectFromGalleryResponse |  |  | Y | The response to a DeleteSubjectFromGallery operation. |
| DeleteSubjectFromGalleryResponsePackage |  |  |  |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### GetIdentifySubjectResults

[GetIdentifyResultsRequest](#GetIdentifyResultsRequest)

[GetIdentifySubjectResultsResponse](#GetIdentifySubjectResultsResponse)

The GetIdentifySubjectResults operation retrieves the identification results for the specified token. This opereation is used in conjunction with the [IdentifySubject](#_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](#_IdentifySubject) operation. |
| * GetIdentifySubjectResultsRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “GetIdentifySubjectResults”. |
| Token | [TokenType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetIdentifySubjectResultsResponse |  |  | Y | The response to a GetIdentifySubjectResults operation, which includes a candidate list. |
| GetIdentifySubjectResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_CandidateListType) | 1 | Y | A rank-ordered list of candidates that have a likelihood of matching the input biometric sample. |
| Candidate | [CandidateType](#_CandidateType) | 0..\* | N | A single candidate. |
| Score | [ScoreType](#_Score) | 0..1 | N | The match score. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 0..1 | N | Biographic data associated with the candidate match. |
| BIRList | [CBEFF\_BIR\_ListType](#_CBEFF_BIR_ListType) | 1 | Y | Biometric data associated with the candidate match. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..\* | N | CBEFF structure containing information about a biometric sample. |

### IdentifySubject

[IdentifySubjectRequest](#IdentifySubjectRequest)

[IdentifySubjectResponse](#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](#_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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “IdentifySubject”. |
| GalleryID | [BIASIDType](#_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](#_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](#_BIASIdentity) | 1 | Y | Contains the BIR, a data structure containing the biometric sample for the search. |
| BiometricData | [BIASBiometricDataType](#_BIASBiometricDataType) | 1 | Y | An Identity’s biometric data. |
| BIR | [CBEFF\_BIR\_Type](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| IdentifySubjectResponse |  |  | Y | The response to an IdentifySubject operation, returning a rank-ordered candidate list. |
| IdentifySubjectResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_ReturnCode) | 1 | Y | The return code indicates the return status of the operation. |
| Message | string | 0..1 | N | A short message corresponding to the return code. |
| CandidateList | [CandidateListResultType](#_CandidateListResultType)  (see [IdentifySubjectResultType](#_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](#_CandidateType) | 0..\* | N | A single candidate. | |
| Score | [ScoreType](#_Score) | 0..1 | N | The match score. | |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 0..1 | N | Biographic data associated with the candidate match. | |
| BIRList | [CBEFF\_BIR\_ListType](#_CBEFF_BIR_ListType) | 1 | Y | Biometric data associated with the candidate match. | |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..\* | N | CBEFF structure containing information about a biometric sample. | |
| Token | [TokenResultType](#_TokenResultType)  (see [IdentifySubjectResultType](#_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.

### ListBiographicData

[ListBiographicDataRequest](#ListBiographicDataRequest)

[ListBiographicDataResponse](#ListBiographicDataResponse)

The ListBiographicData operation lists the biographic data elements stored for a subject using the Biographic Data Elements output parameter. Note that no actual biographic data is returned by this operation (see the [RetrieveBiographicData](#_RetrieveBiographicData) operation to obtain the biographic data). In the encounter-centric model, an encounter ID MAY be specified to indicate that only the biographic 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 biographic data using the Encounter List output parameter, and the Biographic Data Element List output parameter is empty.

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ListBiographicData |  |  | Y | Lists the biographic data elements stored for a subject. |
| ListBiographicDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject. |
| EncounterType | [EncounterCategoryType](#_3.2.29_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. |

**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](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_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](#_3.2.13_BiographicDataType) | 0..1 | C | An Identity’s biographic data elements that are stored in the implementing system. |
| BiographicDataItemList | BiographicDataItemListType | 0..1 | N | A list of biographic data elements. |
| BiographicDataItem | [BiographicDataItemType](#_BiographicDataItemType) | 1..\* | Y | A single biographic data element. |
| Name | string | 1 | Y | The name of the biographic data item. |
| Type | string | 1 | Y | The data type for the biographic data item. |
| EncounterList | [EncounterListType](#_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](#_BIASIDType) | 0..\* | N | The identifier of an encounter. |

### ListBiometricData

[ListBiometricDataRequest](#ListBiometricDataRequest)

[ListBiometricDataResponse](#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](#_RetrieveBiometricInformation) 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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. | |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. | |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. | |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. | |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject. | |
| EncounterType | [EncounterCategoryType](#_3.2.29_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](#_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. | |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ListBiometricDataResponse |  |  | Y | The response to a ListBiometricData operation, containing a list of biometric data elements stored for a subject. In the encounter-centric model, the biometric data elements for a specific encounter are returned. If an encounter ID is not specified and encounter data exists for the subject, the list of encounter IDs which contain biometric data is returned. |
| ListBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_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](#_BIASBiometricDataType) | 0..1 | C | An Identity’s biometric data. |
| BiometricDataList | [BiometricDataListType](#_BiometricDataListType) | 0..1 | N | A list of biometric data elements. |
| BiometricDataRecord | [BiometricDataType](#_BiometricDataElementType) | 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. |
| EncounterList | [EncounterListType](#_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](#_BIASIDType) | 1..\* | Y | The identifier of an encounter. |

### ListDocumentData

[ListDocumentDataRequest](#ListDocumentDataRequest)

[ListDocumentDataResponse](#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](#_SetBiographicData) 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 mustl 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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “ListDocumentData”. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject. |
| EncounterType | [EncounterCategoryType](#_3.2.29_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| ListDocumentDataResponse |  |  | Y | The response to a ListDocumentData operation. |
| ListDocumentDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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 | [3.2.29 DocumentDataListType](#_3.2.28_DocumentDataListType) | 0..1 | C | A list of document categories associated with a subject or encounter; non-empty if the service was successful, document data exists, and either the person-centric model is being used or the encounter-centric model is being used and an encounter identifier was specified. |
| DocumentData | [3.2.28 DocumentDataType](#_3.2.27_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](#_BIASIdentity) | 0..1 | C |  |
| EncounterList | [EncounterListType](#_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](#_BIASIDType) | 1..\* | Y | The identifier of an encounter. |

### PerformFusion

[PerformFusionRequest](#PerformFusionRequest)

[PerformFusionResponse](#PerformFusionResponse)

The PerformFusion operation accepts either match score or match decision information and creates a fused match result. The [FusionInformationListType](#_FusionInformationListType), through the [FusionInformationType](#_FusionInformationType), provides specific elements for match score input and match decision input for a single identity, while the [FusionIdentityListType](#_3.2.32_FusionIdentityListType) provides the ability to submit multiple identities to the Perform Fusion operation. The fusion method and processes are left to the implementing system.

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| PerformFusion |  |  | Y | Accepts either match score or match decision information and creates a fused match result. |
| PerformFusionRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_3.2.32_FusionIdentityListType) | 1 | Y |  |
| FusionIdentity | [FusionInformationListType](#_FusionInformationListType) | 1 | Y | Score or decision input information to the fusion method for each identity. |
| FusionElement | [FusionInformationType](#_FusionInformationType) | 2..\* | Y | A set of fusion information. |
| BiometricType | oasis\_cbeff:MultipleTypesType | 1 | Y | The type of biological or behavioral data stored in the biometric record, as defined by CBEFF. |
| BiometricSubType | oasis\_cbeff:SubtypeType | 0..1 | N | More specifically defines the type of biometric data stored in the biometric record. |
| AlgorithmOwner | string | 1 | Y | The owner or vendor of the algorithm used to determine the score or decision. |
| 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](#_FusionResult) | 0..1 | C | Either [FusionScore](#_FusionScore) or a [FusionDecision](#_FusionDecision) element MUST be used. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| PerformFusionResponse |  |  | Y | The response to the PerformFusion operation. |
| PerformFusionResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_MatchType) | 1 | Y | Indicates the result of the fusion method. |

### QueryCapabilities

[QueryCapabilitiesRequest](#QueryCapabilitiesRequest)

[QueryCapabilitiesResponse](#QueryCapabilitiesResponse)

