

Change Log for OSS Trouble Ticket API version 1.1

OSS through Java™ Initiative

Axel Terfloeth, IP VALUE GmbH.

TT-API-SPEC_change_log.1.1.4.doc

Copyright © 2006 IP VALUE GmbH. All rights reserved. Use is subject to license terms.

Executive Summary

This document summarizes the changes to the OSS Trouble Ticket API (JSR 91) specification Version 1.1. The main purpose of this version is

- follow the new Draft of the OSS/J Design Guideline v1.2
- adapt the OSS/J Common API v1.4
- remove the ITU-T X.790 based Trouble Ticket Value Type definitions from the specification
- use of the trouble ticketing related Core Business Entities (CBE) interface definitions as defined in the OSS Common API (JSR 144) v1.4

There are four lists of changes:

- "proposed" changes are those modifications that are included in OSS Trouble Ticket API version 1.1.
- "accepted" changes are those modifications that are included in OSS Trouble Ticket API version 1.1. as result of JCP Maintenance Draft Review closed on Jan. 9, 2006
- "rejected" changes are those modifications that will not be included in OSS Trouble Ticket API version 1.1.
- "deferred" changes are those modifications that are not included in OSS Trouble Ticket API version 1.1, whether for time reasons or because it was considered that the changes were too significant.

The detailed description of changes in this document is principally of interest to people implementing the OSS Trouble Ticket API specification.

Table of Contents

Executive Summary	2
Table of Contents	3
1 Preface	6
1.1 Objectives	6
1.2 Audience	6
1.3 Approval and Distribution	6
1.4 Related Information	6
1.5 Revision History	7
2 Summary of changes	8
3 Proposed Changes	9
3.1.1 Using newly defined response types	9
3.1.2 New methods	9
3.1.2.1 getTroubleTicketItemTypes	9
3.1.2.2 makeTroubleTicketItemValue	9
3.1.2.3 getTroubleTicketItemByKey	10
3.1.2.4 getTroubleTicketItemsByKeys	10
3.1.2.5 getTroubleTicketItemsByTemplate	10
3.1.2.6 getTroubleTicketItemsByTemplates	11
3.1.2.7 createTroubleTicketItemByValue	11
3.1.2.8 createTroubleTicketItemsByValues	12
3.1.2.9 tryCreateTroubleTicketItemsByValues	12
3.1.2.10 setTroubleTicketItemByValue	12
3.1.2.11 setTroubleTicketItemsByValues	13
3.1.2.12 trySetTroubleTicketItemsByValues	13
3.1.2.13 setTroubleTicketItemsByKeys	14
3.1.2.14 trySetTroubleTicketItemsByKeys	14
3.1.2.15 setTroubleTicketItemsByTemplate	14
3.1.2.16 trySetTroubleTicketItemsByTemplate	15
3.1.2.17 setTroubleTicketItemsByTemplates	15
3.1.2.18 trySetTroubleTicketItemsByTemplates	16
3.1.2.19 removeTroubleTicketItemByKey	16
3.1.2.20 removeTroubleTicketItemsByKeys	16
3.1.2.21 tryRemoveTroubleTicketItemsByKeys	17
3.2 Named Queries and Responses	17
3.2.1 Changes to QueryAllOpenTroubleTicketsValue	17
3.2.2 Added QueryAllTroubleTicketItemsRelatedToTroubleTicket	18
3.2.3 Added QueryTroubleTicketItemsRelatedToTroubleTicket	19
3.2.4 Added QueryTroubleTicketsResponse	20
3.2.5 Added QueryTroubleTicketItemsResponse	20
3.3 Changes to Event Definitions	21
3.3.1 Removed TroubleTicketStatusChangeEvent	21
3.3.2 Changed TroubleTicketCloseOutEvent	21

3.3.3	Changed TroubleTicketCancellationEvent	21
3.3.4	Modified TroubleTicketAttributeValueChangeEvent	21
3.3.5	Added TroubleTicketStateChangeEvent	22
3.3.6	Added TroubleTicketItemCreateEvent	24
3.3.7	Added TroubleTicketItemAttributeValueChangeEvent	25
3.3.8	Added TroubleTicketItemRemoveEvent	26
4	Accepted Changes	27
4.1	<i>Replace X.790 trouble ticket data model by CBE trouble ticket data model</i>	27
4.1.1	Remove X.790 based interfaces and related data types	27
4.2	<i>Adopt OSS Common API / CBE v1.4</i>	28
4.2.1	Introduce new named query pattern	28
4.2.2	Introduce update procedures technically	28
4.2.3	Using newly defined exceptions	29
4.3	<i>Changes to javax.oss.trouble.JVTTroubleTicketSession</i>	29
4.3.1	Removed Methods	30
4.3.2	Changed Methods	30
4.3.2.1	Changes to makeTroubleTicketValue	30
4.3.2.2	Changes to getTroubleTicketByKey	31
4.3.2.3	Changes to getTroubleTicketsByKeys	31
4.3.2.4	Changes to createTroubleTicketByValue	32
4.3.2.5	Changes to createTroubleTicketsByValues	33
4.3.2.6	Changes to setTroubleTicketByValue	33
4.3.2.7	Changes to setTroubleTicketsByValues	34
4.3.2.8	Changes to setTroubleTicketsByKeys	35
4.3.2.9	Changes to setTroubleTicketsByTemplate	36
4.3.2.10	Changes to setTroubleTicketsByTemplates	37
4.3.2.11	Changes to closeTroubleTicketByKey	37
4.3.2.12	Changes to closeTroubleTicketsByKeys	38
4.3.2.13	Changes to closeTroubleTicketsByTemplate	39
4.3.2.14	Changes to closeTroubleTicketsByTemplates	40
4.3.2.15	Changes to cancelTroubleTicketByKey	41
4.3.2.16	Changes to cancelTroubleTicketsByKeys	42
4.3.2.17	Changes to cancelTroubleTicketsByTemplates	43
4.3.2.18	Changes to getTroubleTicketsByTemplate	43
4.3.2.19	Changes to getTroubleTicketsByTemplates	44
4.3.2.20	Changes to tryCreateTroubleTicketsByValues	45
4.3.2.21	Changes to trySetTroubleTicketsByValues	45
4.3.2.22	Changes to trySetTroubleTicketsByKeys	46
4.3.2.23	Changes to trySetTroubleTicketsByTemplate	47
4.3.2.24	Changes to trySetTroubleTicketsByTemplates	47
4.3.2.25	Changes to tryCloseTroubleTicketsByKeys	48
4.3.2.26	Changes to tryCloseTroubleTicketsByTemplate	49
4.3.2.27	Changes to tryCloseTroubleTicketsByTemplates	50
4.3.2.28	Changes to tryCancelTroubleTicketsByKeys	50
4.3.2.29	Changes to tryCancelTroubleTicketsByTemplate	51
4.3.2.30	Changes to tryCancelTroubleTicketsByTemplates	52
5	Rejected Changes	54
5.1	<i>Changes to javax.oss.trouble.JVTTroubleTicketSession</i>	54
5.1.1	Changed Methods	54
5.1.1.1	Changes to queryTroubleTickets	54
5.1.1.2	Changes to getTroubleTicketsByTemplate	54

5.1.1.3	Changes to getTroubleTicketsByTemplates	55
5.1.1.4	Changes to tryCreateTroubleTicketsByValues	55
5.1.1.5	Changes to trySetTroubleTicketsByValues	56
5.1.1.6	Changes to trySetTroubleTicketsByKeys	56
5.1.1.7	Changes to trySetTroubleTicketsByTemplate	57
5.1.1.8	Changes to trySetTroubleTicketsByTemplates	58
5.1.1.9	Changes to tryCloseTroubleTicketsByKeys	58
5.1.1.10	Changes to tryCloseTroubleTicketsByTemplate	59
5.1.1.11	Changes to tryCloseTroubleTicketsByTemplates	60
5.1.1.12	Changes to tryCancelTroubleTicketsByKeys	60
5.1.1.13	Changes to tryCancelTroubleTicketsByTemplate	61
5.1.1.14	Changes to tryCancelTroubleTicketsByTemplates	62
6	Deferred Changes from OSS Trouble Ticket API v1.0	63

1 Preface

1.1 Objectives

This document lists all the changes that have been requested for the maintenance release v1.1 version of the OSS Trouble Ticket API, JSR91.

The changes have been collected through:

- Bug parade: Bug and Request For Evolution (RFE) submitted by Java developers
- OSS/J specification leaders: evolution necessary to incorporate new common objects and to improve the common interfaces and Reference Implementation that will be “inherited” by all maintenance releases of the existing OSS APIs
- OSS/J Architectural Board: The common API needs to reflect the necessary new architectural recommendation (new CBE, etc)

1.2 Audience

This document is used to start a Maintenance Release of the OSS Trouble Ticket API JSR 91.

According to the JCPSM:

The Maintenance Lead (ML) will arrange to have all change items placed into the PROPOSED section of the Change Log (this document) and then send a request to the PMO to initiate a Maintenance Review. The PMO will make a public announcement and begin the review.

1.3 Approval and Distribution

The ML may choose to modify one or more of the proposed changes based on comments received during review.

1.4 Related Information

oss_trouble_ticket-1_0-fr-spec.zip: contains the Version 1.0 of the OSS Trouble Ticket API, JSR 91, http://java.sun.com/products/oss/start_download.html

1.5 Revision History

Date	Version	Author	State	Comments
October 2006	1.1.4	Roland Volk, IP VALUE	Final Release	<ul style="list-style-type: none"> Differentiate between proposed and accepted changes as result of Maintenance Draft Review Adapt OSS/J Common API v 1.4
August 2006	1.1.3	Ramin Roham-Pour, IP Value	Third Release	<ul style="list-style-type: none"> Move proposed to accepted changes
February 2006	1.1.2	Axel Terfloth, IP Value	Second Draft	<ul style="list-style-type: none"> Adapt OSS/J Common API v 1.3 Extend JVTTroubleTicketSession
November 2005	1.1	Axel Terfloth, IP Value	Initial Draft	<ul style="list-style-type: none"> Remove X.790 based trouble ticket definitions Use of trouble ticket definitions from CBE Changes to JVTTroubleTicketSession

2 Summary of changes

- Replace X.790 trouble ticket data model by CBE trouble ticket data model
- Adopt OSS Common API / CBE v1.4
- Introduce new named query pattern
- Introduce update procedures technically
- Using newly defined exceptions
- Changes to javax.oss.trouble.JVTTroubleTicketSession
- Remove X.790-specifics from event interfaces
- Named Queries and Responses
- Changes to Event Definitions

3 Proposed Changes

3.1.1 Using newly defined response types

The API adopts the new response classes defined in OSS Common API v1.4.

