Resource Trouble Management (RTM) - DDP BA

TMF518_RTM

Version 1.2

September, 2011

Notice

No recipient of this document and code shall in any way interpret this material as representing a position or agreement of TM Forum or its members. This material is draft working material of TM Forum and is provided solely for comments and evaluation. It is not “Forum Approved” and is solely circulated for the purposes of assisting TM Forum in the preparation of final material in furtherance of the aims and mission of TM Forum.

Although it is copyrighted material of TM Forum:

- Members of TM Forum are only granted the limited copyright waiver to distribute this material within their companies and may not make paper or electronic copies for distribution outside of their companies.
- Non-members of the TM Forum are not permitted to make copies (paper or electronic) of this draft material other than for their internal use for the sole purpose of making comments thereon directly to TM Forum.
- If this material forms part of a supply of information in support of an Industry Group Liaison relationship, the document may only be used as part of the work identified in the Liaison and may not be used or further distributed for any other purposes

Any use of this material by the recipient, other than as set forth specifically herein, is at its own risk, and under no circumstances will TM Forum be liable for direct or indirect damages or any costs or losses resulting from the use of this material by the recipient.

This material is governed, and all recipients shall be bound, by all of the terms and conditions of the Intellectual Property Rights Policy of the TM Forum (http://www.tmforum.org/Bylaws/1094/home.html) and may involve a claim of patent rights by one or more TM Forum members or by non-members of TM Forum.

Direct inquiries to the TM Forum office:

240 Headquarters Plaza,
East Tower – 10th Floor,
Morristown, NJ 07960 USA
Tel No. +1 973 944 5100
Fax No. +1 973 944 5110
TM Forum Web Page: www.tmforum.org

Table of Contents

List of Requirements

List of Use Cases

List of Figures

List of Tables

Executive Summary

This document is the Business Agreements part of the Resource Trouble Management (RTM) Document Delivery Package (DDP). The business agreements cover requirements and use cases for the following aspects of RTM: resource fault management, protection management, and maintenance and diagnostics control.

See Section 1 for additional introductory material related to this document.

1 Introduction

1.1 DDP Structure

In order to allow for more efficient release delivery, the previous monolithic BA, IA and SS documents have been partitioned into smaller self-contained (though not independent) units called Document Delivery Packages (DDPs).

This is similar to the 3GPP concept of Integration Reference Point (IRP). The basic idea is that the Interface, which is specified by the entire document set (of a release), is partitioned into DDPs where each DDP specifies “a certain aspect” of the Interface, which needs to be very clearly scoped.

There are three kinds of DDPs:

- the FrameWork DDP (FMW) – this DDP contains the generic artifacts that are applicable to all the other DDPs.
- Data Model DDP (DM-DDP) – a DDP that concerns a data model (entities, data structures, attributes, state, but no operations)
- Operation Model DDP (OM-DDP) – a DDP that concerns a computational model (operations, notifications, transactions) for a given functional area (such as resource inventory management)

The unified deliverables structure for any given MTOSI / MTNM product release is as follows:

- Product Release Notes:
  - a scope specification for the type and extent of the delivered product,
  - the partitioning of the release into DDPs (i.e., definitions of various aspects of the release),
  - and an overview of the release’s (delta) deliverables;
- For each DDP:
  - Business Agreements (BAs): a business view specification
  - Information Agreements (IAs): a system view specification
  - Interface Implementation Specifications (ISSs): implementation and deployment view specification per supported enabling technology (mapping of the IA to either CORBA (IDL, services usage) or XML (WSDL, XSD, bindings…)
  - Supporting Documentation: normative and informative supporting documents.
- Reference Implementation (optional) of core IIS fragments for selected interfaces and enabling technologies.

1.2 Document Overview

This document covers requirements and use cases for **resource** fault management, protection management, and maintenance and diagnostics.

In terms of fault management, the focus is on alarm subscription, active alarm synchronization, the control of alarm reporting and the predetermined assignment of severities to given alarm types.

The protect management requirements and use cases entail the retrieval of protection related information such as protection groups and subscription to protect switch events.

The maintenance and diagnostics aspect focuses on the sending of maintenance commands and the determination of active (still running) maintenance actions.

1.3 Document Structure

The following sections are included in this document:

- Section 1 is this introduction.
- Section 2 defines the business problem and project scope
- Section 3 has the requirements and associated descriptive text.
- Section 4 contains the use cases.
- Section 5 has traceability matrices between the use cases and associated requirements, and vice versa.
- Section 6 provides a summary and list of open issues to be considered in later versions of this document.
- Section 7 lists references and states IPR claims, if any.
- Section 8 provides administrative details such as author contact information, document history and acknowledgements.

1.4 Terminology Used In This Document

Many of the object types used in this document are defined in the associated BA for the NRA DDP, i.e., TMF518_NRA.

For other terms refer to the SD0-1 supporting document.

2 Business Problem Description, Project Scope

2.1 Project Scope

The TM Forum Integration Program is responsible for all of the interface and business services work within the TM Forum. In some cases, interface work is delegated to other teams but the final verification for technical uniformity and integrity is the responsibility of the TM Forum Integration Program.

Initially, the TM Forum Integration Program was formed to coordinate the various existing TM Forum interfaces activities (as shown in **Figure 2-1**). In particular, the responsibility for maintaining MTOSI and MTNM is now covered by the MTOSI-MTNM Users Group which is a team within the TM Forum Integration Program. The long term plan (which is already well under progress) is to migration the various input work to a single harmonized suite of interfaces.

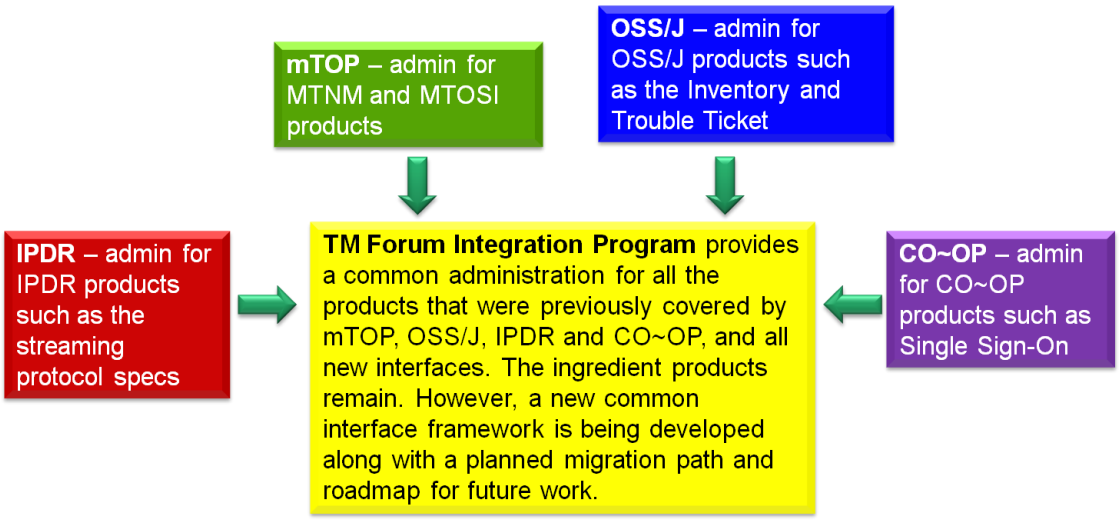

Figure 2-1. Inputs to the TM Forum Integration Program

Figure 2-2 provides a summary of the team within the TM Forum Integration Program as well as a few teams outside of the program but which also do some interface work. In terms of MTOSI and MTNM, the main input for updates come from the Resource and Service Management Team.

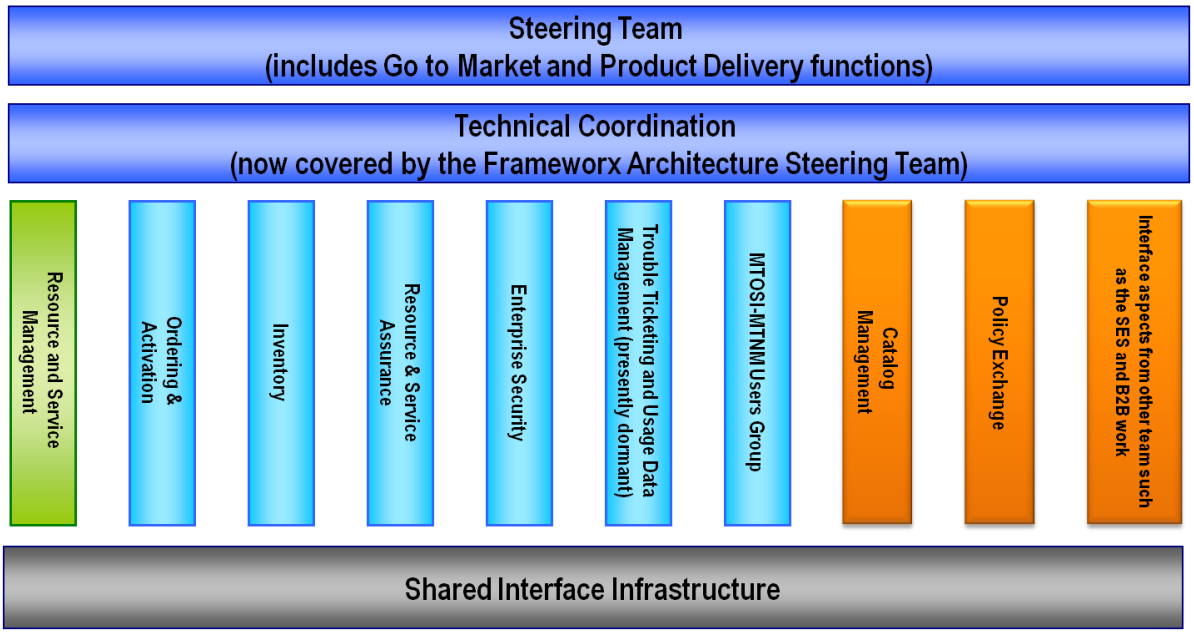

Figure 2-2. TM Forum Integration Program

2.2 Benefits

MTOSI and MTNM provide a set of Interface specifications that allow for resource and service management (with only MTOSI covering service management, but with MTOSI and MTNM both covering resource management, using very much the same information model).

These specifications are intended to lower design, implementation, Verification Validation & Testing (VVT), and maintenance costs for management interfaces. These Interfaces are intended for use by service providers, suppliers of equipment and OSS suppliers. The intention is to also encourage system integrator usage of management systems that make use of the Interfaces.

In particular, the followed approach tends to minimize the cost of integration, provide access to all necessary information and control, and support all vendor/operator differentiation. The intent of the interface is to provide compatibility among different version, for a detailed description see SD2-6_VersioningAndExtensibility.

2.2.1 Service Provider Benefits

The service provider benefits are as follows:

- One stop shopping concerning feature requests for much of the TM Forum contract specification work is part of the defined Change Control Group (CCG) process that TM Forum makes available in order to control the interface.
- The technical deliverables are also of high value to the service provider. The Interface specifications allow for an open, multi-supplier environment, shorten delivery times and lower integration costs.

- The MTOSI and MTNM products provide an integrated, multi-technology interface with support for most key layer 1 and layer 2 transport technologies. This is in contrast to earlier approaches where each technology-specific forum provided a single-technology management interface. The service provider was faced with having to use many different, uncoordinated management interfaces.
- These products are not bound to any one middleware, transport or computing language. So, the service provider will be able to evolve to new technologies as they arise.

2.2.2 Supplier Benefits

The supplier benefits are as follows:

- Fewer Adapters leads to Lower Costs – in as much as MTOSI and MTNM gain market penetration (and there has already been significant market acceptance of these interfaces), the supplier is faced with the need to build fewer adapters between their products and the products of their partners. A supplier can also directly see cost savings in the use of the Interfaces among its own products (as the need for an open interface arises).
- Lower Middleware Transitions Costs – the Interfaces are defined to be middleware and transport independent. So, the supplier can migrate from one middleware or transport technology to another without changing the supporting business logic in the code.
- Increase Usage by System Integrators (SIs) – a supplier's support of their own "open" interfaces goes only so far to encourage SIs. Clearly, an SI would like to make use of supplier products (both equipment and OSS suppliers) that make use of well supported standard interfaces rather than supplier specific interfaces. The latter case forces the SI into a situation characterized by many pair-wise negotiations between various suppliers.
- Lower Training Cost – in as much as a supplier re-uses the Interfaces for multiple products and for multiple customers, the various training costs are lower because the designers, system engineers, developers and testers are using the same Interfaces over and over again.

3 Business Processes

3.1 Business Requirements

R_TMF518_RTM_BR_0001	The Interface shall support subscription to alarm reports based on filtering conditions and the subsequent reporting of alarms to subscribed OSs.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_BR_0002	The Interface shall support the control of alarm reporting in terms of activating and deactivating alarm reporting for a given managed entity or a specified set of managed entities.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_BR_0003	The Interface shall support synchronization with respect to the list of active alarms known to the alarm generator and the alarm subscribers.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_BR_0004	The Interface shall support the retrieval of protection information such as protection groups and the reporting of protection events.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_BR_0005	The Interface shall support requests to perform protection switch commands.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_BR_0006	The interface shall support requests for maintenance and diagnostic tests.
Source	TMF518_RTM, Version 1.0

3.2 Category I: Static and Structural Requirements

The Category I requirements for assurance can be found in the TMF518_NRA (Network Resource Assurance) BA document.

3.3 Category II: Normal Sequences, Dynamic Requirements

3.3.1 Alarm Management

3.3.1.1 Alarm Severity Assignment Profile (ASAP) Retrieval

R_TMF518_RTM_II_0007	The Interface shall allow the requesting OS to retrieve the attributes of all the Alarm Severity Assignment Profiles (ASAPs) that are being managed by the target OS.
Source	TMF 513, Version 3.0, Requirement II.205

R_TMF518_RTM_II_0009	The Interface shall allow the requesting OS to retrieve the attributes of a given Alarm Severity Assignment Profile (ASAP).
Source	TMF 513, Version 3.0, Requirement II.207

R_TMF518_RTM_II_0010	The Interface shall allow the requesting OS to retrieve all the Alarm Severity Assignment Profiles (ASAPs) that are assigned to a given object. The requesting OS shall be able to specify the list of resource layer rates for which assigned ASAPs are to be retrieved. If an empty list is specified, then all ASAPs assigned to the addressed resource will be replied. The list shall also be empty if the addressed resource is not a Termination Point. Note that only Termination Point (TPs) can refer to more than one ASAP, with at most one ASAP per encapsulated layer rate.
Source	TMF 513, Version 3.0, Requirement II.208

3.3.1.2 Control of Alarm Reporting

R_TMF518_RTM_II_0012	The Interface shall allow the requesting OS to activate (allow, or turn on) alarm reporting for a particular Termination Point (TP). Alarm reporting for the TP is to be turned “on” at the specific layerRate provided by the requesting OS. However, setting of this parameter is best-effort. If the target OS does not support

	this granularity, it is acceptable for the target OS to turn on or off alarm reporting for all the layers of the TP regardless of the layerRate specified by the requesting OS. It is also acceptable for the target OS to turn on or off alarm reporting for the contained CTPs, if the ME does not support finer granularity.
Source	TMF 513, Version 3.0, Requirement II.108

R_TMF518_RTM_II_0013	The Interface shall allow the requesting OS to deactivate (inhibit, or turn off) alarm reporting for a particular Termination Point (TP). Alarm reporting is to be turned "off" at the layer represented by the Termination Point (TP). See also the exceptions to this rule noted in R_TMF518_RTM_II_0012.
Source	TMF 513, Version 3.0, Requirement II.109

In order to provide SNC alarm reports, the reporting OS has to correlate TP related information into "arc" related information. Note: How to provide this correlation is behavior of the reporting OS and is therefore outside the scope of the Interface. The activation / de-activation do not imply anything on the alarm reporting flag of any of the related TPs of the SNC / topological link. The requesting OS shall be able to retrieve the status of the activation / de-activation.

