

IDL_Stub_Behavior

Fri Jan 26 16:30:46 2001

I

Test Suite Overview

Test Suite Structure			
Suite Name : IDL_Stub_Behavior Standards Ref : PICS Ref : PIXIT Ref : Test Method(s) : Comments : IDL stub behavior tests.			
Test Group Reference	Selection Ref	Test Group Objective	Page Nr
Substitutability/		To verify the semantics of passing a value instance as a parameter that is declared to be an interface type or a value type (Ref: 5.2.5/CORBA2.3–Spec, 1.17/CORBA2.3–C++–Mapping).	112
ValueBox/		To verify the semantics of passing instances of valuebox types as in and out parameters (Ref: 3.8, 5/CORBA2.3–Spec, 1.17/CORBA2.3–C++–Mapping).	116
Detailed Comments :			

Test Case Index				
Test Group Reference	Test Case Id	Selection Ref	Description	Page Nr
Substitutability/	Sub_intf_1	ValueD_ImplExist		112
Substitutability/	Sub_val_2			114
ValueBox/	ShortBox_in			116
ValueBox/	ShortBox_out			118
ValueBox/	UShortBox_in			120
ValueBox/	UShortBox_out			122
ValueBox/	LongBox_in			124
ValueBox/	LongBox_out			126
ValueBox/	ULongBox_in			128
ValueBox/	ULongBox_out			130
ValueBox/	BooleanBox_in			132
ValueBox/	BooleanBox_out			134
ValueBox/	OctetBox_in			136
ValueBox/	OctetBox_out			138
ValueBox/	CharBox_in			140
ValueBox/	CharBox_out			142
ValueBox/	FloatBox_in			144
ValueBox/	FloatBox_out			146
ValueBox/	DoubleBox_in			148
ValueBox/	DoubleBox_out			150
ValueBox/	StringBox_in			152
ValueBox/	StringBox_out			154
ValueBox/	EnumBox_in			156
ValueBox/	EnumBox_out			158
ValueBox/	FixStructBox_in			160
ValueBox/	FixStructBox_out			162
ValueBox/	VarStructBox_in			164
ValueBox/	VarStructBox_out			166
ValueBox/	FixUnionBox_in			168
ValueBox/	FixUnionBox_out			170
ValueBox/	VarUnionBox_in			172
ValueBox/	VarUnionBox_out		174	
ValueBox/	ShortSeqBox_in		176	
ValueBox/	ShortSeqBox_out		178	
ValueBox/	StrSeqBox_in		180	
ValueBox/	StrSeqBox_out		182	
ValueBox/	Val_out_struct		184	
	Hello		185	
Detailed Comments :				

Test Step Index			
Test Step Group Reference	Test Step Id	Description	Page Nr
	ObjtoPCO	Get the object reference of objectName and associate the PCO pcolD with the object.	186
	CreateObj	Create the object with objectName and associate it with the PCO pcolD.	186
	RegisterFactory	Register the factory for value with rid with the ORB.	187
	UnregisterFactory	Unregister the factory for value with rid with the ORB.	187
Detailed Comments :			

Default Index			
Default Group Reference	Default Id	Description	Page Nr
	Default_A	Default test step for interface TestInterfaceA related tests.	188
	Default_C	Default test step for interface TestInterfaceA related tests.	188
Detailed Comments :			

II

Declarations Part

ASN.1 Type Definition	
Type Name	: CORBA__Long
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type long.
Type Definition	
INTEGER	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__ULong
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type unsigned long.
Type Definition	
INTEGER	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__LongLong
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type long long.
Type Definition	
INTEGER	
Detailed Comments	: Defined in CORBA 2.2

ASN.1 Type Definition	
Type Name	: CORBA__ULongLong
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type unsigned long long.
Type Definition	
INTEGER	
Detailed Comments	: Defined in CORBA 2.2

ASN.1 Type Definition	
Type Name	: CORBA__Short
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type short.
Type Definition	
INTEGER	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__UShort
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type unsigned short.
Type Definition	
INTEGER	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__Char
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type char.
Type Definition	
GraphicString	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__WChar
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type wild char.
Type Definition	
GraphicString	
Detailed Comments	: Defined in CORBA 2.2

ASN.1 Type Definition	
Type Name	: CORBA__Boolean
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type boolean.
Type Definition	
BOOLEAN	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__Octet
Encoding Variation	:
Comments	: Translation of the CORBA IDL basic type octet.
Type Definition	
BIT STRING (SIZE(8))	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__String
Encoding Variation	:
Comments	: Translation of the CORBA IDL type string.
Type Definition	
GraphicString	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__WString
Encoding Variation	:
Comments	: Translation of the CORBA IDL type wild string.
Type Definition	
GraphicString	
Detailed Comments	: Defined in CORBA 2.2

ASN.1 Type Definition	
Type Name	: CORBA__Object
Encoding Variation	:
Comments	: Translation of the CORBA IDL type Object.
Type Definition	
IA5String	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__Void
Encoding Variation	:
Comments	: Translation of the CORBA IDL type void.
Type Definition	
NULL	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: CORBA__Float
Encoding Variation	:
Comments	: CORBA float.
Type Definition	
<pre> SEQUENCE { mt INTEGER, bs INTEGER, ep INTEGER } </pre>	
<p>Detailed Comments : REAL is currently not supported by matching. Therefore CORBA__Float is defined by a SEQUENCE with 3 fields for mantissa, base and exponent.</p> <p>To support generic coding/decoding, the following conversion is used:</p> <ul style="list-style-type: none"> – As an IDL float is stored in 4 bytes and has about 6 decimal places for accuracy, base 10 and exponent –6 are used. In the implementation: <pre> afloat = mt * pow (10, –6) mt = afloat * pow (10, 6) </pre>	

ASN.1 Type Definition	
Type Name	: CORBA__Double
Encoding Variation	:
Comments	: CORBA double.
Type Definition	
<pre> SEQUENCE { mt INTEGER, bs INTEGER, ep INTEGER } </pre>	
<p>Detailed Comments : REAL is currently not supported by matching. Therefore CORBA__Double is defined by a SEQUENCE with 3 fields for mantissa, base and exponent.</p> <p>To support generic coding/decoding, the following conversion is used:</p> <ul style="list-style-type: none"> – As an IDL double is stored in 8 bytes and has about 15 decimal places for accuracy, base 10 and exponent –6 are used. In the implementation: <pre> adouble = mt * pow (10, –15) mt = adouble * pow (10, 15) </pre>	

ASN.1 Type Definition	
Type Name	: CORBA__Any
Encoding Variation	:
Comments	: The is a help type to enable generic coding functions. All types the ATS that may be associated with an CORBA__Any are collected in this CHOICE type.
Type Definition	
<pre> CHOICE { any_CORBA__Long [0] CORBA__Long, any_CORBA__ULong [1] CORBA__ULong, any_CORBA__LongLong [2] CORBA__LongLong, any_CORBA__ULongLong [3] CORBA__ULongLong, any_CORBA__Short [4] CORBA__Short , any_CORBA__UShort [5] CORBA__UShort, any_CORBA__Char [6] CORBA__Char, any_CORBA__WChar [7] CORBA__WChar, any_CORBA__String [8] CORBA__String, any_CORBA__WString [9] CORBA__WString, any_CORBA__Octet [10] CORBA__Octet, any_CORBA__Boolean [11] CORBA__Boolean, any_CORBA__Object [12] CORBA__Object } </pre>	
Detailed Comments	: Currently only basic types are considered.

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestShortBox
Encoding Variation	:
Comments	: Value box TestShortBox
Type Definition	
CORBA__Short	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestUShortBox
Encoding Variation	:
Comments	: Value box TestUShortBox
Type Definition	
CORBA__UShort	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestLongBox
Encoding Variation	:
Comments	: Value box TestLongBox
Type Definition	
CORBA__Long	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestULongBox
Encoding Variation	:
Comments	: Value box TestULongBox
Type Definition	
CORBA__ULong	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestLLongBox
Encoding Variation	:
Comments	: Value box TestLLongBox
Type Definition	
CORBA__LongLong	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestULLongBox
Encoding Variation	:
Comments	: Value box TestULLongBox
Type Definition	
CORBA__ULongLong	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: OBVTest__TestBoolean
Encoding Variation	:
Comments	:
Type Definition	
CORBA__Boolean	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestBooleanBox
Encoding Variation	:
Comments	: Value box TestBooleanBox
Type Definition	
OBVTest__TestBoolean	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: OBVTest__TestOctet
Encoding Variation	:
Comments	:
Type Definition	
CORBA__Octet	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestOctetBox
Encoding Variation	:
Comments	: Value box TestOctetBox
Type Definition	
OBVTest__TestOctet	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestCharBox
Encoding Variation	:
Comments	: Value box TestCharBox
Type Definition	
CORBA__Char	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestWCharBox
Encoding Variation	:
Comments	: Value box TestWCharBox
Type Definition	
CORBA__WChar	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestFloatBox
Encoding Variation	:
Comments	: Value box TestFloatBox
Type Definition	
CORBA__Float	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestDoubleBox
Encoding Variation	:
Comments	: Value box TestDoubleBox
Type Definition	
CORBA__Double	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestStringBox
Encoding Variation	:
Comments	: Value box TestStringBox
Type Definition	
CORBA__String	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: ENUM__OBVTest__TestEnum
Encoding Variation	:
Comments	: Enumerated
Type Definition	
ENUMERATED { red(0), green (1), blue (2) }	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestEnumBox
Encoding Variation	:
Comments	: Value box TestEnumBox
Type Definition	
ENUM__OBVTest__TestEnum	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: ARRAY__OBVTest__TestArray
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE SIZE(1) OF SEQUENCE SIZE(2) OF CORBA__Long	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestArrayBox
Encoding Variation	:
Comments	: Value box TestArrayBox
Type Definition	
ARRAY__OBVTest__TestArray	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: STRUCT__OBVTest__TestFixStruct
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE { x CORBA__Long, y CORBA__Long, radius CORBA__Double }	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestFixStructBox
Encoding Variation	:
Comments	: Value box TestFixStructBox
Type Definition	
STRUCT__OBVTest__TestFixStruct	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: STRUCT__OBVTest__TestVarStruct
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE { name CORBA__String, email CORBA__String }	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestVarStructBox
Encoding Variation	:
Comments	: Value box TestVarStructBox
Type Definition	
STRUCT__OBVTest__TestVarStruct	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: UNION__OBVTest__TestFixUnion
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE { disc CORBA__Boolean, body CHOICE { csT CORBA__Octet, csF CORBA__Double } }	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestFixUnionBox
Encoding Variation	:
Comments	: Value box TestFixUnionBox
Type Definition	
UNION__OBVTest__TestFixUnion	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: UNION__OBVTest__TestVarUnion
Encoding Variation	:
Comments	:
Type Definition	
<pre>SEQUENCE { disc CORBA__Long, body CHOICE { cs0 CORBA__String, cs9 STRUCT__OBVTest__TestFixStruct } }</pre>	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestVarUnionBox
Encoding Variation	:
Comments	: Value box TestVarUnionBox
Type Definition	
UNION__OBVTest__TestVarUnion	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: SEQ__OBVTest__TestShSeq
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE OF CORBA__Short	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestShSeqBox
Encoding Variation	:
Comments	: Value box TestShSeqBox
Type Definition	
SEQ__OBVTest__TestShSeq	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: SEQ__OBVTest__TestStrSeq
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE OF CORBA__String	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestStrSeqBox
Encoding Variation	:
Comments	: Value box TestStrSeqBox
Type Definition	
SEQ__OBVTest__TestStrSeq	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUEBOX__OBVTest__TestAnyBox
Encoding Variation	:
Comments	: Value box TestAnyBox
Type Definition	
CORBA__Any	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: STRUCT__OBVTest__TestValMbStruct
Encoding Variation	:
Comments	:
Type Definition	
SEQUENCE { nr CORBA__Long, valmb VALUETYPE__OBVTest__TestValueA } 	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: ENUM__OBVTest__Res
Encoding Variation	:
Comments	: Enumerated
Type Definition	
ENUMERATED { IS_OK(0), EXC_NOT_EXIST (1), EXC_NO_IMPL (2), IS_NOK(3) } 	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: INTF__OBVTest__TestInterfaceB
Encoding Variation	:
Comments	: Interface
Type Definition	
CORBA__Object	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: INTF__OBVTest__TestInterfaceD
Encoding Variation	:
Comments	: Interface
Type Definition	
CORBA__Object	
Detailed Comments	:

ASN.1 Type Definition	
Type Name	: VALUETYPE__OBVTest__TestValueA
Encoding Variation	:
Comments	: Value type
Type Definition	
SEQUENCE { name CORBA__String, age CORBA__Long }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: VALUETYPE__OBVTest__TestValueB
Encoding Variation	:
Comments	: Value type
Type Definition	
SEQUENCE { www CORBA__String }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: VALUETYPE__OBVTest__TestValueC
Encoding Variation	:
Comments	: Value type
Type Definition	
SEQUENCE { }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: VALUETYPE__OBVTest__TestValueD
Encoding Variation	:
Comments	: Value type
Type Definition	
SEQUENCE { name CORBA__String, age CORBA__Long, id CORBA__Long }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: EXCEPT__CORBA__COMM_FAILURE
Encoding Variation	:
Comments	: CORBA system exception COMM_FAILURE.
Type Definition	
SEQUENCE { }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: EXCEPT__CORBA__UNKNOWN
Encoding Variation	:
Comments	: CORBA system exception UNKNOWN.
Type Definition	
SEQUENCE { }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: EXCEPT__CORBA__TRANSIENT
Encoding Variation	:
Comments	: CORBA system exception TRANSIENT.
Type Definition	
SEQUENCE { }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: EXCEPT__CORBA__OBJECT_NOT_EXIST
Encoding Variation	:
Comments	: CORBA system exception.
Type Definition	
SEQUENCE { }	
Detailed Comments :	

ASN.1 Type Definition	
Type Name	: EXCEPT__CORBA__NO_IMPLEMENT
Encoding Variation	:
Comments	: CORBA system exception.
Type Definition	
SEQUENCE { }	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: getObjRef(objectName: GraphicString)
Result Type	: CORBA__Object
Comments	: Get the reference of the object objectName.
Description	
/* Get the reference of the object objectName. */ return (getObjRef(objectName));	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: registerValueFactory(rcv: BOOLEAN; rid: GraphicString)
Result Type	: CORBA__Boolean
Comments	: Create a factory for the value with rid and register it with the ORB.
Description	
/* Create a factory for the value with rid and register it with the ORB. rcv determines whether the server or the client is the receiver of the value, for which a value factory should be registered. */ return registerValueFactory(rcv, rid);	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: unregisterValueFactory(rcv: BOOLEAN; rid: GraphicString)
Result Type	: CORBA__Boolean
Comments	: Unregister a factory with the ORB.
Description	
/* Unregister a factory with the ORB. rcv determines whether at the server or at the client side the value factory should be unregistered. */ return unregisterValueFactory(rcv, rid);	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: activatePCOtoSUTServer(pcoID: GraphicString; objectName: GraphicString; ref: CORBA__Object)
Result Type	: BOOLEAN
Comments	: Associate the object objectName with the PCO pcoID. ref is the reference of the object. The PCO must be of the type PCOtoSUTServer.
Description	
/* Associate the object objectName with the PCO pcoID. ref is the reference of the object. The PCO must be of the type PCOtoSUTServer. */ return (activatePCOtoSUTServer(pcoID, objectName, ref));	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: activatePCOtoSUTClient(pcoID: GraphicString; objectName: GraphicString; ref: CORBA__Object)
Result Type	: CORBA__Boolean
Comments	: Associate the object objectName with the PCO pcoID. The PCO must be of the type PCOtoSUTClient. The reference of the object is returned as parameter.
Description	
/* Associate the object objectName with the PCO pcoID. The PCO must be of the type PCOtoSUTClient. The reference of the object is returned as parameter. Return TRUE if the operation is successful. Otherwise, FALSE is returned. */ return (activatePCOtoSUTClient(pcoID, objectName, ref));	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: is_valid_ref(rf:INTF__OBVTest__TestInterfaceB)
Result Type	: BOOLEAN
Comments	: Check if the reference is a valid reference of TestInterfaceB.
Description	
/*Check if the reference is a valid reference of TestInterfaceB.*/ return (is_valid_ref(rf));	
Detailed Comments :	

Test Suite Operation Definition	
Operation Name	: is_valid_vld(vld:VALUETYPE__OBVTest__TestValueD)
Result Type	: BOOLEAN
Comments	: Check if the value instance of type TestValueD is valid.
Description	
/*Check if the value instance of type TestValueD is valid.*/ return(is_valid_vld(vld));	
Detailed Comments :	

Test Suite Parameter Declarations			
Parameter Name	Type	PICS/PIXIT Ref	Comments
Timer1_value	INTEGER	PIXIT	TSPar. Value of Timer1 in s. Default: 3 s.
ValueA_Impl	BOOLEAN	PIXIT	True: implementation of TestvalueA can be loaded in the receiving context. False: otherwise.
ValueB_Impl	BOOLEAN	PIXIT	True: implementation of TestvalueB can be loaded in the receiving context. False: otherwise.
ValueD_Impl	BOOLEAN	PIXIT	True: implementation of TestvalueD can be loaded in the receiving context. False: otherwise.
LongDouble_Supp	BOOLEAN	PICS	Support for the IDL type long double is optional. True: long double is supported. False: otherwise.
Detailed Comments :			

Test Case Selection Expression Definitions		
Expression Name	Selection Expression	Comments
ValueA_ImplExist	ValueA_Impl	Implementation of TestvalueA can be loaded in the receiving context.
ValueA_ImplNotExist	NOT(ValueA_Impl)	Implementation of TestvalueA can not be loaded in the receiving context.
ValueB_ImplExist	ValueB_Impl	Implementation of TestvalueB can be loaded in the receiving context.
ValueB_ImplNotExist	NOT(ValueB_Impl)	Implementation of TestvalueB can not be loaded in the receiving context.
ValueD_ImplExist	ValueD_Impl	Implementation of TestvalueD can be loaded in the receiving context.
ValueD_ImplNotExist	NOT(ValueD_Impl)	Implementation of TestvalueD can not be loaded in the receiving context.
LongDouble_Impl	LongDouble_Supp	The type long double is supported.
LongDouble_NotImpl	NOT(LongDouble_Supp)	The type long double is not supported.
ValueAD_ImplNotExist	(NOT(ValueA_Impl)) AND (NOT(ValueD_Impl))	Neither impl. of TestvalueA nor impl. of TestvalueD can not be loaded in the receiving context.
Detailed Comments :		

Test Suite Constant Declarations			
Constant Name	Type	Value	Comments
IFA_NAME	GraphicString	"OBVTest::TestInterfaceA"	Name of the interface OBVTest::TestInterfaceA.
IFC_NAME	GraphicString	"OBVTest::TestInterfaceC"	Name of the interface OBVTest::TestInterfaceC.
IFH_NAME	GraphicString	"OBVTest::TestHelper"	Name of the interface OBVTest::TestHelper
PCOIFA_NAME	GraphicString	"PCO_IFA"	Name of the PCO PCO_IFA to the interface OBVTest::TestInterfaceA.
PCOIFC_NAME	GraphicString	"PCO_IFC"	Name of the PCO PCO_IFC to the interface OBVTest::TestInterfaceC.
PCOIFH_NAME	GraphicString	"PCO_IFH"	Name of the PCO PCO_IFH to the interface OBVTest::TestHelper.
IFA_ID	GraphicString	"IDL:OBVTest/TestInterface A:1.0"	Repository ID of the interface OBVTest::TestInterfaceA.
IFB_ID	GraphicString	"IDL:OBVTest/TestInterface B:1.0"	Repository ID of the interface OBVTest::TestInterfaceB.
IFC_ID	GraphicString	"IDL:OBVTest/TestInterface C:1.0"	Repository ID of the interface OBVTest::TestInterfaceC.
IFD_ID	GraphicString	"IDL:OBVTest/TestInterface D:1.0"	Repository ID of the interface OBVTest::TestInterfaceD.
VLA_ID	GraphicString	"IDL:OBVTest/TestValueA:1. 0"	Repository ID of the interface OBVTest::TestValueA.
VLB_ID	GraphicString	"IDL:OBVTest/TestValueB:1. 0"	Repository ID of the interface OBVTest::TestValueB.
VLC_ID	GraphicString	"IDL:OBVTest/TestValueC:1. 0"	Repository ID of the interface OBVTest::TestValueC.
VLD_ID	GraphicString	"IDL:OBVTest/TestValueD:1. 0"	Repository ID of the interface OBVTest::TestValueD.
SHORTBOX_ID	GraphicString	"IDL:OBVTest/TestShortBox :1.0"	Repository ID of the interface OBVTest::TestShortBox.
USHORTBOX_ID	GraphicString	"IDL:OBVTest/TestUShortB ox:1.0"	Repository ID of the interface OBVTest::TestUShortBox.
LONGBOX_ID	GraphicString	"IDL:OBVTest/TestLongBox: 1.0"	Repository ID of the interface OBVTest::TestLongBox.
ULONGBOX_ID	GraphicString	"IDL:OBVTest/TestULongBo x:1.0"	Repository ID of the interface OBVTest::TestULongBox.
LLONGBOX_ID	GraphicString	"IDL:OBVTest/TestLLongBo x:1.0"	Repository ID of the interface OBVTest::TestLLongBox.
ULLONGBOX_ID	GraphicString	"IDL:OBVTest/TestULLongB ox:1.0"	Repository ID of the interface OBVTest::TestULLongBox.