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

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

**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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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”. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| QueryCapabilitiesResponse |  |  | Y | The response to a QueryCapabilities operation. |
| * QueryCapabilitiesResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_CapabilityListType) | 1 | Y | A list of capabilities supported by the BIAS implementation. |
| Capability | [CapabilityType](#_CapabilityType) | 0..\* | N | A single capability. |
| CapabilityName | [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. |

### RetrieveBiographicData

[RetrieveBiographicDataRequest](#RetrieveBiographicDataRequest)

[RetrieveBiographicDataResponse](#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](#_GenericRequestParameters) | 0..1 | N | | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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: “RetrieveBiographicData”. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | | N | The identifier of an encounter associated with the subject. |
| EncounterType | [EncounterCategoryType](#_3.2.29_EncounterCategoryType) | 0..1 | | N | Identifies the type of encounter during which data was collected from the subject, as determined by the requester. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | | N | The identifier of the gallery or population group from which the biographic information will be retrieved. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveBiographicDataResponse |  |  | Y | The response to a RetrieveBiographicData operation. |
| RetrieveBiographicDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y | Includes the set of biographic data associated with a subject. |
| BiographicDataList | [BiographicDataListType](#_3.2.10_BiographicDataListType) | 1 | Y | An Identity’s biographic data.  One of the following elements MUST be present. |
| LastName | string | 0..1 | C | The last name of a subject. |
| FirstName | string | 0..1 | C | The first name of a subject. |
| BiographicDataItemList | [BiographicDataItemType](#_BiographicDataItemType) | 0..\* | C | A list of biographic data elements. |
| BiographicDataSet | [BiographicDataItemType](#_BiographicDataItemType) | 0..1 | C | A set of biographic data information. |

### RetrieveBiometricData

[RetrieveBiometricDataRequest](#RetrieveBiometricDataRequest)

[RetrieveBiometricDataResponse](#RetrieveBiometricDataResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveBiometricData |  |  | Y | Retrieves the biometric data associated with a subject ID. |
| RetrieveBiometricDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y | Identifies the subject or, in the encounter-centric model, a subject and an encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject. |
| EncounterType | [EncounterCategoryType](#_3.2.29_EncounterCategoryType) | 0..1 | N | Identifies the type of encounter during which data was collected from the subject, as determined by the requester. |
| GalleryID | [BIASIDType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveBiometricDataResponse |  |  | Y | The response to a RetrieveBiometricData operation. |
| RetrieveBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y | Includes the biometric data associated with a subject. |
| BiometricData | [BIASBiometricDataType](#_BIASBiometricDataType) | 1 | Y | An Identity’s biometric data. |
| BIRList | [CBEFF\_BIR\_ListType](#_CBEFF_BIR_ListType) | 1 | Y | A list of CBEFF-BIR elements. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..\* | N | CBEFF structure containing information about a biometric sample. |

### RetrieveDocumentData

[RetrieveDocumentDataRequest](#RetrieveDocumentDataRequest)

[RetrieveDocumentDataResponse](#RetrieveDocumentDataResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveDocumentData |  |  | Y |  |
| * RetrieveDocumentDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject. |
| EncounterType | [EncounterCategoryType](#_3.2.29_EncounterCategoryType) | 0..1 | C | Identifies the category of encounter. |
| DocumentData | [DocumentDataType](#_3.2.27_DocumentDataType) | 0..1 | N | Defines a set of document data elements providing information about the requested identity document. |
| DocumentCategory | string | 0..1 | Y | The category(ies) of the identity documents to be retrieved. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group from which the biographic information will be retrieved. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveDocumentDataResponse |  |  | Y | The response to a RetrieveDocumentData operation. |
| RetrieveDocumentDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_3.2.28_DocumentDataListType) | 1 | Y | A list of document data associated with a subject or encounter. |

### SetBiographicData

[SetBiographicDataRequest](#SetBiographicDataRequest)

[SetBiographicDataResponse](#SetBiometricDataResponse)

The SetBiographicData operation associates biographic data to a given subject record. The identity model of the system determines whether the biographic information should replace any existing biographic information (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 biographic with biometric and/or document information (assuming biometric and/or document information was previously associated using the [SetBiometricData](#_SetBiometricData) and/or [SetDocumentData](#_SetDocumentData) operations). If the encounter ID is omitted for the encounter-centric model, the operation returns a system-assigned encounter ID.

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** | |
| --- | --- | --- | --- | --- | --- |
| SetBiographicData |  |  | Y | Associates biographic data to a given subject record. | |
| SetBiographicDataRequest |  | 1 | Y |  | |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. | |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. | |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 1 | Y | An Identity’s biographic data.  One of the following elements MUST be present. |
| LastName | string | 0..1 | C | The last name of a subject. |
| FirstName | string | 0..1 | C | The first name of a subject. |
| BiographicDataItemList | [BiographicDataItemType](#_BiographicDataItemType) | 0..\* | C | A list of biographic data elements. |
| BiographicDataSet | [BiographicDataSetType](#_BiographicDataSetType) | 0..1 | C | A set of biographic data information. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group to which the biographic will be added. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| SetBiographicDataResponse |  |  | Y | The response to a SetBiographicData operation. |
| * SetBiographicDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 0..1 | C | In an encounter-centric model, identifies the encounter ID assigned to a new encounter. |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of an encounter associated with the subject. |

### SetBiometricData

[SetBiometricDataRequest](#SetBiometricDataRequest)

[SetBiometricDataResponse](#SetBiometricDataResponse)

The SetBiometricData operation associates biometric data to a given subject record. The identity model of the system determines whether the biometric information should replace any existing biometric information (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 biometric with biographic and/or document information (assuming biographic and/or document information was previously associated using the [SetBiographicData](#_SetBiographicData_1) and/or [SetDocumentData](#_SetDocumentData) operation). If the encounter ID is omitted for the encounter-centric model, the operation returns a system-assigned encounter ID.

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| SetBiometricData |  |  | Y | Associates biometric data to a given subject record. |
| SetBiometricDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject. |
| BiometricData | [BIASBiometricDataType](#_BIASBiometricDataType) | 1 | Y | An Identity’s biometric data. |
| BIRList | [CBEFF\_BIR\_ListType](#_CBEFF_BIR_ListType) | 1 | Y | A list of CBEFF-BIR elements. |
| BIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 1..\* | Y | CBEFF structure containing information about a biometric sample. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group to which the biometric will be added. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| SetBiometricDataResponse |  |  | Y | The response to a SetBiometricData operation. |
| SetBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### SetDocumentData

[SetDocumentDataRequest](#SetDocumentDataRequest)

[SetDocumentDataResponse](#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](#_SetBiographicData) and/or [SetBiometricData](#_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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 1 | Y | A system unique identifier for a subject |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of an encounter associated with the subject. |
| DocumentDataList | [DocumentDataListTyp](#_3.2.28_DocumentDataListType)e | 1 | Y | A list of document data to associate with the subject or encounter. |
| GalleryID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier of the gallery or population group to which the document information will be added. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| SetDocumentDataResponse |  |  | Y | The response to a SetDocumentData operation. |
| SetDocumentDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of an encounter associated with the subject. |

### TransformBiometricData

[TransformBiometricDataRequest](#TransformBiometricDataRequest)

[TransformBiometricDataResponse](#TransformBiometricDataResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| TransformBiometricData |  |  | Y | Transforms or processes a given biometric in one format into a new target format. |
| TransformBiometricDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_CBEFF_BIR_Type) | 1 | Y | Data structure containing the biometric information to be transformed. |
| TransformOperation | unsignedLong | 1 | Y | Value indicating the type of transformation to perform. |
| TransformControl | string | 0..1 | N | Specifies controls for the requested transform operation.  Note: This could be a compression ratio, target data format, etc. |

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

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| TransformBiometricDataResponse |  |  | Y | The response to a TransformBiometricData operation. |
| TransformBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_CBEFF_BIR_Type) | 1 | Y | Data structure containing the new, transformed biometric information. |

### UpdateBiographicData

[UpdateBiographicDataRequest](#UpdateBiographicDataRequest)