R_TMF518_RTM_II_0014	The Interface shall allow the requesting OS to activate (allow, or turn on) alarm reporting for a particular Equipment, Equipment Holder, Equipment Protection Group, Flow Domain, Flow Domain Fragment, Group Termination Point, Managed Element, Matrix Flow Domain, Multi-Layer Subnetwork, OS, Protection Group, Subnetwork Connection (SNC) and Topological Link.
Source	TMF 513, Version 3.0, Requirement II.078, 159, 161, 219 and for additional object types based on alignment with the IA.

R_TMF518_RTM_II_0015	The Interface shall allow the requesting OS to de-activate (inhibit, or turn off) alarm reporting for a particular Equipment, Equipment Holder, Equipment Protection Group, Flow Domain, Flow Domain Fragment, Group Termination Point, Managed Element, Matrix Flow Domain, Multi-Layer Subnetwork, OS, Protection Group, Subnetwork Connection (SNC) and Topological Link.
Source	TMF 513, Version 3.0, Requirement II.079, 160, 162, 220 and for additional object types based on alignment with the IA.

3.3.1.3 Alarm Severity Assignment Profile (ASAP) Control

R_TMF518_RTM_II_0022	The Interface shall allow the requesting to create an Alarm Severity Assignment Profile (ASAP) in the target OS. The following parameters are supplied by the requesting OS in conjunction with the ASAP creation request : • User label – see definition in TMF518_FRM. • User label uniqueness – see definition in TMF518_FRM. • Owner – see definition in TMF518_FRM. • Alarm severity assignments – This attribute shall represent the set of alarm severity assignments (refer to TMF518_NRA). • newASAP – This attribute returns the complete information of the created ASAP.
Source	TMF 513, Version 3.0, Requirement II.196 and 197

R_TMF518_RTM_II_0023	The Interface shall allow the requesting OS to modify an Alarm Severity Assignment Profile (ASAP) in the target OS. The target OS shall refuse/fail this request if the ASAP is fixed, i.e., it can neither be modified nor deleted by the requesting OS. The following parameters are supplied by the requesting OS in conjunction with the ASAP modification request : • ASAP name – this parameter shall represent the name of the ASAP that is to be modified. • User label – see definition in TMF518_FRM. • User label uniqueness – see definition in TMF518_FRM. • Owner – see definition in TMF518_FRM. • Alarm severity assignments – this attribute shall represent the new set of alarm severity assignments that are to be applied to the ASAP (refer to TMF518_NRA).
Source	TMF 513, Version 3.0, Requirement II.198 and 199

R_TMF518_RTM_II_0024	The Interface shall allow the requesting OS to delete a given Alarm Severity Assignment Profile (ASAP). The target OS shall refuse/fail this request if at least one object is pointing to this ASAP instance, or the ASAP cannot be deleted, i.e., neither can be modified nor deleted by the requesting OS.
Source	TMF 513, Version 3.0, Requirement II.200

R_TMF518_RTM_II_0025	The Interface shall allow the requesting OS to assign an Alarm Severity Assignment Profile (ASAP) to an instance of any of the following object classes: • Equipment • Equipment Holder • Equipment Protection Group (EPG) • Group Termination Point (GTP) • Managed Element (ME) • Management Domain (MD) • Operations System (OS) • Protection Group (PG) • Subnetwork Connection (SNC) • Termination Point (TP) • Topological Link (TL). The Interface shall allow the requesting OS to specify the following parameters when it requests that a target OS assign an Alarm Severity Assignment Profile (ASAP) to an object: • ASAP name – this parameter shall represent the name of the ASAP that is to be assigned. • Resource ref – this parameter shall represent the name of the object to which the ASAP is to be assigned. • Layer rate – this parameter shall represent the layer rate to which the ASAP is applicable. This shall be need when the addressed object is a Termination Point (TP).
Source	TMF 513, Version 3.0, Requirement II.201 and 202

R_TMF518_RTM_II_0026	The Interface shall allow the requesting OS to de-assign an Alarm Severity Assignment Profile (ASAP) from an instance of any of the object classes listed in R_TMF518_RTM_II_0025. The target OS shall refuse/fail this request if the ASAP is assigned in a fixed way to the object. The Interface shall allow the requesting OS to specify the following parameters when it requests that a target OS de-assign an Alarm Severity Assignment Profile (ASAP) from an object: • Resource Ref – this parameter shall represent the name of the object from which the ASAP is to be de-assigned. • Layer rate – this parameter shall represent the layer

	rate to which the ASAP is applicable. This shall be need when the addressed object is a Termination Point (TP)
Source	TMF 513, Version 3.0, Requirement II.203 and 204

3.3.1.4 Alarm Surveillance

3.3.1.4.1 Alarm Administration

R_TMF518_RTM_II_0027	The Interface shall allow an OS on the CCV to subscribe to non-OS system related alarms from one or more OSs on the CCV, (e.g. Managed Element (ME), Subnetwork, Subnetwork Connection (SNC), Equipment and Environmental alarms, etc.).
Source	TMF 517, Version 1.1, Requirement II.31

R_TMF518_RTM_II_0028	The Interface shall allow an OS on the CCV to subscribe to OS system related alarms from one or more OSs on the CCV.
Source	TMF 517, Version 1.1, Requirement II.32

R_TMF518_RTM_II_0029	The Interface shall allow an OS to specify zero or more alarm filters to be applied by the alarm generating OS as part of the OS's alarm subscription request/update. The filters are such that specific types of alarms are included based on parameter value matching. The allowable filter parameters are listed in Table **3-1**.
Source	TMF 513, Version 3.0, Requirement II.101, 102, 103, 104 and 105. TMF 517, Version 1.1, Requirement II.34 and 35

R_TMF518_RTM_II_0030	The Interface shall allow an OS to un-subscribe from a previous alarm subscription.
Source	TMF 513, Version 3.0, Requirement II.106 TMF 517, Version 1.1, Requirement II.33

R_TMF518_RTM_II_0031	The Interface shall allow an OS to send alarm(s) to registered OSs immediately after detecting an alarm condition.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_II_0032	The Interface shall allow an OS to modify the filtering criteria with regard to a successful previous subscription for alarms. In

	particular, this means an OS may ○ remove one or more of the filtering criteria that were specified in a previous successful alarm subscription request/update, and/or ○ add one or more filtering criteria to the set of previously applied criteria.
Source	TMF 513, Version 3.0, Requirement II.107

R_TMF518_RTM_II_0033	The Interface shall allow an event/alarm generating OS to inform the subscribing OSs about the status of event forwarding and whether lifecycle events and/or alarms have been discarded. This concerns event and/or alarm loss **within** the alarm generating OS itself **not** related to the notification service.
Source	TMF 513, Version 3.0, Requirement II.177 TMF 517, Version 1.1, Requirement II.36

R_TMF518_RTM_II_0034	With regard to R_TMF518_RTM_II_0033, the event/alarm generating OS shall inform the subscribing OSs when the event/alarm loss situation is over, i.e., when events/alarms are no longer being discarded by the generating OS.
Source	TMF 513, Version 3.0, Requirement II.178

R_TMF518_RTM_II_0035	The Interface shall support the distribution of heartbeat notifications. This allows an OS to inform other OSs on the CCV that it is connected to the notification service and able to send notifications.
Source	TMF518_RTM, Version 1.0

3.3.1.4.2 Alarm Summary

R_TMF518_RTM_II_0036	The Interface shall support requests to retrieve the number of active alarms known to the target OS.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_II_0037	The Interface shall support the capability for a requesting OS to retrieve the active alarms known to a target OS.
Source	TMF 517, Version 1.1, Requirement II.37

R_TMF518_RTM_II_0038	The alarm count retrieval capability noted in

	R_TMF518_RTM_II_0036 and the active alarm retrieval capability noted in R_TMF518_RTM_II_0037 shall allow the requesting OS to stipulate a filter. The filter is applied by the target OS before sending the active alarms to the requesting OS. Table **3-1** lists the filter conditions that shall be supported.
Source	TMF 517, Version 1.1, Requirement II.38 This also covers the following requirements from TMF 513, Version 3.0: II.110, 111, 154, 270, 287 and 288.

Table 3-1. Structure of Active Alarm Filter

Name	Meaning	Default Value
Source	Source – the field indicates the source of the alarm. There are three possibilities, i.e., • INTERNAL – alarms that have been generated internally by the target OS. The INTERNAL alarms include all root cause alarms (as determined by the target OS), all alarms related to the OS application itself, and any other alarm deduced or generated by the target OS. • EXTERNAL – alarms that have been collected from external sources and not further processed by the target OS. The EXTERNAL alarms are basically alarms that the target OS collects from other sources and simple forwards without further consolidation. • NOT_APPLICABLE – this includes both the INTERNAL and EXTERNAL alarms. The source can include only internal alarms, only external alarms or if it is left empty it means that alarms for all sources are considered.	empty
Scope	Scope (support for this field is optional) – this is either • all (i.e., all active alarms known to the top-level OS) • all the active alarms for a specified list of one or more MEs. In cases where the target OS does not know the ME associated with the alarm (e.g., the subtending management system may not provide this information), the Scope field may **not** be supported by the target OS. Also, it is recommended that the target OS supplier state on restrictions on the supported ME list in their Implementation Statement (e.g., "the MEs in the list must be from the same subnetwork", or "the ME list may not exceed ten elements"). • the top-level OS itself (i.e., just alarms pertaining to the OS application).	empty list
PerceivedSeverityList	This list is used to include only active alarms with the specified severity.	empty list
ProbableCauseList	This list is used to include only active alarms with the specified probable cause. See SD1-33 for the list of probable causes.	empty list
AcknowledgeIndication	This optional parameter can have the values ACKNOWLEDGED UNACKNOWLEDGED or	empty

	NOT_APPLICABLE. When it is not included in the filter, it means that alarms should be returned regardless of the value of this attribute.	

In addition to the filtering described in Table **3-1**, it is also possible to do attribute value matching on any combination of the attributes in an alarm. This capability is summarized in R_TMF518_RTM_II_0039. Further, note that the filter in R_TMF518_RTM_II_0039 overlaps the filtering capability in Table **3-1** with respect to PerceivedSeverityList, ProbableCauseList and AcknowledgeIndication. If both filters are used, it is recommended that filtering with respect to PerceivedSeverityList, ProbableCauseList and AcknowledgeIndication be specified in the filter described in R_TMF518_RTM_II_0039 rather than in the filter described in Table **3-1**. For backward compatibility issues, the PerceivedSeverityList, ProbableCauseList and AcknowledgeIndication filter items are not yet removed from Table **3-1**.

R_TMF518_RTM_II_0039	The Interface shall support the retrieval of active alarms based on attribute value matching performed on any combination of the attributes of an alarm. The supported attribute value matching patterns shall be conformant to the requirements TMF518_FMW_II_0023 and TMF518_FMW_II_0024.
Source	TMF518_RTM, Version 1.0

This following are examples concerning the attribute matching filter for alarm retrieval.

Simulative data:

Suppose that an OS system has alarms data described as below:

```
<Alarm>
    <eventInfo>
        <notificationId>1</notificationId>
        <objectName>
            <mdNm>HUAWEI/EMS-1</mdNm>
            <meNm>Metro1000-10.71.62.40</meNm>
        </objectName>
        <objectType>OT_MANAGED_ELEMENT</objectType>
        <osTime/>
    </eventInfo>
    <isClearable>false</isClearable>
    <layerRate>LR_Async_FOTS_1130M</layerRate>
    <probableCause>
        <type>BER_SD</type>
    </probableCause>
    <perceivedSeverity>PS_CRITICAL</perceivedSeverity>
    <serviceAffecting>SA_SERVICE_AFFECTING</serviceAffecting>
    <rcaiIndicator>true</rcaiIndicator>
    <acknowledgeIndication>AI_EVENT_ACKNOWLEDGED</acknowledgeIndication>
</Alarm>

<Alarm>
    <eventInfo>
        <notificationId>2</notificationId>
        <objectName>
            <mdNm>HUAWEI/EMS-1</mdNm>
            <meNm>OptiX3000-10.71.62.41</meNm>
```

```
            </objectName>
            <objectType>OT_MANAGED_ELEMENT</objectType>
            <osTime/>
        </eventInfo>
        <isClearable>true</isClearable>
        <layerRate>LR_Async_FOTS_150M</layerRate>
        <probableCause>
            <type>BLOCKED_FE</type>
        </probableCause>
        <perceivedSeverity>PS_INDETERMINATE</perceivedSeverity>
        <serviceAffecting>SA_NON_SERVICE_AFFECTING</serviceAffecting>
        <rcaiIndicator>false</rcaiIndicator>
        <acknowledgeIndication>AI_EVENT_UNACKNOWLEDGED</acknowledgeIndication>
    </Alarm>

    <Alarm>
        <eventInfo>
            <notificationId>3</notificationId>
            <objectName>
                <mdNm>HUAWEI/EMS-1</mdNm>
                <meNm>AMG5000-10.71.62.42</meNm>
            </objectName>
            <objectType>OT_MANAGED_ELEMENT</objectType>
            <osTime/>
        </eventInfo>
        <isClearable>true</isClearable>
        <layerRate>LR_Async_FOTS_1G8</layerRate>
        <probableCause>
            <type>DCC_FAILURE</type>
        </probableCause>
        <perceivedSeverity>PS_WARNING</perceivedSeverity>
        <serviceAffecting>SA_UNKNOWN</serviceAffecting>
        <rcaiIndicator>false</rcaiIndicator>
        <acknowledgeIndication>AI_NA</acknowledgeIndication>
    </Alarm>
```

Presence

Sample: A consumer would like to query all alarms which have an “acknowledgeIndication” value

Simulative XPath expression: “//AlarmT[exist(acknowledgeIndication)]”

Query results

All the alarms listed in the simulative data

Explanation: All the alarms have attribute "acknowledgeIndication”

Equality

Sample: A consumer would like to query all alarms whose acknowledgeIndication is ‘EVENT_UNACKNOWLEDGED’

Simulative XPath expression: “`//AlarmT[acknowledgeIndication= 'EVENT_UNACKNOWLEDGED']`”

Query results:

The second alarm

Explanation: The second alarm has acknowledgeIndication of ` EVENT_UNACKNOWLEDGED'

Comparison

Not Applicable

Substrings

Not Applicable

Any String

Sample: A consumer would like to query all alarms whose meNm contain "ME1"

Simulative XPath expression:

```
"//AlarmT[contains(eventInfo/objectName/mdNm,'HUAWEI/EMS-1')]"
```

Query results

All of the alarms

Explanation: All of the alarms have an mdNm equal to `'HUAWEI/EMS-1'

Initial String

Simulative XPath expression:

```
"//AlarmT[starts-with(eventInfo/objectName/meNm,'AMG')]"
```

Query results

The third alarm

Explanation: The third alarm has an meNm of 'AMG5000-10.71.62.42' which starts with 'AMG'

Final String

Simulative XPath expression:

```
"//AlarmT[ends-with(eventInfo/objectName/meNm,'.40')]"
```

Query results

The first alarm

Explanation: The first alarm has a meNm "Metro1000-10.71.62.40" and that is the only one ending with ".40".

