

```
#####  
##  
##          OSS/J Trouble Ticket API (JSR 91)      ##  
##          Version 1.2                          ##  
##  
## Questionnaire for Self-Certified Products - Client Side ##  
##          Version v.1.6 - Oct 23, 2009          ##  
##  
#####
```

Company: **Systems Mechanics Limited**
Product: **CrossTalk™ Trouble Ticketing OSS/J Client Adapter for HP TeMIP**
Version: **1.0**

G.01

Have you implemented the product using Java technology? If not, which technologies have been used?

Yes Other

Comments:

G.02

Is the product tested against the Reference Implementation?

Yes No Not Applicable

Comments:

G.03

Is the product tested against certified products? If yes, list for each product the company name, product name, version, and testing procedure.

Yes No Not Applicable

Company: Ascom
Product: ARS over OSS/J
Version: 2.1
Testing Procedure: Verify ability to create, update, query, cancel, close trouble tickets with remote trouble ticketing system via OSS/J

Comments: Ascom adapter used to provide server implementation of JSR91 for integration of fault management (TeMIP) to trouble ticketing (ARS Remedy)

G.04

Which integration profile is used by your product?

JVT XML/JMS WS

Comments:

G.05

Is the product client of a JVT implementation of the API? If yes list the company name, product name, and version of all application servers the product has been tested against.

Yes No

Company:

Product:

Version:

Comments:

G.06

Is the product client of a XML/JMS implementation of the API? If yes list the company name, product name, and version of all JMS providers the product has been tested against.

Yes No

Company: Ascom

Product: ARS over OSS/J

Version: 2.1

Comments: Ascom adapter integrated through Sun Enterprise Service Bus

G.07

Is the product client of a Web Services implementation of the API? If yes list the company name, product name, and version of all server side deployment technologies the product has been tested against.

Yes No

Company:
Product:
Version:
Comments:

G.08

Does the product consist of EJBs that run in the same application server(s) of the OSS/J API implementation? If yes, please list the company name, product name, and version of all application servers supporting the product.

Yes No

Company:
Product:
Version:
Comments:

G.09

Does the product use the full API? If not please list features not used (or used, if only few)

Yes No

(The method list below is taken from the JVT Session Interface just to identify the methods, the list still applies to the XML/JMS and Web Services integration profile):

cancelTroubleTicketByKey	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
cancelTroubleTicketsByKeys	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
cancelTroubleTicketsByTemplate	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
cancelTroubleTicketsByTemplates	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No

Comments:		
closeTroubleTicketByKey	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
closeTroubleTicketsByKeys	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
closeTroubleTicketsByTemplate	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
closeTroubleTicketsByTemplates	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
createTroubleTicketByValue	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
createTroubleTicketItemByValue	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
createTroubleTicketItemsByValues	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
createTroubleTicketsByValues	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketByKey	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
getTroubleTicketItemByKey	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketItemsByKeys	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketItemsByTemplate	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketItemsByTemplates	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketItemTypes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketsByKeys	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
getTroubleTicketsByTemplate	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketsByTemplates	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
getTroubleTicketTypes	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
makeTroubleTicketItemValue	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
makeTroubleTicketValue	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
removeTroubleTicketItemByKey	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		

removeTroubleTicketItemsByKeys	[] Yes	[X] No
Comments:		
setTroubleTicketByValue	[X] Yes	[] No
Comments:		
setTroubleTicketItemByValue	[] Yes	[X] No
Comments:		
setTroubleTicketItemsByKeys	[] Yes	[X] No
Comments:		
setTroubleTicketItemsByTemplate	[] Yes	[X] No
Comments:		
setTroubleTicketItemsByTemplates	[] Yes	[X] No
Comments:		
setTroubleTicketItemsByValues	[] Yes	[X] No
Comments:		
setTroubleTicketsByKeys	[] Yes	[X] No
Comments:		
setTroubleTicketsByTemplate	[] Yes	[X] No
Comments:		
setTroubleTicketsByTemplates	[] Yes	[X] No
Comments:		
setTroubleTicketsByValues	[] Yes	[X] No
Comments:		
tryCancelTroubleTicketsByKeys	[] Yes	[X] No
Comments:		
tryCancelTroubleTicketsByTemplate	[] Yes	[X] No
Comments:		
tryCancelTroubleTicketsByTemplates	[] Yes	[X] No
Comments:		
tryCloseTroubleTicketsByKeys	[] Yes	[X] No
Comments:		
tryCloseTroubleTicketsByTemplate	[] Yes	[X] No
Comments:		
tryCloseTroubleTicketsByTemplates	[] Yes	[X] No
Comments:		
tryCreateTroubleTicketItemsByValues	[] Yes	[X] No
Comments:		
tryCreateTroubleTicketsByValues	[] Yes	[X] No
Comments:		
tryRemoveTroubleTicketItemsByKeys	[] Yes	[X] No
Comments:		
trySetTroubleTicketItemsByKeys	[] Yes	[X] No
Comments:		
trySetTroubleTicketItemsByTemplate	[] Yes	[X] No