[UpdateBiographicDataResponse](#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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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: “UpdateBiographicData”. |
| Identity | [BIASIdentity](#_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](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| BiographicData | [BiographicDataType](#_BiographicDataType) | 1 | Y | An Identity’s biographic data.  One of the following elements MUST be present. |
| LastName | string | 0..1 | C | The last name of a subject. |
| FirstName | string | 0..1 | C | The first name of a subject. |
| BiographicDataItemList | [BiographicDataItemType](#_BiographicDataItemType) | 0..\* | C | A list of biographic data elements. |
| BiographicDataSet | [BiographicDataSetType](#_BiographicDataSetType) | 0..1 | C | A set of biographic data information. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| UpdateBiographicDataResponse |  |  | Y | The response to an UpdateBiographicData operation. |
| UpdateBiographicDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### UpdateBiometricData

[UpdateBiometricDataRequest](#UpdateBiometricDataRequest)

[UpdateBiometricDataResponse](#UpdateBiometricDataResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| UpdateBiometricData |  |  | Y | Updates a single biometric sample for a given subject record. |
| UpdateBiometricDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “UpdateBiometricData”. |
| Identity | [BIASIdentity](#_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](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| BiometricData | [BIASBiometricDataType](#_BIASBiometricDataType) | 1 | Y | An Identity’s biometric data. |
| BIR | [CBEFF\_BIR\_Type](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| UpdateBiometricDataResponse |  |  | Y | The response to an UpdateBiometricData operation. |
| UpdateBiometricDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### UpdateDocumentData

[UpdateDocumentDataRequest](#UpdateDocumentDataRequest)

[UpdateDocumentDataResponse](#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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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](#_BIASIDType) | 1 | Y | A system unique identifier for a subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required for encounter-centric models. |
| DocumentDataList | [DocumentDataListType](#_3.2.28_DocumentDataListType) | 1 | Y | A list of updated document data. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| UpdateDocumentDataResponse |  |  | Y | The response to an UpdateDocumentData operation. |
| UpdateDocumentDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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. |

### VerifySubject

[VerifySubjectRequest](#VerifySubjectRequest)

[VerifySubjectResponse](#VerifySubjectResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| VerifySubject |  |  | Y | Performs a 1:1 verification match between a given biometric and either a claim to identity in a given gallery or another given biometric. |
| VerifySubjectRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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](#_BIASIdentity) | 1 | Y | Includes the identifying information and/or input and reference biometric samples. |
| IdentityClaim | [BIASIDType](#_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](#_BIASBiometricDataType) | 1 | Y | An Identity’s biometric data. |
| InputBIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 1 | Y | Maps to specific ISO/IEC BIAS elements as required by that specification.  When multiple samples are included as input (e.g. in a multimodal operation), a complex BIR is used. |
| ReferenceBIR | [CBEFF\_BIR\_Type](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| VerifySubjectResponse |  |  | Y | The response to a VerifySubject operation. |
| * VerifySubjectResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_Score) | 0..1 | N | The score if the biometric information matched. |

## Aggregate Operations

### Delete

[DeleteRequest](#DeleteRequest)

[DeleteResponse](#DeleteResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| Delete |  |  | Y | Deletes a subject or, in an encounter-centric model, an existing encounter from the system. |
| DeleteRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_ProcessingOptionsType) | 1 | Y | Options that guide how the aggregate service request is processed. |
| Option | [OptionType](#_3.2.42_OptionType) | 0..\* | N | An option supported by the implementing system. |
| InputData | [InformationType](#_InformationType) | 0..1 | N | Contains the input data for the operation, as required by the implementing system. |
| Identity | [BIASIdentity](#_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](#_BIASIDType) | 0..1 | C | The identifier assigned to the subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier for the encounter; required for encounter-centric models. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| DeleteResponse |  |  | Y | The response to a Delete operation. |
| DeleteResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier assigned to the subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of the encounter; required for encounter-centric models. |
| ReturnData | [InformationType](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Token | [TokenType](#_3.2.50_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. |

### Enroll

[EnrollRequest](#EnrollRequest)

[EnrollResponse](#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](#_GetEnrollResults_1) 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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_ProcessingOptionsType) | 1 | Y | Options that guide how the aggregate service request is processed. |
| Option | [OptionType](#_3.2.42_OptionType) | 0..\* | N | An option supported by the implementing system. |
| InputData | [InformationType](#_InformationType) | 1 | Y | Contains the input data for the operation, as required by the implementing system. |
| Identity | [BIASIdentity](#_BIASIdentity) | 0..1 | N |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | N | The identifier for the encounter; required for encounter-centric models. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| EnrollResponse |  |  | Y | The response to an Enroll operation. |
| EnrollResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier assigned to the subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of the encounter; required for encounter-centric models. |
| ReturnData | [InformationType](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Token | [TokenType](#_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. |

### GetDeletionResults

[GetDeletionResultsRequest](#GetDeletionResultsRequest)

[GetDeletionResultsResponse](#GetDeletionResultsResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetDeletionResults |  |  | Y | Retrieves the deletion results for the specified token. |
| * GetDeletionResultsRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetDeletionResultsResponse |  |  | Y | The response to a GetDeletionResults operation. |
| GetDeletionResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |

### GetEnrollResults

[GetEnrollResultsRequest](#GetEnrollResultsRequest)

[GetEnrollResultsResponse](#GetEnrollResultsResponse)

The GetEnrollResults operation retrieves the enrollment results for the specified token. This operation is used in conjunction with the [Enroll](#_Enroll_1) 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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “GetEnrollResults”. |
| Token | [TokenType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetEnrollResultsResponse |  |  | Y | The response to a GetEnrollResults operation. |
| GetEnrollResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y |  |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier assigned to the subject. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of the encounter; if assigned. |

### GetIdentifyResults

[GetIdentifyResultsRequest](#GetIdentifyResultsRequest)

[GetIdentifyResultsResponse](#GetIdentifyResultsResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetIdentifyResults |  |  | Y | Retrieves the identification results for the specified token |
| GetIdentifyResultsRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetIdentifyResultsResponse |  |  | Y | The response to a GetIdentifyResults operation. |
| * GetIdentifyResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_BIASIdentity) |  |  |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of the encounter, if assigned. |
| CandidateList | [CandidateListType](#_CandidateListType) | 0..1 | C | A rank-ordered list of candidates that have a likelihood of matching the input biometric sample. |
| ReturnData | [InformationType](#_InformationType) | 0..1 | N | Contains the output data for the response. |

### GetUpdateResults

[GetUpdateResultsRequest](#GetUpdateResultsRequest)

[GetUpdateResultsResponse](#GetUpdateResultsResponse)

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

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetUpdateResults |  |  | Y | Retrieves the Update results for the specified token |
| GetUpdateResultsRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_ApplicationUserIdentifier) | 0..1 | N | Identifies the user or instance of the requesting application. |
| BIASOperationName | string | 0..1 | N | Identifies the BIAS operation that is being requested: “GetUpdateResults”. |
| Token | [TokenType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetUpdateResultsResponse |  |  | Y | The response to a GetUpdateResults operation. |
| * GetUpdateResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |

### GetVerifyResults

[GetVerifyResultsRequest](#GetVerifyResultsRequest)

[GetVerifyResultsResponse](#GetVerifyResultsResponse)

The GetVerifyResults operation retrieves the verification results for the specified token. This operation is used in conjunction with the [Verify](#_Verify_1) operation. If the Verify operation is implemented as an asynchronous service, the implementing system returns a non-zero token and the GetVerifyResults operation is used to poll for the results of the original Verify request.