3.3.1.4.3 Alarm Acknowledgement

R_TMF518_RTM_II_0040	The Interface shall allow an OS to acknowledge an alarm.

Source	TMF 517, Version 1.1, Requirement II.155

R_TMF518_RTM_II_0041	The Interface shall allow an OS to un-acknowledge an alarm.
Source	TMF 517, Version 1.1, Requirement II.156

R_TMF518_RTM_II_0042	The Interface shall allow an OS to subscribe to notifications (from an alarm owning OS) that indicate when an alarm has been acknowledged. The OS that actually acknowledges an alarm could be the OS that owns the alarm or yet another OS.
Source	TMF 517, Version 1.1, Requirement II.157

R_TMF518_RTM_II_0043	The Interface shall allow an OS to subscribe to notifications (from an alarm owning OS) that indicate when an alarm has been un-acknowledged. The OS that actually un-acknowledges an alarm could be the OS that owns the alarm or yet another OS.
Source	TMF 517, Version 1.1, Requirement II.158

3.3.1.4.4 Alarm Severity Assignment

R_TMF518_RTM_II_0044	To control the severities of alarms that may be reported by an alarm generating OS the Interface shall allow an OS to assign an Alarm Severity Assignment Profile (ASAP) to the instances of the following object classes: • Equipment • Equipment Holder • Equipment Protection Group (EPG) • Group Termination Point (GTP) • Managed Element (ME) • Operations System (OS) • Protection Group (PG) • Subnetwork Connection (SNC) • Termination Point (TP) • Topological Link (TL).
Source	TMF 517, Version 1.1, Section 4.2.4.1.4 (not listed as a requirement, however)

3.3.1.4.5 Alarm Root Cause Indication

R_TMF518_RTM_II_0045	The Interface shall allow an alarm generating OS to indicate whether an alarm is a raw (un-correlated) alarm or a root cause alarm indication.
Source	TMF 517, Version 1.1, Requirement II.223

R_TMF518_RTM_II_0046	The Interface shall allow an alarm generating OS to change its diagnosis of a root cause and send an appropriate update to other OSs that have subscribed to such events.
Source	TMF 517, Version 1.1, Requirement II.224

3.3.1.4.6 Alarm Correlation

R_TMF518_RTM_II_0047	The Interface shall allow an alarm generating OS to indicate the set of alarms (just identifiers) that are correlated to a given alarm.
Source	TMF518_RTM, Version 1.0

3.3.2 Protection Management

3.3.2.1 TP Protection Inventory

R_TMF518_RTM_II_0048	The Interface shall allow an OS to retrieve all the Protection Group (PGs) available in a specified Managed Element (ME).
Source	TMF 517, Version 1.1, Requirement II.059

The capability stated in R_TMF518_RTM_II_0048 can be used by an OS to manage protected trails between subnetworks. In the case of MSSPRing (BLSR), these protection groups also contain information about the, SPRING_NODE_ID which is needed at the time of subnetwork connection creation (i.e. the ingress/egress nodes of a ring).

R_TMF518_RTM_II_0067	The Interface shall allow an OS to retrieve a given Protection Group.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_II_0049	The Interface shall allow an OS to retrieve the names of all the Connection Termination Points (CTPs) that support Non-Preemtible Unprotected Traffic (NUT) services associated with a given Protection Group (PG).
Source	TMF 517, Version 1.1, Requirement II.274

R_TMF518_RTM_II_0050	The Interface shall allow an OS to retrieve the names of all the Connection Termination Points (CTPs) that support protected services associated with a given Protection Group (PG).
Source	TMF 517, Version 1.1, Requirement II.275

R_TMF518_RTM_II_0051	The Interface shall allow an OS to retrieve the names of all the Connection Termination Points (CTPs) that support preemptible extra traffic (unprotected services that may be preempted by other services) associated with a given Protection Group (PG).
Source	TMF 517, Version 1.1, Requirement II.276

R_TMF518_RTM_II_0069	The Interface shall allow the requesting OS to retrieve the names of the Protection Groups (PGs) containing a given Physical Termination Point (PTP).
Source	TMF 513 Version 3.0, Requirement II.289

R_TMF518_RTM_II_0052	The Interface shall allow for the delivery of and subscription to lifecycle notifications (e.g., object creation and deletion) with respect to TP protection groups.
Source	TMF518_RTM, Version 1.0

3.3.2.2 Equipment Protection Inventory

R_TMF518_RTM_II_0053	The Interface shall allow an OS to retrieve the attributes of all the Equipment Protection Groups (EPGs) available in a Managed Element (ME).
Source	TMF 517, Version 1.1, Requirement II.174

R_TMF518_RTM_II_0068	The Interface shall allow an OS to retrieve a given Equipment Protection Group.
Source	TMF518_RTM, Version 1.0

R_TMF518_RTM_II_0054	The Interface shall allow for the delivery of and subscription to lifecycle notifications (e.g., object creation and deletion) with respect to equipment protection groups.
Source	TMF518_RTM, Version 1.0

3.3.2.3 Trail and Subnetwork Connection Protection

This section addresses the interface requirements that enable an OS to discover and manage trail and subnetwork connection protection and the switching of both trails and the subnetwork connection protection.

The basic principle is one of discovery of trail protection rather than to manage protection switching via the interface.

This section only applies to SONET/SDH.

R_TMF518_RTM_II_0055	The Interface shall allow an OS to discover all trail protection schemes (both linear and ring configurations) that exist in the underlying network known to the target OS to the extent known by the target OS. It is possible that the resources of a ring (or a linear system) are split among more than one managing OS. The Interface shall not indicate if the ring is a complete ring, a portion of a complete ring or an open ring that is still in the process of being provisioned (or any linear system). The ordering of Network Elements within a ring is not explicitly indicated across the Interface. Such information may be inferred from the Topological Links passed across the Interface.
Source	TMF 517, Version 1.1, Requirement II.112

R_TMF518_RTM_II_0056	The Interface shall allow an OS to determine the traffic source of a Protection Group (PG) or a Subnetwork Connection Protection (SNCP). In addition, the requesting OS can determine the following over the interface: • The current protection switch state (whether protection switching is locked, automatic or forced). • The protection attributes (e.g. whether the scheme is unidirectional or bi-directional (also known as single or dual ended) or the protocol used for MSSPRING). • If the switching is revertive or not. • Support for 1+1 (with no extra traffic capability) or 1:N which does support extra traffic on the protection resources.
Source	TMF 517, Version 1.1, Requirement II.114

R_TMF518_RTM_II_0057	The Interface shall allow a subscribing OS to register for and the target OS to send notifications in case of switching events related to trail and subnetwork connection protection (SNCP).
Source	TMF 517, Version 1.1, Requirement II.115

R_TMF518_RTM_II_0058	The Interface shall allow an OS to request the execution of protection switch commands that are supported by a Connection Termination Point (CTP) or a Protection Group (PGP) that is currently able to perform a protection switch. CTPs are used only for protection switch commands that cannot be performed via the PGP object. For example for SNCP no PGP object exists and the protection switch operation is applied directly to a CTP. The following are the known values for SDH APS and VC Trail Protection schemes: • Lockout • Clear • Forced Switch • Manual Switch • Exerciser. See ITU-T Recommendation G.841 for definitions of the above commands.
Source	TMF 517, Version 1.1, Requirement II.116

R_TMF518_RTM_II_0059	The Interface shall allow a requesting OS to query a target OS to determine if any persistent protection switch commands have been invoked. This query shall be supported on a Connection Termination Point (CTP) and on a Protection Group (PG) basis. The query on CTP is only applicable for protection schemes that do not employ a PG. For example for SNCP protection no protection group object exists and the protection switch operation and query is applied directly on a CTP. In particular, the following protection switch information shall be obtainable from the target OS: • Type – this attribute shall represent the type of the protection for which the switch has occurred. • Switch reason – this attribute shall represent the reason for the switch. • Layer rate – this attribute shall represent the layer at which the switch has occurred. • PG – this attribute shall represent the name of the Protection Group (PG) in the case of a trail switch. Not used if the protection type is Subnetwork Connection Protection (SNCP). • Protected TP – this attribute shall represent the name of the Termination Point (TP) being protected.

	• Switch away from TP – this attribute shall represent the name of the TP being switched away from. • Switch to TP – this attribute shall represent the name of the TP that is switched to.
Source	TMF 517, Version 1.1, Requirement II.95 and 117

3.3.2.4 Equipment and TP Protection Management

R_TMF518_RTM_II_0060	The Interface shall allow an OS to determine the active Equipment instances within an Equipment Protection Group (EPG). In addition, the OS can determine following (over the Interface): • The current protection switch state (whether protection switching is locked, automatic or forced). • The protection attributes. • If the switching is revertive or not. In particular, the Interface shall allow an OS to retrieve the following switch status information for a given Equipment Protection Group (EPG): • Type – this attribute shall represent the type of the protection for which the switch has occurred. • Switch reason – this attribute shall represent the reason for the switch. • EPG – this attribute shall represent the name of the Equipment Protection Group (EPG). • Protected Equipment – this attribute shall represent the name of the Equipment being protected. • Switch to Equipment – this attribute shall represent the name of the Equipment that is switched to.
Source	TMF 517, Version 1.1, Requirement II.175 and 269

R_TMF518_RTM_II_0061	The Interface shall allow a subscribing OS to register for and the target OS to send notifications in case of an Equipment protection switch.
Source	TMF 517, Version 1.1, Requirement II.176

3.3.3 Maintenance and Diagnostic Test Management

R_TMF518_RTM_II_0062	The Interface shall allow the OS to request the set and release

	of maintenance commands that are supported by a Termination Point (TP). The following is a list of maintenance of operations that shall be supported: • Facility Loopback • Terminal Loopback • Facility Forced AIS (Upstream) • Terminal Forced AIS (Downstream) • Force RDI • Set as segment end point (ATM) – Note that un-set is provided by the already-included release action • Launch end-to-end loopback OAM cell (ATM) • Launch segment loopback OAM cell (ATM) • Local Loop Qualification (DSL) • DSL Line Supervision (DSL) See SD1-20 for further details on the specific maintenance operations. A distinct error message will be returned to distinguish between the case where a command is rejected because the current state of the target object does allow for the command to be executed and the case where the command is simply not supported.
Source	TMF 517, Version 1.1, Requirement II.137

R_TMF518_RTM_II_0063	The Interface shall allow an OS to query the target OS to determine if any persistent maintenance commands have been invoked. This query is supported with respect to the Managed Element (ME) and Termination Point (TP) objects.
Source	TMF 517, Version 1.1, Requirement II.138

3.4 Category III: Abnormal or Exception Conditions, Dynamic Requirements

3.4.1 Alarm Summary

R_TMF518_RTM_III_0064	With regard to R_TMF518_RTM_II_0037, active alarm retrieval

	can be done in the following ways: • Batched responses– in this approach the server (target OS) sends the requested information back to the client (requesting OS) in a series of responses (batches). For example, using asynchronous communication over JMS. • Batched retrieval– in this approach the client (requesting OS) retrieves the requested information from the server (target OS) via a series of requests. This is analogous to the iterator approach. For example, using http/s with synchronous communication. • Batched File transfer – in this approach the requested inventory is delivered in the form of file(s). The approach allows the client (requesting OS) to designate the location of the file(s) (by providing a URI). For example, using FTP. All of the above active alarm retrieval methods shall support the same Active Alarm output data structure schema, as well as, its batching capabilities. The partitioning of the Active Alarm document is constrained based on the atomic (non-divisible) Alarm XML object structure. Therefore, the Active Alarm document is divided in such a way that the receiver can reconstruct the document as the various segments are received.
Source	TMF 517, Version 1.1, Requirement II.37

A requesting OS may need to know the number of the current active alarms prior to retrieval of these alarms from the target OS (see Alarm Summary requirement). This knowledge will determine the appropriate retrieval mechanism: via the CCV or File Transfer.

It should be noticed that when the “all” default filter element is requested, or, in other words, when a filtering element type is not required, it is not present in the filter structure.

R_TMF518_RTM_III_0065	The Interface shall support the capability for a requesting OS to retrieve the number of active alarms known to a target OS. The request for the number of active alarms shall contain a filter parameter. The supported filters are the same as those for R_TMF518_RTM_II_0038.
Source	TMF 517, Version 1.1, Requirement II.41

R_TMF518_RTM_III_0066	The Interface shall allow a client to detect when a server OS is no longer available.
Source	TMF 517, Version 1.1, Requirement III.1

3.5 Category IV: Expectations and Non-Functional Requirements

No requirements in this category have been identified.

3.6 Category V: System Administration Requirements

No requirements in this category have been identified.

4 Use Cases

Note that all of the following use cases assume the OS (Re)starts use case has occurred as pre-condition. The OS (Re)starts use cases can be found in TMF518_FMW.