- `javax.oss.cbe.trouble.TroubleTicketKeyResult`
replaces `javax.oss.trouble.TroubleTicketKeyResult`
- `javax.oss.cbe.trouble.TroubleTicketKeyResultIterator`
or replaces
`javax.oss.trouble.TroubleTicketKeyResultIterator`
- `javax.oss.cbe.trouble.TroubleTicketValueIterator`
replaces
`javax.oss.trouble.TroubleTicketValueIterator`

See list of changes to `JVTTroubleTicketSession` for more details.

3.1.2 New methods

3.1.2.1 getTroubleTicketItemTypes

The definition of the method is:

```
/**
 * Get the TroubleTicketItemValue types supported by JVT Session
 *
 * @return String array which contains the fully qualified names of the leaf
 * node interfaces representing the supported TroubleTicketItemValue types.
 * @exception java.rmi.RemoteException
 */
String[] getTroubleTicketItemTypes() throws java.rmi.RemoteException;
```

3.1.2.2 makeTroubleTicketItemValue

The definition of the method is:

```
/**
 * Create a Value Type object for a specific trouble ticket item type.
 * <p>
 * Not to be confused with the creation of a trouble ticket item.
 * <p>
 * The Session Bean is used as a factory for the creation of value objects.
 *
 * @param valueType The fully qualified type of the TroubleTicketItemValue to be created.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
```

```

*/
TroubleTicketValue makeTroubleTicketItemValue( String valueType )
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;

```

3.1.2.3 getTroubleTicketItemByKey

The definition of the method is:

```

/**
 * Get a single trouble ticket item given its key.
 *
 * @param key          trouble ticket item Item Key
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketItemValue
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.ObjectNotFoundException No trouble ticket item found with
 * provided key
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketItemValue
    getTroubleTicketItemByKey(TroubleTicketItemKey key, String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.ObjectNotFoundException ,
               java.rmi.RemoteException;

```

3.1.2.4 getTroubleTicketItemsByKeys

The definition of the method is:

```

/**
 * Get multiple trouble ticket items given their keys
 *
 *
 *
 * @param keys trouble ticket item keys
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketItemValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.FinderException No trouble ticket item matching any of the keys
 * could be found
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketItemValue[]
    getTroubleTicketItemsByKeys(TroubleTicketItemKey[] keys, String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.FinderException,
               java.rmi.RemoteException;

```

3.1.2.5 getTroubleTicketItemsByTemplate

The definition of the method is:

```

/**
 * Get multiple trouble ticket items matching a single
 * trouble ticket item value template.
 *
 * @param template trouble ticket item value populated with attributes to be matched.

```

```

* @param attrNames Names of the attributes to be populated in returned
TroubleTicketItemValue(s)
* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception javax.ejb.FinderException No trouble ticket item matching the template
could be found.
* @exception java.rmi.RemoteException System Level Exception
*/
TroubleTicketItemValueIterator
    getTroubleTicketItemsByTemplate( TroubleTicketItemValue template, String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
            javax.ejb.FinderException,
            java.rmi.RemoteException;

```

3.1.2.6 getTroubleTicketItemsByTemplates

The definition of the method is:

```

/**
 * Get trouble tickets items matching at least one
 * of multiple trouble ticket item value templates.
 *
 * @param templates trouble ticket item values populated with attributes to be matched.
 * @param attrNames Names of the attributes to be populated in returned
TroubleTicketItemValue(s)
* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception javax.ejb.FinderException No trouble ticket item matching any of the
templates could be found.
* @exception java.rmi.RemoteException System Level Exception
*/
TroubleTicketItemValueIterator
    getTroubleTicketItemsByTemplates( TroubleTicketItemValue[] templates,
                                       String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
            javax.ejb.FinderException,
            java.rmi.RemoteException;

```

3.1.2.7 createTroubleTicketItemByValue

The definition of the method is:

```

/**
 * Create a single trouble ticket item.
 *
 * @param value trouble ticket item to be created
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.CreateException Application level exception while trying to
create the trouble ticket item.
 * @exception java.rmi.RemoteException System Level Exception
*/
TroubleTicketItemKey
    createTroubleTicketItemByValue(TroubleTicketItemValue value)
        throws javax.oss.OssIllegalArgumentException,
            javax.ejb.CreateException,
            javax.ejb.DuplicateKeyException,
            java.rmi.RemoteException;

```

3.1.2.8 createTroubleTicketItemsByValues

The definition of the method is:

```
/**
 * Atomic Creation of multiple trouble ticket items each with a different values
 *
 * @param values trouble ticket items to be created
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.CreateException Application level exception while trying to
 * create the trouble ticket items.
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketItemKey[]
    createTroubleTicketItemsByValues(TroubleTicketItemValue[] values)
        throws javax.oss.OssUnsupportedOperationException,
            javax.oss.OssIllegalArgumentException,
            javax.ejb.CreateException,
            javax.ejb.DuplicateKeyException,
            java.rmi.RemoteException;
```

3.1.2.9 tryCreateTroubleTicketItemsByValues

The definition of the method is:

```
/**
 * Best effort creation of multiple trouble ticket items
 *
 * @param values trouble ticket items to be created
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketItemKeyResult[]
    tryCreateTroubleTicketItemsByValues(TroubleTicketItemValue[] values)
        throws javax.oss.OssIllegalArgumentException,
            java.rmi.RemoteException;
```

3.1.2.10 setTroubleTicketItemByValue

The definition of the method is:

```
/**
 * Update a single trouble ticket item given a TroubleTicketItemValue instance.
 *
 * @param value trouble ticket item value with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while trying to set
 * the attributes.
 * @exception javax.ejb.ObjectNotFoundException No trouble ticket item matching provided
 * key could be found.
 * @exception java.rmi.RemoteException System Level Exception
 * @exception javax.oss.OssResyncRequiredException If resyncRequired is set and targeted
 * ticket has a different lastUpdateVersionNumber.
 */
void setTroubleTicketItemByValue(TroubleTicketItemValue value,
                                boolean resyncRequired )
    throws javax.oss.OssIllegalArgumentException,
```

```
javax.oss.OssSetException,
javax.ejb.ObjectNotFoundException,
java.rmi.RemoteException,
javax.oss.OssResyncRequiredException;
```

3.1.2.11 setTroubleTicketItemsByValues

The definition of the method is:

```
/**
 * Atomic Update multiple trouble ticket items each with a different values
 *
 * @param values trouble ticket item values with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception
while trying to update the trouble ticket items.
 * @exception javax.ejb.DuplicateKeyException The same Key is present in more
than one of the provided trouble ticket items.
 * @exception javax.ejb.FinderException At least one of the trouble ticket
items could not be found
 * @exception java.rmi.RemoteException System Level Exception
 * @exception javax.oss.OssResyncRequiredException resyncRequired is set to true
and at least one of the trouble ticket item value is
 * not synchronized with the back-end
trouble ticket item.
 */

void setTroubleTicketItemsByValues( TroubleTicketItemValue[] values,
                                   boolean resyncRequired )
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.OssSetException,
           javax.ejb.DuplicateKeyException ,
           javax.ejb.FinderException ,
           java.rmi.RemoteException,
           javax.oss.OssResyncRequiredException;
```

3.1.2.12 trySetTroubleTicketItemsByValues

The definition of the method is:

```
/**
 * Best Effort Update of multiple trouble ticket items given their values.
 *
 * @param values trouble ticket item values with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */

TroubleTicketItemKeyResult[]
    trySetTroubleTicketItemsByValues( TroubleTicketItemValue[] values,
                                     boolean resyncRequired )
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

3.1.2.13 setTroubleTicketItemsByKeys

The definition of the method is:

```
/**
 * Atomic Update multiple trouble ticket items to the same value.
 *
 * @param value keys Targeted trouble ticket item Keys
 * @param value trouble ticket item value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception
while trying to update the trouble ticket items.
 * @exception javax.ejb.FinderException At least one of the trouble ticket
items could not be found
 * @exception java.rmi.RemoteException System Level Exception
 */
void setTroubleTicketItemsByKeys( TroubleTicketItemKey[] keys,
                                TroubleTicketItemValue value )
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.OssSetException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;
```

3.1.2.14 trySetTroubleTicketItemsByKeys

The definition of the method is:

```
/**
 * Best Effort Update multiple trouble ticket items given their Keys
 * to the same value.
 *
 * @param value keys Targeted trouble ticket item Keys
 * @param value trouble ticket item value with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketItemKeyResult[]
    trySetTroubleTicketItemsByKeys( TroubleTicketItemKey[] keys,
                                    TroubleTicketItemValue value )
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;
```

3.1.2.15 setTroubleTicketItemsByTemplate

The definition of the method is:

```
/**
 * Atomic Update multiple trouble ticket items matching a template to the same value.
 *
 * @param template trouble ticket item value populated with attributes to be matched.
 * @param value trouble ticket item value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception
```

```
while trying to update the trouble ticket items.
 * @exception javax.ejb.FinderException          At least one of the trouble ticket
items could not be found
 * @exception java.rmi.RemoteException          System Level Exception
 */

void setTroubleTicketItemsByTemplate( TroubleTicketItemValue template,
                                     TroubleTicketItemValue value )
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.OssSetException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;
```

3.1.2.16 trySetTroubleTicketItemsByTemplate

The definition of the method is:

```
/**
 * Best Effort Update of multiple trouble ticket items matching a single
 * trouble ticket item value template.
 *
 * @param template trouble ticket item value populated with attributes to be matched.
 * @param value trouble ticket item value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException          System Level Exception
 */
TroubleTicketItemKeyResultIterator
    trySetTroubleTicketItemsByTemplate( TroubleTicketItemValue template,
                                       TroubleTicketItemValue value,
                                       boolean failuresOnly)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

3.1.2.17 setTroubleTicketItemsByTemplates

The definition of the method is:

```
/**
 * Atomic Update of multiple trouble ticket items matching at least one of the templates
 * to the same value
 *
 * @param templates trouble ticket item values populated with attributes to be matched.
 * @param value trouble ticket item value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while trying to
update the trouble ticket items.
 * @exception javax.ejb.DuplicateKeyException The same Key is present in more than one
template.
 * @exception javax.ejb.FinderException At least one of the trouble ticket items could
not be found
 * @exception java.rmi.RemoteException System Level Exception
 */

void setTroubleTicketItemsByTemplates( TroubleTicketItemValue[] templates,
                                       TroubleTicketItemValue value )
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
```

```
javax.oss.OssSetException,
javax.ejb.FinderException,
javax.ejb.DuplicateKeyException,
java.rmi.RemoteException;
```

3.1.2.18 trySetTroubleTicketItemsByTemplates

The definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets matching at least one of the templates
 * to the same value
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */

TroubleTicketKeyResultIterator
    trySetTroubleTicketsByTemplates(TroubleTicketValue[] templates,
                                    TroubleTicketValue value,
                                    boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;
```

3.1.2.19 removeTroubleTicketItemByKey

The definition of the method is:

```
/**
 * Removing an trouble ticket item by key.
 *
 * @param key
 * @throws javax.oss.OssUnsupportedOperationException
 * @throws javax.oss.OssIllegalArgumentException
 * @throws javax.ejb.ObjectNotFoundException
 * @throws javax.ejb.RemoveException
 * @throws java.rmi.RemoteException
 */
void removeTroubleTicketItemByKey(TroubleTicketItemKey key)
    throws
        javax.oss.OssUnsupportedOperationException,
        javax.oss.OssIllegalArgumentException,
        javax.ejb.ObjectNotFoundException,
        javax.ejb.RemoveException,
        java.rmi.RemoteException;
```

3.1.2.20 removeTroubleTicketItemsByKeys

The definition of the method is:

```
/**
 * Atomic remove operation of multiple trouble ticket items by the given keys.
 *
```



```

* @param keys
* @throws javax.oss.OssUnsupportedOperationException
* @throws javax.oss.OssIllegalArgumentException
* @throws javax.ejb.FinderException
* @throws javax.ejb.RemoveException
* @throws java.rmi.RemoteException
*/
void removeTroubleTicketItemsByKeys(TroubleTicketItemKey[] keys)
    throws
        javax.oss.OssUnsupportedOperationException,
        javax.oss.OssIllegalArgumentException,
        javax.ejb.FinderException,
        javax.ejb.RemoveException,
        java.rmi.RemoteException;

```

3.1.2.21 tryRemoveTroubleTicketItemsByKeys

The definition of the method is:

```

/**
 * Best effort remove operation of multiple trouble ticket items by the given keys.
 *
 * @param keys
 * @return
 * @throws javax.oss.OssUnsupportedOperationException
 * @throws javax.oss.OssIllegalArgumentException
 * @throws java.rmi.RemoteException
 */
TroubleTicketItemKeyResult[]
    tryRemoveTroubleTicketItemsByKeys( TroubleTicketItemKey[] keys )
        throws javax.oss.OssUnsupportedOperationException,
                javax.oss.OssIllegalArgumentException,
                java.rmi.RemoteException;

```

3.2 Named Queries and Responses

3.2.1 Changes to QueryAllOpenTroubleTicketsValue

Due to the adoption of the named query pattern the query value `javax.oss.trouble.QueryAllOpenTroubleTicketsValue` now inherits from `javax.oss.NamedQueryValue`.

The definition of the interface is:

```

/**
 * A NamedQueryValue used to get all the open trouble tickets.
 *
 * @author Pierre Gauthier Nortel Networks
 * @author Axel Terfloeth, IP Value
 * @version 1.1
 * @since November 2001
 * @see javax.oss.JVTSession#query(javax.oss.NamedQueryValue)
 */

```

```
public interface QueryAllOpenTroubleTicketsValue extends javax.oss.NamedQueryValue
{
}

```

The response type for this named query value is
`javax.oss.trouble.QueryTroubleTicketsResponse`

3.2.2 Added QueryAllTroubleTicketItemsRelatedToTroubleTicket

A `NamedQueryValue` used to get all the trouble ticket items which are related to a trouble ticket which is specified by a key. The response type for this named query value is
`javax.oss.trouble.QueryTroubleTicketItemsResponse`

Implementors may implement this query.

The definition of the interface is:

```
/**
 * A NamedQueryValue used to get all the trouble ticket items which are related to a
 trouble ticket
 * specified by a key.<br>
 * The response type of this named query is {@link QueryTroubleTicketItemsResponse}.<br>
 * <br>
 * Implementors may implement this query.<br>
 * <br>
 *
 * @author Axel Terfloeth, IP Value
 * @version 1.0
 * @since 1.1
 * @see javax.oss.JVTSession#query(javax.oss.NamedQueryValue)
 * @ssj:queryvalue
 */
public interface QueryAllTroubleTicketItemsRelatedToTroubleTicketValue
    extends javax.oss.NamedQueryValue
{
    public static final String RELATED_TROUBLE_TICKET_KEY = "relatedTroubleTicketKey";

    /**
     * Gets the key of the trouble ticket to which the trouble ticket items are related
     to.
     * @return the trouble ticket key
     */
    TroubleTicketKey getRelatedTroubleTicketKey();

    /**
     * Sets the key of the trouble ticket to which the trouble ticket items are related
     to.
     * @param key the trouble ticket key
     */
    void setRelatedTroubleTicketKey(TroubleTicketKey key);
}

```

3.2.3 Added QueryTroubleTicketItemsRelatedToTroubleTicket

A NamedQueryValue used to get all the trouble ticket items which are related to a trouble ticket which is specified by a key and matches at least one of the TroubleTicketItemValue templates specified.

The template matching rules, as defined for the template based management operations defined in the JVT session façade, must be applied. If no template is specified all possible trouble ticket items have to be included in the response.

The response type for this named query value is
`javax.oss.trouble.QueryTroubleTicketItemsResponse`

Implementors may implement this query.

The definition of the interface is:

```
/**
 * A NamedQueryValue used to get all the trouble ticket items which are related to a
 * trouble ticket
 * specified by a key and matches at least one of the TroubleTicketItemValue templates.
 * The template matching rules as defined for the template based management operations
 * defined in the JVT
 * session facade must be applied. If no template is specified all possible trouble ticket
 * items have to
 * be included in the response.
 * The response type of this named query is {@link QueryTroubleTicketItemsResponse}.<br>
 * <br>
 * Implementors may implement this query.<br>
 * <br>
 *
 * @author Axel Terfloeth, IP Value
 * @version 1.0
 * @since 1.1
 * @see javax.oss.JVTSession#query(javax.oss.NamedQueryValue)
 * @see QueryTroubleTicketItemsResponse
 * @ossj:queryvalue
 */
public interface QueryTroubleTicketItemsRelatedToTroubleTicketValue
    extends QueryAllTroubleTicketItemsRelatedToTroubleTicketValue
{
    public static final String TROUBLE_TICKET_ITEM_TEMPLATES =
        "troubleTicketItemTemplates";

    /**
     * Gets the array of templates which are used to match the trouble ticket item values.
     * @return array of template values
     */
    TroubleTicketItemValue[] getTroubleTicketItemTemplates();

    /**
     * Sets the array of templates which are used to match the trouble ticket item values.
     * @param templates array of template values
     */
    void setTroubleTicketItemTemplates(TroubleTicketItemValue[] templates);
}
```

3.2.4 Added QueryTroubleTicketsResponse

A named query response type for named queries, which return trouble ticket values. This response is a

`javax.oss.cbe.trouble.TroubleTicketValueIterator`.

The response interface is defined as:

```
/**
 * A NamedQueryResponse of named queries which return trouble ticket values.
 * This response is a {@link TroubleTicketValueIterator}.
 *
 * @author Axel Terfloeth, IP Value
 * @version 1.0
 * @since 1.1
 * @see javax.oss.JVTSession#query(javax.oss.NamedQueryValue)
 * @see QueryAllOpenTroubleTicketsValue
 */
public interface QueryTroubleTicketsResponse extends NamedQueryResponse,
                                                    TroubleTicketValueIterator
{
}
```

3.2.5 Added QueryTroubleTicketItemsResponse

A named query response type for named queries, which return trouble ticket item values. This response is a

`javax.oss.cbe.trouble.TroubleTicketItemValueIterator`.

The response interface is defined as:

```
/**
 * A NamedQueryResponse of named queries which return trouble ticket item values.
 * This response is a {@link TroubleTicketItemValueIterator}.
 *
 * @author Axel Terfloeth, IP Value
 * @version 1.0
 * @since 1.1
 * @see javax.oss.JVTSession#query(javax.oss.NamedQueryValue)
 */
public interface QueryTroubleTicketItemsResponse extends NamedQueryResponse,
                                                         TroubleTicketItemValueIterator
{
}
```

3.3 Changes to Event Definitions

3.3.1 Removed TroubleTicketStatusChangeEvent

Due to the changed trouble ticket state model this event definition is not applicable anymore. It will be replaced by the new event type `javax.oss.trouble.TroubleTicketStateChangeEvent`.

3.3.2 Changed TroubleTicketCloseOutEvent

The interface `javax.oss.trouble.TroubleTicketCloseOutEvent` now inherits from `javax.oss.trouble.TroubleTicketStateChangeEvent`. It defines no additional attributes.

The event interface is defined as:

```
public interface TroubleTicketCloseOutEvent extends TroubleTicketStateChangeEvent {  
}
```

3.3.3 Changed TroubleTicketCancellationEvent

The interface `javax.oss.trouble.TroubleTicketCancellationEvent` now inherits from `javax.oss.trouble.TroubleTicketStateChangeEvent`. It defines no additional attributes.

The attribute 'closeOutNarr' is removed from the `javax.oss.trouble.TroubleTicketCancellationEvent` since it is X.790 specific.

The event interface is defined as:

```
public interface TroubleTicketCancellationEvent extends TroubleTicketStateChangeEvent {  
}
```

3.3.4 Modified TroubleTicketAttributeValueChangeEvent

Due to adoption of the OSS/J Design Guidelines v1.2 the event definition was changed. The attribute 'troubleTicketValue' is replaced by

‘newTroubleTicketValue’. Additionally an optional attribute ‘oldTroubleTicketValue’ is added.

The event interface is defined as:

```
public interface TroubleTicketAttributeValueChangeEvent extends javax.oss.Event
{
    public static final String NEW_TROUBLE_TICKET_VALUE = "newTroubleTicketValue";
    public static final String OLD_TROUBLE_TICKET_VALUE = "oldTroubleTicketValue";

    /**
     * Returns the new Trouble Ticket Value with only the changed attributes being
     * populated.
     *
     * @return TroubleTicketValue - Trouble Ticket Value
     * @exception java.lang.IllegalStateException -Is thrown if the attribute has not been
     * populated.
     */
    public TroubleTicketValue getNewTroubleTicketValue()
        throws java.lang.IllegalStateException;

    /**
     * Sets the new Trouble Ticket Value with only the changed attributes being
     * populated.
     *
     * @param newValue Trouble Ticket Value
     * @exception java.lang.IllegalArgumentException - Is thrown to report that a bad
     * argument was provided to the method.
     */
    public void setNewTroubleTicketValue( TroubleTicketValue newValue )
        throws java.lang.IllegalArgumentException;

    /**
     * Returns the old Trouble Ticket Value with only the changed attributes being
     * populated. This attribute is optional.
     *
     * @return TroubleTicketValue - Trouble Ticket Value
     * @exception java.lang.IllegalStateException -Is thrown if the attribute has not been
     * populated.
     */
    public TroubleTicketValue getOldTroubleTicketValue()
        throws javax.oss.OssUnsupportedAttributeException,
            java.lang.IllegalStateException;

    /**
     * Sets the old Trouble Ticket Value with only the changed attributes being
     * populated. This attribute is optional.
     *
     * @param newValue Trouble Ticket Value
     * @exception java.lang.IllegalArgumentException - Is thrown to report that a bad
     * argument was provided to the method.
     */
    public void setOldTroubleTicketValue( TroubleTicketValue newValue )
        throws javax.oss.OssUnsupportedAttributeException,
            java.lang.IllegalArgumentException;
}
```

3.3.5 Added TroubleTicketStateChangeEvent

A `javax.oss.trouble.TroubleTicketStateChangeEvent` is emitted when a trouble ticket has changed its state. The attributes present in the event are:

- the trouble ticket key
- the current state

The attributes which may be present in the event are:

- the previous state.

The event interface is defined as:

```
public interface TroubleTicketStateChangeEvent extends javax.oss.Event
{
    public static final String TROUBLE_TICKET_KEY      = "troubleTicketKey";
    public static final String NEW_STATE               = "newState";
    public static final String OLD_STATE               = "oldState";

    /**
     * Returns the Trouble Ticket Key.
     *
     * @return TroubleTicketKey - Trouble Ticket Key
     * @exception java.lang.IllegalStateException -Is thrown if the attribute has not
    been populated.
     */
    TroubleTicketKey getTroubleTicketKey()
        throws java.lang.IllegalStateException;

    /**
     * Sets the Trouble Ticket Key.
     *
     * @param key Trouble Ticket Key
     * @exception java.lang.IllegalArgumentException - Is thrown to report that a bad
    argument was provided to the method.
     */
    void setTroubleTicketKey(TroubleTicketKey key)
        throws java.lang.IllegalArgumentException;

    /**
     * Returns the new TroubleTicketState. For example
    <CODE>TroubleTicketState.OPENACTIVE</CODE>
     *
     * @return String - Trouble Ticket State as given by one of the
    <CODE>TroubleTicketState</CODE> interface constant
     * @exception java.lang.IllegalStateException -Is thrown if the attribute has not
    been populated.
     */
    String getNewState()
        throws java.lang.IllegalStateException;

    /**
     * Sets the new trouble ticket state. For example
    <CODE>TroubleTicketStat.OPENACTIVE</CODE>
     *
     * @param state as given by one of the <CODE>TroubleTicketStatus</CODE> interface
    constant
     * @exception java.lang.IllegalArgumentException - Is thrown to report that a bad
    argument was provided to the method.
     */
    void setNewState(String state)
        throws java.lang.IllegalArgumentException;

    /**
     * Returns the previous TroubleTicketState. For example
    <CODE>TroubleTicketState.OPENACTIVE</CODE>

```

```

    *
    * @return String - Trouble Ticket State as given by one of the
    <CODE>TroubleTicketState</CODE> interface constant
    * @exception java.lang.IllegalStateException -Is thrown if the attribute has not
    been populated.
    */
    String getOldState()
        throws javax.oss.OssUnsupportedAttributeException,
            java.lang.IllegalStateException;

    /**
    * Sets the previous trouble ticket state. For example
    <CODE>TroubleTicketStat.OPENACTIVE</CODE>
    *
    * @param state as given by one of the <CODE>TroubleTicketStatus</CODE> interface
    constant
    * @exception java.lang.IllegalArgumentException - Is thrown to report that a bad
    argument was provided to the method.
    */
    void setOldState(String state)
        throws javax.oss.OssUnsupportedAttributeException,
            java.lang.IllegalArgumentException;
}

```

3.3.6 Added TroubleTicketItemCreateEvent

Events of this type will be sent, when a new trouble ticket item has been created. It is up to the vendor to determine, what attributes are contained in the trouble ticket item value. The minimal set of populated attributes is the TroubleTicketItemKey which identifies the trouble ticket item.

The event interface is defined as:

```

public interface TroubleTicketItemCreateEvent extends javax.oss.Event
{
    public static final String TROUBLE_TICKET_ITEM_VALUE = "troubleTicketItemValue";
    /**
    * Returns the created trouble ticket item value
    *
    * @return TroubleTicketValue - created trouble ticket value
    * @exception IllegalStateException -Is thrown if the attribute has not been
    populated.
    */
    public TroubleTicketItemValue getTroubleTicketItemValue()
        throws IllegalStateException;

    /**
    * Sets the new Trouble Ticket Value
    *
    * @param value trouble ticket item value
    * @exception IllegalArgumentException - Is thrown to report that a bad argument was
    provided to the method.
    */
    public void setTroubleTicketItemValue(TroubleTicketItemValue value)
        throws IllegalArgumentException;
}

```


3.3.7 Added TroubleTicketItemAttributeValueChangeEvent

Event that is sent when the attribute values of a trouble ticket item has changed. The same event may contain multiple attribute changes. The 'newTroubleTicketItemValue' attribute contains the troubleTicketItemValue which contains the changed attribute values. Only the changed attribute must be populated in the TroubleTicketItemValue. The 'oldTroubleTicketItemValue' attribute holds the TroubleTicketItemValue which contains the old attribute values. This attribute is optional and may be supported by an implementer.

The event interface is defined as:

```
public interface TroubleTicketItemAttributeValueChangeEvent extends javax.oss.Event
{
    public static final String NEW_TROUBLE_TICKET_ITEM_VALUE =
"newTroubleTicketItemValue";
    public static final String OLD_TROUBLE_TICKET_ITEM_VALUE =
"oldTroubleTicketItemValue";

    /**
     * Returns the new Trouble Ticket Item Value with only the changed attributes being
     * populated.
     *
     * @return Trouble Ticket Item Value containing the new values of the changed
     * attributes
     * @exception IllegalStateException -Is thrown if the attribute has not been
     * populated.
     */
    public TroubleTicketItemValue getNewTroubleTicketItemValue()
        throws IllegalStateException;

    /**
     * Sets the new Trouble Ticket Item Value with only the changed attributes being
     * populated.
     *
     * @param newValue Trouble Ticket Value
     * @exception IllegalArgumentException - Is thrown to report that a bad argument was
     * provided to the method.
     */
    public void setNewTroubleTicketItemValue( TroubleTicketItemValue newValue )
        throws IllegalArgumentException;

    /**
     * Returns the old Trouble Ticket Item Value with only the changed attributes being
     * populated. This attribute is optional.
     *
     * @return Trouble Ticket Item Value containing the old values of the changed
     * attributes
     * @exception IllegalStateException -Is thrown if the attribute has not been
     * populated.
     */
    public TroubleTicketItemValue getOldTroubleTicketItemValue()
        throws javax.oss.OssUnsupportedAttributeException,
        IllegalStateException;

    /**
     * Sets the old Trouble Ticket Item Value with only the changed attributes being
     * populated. This attribute is optional.
     *
     * @param newValue Trouble Ticket Item Value
     * @exception IllegalArgumentException - Is thrown to report that a bad argument was
     * provided to the method.
     */
}
```

```

public void setOldTroubleTicketItemValue( TroubleTicketItemValue newValue )
    throws javax.oss.OssUnsupportedAttributeException,
        IllegalArgumentException;
}

```

3.3.8 Added TroubleTicketItemRemoveEvent

Events of this type will be sent, when a new trouble ticket item has been removed. It is up to the vendor to determine, what attributes are contained in the trouble ticket item value. The minimal set of populated attributes is the TroubleTicketItemKey which identifies the trouble ticket item.

The event interface is defined as:

