
**Information technology —
Conformance testing for the biometric
application programming interface
(BioAPI) —**

**Part 1:
Methods and procedures**

*Technologies de l'information — Test de conformité pour l'interface
de programmation d'applications biométriques (BioAPI) —*

Partie 1: Méthodes et procédures

[ISO/IEC 24709-1:2017](https://standards.iso.org/iso/24709-1:2017)

<https://standards.iso.org/iso/24709-1:2017>



iTeh Standards
(<https://standards.iteh.ai>)
Document Preview

[ISO/IEC 24709-1:2017](https://standards.iteh.ai/catalog/standards/iso/a1a58e7f-d5f1-4b43-a2ae-ba6f078030f5/iso-iec-24709-1-2017)

<https://standards.iteh.ai/catalog/standards/iso/a1a58e7f-d5f1-4b43-a2ae-ba6f078030f5/iso-iec-24709-1-2017>



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword	xix
Introduction	xx
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Abbreviated terms	3
5 Conformance	3
6 Conformance testing methodology	4
6.1 General	4
6.1.1 Implementation under test	4
6.1.2 Test method	5
6.1.3 Standard BioAPI components and standard BioAPI interfaces	6
6.1.4 Physical architectures	7
6.2 Conformance testing models	8
6.3 Abstract test engine	13
7 General properties of the assertion language	14
7.1 General	14
7.2 Variables	17
7.3 Built-in variables	18
7.4 Representation of integers	18
7.5 Representation of Booleans	19
7.6 Representation of universally unique identifiers (UUIDs)	19
7.7 Representation of octet strings	19
7.8 XML documents	20
8 Elements of the assertion language	20
8.1 Element <package>	20
8.1.1 Syntax	20
8.1.2 Semantics	20
8.1.3 Example (non-normative)	21
8.2 Element <assertion> (child of <package>)	21
8.2.1 Syntax	21
8.2.2 Semantics	22
8.2.3 Example (non-normative)	22
8.3 Element <input> (child of <assertion>)	23
8.3.1 Syntax	23
8.3.2 Semantics	23
8.3.3 Example (non-normative)	23
8.4 Element <invoke> (child of <assertion>)	23
8.4.1 Syntax	23
8.4.2 Semantics	24
8.4.3 Example (non-normative)	24
8.5 Element <input> (child of <invoke>)	24
8.5.1 Syntax	24
8.5.2 Semantics	25
8.5.3 Example (non-normative)	25
8.6 Element <output> (child of <invoke>)	25
8.6.1 Syntax	25
8.6.2 Semantics	26
8.7 Element <return> (child of <invoke>)	26
8.7.1 Syntax	26
8.7.2 Semantics	27

8.8	Element <bind> (child of <assertion>)	27
8.8.1	Syntax	27
8.8.2	Semantics	27
8.9	Element <activity> (child of <package>)	28
8.9.1	Syntax	28
8.9.2	Semantics	29
8.9.3	Example (non-normative)	32
8.10	Element <input> (child of <activity>)	35
8.10.1	Syntax	35
8.10.2	Semantics	35
8.11	Element <output> (child of <activity>)	35
8.11.1	Syntax	35
8.11.2	Semantics	35
8.12	Element <set>	36
8.12.1	Syntax	36
8.12.2	Semantics	36
8.13	Element <add>	37
8.13.1	Syntax	37
8.13.2	Semantics	37
8.14	Element <subtract>	38
8.14.1	Syntax	38
8.14.2	Semantics	38
8.15	Element <invoke> (child of <activity>)	39
8.15.1	Syntax	39
8.15.2	Semantics	39
8.15.3	Example (non-normative)	42
8.16	Element <only_if>	42
8.16.1	Syntax	42
8.16.2	Semantics	43
8.17	Element <wait_until>	43
8.17.1	Syntax	43
8.17.2	Semantics	44
8.18	Element <assert_condition>	45
8.18.1	Syntax	45
8.18.2	Semantics	45
8.19	Element <and>	46
8.19.1	Syntax	46
8.19.2	Semantics	46
8.20	Element <or>	47
8.20.1	Syntax	47
8.20.2	Semantics	47
8.21	Element <xor>	47
8.21.1	Syntax	47
8.21.2	Semantics	48
8.22	Element <not>	48
8.22.1	Syntax	48
8.22.2	Semantics	48
8.23	Element <equal_to>	49
8.23.1	Syntax	49
8.23.2	Semantics	49
8.24	Element <not_equal_to>	50
8.24.1	Syntax	50
8.24.2	Semantics	50
8.25	Element <greater_than>	50
8.25.1	Syntax	50
8.25.2	Semantics	50
8.26	Element <greater_than_or_equal_to>	50
8.26.1	Syntax	50