4.1 Provisioning

4.1.1 OS turns alarm reporting "on" for a TP

Use Case Id	UC_TMF518_RTM_0001
Use Case Name	OS turns alarm reporting "on" for a TP
Summary	An OS requests the target OS to activate all alarm reporting on a termination point.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends the request to activate alarm reporting to the target OS.
Description	1. The requesting OS sends the request to activate alarm reporting "on" for a specified TP. 2. The target OS validates the TP reference (e.g., name). 3. The target OS enables alarm reporting on the specified TP. The alarm reporting state of the contained TP(s) may also be enabled. 4. The target OS replies with a success indication. 5. Attribute Value Change notification(s) for the specified TP and the contained TP(s), if any, are forwarded to the notification service indicating that alarm reporting has been activated for these TP(s).
Ends When	In case of success: The requesting OS receives an indication of success of the action. In case of failure: The requesting OS receives an indication of failure of the action.
Post-Conditions	In case of success: • Alarm monitoring is enabled on the specified TP. This does not mean that alarm is reported anyway, because Alarm

	Severity Assignment Profile may perform further filtering. • The target OS has forwarded an attribute value change notification if there was an attribute value change with the enabling of alarm monitoring on the TP. In case of failure: None.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Invalid input: The TP reference is invalid. 3. Communication loss: It was not possible to reach the given ME(s).
Traceability	R_TMF518_RTM_II_0012 This is use case is a generalization of Use Case 5.5.8 from TMF 513 v3.0.

4.1.2 OS turns alarm reporting "off" for a TP

Use Case Id	UC_TMF518_RTM_0002
Use Case Name	OS turns alarm reporting "off" for a TP
Summary	The requesting OS asks that the target OS deactivate alarm reporting on a specified termination point (TP). Note: There are no side effects upon transmission behavior (propagated alarm signals e.g. AIS) associated with the TP.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends the request to deactivate alarm reporting to the target OS.
Description	1. The requesting OS sends the request to deactivate alarm reporting on the specified TP. 2. The target OS validates the TP reference (e.g., name). 3. The target OS disables alarm reporting on the specified TP. The alarm reporting state of the contained TP(s) may also be disabled. This disables alarm reporting even if an assigned Alarm Severity Assignment Profile would allow it. 4. The target OS replies with a success indication. 5. Attribute Value Change notification(s) for the specified TP and the contained TP(s), if any, are forwarded to the notification service indicating that alarm reporting has been deactivated for these TP(s).

Ends When	In case of success: The requesting OS receives an indication of success of the action. In case of failure: The requesting OS receives an indication of failure of the action.
Post-Conditions	In case of success: Alarm reporting is disabled on the specified TP and all the contained TP(s). The target OS has forwarded an attribute value change notification. In case of failure: None.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Invalid input: The TP reference is invalid. 3. Communication loss: It was not possible to reach the given ME(s).
Traceability	R_TMF518_RTM_II_0013 This is use case is a generalization of Use Case 5.5.9 from TMF 513 v3.0.

4.1.3 OS assigns an Alarm Severity Assignment Profile (ASAP) to a CTP

Use Case Id	UC_TMF518_RTM_0003
Use Case Name	OS assigns an Alarm Severity Assignment Profile (ASAP) to a CTP (or some other object type that supports the assignment of ASAPs)
Summary	The requesting OS assigns an ASAP, either previously created by the requesting OS or created by some other OS (including, possibly, the target OS), to a CTP, at a specified layer rate.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The identified ASAP already exists in the target OS. 3. In case the resource is a TP: the provided layer rate is an encapsulated layer rate of the TP. 4. The identified object (to which the ASAP is to be assigned) has to exist. If not, the ASAP should be created before starting this use case. 5. The identified object has to support the ASAP pointer feature.

Begins When	The requesting OS sends the assign ASAP request to the target OS with the specified CTP.
Description	1. The requesting OS sends a request to assign the ASAP to the addressed CTP. 2. The target OS validates the assignment request. 3. If the target OS does not support assignment of ASAPs via this interface, an exception is thrown. 4. If the ASAP name does not refer to an ASAP object, or the specified layerRate is invalid for the addressed resource, i.e., it is not an encapsulated layer rate, then an exception is thrown. 5. If the ASAP name or the resource name reference a non-existent object, then an exception is thrown. 6. If there is a currently assigned ASAP, and this assignment is fixed on target OS side, then an exception is thrown. 7. If the resource name refers to an object not supporting the ASAP pointer feature then an exception is thrown. 8. The requesting OS connects to the notification service and thus is able to receive notifications matching the filter conditions specified (if this has not been done earlier). Note: The main filtering criteria are on the notification type (i.e., alarm and/or threshold crossing alert). In addition, the requesting OS can request other filtering criteria. Any of the parameters of the parameters of the alarm can be used. See UC_TMF518_RTM_0009 and UC_TMF518_RTM_0013.
Ends When	The target OS sends a reply to the requesting OS.
Post-Conditions	In case of success: This operation causes an alarm re-evaluation of the already detected defects according to the following rules. If alarms are reportable (*): • if the severity changes from any of critical, major, minor, warning, to not alarmed, then an alarm notification with cleared is sent and the alarm is no longer available for any alarm retrieval operation. • if the severity changes from not alarmed to any of critical, major, minor, warning, then an alarm notification with the new perceivedSeverity is sent (with the current target OS/NE time) and the alarm is available for any alarm retrieval operation. • if the severity changes from any of critical, major, minor, warning, to any of critical, major, minor, warning, then the alarm re-evaluation process is not performed.

	(*) an alarm is reportable by ME/target OS when • AlarmReporting = “on” (for PTP, CTP, FTP) • alarmReportingIndicator = true (for SNC, TopologicalLink, Equipment, EquipmentHolder, GTP) • always reportable for all other objects which do not have any alarm reporting attribute. Moreover, once an alarm becomes reportable by ME/target OS then the following procedure is performed: • If the managed object has a valid aSAPpointer, then the referenced ASAP is searched for an entry that satisfies the following conditions: i) The probableCause value is the same in the alarm and in the entry, AND ii) The probableCauseQualifier value is the same in the alarm and in the entry (or the probableCauseQualifier value in the entry is empty) AND iii) The nativeProbableCause value is the same in the alarm and in the entry (or the nativeProbableCause value in the entry is empty). E.g., if the reportable alarm has LOS probableCause and an ASAP entry is found with LOS probableCause and both probableCauseQualifier and nativeProbableCause are empty strings, then that ASAP entry is accepted. If the search is successful then the associated severities are assigned. There are three possible cases: ○ If the alarm is service affecting, it is assigned the severity specified in the serviceAffecting attribute, if any. If no severity is explicitly assigned, i.e. the value of serviceAffecting attribute is ANY, then see below (*) ○ If the alarm is non service affecting, it is assigned the severity specified in the serviceNonAffecting attribute, if any. If no severity is explicitly assigned, i.e. the value of serviceNonAffecting is attribute is ANY, then see below (*) ○ If the alarm is service independent, or if the target OS does not know whether the alarm actually affects the service or not, it is assigned the severity specified in the serviceIndependentOrUnknown attribute, if any. If

	no severity is explicitly assigned, i.e. the value of serviceIndependentOrUnknown attribute is ANY, then see below (*) • If the corresponding probableCause is not found in the ASAP, or the managed object has no aSAPpointer (or the aSAPpointer value is invalid) then: (*) the alarm is assigned the default / native severity at target OS/NE side, if any, otherwise; the INDETERMINATE severity is assigned. Once a severity (including the INDETERMINATE) has been assigned, the alarm notification is emitted, except in the case of the "NOTALARMED" - cleared severity, which causes the non emission of the alarm notification. Any operation of alarm retrieval will not include such "NOTALARMED" alarms. In case of failure: Either the currently assigned ASAP is maintained, e.g. because the assignment is fixed on target OS side, or no ASAP is assigned.
Exceptions	1. Not implemented: The target OS does not support this service. 2. Processing failure: The requested operation could not be performed. 3. Invalid input: The aSAPName does not refer to an ASAP object, or layerRate is invalid for the addressed resource, i.e. it is not an encapsulated layerRate. 4. Entity not found: The aSAPName or resourceName reference an object that does not exist. 5. Unable to comply: The currently assigned ASAP object cannot be de-assigned, or resourceName refers to object not supporting ASAP pointer feature.
Traceability	R_TMF518_RTM_II_0022 and R_TMF518_RTM_II_0025 This is use case is a generalization of Use Case 5.5.21 from TMF 513 v3.0.

4.2 Protection Management

4.2.1 OS retrieves all the Protection Groups of a Managed Element

Use Case Id	UC_TMF518_RTM_0004
Use Case Name	OS retrieves all the Protection Groups of a Managed Element

Summary	The requesting OS attempts to learn about the existence of all protection groups that exist in a network element. This use case applies to both TP and equipment protection groups.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The Managed Element exists within the control of the target OS.
Begins When	Requesting OS inquires about the existence of the protection groups in a Managed Element.
Description	1. The requesting OS asks for the list of protection groups in a Managed Element. The requesting OS will send the name of the Managed Element as input. Note that the requesting OS can ask for all TP protection groups or all Equipment protection groups, but not both in the same request. 2. The target OS returns the list of all the protection groups contained in the Managed Element. 3. In the case of non-Equipment Protection Groups the target OS orders the protection group TPs in the list as follows: • The ProtectedTPs are always presented ahead of the protecting TP. • The TPs in the East direction are always presented contiguously ahead of the West directions. • In case of 4-fiber rings, there are three groups presented, two span groups and one 4-fiber ring group. • This ordering and scheme is applicable to all technologies. 4. If the target OS does not know the reversion Mode or the protection Scheme state, a value of UNKNOWN is returned. 5. For BLSR and 1:N MSP, non Pre-emptible traffic shall be ALLOWED, or NOT_ALLOWED. 6. The applicable parameters of each protection group type is returned. If not known, a value of UNKNOWN is returned. 7. The ProtectionScheme State is identified to be AUTOMATIC or FORCED_OR_LOCKED_OUT to switch. This indicates whether the protection scheme is free to switch or is constrained from switching. The protection scheme is constrained from switching when it is forced or locked. 8. The wtrTime is provided in seconds. If the target OS cannot obtain that value, a value of -1 is returned.
Ends When	The target OS completes the service.
Post-Conditions	The requesting OS is aware of the protection groups in a Network

	Element.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Invalid input: The name of the Network Element in the request does not reference a managedElement object. 3. Entity not found: The name of the Network Element references object which does not exist. 4. Communication loss.
Traceability	R_TMF518_RTM_II_0048, R_TMF518_RTM_II_0049, R_TMF518_RTM_II_0050 and R_TMF518_RTM_II_0051 This is use case is a generalization of Use Case 5.7.1 from TMF 513 v3.0.

4.2.2 Protection Switch Notification for Equipment, Trail and SNC Protection

Use Case Id	UC_TMF518_RTM_0005
Use Case Name	Protection Switch Notification for Equipment, Trail and SNC Protection
Summary	This use case describes events that occur at the network level and how the requesting OS learns of them.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The requesting has executed UC_TMF518_RTM_0007. In the case of equipment protection, the requesting OS has registered to receive equipment protection switch notifications. 3. In the case of Trail and SNC Protection the Termination Points in question are in a protection configuration. 4. In the case of Equipment, the Equipment instances in question are in a protection configuration (only M:N equipment protection has been identified, so far)
Begins When	Either a network fault has occurred or a user triggers a switch from the target OS or the Craft creating a switch in the traffic source or the requesting OS triggers a switch.
Description	1. In case of Trail protection switch (including the span switch in a 4-fiber ring configurations), the traffic source has switched from the protected to protecting or vice versa. 2. In case of a ring switch, the traffic has switched from the protected channels of a span to the protecting channels of the other span. 3. In case of a SNC protection switch, the traffic being received at

	the reliable TP (the output of the service selector), is switched from the worker TP to the protection TP or switched back. 4. The 1+1 and 1:N Trail protection (including the span switch in a 4-fiber ring) notification is raised against the Trail protection groups. 5. In the case of M:N equipment protection, the notification is raised against the equipment protection group. 6. In case of a ring switch the notification is raised against the Ring groups. 7. In case of a SNC protection switch, the notification is raised against the reliable TP. The target OS provides the following information to the requesting OS in the notification: • The protection type shall be provided to identify whether a protection switch is an Equipment protection, Trail protection or an SNC protection. • The switch reason shall be provided, which shall be Restored, Signal Fail, Signal Mismatch, Signal Degrade, Automatic Switch, Manual Switch, or Not Applicable. • In the case of Trail or SNC protection the layer rate shall be provided to which this switch is related. • The group name shall be provided, which identifies the protection group for which protection switch status is being reported. The group name shall be NULL if the protection type is SNC protection. • The protected TP shall be provided. For a SNC, this is always the reliable TP. For a 2F MSSP ring notification, this is the TP that is/was inactive during the switch. For a 4F MSSP ring switch notification, this is the worker TP that is/was inactive during the switch. For a 1:N MSP switch notification, this is the worker TP for which the protection switch occurred. For a revertive 1+1 MSP, this is always the worker TP. For a non-revertive 1+1 MSP switch notification, this is the TP that is inactive after the switch. In the case of equipment protection, the protected equipment instance shall be provided. For an M:N group, the protected equipment instance always identifies the worker equipment instance for which the switch occurred. • The switchAwayFromTP shall be provided. For a 2F MSSP ring switch, this is the TP that switched. For a 4F MSSP ring span switch, this is one of the TPs in the Trail1:N groups (worker or protection). In the case of equipment protection, the switchAwayFromEquipment is provided (this identifies the equipment instance being switched away from). • The switchToTP shall be provided, which identifies the TP that is the active source after the switch, or currently

	active if no protection switch is currently active. In the case of equipment protection, the switchToEquipment is provided (this identifies the equipment instance which is being switched to).
Ends When	The requesting OS is notified of the switch.
Post-Conditions	Subject to filter conditions, the requesting OS knows of the present traffic source.
Exceptions	Not applicable.
Traceability	R_TMF518_RTM_II_0061 This is use case is a generalization of Use Case 5.7.2 from TMF 513 v3.0.