```

public interface TroubleTicketItemRemoveEvent extends javax.oss.Event
{
    public static final String TROUBLE_TICKET_ITEM_VALUE = "troubleTicketItemValue";

    /**
     * Returns the removed trouble ticket item value
     *
     * @return TroubleTicketValue - created trouble ticket value
     * @exception IllegalStateException -Is thrown if the attribute has not been
     * populated.
     */
    public TroubleTicketItemValue getTroubleTicketItemValue()
        throws IllegalStateException;

    /**
     * Sets the removed Trouble Ticket Value
     *
     * @param value trouble ticket item value
     * @exception IllegalArgumentException - Is thrown to report that a bad argument was
     * provided to the method.
     */
    public void setTroubleTicketItemValue(TroubleTicketItemValue value)
        throws IllegalArgumentException;
}

```

4 Accepted Changes

4.1 Replace X.790 trouble ticket data model by CBE trouble ticket data model

The trouble ticket data model as provided by the OSS Trouble Ticket API v1.0 is based on the ITU-T X.790 standard. This standard defines a specific trouble ticket data model which is very detailed.

For the adoption of the OSS Trouble Ticket API the X.790 based data model is too specific. In order to improve the adoptability of the OSS Trouble Ticket API the X.790 based data model is removed from the specification and is replaced by a simpler and more generic data model from the Common Business Entities (CBE) defined in OSS Common API v1.4.

The X.790 based data model is provided as open source as an extension to the OSS Trouble Ticket API v1.1.

4.1.1 Remove X.790 based interfaces and related data types

The following java interfaces of are removed from the java package `javax.oss.trouble`.

- `AccessHours`
- `ActivityCode`
- `ActivityDuration`
- `ActivityDurationList`
- `ActivityType`
- `Address`
- `AlarmValueList`
- `Authorization`
- `CloseOutVerification`
- `CustomerRoleAssignment`
- `DayOfWeek`
- `Escalation`
- `EscalationException`
- `EscalationList`
- `InitiatingMode`
- `NorthAmericaAddress`
- `OrgLevel`
- `PerceivedTroubleSeverity`
- `PersonReach`
- `PreferredPriority`
- `QueryByEscalationResponse`
- `QueryByEscalationValue`
- `RepairActivity`
- `RepairActivityList`

- RepeatReport
- RequestState
- Role
- SPRoleAssignment
- SuspectObject
- Time
- TimeInterval
- TimeLength
- TroubleFound
- TroubleLocationInfo
- TroubleStatus
- TroubleTicketStatus
- TroubleTicketValue
- TroubleTicketStatusChangeEvent
- TroubleTicketStatusChangeEventPropertyDescriptor
- TroubleType

4.2 Adopt OSS Common API / CBE v1.4

OSS Trouble Ticket API v1.1 is based on OSS Common API v1.4 and follows the new Draft of the OSS/J Design Guideline v1.2. This leads to some changes in the OSS Trouble Ticket API v1.1.

4.2.1 Introduce new named query pattern

The patterns for performing and defining queries have been updated in OSS Common API v1.4. The OSS Trouble Ticket API changes accordingly.

- The query value interface
`javax.oss.trouble.QueryAllOpenTroubleTicketsValue`
now extends the new interface `javax.oss.NamedQueryValue`.
- The `queryTroubleTickets()` method of the `JVTTroubleTicketSession` is removed. The new `query()` method as defined in `javax.oss.JVTSession` will be used instead.

4.2.2 Introduce update procedures technically

By extending the `javax.oss.JVTSession` interface the `javax.oss.trouble.JVTTroubleTicketSession` interface inherits the definition of the update procedure related methods.

Even there are no concrete update procedures defined in the OSS Trouble Ticket API the specification mandates implementers to support these methods.

4.2.3 Using newly defined exceptions

The API adopts the new exception classes defined in OSS Common API v1.4.

- `javax.oss.OssUnsupportedOperationException` replaces `javax.oss.UnsupportedOperationException`
- `javax.oss.OssIllegalArgumentException` replaces `javax.oss.IllegalArgumentException`
- `javax.oss.OssSetException` replaces `javax.oss.SetException`
- `javax.oss.OssResyncRequiredException` replaces `javax.oss.ResyncRequiredException`

See list of changes to `JVTTroubleTicketSession` for more details.

4.3 Changes to `javax.oss.trouble.JVTTroubleTicketSession`

The changes to the interface `javax.oss.trouble.JVTTroubleTicketSession` are driven by the adoption of the OSS Common API v1.4 on the one hand and the replacement of the X.790 based trouble ticket model on the other hand.

The changes include:

- Removed X.790 specific method definitions
- Replace the interface `javax.oss.trouble.TroubleTicketValue` by `javax.oss.cbe.trouble.TroubleTicketValue` in parameter and return types
- Replace the interface `javax.oss.trouble.TroubleTicketKey` by `javax.oss.cbe.trouble.TroubleTicketKey` in parameter and return types
- Replace the interface `javax.oss.trouble.TroubleTicketKeyResult` by `javax.oss.cbe.trouble.TroubleTicketKeyResult` in parameter and return types
- Replace the interface `javax.oss.trouble.TroubleTicketKeyResultIterator` by `javax.oss.cbe.trouble.TroubleTicketKeyResultIterator` in parameter and return types

- Replace the interface `javax.oss.trouble.TroubleTicketValueIterator` by `javax.oss.cbe.trouble.TroubleTicketValueIterator` in parameter and return types
- Removed or replaced X.790 specific parameters in operations.
- Remove old query style operations.
- Added methods to manage instances of type `javax.oss.cbe.trouble.TroubleTicketItemValue`

4.3.1 Removed Methods

All X.790 specific methods and the `queryTroubleTickets()` method are removed from the `JVTTroubleTicketSession`. The following methods are removed:

- `escalateTroubleTicketByKey`
- `escalateTroubleTicketsByKeys`
- `escalateTroubleTicketsByTemplate`
- `escalateTroubleTicketsByTemplates`
- `tryEscalateTroubleTicketsByKeys`
- `tryEscalateTroubleTicketsByTemplate`
- `tryEscalateTroubleTicketsByTemplates`
- `queryTroubleTickets`

4.3.2 Changed Methods

4.3.2.1 Changes to `makeTroubleTicketValue`

The following changes apply:

- replace return type `javax.oss.trouble.TroubleTicketValue` with `javax.oss.cbe.trouble.TroubleTicketValue`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Create a Value Type object for a specific Trouble Ticket type.
 * Not to be confused with the creation of a Trouble Ticket.
 * The Session Bean is used as a factory for the creation of Value Types.
 *
 * @param valueType The fully qualified type of the TroubleTicketValue to be created.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
```

```
javax.oss.cbe.trouble.TroubleTicketValue makeTroubleTicketValue( String valueType)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

4.3.2.2 Changes to getTroubleTicketByKey

The following changes apply:

- replace return type `javax.oss.trouble.TroubleTicketValue` with `javax.oss.cbe.trouble.TroubleTicketValue`
- replace first parameter type `javax.oss.trouble.TroubleTicketKey` with `javax.oss.cbe.trouble.TroubleTicketKey`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Get a Single Trouble Ticket given its Key.
 *
 * @param key          Trouble Ticket Key
 * @param attrNames Names of the attributes to be populated in returned TroubleTicketValue
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.ObjectNotFoundException No Trouble Ticket found with provided Key
 * @exception java.rmi.RemoteException          System Level Exception
 */
javax.oss.cbe.trouble.TroubleTicketValue
    getTroubleTicketByKey( javax.oss.cbe.trouble.TroubleTicketKey key,
                          String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.ObjectNotFoundException,
               java.rmi.RemoteException;
```

4.3.2.3 Changes to getTroubleTicketsByKeys

The following changes apply:

- replace return type `javax.oss.trouble.TroubleTicketValue[]` with `javax.oss.cbe.trouble.TroubleTicketValue[]`
- replace first parameter type `javax.oss.trouble.TroubleTicketKey[]` with `javax.oss.cbe.trouble.TroubleTicketKey[]`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Get Multiple Trouble Tickets given their Keys
 *
 *
 * @param keys Trouble Ticket Keys
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.FinderException No Trouble Ticket matching any of the
 * keys could be found
 * @exception java.rmi.RemoteException System Level Exception
 */
javax.oss.cbe.trouble.TroubleTicketValue[]
    getTroubleTicketsByKeys(javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                           String[] attrNames)
    throws javax.oss.OssIllegalArgumentException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;
```

4.3.2.4 Changes to createTroubleTicketByValue

The following changes apply:

- replace return type `javax.oss.trouble.TroubleTicketKey` with `javax.oss.cbe.trouble.TroubleTicketKey`
- replace first parameter type `javax.oss.trouble.TroubleTicketValue` with `javax.oss.cbe.trouble.TroubleTicketValue`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Create a Single Trouble Ticket.
 *
 *
 * @param value Trouble Ticket to be created
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.CreateException Application level exception while trying
 * to create the Trouble Ticket.
 * @exception java.rmi.RemoteException System Level Exception
 */
javax.oss.cbe.trouble.TroubleTicketKey
    createTroubleTicketByValue(javax.oss.cbe.trouble.TroubleTicketValue value)
    throws javax.oss.OssIllegalArgumentException,
           javax.ejb.CreateException,
           javax.ejb DuplicateKeyException,
           java.rmi.RemoteException;
```


4.3.2.5 Changes to createTroubleTicketsByValues

The following changes apply:

- replace return type `javax.oss.trouble.TroubleTicketKey[]` with `javax.oss.cbe.trouble.TroubleTicketKey[]`
- replace first parameter type `javax.oss.trouble.TroubleTicketValue[]` with `javax.oss.cbe.trouble.TroubleTicketValue[]`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`
- replace Exception `javax.oss.UnsupportedOperationException` with `javax.oss.OssUnsupportedOperationException`