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	Type	Value	Comments
BOOLEANBOX_ID	GraphicString	"IDL:OBVTest/TestBooleanBox:1.0"	Repository ID of the interface OBVTest::TestBooleanBox.
OCTETBOX_ID	GraphicString	"IDL:OBVTest/TestOctetBox:1.0"	Repository ID of the interface OBVTest::TestOctetBox.
CHARBOX_ID	GraphicString	"IDL:OBVTest/TestCharBox:1.0"	Repository ID of the interface OBVTest::TestCharBox.
WCHARBOX_ID	GraphicString	"IDL:OBVTest/TestWCharBox:1.0"	Repository ID of the interface OBVTest::TestWCharBox.
FLOATBOX_ID	GraphicString	"IDL:OBVTest/TestFloatBox:1.0"	Repository ID of the interface OBVTest::TestFloatBox.
DOUBLEBOX_ID	GraphicString	"IDL:OBVTest/TestDoubleBox:1.0"	Repository ID of the interface OBVTest::TestDoubleBox.
LDOUBLEBOX_ID	GraphicString	"IDL:OBVTest/TestLDoubleBox:1.0"	Repository ID of the interface OBVTest::TestLDoubleBox.
STRINGBOX_ID	GraphicString	"IDL:OBVTest/TestStringBox:1.0"	Repository ID of the interface OBVTest::TestStringBox.
ENUMBOX_ID	GraphicString	"IDL:OBVTest/TestEnumBox:1.0"	Repository ID of the interface OBVTest::TestEnumBox.
ARRAYBOX_ID	GraphicString	"IDL:OBVTest/TestArrayBox:1.0"	Repository ID of the interface OBVTest::TestArrayBox.
FSTRUCTBOX_ID	GraphicString	"IDL:OBVTest/TestFixStructBox:1.0"	Repository ID of the interface OBVTest::TestFixStructBox.
VSTRUCTBOX_ID	GraphicString	"IDL:OBVTest/TestVarStructBox:1.0"	Repository ID of the interface OBVTest::TestVarStructBox.
FUNIONBOX_ID	GraphicString	"IDL:OBVTest/TestFixUnionBox:1.0"	Repository ID of the interface OBVTest::TestFixUnionBox.
VUNIONBOX_ID	GraphicString	"IDL:OBVTest/TestVarUnionBox:1.0"	Repository ID of the interface OBVTest::TestVarUnionBox.
SHSEQBOX_ID	GraphicString	"IDL:OBVTest/TestShSeqBox:1.0"	Repository ID of the interface OBVTest::TestShSeqBox.
STRSEQBOX_ID	GraphicString	"IDL:OBVTest/TestStrSeqBox:1.0"	Repository ID of the interface OBVTest::TestStrSeqBox.
ANYBOX_ID	GraphicString	"IDL:OBVTest/TestAnyBox:1.0"	Repository ID of the interface OBVTest::TestAnyBox.
SRV_RCV	BOOLEAN	TRUE	Server is the receiver of the value, for which a value factory should be registered.

Continued on next page

Continued from previous page

Test Suite Constant Declarations			
Constant Name	Type	Value	Comments
CLT_RCV	BOOLEAN	FALSE	Client is the receiver of the value, for which a value factory should be registered.
LNG1	CORBA__Long	573	
LNG2	CORBA__Long	8037	
SHRT1	CORBA__Short	34	
SHRT2	CORBA__Short	79	
SHRT3	CORBA__Short	102	
STR1	CORBA__String	"GMD"	
STR2	CORBA__String	"FOKUS"	
STR3	CORBA__String	"www.fokus.gmd.de"	
Detailed Comments :			

Test Case Variable Declarations			
Variable Name	Type	Value	Comments
boolResult	BOOLEAN		
reflFA	CORBA__Object		Reference of TestInterfaceA.
reflFB	INTF__OBVTest__TestInterfaceB		Reference of TestInterfaceB.
reflFC	CORBA__Object		Reference of TestInterfaceC.
reflFD	INTF__OBVTest__TestInterfaceD		Reference of TestInterfaceD.
reflFH	CORBA__Object		Reference of TestHelper.
instVLA	VALUETYPE__OBVTest__TestValueA		Instance of TestValueA.
instVLB	VALUETYPE__OBVTest__TestValueB		Instance of TestValueB.
instVLC	VALUETYPE__OBVTest__TestValueC		Instance of TestValueC.
instVLD	VALUETYPE__OBVTest__TestValueD		Instance of TestValueD.
num	INTEGER		Integer value.
exp	INTEGER		Integer value for exponent.
oct_var	CORBA__Octet		
char_var	CORBA__Char		
float_var	CORBA__Float		
double_var	CORBA__Double		
str_var	CORBA__String		
enum_var	ENUM__OBVTest__TestEnum		
fst_var	STRUCT__OBVTest__TestFixtureStruct		
vst_var	STRUCT__OBVTest__TestVarStruct		
fun_var	UNION__OBVTest__TestFixtureUnion		
vun_var	UNION__OBVTest__TestVarUnion		
shs_var	SEQ__OBVTest__TestSeq		
shr_var	SEQ__OBVTest__TestStrSeq		
lg_var	CORBA__Long		
age_var	CORBA__Long		
vla_var	VALUETYPE__OBVTest__TestValueA		
vld_var	VALUETYPE__OBVTest__TestValueD		
Detailed Comments :			

PCO Type Declarations		
PCO Type	Role	Comments
PCOtoSUTServer	LT	
PCOtoSUTClient	LT	
Detailed Comments :		

PCO Declarations			
PCO Name	PCO Type	Role	Comments
PCO_IFA	PCOtoSUTClient	LT	PCO PCO_IFA to the interface OBVTest::TestInterfaceA.
PCO_IFC	PCOtoSUTClient	LT	PCO PCO_IFB to the interface OBVTest::TestInterfaceC.
PCO_IFH	PCOtoSUTServer	LT	PCO PCO_IFH to ther interface OBVTest::TestHelper.
Detailed Comments :			

Timer Declarations			
Timer Name	Duration	Unit	Comments
Timer1	Timer1_value	s	
Detailed Comments :			

ASN.1 ASP Type Definition
ASP Name : pRAISE__CORBA__COMM_FAILURE PCO Type : PCOtoSUTClient Comments : CORBA system exception.
Type Definition
EXCEPT__CORBA__COMM_FAILURE
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pRAISE__CORBA__UNKNOWN PCO Type : PCOtoSUTClient Comments : CORBA system exception.
Type Definition
EXCEPT__CORBA__UNKNOWN
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pRAISE__CORBA__TRANSIENT PCO Type : PCOtoSUTClient Comments : CORBA system exception.
Type Definition
EXCEPT__CORBA__TRANSIENT
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pRAISE__CORBA__OBJECT_NOT_EXIST PCO Type : PCOtoSUTClient Comments : CORBA system exception.
Type Definition
EXCEPT__CORBA__OBJECT_NOT_EXIST
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pRAISE__CORBA__NO_IMPLEMENT PCO Type : PCOtoSUTClient Comments : CORBA system exception.
Type Definition
EXCEPT__CORBA__NO_IMPLEMENT
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceA__val_sub_intf PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation val_sub_intf (value instance substitutes interface type).
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, ifb INTF__OBVTest__TestInterfaceB }
Detailed Comments : Use instance of TestValueC.

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceA__val_sub_intf PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation val_sub_intf (value instance substitutes interface type).
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceA__val_sub_val PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation val_sub_val (value instance substitutes value type).
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, val VALUETYPE__OBVTest__TestValueD }
Detailed Comments : Use instance of TestValueA.

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceA__val_sub_val PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation val_sub_intf (value instance substitutes value type).
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res } </pre>
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceA__sub_abstr_intf__1 PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation sub_abstr_intf (value instance or interface instance substitutes abstract interface type).
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, abif VALUETYPE__OBVTest__TestValueB } </pre>
Detailed Comments : Use instance of TestValueC.

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceA__sub_abstr_intf PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation val_sub_intf (value instance or interface instance substitutes abstract interface type).
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res } </pre>
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceA__sub_abstr_intf__2 PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation sub_abstr_intf (value instance or interface instance substitutes abstract interface type).
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, abif INTF__OBVTest__TestInterfaceB }
Detailed Comments : Use instance of TestValueC.