4.2.3 OS retrieves the protection switch information for Equipment, Trail and SNC Protection

Use Case Id	UC_TMF518_RTM_0006
Use Case Name	OS retrieves the protection switch information for Equipment, Trail and SNC Protection
Summary	This use case describes how an OS learns about the traffic source of the protection groups and protected SNCs.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The Termination Points in question are in a protection configuration (Trail or SNC Protection). 3. In the case of Equipment, the equipment instances in question are in a protection configuration.
Begins When	The requesting OS wishes to discover the present traffic source of a Trail or a SNC Protection configuration, or the active equipment instance in an Equipment protection group.
Description	The target OS provides the following information to the requesting OS in the response to a query regarding the current protection switch status of a protection group or a SNC: 1. The protection type shall be provided to identify whether a protection switch is a Trail protection switch or a SNC protection switch. 2. The switch reason shall be provided, which shall be Restored, Signal Fail, Signal Mismatch, Signal Degrade, Automatic Switch, Manual Switch, or Not Applicable. 3. The layerRate shall be provided, to which this switch is

	relevant (not applicable for equipment protection). 4. The group name shall be provided, which identifies the protection group for which protection switch status is being reported. The group name shall be NULL if the protection type indicates SNC protection. 5. TP Protection: The protected TP shall be provided. For a SNC protection, this is always the reliable TP. For a retrieval of a 2Fiber MS SP ring, each TP is protected, and two SwitchData structures are returned. For a retrieval of a 4Fiber MS SP ring, each worker TP is protected, and two SwitchData structures are returned. For a retrieval of a 1:N Trail protection, each worker TP is protected, and N SwitchData structures are returned. For a revertive 1+1 Trail protection, this is always the worker TP. For a retrieval of a non-revertive 1+1 Trail protection switch, this is the active TP. 6. Equipment Protection: For a retrieval of an M:N group, the protected equipment always identifies a worker equipment instance. In this case, N ESwitchData structures are returned as a result of retrieve ESwitchData request (one for each worker equipment instance). 7. The switchToTP shall be provided, which identifies the TP that is the active source after the switch, or currently active if no protection switch is currently active. 8. In the case of equipment protection, the protected equipment instance shall be provided. For an M:N group, the protected equipment instance always identifies the worker equipment instance for which the switch occurred.
Ends When	The requesting OS is presented with all the information.
Post-Conditions	The requesting OS knows about the traffic source.
Exceptions	1. Not implemented: The target OS is unable to support this service. 2. Processing failure: The requested operation could not be performed. 3. Invalid input: The input object does not reference a protection group or a reliable CTP of a SNC object. 4. Entity not found: The input object does not exist. 5. Communication loss.
Traceability	R_TMF518_RTM_II_0053 and R_TMF518_RTM_II_0055 This is use case is a generalization of Use Case 5.7.3 from TMF 513 v3.0.

4.2.4 OS registers to receive protection switch notifications

Use Case Id	UC_TMF518_RTM_0007
Use Case Name	OS registers to receive protection switch notifications
Summary	The requesting OS registers at the notification service related to the target, sets the appropriate filter to receive protection switch notifications, and connects to the notification service.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The requesting OS has a reference to the notification service used by the target OS.
Begins When	The requesting OS sends a request to register itself at the notification service related to the target OS.
Description	**Note:** The requesting OS registers at the notification service related to the target OS as a consumer of notifications (if this has not been done earlier). **Note:** The requesting OS sets the filter criteria needed to receive protection switch notifications from the target OS via the notification service. **Note:** The requesting OS connects to the notification service and thus is able to receive notifications matching the filter conditions specified (if this has not been done earlier). 3) The main filtering criteria are on the notification type (i.e., protection switch). In addition, the requesting OS can request other filtering criteria. Any of the parameters of the filterable body of the protection switch notification (defined in TMF518_NRA) can be used.
Ends When	In case of success: The requesting OS receives a positive acknowledgement to its connection request to the notification service. In case of failure: The target OS returns an error indication.
Post-Conditions	In case of success: The specified filter(s) are set up or modified. In case of failure: The requesting OS receives a negative acknowledgement to a request for registration, filter building or connection or a request times out.
Exceptions	Filter creation: Invalid grammar

	Filter building: Invalid constraint Connection phase: Illegal consumer type Consumer already connected
Traceability	R_TMF518_RTM_II_0057 and R_TMF518_RTM_II_0061 This is use case is a generalization of Use Case 5.7.4 from TMF 513 v3.0.

4.2.5 OS invokes protection switch lockout to an SNC

Use Case Id	UC_TMF518_RTM_0008
Use Case Name	OS invokes protection switch lockout to an SNC
Summary	The requesting OS applies protection switch lockout to a reliable CTP of a SNC that is protected by SNCP.
Actor(s)	Requesting OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The requesting OS has determined which CTPs participate in the SNCP switch.
Begins When	A request to apply a protection command is applied.
Description	The command is applied to the reliable CTP that is defined as being able to perform a protection switch.
Ends When	The target responds that the command was applied or an exception is thrown.
Post-Conditions	1. Traffic has been switched to the TP identified by toTPName. 2. The protection switch status of the reliable CTP has changed.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Unable to comply: The CTP is not performing a protection switch in a SNCP. 3. Not implemented: The target OS does not support this service. 4. Invalid input: The input object does not reference a protection group or a reliable CTP of a SNC object. 5. Entity not found: The input object does not exist. 6. Communication loss.

Traceability	R_TMF518_RTM_II_0058 This is use case is a generalization of Use Case 5.7.5 from TMF 513 v3.0.

4.3 Fault Management

4.3.1 Active Alarm Retrieval

The use cases in this subsection describe several scenarios for retrieving active alarms from an existing *target OS* by another operation system (*requesting OS*) and a use case concerning the retrieval of the count of active alarms. The alarm retrieval use cases are the same except for the communication style.

4.3.1.1 Get Active Alarm Counts

Use Case Id	UC_TMF518_RTM_0009
Use Case Name	Get Active Alarm Counts
Summary	The requesting OS requests a count of the current number of active alarms meeting some specified criteria from the target OS. Due to the dynamic nature of the active alarms during the counting process, this is a best effort operation. The target OS returns the result of the counting process.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends a request to the target OS to count the active alarms in the alarms repository.
Description	1. The requesting OS sends to the target OS over the CCV a request to count the current number of active alarms. The request message has the following parameters: a. ***Filter*** *– attribute used to restrict the set of alarms count returned to the requesting OS (the allowable filters are defined in* Table **3-1***)* 2. The *target OS* counts the current number of active alarms satisfying the filtering criteria and returns the result information as an Integer value. If the Filter parameter is empty, all active alarms will be counted.
Ends When	In case of success: The requesting OS receives the response messages with an integer representing the number of active alarms. In case of failure: The requesting OS receives an exception.

Post-Conditions	In case of success: The requesting OS has the requested information from the target OS. In case of failure: The requesting OS does not have the requested information from the target OS.
Exceptions	1. Not implemented: *The target OS* does not support this service. 2. Invalid input: An input parameter is invalid (e.g., invalid parameter syntax) 3. Processing failure: The requested operation could not be performed. 4. Communications failure: Anticipated replies have not been received by the requesting OS.
Traceability	R_TMF518_RTM_II_0036 This is use case is based on UC_MTOSI_21 from TMF 517 v1.2.

4.3.1.2 Active Alarms Retrieval – Message Communication Style

Use Case Id	UC_TMF518_RTM_0010
Use Case Name	Active Alarms Retrieval – Message Communication Style
Summary	The requesting OS requests all managed active alarms meeting some specified criteria from the target OS. The result is returned in one or more batches over the CCV using the message communication style (see definition of communication styles). The target OS returns all active alarms satisfying the scope and filter constraints of the requesting OS.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends a request to the target OS to retrieve all active alarms.
Description	1. The *requesting OS* sends an active alarms retrieval request to the *target OS* over the CCV. The request message has the following parameters: a. ***Filter*** *– attribute used to restrict the set of alarms returned to the requesting OS (the allowable filters are defined in Table* **3-1***)* b. ***ReplyToAddress****–address (e.g., URI or JMS topic name) where the target OS shall send the responses.* c. ***RequestedBatchSize*** *– the maximum*

	number of alarm units to be returned in each batch from the target OS. d. ***CorrelationID*** *– this attribute is provided by the requesting OS and then returned in each response by the target OS to allow for correlation by the requesting OS.* 2. The *target OS* sends an acknowledgment message to the **ReplyToAddress** over the CCV, signifying successful validation of the input parameters. This allows the *requesting OS* to manage the returned batches more efficiently. 3. The *target OS* prepares the specified active alarms list and returns the information in a series of batches. Each batch has (at most) the number of items defined in the **RequestedBatchSize** parameter. Each batch has a **CorrelationID**, a **SequenceNumber** and a Boolean marker **endOfReply** which is set to true in the last batch.
Ends When	In case of success: The *requesting OS* receives all response messages up to the sequence number of the last batch with **endOfReply** set to true. In case of failure: The *requesting OS* receives an exception.
Post-Conditions	In case of success: The *requesting OS* has the requested active alarms information from the *target OS*. In case of failure: The *requesting OS* does not have the entire requested inventory information from the *target OS*.
Exceptions	1. Not implemented: *The target OS* does not support this service. 2. Invalid input: An input parameter is invalid (e.g., invalid parameter syntax) 3. Processing failure: The requested operation could not be performed. 4. Communications failure: Anticipated replies have not been received by the *requesting OS.*
Traceability	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_22 from TMF 517 v1.2.

4.3.1.3 Active Alarms Retrieval – RPC Communication Style

Use Case Id	UC_TMF518_RTM_0011
Use Case Name	Active Alarms Retrieval – RPC Communication Style
Summary	The *requesting OS* requests that the *target OS* prepare a specified portion of its active alarms for subsequent retrieval. The results will be retrieved over the CCV using the RPC communication style (see definition of *communication styles*).
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends a request to the target OS to retrieve all active alarms.
Description	1. The requesting OS sends an active alarms retrieval request to the target OS over the CCV. The request message shall provide the following parameters: a. ***Filter*** *– attribute used to restrict the set of alarms returned to the requesting OS (the allowable filters are defined in Table* **3-1***).* b. ***RequestedBatchSize*** *– the maximum number of alarm units to be returned in each requested batch from the target OS.* 2. The *target OS* sends the first batch of active alarms back to the requesting OS (the number of items is determined by the **RequestedBatchSize**). In the same response, the *target OS* also provides the **IteratorAddress**. This address (e.g., URI) is used by the *requesting OS* to control the batch retrieval. 3. The *requesting OS* proceeds to retrieve the next batch of active alarms from the iterator. The **RequestedBatchSize** can be different from the original value provided for **RequestedBatchSize** and can, in fact, vary with each request made by the requesting OS. 4. Once the *requesting OS* has the address of the iterator, it can (at any time) request the size of the iterator. Note that this is the total number of items in the iterator (minus the initial batch). 5. The *target OS* should provide an end of file indication in the last batch retrieved by the *requesting OS*. 6. Once the last batch is retrieved, the *requesting OS* could request that the *target OS* delete the iterator. The target OS could also automatically delete the iterator after the last batch.
Ends When	In case of success: The requesting OS has retrieved the last batch.

	In case of failure: The requesting OS receives an exception.
Post-Conditions	In case of success: The requesting OS has all the requested active alarms information from the target OS. In case of failure: The requesting OS does not have the entire requested active alarms information from the target OS.
Exceptions	1. Not implemented: The target OS does not support this service. 2. Invalid input: Any input parameter is invalid (e.g., invalid filter syntax) 3. Processing failure: The requested operation could not be performed. 4. Communications failure: The requesting OS can not reach the iterator.
Traceability	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_23 from TMF 517 v1.2.

4.3.1.4 Active Alarms Retrieval – File Transfer

Use Case Id	UC_TMF518_RTM_0012
Use Case Name	Active Alarms Retrieval – File Transfer
Summary	The *requesting OS* requests all active alarms from the *target OS*. Results are returned in file(s) (via ftp, http, or some other file transfer approach). The *requesting OS* provides through the request all the necessary instructions on processing the response into a desired active alarms file format in terms of maximum file size (in terms of the number of object instances), file compression, file packing, and the rootname of the generated file(s). The *requesting OS* specifies in the request all the necessary instructions to allow transfer of the file(s) to a remote destination. The *target OS* returns all active alarms satisfying the filter constraints of the *requesting OS*.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The *requesting OS* sends a request to the *target OS* over the CCV to return the active alarms data in one or more files.
Description	1. The *requesting OS* sends the request command to the *target OS* over the CCV. The request message has the following parameters:

a. ***Filter*** *– attribute used to restrict the set of alarms returned to the requesting OS (the allowable filters are defined in Table* **3-1***)*

b. **File Location –** URI provided by the requesting OS indicating the indicating the rootname of the file(s) to be produced and location of where to place the retrieved XML active alarms file(s). See RFC 3986 for details on URI syntax.

c. **Batch Size** - The maximum number of alarm objects to be included in an active alarm file (batch). These objects are atomic from an XML structure standpoint, which means they cannot be divided in two different batches. Therefore, depending on the active alarm output size several files may be generated.

Default behavior (when request parameter is omitted) is to generate the active alarms retrieval in a single file (unbounded number of objects).

d. **Compression Type** – The type of compression to apply to the generated active alarm file(s). The following two options are not currently supported; NO_COMPRESSION, GZIP.

Default behavior (when request parameter is omitted) is NO_COMPRESSION. Implementation of this file processing instruction by the *Target OS* is optional, and any incompatible request shall be handled with the appropriate exception.

Vendor extension of this attribute shall be permitted.

e. **Packing Type** – The type of packing to apply to all the active alarm XML file(s) generated from the same request. The following three options are defined; NO_PACKING, ZIP, TAR.

Default behavior (when request parameter is omitted) is NO_ PACKING. Implementation of this file processing instruction by the *Target OS* is optional, and any incompatible request shall be handled with the appropriate exception.

Vendor extension of this attribute shall be permitted.

2. The *target OS* performs the upload of the file(s) to the specified location.
3. The *target OS* uses the NT_FILE_TRANSFER_STATUS notification (defined in TMF518_FMW) to indicate when the file transfer is complete or when a failure has occurred. Intermediate progress events may be indicated by setting the “transferStatus” to FT_IN_PROGRESS with “percentCompete” in the range of 0…100. The only mandatory event is sent at the end of the file transfer when either “transferStatus” is set to FT_COMPLETED with “percentCompete” = 100, or “transferStatus” is set to FT_FAILED with “failureReason” filled in.

Remarks

- The *requesting OS* specifies the Universal Resource Indicator

(URI) denoting the filename and location of where to send the retrieved file. See RFC 3986 for details on URI syntax.

- The URI shall contain:
  a. Scheme to identify the file transfer protocol (e.g., FTP)
  b. Authority to identify the user information (userid and password) as well as the remote destination network access (host and port),
  c. Path to identify the system location where the file(s) are to be sent, as well as the rootname used in all generated active alarm file(s).

e.g.,
ftp://userName:password@hostName.domain.com/uploadDir/fileName.extention

Note that ftp may be replaced with another supported protocol such as http/s.

- The generation of the active alarms file(s) is subject to the following recommendations regarding batch files and packing:
  - All batch files and package file generated for a given request shall use the ***rootname*** that is provided in the ***File Location*** URI.
  - Any compressed batch file shall use the following filename extension based on the associated compression format:
    - GZIP: ".gz"
  - Any package file shall use the following filename extension based on the associated packing format:
    - ZIP: ".zip"
    - TAR: ".tar"
  - Packing of the batch files shall always be carried out with the compressed batch files.
  - When multiple batch files are produced as output of a request, a unique sequential number is associated to each one of them. The sequential number always starts with the number 1 for the first generated batch file. And, sequential number is incremented by 1 for each subsequently generated batch file. The sequential number is always found just before the .xml extension in the batch filename and separated from the rootname by an underscore character.
  - Refer to table below for details on XML file naming conventions
    - The transfer of the file(s) will overwrite any file(s) that already exist with the same name in the destination system.