The new definition of the method is:

```
/**
 * Atomic Creation of Multiple Trouble Tickets each with a different values
 *
 * @param values Trouble Tickets to be created
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.CreateException Application level exception while
trying to create the Trouble Tickets.
 * @exception java.rmi.RemoteException System Level Exception
 */
javax.oss.cbe.trouble.TroubleTicketKey[]
    createTroubleTicketsByValues(javax.oss.cbe.trouble.TroubleTicketValue[] values)
        throws javax.oss.OssUnsupportedOperationException,
            javax.oss.OssIllegalArgumentException,
            javax.ejb.CreateException,
            javax.ejb.DuplicateKeyException,
            java.rmi.RemoteException;
```

4.3.2.6 Changes to setTroubleTicketByValue

The following changes apply:

- replace first parameter type `javax.oss.trouble.TroubleTicketValue` with `javax.oss.cbe.trouble.TroubleTicketValue`
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`
- replace Exception `javax.oss.SetException` with `javax.oss.OssSetException`

- replace Exception `javax.oss.ResyncRequiredException` with `javax.oss.OssResyncRequiredException`

The new definition of the method is:

```
/**
 * Update a Single Trouble Ticket given a Trouble Ticket Value
 *
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while
 * trying to set the attributes.
 * @exception javax.ejb.ObjectNotFoundException No Trouble Ticket matching provided key
 * could be found.
 * @exception java.rmi.RemoteException System Level Exception
 * @exception javax.oss.OssResyncRequiredException If resyncRequired is set and targeted
 * ticket has a different lastUpdateVersionNumber.
 */
void setTroubleTicketByValue( javax.oss.cbe.trouble.TroubleTicketValue value,
                             boolean resyncRequired )
    throws javax.oss.OssIllegalArgumentException,
           javax.oss.OssSetException,
           javax.ejb.ObjectNotFoundException,
           java.rmi.RemoteException,
           javax.oss.OssResyncRequiredException;
```

4.3.2.7 Changes to setTroubleTicketsByValues

The following changes apply:

- replace first parameter type
`javax.oss.trouble.TroubleTicketValue[]` with
`javax.oss.cbe.trouble.TroubleTicketValue[]`
- replace Exception
`javax.oss.UnsupportedOperationException` with
`javax.oss.OssUnsupportedOperationException`
- replace Exception `javax.oss.IllegalArgumentException` with
`javax.oss.OssIllegalArgumentException`
- replace Exception `javax.oss.SetException` with
`javax.oss.OssSetException`
- replace Exception `javax.oss.ResyncRequiredException` with
`javax.oss.OssResyncRequiredException`

The new definition of the method is:

```
/**
 * Atomic Update Multiple Trouble Tickets each with a different values
 *
 * @param values Trouble Ticket Values with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
```

```

* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception javax.oss.OssSetException Application level exception while trying to update
the Trouble Tickets.
* @exception javax.ejb.DuplicateKeyException The same Key is present in more than one of
the provided Trouble Tickets.
* @exception javax.ejb.FinderException At least one of the Trouble Tickets could not be
found
* @exception java.rmi.RemoteException System Level Exception
* @exception javax.oss.OssResyncRequiredException resyncRequired is set to true and at
least one of the Trouble Ticket Value is
*
not synchronized with the back-end
Trouble Ticket.
*/
void setTroubleTicketsByValues( javax.oss.cbe.trouble.TroubleTicketValue[] values,
                               boolean resyncRequired)
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.OssSetException,
           javax.ejb.DuplicateKeyException ,
           javax.ejb.FinderException ,
           java.rmi.RemoteException,
           javax.oss.OssResyncRequiredException;

```

4.3.2.8 Changes to setTroubleTicketsByKey

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception
javax.oss.UnsupportedOperationException with
javax.oss.OssUnsupportedOperationException
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException
- replace Exception javax.oss.SetException with
javax.oss.OssSetException

The new definition of the method is:

```

/**
 * Atomic Update Multiple Trouble Tickets to the same value.
 *
 * @param value keys Targeted Trouble Ticket Keys
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not upported
by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while trying to
update the Trouble Tickets.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets could not be

```

```
found
 * @exception java.rmi.RemoteException System Level Exception
 */
void setTroubleTicketsByKeys( javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                             javax.oss.cbe.trouble.TroubleTicketValue value)
                             throws javax.oss.OssUnsupportedOperationException,
                                   javax.oss.OssIllegalArgumentException,
                                   javax.oss.OssSetException,
                                   javax.ejb.FinderException,
                                   java.rmi.RemoteException;
```

4.3.2.9 Changes to setTroubleTicketsByTemplate

The following changes apply:

- replace first and second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception
javax.oss.UnsupportedOperationException with
javax.oss.OssUnsupportedOperationException
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException
- replace Exception javax.oss.SetException with
javax.oss.OssSetException

The new definition of the method is:

```
/**
 * Atomic Update Multiple Trouble Tickets matching a template to the same value.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while trying to update
the Trouble Tickets.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets could not be
found
 * @exception java.rmi.RemoteException System Level Exception
 */
void setTroubleTicketsByTemplate(javax.oss.cbe.trouble.TroubleTicketValue template,
                                javax.oss.cbe.trouble.TroubleTicketValue value)
                                throws javax.oss.OssUnsupportedOperationException,
                                      javax.oss.OssIllegalArgumentException,
                                      javax.oss.OssSetException,
                                      javax.ejb.FinderException,
                                      java.rmi.RemoteException;
```

4.3.2.10 Changes to setTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception
javax.oss.UnsupportedOperationException with
javax.oss.OssUnsupportedOperationException
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException
- replace Exception javax.oss.SetException with
javax.oss.OssSetException

The new definition of the method is:

```
/**
 * Atomic Update of Multiple Trouble Tickets matching at least one of the templates
 * to the same value
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.OssSetException Application level exception while trying to update
 * the Trouble Tickets.
 * @exception javax.ejb.DuplicateKeyException The same Key is present in more than one
 * template.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets could not be
 * found
 * @exception java.rmi.RemoteException System Level Exception
 */
void setTroubleTicketsByTemplates( javax.oss.cbe.trouble.TroubleTicketValue[] templates,
                                javax.oss.cbe.trouble.TroubleTicketValue value)
                                throws javax.oss.OssUnsupportedOperationException,
                                javax.oss.OssIllegalArgumentException,
                                javax.oss.OssSetException,
                                javax.ejb.FinderException,
                                javax.ejb.DuplicateKeyException,
                                java.rmi.RemoteException;
```

4.3.2.11 Changes to closeTroubleTicketByKey

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey with
javax.oss.cbe.trouble.TroubleTicketKey

- change type of second parameter 'troubleTicketState' from int to java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790 specific attribute
- replace Exception javax.oss.IllegalArgumentException with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Close a Trouble Ticket given a Key
 *
 * @param key Targeted Trouble Ticket Key
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CloseException Application level exception while trying
to close the Trouble Ticket.
 * @exception javax.ejb.ObjectNotFoundException No Trouble Ticket matching provided key
could be found.
 * @exception java.rmi.RemoteException System Level Exception
 *
 * @see javax.oss.cbe.trouble.TroubleTicketState
 *
 * @ssj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */
void closeTroubleTicketByKey( javax.oss.cbe.trouble.TroubleTicketKey key,
                             String troubleTicketState )
                             throws javax.oss.OssIllegalArgumentException,
                             javax.oss.trouble.CloseException,
                             javax.ejb.ObjectNotFoundException,
                             java.rmi.RemoteException;
```

4.3.2.12 Changes to closeTroubleTicketsByKeys

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790 specific attribute
- replace Exception javax.oss.IllegalArgumentException with javax.oss.OssIllegalArgumentException

- replace Exception
 `javax.oss.UnsupportedOperationException` with
 `javax.oss.OssUnsupportedOperationException`

The new definition of the method is:

```
/**
 * Atomic Close of Multiple Trouble Tickets given their keys.
 *
 * @param keys keys Targeted Trouble Ticket Keys
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not supported
 * by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CloseException Application level exception while trying to
 * close the Trouble Tickets.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets could not be
 * found
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
void closeTroubleTicketsByKeys(javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                               String troubleTicketState)
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.trouble.CloseException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;
```

4.3.2.13 Changes to closeTroubleTicketsByTemplate

The following changes apply:

- replace first parameter type
 `javax.oss.trouble.TroubleTicketValue` with
 `javax.oss.cbe.trouble.TroubleTicketValue`
- change type of second parameter 'troubleTicketState' from `int` to
 `java.lang.String` since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
 specific attribute
- replace Exception `javax.oss.IllegalArgumentException`
 with `javax.oss.OssIllegalArgumentException`
- replace Exception
 `javax.oss.UnsupportedOperationException` with
 `javax.oss.OssUnsupportedOperationException`

The new definition of the method is:

```

/**
 * Atomic Close of Multiple Trouble Tickets matching a template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CloseException Application level exception while
 * trying to close the Trouble Tickets.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets
 * could not be found
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
void closeTroubleTicketsByTemplate( javax.oss.cbe.trouble.TroubleTicketValue template,
                                   String troubleTicketState)
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.trouble.CloseException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;

```

4.3.2.14 Changes to closeTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException
- replace Exception
javax.oss.UnsupportedOperationException with
javax.oss.OssUnsupportedOperationException

The new definition of the method is:

```

/**
 * Atomic Close of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CloseException Application level exception while

```