8.26.2	Semantics	50
8.27	Element <less_than>	50
8.27.1	Syntax	50
8.27.2	Semantics	50
8.28	Element <less_than_or_equal_to>	51
8.28.1	Syntax	51
8.28.2	Semantics	51
8.29	Element <same_as>	51
8.29.1	Syntax	51
8.29.2	Semantics	51
8.30	Element <not_same_as>	52
8.30.1	Syntax	52
8.30.2	Semantics	52
8.31	Element <existing>	52
8.31.1	Syntax	52
8.31.2	Semantics	52
8.32	Element <not_existing>	53
8.32.1	Syntax	53
8.32.2	Semantics	53
9	Standard interface functions	53
9.1	General	53
9.2	Parameter groups	60
9.2.1	General	60
9.2.2	Parameter group "Biometric type"	60
9.2.3	Parameter group "Operations"	62
9.2.4	Parameter group "Options"	64
9.2.5	Parameter group "Events"	66
9.2.6	Parameter group "Biometric data type"	67
9.2.7	Parameter group "Biometric subtype"	68
9.2.8	Parameter group "Date"	69
9.2.9	Parameter group "Date and time"	70
9.2.10	Parameter group "Framework schema"	71
9.2.11	Parameter group "BSP schema"	72
9.2.12	Parameter group "BFP schema"	76
9.2.13	Parameter group "Unit schema"	78
9.2.14	Parameter group "BIR header"	81
9.2.15	Parameter group "BIR"	84
9.2.16	Parameter group "Input BIR"	85
9.2.17	Parameter group "Identify population"	86
9.2.18	Parameter group "Candidate"	88
9.2.19	Parameter group "GUI state"	90
9.2.20	Parameter group "Access type"	90
9.2.21	Parameter group "GUI Event Subscription"	91
9.2.22	Parameter group "Key Information"	92
9.2.23	Parameter group "Key Transport"	93
9.2.24	Parameter group "MAC Algorithm Information"	93
9.2.25	Parameter group "Security Options Mask"	94
9.2.26	Parameter group "Security Profile"	95
9.2.27	Parameter group "ACBio parameters"	97
9.2.28	Parameter group "BFP List Element parameters"	98
9.2.29	Parameter group "Biometric subtype mask"	98
9.2.30	Parameter group "ENCRYPTION INFO"	100
9.3	BioAPI_Init	101
9.3.1	Function invocation scheme	101
9.3.2	Constraints on the parameters	101
9.3.3	Function invocation input	101
9.3.4	Function invocation output	101
9.3.5	Bound activity invocation input	101