Batch File	Compression	Packing	Dataset description	Notes
Single	OFF	OFF	rootname.xml	
		ON	rootname.xml	No packing (unnecessary)
	ON	OFF	rootname.xml.gz	
		ON	rootname.[tar|zip] rootname.xml.gz	

	Multiple	OFF	OFF	rootname_<SN>.xml	The sequential number <SN> for a set of batch files
			ON	rootname.[tar|zip] rootname_<SN>.xml	The sequential number <SN> for a set of batch files
		ON	OFF	rootname_<SN>.xml.gz	The sequential number <SN> for a set of batch files
			ON	rootname.[tar|zip] rootname_<SN>.xml.gz	The sequential number <SN> for a set of batch files
Ends When	In case of success: The requesting OS receives the response message signifying the file transfer is completed. In case of failure: The requesting OS receives an exception or gets an indication that the file transfer has failed.				
Post-Conditions	In case of success: The requesting OS has the requested active alarms information from the target OS. In case of failure: The requesting OS does not have the requested active alarms information from the target OS.				
Exceptions	1. Not implemented: The target OS does not support this service. 2. Invalid input: Any input parameter is invalid (e.g., invalid filter syntax) 3. Processing failure: The requested operation could not be performed. 4. Communications failure: The requesting OS can not reach the file location 5. Unsupported Compression Format 6. Unsupported Packing Format				
Traceability	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_24 from TMF 517 v1.2.				

4.3.2 OS registers to receive alarms from a target OS

Use Case Id	UC_TMF518_RTM_0013
Use Case Name	OS registers to receive alarms from a target OS

Summary	An OS registers with the notification service on the CCV, sets the appropriate filter to receive alarm notifications and connects to the notification service.
Actor(s)	Registering OS
Pre-Conditions	• All OSs involved in the use case have successfully executed the OS (Re)starts use case. • The registering OS has a reference for the notification service on the CCV. • The target OS is on the same CCV as the registering OS and is associated with the notification service.
Begins When	The registering OS sends a request to register itself at the notification service for the alarms generated by the target OS (or some subset of the alarms as restricted by a specified filter).
Description	1. The OS registers at the notification service related to the target OS as a consumer of notifications (if this has not been done earlier). 2. The registering OS sets the filter criteria needed to receive alarm notifications from the target OS via the notification service. 3. The registering OS connects to the notification service and thus is able to receive notifications matching the filter conditions specified (if this has not been done earlier).
Ends When	In case of success: The registering OS receives a positive acknowledgement to its connection request to the notification service. In case of failure: The notification service returns an error indication.
Post-Conditions	In case of success: The specified filter(s) are set-up or modified. In case of failure: The registering receives a negative acknowledgement to a request for registration, filter building or connection or a request times out.
Exceptions	Filter creation: Invalid grammar Filter building Invalid constraint Connection phase: Illegal consumer type Consumer already connected
Traceability	R_TMF518_RTM_II_0027, R_TMF518_RTM_II_0028 and

	R_TMF518_RTM_II_0029 This is use case is a generalization of Use Case 5.8.3 from TMF 513 v3.0.

4.3.3 OS registers to receive RCAIs only, raw alarms only, or both RCAIs and raw alarms from a target OS

This use case is a special case of UC_TMF518_RTM_0013.

Use Case Id	UC_TMF518_RTM_0014
Use Case Name	OS registers to receive RCAIs only, raw alarms only, or both RCAIs and raw alarms from a target OS
Summary	An OS registers at the notification service on the CCV, sets the appropriate filter to receive RCAIs, raw alarms or both, and connects to the notification service.
Actor(s)	Registering OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The registering OS has a reference for the notification service on the CCV. 3. The target OS is on the same CCV as the registering OS and is associated with the notification service.
Begins When	The registering OS sends a request to register itself at the notification service for RCAIs only, raw alarms only or both with respect to a give target OS.
Description	1. The OS registers at the notification service as a consumer of notifications with respect to the give target OS (if this has not been done earlier). 2. The registering OS sets the filter criteria needed to receive RCAIs, raw alarms or both from the given target OS via the notification service. 3. The registering OS connects to the notification service and thus is able to receive notifications matching the filter conditions specified (if this has not been done earlier). **Note**: In addition to the notification type, the registering OS can request filtering on any of the alarm parameters noted in Table **3-1**. For this use case, it would be necessary to filter on the RCAI field.
Ends When	In case of success: The registering OS receives a positive acknowledgement to its connection request to the notification service. In case of failure:

	The notification service returns an error indication.
Post-Conditions	In case of success: The specified filter(s) are set-up or modified. In case of failure: The registering receives a negative acknowledgement to a request for registration, filter building or connection or a request times out.
Exceptions	Filter creation: Invalid grammar Filter building Invalid constraint Connection phase: Illegal consumer type Consumer already connected
Traceability	R_TMF518_RTM_II_0029 and R_TMF518_RTM_II_0045 This is use case is a generalization of Use Case 5.8.4 from TMF 513 v3.0.

4.3.4 Alarm owning OS determines a more appropriate root cause than one previously reported

Use Case Id	UC_TMF518_RTM_0015
Use Case Name	Alarm owning OS determines a more appropriate root cause than one previously reported
Summary	The OS owning a particular alarm has previously reported a root cause and would like to revise the reported root cause.
Actor(s)	Alarm owning OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The alarm owning OS is associated with the notification service on the CCV. 3. The alarm owning OS has previously sent a root cause and the alarm is still active.
Begins When	The alarm owning OS determines a root cause that supersedes one or more previously reported (and still uncleared) root cause alarm indications.
Description	1. The alarm owning OS clears one or more root cause alarms that have been previously sent to the notification service. The alarm owning OS does not clear the raw alarms associated with the

	previous root cause alarms. 2. The alarm owning OS sends a new root cause alarm to the notification service. The idea is that this new root cause is more accurate than the initial root cause alarm(s).
Ends When	In case of success: The registered OSs has received the new root cause alarm indication, and have received the clear(s) for the previous root cause alarm indication(s) that are superseded by the new root cause alarm indication. In case of failure: The registered OSs have not received the new root cause alarm and clear(s) for the previous root cause alarm indication(s) that are superseded by the new root cause alarm indication.
Post-Conditions	In case of success: The registered OSs understand that the initial root cause alarm(s) is cleared and the new root cause alarm is active. In case of failure: The registered OSs are not aware that the initial root cause alarm has been upgraded/corrected.
Exceptions	None
Traceability	R_TMF518_RTM_II_0046 This is use case is a generalization of Use Case 5.8.5 from TMF 513 v3.0.

4.3.5 Alarm generating OS Notifies Registered OSs of Alarms

Use Case Id	UC_TMF518_RTM_0016
Use Case Name	Alarm generating OS Notifies Registered OSs of Alarms or Threshold Crossing Alert (TCA)s
Summary	
Actor(s)	Alarm generating OS
Pre-Conditions	1. The OS on the CCV that are interested in receiving alarms for the given OS have executed UC_TMF518_RTM_0013. 2. The notification service is available.
Begins When	The alarm generating OS detects an alarm.
Description	1) The alarm generating OS detects an alarm and generates a notification to inform registered OSs. The alarm generating OS

	does not generate the alarm notification if the object emitting the alarm has the alarm reporting attribute switched off, or the Alarm Severity Assignment Profile severity is assigned to NOTALARMED for that detected alarm probable cause. 2) The registered OS receive the notification from the notification service.
Ends When	In case of success: The registered OSs receive the notification. In case of failure: The registered OSs do not receive the notification.
Post-Conditions	In case of success: The databases of the registered OSs and alarm generating OS remains aligned with respect to the alarm. In case of failure: The databases of the registered OSs and alarm generating OS are misaligned with respect to the alarm.
Exceptions	None
Traceability	R_TMF518_RTM_II_0031 This is use case is a generalization of Use Case 5.8.6 from TMF 513 v3.0.

4.3.6 Alarm Acknowledgement in the OS (other than the alarm owning OS)

Use Case Id	UC_TMF518_RTM_0017
Use Case Name	Alarm Acknowledgement in the OS (other than the alarm owning OS)
Summary	The operator acknowledges one or more active alarms in an OS (other than the alarm owning OS). This is referred to as the “acknowledging OS”. The acknowledging OS forwards the alarm acknowledgment through an interface operation to the alarm owning OS. The acknowledging OS will be notified in case that the alarm acknowledgment operation is successful.
Actor(s)	Acknowledging OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The alarms to be acknowledged are active and unacknowledged and the Alarm bearing objects are managed by the target OS.
Begins When	The operator acknowledges one or more alarms in the acknowledging OS.

Description	1. The acknowledging OS issues an interface request to acknowledge the alarm in the alarm owning OS. 2. The alarm owning OS acknowledges the alarm in the NE (if applicable). This could involve communications via several intermediate systems. 3. The alarm is updated as “acknowledged” in the alarm owning OS. 4. The acknowledging OS is then notified that the alarm has been acknowledged in the alarm owning OS.
Ends When	In case of success: The acknowledging OS receives an alarm acknowledged notification after the operation completes successfully. In case of failure: No alarms are acknowledged in the alarm owning OS and the acknowledging OS does not receive any “alarm acknowledged” notifications.
Post-Conditions	In case of success: All successfully acknowledged alarms are in an acknowledged state in the alarm owning OS. In case of failure: Alarms that cannot be acknowledged are indicated to the acknowledging OS. The acknowledging OS receives alarm notifications with an “alarm acknowledged” indication for all alarms successfully acknowledged in the alarm owning OS.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Invalid input: The input parameter are syntactical incorrect. 3. Not implemented: The alarm owning OS does not support this service.
Traceability	R_TMF518_RTM_II_0040 and R_TMF518_RTM_II_0042 This is use case is a generalization of Use Case 5.8.7 from TMF 513 v3.0.

4.3.7 Alarm Unacknowledgement in the OS (other than the alarm owning OS)

Use Case Id	UC_TMF518_RTM_0018
Use Case Name	Alarm Unacknowledgement in the OS (other than the alarm owning

	OS)
Summary	The operator unacknowledges one or more active alarms in an OS (other than the alarm owning OS). This is referred to as the "unacknowledging OS". The unacknowledging OS forwards the alarm unacknowledgment through an interface operation to the alarm owning OS. The unacknowledging OS will be notified in case that the alarm unacknowledgment operation is successful.
Actor(s)	Unacknowledging OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The alarms to be unacknowledged are active, acknowledged, and the Alarm bearing objects are managed by the target OS.
Begins When	The operator unacknowledges one or more alarms in the unacknowledging OS.
Description	1. The unacknowledging OS issues an interface request to unacknowledge the alarm in the alarm owning OS. 2. The alarm owning OS unacknowledges the alarm in the NE (if applicable). This could involve communications via several intermediate systems. 3. The alarm is updated as "unacknowledged" in the alarm owning OS. 4. The unacknowledging OS is then notified that the alarm has been unacknowledged in the alarm owning OS. **Note**: If an alarm is already unacknowledged, the alarm owning OS should just confirm this fact to the unacknowledging OS rather than raise an exception.
Ends When	In case of success: The unacknowledging OS receives an alarm unacknowledged notification after the operation completes successfully. In case of failure: No alarms are unacknowledged in the alarm owning OS and the unacknowledging OS does not receive any "alarm unacknowledged" notifications.
Post-Conditions	In case of success: All successfully unacknowledged alarms are in an unacknowledged state in the alarm owning OS. In case of failure: Alarms that cannot be unacknowledged are indicated to the unacknowledging OS. The acknowledging OS receives alarm notifications with an

	"alarm acknowledged" indication for all alarms successfully acknowledged in the alarm owning OS.
Exceptions	1. Processing failure: The requested operation could not be performed. 2. Invalid input: The input parameter are syntactical incorrect. 3. Not implemented: The alarm owning OS does not support this service.
Traceability	R_TMF518_RTM_II_0041 and R_TMF518_RTM_II_0043 This is use case is a generalization of Use Case 5.8.8 from TMF 513 v3.0.

4.3.8 Alarm Acknowledgement in the alarm owning OS

Use Case Id	UC_TMF518_RTM_0019
Use Case Name	Alarm Acknowledgement in the alarm owning OS
Summary	The operator acknowledges alarms in the alarm owning OS through GUI Cut-Through or directly on the OS GUI according to the OS's alarm lifecycle acknowledgement steps.
Actor(s)	Operator (of the alarm owning OS)
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. OSs interested in receiving notification of the alarm acknowledgement have executed UC_TMF518_RTM_0013.
Begins When	The operator acknowledges one or more alarms in the alarm owning OS.
Description	1. The alarm owning OS attempts to acknowledge the alarm in the NE (if applicable). This could be done via several intermediate systems. 2. The alarm is updated as "acknowledged" in the alarm owning OS and the NE, if applicable. 3. Registered OSs are then notified that the alarm has been acknowledged in the alarm owning OS.
Ends When	In case of success: The registered OSs receive an alarm acknowledged notification after the alarm owning OS operation completes successfully. In case of failure:

	The alarm is not acknowledged in the alarm owning OS and the registered OSs do not receive an “alarm acknowledged” notification.
Post-Conditions	In case of success: The alarms are acknowledged in the alarm owning OS. The registered OSs receive an alarm notification with an “alarm acknowledged” indication for those alarms that are acknowledged. In case of failure: The alarms are still unacknowledged in the alarm owning OS. The registered OSs have not received an indication that the alarms in question have been acknowledged.
Exceptions	1. Internal Error 2. Communication Loss
Traceability	R_TMF518_RTM_II_0042 This is use case is a generalization of Use Case 5.8.9 from TMF 513 v3.0.