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_shortbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_shortbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestShortBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_shortbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_shortbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Short }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__out_shortbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_shortbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Short } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_shortbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_shortbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestShortBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_ushortbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_ushortbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestUShortBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_ushortbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_ushortbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__UShort }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_ushortbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_ushortbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__UShort }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_ushortbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_ushortbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestUShortBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_longbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_longbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestLongBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_longbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_longbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Long }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_longbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_longbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Long }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_longbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_longbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestLongBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_ulongbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_ulongbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestULongBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_ulongbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_ulongbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__ULong }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__out_ulongbox PCO Type : PC OtoSUTClient Comments : Request-related ASP for the operation out_ulongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__ULong } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_ulongbox PCO Type : PC OtoSUTClient Comments : Reply-related ASP for the operation out_ulongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestULongBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_llongbox PCO Type : PC OtoSUTClient Comments : Request-related ASP for the operation in_llongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestLLongBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_llongbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_llongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__LongLong } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__out_llongbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_llongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__LongLong } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_llongbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_llongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestLLongBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_ullongbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_ullongbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestULLongBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_ullongbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_ullongbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__ULongLong }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_ullongbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_ullongbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__ULongLong }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_ullongbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_ullongbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestULLongBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_booleanbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_booleanbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestBooleanBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_booleanbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_booleanbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Boolean } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_booleanbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_booleanbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Boolean }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_booleanbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_booleanbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestBooleanBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_octetbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_octetbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestOctetBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in__octetbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in__octetbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Octet }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out__octetbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out__octetbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Octet }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out__octetbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out__octetbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestOctetBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_charbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_charbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestCharBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_charbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_charbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Char }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_charbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_charbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Char }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_charbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_charbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestCharBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_wcharbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_wcharbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestWCharBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_wcharbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_wcharbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__WChar } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_wcharbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_wcharbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__WChar }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_wcharbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_wcharbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestWCharBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_floatbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_floatbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestFloatBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_floatbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_floatbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Float }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_floatbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_floatbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Float }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_floatbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_floatbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestFloatBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_doublebox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_doublebox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestDoubleBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_doublebox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_doublebox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Double }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_doublebox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_doublebox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Double }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_doublebox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_doublebox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestDoubleBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_stringbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_stringbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestStringBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_stringbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_stringbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__String }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_stringbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_stringbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__String }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_stringbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_stringbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestStringBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_enumbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_enumbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestEnumBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_enumbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_enumbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov ENUM__OBVTest__TestEnum }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_enumbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_enumbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv ENUM__OBVTest__TestEnum }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_enumbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_enumbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestEnumBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_arraybox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_arraybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestArrayBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_arraybox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_arraybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov ARRAY__OBVTest__TestArray }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_arraybox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_arraybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv ARRAY__OBVTest__TestArray }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_arraybox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_arraybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestArrayBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_fstructbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_fstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestFixStructBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_fstructbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_fstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov STRUCT__OBVTest__TestFixStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_fstructbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_fstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv STRUCT__OBVTest__TestFixStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_fstructbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_fstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestFixStructBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_vstructbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_vstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestVarStructBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_vstructbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_vstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov STRUCT__OBVTest__TestVarStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_vstructbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_vstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv STRUCT__OBVTest__TestVarStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_vstructbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_vstructbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestVarStructBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_funionbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_funionbox
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestFixUnionBox } </pre>
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_funionbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_funionbox
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov UNION__OBVTest__TestFixUnion } </pre>
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_funionbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_funionbox
Type Definition
<pre> SEQUENCE { oid CORBA__String, opn CORBA__String, iv UNION__OBVTest__TestFixUnion } </pre>
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_funionbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_funionbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestFixUnionBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_vunionbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_vunionbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestVarUnionBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_vunionbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_vunionbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov UNION__OBVTest__TestVarUnion }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__out_vunionbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_vunionbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv UNION__OBVTest__TestVarUnion } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_vunionbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_vunionbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestVarUnionBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_shseqbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_shseqbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestShSeqBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_shseqbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_shseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov SEQ__OBVTest__TestShSeq }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_shseqbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_shseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv SEQ__OBVTest__TestShSeq }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_shseqbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_shseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestShSeqBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__in_strseqbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_strseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestStrSeqBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_strseqbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_strseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov SEQ__OBVTest__TestStrSeq }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_strseqbox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_strseqbox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv SEQ__OBVTest__TestStrSeq }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_strseqbox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_strseqbox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestStrSeqBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pCALL__OBVTest__TestInterfaceC__in_anybox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation in_anybox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUEBOX__OBVTest__TestAnyBox } 	
Detailed Comments :	

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__in_anybox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation in_anybox	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov CORBA__Any } 	
Detailed Comments :	

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__out_anybox PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation out_anybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv CORBA__Any }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__out_anybox PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation out_anybox
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUEBOX__OBVTest__TestAnyBox }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__val_in_struct PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation val_in_struct
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv VALUETYPE__OBVTest__TestValueA, nr CORBA__Long }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__val_in_struct PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation val_in_struct
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov STRUCT__OBVTest__TestValMbStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__val_out_struct PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation val_out_struct
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, iv STRUCT__OBVTest__TestValMbStruct }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestInterfaceC__val_out_struct PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation val_out_struct
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, ov VALUETYPE__OBVTest__TestValueA }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestHelper__make_call PCO Type : PCOtoSUTServer Comments : Request-related ASP for the operation make_call on TestHelper.
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, cif CORBA__String, cop CORBA__String }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pREPLY__OBVTest__TestHelper__make_call PCO Type : PCOtoSUTServer Comments : Reply-related ASP for the operation make_call on TestHelper.
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue CORBA__Void, rs ENUM__OBVTest__Res }
Detailed Comments :

ASN.1 ASP Type Definition
ASP Name : pCALL__OBVTest__TestInterfaceC__hello PCO Type : PCOtoSUTClient Comments : Request-related ASP for the operation hello.
Type Definition
SEQUENCE { oid CORBA__String, opn CORBA__String }
Detailed Comments :

ASN.1 ASP Type Definition	
ASP Name : pREPLY__OBVTest__TestInterfaceC__hello PCO Type : PCOtoSUTClient Comments : Reply-related ASP for the operation hello	
Type Definition	
SEQUENCE { oid CORBA__String, opn CORBA__String, returnValue ENUM__OBVTest__Res, istr CORBA__String }	
Detailed Comments :	

III

Constraints Part

ASN.1 Type Constraint Declaration	
Constraint Name	: TestValueB__s1
ASN1 Type	: VALUETYPE__OBVTest__TestValueB
Derivation Path	:
Encoding Variation	:
Comments	: Value type
Constraint Value	
{ www "fokus" }	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestValueD__s1
ASN1 Type	: VALUETYPE__OBVTest__TestValueD
Derivation Path	:
Encoding Variation	:
Comments	: Value type
Constraint Value	
{ name "Fritz", age 7, id 102 }	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestValueC__s1
ASN1 Type	: VALUETYPE__OBVTest__TestValueC
Derivation Path	:
Encoding Variation	:
Comments	: Value type
Constraint Value	
{ }	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestShSeq__s1
ASN1 Type	: SEQ__OBVTest__TestShSeq
Derivation Path	:
Encoding Variation	:
Comments	: TestShSeq
Constraint Value	
{SHRT1, SHRT2, SHRT3}	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestStrSeq__s1
ASN1 Type	: SEQ__OBVTest__TestStrSeq
Derivation Path	:
Encoding Variation	:
Comments	: TestStrSeq
Constraint Value	
{STR1, STR2, STR3}	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestShSeqBox__s1
ASN1 Type	: VALUEBOX__OBVTest__TestShSeqBox
Derivation Path	:
Encoding Variation	:
Comments	: Value box TestShSeqBox
Constraint Value	
{SHRT1, SHRT2, SHRT3}	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestStrSeqBox__s1
ASN1 Type	: VALUEBOX__OBVTest__TestStrSeqBox
Derivation Path	:
Encoding Variation	:
Comments	: Value box TestStrSeqBox
Constraint Value	
{STR1, STR2, STR3}	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestArray__s1
ASN1 Type	: ARRAY__OBVTest__TestArray
Derivation Path	:
Encoding Variation	:
Comments	:
Constraint Value	
{ {LNG1, LNG2} }	
Detailed Comments :	

ASN.1 Type Constraint Declaration	
Constraint Name	: TestArrayBox__s1
ASN1 Type	: VALUEBOX__OBVTest__TestArrayBox
Derivation Path	:
Encoding Variation	:
Comments	: Value box TestArrayBox
Constraint Value	
{ {LNG1, LNG2} }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: COMM_FAILURE__r1
ASP Type	: pRAISE__CORBA__COMM_FAILURE
Derivation Path	:
Comments	: CORBA system exception.
Constraint Value	
{}	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: UNKNOWN__r1
ASP Type	: pRAISE__CORBA__UNKNOWN
Derivation Path	:
Comments	: CORBA system exception.
Constraint Value	
{}	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: TRANSIENT__r1
ASP Type	: pRAISE__CORBA__TRANSIENT
Derivation Path	:
Comments	: CORBA system exception.
Constraint Value	
{}	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: OBJECT_NOT_EXIST__r1
ASP Type	: pRAISE__CORBA__OBJECT_NOT_EXIST
Derivation Path	:
Comments	: CORBA system exception.
Constraint Value	
{}	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: NO_IMPLEMENT__r1
ASP Type	: pRAISE__CORBA__NO_IMPLEMENT
Derivation Path	:
Comments	: CORBA system exception.
Constraint Value	
{}	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__val_sub_intf__s1(ref: INTF__OBVTest__TestInterfaceB)
ASP Type	: pCALL__OBVTest__TestInterfaceA__val_sub_intf
Derivation Path	:
Comments	: Request-related ASP for the operation val_sub_intf (value instance substitutes interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_intf", ifb ref }</pre>	
Detailed Comments : Use instance of TestValueC.	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_sub_intf__r1(res: ENUM__OBVTest__Res)
ASP Type	: pREPLY__OBVTest__TestInterfaceA__val_sub_intf
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance substitutes interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_intf", returnValue res }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_sub_intf__r2
ASP Type	: pREPLY__OBVTest__TestInterfaceA__val_sub_intf
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance substitutes interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_intf", returnValue EXC_NOT_EXIST }</pre>	
Detailed Comments	:

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__val_sub_val__s1
ASP Type	: pCALL__OBVTest__TestInterfaceA__val_sub_val
Derivation Path	:
Comments	: Request-related ASP for the operation val_sub_val (value instance substitutes value type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_val", val ? }</pre>	
Detailed Comments	: Use instance of TestValueA.

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_sub_val__r1(res: ENUM__OBVTest__Res)
ASP Type	: pREPLY__OBVTest__TestInterfaceA__val_sub_val
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance substitutes value type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_val", returnValue res }</pre>	
Detailed Comments	:

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_sub_val__r2
ASP Type	: pREPLY__OBVTest__TestInterfaceA__val_sub_val
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance substitutes value type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "val_sub_val", returnValue EXC_NO_IMPL }</pre>	
Detailed Comments	:

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__sub_abstr_intf__s1
ASP Type	: pCALL__OBVTest__TestInterfaceA__sub_abstr_intf__1
Derivation Path	:
Comments	: Request-related ASP for the operation sub_abstr_intf (value instance or interface instance substitutes abstract interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "sub_abstr_intf", abif TestValueB__s1 }</pre>	
Detailed Comments	: Use instance of TestValueC.