9.3.6	Bound activity invocation output.....	102
9.4	BioAPI_Terminate.....	102
9.4.1	Function invocation scheme.....	102
9.4.2	Constraints on the parameters.....	102
9.4.3	Function invocation input.....	102
9.4.4	Function invocation output.....	102
9.4.5	Bound activity invocation input.....	102
9.4.6	Bound activity invocation output.....	102
9.5	BioAPI_GetFrameworkInfo.....	103
9.5.1	Function invocation scheme.....	103
9.5.2	Constraints on the parameters.....	103
9.5.3	Function invocation input.....	103
9.5.4	Function invocation output.....	103
9.5.5	Bound activity invocation input.....	104
9.5.6	Bound activity invocation output.....	104
9.6	BioAPI_EnumBSPs.....	104
9.6.1	Function invocation scheme.....	104
9.6.2	Constraints on the parameters.....	105
9.6.3	Function invocation input.....	105
9.6.4	Function invocation output.....	105
9.6.5	Bound activity invocation input.....	105
9.6.6	Bound activity invocation output.....	106
9.7	BioAPI_BSPLoad.....	106
9.7.1	Function invocation scheme.....	106
9.7.2	Constraints on the parameters.....	106
9.7.3	Function invocation input.....	106
9.7.4	Function invocation output.....	107
9.7.5	Bound activity invocation input.....	107
9.7.6	Bound activity invocation output.....	107
9.8	BioAPI_BSPUnload.....	107
9.8.1	Function invocation scheme.....	107
9.8.2	Constraints on the parameters.....	108
9.8.3	Function invocation input.....	108
9.8.4	Function invocation output.....	108
9.8.5	Bound activity invocation input.....	108
9.8.6	Bound activity invocation output.....	109
9.9	BioAPI_BSPAttach.....	109
9.9.1	Function invocation scheme.....	109
9.9.2	Constraints on the parameters.....	109
9.9.3	Function invocation input.....	110
9.9.4	Function invocation output.....	110
9.9.5	Bound activity invocation input.....	110
9.9.6	Bound activity invocation output.....	111
9.10	BioAPI_BSPDetach.....	111
9.10.1	Function invocation scheme.....	111
9.10.2	Constraints on the parameters.....	111
9.10.3	Function invocation input.....	111
9.10.4	Function invocation output.....	112
9.10.5	Bound activity invocation input.....	112
9.10.6	Bound activity invocation output.....	112
9.11	BioAPI_QueryUnits.....	112
9.11.1	Function invocation scheme.....	112
9.11.2	Constraints on the parameters.....	113
9.11.3	Function invocation input.....	113
9.11.4	Function invocation output.....	114
9.11.5	Bound activity invocation input.....	114
9.11.6	Bound activity invocation output.....	114
9.12	BioAPI_EnumBFPs.....	114

9.12.1	Function invocation scheme	114
9.12.2	Constraints on the parameters	115
9.12.3	Function invocation input	115
9.12.4	Function invocation output	116
9.12.5	Bound activity invocation input	116
9.12.6	Bound activity invocation output	116
9.13	BioAPI_QueryBFPs	116
9.13.1	Function invocation scheme	116
9.13.2	Constraints on the parameters	117
9.13.3	Function invocation input	117
9.13.4	Function invocation output	118
9.13.5	Bound activity invocation input	118
9.13.6	Bound activity invocation output	118
9.14	BioAPI_ControlUnit	118
9.14.1	Function invocation scheme	118
9.14.2	Constraints on the parameters	119
9.14.3	Function invocation input	119
9.14.4	Function invocation output	119
9.14.5	Bound activity invocation input	120
9.14.6	Bound activity invocation output	120
9.15	BioAPI_LinkToEndpoint	120
9.15.1	Function invocation scheme	120
9.15.2	Constraints on the parameter	120
9.15.3	Function invocation input	121
9.15.4	Function invocation output	121
9.15.5	Bound activity invocation input	121
9.15.6	Bound activity invocation output	121
9.16	BioAPI_UnlinkFromEndpoint	121
9.16.1	Function invocation scheme	121
9.16.2	Constraints on the parameter	121
9.16.3	Function invocation input	122
9.16.4	Function invocation output	122
9.16.5	Bound activity invocation input	122
9.16.6	Bound activity invocation output	122
9.17	BioAPI_EnumFrameworks	122
9.17.1	Function invocation scheme	122
9.17.2	Constraints on the parameters	123
9.17.3	Function invocation input	123
9.17.4	Function invocation output	123
9.17.5	Bound activity invocation input	123
9.17.6	Bound activity invocation output	123
9.18	BioAPI_FreeBIRHandle	124
9.18.1	Function invocation scheme	124
9.18.2	Constraints on the parameters	124
9.18.3	Function invocation input	124
9.18.4	Function invocation output	124
9.18.5	Bound activity invocation input	124
9.18.6	Bound activity invocation output	125
9.19	BioAPI_GetBIRFromHandle	125
9.19.1	Function invocation scheme	125
9.19.2	Constraints on the parameters	125
9.19.3	Function invocation input	126
9.19.4	Function invocation output	126
9.19.5	Bound activity invocation input	126
9.19.6	Bound activity invocation output	126
9.20	BioAPI_GetHeaderFromHandle	126
9.20.1	Function invocation scheme	126
9.20.2	Constraints on the parameters	127