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetVerifyResults |  |  | Y | Retrieves the verification results for the specified token |
| GetVerifyResultsRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| GetVerifyResultsResponse |  |  | Y | The response to a GetVerifyResults operation. |
| GetVerifyResultsResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Identity | [BIASIdentity](#_BIASIdentity) | 0..1 | C |  |
| EncounterID | [BIASIDType](#_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 | [Score](#_Score)Type | 0..1 | N | The score if the biometric information matched. |

### Identify

[IdentifyRequest](#IdentifyRequest)

[IdentifyResponse](#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](#_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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_3.2.43_ProcessingOptionsType) | 1 | Y | Options that guide how the aggregate service request is processed. |
| Option | [OptionType](#_3.2.42_OptionType) | 0..\* | N | An option supported by the implementing system. |
| InputData | [InformationType](#_InformationType) | 1 | Y | Contains the input data for the aggregate services. |
| GalleryID | [BIASIDType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| IdentifyResponse |  |  | Y | The response to an Identify operation. |
| IdentifyResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Identity | [BIASIdentity](#_BIASIdentity) | 0..1 | C |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of the encounter, if assigned. |
| CandidateList | [CandidateListType](#_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](#_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. |

### RetrieveData

[RetrieveDataRequest](#RetrieveDataRequest)

[RetrieveDataResponse](#RetrieveDataResponse)

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveData |  |  | Y | Retrieves requested information about a subject or encounter. |
| RetrieveDataRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_3.2.43_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](#_3.2.43_OptionType) | 0..\* | N | An option supported by the implementing system. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | Includes the identifier of the subject or encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | A system unique identifier for a subject.  Required if an Encounter ID is not provided. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required if a Subject ID is not provided. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| RetrieveDataResponse |  |  | Y | Response to a RetrieveData operation. |
| * RetrieveDataResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |

### Update

[UpdateRequest](#UpdateRequest)

[UpdateResponse](#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](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_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](#_3.2.42_OptionType) | 0..\* | N | An option supported by the implementing system. |
| InputData | [InformationType](#_InformationType) | 1 | Y | Contains the subject data to update. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | Includes the identifier of the subject or encounter. |
| SubjectID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of the subject;  Required if an Encounter ID is not provided. |
| EncounterID | [BIASIDType](#_BIASIDType) | 0..1 | C | The identifier of an encounter associated with the subject.  Required if a Subject ID is not provided. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| UpdateResponse |  |  | Y | Response to an Update operation. |
| * UpdateResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_InformationType) | 0..1 | N | Contains the output data for the response. |
| Token | [TokenType](#_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. |

### Verify

[VerifyRequest](#VerifyRequest)

[VerifyResponse](#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](#_GetVerifyResults_1) operation is used to poll for the results of the Verify request.

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

**Request Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| Verify |  |  | Y | Performs a 1:1 verification function. |
| VerifyRequest |  | 1 | Y |  |
| GenericRequestParameters | [GenericRequestParameters](#_GenericRequestParameters) | 0..1 | N | Common request parameters that can be used to identify the requester. |
| Application | [ApplicationIdentifier](#_ApplicationIdentifier) | 0..1 | N | Identifies the requesting application. |
| ApplicationUser | [ApplicationUserIdentifier](#_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](#_ProcessingOptionsType) | 1 | Y | Options that guide how the aggregate service request is processed. |
| Option | [OptionType](#_ProcessingOptionsType) | 0..\* | N | An option supported by the implementing system. |
| InputData | [InformationType](#_InformationType) | 1 | Y | Contains the input data for the aggregate services. |
| Identity | [BIASIdentity](#_BIASIdentity) | 1 | Y | Includes either the Identity Claim or Reference BIR. |
| IdentityClaim | [BIASIDType](#_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](#_BIASBiometricDataType) | 0..1 | N | An Identity’s biometric data. |
| ReferenceBIR | [CBEFF\_BIR\_Type](#_CBEFF_BIR_Type) | 0..1 | C | Maps to specific ISO/IEC BIAS elements as required by that specification.  Required if an Identity Claim is not provided. |
| GalleryID | [BIASIDType](#_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. |

**Response Message**

| **Field** | **Type** | **#** | **?** | **Meaning** |
| --- | --- | --- | --- | --- |
| VerifyResponse |  |  | Y | The response to a Verify operation. |
| VerifyResponsePackage |  | 1 | Y |  |
| ResponseStatus | [ResponseStatus](#_ResponseStatus) | 1 | Y | Returned status for the operation. |
| Return | [ReturnCode](#_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](#_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](#_3.2.48_ScoreType) | 0..1 | N | The score if the biometric information matched. |
| Identity | [BIASIdentity](#_BIASIdentity) | 0..1 | C |  |
| EncounterID | [BIASIDType](#_BIASIDType) | 1 | Y | The identifier of the encounter, if assigned. |
| Token | [TokenType](#_TokenType) | 0..1 | C | A value used to retrieve the results of the Verify request; returned with asynchronous request processing. If set to zero, operation is processed synchronously. If set to a non-zero value, operation is processed asynchronously and Get Verify Results must be used to retrieve the results. |
| TokenValue | string | 1 | Y | A value returned by the implementing system that is used to retrieve the results to an operation at a later time. |
| Expiration | date | 1 | Y | A date and time at which point the token expires and the operation results are no longer guaranteed to be available. |

# 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 Section3 and BIAS messages in Section 4.



**Figure 4. BIAS SOAP Structure**

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

**5.1 Purpose and constraints**

This document defines a SOAP profile describing how the XML elements defined in ISO/IEC 30108 are to be used as the payload of a SOAP message and the rules for structuring and exchanging such messages. Philosophical tenets include:

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

**5.2 Message requirements**

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

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



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

The basic process for using SOAP for BIAS operations is:

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

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

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

**5.3 Handling binary data**

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

* Embedded Base64 encoding
* XOP [XOP]

Use of SOAP with Attachments (SWA) is deprecated.

**5.3.1 Base64 encoding**

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

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

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

However, even an XML\_BIR as defined within [[CBEFF3](#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](#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](#_QueryCapabilities) operation to provide dynamic information regarding BIAS capabilities, options, galleries, etc. that are supported.

**5.5 Identifying operations**

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

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

**5.5.1 Operation name element**

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

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

POST /bias HTTP/1.1

Host: www.acme.com

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

Content-Length: nnnn

SOAPAction: “”

<?xml version=”1.0”?>

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

    <soap:Body>

<tns:CreateSubjectRequest                             xmlns:tns=”http://docs.oasis‐open.org/bias/bias‐1.0/”>

              <tns:GenericRequestParameters>

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

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

                  <tns:BIASOperationName>CreateSubject</tns:BIASOperationName>

               </tns:GenericRequestParameters>

               <tns:Identity>

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

               </tns:Identity>

          </tns:CreateSubjectRequest>

      </soap:Body>

</soap:Envelope>

**5.5.2 WS-Addressing Action**

WS-Addressing [WS-Addr] provides a mechanism for including action information inside any SOAP message.  The information is in the SOAP Header.  The WS-Addressing ‘Action’ element is used to indicate the intent of the message.  The value is a URI/IRI identifying that intent; however, there are no restrictions on the format or specificity of the URI/IRInor a requirement that it can be resolved.  Adoption of this option also requires that the WS-Addressing ‘To’, ‘ReplyTo’, and ‘MessageID’ elements are supplied, as they are mandatory elements in a request-reply message pattern as used within BIAS. Response messages would also need to use WS-Addressing, requiring the ‘To’ (matching the ‘ReplyTo’ element in the request), ‘RelatesTo’ (matching the ‘MessageID’ element in the request), and ‘RelationshipType’ (default value to “wsa:Reply”) elements.

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

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

POST /bias HTTP/1.1

Host: www.acme.com

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

Content-Length: nnnn

SOAPAction: “”

<?xml version=”1.0”?>

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

xmlns:wsa=”http://schemas.xmlsoap.org/ws/2004/08/addressing”>

      <soap:Header>

            <wsa:MessageID>some-ID</wsa:MessageID>

            <wsa:ReplyTo>

                  <wsa:Address>response-URI</wsa:Address>

            </wsa:ReplyTo>

            <wsa:To>destination-URI</wsa:To>

            <wsa:Action>CreateSubject</wsa:Action>

      </soap:Header>

      <soap:Body>

            <tns:CreateSubjectRequest

                              xmlns:tns=”http://docs.oasis‐open.org/bias/bias‐1.0/” />

      </soap:Body>

</soap:Envelope>

**5.6 Security**

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

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

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

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

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

**5.6.2 Data Origin Authentication**

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

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

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

**5.6.3 Message Integrity**

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

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

**5.6.4 Message Confidentiality**

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

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

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

**5.6.5 CBEFF BIR security features**

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

**5.6.6 Security Considerations**

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

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

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

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

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

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

**5.6.7 Security of Stored Data**

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

**5.6.8 Key Management**

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

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

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

**5.8 Tailoring**

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

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

* 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)

# 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.

## 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.

## 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.

# Conformance

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

ISO/IEC 30108 Error! Reference source not found. (Annex A) specifies seven BIAS conformance classes. 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

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

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

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

1. XML Schema

<?xml version="1.0" encoding="UTF-8"?>

<!-- edited with XMLSpy v2016 (x64) (http://www.altova.com) by Karen Marshall (DoC/NIST) -->

<!--

Biometric Identity Assurance Services (BIAS) SOAP Profile Version 1.0

OASIS Standard incorporating Approved Errata 02

06 May 2014

Copyright (c) OASIS Open 2014. All Rights Reserved.