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__sub_abstr_intf__s2(inst: INTF__OBVTest__TestInterfaceB)
ASP Type	: pCALL__OBVTest__TestInterfaceA__sub_abstr_intf__2
Derivation Path	:
Comments	: Request-related ASP for the operation sub_abstr_intf (value instance or interface instance substitutes abstract interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "sub_abstr_intf", abif inst }</pre>	
Detailed Comments	: Use instance of TestValueC.

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__sub_abstr_intf__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceA__sub_abstr_intf
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance or interface instance substitutes abstract interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "sub_abstr_intf", returnValue IS_OK }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__sub_abstr_intf__r2
ASP Type	: pREPLY__OBVTest__TestInterfaceA__sub_abstr_intf
Derivation Path	:
Comments	: Reply-related ASP for the operation val_sub_intf (value instance or interface instance substitutes abstract interface type).
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceA", opn "sub_abstr_intf", returnValue EXC_NO_IMPL }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_shortbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_shortbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_shortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_shortbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_shortbox__r1(n:INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_shortbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_shortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_shortbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_shortbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_shortbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_shortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_shortbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_shortbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_shortbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_shortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_shortbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_ushortbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_ushortbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_ushortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ushortbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_ushortbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_ushortbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_ushortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ushortbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_ushortbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_ushortbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_ushortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ushortbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_ushortbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_ushortbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_ushortbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ushortbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_longbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_longbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_longbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_longbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_longbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_longbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_longbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_longbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_longbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_longbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_longbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_longbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_longbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_longbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_longbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_longbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_ulongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_ulongbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_ulongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ulongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_ulongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_ulongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_ulongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ulongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_ulongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_ulongbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_ulongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ulongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_ulongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_ulongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_ulongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ulongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_llongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_llongbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_llongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_llongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_llongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_llongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_llongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_llongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_llongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_llongbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_llongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_llongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_llongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_llongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_llongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_llongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_ullongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_ullongbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_ullongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ullongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_ullongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_ullongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_ullongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_ullongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_ullongbox__s1(n: INTEGER)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_ullongbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_ullongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ullongbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_ullongbox__r1(n: INTEGER)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_ullongbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_ullongbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_ullongbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_booleanbox__s1(n: BOOLEAN)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_booleanbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_booleanbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_booleanbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_booleanbox__r1(n:BOOLEAN)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_booleanbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_booleanbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_booleanbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_booleanbox__s1(n:BOOLEAN)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_booleanbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_booleanbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_booleanbox", iv n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_booleanbox__r1(n:BOOLEAN)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_booleanbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_booleanbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_booleanbox", returnValue IS_OK, ov n }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_octetbox__s1(str:CORBA__Octet)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_octetbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_octetbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_octetbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_octetbox__r1(str:CORBA__Octet)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_octetbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_octetbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_octetbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_octetbox__s1(str:CORBA__Octet)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_octetbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_octetbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_octetbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_octetbox__r1(str:CORBA__Octet)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_octetbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_octetbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_octetbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_charbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_charbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_charbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_charbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_charbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_charbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_charbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_charbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_charbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_charbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_charbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_charbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_charbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_charbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_charbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_charbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_wcharbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_wcharbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_wcharbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_wcharbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_wcharbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_wcharbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_wcharbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_wcharbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_wcharbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_wcharbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_wcharbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_wcharbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_wcharbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_wcharbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_wcharbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_wcharbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_floatbox__s1(flt: CORBA__Float)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_floatbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_floatbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_floatbox", iv flt }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_floatbox__r1(flt: CORBA__Float)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_floatbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_floatbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_floatbox", returnValue IS_OK, ov flt }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_floatbox__s1(flt: CORBA__Float)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_floatbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_floatbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_floatbox", iv flt }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_floatbox__r1(flt: CORBA__Float)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_floatbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_floatbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_floatbox", returnValue IS_OK, ov flt }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_doublebox__s1(db: CORBA__Double)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_doublebox
Derivation Path	:
Comments	: Request-related ASP for the operation in_doublebox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_doublebox", iv db }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_doublebox__r1(db: CORBA__Double)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_doublebox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_doublebox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_doublebox", returnValue IS_OK, ov db }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_doublebox__s1(db: CORBA__Double)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_doublebox
Derivation Path	:
Comments	: Request-related ASP for the operation out_doublebox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_doublebox", iv db }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_doublebox__r1(db: CORBA__Double)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_doublebox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_doublebox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_doublebox", returnValue IS_OK, ov db }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_stringbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_stringbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_stringbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_stringbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_stringbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_stringbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_stringbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_stringbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_stringbox__s1(str:GraphicString)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_stringbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_stringbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_stringbox", iv str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_stringbox__r1(str:GraphicString)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_stringbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_stringbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_stringbox", returnValue IS_OK, ov str }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_enumbox__s1(en: ENUM__OBVTest__TestEnum)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_enumbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_enumbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_enumbox", iv en }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_enumbox__r1(en: ENUM__OBVTest__TestEnum)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_enumbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_enumbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_enumbox", returnValue IS_OK, ov en }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_enumbox__s1(en: ENUM__OBVTest__TestEnum)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_enumbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_enumbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_enumbox", iv en }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_enumbox__r1(en: ENUM__OBVTest__TestEnum)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_enumbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_enumbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_enumbox", returnValue IS_OK, ov en }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_arraybox__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_arraybox
Derivation Path	:
Comments	: Request-related ASP for the operation in_arraybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_arraybox", iv TestArrayBox__s1 }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_arraybox__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_arraybox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_arraybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_arraybox", returnValue IS_OK, ov TestArray__s1 }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_arraybox__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_arraybox
Derivation Path	:
Comments	: Request-related ASP for the operation out_arraybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_arraybox", iv TestArray__s1 }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_arraybox__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_arraybox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_arraybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_arraybox", returnValue IS_OK, ov TestArrayBox__s1 }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_fstructbox__s1(fst:STRUCT__OBVTest__TestFixStruct)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_fstructbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_fstructbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_fstructbox", iv fst }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_fstructbox__r1(fst:STRUCT__OBVTest__TestFixStruct)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_fstructbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_fstructbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_fstructbox", returnValue IS_OK, ov fst }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_fstructbox__s1(fst:STRUCT__OBVTest__TestFixStruct)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_fstructbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_fstructbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_fstructbox", iv fst }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_fstructbox__r1(fst:STRUCT__OBVTest__TestFixStruct)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_fstructbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_fstructbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_fstructbox", returnValue IS_OK, ov fst }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_vstructbox__s1(vst:STRUCT__OBVTest__TestVarStruct)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_vstructbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_vstructbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "in_vstructbox", iv vst }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_vstructbox__r1(vst:STRUCT__OBVTest__TestVarStruct)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_vstructbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_vstructbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "in_vstructbox", returnValue IS_OK, ov vst }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_vstructbox__s1(vst:STRUCT__OBVTest__TestVarStruct)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_vstructbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_vstructbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "out_vstructbox", iv vst }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_vstructbox__r1(vst:STRUCT__OBVTest__TestVarStruct)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_vstructbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_vstructbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_vstructbox", returnValue IS_OK, ov vst }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_funionbox__s1(fun:UNION__OBVTest__TestFixUnion)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_funionbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_funionbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_funionbox", iv fun }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_funionbox__r1(fun:UNION__OBVTest__TestFixUnion)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_funionbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_funionbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_funionbox", returnValue IS_OK, ov fun }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_funionbox__s1(fun:UNION__OBVTest__TestFixUnion)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_funionbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_funionbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_funionbox", iv fun }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_funionbox__r1(fun:UNION__OBVTest__TestFixUnion)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_funionbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_funionbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_funionbox", returnValue IS_OK, ov fun }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_vunionbox__s1(vun:UNION__OBVTest__TestVarUnion)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_vunionbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_vunionbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_vunionbox", iv vun }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_vunionbox__r1(vun:UNION__OBVTest__TestVarUnion)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_vunionbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_vunionbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "in_vunionbox", returnValue IS_OK, ov vun }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_vunionbox__s1(vun:UNION__OBVTest__TestVarUnion)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_vunionbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_vunionbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "out_vunionbox", iv vun }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_vunionbox__r1(vun:UNION__OBVTest__TestVarUnion)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_vunionbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_vunionbox
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "out_vunionbox", returnValue IS_OK, ov vun }	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_shseqbox__s1(shs: SEQ__OBVTest__TestShSeq)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_shseqbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_shseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_shseqbox", iv shs }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_shseqbox__r1(shs: SEQ__OBVTest__TestShSeq)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_shseqbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_shseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_shseqbox", returnValue IS_OK, ov shs }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_shseqbox__s1(shs: SEQ__OBVTest__TestShSeq)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_shseqbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_shseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_shseqbox", iv shs }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_shseqbox__r1(shs: SEQ__OBVTest__TestShSeq)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_shseqbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_shseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_shseqbox", returnValue IS_OK, ov shs }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_strseqbox__s1(shr: SEQ__OBVTest__TestStrSeq)
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_strseqbox
Derivation Path	:
Comments	: Request-related ASP for the operation in_strseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_strseqbox", iv shr }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_strseqbox__r1(shr: SEQ__OBVTest__TestStrSeq)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_strseqbox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_strseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_strseqbox", returnValue IS_OK, ov shr }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_strseqbox__s1(shr: SEQ__OBVTest__TestStrSeq)
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_strseqbox
Derivation Path	:
Comments	: Request-related ASP for the operation out_strseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_strseqbox", iv shr }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_strseqbox__r1(shr: SEQ__OBVTest__TestStrSeq)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_strseqbox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_strseqbox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_strseqbox", returnValue IS_OK, ov shr }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__in_anybox__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__in_anybox
Derivation Path	:
Comments	: Request-related ASP for the operation in_anybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_anybox", iv any_CORBA__Char "A" }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__in_anybox__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceC__in_anybox
Derivation Path	:
Comments	: Reply-related ASP for the operation in_anybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "in_anybox", returnValue IS_OK, ov any_CORBA__Char "A" }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__out_anybox__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__out_anybox
Derivation Path	:
Comments	: Request-related ASP for the operation out_anybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_anybox", iv any_CORBA__Char "A" }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__out_anybox__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceC__out_anybox
Derivation Path	:
Comments	: Reply-related ASP for the operation out_anybox
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "out_anybox", returnValue IS_OK, ov any_CORBA__Char "A" }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__val_in_struct__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__val_in_struct
Derivation Path	:
Comments	: Request-related ASP for the operation val_in_struct
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "val_in_struct", iv ?, nr ? }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_in_struct__r1(nm:CORBA__String;lg, g:CORBA__Long)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__val_in_struct
Derivation Path	:
Comments	: Reply-related ASP for the operation val_in_struct
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "val_in_struct", returnValue IS_OK, ov {nr lg, valmb {name nm, age g}} }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__val_out_struct__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__val_out_struct
Derivation Path	:
Comments	: Request-related ASP for the operation val_out_struct
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "val_out_struct", iv ? }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__val_out_struct__r1(vla:VALUETYPE__OBVTest__TestValueA)
ASP Type	: pREPLY__OBVTest__TestInterfaceC__val_out_struct
Derivation Path	:
Comments	: Reply-related ASP for the operation val_out_struct
Constraint Value	
<pre>{ oid "OBVTest::TestInterfaceC", opn "val_out_struct", returnValue IS_OK, ov vla }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__make_call__s1(cif_id, cop_id: CORBA__String)
ASP Type	: pCALL__OBVTest__TestHelper__make_call
Derivation Path	:
Comments	: Request-related ASP for the operation make_call on TestHelper.
Constraint Value	
<pre>{ oid "OBVTest::TestHelper", opn "make_call", cif cif_id, cop cop_id }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__make_call__r1(res: ENUM__OBVTest__Res)
ASP Type	: pREPLY__OBVTest__TestHelper__make_call
Derivation Path	:
Comments	: Reply-related ASP for the operation make_call on TestHelper.
Constraint Value	
<pre>{ oid "OBVTest::TestHelper", opn "make_call", returnValue NULL, rs res }</pre>	
Detailed Comments :	