4.3.9 Requesting OS reconciles Unacknowledged Active Alarms from a target OS

Use Case Id	UC_TMF518_RTM_0020
Use Case Name	Requesting OS reconciles Unacknowledged Active Alarms from a target OS
Summary	The requesting OS requests the current list of active unacknowledged OS-specific and non-OS-specific alarms that are under the control of a target OS(both those raised by the underlying systems and those raised by the target OS itself). Some alarms may be filtered out (excluded) by for example specifying their probable causes or severities. Other filters are possible as noted in Table **3-1**.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends a request to the target OS for the current list of active unacknowledged OS-specific and non-OS-specific alarms.
Description	1. The requesting OS sends a request to the target OS for its summary of unacknowledged active OS-specific and non-OS-

	specific alarms, with parameters to filter out (exclude) certain kinds of alarms if the requesting OS does not wish to receive the whole list. These filtering criteria are independent of the filtering set up by the requesting OS for the notification service. 2. The target OS responds to the requesting OS by returning a list of active unacknowledged alarms that meet the specified filtering criteria or the request times out.
Ends When	In case of success: The requesting OS receives the list of active unacknowledged OS-specific and non-OS-specific alarms. In case of failure: The requesting OS receives a failure indication.
Post-Conditions	In case of success: The requesting OS receives the list of active unacknowledged OS-specific and non-OS-specific alarms. In case of failure: The requesting OS receives a failure indication.
Exceptions	Processing failure: The requested operation could not be performed.
Traceability	R_TMF518_RTM_II_0037 and R_TMF518_RTM_II_0038 This is use case is a generalization of Use Case 5.8.10 from TMF 513 v3.0.

4.3.10 Requesting OS reconciles Unacknowledged Active Alarms for a specified list of Managed Elements

This use case is a specialization of UC_TMF518_RTM_0020.

Use Case Id	UC_TMF518_RTM_0021
Use Case Name	Requesting OS reconciles Unacknowledged Active Alarms for a specified list of Managed Elements
Summary	The requesting OS requests the current list of active unacknowledged OS-specific and non-OS-specific alarms that are under the control of a target OS (both those raised by the underlying systems and those raised by the target OS itself) with respect to a given list of MEs. Some alarms may be filtered out (excluded) by for example specifying their probable causes or severities. Other filters are possible as noted in Table **3-1**.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS

	(Re)starts use case.
Begins When	The requesting OS sends a request to the target OS for the current list of active unacknowledged OS-specific and non-OS-specific alarms with respect to a given list of MEs.
Description	
Ends When	In case of success: The requesting OS receives the list of active unacknowledged OS-specific and non-OS-specific alarms with respect to the given list of MEs. In case of failure: The requesting OS receives a failure indication.
Post-Conditions	In case of success: The requesting OS receives the list of active unacknowledged OS-specific and non-OS-specific alarms with respect to the given list of MEs. In case of failure: The requesting OS receives a failure indication.
Exceptions	Processing failure: The requested operation could not be performed.
Traceability	R_TMF518_RTM_II_0037 and R_TMF518_RTM_II_0038 This is use case is a generalization of Use Case 5.8.11 from TMF 513 v3.0.

4.3.11 Event generating OS discards an event that was to be sent to the notification service

This use case assumes the notification service can fail independently of the CCV. This is the case when CORBA is used to support the CCV. In the case that JMS is used both for the CCV transport and for notification services, it may not be possible for independent failure of the CCV transport and notification service. In this case, the use case does not apply.

Use Case Id	UC_TMF518_RTM_0022
Use Case Name	Event generating OS discards an event that was to be sent to the notification service
Summary	The event generating OS discards an event because the notification service is unavailable or for some issue internal to the OS. The registered OSs are informed about this and must not assume to be synchronized any longer.
Actor(s)	Event generating OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the

	OS (Re)starts use case. 2. Event generating OS discards an event.
Begins When	The event generating OS discards an event to be sent to the registered OSs.
Description	1. The event generating OS determines the type of notification that has been discarded. 2. If the notification is about object creation (OC), object deletion (OD), attribute value change (AVC), state change (SC) or route change (RC), the event generating OS informs the registered OSs that a lifecycle event was discarded. For all other notification types the registered OSs are informed that an alarm was discarded. The event generating OS may use operations offered by the registered OSs to do this (i.e., to report event loss before delivery to the notification service) or when the notification service is available, it may send alarms with an appropriate probable cause. 3. If a 2nd event has to be discarded, the event generating OS informs the registered OSs only if it didn't inform these OSs about this type of notification already (i.e., lifecycle event vs. alarm). Thus, during a problem interval when events cannot be delivered to the notification service, the registered OSs will get informed about the loss of events at most twice, once about lifecycle events and once about alarms.
Ends When	In case of success: The registered OSs are aware that an event has been discarded. In case of failure: The registered OSs are unaware that an event has been discarded.
Post-Conditions	In case of success: The registered OSs are aware that an event has been discarded. In case of failure: The registered OSs are unaware that an event has been discarded.
Exceptions	None
Traceability	R_TMF518_RTM_II_0033 This is use case is a generalization of Use Case 5.8.12 from TMF 513 v3.0.

4.3.12 Event generating OS succeeds in forwarding an event to the notification service

Use Case Id	UC_TMF518_RTM_0023
Use Case Name	Event generating OS succeeds in forwarding an event to the notification service
Summary	The event generating OS is able to resume event forwarding and informs the registered OSs about this. The registered OSs may synchronize and can rely on upcoming events again.
Actor(s)	Event generating OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the OS (Re)starts use case. 2. The event generating OS has informed the NMS that an event has been discarded before.
Begins When	The event generating OS succeeds to forward an event to the registered OSs.
Description	1. The event generating OS informs the registered OSs that event forwarding is resumed. There is an operation provided by the registered OSs for this purpose. If the event generating OS has sent alarms about the event loss before it shall clear those alarms. 2. The registered OSs may now start synchronization for alarms, TCA and/or protection switch states alone or for the complete configuration dependent on what it was informed about before (i.e., in the case lifecycle events were lost).
Ends When	In case of success: The registered OSs are informed that event generating OS has resumed event forwarding. In case of failure: The registered OSs are not informed that event generating OS has resumed event forwarding
Post-Conditions	In case of success: The registered OSs are informed that the event generating OS has resumed event forwarding and may have started synchronization. In case of failure: The registered OSs are not informed that the event generating OS has resumed event forwarding and may have started synchronization.
Exceptions	None
Traceability	R_TMF518_RTM_II_0034

	This is use case is a generalization of Use Case 5.8.13 from TMF 513 v3.0.

4.3.13 OS sends a heartbeat notification to the notification service

Use Case Id	UC_TMF518_RTM_0024
Use Case Name	OS sends a heartbeat notification to the notification service
Summary	An OS (referred to as the heartbeat sending OS) sends a heartbeat notification to the other OSs on the CCV if there are no other notifications to be forwarded for some time. The other OSs thus knows that heartbeat sending OS is still available even if there are no other notifications for a longer time. All OSs on the CCV are likely to play the role of “heartbeat sending OS”.
Actor(s)	Heartbeat sending OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The heartbeat sending OS has executed the OS (Re)starts use case.
Description	1. On a regular basis, the heartbeat sending OS sends out heartbeat notifications to the other OSs on the CCV. 2. The other OSs on the CCV can conclude that the heartbeat sending OS is unavailable if they doesn't receive any heartbeat notification for the OS in question.
Ends When	The heartbeat sending OS is shutdown or is in a failure mode.
Post-Conditions	The other OSs on the CCV have assumed the OS in question is no longer on the CCV and will expect to see a restart procedure if and when the OS is inserted into the CCV.
Exceptions	None
Traceability	R_TMF518_RTM_II_0035 This is use case is a generalization of Use Case 5.8.14 from TMF 513 v3.0.

4.4 Equipment Management

4.4.1 OS provisions alarm reporting on/off for equipment

Use Case Id	UC_TMF518_RTM_0025
Use Case Name	OS provisions alarm reporting on/off for equipment

Summary	The requesting OS asks that the target OS activate/deactivate all alarm reporting on an equipment.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends the request to activate/deactivate alarm reporting to the target OS.
Description	1. The requesting OS sends a request to activate/deactivate alarm reporting for a specified equipment. 2. The target OS validates the equipment reference (e.g., name). 3. The target OS enables/disables alarm reporting on the specified equipment. 4. The target OS replies with a success indication. 5. Attribute Value Change notification(s) for the specified equipment are forwarded to the notification service indicating that alarm reporting has been activated/deactivated for the specified equipment.
Ends When	In case of success: The requesting OS receives an indication of success for the requested action. In case of failure: The requesting OS receives an indication of failure for the requested action.
Post-Conditions	In case of success: Alarm monitoring is enabled/disabled on the specified equipment. [Note: If alarm monitoring is enabled, this does not necessarily mean that alarms are reported, because an applied Alarm Severity Assignment Profile may perform further filtering. If alarm monitoring is disabled, then alarm reporting is disabled even if an applied Alarm Severity Assignment Profile would allow it.] The target OS has forwarded an AVC notification if there was an attribute value change associated with the enabling/disabling of alarm monitoring on the equipment. In case of failure: None.

Exceptions	1. Invalid input: The equipment reference is invalid. 2. Communications loss. 3. Entity not found: The specified equipment object does not exist. 4. Processing failure: The requested operation could not be performed. 5. Unable to comply: Alarm reporting can not be enabled/disabled for the give equipment instance
Traceability	R_TMF518_RTM_II_0020 This is use case is a generalization of Use Case 5.9.3 from TMF 513 v3.0.

4.4.2 OS provisions alarm reporting on/off for an equipment holder

Use Case Id	UC_TMF518_RTM_0026
Use Case Name	OS provisions alarm reporting on/off for an equipment holder
Summary	The requesting OS asks that the target OS activate/deactivate alarm reporting on an equipment holder.
Actor(s)	Requesting OS
Pre-Conditions	All OSs involved in the use case have successfully executed the OS (Re)starts use case.
Begins When	The requesting OS sends the request to activate/deactivate alarm reporting to the target OS.
Description	1. The requesting OS sends the request to activate/deactivate alarm reporting on/off for a specified equipment holder. 2. The target OS validates the equipment holder reference (e.g. name). 3. The target OS enables/disables alarm reporting on the specified equipment holder. 4. The target OS replies with a success indication. 5. Attribute Value Change notification(s) for the specified equipment holder are forwarded to the notification service indicating that alarm reporting has been activated/deactivated for the specified equipment holder.
Ends When	In case of success: The requesting OS receives an indication of success for the requested action. In case of failure: The requesting OS receives an indication of failure for the

	requested action.
Post-Conditions	In case of success: Alarm monitoring is enabled/disabled on the specified equipment holder. [Note: If alarm monitoring is enabled, this does not necessarily mean that alarms are reported, because an applied Alarm Severity Assignment Profile may perform further filtering. If alarm monitoring is disabled, then alarm reporting is disabled even if an applied Alarm Severity Assignment Profile would allow it.] The target OS has forwarded an AVC notification if there was an attribute value change with the enabling/disabling of alarm monitoring on the equipment holder. In case of failure: None.
Exceptions	1. Invalid input: The equipment reference is invalid. 2. Communications loss. 3. Processing failure: The requested operation could not be performed. 4. Unable to comply: The alarm reporting can not be enabled/disabled.
Traceability	R_TMF518_RTM_II_0014 and R_TMF518_RTM_II_0015 This is use case is a generalization of Use Case 5.9.4 from TMF 513 v3.0.

4.5 Craft Related

4.5.1 Craft/Target OS creates a Protection Group

Use Case Id	UC_TMF518_RTM_0027
Use Case Name	Craft/Target OS creates a Protection Group
Summary	The Craft creates a Protection Group on a network element via the target OS (e.g., an EMS) or on the network element itself, or the target OS detects a new protection group has been created on a network element. This use case is to cover both TP and equipment protection groups.
Actor(s)	Craft or target OS
Pre-Conditions	1. All OSs involved in the use case have successfully executed the

	OS (Re)starts use case. 2. The target OS and registered OSs are connected to the notification service.
Begins When	The target OS detects that a Protection Group was created on a Managed Element.
Description	1. The target OS identifies the protection group type. If the protection group identified pertains to a 4 fiber MS SP ring (BLSR) protection, the target OS sends three separate object creation notifications (one each for the span groups and one for the combined groups). In all other cases, a single group is identified to be sent to the registered OSs. 2. The object creation notification identifies the steady state switch status of the protection group. 3. Edge point Boolean is set for this notification if any of the TPs forming the protection group is an edge point.
Ends When	The target OS sends applicable notifications to the registered OSs.
Post-Conditions	The registered OSs are aware of the existence of the line level protection.
Exceptions	None
Traceability	R_TMF518_RTM_II_0052 and R_TMF518_RTM_II_0054 This is use case is a generalization of Use Case 5.10.6 from TMF 513 v3.0.

5 Traceability Matrices

Table 5-1. Requirements – Use Cases Traceability Matrix

Requirement Id	Use Case Name	Use Case Id
R_TMF518_RTM_BR_0001		
R_TMF518_RTM_BR_0002		
R_TMF518_RTM_BR_0003		
R_TMF518_RTM_BR_0004		
R_TMF518_RTM_BR_0005		
R_TMF518_RTM_BR_0006		
R_TMF518_RTM_II_0007		
R_TMF518_RTM_II_0009		
R_TMF518_RTM_II_0010		
R_TMF518_RTM_II_0012	OS turns alarm reporting “on” for a TP	UC_TMF518_RTM_0001
R_TMF518_RTM_II_0013	OS turns alarm reporting “off” for a TP	UC_TMF518_RTM_0002
R_TMF518_RTM_II_0014	OS provisions alarm reporting on/off for an equipment holder	UC_TMF518_RTM_0026
R_TMF518_RTM_II_0015	OS provisions alarm reporting on/off for an equipment holder	UC_TMF518_RTM_0026
R_TMF518_RTM_II_0022	OS assigns an Alarm Severity Assignment Profile (ASAP) to a CTP (or some other object type that supports the assignment of ASAPs)	UC_TMF518_RTM_0003
R_TMF518_RTM_II_0023		
R_TMF518_RTM_II_0024		
R_TMF518_RTM_II_0025	OS assigns an Alarm Severity Assignment Profile (ASAP) to a CTP (or some other object type that supports the assignment of ASAPs)	UC_TMF518_RTM_0003
R_TMF518_RTM_II_0026		
R_TMF518_RTM_II_0027	OS registers to receive alarms from a target OS	UC_TMF518_RTM_0013
R_TMF518_RTM_II_0028	OS registers to receive alarms from a target OS	UC_TMF518_RTM_0013

R_TMF518_RTM_II_0029	OS registers to receive RCAIs only, raw alarms only, or both RCAIs and raw alarms from a target OS OS registers to receive alarms from a target OS	UC_TMF518_RTM_0014 UC_TMF518_RTM_0013
R_TMF518_RTM_II_0030		
R_TMF518_RTM_II_0031	Alarm generating OS Notifies Registered OSs of Alarms or Threshold Crossing Alert (TCA)s	UC_TMF518_RTM_0016
R_TMF518_RTM_II_0032		
R_TMF518_RTM_II_0033	Event generating OS discards an event that was to be sent to the notification service	UC_TMF518_RTM_0022
R_TMF518_RTM_II_0034	Event generating OS succeeds in forwarding an event to the notification service	UC_TMF518_RTM_0023
R_TMF518_RTM_II_0035	OS sends a heartbeat notification to the notification service	UC_TMF518_RTM_0024
R_TMF518_RTM_II_0036	Get Active Alarm Counts	UC_TMF518_RTM_0009
R_TMF518_RTM_II_0037	Requesting OS reconciles Unacknowledged Active Alarms for a specified list of Managed Elements Requesting OS reconciles Unacknowledged Active Alarms from a target OS Active Alarms Retrieval – File Transfer Active Alarms Retrieval – RPC Communication Style Active Alarms Retrieval – Message Communication Style	UC_TMF518_RTM_0021 UC_TMF518_RTM_0020 UC_TMF518_RTM_0012 UC_TMF518_RTM_0011 UC_TMF518_RTM_0010