Source: http://docs.oasis-open.org/bias/soap-profile/v1.0/errata02/os/

-->

<!--

Document : BIAS.xml

Created on : December 17, 2009, 9:31 AM by Kevin Mangold, National Institute of Standards and Technology

Last Edit : February 12, 2016 by Karen Marshall, National Institute of Standards and Technology

Description:

This document represents the WSDL definitions of BIAS Profile.

-->

<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"

xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"

xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"

xmlns:xsd="http://www.w3.org/2001/XMLSchema"

xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"

xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"

xmlns:oasis\_cbeff="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"

xmlns:iso\_3166="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"

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

targetNamespace="http://docs.oasis-open.org/bias/ns/bias-1.0/">

<wsdl:types>

<xsd:schema elementFormDefault="qualified" targetNamespace="http://docs.oasis-open.org/bias/ns/bias-1.0/">

<xsd:import namespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/" schemaLocation="cbeff.xsd"/>

<xsd:import namespace="http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/" 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>

<xsd:element name="ProcessingOptions" type="tns:ProcessingOptionsType" minOccurs="1" maxOccurs="1">

<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>

<xsd:documentation>Maps to specific INCITS BIAS elements as required by that specification.</xsd:documentation>

</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>

<xsd:documentation>The data in a service input parameter is invalid.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="BIR\_QUALITY\_ERROR">

<xsd:annotation>

<xsd:documentation>Biometric sample quality is too poor for the service to succeed.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="INVALID\_BIR">

<xsd:annotation>

<xsd:documentation>The input BIR is empty or in an invalid or unrecognized format.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="BIR\_SIGNATURE\_FAILURE">

<xsd:annotation>

<xsd:documentation>The service could not validate the signature, if used, on the input BIR.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="BIR\_DECRYPTION\_FAILURE">

<xsd:annotation>

<xsd:documentation>The service could not decrypt an encrypted input BIR.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="INVALID\_ENCOUNTER\_ID">

<xsd:annotation>

<xsd:documentation>The input encounter ID is empty or in an invalid format.</xsd:documentation>

</xsd:annotation>

</xsd:enumeration>

<xsd:enumeration value="INVALID\_SUBJECT\_ID">

<xsd:annotation>

<xsd:documentation>The input subject ID is empty or in an invalid format.</xsd:documentation>

</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 value="NONEXISTANT\_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="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:BiographicDataItemType" minOccurs="1" maxOccurs="unbounded">

<xsd:annotation>

<xsd:documentation>A single biographic data element.</xsd:documentation>

</xsd:annotation>

</xsd:element>

<xsd:element name="BiographicDataSet" type="tns:BiographicDataSetType" minOccurs="0" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>A set of biographic data information.</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="oasis\_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="oasis\_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: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" >

<xsd:complexType>

<xsd:sequence>

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

</xsd:sequence>

</xsd:complexType>

</xsd:element>

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

<xsd:element name="Rank" type="xsd:integer" minOccurs="1" maxOccurs="1" >

<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" >

<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: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: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 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: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="oasis\_cbeff:BIRInfoType" minOccurs="0" maxOccurs="1"/>

<xsd:element name="BDB\_Info" type="oasis\_cbeff:BDBInfoType" minOccurs="0" maxOccurs="1"/>

<xsd:element name="SB\_Info" type="oasis\_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="oasis\_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: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="oasis\_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="oasis\_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: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>Identifers 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>Identifers 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: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="FingerPosition" type="xsd:string" minOccurs="0" />

<xsd:element name="HandPosition" 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="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="oasis\_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: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" >

<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" >

<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: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="oasis\_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: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="oasis\_cbeff:MultipleTypesType" minOccurs="0"/>

<xsd:element name="BiometricSubType" type="oasis\_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: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:element>

</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>

<xsd:documentation>Specifies a particular algorithm vendor and vender product ID.</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:annotation>

<xsd:documentation>Contains the quality information for the submitted 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="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: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: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: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">

<xsd:sequence>

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

<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: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 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="oasis\_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: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>

<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 presented identity document. </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: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:annotation>

<xsd:documentation>A value used to retrieve the results of an IdentifySubject 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="GetIdentifySubjectResultsResponse">

<xsd:complexType>

<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

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.

</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: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 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.

</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>

<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 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.

</xsd:documentation>

</xsd:annotation>

</xsd:element>

<xsd:element name="ListFilter" type="tns:ListFilterType" minOccurs="0" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>Indicates what biometric information 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="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

with a subject or encounter or a list of encounter ID's

associated with a subject and which contain biometric

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: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 and encounter data exists for

the subject, the operation shall return the list of encounter IDs which contain document data using the Encounter List

Output parameter, and the Document Data Elements output parameter shall be empty. Should not be used in conjunction

with EncounterID.

</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>

<xsd:documentation>

The response to a ListDocumentData request, containing a list of document categories stored for a subject.

In the encounter-centric model, an encounterID may be specified to indicate that only document data categories

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

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.

</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

if the service was 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: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: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="oasis\_cbeff:MultipleTypesType" minOccurs="0" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>The type of biological or behavioral data to retrieve.</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="RetrieveBiometricDataResponse">

<xsd:complexType>

<xsd:annotation>

<xsd:documentation>

The response to a RetrieveBiometricData request,

containing the biometric data associated with a subject ID. In

the encounter-centric model, the biometric data associated with

a specified encounter is returned. If the encounter ID is not

specified in the encounter-centric model, the biometric

information associated with the most recent encounter is returned.

</xsd:documentation>

</xsd:annotation>

<xsd:sequence>

<xsd:element name="RetrieveBiometricDataResponsePackage" 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 biometric 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="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>

<xsd:documentation>Identifies the 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: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: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>

<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 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 document data will be added.</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 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>

<xsd:documentation>Data structure containing the biometric information to be transformed.</xsd:documentation>

</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 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: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: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 deleted; 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="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">

<xsd:annotation>

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

</xsd:annotation>

</xsd:element>

<xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>A value used to retrieve the results of the Delete 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="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>

<xsd:element name="Identity" type="tns:BIASIdentity" minOccurs="0" maxOccurs="1">

<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" minOccurs="1" maxOccurs="1">

<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" minOccurs="0" 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="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:annotation>

<xsd:documentation>A value used to retrieve the results of the Deletion 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="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: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">

<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: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: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:annotation>

<xsd:documentation>A value used to retrieve the results of the Verify 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="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 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: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>

<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="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="1" 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;

returned with successful, synchronous processing.

</xsd:documentation>

</xsd:annotation>

</xsd:element>

<xsd:element name="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">

<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="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 type(s) of information should be returned.</xsd:documentation>

</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>

<xsd:extension base="tns:AggregateResponseTemplate">

<xsd:sequence>

<xsd:element name="Token" type="tns:TokenType" minOccurs="0" 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="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" minOccurs="1" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>Includes either 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 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: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="Token" type="tns:TokenType" minOccurs="0" maxOccurs="1">

<xsd:annotation>

<xsd:documentation>A value used to retrieve the results of the Verify 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="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\_v1">

<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 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\_v1HttpBinding" type="tns:BIAS\_v1">

<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: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: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\_v1">

<wsdl:port name="BIAS\_v1HttpBindingPort" binding="tns:BIAS\_v1HttpBinding">

<soap:address location="http://your\_URI/IRI\_to\_your\_BIAS\_implementation"/>

</wsdl:port>

</wsdl:service>

</wsdl:definitions>

1. BIAS Patron format specification

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

* 1. Patron

Organization for the Advancement of Structured Information Standards (OASIS)

* 1. Patron identifier

**82 (0052 Hex)**.

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

* 1. Patron format name

**OASIS BIAS CBEFF XML Patron Format**

* 1. Patron format identifier

**01 (0001 Hex)**.

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

* 1. ASN.1 object identifier for this patron format

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

* 1. Domain of use

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

* 1. Version identifier

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

* 1. CBEFF version

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

* 1. General

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

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

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

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

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

* 1. Specification

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

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

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

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

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

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

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

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

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

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

* 1. Element <BIR>
     1. Syntax

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

1. an optional **<Version>** element (see B.12);
2. an optional **<CBEFFVersion>** element (see B.13);
3. zero or more application-specific elements;
4. a mandatory **<BIRInfo>** element (see B.14);
5. an optional **<BDBInfo>** element (see B.15);
6. an optional **<SBInfo>** element (see B.16);
7. zero or more **<BIR>** elements (see B.11);
8. either an optional **<BDB>** element that shall contain a valid representation of an octect string (see B.18), or an optional **<bdbX>** element that shall contain a valid XML string;
9. an optional **<SB>** element – the content of this element shall be a valid representation of an octet string.