ASN.1 ASP Constraint Declaration	
Constraint Name	: pCALL__hello__s1
ASP Type	: pCALL__OBVTest__TestInterfaceC__hello
Derivation Path	:
Comments	: Request-related ASP for the operation hello.
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "hello" }	
Detailed Comments	:

ASN.1 ASP Constraint Declaration	
Constraint Name	: pREPLY__hello__r1
ASP Type	: pREPLY__OBVTest__TestInterfaceC__hello
Derivation Path	:
Comments	: Reply-related ASP for the operation hello
Constraint Value	
{ oid "OBVTest::TestInterfaceC", opn "hello", returnValue IS_OK, istr "hello!" }	
Detailed Comments	:

IV

Dynamic Part

Test Case Dynamic Behaviour					
Test Case Name : Sub_intf_1 Group : Substitutability/ Purpose : To verify that in case a value instance is registered with the ORB, it is passed as an object reference to the operation parameter which is declared as an interface type supported by the value type. Configuration : Default : Default_A Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFA_NAME, IFA_NAME, refIFA)			NOTE 2
3		PCO_IFH ! pCALL__OBVTest__TestHelper__make_ call	pCALL__make_call__s1 (IFA_NAME, "val_sub_intf")		NOTE 4
4		START Timer1			NOTE 5
5		PCO_IFA ? pCALL__OBVTest__TestInterfaceA_ _val_sub_intf (refIFB:=pCALL__OBVTest__TestInte rfaceA__val_sub_intf.ifb) CANCEL Timer1	pCALL__val_sub_intf__s1(?)		NOTE 6
6		(boolResult:=is_valid_ref(refIFB))			
7		[boolResult=TRUE]		(P)	
8		PCO_IFA ! pREPLY__OBVTest__TestInterf aceA__val_sub_intf	pREPLY__val_sub_intf__r1 (IS_OK)		NOTE 7
9		START Timer1			NOTE 5
10		PCO_IFH ? pREPLY__OBVTest__TestH elper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	P	NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestH elper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	F	NOTE 8
12		?TIMEOUT Timer1		I	NOTE 10
13		[boolResult=FALSE]		(F)	
14		PCO_IFA ! pREPLY__OBVTest__TestInterf aceA__val_sub_intf	pREPLY__val_sub_intf__r1 (IS_NOK)		NOTE 7
15		START Timer1			NOTE 5
16		PCO_IFH ? pREPLY__OBVTest__TestH elper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	R	NOTE 8
17		PCO_IFH ? pREPLY__OBVTest__TestH elper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	R	NOTE 8
18		?TIMEOUT Timer1		R	NOTE 10
19		?TIMEOUT Timer1		I	NOTE 10

Continued on next page

*Continued from previous page***Test Case Dynamic Behaviour****Detailed Comments :** NOTE:

1. Get the reference of the TestHelper interface.
2. Create an object that emulates the interface supported by the test server.
3. Register the valuebox factory.
4. Send a request to TestHelper for initiation of an invocation on the operation under consideration.
5. Start Timer1.
6. Receive a request on the desired operation.
7. Send a reply of the invoked operation.
8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).
9. Unregister the valuebox factory.
10. Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : Sub_val_2 Group : Substitutability/ Purpose : To verify, when a value instance is passed to a parameter declared as the value type, in case that the receiving context holds an implementation, an instance of the value type is reconstructed at the receiving context by loading the implementation. Configuration : Default : Default_A Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFA_NAME, IFA_NAME, refIFA)			NOTE 2
3		+RegisterFactory(SRV_RCV,VLD_ID)			NOTE 3
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFA_NAME, "val_sub_val")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFA ? pCALL__OBVTest__TestInterfaceA __val_sub_val (vld_var:=pCALL__OBVTest__TestI nterfaceA__val_sub_val.val) CANCEL Timer1	pCALL__val_sub_val__s1		NOTE 6
7		(boolResult:=is_valid_vld(vld_var))			
8		[boolResult=TRUE]		(P)	
9		PCO_IFA ! pREPLY__OBVTest__TestInte rfaceA__val_sub_val	pREPLY__val_sub_val__r1(IS_OK)		NOTE 7
10		START Timer1			NOTE 5
11		PCO_IFH ? pREPLY__OBVTest__Test Helper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)		NOTE 8
12		+UnregisterFactory(SRV _RCV,VLA_ID)			NOTE 8
13		PCO_IFH ? pREPLY__OBVTest__Test Helper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)		NOTE 8
14		+UnregisterFactory(SRV _RCV,VLA_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_R CV,VLA_ID)			NOTE 8
17		[boolResult=FALSE]		(F)	
18		PCO_IFA ! pREPLY__OBVTest__TestInte rfaceA__val_sub_val	pREPLY__val_sub_val__r1(IS_NOK)		NOTE 7
19		START Timer1			NOTE 5
20		PCO_IFH ? pREPLY__OBVTest__Test Helper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)		NOTE 8

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
21		+UnregisterFactory(SRV_RCV,VLA_ID)			NOTE 8
22		PCO_IFH ? pREPLY__OBVTest__Test Helper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)		NOTE 8
23		+UnregisterFactory(SRV_RCV,VLA_ID)			NOTE 8
24		?TIMEOUT Timer1		(I)	NOTE 10
25		+UnregisterFactory(SRV_RCV,VLA_ID)			NOTE 8
26		?TIMEOUT Timer1		(I)	NOTE 10
27		+UnregisterFactory(SRV_RCV,VLD_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL). 9. Unregister the valuebox factory. 10. Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned.					

Test Case Dynamic Behaviour					
Test Case Name : ShortBox_in Group : ValueBox/ Purpose : To verify that an instance of a short valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,SHORTBOX _ID)			NOTE 3
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_shortbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_shortbox (num:=pCALL__OBVTest__TestInte rfaceC__in_shortbox.iv) CANCEL Timer1	pCALL__in_shortbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_shortbox	pREPLY__in_shortbox__r1 (num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,SHORTBOX_ID)			NOTE 9
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,SHORTBOX_ID)			NOTE 9
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,SHORTBOX_ID)			NOTE 9
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,SH ORTBOX_ID)			NOTE 9
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : ShortBox_out Group : ValueBox/ Purpose : To verify that an instance of a short valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,SHORTBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "out_shortbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_shortbox (num:=pCALL__OBVTest__TestInte rfaceC__out_shortbox.iv) CANCEL Timer1	pCALL__out_shortbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_shortbox	pREPLY__out_shortbox__r 1(num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,SHORTBOX_ID)			NOTE 9
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,SHORTBOX_ID)			NOTE 9
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,SHORTBOX_ID)			NOTE 9
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,SH ORTBOX_ID)			NOTE 9
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : UShortBox_in Group : ValueBox/ Purpose : To verify that an instance of a unsigned short valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,USHORTB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_ushortbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_ushortbox (num:=pCALL__OBVTest__TestInte rfaceC__in_ushortbox.iv) CANCEL Timer1	pCALL__in_ushortbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_ushortbox	pREPLY__in_ushortbox__r 1(num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,USHORTBOX_ID)			NOTE 9
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,USHORTBOX_ID)			NOTE 9
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,USHORTBOX_ID)			NOTE 9
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,US HORTBOX_ID)			NOTE 9
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : UShortBox_out Group : ValueBox/ Purpose : To verify that an instance of a unsigned short valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,USHORTBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_ushortbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_ushortbox (num:=pCALL__OBVTest__TestInte rfaceC__out_ushortbox.iv) CANCEL Timer1	pCALL__out_ushortbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_ushortbox	pREPLY__out_ushortbox__ r1(num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,USHORTBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,USHORTBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, USHORTBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,USH ORTBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : LongBox_in Group : ValueBox/ Purpose : To verify that an instance of a long valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV, LONGBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_longbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_longbox (num:=pCALL__OBVTest__TestInte rfaceC__in_longbox.iv) CANCEL Timer1	pCALL__in_longbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_longbox	pREPLY__in_longbox__r1(num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V, LONGBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V, LONGBOX_ID)			NOTE 9
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV , LONGBOX_ID)			NOTE 9
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV, LON GBOX_ID)			NOTE 9
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	<div>9. Unregister the valuebox factory.</div> <div>10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.</div> <div>Thus, the verdict INCONCLUSIVE (I) is assigned.</div>