R_TMF518_RTM_II_0038	Requesting OS reconciles Unacknowledged Active Alarms for a specified list of Managed Elements Requesting OS reconciles Unacknowledged Active Alarms from a target OS Active Alarms Retrieval – File Transfer Active Alarms Retrieval – RPC Communication Style Active Alarms Retrieval – Message Communication Style	UC_TMF518_RTM_0021 UC_TMF518_RTM_0020 UC_TMF518_RTM_0012 UC_TMF518_RTM_0011 UC_TMF518_RTM_0010
R_TMF518_RTM_II_0039		
R_TMF518_RTM_II_0040	Alarm Acknowledgement in the OS (other than the alarm owning OS)	UC_TMF518_RTM_0017
R_TMF518_RTM_II_0041	Alarm Unacknowledgement in the OS (other than the alarm owning OS)	UC_TMF518_RTM_0018
R_TMF518_RTM_II_0042	Alarm Acknowledgement in the alarm owning OS Alarm Acknowledgement in the OS (other than the alarm owning OS)	UC_TMF518_RTM_0019 UC_TMF518_RTM_0017
R_TMF518_RTM_II_0043	Alarm Unacknowledgement in the OS (other than the alarm owning OS)	UC_TMF518_RTM_0018
R_TMF518_RTM_II_0044		
R_TMF518_RTM_II_0045	OS registers to receive RCAIs only, raw alarms only, or both RCAIs and raw alarms from a target OS	UC_TMF518_RTM_0014
R_TMF518_RTM_II_0046	Alarm owning OS determines a more appropriate root cause than one previously reported	UC_TMF518_RTM_0015
R_TMF518_RTM_II_0047		
R_TMF518_RTM_II_0048	OS retrieves all the Protection Groups of a Managed Element	UC_TMF518_RTM_0004
R_TMF518_RTM_II_0049	OS retrieves all the Protection Groups of a Managed Element	UC_TMF518_RTM_0004
R_TMF518_RTM_II_0050	OS retrieves all the Protection Groups of a Managed Element	UC_TMF518_RTM_0004

R_TMF518_RTM_II_0051	OS retrieves all the Protection Groups of a Managed Element	UC_TMF518_RTM_0004
R_TMF518_RTM_II_0052	Craft/Target OS creates a Protection Group	UC_TMF518_RTM_0027
R_TMF518_RTM_II_0053	OS retrieves the protection switch information for Equipment, Trail and SNC Protection	UC_TMF518_RTM_0006
R_TMF518_RTM_II_0054	Craft/Target OS creates a Protection Group	UC_TMF518_RTM_0027
R_TMF518_RTM_II_0055	OS retrieves the protection switch information for Equipment, Trail and SNC Protection	UC_TMF518_RTM_0006
R_TMF518_RTM_II_0056		
R_TMF518_RTM_II_0057	OS registers to receive protection switch notifications	UC_TMF518_RTM_0007
R_TMF518_RTM_II_0058	OS invokes protection switch lockout to an SNC	UC_TMF518_RTM_0008
R_TMF518_RTM_II_0059		
R_TMF518_RTM_II_0060		
R_TMF518_RTM_II_0061	OS registers to receive protection switch notifications Protection Switch Notification for Equipment, Trail and SNC Protection	UC_TMF518_RTM_0007 UC_TMF518_RTM_0005
R_TMF518_RTM_II_0062		
R_TMF518_RTM_II_0063		
R_TMF518_RTM_II_0067		
R_TMF518_RTM_II_0068		
R_TMF518_RTM_II_0069		
R_TMF518_RTM_III_0064	Active Alarms Retrieval – File Transfer Active Alarms Retrieval – RPC Communication Style Active Alarms Retrieval – Message Communication Style	UC_TMF518_RTM_0012 UC_TMF518_RTM_0011 UC_TMF518_RTM_0010
R_TMF518_RTM_III_0065		
R_TMF518_RTM_III_0066		

Table 5-2. Use Cases – Requirements Traceability Matrix

Use Case Id	Use Case Name	Requirements
UC_TMF518_RTM_0001	OS turns alarm reporting “on” for a TP	R_TMF518_RTM_II_0012 This is use case is a generalization of Use Case 5.5.8 from TMF 513 v3.0.
UC_TMF518_RTM_0002	OS turns alarm reporting “off” for a TP	R_TMF518_RTM_II_0013 This is use case is a generalization of Use Case 5.5.9 from TMF 513 v3.0.
UC_TMF518_RTM_0003	OS assigns an Alarm Severity Assignment Profile (ASAP) to a CTP (or some other object type that supports the assignment of ASAPs)	R_TMF518_RTM_II_0022 and R_TMF518_RTM_II_0025 This is use case is a generalization of Use Case 5.5.21 from TMF 513 v3.0.
UC_TMF518_RTM_0004	OS retrieves all the Protection Groups of a Managed Element	R_TMF518_RTM_II_0048, R_TMF518_RTM_II_0049, R_TMF518_RTM_II_0050 and R_TMF518_RTM_II_0051 This is use case is a generalization of Use Case 5.7.1 from TMF 513 v3.0.
UC_TMF518_RTM_0005	Protection Switch Notification for Equipment, Trail and SNC Protection	R_TMF518_RTM_II_0061 This is use case is a generalization of Use Case 5.7.2 from TMF 513 v3.0.
UC_TMF518_RTM_0006	OS retrieves the protection switch information for Equipment, Trail and SNC Protection	R_TMF518_RTM_II_0053 and R_TMF518_RTM_II_0055 This is use case is a generalization of Use Case 5.7.3 from TMF 513 v3.0.
UC_TMF518_RTM_0007	OS registers to receive protection switch notifications	R_TMF518_RTM_II_0057 and R_TMF518_RTM_II_0061 This is use case is a generalization of Use Case 5.7.4 from TMF 513 v3.0.
UC_TMF518_RTM_0008	OS invokes protection switch lockout to an SNC	R_TMF518_RTM_II_0058 This is use case is a generalization of Use Case 5.7.5 from TMF 513 v3.0.

UC_TMF518_RTM_0009	Get Active Alarm Counts	R_TMF518_RTM_II_0036 This is use case is based on UC_MTOSI_21 from TMF 517 v1.2.
UC_TMF518_RTM_0010	Active Alarms Retrieval – Message Communication Style	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_22 from TMF 517 v1.2.
UC_TMF518_RTM_0011	Active Alarms Retrieval – RPC Communication Style	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_23 from TMF 517 v1.2.
UC_TMF518_RTM_0012	Active Alarms Retrieval – File Transfer	R_TMF518_RTM_II_0037, R_TMF518_RTM_II_0038 and R_TMF518_RTM_III_0064 This is use case is based on UC_MTOSI_24 from TMF 517 v1.2.
UC_TMF518_RTM_0013	OS registers to receive alarms from a target OS	R_TMF518_RTM_II_0027, R_TMF518_RTM_II_0028 and R_TMF518_RTM_II_0029 This is use case is a generalization of Use Case 5.8.3 from TMF 513 v3.0.
UC_TMF518_RTM_0014	OS registers to receive RCAIs only, raw alarms only, or both RCAIs and raw alarms from a target OS	R_TMF518_RTM_II_0029 and R_TMF518_RTM_II_0045 This is use case is a generalization of Use Case 5.8.4 from TMF 513 v3.0.
UC_TMF518_RTM_0015	Alarm owning OS determines a more appropriate root cause than one previously reported	R_TMF518_RTM_II_0046 This is use case is a generalization of Use Case 5.8.5 from TMF 513 v3.0.
UC_TMF518_RTM_0016	Alarm generating OS Notifies Registered OSs of Alarms or Threshold Crossing Alert (TCA)s	R_TMF518_RTM_II_0031 This is use case is a generalization of Use Case 5.8.6 from TMF 513 v3.0.

UC_TMF518_RTM_0017	Alarm Acknowledgement in the OS (other than the alarm owning OS)	R_TMF518_RTM_II_0040 and R_TMF518_RTM_II_0042 This is use case is a generalization of Use Case 5.8.7 from TMF 513 v3.0.
UC_TMF518_RTM_0018	Alarm Unacknowledgement in the OS (other than the alarm owning OS)	R_TMF518_RTM_II_0041 and R_TMF518_RTM_II_0043 This is use case is a generalization of Use Case 5.8.8 from TMF 513 v3.0.
UC_TMF518_RTM_0019	Alarm Acknowledgement in the alarm owning OS	R_TMF518_RTM_II_0042 This is use case is a generalization of Use Case 5.8.9 from TMF 513 v3.0.
UC_TMF518_RTM_0020	Requesting OS reconciles Unacknowledged Active Alarms from a target OS	R_TMF518_RTM_II_0037 and R_TMF518_RTM_II_0038 This is use case is a generalization of Use Case 5.8.10 from TMF 513 v3.0.
UC_TMF518_RTM_0021	Requesting OS reconciles Unacknowledged Active Alarms for a specified list of Managed Elements	R_TMF518_RTM_II_0037 and R_TMF518_RTM_II_0038 This is use case is a generalization of Use Case 5.8.11 from TMF 513 v3.0.
UC_TMF518_RTM_0022	Event generating OS discards an event that was to be sent to the notification service	R_TMF518_RTM_II_0033 This is use case is a generalization of Use Case 5.8.12 from TMF 513 v3.0.
UC_TMF518_RTM_0023	Event generating OS succeeds in forwarding an event to the notification service	R_TMF518_RTM_II_0034 This is use case is a generalization of Use Case 5.8.13 from TMF 513 v3.0.
UC_TMF518_RTM_0024	OS sends a heartbeat notification to the notification service	R_TMF518_RTM_II_0035 This is use case is a generalization of Use Case 5.8.14 from TMF 513 v3.0.
UC_TMF518_RTM_0025	OS provisions alarm reporting on/off for equipment	R_TMF518_RTM_II_0020 This is use case is a generalization of Use Case 5.9.3 from TMF 513 v3.0.

UC_TMF518_RTM_0026	OS provisions alarm reporting on/off for an equipment holder	R_TMF518_RTM_II_0014 and R_TMF518_RTM_II_0015 This is use case is a generalization of Use Case 5.9.4 from TMF 513 v3.0.
UC_TMF518_RTM_0027	Craft/Target OS creates a Protection Group	R_TMF518_RTM_II_0052 and R_TMF518_RTM_II_0054 This is use case is a generalization of Use Case 5.10.6 from TMF 513 v3.0.

6 Future Directions

The following possible future work items have been identified.

1. The requirements and use cases in this document may be extended to cover service fault management.
2. At some future point, it is desirable to harmonize the work in this document with the OSS/J resource fault management API.

7 References

7.1 References

[1] TMF518_FMW, Framework DDP-BA.

[2] TMF518_NRB, Network Resource Basic DDP-BA

[3] TMF518_NRA, Network Resource Assurance DDP-BA

[4] TMF513, Multi-Technology Network Management (MTNM) Business Agreement, Version 3.1, March 2007

[5] TMF517, Multi-Technology Operations System Interface (MTOSI) Business Agreement, Version 1.2, December 2006.

[6] TMF612_RTM, Resource Trouble Management IA.

[7] TMF864_RTM, Resource Trouble Management IIS

[8] SD0-1, Dictionary

[9] SD1-20, Maintenance Commands

[10] SD1-33, Probable Causes

[11] ITU-T Recommendation G.841, Types and characteristics of SDH network protection architectures, October 1998.

7.2 Source or Use

The various sources for the requirements in this document are listed in the “Source” field of each requirement.

The following documents make use of this document:

- TMF612_RTM, *Resource Trouble Management IA*, Version 1.0, date TBD.
- TMF864_RTM, *Resource Trouble Management IIS*, Version 1.0, date TBD.

7.3 IPR Releases and Patent Disclosure

There are no known IPR claims on the material in this document. As per the TM Forum bylaws, any TM Forum member company that has IPR claims on this or any TM Forum specification needs to make the claims known to the TM Forum membership immediately.

8 Administrative Appendix

This Appendix provides additional background material about the TM Forum and this document.

8.1 About this document

This document has been generated from the SD0-3_Template_BA.dot Word template.

8.2 Use and Extension of a TM Forum Business Agreement

This document defines the business problem and requirement model resource trouble management. The Business Agreement is used to gain consensus on the business requirements for exchanging information among processes and systems in order to solve a specific business problem. The Business Agreement should feed the development of Information Agreement(s), which is a technology-neutral model of one or more interfaces. While the Business Agreement contains sufficient information to be a "stand alone" document, it is better read together with the Information Agreement document TMF612_RTM when the Information Agreement is available. Reviewing the two documents together helps in gaining a full understanding of how the technology neutral information model solution is defined for this requirement model. An initial Business Agreement may only deal with a subset of the requirements. It is acceptable for subsequent issues of the document to add additional requirements not addressed by earlier releases of the BA. Business Agreements are the basis for requirement traceability for information models.

It is expected that this document will be used:

- As the foundation for a TM Forum Information Agreement(s)
- To facilitate requirement agreement between Service Providers and vendors
- As input to a service Provider's Request for Information / Request for Proposal (RFI/RFP—RFX)
- As input for vendors developing COTS products
- As a source of requirements for other bodies working in this area

8.3 Document History

Version	Date Modified	Description of changes
1.0	October 2007	This is the first version of the document and as such, there are no changes to report.
1.1	May 2008	Made updates based on member evaluation of MTOSI 2.0 Bas.
1.2	September 2011	Updated sections 1.1 and 2. Replaced mTOP by MTNM / MTOSI everywhere in the document

8.4 Company Contact Details

Document Contact
Stephen Fratini Telcordia Technologies *Email:* sfratini@telcordia.com *Phone:* +1 732 699 2226

8.5 Acknowledgments

This document was prepared by the members of the TM Forum MTNM / MTOSI RM team.

- Stephen Fratini, Telcordia Technologies, MTNM / MTOSI Program Director and document editor
- Keith Dorking, Ciena, MTNM / MTOSI Resource Management team leader
- Michel Besson, Amdocs, MTOSI Product Manager

Additional input was provided by the following people:

- Marc Flauw, HP