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

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

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

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

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

* + 1. Semantics

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

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

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

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

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

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

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

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

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

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

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

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

* 1. Element <Version>
     1. Syntax

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

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

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

* + 1. Semantics

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

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

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

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

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

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

* 1. Element <CBEFFVersion>
     1. Syntax

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

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

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

**B.13.2 Semantics**

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

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

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

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

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

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

* 1. Element <BIRInfo>
     1. Syntax

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

a) an optional **<Creator>** element – the content of this element shall be a string of ISO/IEC 10646 characters;

b) an optional **<Index>** element – the content of this element shall be a valid representation of a universally unique identifier (see B.20), and shall not inherit its value from any other level BIR;

c) an optional **<Payload>** element – the content of this element shall be a valid representation of an octet string, and shall not inherit its value from any other level BIR.

d) a required **<Integrity>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.1;

e) an optional **<CreationDate>** element – the value of this element shall be a valid representation of a date and time of the day (see B.19);

f) an optional **<NotValidBefore>** element – the value of this element shall be a valid representation of a date and time of the day;

g) an optional **<NotValidAfter>** element – the value of this element shall be a valid representation of a date and time of the day.

* + 1. Semantics

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

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

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

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

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

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

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

**Table B.1 – BIR information**

| **CBEFF data element name** | **XML element** | **Supported abstract values and encodings** | **Reference** |
| --- | --- | --- | --- |
| CBEFF\_BIR\_creator | <Creator> | All ISO/IEC 10646 character strings are supported.  The character string shall be encoded as the string itself. |  |
| CBEFF\_BIR\_index | <Index> | All well-formed UUIDs are supported.  The UUIDs shall be encoded as specified in B.20.  Shall not inherit its value from any other BIR level. |  |
| CBEFF\_BIR\_payload | <Payload> | All octet strings are supported.  The octet strings shall be encoded as specified in B.18.  Shall not inherit its value from any other BIR level. |  |
| CBEFF\_BIR\_integrity\_options | <Integrity> | The following abstract values are supported.  The abstract values shall be encoded as shown below.  NO INTEGRITY:  "**false**"  INTEGRITY: "**true**" |  |
| CBEFF\_BIR\_creation\_date | <CreationDate> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |
| CBEFF\_BIR\_validity\_period (lower end) | <NotValidBefore> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |
| CBEFF\_BIR\_validity\_period (upper end) | <NotValidAfter> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |

* 1. Element <BDBInfo>
     1. Syntax

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

1. an optional **<ChallengeResponse>** element – the content of this element shall be a valid representation of an octet string (see B.18);
2. an optional **<Index>** element – the content of this element shall be a valid representation of a universally unique identifier (see B.20).
3. an optional **<FormatOwner>** element – the value of this element shall be a valid representation of an integer in the range 1 to 65535 (see B.17);
4. an optional **<FormatType>** element – the value of this element shall be a valid representation of an integer in the range 1 to 65535;
5. an optional **<Encryption>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.2;
6. an optional **<CreationDate>** element – the value of this element shall be a valid representation of a date and time of the day (see B.19);
7. an optional **<NotValidBefore>** element – the value of this element shall be a valid representation of a date and time of the day;
8. an optional **<NotValidAfter>** element – the value of this element shall be a valid representation of a date and time of the day;
9. an optional **<Type>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.2;
10. an optional **<Subtype>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.2;
11. an optional **<Level>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.2;
12. an optional **<ProductOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
13. an optional **<ProductType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
14. an optional **<CaptureDeviceOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
15. an optional **<CaptureDeviceType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
16. an optional **<FeatureExtractionAlgorithmOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
17. an optional **<FeatureExtractionAlgorithmType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
18. an optional **<ComparisonAlgorithmOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
19. an optional **<ComparisonAlgorithmType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
20. an optional **<QualityAlgorithmOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
21. an optional **<QualityAlgorithmType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
22. an optional **<CompressionAlgorithmOwner>** element – the value of this element shall be a valid representation of an integer in the range 1..65535 (see B.17);
23. an optional **<CompressionAlgorithmType>** element – the value of this element shall be a valid representation of an integer in the range 1..65535;
24. an optional **<Purpose>** element – the value of this element shall be one of the character strings in the third cell of the corresponding row of Table B.2;
25. an optional **<Quality>** element – the value of this element shall be a valid representation of an integer in the range –2..100 (see B.17), as specified in the third cell of the corresponding row of Table B.2.

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

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

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

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

* + 1. Semantics

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

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

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

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

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

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

**Table B.2 – BDB information**

| **CBEFF data element name** | **XML element** | **Supported abstract values and encodings** | **Reference** |
| --- | --- | --- | --- |
| CBEFF\_BDB\_format\_owner | <FormatOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_format\_type | <FormatType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_encryption\_options | <Encryption> | The following abstract values are supported.  The abstract values shall be encoded as shown below.  NO ENCRYPTION:  "**false**"  ENCRYPTION:  "**true**" |  |
| CBEFF\_BDB\_creation\_date | <CreationDate> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |
| CBEFF\_BDB\_validity\_period (lower end) | <NotValidBefore> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |
| CBEFF\_BDB\_challenge\_response | <ChallengeResponse> | All octet strings are supported.  The octet strings shall be encoded as specified in B.18.  Shall appear only in BIRs that have a BDB. |  |
| CBEFF\_BDB\_index | <Index> | All well-formed UUIDs are supported.  The UUIDs shall be encoded as specified in B.20  Shall appear only in BIRs that have a BDB. |  |
| CBEFF\_BDB\_validity\_period (upper end) | <NotValidAfter> | All date and time-of-the-day abstract values permitted by CBEFF are supported.  The abstract values shall be encoded as specified in B.19. |  |
| CBEFF\_BDB\_biometric\_type | <Type> | The following abstract values and all their unordered combinations are supported.  A single abstract value shall be encoded as the corresponding string shown below. A combination of two or more abstract values shall be encoded as the concatenation of the corresponding strings, using a single space as separator.  SCENT: "**Scent**"  DNA: "**DNA**"  EAR: "**Ear**"  FACE: "**Face**"  FINGER: "**Finger**"  FOOT: "**Foot**"  VEIN: "**Vein**"  HAND GEOMETRY: "**HandGeometry**"  IRIS: "**Iris**"  RETINA: "**Retina**"  VOICE: "**Voice**"  GAIT: "**Gait**"  KEYSTROKE: "**Keystroke**"  LIP MOVEMENT: "**LipMovement**"  SIGNATURE OR SIGN: "**SignatureSign**" |  |
| CBEFF\_BDB\_biometric\_subtype | <Subtype> | The following abstract values are supported.  The abstract values shall be encoded as shown below. A combination of two or more abstract values shall be encoded as the concatenation of the corresponding strings, using a single space as separator.  LEFT:  "**Left**"  RIGHT:  "**Right**"  THUMB:  "**Thumb**"  INDEX FINGER:  "**IndexFinger**"  MIDDLE FINGER:  "**MiddleFinger**"  RING FINGER:  "**RingFinger**"  LITTLE FINGER:  "**LittleFinger**" |  |
| CBEFF\_BDB\_processed\_level | <Level> | The following abstract values are supported.  The abstract values shall be encoded as shown below.  RAW:  "**Raw**"  INTERMEDIATE:  "**Intermediate**"  PROCESSED: "**Processed**" |  |
| CBEFF\_BDB\_product\_owner | <ProductOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_product\_type | <ProductType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_capture\_device\_owner | <CaptureDeviceOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_capture\_device\_type | <CaptureDeviceType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_feature\_extraction\_algorithm\_owner | <FeatureExtractionAlgorithmOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_feature\_extraction\_algorithm\_type | <FeatureExtractionAlgorithmType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_comparison\_algorithm\_owner | <ComparisonAlgorithmOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_comparison\_algorithm\_type | <ComparisonAlgorithmType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_quality\_algorithm\_owner | <QualityAlgorithmOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_quality\_algorithm\_type | <QualityAlgorithmType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_compression\_algorithm\_owner | <CompressionAlgorithmOwner> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_compression\_algorithm\_type | <CompressionAlgorithmType> | All integers in the range 1 to 65535 are supported.  The integers shall be encoded as specified in B.17. |  |
| CBEFF\_BDB\_purpose | <Purpose> | The following abstract values are supported.  The abstract values shall be encoded as shown below.  VERIFY: "**Verify**"  IDENTIFY: "**Identify**"  ENROLL: "**Enroll**"  ENROLL FOR VERIFICATION ONLY: "**EnrollVerify**"  ENROLL FOR IDENTIFICATION ONLY: "**EnrollIdentify**"  AUDIT: "**Audit**" |  |
| CBEFF\_BDB\_quality | <Quality> | The following abstract values are supported.  The integers shall be encoded as specified in B.17. The other abstract values shall be encoded as shown below.  INTEGER  QUALITY NOT SUPPORTED BY BDB CREATOR:  "**-2**"  QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET:  "**-1**" |  |