```
trying to close the Trouble Tickets.
 * @exception javax.ejb.DuplicateKeyException          The same Key is present in more
than one template.
 * @exception javax.ejb.FinderException                At least one of the Trouble Tickets
could not be found
 * @exception java.rmi.RemoteException                System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */

void closeTroubleTicketsByTemplates(javax.oss.cbe.trouble.TroubleTicketValue[] templates,
                                   String troubleTicketState)
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.trouble.CloseException,
           javax.ejb.FinderException,
           javax.ejb.DuplicateKeyException,
           java.rmi.RemoteException;
```

4.3.2.15 Changes to cancelTroubleTicketByKey

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey with
javax.oss.cbe.trouble.TroubleTicketKey
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Cancel a Trouble Ticket given a Key
 *
 * @param key Targeted Trouble Ticket Key
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CancelException Application level exception while trying
to cancel the Trouble Ticket.
 * @exception javax.ejb.ObjectNotFoundException No Trouble Ticket matching provided key
could be found.
 * @exception java.rmi.RemoteException          System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */
void cancelTroubleTicketByKey(javax.oss.cbe.trouble.TroubleTicketKey key,
                              String troubleTicketState)
    throws javax.oss.OssIllegalArgumentException,
           javax.ejb.ObjectNotFoundException;
```

```
javax.oss.trouble.CancelException,
java.rmi.RemoteException;
```

4.3.2.16 Changes to cancelTroubleTicketsByKeys

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException
- replace Exception
javax.oss.UnsupportedOperationException with
javax.oss.OssUnsupportedOperationException

The new definition of the method is:

```
/**
 * Atomic Cancel of Multiple Trouble Tickets given their keys.
 *
 * @param keys keys Targeted Trouble Ticket Keys
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
 * supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CancelException Application level exception while
 * trying to cancel the Trouble Tickets.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets
 * could not be found
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
void cancelTroubleTicketsByKeys(javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                                String troubleTicketState)
    throws javax.oss.OssUnsupportedOperationException,
           javax.oss.OssIllegalArgumentException,
           javax.oss.trouble.CancelException,
           javax.ejb.FinderException,
           java.rmi.RemoteException;
```

4.3.2.17 Changes to cancelTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
`javax.oss.trouble.TroubleTicketValue[]` with
`javax.oss.cbe.trouble.TroubleTicketValue[]`
- change type of second parameter 'troubleTicketState' from `int` to
`java.lang.String` since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception `javax.oss.IllegalArgumentException`
with `javax.oss.OssIllegalArgumentException`
- replace Exception
`javax.oss.UnsupportedOperationException` with
`javax.oss.OssUnsupportedOperationException`

The new definition of the method is:

```
/**
 * Atomic Cancel of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @exception javax.oss.OssUnsupportedOperationException If the operation is not
supported by the implementation.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.oss.trouble.CancelException Application level exception while
trying to cancel the Trouble Tickets.
 * @exception javax.ejb.DuplicateKeyException The same Key is present in more
than one template.
 * @exception javax.ejb.FinderException At least one of the Trouble Tickets
could not be found
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */
void cancelTroubleTicketsByTemplates(javax.oss.cbe.trouble.TroubleTicketValue[] templates,
String troubleTicketState)
throws javax.oss.OssUnsupportedOperationException,
javax.oss.OssIllegalArgumentException,
javax.oss.trouble.CancelException,
javax.ejb.FinderException,
javax.ejb.DuplicateKeyException,
java.rmi.RemoteException;
```

4.3.2.18 Changes to getTroubleTicketsByTemplate

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketValueIterator with
javax.oss.cbe.trouble.TroubleTicketValueIterator
- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException with
javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Get Multiple Trouble Tickets matching a single
 * TroubleTicketValue template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.FinderException No Trouble Ticket matching the template could be
 * found.
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketValueIterator
    getTroubleTicketsByTemplate(javax.oss.cbe.trouble.TroubleTicketValue template,
                               String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.FinderException,
               java.rmi.RemoteException;
```

4.3.2.19 Changes to getTroubleTicketsByTemplates

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketValueIterator with
javax.oss.cbe.trouble.TroubleTicketValueIterator
- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace Exception javax.oss.IllegalArgumentException with
javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Get Trouble Tickets matching at least one
 * of multiple Trouble Ticket Value templates.
 *
 *
```

```

* @param templates Trouble Ticket Values populated with attributes to be matched.
* @param attrNames Names of the attributes to be populated in returned
TroubleTicketValue(s)
* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception javax.ejb.FinderException No Trouble Ticket matching any of the templates
could be found.
* @exception java.rmi.RemoteException          System Level Exception
*/
TroubleTicketValueIterator
    getTroubleTicketsByTemplates(javax.oss.cbe.trouble.TroubleTicketValue[] templates,
                                String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.FinderException,
               java.rmi.RemoteException;

```

4.3.2.20 Changes to tryCreateTroubleTicketsByValues

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResult[] with
javax.oss.cbe.trouble.TroubleTicketKeyResult[]
- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```

/**
 * Best Effort Creation of Multiple Trouble Tickets
 *
 * @param values Trouble Tickets to be created
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException          System Level Exception
 */
TroubleTicketKeyResult[]
    tryCreateTroubleTicketsByValues(javax.oss.cbe.trouble.TroubleTicketValue[] values)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;

```

4.3.2.21 Changes to trySetTroubleTicketsByValues

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResult[] with
javax.oss.cbe.trouble.TroubleTicketKeyResult[]

- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets given their values.
 *
 * @param values Trouble Ticket Values with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResult[]
    trySetTroubleTicketsByValues( javax.oss.cbe.trouble.TroubleTicketValue[] values,
                                boolean resychRequired)
                                throws javax.oss.OssIllegalArgumentException,
                                java.rmi.RemoteException;
```

4.3.2.22 Changes to trySetTroubleTicketsByKey

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResult[] with
javax.oss.cbe.trouble.TroubleTicketKeyResult[]
- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue

The new definition of the method is:

```
/**
 * Best Effort Update Multiple Trouble Tickets given their Keys
 * to the same value.
 *
 * @param value keys Targeted Trouble Ticket Keys
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResult[]
    trySetTroubleTicketsByKey( javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                              javax.oss.cbe.trouble.TroubleTicketValue value)
                              throws javax.oss.OssIllegalArgumentException,
                              java.rmi.RemoteException;
```

4.3.2.23 Changes to trySetTroubleTicketsByTemplate

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResultIterator
with
javax.oss.cbe.trouble.TroubleTicketKeyResultIterator
- replace first and second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets matching a single
 * Trouble Ticket value template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResultIterator
    trySetTroubleTicketsByTemplate( javax.oss.cbe.trouble.TroubleTicketValue template,
                                   javax.oss.cbe.trouble.TroubleTicketValue value,
                                   boolean failuresOnly)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

4.3.2.24 Changes to trySetTroubleTicketsByTemplates

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResultIterator
with
javax.oss.cbe.trouble.TroubleTicketKeyResultIterator
- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]

- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets matching at least one of the templates
 * to the same value
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResultIterator
    trySetTroubleTicketsByTemplates(
        javax.oss.cbe.trouble.TroubleTicketValue[] templates,
        javax.oss.cbe.trouble.TroubleTicketValue value,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;
```

4.3.2.25 Changes to tryCloseTroubleTicketsByKeys

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResult[] with
javax.oss.cbe.trouble.TroubleTicketKeyResult[]
- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Close of Multiple Trouble Tickets given their keys.
```



```

*
* @param keys Targeted Trouble Ticket Keys
* @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception java.rmi.RemoteException System Level Exception
* @ssj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
*/
TroubleTicketKeyResult[]
    tryCloseTroubleTicketsByKeys( javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                                String troubleTicketState)
                                throws javax.oss.OssIllegalArgumentException,
                                java.rmi.RemoteException;

```

4.3.2.26 Changes to tryCloseTroubleTicketsByTemplate

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResultIterator
with
javax.oss.cbe.trouble.TroubleTicketKeyResultIterat
or
- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```

/**
* Best Effort Close of Multiple Trouble Tickets matching a single
* Trouble Ticket value template.
*
* @param template Trouble Ticket Value populated with attributes to be matched.
* @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
* @param failuresOnly Returns only the failed updates in the KeyResults.
* @exception javax.oss.OssIllegalArgumentException Invalid Arguments
* @exception java.rmi.RemoteException System Level Exception
* @ssj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
*/
TroubleTicketKeyResultIterator
    tryCloseTroubleTicketsByTemplate(
                                javax.oss.cbe.trouble.TroubleTicketValue template,
                                String troubleTicketState,

```

```

        boolean failuresOnly)
throws  javax.oss.OssIllegalArgumentException,
        java.rmi.RemoteException;

```

4.3.2.27 Changes to tryCloseTroubleTicketsByTemplates

The following changes apply:

- replace return type
`javax.oss.trouble.TroubleTicketKeyResultIterartor`
with
`javax.oss.cbe.trouble.TroubleTicketKeyResultIterat`
or
- replace first parameter type
`javax.oss.trouble.TroubleTicketValue` with
`javax.oss.cbe.trouble.TroubleTicketValue`
- change type of second parameter 'troubleTicketState' from `int` to
`java.lang.String` since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception `javax.oss.IllegalArgumentException`
with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```

/**
 * Best Effort Close of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResultIterator
    tryCloseTroubleTicketsByTemplates(
        javax.oss.cbe.trouble.TroubleTicketValue[] templates,
        String troubleTicketState,
        boolean failuresOnly)
throws  javax.oss.OssIllegalArgumentException,
        java.rmi.RemoteException;

```

4.3.2.28 Changes to tryCancelTroubleTicketsByKeys

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResult[] with
javax.oss.cbe.trouble.TroubleTicketKeyResult[]
- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Cancel of Multiple Trouble Tickets given their keys.
 *
 * @param keys Targeted Trouble Ticket Keys
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResult[]
    tryCancelTroubleTicketsByKeys(
        javax.oss.cbe.trouble.TroubleTicketKey[] keys,
        String troubleTicketState)
    throws javax.oss.OssIllegalArgumentException,
        java.rmi.RemoteException;
```

4.3.2.29 Changes to tryCancelTroubleTicketsByTemplate

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResultIterator
with
javax.oss.cbe.trouble.TroubleTicketKeyResultIterat
or
- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue

- change type of second parameter 'troubleTicketState' from int to java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790 specific attribute
- replace Exception javax.oss.IllegalArgumentException with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Cancel of Multiple Trouble Tickets matching a single
 * Trouble Ticket value template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ssj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */

TroubleTicketKeyResultIterator
    tryCancelTroubleTicketsByTemplate(
        TroubleTicketValue template,
        String troubleTicketState,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;
```