Comments:
trySetTroubleTicketItemsByTemplates [] Yes No
Comments:
trySetTroubleTicketItemsByValues [] Yes No
Comments:
trySetTroubleTicketsByKeys [] Yes No
Comments:
trySetTroubleTicketsByTemplate [] Yes No
Comments:
trySetTroubleTicketsByTemplates [] Yes No
Comments:
trySetTroubleTicketsByValues [] Yes No
Comments:

Comments: The product supports the ability to create, update, query, cancel, and close trouble tickets and implements only the API required to satisfy this functionality

G.10

Does the product use extended features of the API? If yes please list features and extensions, and add own extensions if necessary.

[] Yes No

Comments:

G.11

Does the product support extensibility as defined by the OSS/J Design Guidelines? Can the product handle types that are specific to an implementation of the API, but derived from the API types.

Yes [] No

Comments:

G.12

For each event defined by the API, please indicate how the event is handled: a) event is ignored; b) event is simply displayed; c) event is used as input for further processing; please explain how it is processed.

`TroubleTicketAttributeValueChangeEvent` Ignored Displayed Processed
Comments:

`TroubleTicketCancellationEvent` Ignored Displayed Processed

Comments: When a cancellation event is received all alarms associated to the ticket are un-acknowledged and un-handled in TeMIP and Case object updated

`TroubleTicketCloseOutEvent` Ignored Displayed Processed

Comments: When a close event is received all alarms associated to the ticket are un-acknowledged and un-handled in TeMIP and Case object updated

`TroubleTicketCreateEvent` Ignored Displayed Processed

Comments: When a create event is received all alarms associated to the ticket are acknowledged, handled, and annotated

`TroubleTicketStateChangeEvent` Ignored Displayed Processed
Comments:

`TroubleTicketItemAttributeValueChangeEvent` Ignored Displayed Processed
Comments:

`TroubleTicketItemCreateEvent` Ignored Displayed Processed
Comments:

`TroubleTicketItemRemoveEvent` Ignored Displayed Processed
Comments:

G.13

Is the product able to handle all exceptions thrown by the used features of the API?

Yes No

`java.rmi.RemoteException` Yes No
Comments:

`javax.oss.OssIllegalArgumentException` Yes No
Comments:

javax.ejb.FinderException	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.OssUnsupportedOperationException	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
java.lang.IllegalStateException	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.trouble.CancelException	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.trouble.CloseException	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		

Comments: All exceptions are logged and if relevant displayed to the operator

G.14

Is the product able to handle all return types? Please list for every return type if the product is able to handle it.

javax.oss.cbe.trouble.TroubleTicketValue	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.cbe.trouble.TroubleTicketValue[]	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.cbe.trouble.TroubleTicketValueIterator	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.cbe.trouble.TroubleTicketKeyResult	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Comments:		
javax.oss.cbe.trouble.TroubleTicketItemValue	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments: Not required by the product		
javax.oss.cbe.trouble.TroubleTicketItemValue[]	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments: Not required by the product		
javax.oss.cbe.trouble.TroubleTicketItemValueIterator	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments: Not required by the product		
javax.oss.cbe.trouble.TroubleTicketItemKeyResult	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments: Not required by the product		

G.15

How does the product cope with XSD extensions on the server?

Comments: The adapter supports extensions in two ways:

- If a response or notification event is received with unsupported attributes (extensions) these are ignored, and the message is still correctly processed;
- The adapter's Mediation and Mapping Engine component can be configured through the use of Mapping Rules to support extensions and populate/translate new attributes on requests, responses and notifications
-

G.16

Does the product add CharacteristicSpecifications to any entity (extending from CBEManagedEntityValue)? If yes, please list name, type and the circumstances, when it is added.

Yes No

Name:

Type:

Usage:

Comments:

S.01

Does the product use named queries? If yes please list them

AllOpenTroubleTicketsQuery	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
AllTroubleTicketItemsRelatedToTroubleTicketQuer	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
AllTroubleTicketsRelatedToTroubleTicketItemQuery	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
TroubleTicketItemsRelatedToTroubleTicketQuery	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Comments:		
TroubleTicketsRelatedToTroubleTicketItemQuery	<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No

Comments:
<please list extension specific queries> Yes No

Comments: Not required by the product, query of a single ticket is performed using getTroubleticketByKeyRequest operation

S.02

Is the product able to handle item lists attached to tickets?

Yes No Not Applicable

Comments: The product maintains synchronisation between the relatedAlarmList of a ticket and the related alarms in TeMIP

S.03

Does the product fully support the OSS/J Trouble Ticket state model?

Yes No Not Applicable

<if not, please specify the subset used>

Comments:

#####