* 1. Element <SBInfo>
     1. Syntax

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

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

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

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

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

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

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

* + 1. Semantics

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

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

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

**Table B.3 – SB information**

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

* 1. Representation of Integers

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

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

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

* 1. Representation of Octet Strings

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

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

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

c) DIGIT ZERO to DIGIT NINE;

d) PLUS SIGN;

e) SOLIDUS;

f) EQUALS SIGN.

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

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

* 1. Representation of Date and Time of the Day

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

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

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

b) the hyphen character “-“

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

d) the hyphen character “-“

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

f) the letter "**T**";

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

h) the colon character “:”

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

j) the colon character “:”

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

l) the letter "**Z**".

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

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

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

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

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

* 1. Representation of Universally Unique Identifiers

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

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

a) DIGIT ZERO to DIGIT NINE ("**0**" to "**9**"), each representing a hexadecimal digit 0 through 9;

b) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER F ("**A**" to "**F**"), each representing a hexadecimal digit A through F;

c) LATIN SMALL LETTER A to LATIN SMALL LETTER F ("**a**" to "**f**"), each representing a hexadecimal digit A through F; and

d) HYPHEN-MINUS ("**-**").

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

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

* 1. Patron format conformance statement
     1. Identifying information

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

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

| **CBEFF data element name** | **Mandatory/ optional** | **Patron format field name** | **Abstract values specified?** | **Encodings specified?** |
| --- | --- | --- | --- | --- |
| CBEFF\_BDB\_format\_owner | Mandatory (specified or inherited) if a BDB is present | **<FormatOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_format\_type | Mandatory (specified or inherited) if a BDB is present | **<FormatType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_encryption\_options | Mandatory (specified or inherited) if a BDB is present | **<Encryption>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BIR\_integrity\_options | Mandatory | **<Integrity>** child of **<BIRInfo>** | Yes | Yes |
| CBEFF\_BDB\_subheader\_count | Mandatory | implied in the number of occurrences of the child **<BIR>** element | No (implied) | No (implied) |
| CBEFF\_BDB\_biometric\_type | Optional | **<Type>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_biometric\_subtype | Optional | **<Subtype>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_challenge\_response | Optional | **<ChallengeResponse>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_creation\_date | Optional | **<CreationDate>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_index | Optional | **<Index>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_product\_owner | Optional | **<ProductOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_product\_type | Optional | **<ProductType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_capture\_device\_owner | Optional | **<CaptureDeviceOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_capture\_device\_type | Optional | **<CaptureDeviceType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_feature\_extraction\_algorithm\_owner | Optional | **<FeatureExtractionAlgorithm Owner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_feature\_extraction\_algorithm\_type | Optional | **<FeatureExtractionAlgorithm Type>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_comparison\_algorithm\_owner | Optional | **<ComparisonAlgorithmOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_comparison\_algorithm\_type | Optional | **<ComparisonAlgorithmType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_quality\_algorithm\_owner | Optional | **<QualityAlgorithmOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_quality\_algorithm\_type | Optional | **<QualityAlgorithmType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_compression\_algorithm\_owner | Optional | **<CompressionAlgorithmOwner>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_compression\_algorithm\_type | Optional | **<CompressionAlgorithmType>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_processed\_level | Optional | **<Level>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_purpose | Optional | **<Purpose>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_quality | Optional | **<Quality>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BDB\_validity\_period | Optional | **<NotValidBefore>** and <**NotValidAfter>** children of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BIR\_creation\_date | Optional | **<CreationDate>** child of **<BIRInfo>** | Yes | Yes |
| CBEFF\_BIR\_creator | Optional | **<Creator>** child of **<BIRInfo>** | Yes | Yes |
| CBEFF\_BIR\_index | Optional | **<Index>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_BIR\_patron\_format\_owner | N/A |  | No | No |
| CBEFF\_BIR\_patron\_format\_type | N/A |  | No | No |
| CBEFF\_BIR\_payload | Optional | **<Payload>** child of **<BDBInfo>** | Yes | Yes |
| CBEFF\_SB\_format\_owner | Optional | **<FormatOwner>** child of **<SBInfo>** | Yes | Yes |
| CBEFF\_SB\_format\_type | Optional | **<FormatType>** child of **<sb-info>** | Yes | Yes |
| CBEFF\_BIR\_validity\_period | Optional | **<NotValidBefore>** and **<NotValidAfter>** attributes of **<BIRInfo>** | Yes | Yes |
| patron\_header\_version | Optional | **<Major>** and **<Minor>** children of **<Version>** | Yes | Yes |
| CBEFF\_version | Optional | **<Major>** and **<Minor>** children of **<CBEFFVersion>** | Yes | Yes |
| BDB | Optional | **<BDB>** | Yes | Yes |
| SB | Optional | **<SB>** | Yes | Yes |