4.3.2.30 Changes to tryCancelTroubleTicketsByTemplates

The following changes apply:

- replace return type
javax.oss.trouble.TroubleTicketKeyResultIterator
with
javax.oss.cbe.trouble.TroubleTicketKeyResultIterator
- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- change type of second parameter 'troubleTicketState' from int to java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790 specific attribute

- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Best Effort Cancel of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResultIterator
    tryCancelTroubleTicketsByTemplates(
        TroubleTicketValue[] templates,
        String troubleTicketState,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
            java.rmi.RemoteException;
```

5 Rejected Changes

5.1 Changes to javax.oss.trouble.JVTTroubleTicketSession

5.1.1 Changed Methods

5.1.1.1 Changes to queryTroubleTickets

The following changes apply:

- the method is deprecated now since it uses the deprecated javax.oss.QueryValue interface. The query() method defined the javax.oss.JVTSession interface, which uses the new NamedQueryValue concept, should be used instead.
- replace Exception javax.oss.IllegalArgumentException with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Query Trouble Tickets using a Query Value.
 *
 * @param queryValue A QueryValue created via <CODE>makeTroubleTicketValue</CODE>
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 *
 * @deprecated Use {@link javax.oss.JVTSession#query} instead for {@link NamedQueryValue}
 * query types.
 */
TroubleTicketValueIterator
    queryTroubleTickets( QueryValue queryValue , String[] attrNames)
                        throws javax.oss.OssIllegalArgumentException,
                               java.rmi.RemoteException;
```

5.1.1.2 Changes to getTroubleTicketsByTemplate

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException with
javax.oss.OssIllegalArgumentException

The new definition of the method is:

```

/**
 * Get Multiple Trouble Tickets matching a single
 * TroubleTicketValue template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.FinderException No Trouble Ticket matching the template could be
 * found.
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketValueIterator
    getTroubleTicketsByTemplate(javax.oss.cbe.trouble.TroubleTicketValue template,
                               String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.FinderException,
               java.rmi.RemoteException;

```

5.1.1.3 Changes to getTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
`javax.oss.trouble.TroubleTicketValue[]` with
`javax.oss.cbe.trouble.TroubleTicketValue[]`
- replace Exception `javax.oss.IllegalArgumentException`
with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```

/**
 * Get Trouble Tickets matching at least one
 * of multiple Trouble Ticket Value templates.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param attrNames Names of the attributes to be populated in returned
 * TroubleTicketValue(s)
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception javax.ejb.FinderException No Trouble Ticket matching any of the templates
 * could be found.
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketValueIterator
    getTroubleTicketsByTemplates(javax.oss.cbe.trouble.TroubleTicketValue[] templates,
                                 String[] attrNames)
        throws javax.oss.OssIllegalArgumentException,
               javax.ejb.FinderException,
               java.rmi.RemoteException;

```

5.1.1.4 Changes to tryCreateTroubleTicketsByValues

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Creation of Multiple Trouble Tickets
 *
 * @param values Trouble Tickets to be created
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResult[]
    tryCreateTroubleTicketsByValues(javax.oss.cbe.trouble.TroubleTicketValue[] values)
        throws javax.oss.OssIllegalArgumentException,
            java.rmi.RemoteException;
```

5.1.1.5 Changes to trySetTroubleTicketsByValues

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets given their values.
 *
 * @param values Trouble Ticket Values with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResult[]
    trySetTroubleTicketsByValues(javax.oss.cbe.trouble.TroubleTicketValue[] values,
        boolean resynchRequired)
        throws javax.oss.OssIllegalArgumentException,
            java.rmi.RemoteException;
```

5.1.1.6 Changes to trySetTroubleTicketsByKeys

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue

The new definition of the method is:

```
/**
 * Best Effort Update Multiple Trouble Tickets given their Keys
 * to the same value.
 *
 * @param value keys Targeted Trouble Ticket Keys
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResult[]
    trySetTroubleTicketsByKeys( javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                               javax.oss.cbe.trouble.TroubleTicketValue value)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

5.1.1.7 Changes to trySetTroubleTicketsByTemplate

The following changes apply:

- replace first and second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets matching a single
 * Trouble Ticket value template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResultIterator
    trySetTroubleTicketsByTemplate( javax.oss.cbe.trouble.TroubleTicketValue template,
                                   javax.oss.cbe.trouble.TroubleTicketValue value,
                                   boolean failuresOnly)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

5.1.1.8 Changes to trySetTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue[] with
javax.oss.cbe.trouble.TroubleTicketValue[]
- replace second parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- replace Exception javax.oss.IllegalArgumentException with
javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Update of Multiple Trouble Tickets matching at least one of the templates
 * to the same value
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param value Trouble Ticket Value with populated attributes to be set.
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
TroubleTicketKeyResultIterator
    trySetTroubleTicketsByTemplates(
        javax.oss.cbe.trouble.TroubleTicketValue[] templates,
        javax.oss.cbe.trouble.TroubleTicketValue value,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;
```

5.1.1.9 Changes to tryCloseTroubleTicketsByKeys

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException with
javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Close of Multiple Trouble Tickets given their keys.
 *
 * @param keys Targeted Trouble Ticket Keys
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResult[]
    tryCloseTroubleTicketsByKeys( javax.oss.cbe.trouble.TroubleTicketKey[] keys,
                                String troubleTicketState)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

5.1.1.10 Changes to tryCloseTroubleTicketsByTemplate

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Close of Multiple Trouble Tickets matching a single
 * Trouble Ticket value template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResultIterator
    tryCloseTroubleTicketsByTemplate(
        javax.oss.cbe.trouble.TroubleTicketValue template,
        String troubleTicketState,
        boolean failuresOnly)
    throws javax.oss.OssIllegalArgumentException,
           java.rmi.RemoteException;
```

5.1.1.11 Changes to tryCloseTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketValue with
javax.oss.cbe.trouble.TroubleTicketValue
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed
- remove third parameter 'closeOutNarr' since it mapped to a X.790
specific attribute
- replace Exception javax.oss.IllegalArgumentException
with javax.oss.OssIllegalArgumentException

The new definition of the method is:

```
/**
 * Best Effort Close of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
 * constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
 * class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResultIterator
    tryCloseTroubleTicketsByTemplates(
        javax.oss.cbe.trouble.TroubleTicketValue[] templates,
        String troubleTicketState,
        boolean failuresOnly)
    throws javax.oss.OssIllegalArgumentException,
        java.rmi.RemoteException;
```

5.1.1.12 Changes to tryCancelTroubleTicketsByKeys

The following changes apply:

- replace first parameter type
javax.oss.trouble.TroubleTicketKey[] with
javax.oss.cbe.trouble.TroubleTicketKey[]
- change type of second parameter 'troubleTicketState' from int to
java.lang.String since the trouble ticket state model changed

- remove third parameter ‘closeOutNarr’ since it mapped to a X.790 specific attribute
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Best Effort Cancel of Multiple Trouble Tickets given their keys.
 *
 * @param keys Targeted Trouble Ticket Keys
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
 */
TroubleTicketKeyResult[]
    tryCancelTroubleTicketsByKeys(
        javax.oss.cbe.trouble.TroubleTicketKey[] keys,
        String troubleTicketState)
    throws javax.oss.OssIllegalArgumentException,
        java.rmi.RemoteException;
```

5.1.1.13 Changes to tryCancelTroubleTicketsByTemplate

The following changes apply:

- replace first parameter type `javax.oss.trouble.TroubleTicketValue` with `javax.oss.cbe.trouble.TroubleTicketValue`
- change type of second parameter ‘troubleTicketState’ from `int` to `java.lang.String` since the trouble ticket state model changed
- remove third parameter ‘closeOutNarr’ since it mapped to a X.790 specific attribute
- replace Exception `javax.oss.IllegalArgumentException` with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```
/**
 * Best Effort Cancel of Multiple Trouble Tickets matching a single
 * Trouble Ticket value template.
 *
 * @param template Trouble Ticket Value populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 */
```

```

* @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
*/

TroubleTicketKeyResultIterator
    tryCancelTroubleTicketsByTemplate(
        TroubleTicketValue template,
        String troubleTicketState,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;

```

5.1.1.14 Changes to tryCancelTroubleTicketsByTemplates

The following changes apply:

- replace first parameter type
`javax.oss.trouble.TroubleTicketValue[]` with
`javax.oss.cbe.trouble.TroubleTicketValue[]`
- change type of second parameter ‘troubleTicketState’ from `int` to
`java.lang.String` since the trouble ticket state model changed
- remove third parameter ‘closeOutNarr’ since it mapped to a X.790
specific attribute
- replace Exception `javax.oss.IllegalArgumentException`
with `javax.oss.OssIllegalArgumentException`

The new definition of the method is:

```

/**
 * Best Effort Cancel of Multiple Trouble Tickets matching at least one of the template.
 *
 * @param templates Trouble Ticket Values populated with attributes to be matched.
 * @param troubleTicketState as given by one of the <CODE>TroubleStatus</CODE> interface
constant
 * @param failuresOnly Returns only the failed updates in the KeyResults.
 * @exception javax.oss.OssIllegalArgumentException Invalid Arguments
 * @exception java.rmi.RemoteException System Level Exception
 * @ossj:enumerationfield name=troubleTicketState
class=javax.oss.cbe.trouble.TroubleTicketState
*/
TroubleTicketKeyResultIterator
    tryCancelTroubleTicketsByTemplates(
        TroubleTicketValue[] templates,
        String troubleTicketState,
        boolean failuresOnly)
        throws javax.oss.OssIllegalArgumentException,
               java.rmi.RemoteException;

```

6 Deferred Changes from OSS Trouble Ticket API v1.0