Test Case Dynamic Behaviour					
Test Case Name : LongBox_out Group : ValueBox/ Purpose : To verify that an instance of a long valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV, LONGBOX_ ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_longbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_longbox (num:=pCALL__OBVTest__TestInte rfaceC__out_longbox.iv) CANCEL Timer1	pCALL__out_longbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_longbox	pREPLY__out_longbox__r1 (num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V, LONGBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V, LONGBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, LONGBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV, LON GBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : ULONGBox_in Group : ValueBox/ Purpose : To verify that an instance of a unsigned long valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,ULONGBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_ulongbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_ulongbox (num:=pCALL__OBVTest__TestInte rfaceC__in_ulongbox.iv) CANCEL Timer1	pCALL__in_ulongbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_ulongbox	pREPLY__in_ulongbox__r1 (num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,ULONGBOX_ID)			NOTE 9
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,ULONGBOX_ID)			NOTE 9
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,ULONGBOX_ID)			NOTE 9
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,ULO NGBOX_ID)			NOTE 9
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	<div>9. Unregister the valuebox factory.</div> <div>10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.</div> <div>Thus, the verdict INCONCLUSIVE (I) is assigned.</div>

Test Case Dynamic Behaviour					
Test Case Name : ULongBox_out Group : ValueBox/ Purpose : To verify that an instance of a unsigned long valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,ULONGBO X_ID)			NOTE 3
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_ulongbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_ulongbox (num:=pCALL__OBVTest__TestInte rfaceC__out_ulongbox.iv) CANCEL Timer1	pCALL__out_ulongbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_ulongbox	pREPLY__out_ulongbox__r 1(num)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,ULONGBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,ULONGBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, ULONGBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,ULO NGBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	<div>9. Unregister the valuebox factory.</div> <div>10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.</div> <div>Thus, the verdict INCONCLUSIVE (I) is assigned.</div>

Test Case Dynamic Behaviour					
Test Case Name : BooleanBox_in Group : ValueBox/ Purpose : To verify that an instance of a boolean valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,BOOLEAN BOX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_booleanbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_booleanbox (boolResult:=pCALL__OBVTest__Te stInterfaceC__in_booleanbox.iv) CANCEL Timer1	pCALL__in_booleanbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_booleanbox	pREPLY__in_booleanbox__ r1(boolResult)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,BOOLEANBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,BOOLEANBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,BOOLEANBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,BO OLEANBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	(PASS) if the result is positive, otherwise (FAIL). 9. Unregister the valuebox factory. 10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : BooleanBox_out Group : ValueBox/ Purpose : To verify that an instance of a boolean valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,BOOLEANB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_booleanbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_booleanbox (boolResult:=pCALL__OBVTest__Te stInterfaceC__out_booleanbox.iv) CANCEL Timer1	pCALL__out_booleanbox__ s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_booleanbox	pREPLY__out_booleanbox_ _r1(boolResult)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,BOOLEANBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,BOOLEANBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, BOOLEANBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,BOO LEANBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : OctetBox_in Group : ValueBox/ Purpose : To verify that an instance of an octet valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,OCTETBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_octetbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_octetbox (oct_var:=pCALL__OBVTest__TestI nterfaceC__in_octetbox.iv) CANCEL Timer1	pCALL__in_octetbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_octetbox	pREPLY__in_octetbox__r1 (oct_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,OCTETBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,OCTETBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,OCTETBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,OCT ETBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : OctetBox_out Group : ValueBox/ Purpose : To verify that an instance of an octet valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,OCTETBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "out_octetbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_octetbox (oct_var:=pCALL__OBVTest__TestI nterfaceC__out_octetbox.iv) CANCEL Timer1	pCALL__out_octetbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_octetbox	pREPLY__out_octetbox__r 1(oct_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,OCTETBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,OCTETBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, OCTETBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,OCT ETBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : CharBox_in Group : ValueBox/ Purpose : To verify that an instance of a char valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,CHARBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_charbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_charbox (char_var:=pCALL__OBVTest__Tes tInterfaceC__in_charbox.iv) CANCEL Timer1	pCALL__in_charbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_charbox	pREPLY__in_charbox__r1(char_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,CHARBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,CHARBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,CHARBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,CH ARBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : CharBox_out Group : ValueBox/ Purpose : To verify that an instance of a char valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,CHARBOX_ ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_charbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_charbox (char_var:=pCALL__OBVTest__Tes tInterfaceC__out_charbox.iv) CANCEL Timer1	pCALL__out_charbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_charbox	pREPLY__out_charbox__r1 (char_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,CHARBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,CHARBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, CHARBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,CHA RBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FloatBox_in Group : ValueBox/ Purpose : To verify that an instance of a float valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,FLOATBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_floatbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_floatbox (float_var:=pCALL__OBVTest__Tes tInterfaceC__in_floatbox.iv) CANCEL Timer1	pCALL__in_floatbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_floatbox	pREPLY__in_floatbox__r1(float_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,FLOATBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,FLOATBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,FLOATBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,FLO ATBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FloatBox_out Group : ValueBox/ Purpose : To verify that an instance of a float valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,FLOATBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_floatbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_floatbox (float_var:=pCALL__OBVTest__Tes tInterfaceC__out_floatbox.iv) CANCEL Timer1	pCALL__out_floatbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_floatbox	pREPLY__out_floatbox__r 1(float_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,FLOATBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,FLOATBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, FLOATBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,FLO ATBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : DoubleBox_in Group : ValueBox/ Purpose : To verify that an instance of a double valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,DOUBLEB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_doublebox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_doublebox (double_var:=pCALL__OBVTest__T estInterfaceC__in_doublebox.iv) CANCEL Timer1	pCALL__in_doublebox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_doublebox	pREPLY__in_doublebox__r 1(double_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,DOUBLEBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,DOUBLEBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,DOUBLEBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,DO UBLEBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : DoubleBox_out Group : ValueBox/ Purpose : To verify that an instance of a double valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,DOUBLEBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_doublebox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_doublebox (double_var:=pCALL__OBVTest__T estInterfaceC__out_doublebox.iv) CANCEL Timer1	pCALL__out_doublebox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_doublebox	pREPLY__out_doublebox__ r1(double_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,DOUBLEBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,DOUBLEBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, DOUBLEBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,DOU BLEBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : StringBox_in Group : ValueBox/ Purpose : To verify that an instance of a string valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,STRINGBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_stringbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_stringbox (str_var:=pCALL__OBVTest__TestI nterfaceC__in_stringbox.iv) CANCEL Timer1	pCALL__in_stringbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_stringbox	pREPLY__in_stringbox__r1 (str_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,STRINGBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,STRINGBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,STRINGBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,STRI NGBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : StringBox_out Group : ValueBox/ Purpose : To verify that an instance of a string valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,STRINGBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_stringbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_stringbox (str_var:=pCALL__OBVTest__TestI nterfaceC__out_stringbox.iv) CANCEL Timer1	pCALL__out_stringbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_stringbox	pREPLY__out_stringbox__ r1(str_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,STRINGBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,STRINGBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, STRINGBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,STRI NGBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : EnumBox_in Group : ValueBox/ Purpose : To verify that an instance of a enum valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,ENUMBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_enumbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_enumbox (enum_var:=pCALL__OBVTest__Te stInterfaceC__in_enumbox.iv) CANCEL Timer1	pCALL__in_enumbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_enumbox	pREPLY__in_enumbox__r1 (enum_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,ENUMBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,ENUMBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,ENUMBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,EN UMBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : EnumBox_out Group : ValueBox/ Purpose : To verify that an instance of a enum valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,ENUMBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "out_enumbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_enumbox (enum_var:=pCALL__OBVTest__Te stInterfaceC__out_enumbox.iv) CANCEL Timer1	pCALL__out_enumbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_enumbox	pREPLY__out_enumbox__r 1(enum_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,ENUMBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,ENUMBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, ENUMBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,ENU MBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FixStructBox_in Group : ValueBox/ Purpose : To verify that an instance of a fix struct valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,FSTRUCTB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "in_fstructbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_fstructbox (fst_var:=pCALL__OBVTest__TestI nterfaceC__in_fstructbox.iv) CANCEL Timer1	pCALL__in_fstructbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_fstructbox	pREPLY__in_fstructbox__r 1(fst_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,FSTRUCTBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,FSTRUCTBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,FSTRUCTBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,FST RUCTBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FixStructBox_out Group : ValueBox/ Purpose : To verify that an instance of a fix struct valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,FSTRUCTB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_fstructbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_fstructbox (fst_var:=pCALL__OBVTest__TestI nterfaceC__out_fstructbox.iv) CANCEL Timer1	pCALL__out_fstructbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_fstructbox	pREPLY__out_fstructbox_ _r1(fst_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,FSTRUCTBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,FSTRUCTBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, FSTRUCTBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,FST RUCTBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : VarStructBox_in Group : ValueBox/ Purpose : To verify that an instance of a variable struct valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,VSTRUCTB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_vstructbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_vstructbox (vst_var:=pCALL__OBVTest__TestI nterfaceC__in_vstructbox.iv) CANCEL Timer1	pCALL__in_vstructbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_vstructbox	pREPLY__in_vstructbox__r 1(vst_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,VSTRUCTBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,VSTRUCTBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,VSTRUCTBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,VST RUCTBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : VarStructBox_out Group : ValueBox/ Purpose : To verify that an instance of a variable struct valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, reflFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, reflFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,VSTRUCTB OX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "out_vstructbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_vstructbox (vst_var:=pCALL__OBVTest__TestI nterfaceC__out_vstructbox.iv) CANCEL Timer1	pCALL__out_vstructbox__ s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_vstructbox	pREPLY__out_vstructbox__ r1(vst_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,VSTRUCTBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,VSTRUCTBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, VSTRUCTBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,VST RUCTBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FixUnionBox_in Group : ValueBox/ Purpose : To verify that an instance of a fix union valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,FUNIONBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_funionbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_funionbox (fun_var:=pCALL__OBVTest__TestI nterfaceC__in_funionbox.iv) CANCEL Timer1	pCALL__in_funionbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_funionbox	pREPLY__in_funionbox__r 1(fun_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,FUNIONBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,FUNIONBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,FUNIONBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,FUN IONBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : FixUnionBox_out Group : ValueBox/ Purpose : To verify that an instance of a fix union valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, reflFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, reflFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,FUNIONBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_funionbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_funionbox (fun_var:=pCALL__OBVTest__TestI nterfaceC__out_funionbox.iv) CANCEL Timer1	pCALL__out_funionbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_funionbox	pREPLY__out_funionbox__ r1(fun_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,FUNIONBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,FUNIONBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, FUNIONBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,FUN IONBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : VarUnionBox_in Group : ValueBox/ Purpose : To verify that an instance of a variable union valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,VUNIONBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_vunionbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_vunionbox (vun_var:=pCALL__OBVTest__Test InterfaceC__in_vunionbox.iv) CANCEL Timer1	pCALL__in_vunionbox__s1 (?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_vunionbox	pREPLY__in_vunionbox__r 1(vun_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,VUNIONBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,VUNIONBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,VUNIONBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,VU NIONBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : VarUnionBox_out Group : ValueBox/ Purpose : To verify that an instance of a variable union valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, reflFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, reflFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,VUNIONBO X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "out_vunionbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_vunionbox (vun_var:=pCALL__OBVTest__Test InterfaceC__out_vunionbox.iv) CANCEL Timer1	pCALL__out_vunionbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_vunionbox	pREPLY__out_vunionbox_ _r1(vun_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,VUNIONBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,VUNIONBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, VUNIONBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,VUN IONBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : ShortSeqBox_in Group : ValueBox/ Purpose : To verify that an instance of a short sequence valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,SHSEQBOX X_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make __call	pCALL__make_call__s1 (IFC_NAME, "in_shseqbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __in_shseqbox (shs_var:=pCALL__OBVTest__Test InterfaceC__in_shseqbox.iv) CANCEL Timer1	pCALL__in_shseqbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__in_shseqbox	pREPLY__in_shseqbox__r1 (shs_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RC V,SHSEQBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RC V,SHSEQBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV ,SHSEQBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,SHS EQBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : ShortSeqBox_out Group : ValueBox/ Purpose : To verify that an instance of a short sequence valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,SHSEQBOX _ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "out_shseqbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __out_shseqbox (shs_var:=pCALL__OBVTest__Test InterfaceC__out_shseqbox.iv) CANCEL Timer1	pCALL__out_shseqbox__s 1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__out_shseqbox	pREPLY__out_shseqbox__ r1(shs_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,SHSEQBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,SHSEQBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, SHSEQBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,SHS EQBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : StrSeqBox_in Group : ValueBox/ Purpose : To verify that an instance of a string sequence valuebox is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(SRV_RCV,STRSEQBOX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make__call	pCALL__make_call__s1 (IFC_NAME, "in_strseqbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC__in_strseqbox (shr_var:=pCALL__OBVTest__TestInterfaceC__in_strseqbox.iv) CANCEL Timer1	pCALL__in_strseqbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfaceC__in_strseqbox	pREPLY__in_strseqbox__r1(shr_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHelper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(SRV_RCV,STRSEQBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHelper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(SRV_RCV,STRSEQBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(SRV_RCV,STRSEQBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(SRV_RCV,STRSEQBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : StrSeqBox_out Group : ValueBox/ Purpose : To verify that an instance of a string sequence valuebox is passed as out parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,STRSEQBOX_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make__call	pCALL__make_call__s1 (IFC_NAME, "out_strseqbox")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC__out_strseqbox (shr_var:=pCALL__OBVTest__TestInterfaceC__out_strseqbox.iv) CANCEL Timer1	pCALL__out_strseqbox__s1(?)		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfaceC__out_strseqbox	pREPLY__out_strseqbox__r1(shr_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHelper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RCV,STRSEQBOX_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHelper__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RCV,STRSEQBOX_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV,STRSEQBOX_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,STRSEQBOX_ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments : ...	
	9. Unregister the valuebox factory.
	10.Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test.
	Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : Val_out_struct Group : ValueBox/ Purpose : To verify that an instance of a struct with value member is passed as in parameter corretly. Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		+RegisterFactory(CLT_RCV,VLA_ID)			NOTE 2
4		PCO_IFH ! pCALL__OBVTest__TestHelper__make _call	pCALL__make_call__s1 (IFC_NAME, "val_out_struct")		NOTE 4
5		START Timer1			NOTE 5
6		PCO_IFC ? pCALL__OBVTest__TestInterfaceC __val_out_struct (vla_var:=pCALL__OBVTest__TestI nterfaceC__val_out_struct.iv.valmb) CANCEL Timer1	pCALL__val_out_struct__s 1		NOTE 6
7		PCO_IFC ! pREPLY__OBVTest__TestInterfac eC__val_out_struct	pREPLY__val_out_struct__ r1(vla_var)		NOTE 7
8		START Timer1			NOTE 5
9		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	(P)	NOTE 8
10		+UnregisterFactory(CLT_RC V,VLA_ID)			NOTE 8
11		PCO_IFH ? pREPLY__OBVTest__TestHel per__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	(F)	NOTE 8
12		+UnregisterFactory(CLT_RC V,VLA_ID)			NOTE 8
13		?TIMEOUT Timer1		(I)	NOTE 10
14		+UnregisterFactory(CLT_RCV, VLA_ID)			NOTE 8
15		?TIMEOUT Timer1		(I)	NOTE 10
16		+UnregisterFactory(CLT_RCV,VLA _ID)			NOTE 8
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL).					