* 1. XML schema of the BIAS patron format

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

**<xs:schema**

**xmlns:xs="http://www.w3.org/2001/XMLSchema"**

**xmlns=" http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"**

**targetNamespace=" http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/"**

**elementFormDefault="qualified">**

**<xs:element name="BIR" type="BIRType"/>**

**<xs:complexType name="BIRType">**

**<xs:sequence>**

**<xs:element name="Version" type="VersionType" minOccurs="0"/>**

**<xs:element name="CBEFFVersion" type="VersionType" minOccurs="0"/>**

**<xs:any namespace="##other" processContents="skip" minOccurs="0" maxOccurs="unbounded"/>**

**<xs:element name="BIRInfo" type="BIRInfoType"/>**

**<xs:element name="BDBInfo" type="BDBInfoType" minOccurs="0"/>**

**<xs:element name="SBInfo" type="SBInfoType" minOccurs="0"/>**

**<xs:element name="BIR" type="BIRType" minOccurs="0" maxOccurs="unbounded"/>**

**<xs:element name="BDB" type="xs:base64Binary" minOccurs="0"/>**

**<xs:element name="SB" type="xs:base64Binary" minOccurs="0"/>**

**</xs:sequence>**

**</xs:complexType>**

**<xs:complexType name="VersionType">**

**<xs:sequence>**

**<xs:element name="Major" type="xs:nonNegativeInteger" />**

**<xs:element name="Minor" type="xs:nonNegativeInteger" />**

**</xs:sequence>**

**</xs:complexType>**

**<xs:complexType name="BIRInfoType">**

**<xs:sequence>**

**<xs:element name="Creator" type="xs:string" minOccurs="0"/>**

**<xs:element name="Index" type="UUIDType" minOccurs="0"/>**

**<xs:element name="Payload" type="xs:base64Binary" minOccurs="0"/>**

**<xs:element name="Integrity" type="xs:boolean" />**

**<xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />**

**<xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />**

**<xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />**

**</xs:sequence>**

**</xs:complexType>**

**<xs:complexType name="BDBInfoType">**

**<xs:sequence>**

**<xs:element name="ChallengeResponse" type="xs:base64Binary" minOccurs="0"/>**

**<xs:element name="Index" type="UUIDType" minOccurs="0"/>**

**<xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="Encryption" type="xs:boolean" minOccurs="0" />**

**<xs:element name="CreationDate" type="xs:dateTime" minOccurs="0" />**

**<xs:element name="NotValidBefore" type="xs:dateTime" minOccurs="0" />**

**<xs:element name="NotValidAfter" type="xs:dateTime" minOccurs="0" />**

**<xs:element name="Type" type="MultipleTypesType" minOccurs="0" />**

**<xs:element name="Subtype" type="SubtypeType" minOccurs="0" />**

**<xs:element name="Level" type="ProcessedLevelType" minOccurs="0" />**

**<xs:element name="ProductOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="ProductType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="CaptureDeviceOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="CaptureDeviceType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="FeatureAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="FeatureAlgorithmType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="ComparisonAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="ComparisonAlgorithmType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="QualityAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="QualityAlgorithmType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="CompressionAlgorithmOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="CompressionAlgorithmType" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="Purpose" type="PurposeType" minOccurs="0" />**

**<xs:element name="Quality" type="QualityType" minOccurs="0" />**

**</xs:sequence>**

**</xs:complexType>**

**<xs:complexType name="SBInfoType">**

**<xs:sequence>**

**<xs:element name="FormatOwner" type="xs:positiveInteger" minOccurs="0" />**

**<xs:element name="FormatType" type="xs:positiveInteger" minOccurs="0" />**

**</xs:sequence>**

**</xs:complexType>**

**<xs:simpleType name="QualityType">**

**<xs:restriction base="xs:integer">**

**<xs:minInclusive value="-2"/>**

**<xs:maxInclusive value="100"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="SingleTypeType">**

**<xs:restriction base="xs:string">**

**<xs:enumeration value="Scent"/>**

**<xs:enumeration value="DNA"/>**

**<xs:enumeration value="Ear "/>**

**<xs:enumeration value="Face"/>**

**<xs:enumeration value="Finger"/>**

**<xs:enumeration value="Foot"/>**

**<xs:enumeration value="HandGeometry"/>**

**<xs:enumeration value="Vein"/>**

**<xs:enumeration value="Iris"/>**

**<xs:enumeration value="Retina"/>**

**<xs:enumeration value="Voice"/>**

**<xs:enumeration value="Gait"/>**

**<xs:enumeration value="Keystroke"/>**

**<xs:enumeration value="LipMovement"/>**

**<xs:enumeration value="SignatureSign"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="MultipleTypesType">**

**<xs:list itemType="SingleTypeType"/>**

**</xs:simpleType>**

**<xs:simpleType name="SingleAnySubtypeType">**

**<xs:restriction base="xs:string">**

**<xs:enumeration value="Left"/>**

**<xs:enumeration value="Right"/>**

**<xs:enumeration value="Thumb"/>**

**<xs:enumeration value="IndexFinger"/>**

**<xs:enumeration value="MiddleFinger"/>**

**<xs:enumeration value="RingFinger"/>**

**<xs:enumeration value="LittleFinger"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="SingleVeinOnlySubtypeType">**

**<xs:restriction base="xs:string">**

**<xs:enumeration value="LeftVein"/>**

**<xs:enumeration value="RightVein"/>**

**<xs:enumeration value="Palm"/>**

**<xs:enumeration value="BackOfHand"/>**

**<xs:enumeration value="Wrist"/>**

**<xs:enumeration value="Reserved1"/>**

**<xs:enumeration value="Reserved2"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="MultipleAnySubtypesType">**

**<xs:list itemType="SingleAnySubtypeType"/>**

**</xs:simpleType>**

**<xs:simpleType name="MultipleVeinOnlySubtypesType">**

**<xs:list itemType="SingleVeinOnlySubtypeType"/>**

**</xs:simpleType>**

**<xs:simpleType name="SubtypeType">**

**<xs:union memberTypes="MultipleAnySubtypesType MultipleVeinOnlySubtypesType"/>**

**</xs:simpleType>**

**<xs:simpleType name="ProcessedLevelType">**

**<xs:restriction base="xs:string">**

**<xs:enumeration value="Raw"/>**

**<xs:enumeration value="Intermediate"/>**

**<xs:enumeration value="Processed"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="PurposeType">**

**<xs:restriction base="xs:string">**

**<xs:enumeration value="Verify"/>**

**<xs:enumeration value="Identify"/>**

**<xs:enumeration value="Enroll"/>**

**<xs:enumeration value="EnrollVerify"/>**

**<xs:enumeration value="EnrollIdentify"/>**

**<xs:enumeration value="Audit"/>**

**</xs:restriction>**

**</xs:simpleType>**

**<xs:simpleType name="UUIDType">**

**<xs:restriction base="xs:string">**

**<xs:pattern value="[a-fA-F0-9]{8}\-([a-fA-F0-9]{4}\-){3}[a-fA-F0-9]{12}"/>**

**</xs:restriction>**

**</xs:simpleType>**

**</xs:schema>**

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

* 1. Sample BIR encoding

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

**<?xml version="1.0" encoding="utf-8"?>**

**<BIR xmlns="** **http://docs.oasis-open.org/bias/ns/biaspatronformat-1.0/">**

**<Version>**

**<Major>1</Major>**

**<Minor>0</Minor>**

**</Version>**

**<CBEFFVersion>**

**<Major>2</Major>**

**<Minor>0</Minor>**

**</CBEFFVersion>**

**<BIRInfo>**

**<Creator>ABCDE</Creator>**

**<Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>**

**<Payload>a2V2aW4ubWFuZ29sZEBuaXN0Lmdvdg==</Payload>**

**<Integrity>true</Integrity>**

**<CreationDate>2004-03-02T15:03:15Z</CreationDate>**

**<NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>**

**<NotValidAfter>2004-03-03T15:00:00Z</NotValidAfter>**

**</BIRInfo>**

**<BDBInfo>**

**<ChallengeResponse>VmlzaXQgaHR0cDovL2J3cy5uaXN0LmdvdiBmb3Igc29tZSBhd2Vzb21lIGJpb21ldHJpY3Mvd2ViIHNlcnZpY2UgcHJvamVjdHMh</ChallengeResponse>**

**<Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>**

**<FormatOwner>51</FormatOwner>**

**<FormatType>88</FormatType>**

**<Encryption>true</Encryption>**

**<CreationDate>2004-03-02T15:00:00Z</CreationDate>**

**<NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>**

**<NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>**

**<Type>Iris</Type>**

**<Subtype>Left</Subtype>**

**<Level>Processed</Level>**

**<ProductOwner>16</ProductOwner>**

**<ProductType>2</ProductType>**

**<Purpose>Verify</Purpose>**

**<Quality>100</Quality>**

**</BDBInfo>**

**<SBInfo>**

**<FormatOwner>51</FormatOwner>**

**<FormatType>99</FormatType>**

**</SBInfo>**

**<BDB>a2V2aW4ubWFuZ29sZEBuaXN0Lmdvdg==</BDB>**

**<SB>TmF0aW9uYWwgSW5zdGl0dXRlIG9mIFN0YW5kYXJkcyBhbmQgVGVjaG5vbG9neQ==</SB>**

**</BIR>**

1. Use Cases (non-normative)

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

* Verification (synchronous/aggregate)
* Verification (asynchronous/aggregate)
* Verification (primitive)
* Identification (primitive)
* Enrollment (aggregate)
* Enrollment (primitive)
  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.



* 1. 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.



* 1. 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.



* 1. 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.



* 1. 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.).



* 1. 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.



1. Samples (non-normative)
   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](http://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](http://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](http://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>

* 1. 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](http://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: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>

* 1. 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](http://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>

1. 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.

1. Revision History

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