Continued on next page

Continued from previous page

Test Case Dynamic Behaviour	
Detailed Comments :	... 9. Unregister the valuebox factory. 10. Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned.

Test Case Dynamic Behaviour					
Test Case Name : Hello Group : Purpose : Configuration : Default : Default_C Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		+ObjtoPCO (PCOIFH_NAME, IFH_NAME, refIFH)			NOTE 1
2		+CreateObj(PCOIFC_NAME, IFC_NAME, refIFC)			NOTE 2
3		PCO_IFH ! pCALL__OBVTest__TestHelper__make__ call	pCALL__make_call__s1 (IFC_NAME, "hello")		NOTE 4
4		START Timer1			NOTE 5
5		PCO_IFC ? pCALL__OBVTest__TestInterfaceC__ _hello CANCEL Timer1	pCALL__hello__s1		NOTE 6
6		PCO_IFC ! pREPLY__OBVTest__TestInterface C__hello	pREPLY__hello__r1		NOTE 7
7		START Timer1			NOTE 5
8		PCO_IFH ? pREPLY__OBVTest__TestHelpe r__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_OK)	P	NOTE 8
9		PCO_IFH ? pREPLY__OBVTest__TestHelpe r__make_call CANCEL Timer1	pREPLY__make_call__r1 (IS_NOK)	F	NOTE 8
10		?TIMEOUT Timer1		I	NOTE 10
11		?TIMEOUT Timer1		I	NOTE 10
Detailed Comments : NOTE: 1. Get the reference of the TestHelper interface. 2. Create an object that emulates the interface supported by the test server. 3. Register the valuebox factory. 4. Send a request to TestHelper for initiation of an invocation on the operation under consideration. 5. Start Timer1. 6. Receive a request on the desired operation. 7. Send a reply of the invoked operation. 8. Receive the indication of the result of operation invocation from TestHelper. Assign verdict (PASS) if the result is positive, otherwise (FAIL). 9. Unregister the valuebox factory. 10. Timeout of Timer1 occurs, which does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned.					

Test Step Dynamic Behaviour					
Test Step Name : ObjtoPCO(pcoID: GraphicString; objectName: GraphicString; ref: CORBA__Object) Group : Objective : Default : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(ref := getObjRef(objectName))			Note 1
2		(boolResult:= activatePCOtoSUTServer(pcoID, objectName, ref))			Note 2
3		[boolResult = TRUE]		(P)	Note 3
4		[boolResult = FALSE]		I	Note 4
Detailed Comments : Note: 1. Get reference of object objectName. 2. Associate the PCO pcoID with the object. 3. Continue in case of successful operation in 2. 4. Otherwise, the test is terminated with verdict INCONCLUSIVE (I).					

Test Step Dynamic Behaviour					
Test Step Name : CreateObj(pcoID: GraphicString; objectName: GraphicString; ref: CORBA__Object) Group : Objective : Default : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(boolResult:= activatePCOtoSUTClient(pcoID, objectName, ref))			Note 1
2		[boolResult = TRUE]		(P)	Note 2
3		[boolResult = FALSE]		I	Note 3
Detailed Comments : Note: 1. Create the object with objectName and associate it with the PCO pcoID. 2. Continue in case of successful operation in 2. 3. Otherwise, the test is terminated with verdict INCONCLUSIVE (I).					

Test Step Dynamic Behaviour					
Test Step Name : RegisterFactory(rcv:BOOLEAN; rid: GraphicString) Group : Objective : Default : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(boolResult:= registerValueFactory(rcv, rid))			Note 1
2		[boolResult = TRUE]		(P)	Note 2
3		[boolResult = FALSE]		I	Note 3
Detailed Comments : Note: 1. Create a factory for the value with rid and register it with the ORB. 2. Continue in case of successful operation in 1. 3. Otherwise, the test is terminated with verdict INCONCLUSIVE (I).					

Test Step Dynamic Behaviour					
Test Step Name : UnregisterFactory(rcv:BOOLEAN; rid: GraphicString) Group : Objective : Default : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(boolResult:= unregisterValueFactory(rcv, rid))			Note 1
2		[boolResult = TRUE]		R	Note 2
3		[boolResult = FALSE]		R	Note 2
Detailed Comments : Note: 1. Unregister a factory for the value with rid with the ORB. 2. Final verdict in either case.					

Default Dynamic Behaviour					
Default Name : Default_A Group : Objective : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		PCO_IFA ? pRAISE__CORBA__COMM_FAILURE	COMM_FAILURE__r1	I	NOTE 1
2		PCO_IFA ? pRAISE__CORBA__TRANSIENT	TRANSIENT__r1	I	NOTE 1
3		PCO_IFA ? pRAISE__CORBA__UNKNOWN	UNKNOWN__r1	I	NOTE 1
4		PCO_IFA ? OTHERWISE		F	NOTE 2
Detailed Comments : NOTE: 1. Receive of CORBA system exceptions that does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned. 2. Receive of test event that implies invalid behavior of the system under test. The verdict FAIL (F) is assigned.					

Default Dynamic Behaviour					
Default Name : Default_C Group : Objective : Comments :					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		PCO_IFH ? OTHERWISE		I	NOTE 1
2		PCO_IFC ? pRAISE__CORBA__COMM_FAILURE	COMM_FAILURE__r1	I	NOTE 1
3		PCO_IFC ? pRAISE__CORBA__TRANSIENT	TRANSIENT__r1	I	NOTE 1
4		PCO_IFC ? pRAISE__CORBA__UNKNOWN	UNKNOWN__r1	I	NOTE 1
5		PCO_IFC ? OTHERWISE		F	NOTE 2
Detailed Comments : NOTE: 1. Receive of CORBA system exceptions that does not necessarily leads to failure of the system under test. Thus, the verdict INCONCLUSIVE (I) is assigned. 2. Receive of test event that implies invalid behavior of the system under test. The verdict FAIL (F) is assigned.					