
HP OSS Fault Analytics & Statistics

Version 1.1



Customization Guide

Edition: 1.0

For Linux, RHEL 6.5

October 2015

Legal notices

Warranty

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

License requirement and U.S. Government legend

Confidential computer software. Valid license from HP required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright notices

© Copyright 2015 Hewlett-Packard Development Company, L.P.

Trademark notices

Adobe®, Acrobat® and PostScript® are trademarks of Adobe Systems Incorporated.

Java™ is a trademark of Oracle and/or its affiliates.

Microsoft®, Internet Explorer, Windows®, Windows Server®, and Windows NT® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

HP Vertica™, the HP Vertica Analytics Platform™ are trademarks of Hewlett-Packard

Firefox® is a registered trademark of the Mozilla Foundation.

Google Chrome® is a trademark of Google Inc.

UNIX® is a registered trademark of The Open Group.

Red Hat® is a registered trademark of the Red Hat Company.

Linux® is a registered trademark of Linus Torvalds in the U.S. and other countries.

JBoss®, Wildfly and PicketLink are registered trademarks of RedHat Inc.

Contents

Preface	6
Chapter 1.....	8
Product overview	8
1.1 HP OSS Fault Analytics & Statistics.....	8
1.2 OSS Fault Analytics & Statistics architecture	9
Chapter 2.....	10
Datamart	10
2.1 FAS datamart	10
2.1.1 Dimension tables.....	11
2.1.2 FCT_FAULT table	12
2.1.3 Summarization tables	14
2.1.4 FCT_FAULT_COMMENT	20
Chapter 3.....	21
Metadata.....	21
3.1 Metadata concept	21
3.2 FAS metadata description.....	21
Chapter 4.....	26
FAS default reports	26
Chapter 5.....	27
FAS Customization	27
5.1 Mapping rules from TeMIP alarms to FAS datamart	27
5.1.1 Specific Problems configuration	27
5.2 FAS solution enrichment	28
5.2.1 User Defined Attributes example.....	28
5.2.2 Other customizations	31

Figures

Figure 1 – OSS Fault Analytics & Statistics architecture.....	9
Figure 2 – FCT_FAULT table with dimensions	10

Tables

Table 1 - Software versions6

Preface

This guide describes how to integrate and customize the HP Fault Analytics & Statistics software component.

Software component name: HP OSS Fault Analytics & Statistics

Software component version: 1.1

Software kit version: V1.1

Intended audience

- This customization guide is for anyone who is responsible for integrating or customizing a HP OSS Fault Analytics & Statistics solution.

The readers are assumed to understand Linux shell concepts, database concepts and xml syntax.

Software versions

The terms Unix and Linux are used as a generic reference to the operating system, unless otherwise specified.

The software versions referred to in this document are as follows:

Product Version	Supported Operating systems
<i>HP OSS Fault Analytics & Statistics 1.1.0</i>	• Red Hat Enterprise Linux Server release RHEL 6.5
<i>HP OSS Analytics Foundation 1.1.1</i>	• Red Hat Enterprise Linux Server release RHEL 6.5
<i>HP Vertica Version 7.1</i>	• Red Hat Enterprise Linux Server release RHEL 6.5
<i>HP UMB Server Version 1.0</i>	• Red Hat Enterprise Linux Server release RHEL 6.5
<i>HP Unified OSS Console 2.1</i>	• Red Hat Enterprise Linux Server release RHEL 6.5
<i>HP TeMIP 6.2</i>	• Red Hat Enterprise Linux Server release RHEL 6.5

Table 1 - Software versions

Typographical conventions

Courier font:

- Source code and examples of file contents
- Commands that you enter on the screen
- Path names
- Keyboard key names

Italic text:

- File names, programs, and parameters
- The names of other documents referenced in this manual

Bold text:

- To introduce new terms and to emphasize important words

Associated documents

- HP OSS Fault Analytics & Statistics Installation and Administration Guide
- HP OSS Fault Analytics & Statistics User Guide
- HP OSS Analytics Foundation Installation Configuration and Administration guide
- HP OSS Analytics Foundation Integration guide
- HP UOC User Guide

Support

Visit the HP Software Support Online website at <https://softwaresupport.hp.com/> for contact information, and for details about HP software products, services, and support.

The software support area of the website includes the following:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

Product overview

1.1 HP OSS Fault Analytics & Statistics

HP OSS Fault Analytics and Statistics (FAS) is a software product that enables telecommunications service providers with the capabilities to collect and persist fault information from fault and surveillance systems, transform the data as needed and deliver actionable insight to operations staff to operate and manage their network. The actionable insight is inferred using a host of statistical and analytical techniques.

OSS FAS is positioned as an independent product, working with fault information consolidated in HP TeMIP, as well as any other surveillance system from an independent software vendor.

OSS FAS is based on HP Vertica, complemented by a mediation layer allowing for collection of fault information in real time.

A brief summary of the key features and benefits part of this FAS V1.1 release:

- Transformation of vast amounts of alarm data received from HP TeMIP into meaningful information
- Use of the HP Vertica database, optimized for data warehousing, data analytics and data reporting
- Optional activation of default summarizations batch jobs in order to populate new tables containing information data about alarms (aggregation based on different time granularities and different dimensions)
- Optional default FAS reports (built with Unified OSS Console) based on those summarized tables

1.2 OSS Fault Analytics & Statistics architecture

HP OSS Fault Analytics and Statistics (FAS) solution is composed of three software components:

- TeMIP Analytics
- OSS Analytics foundation
- OSS FAS Core

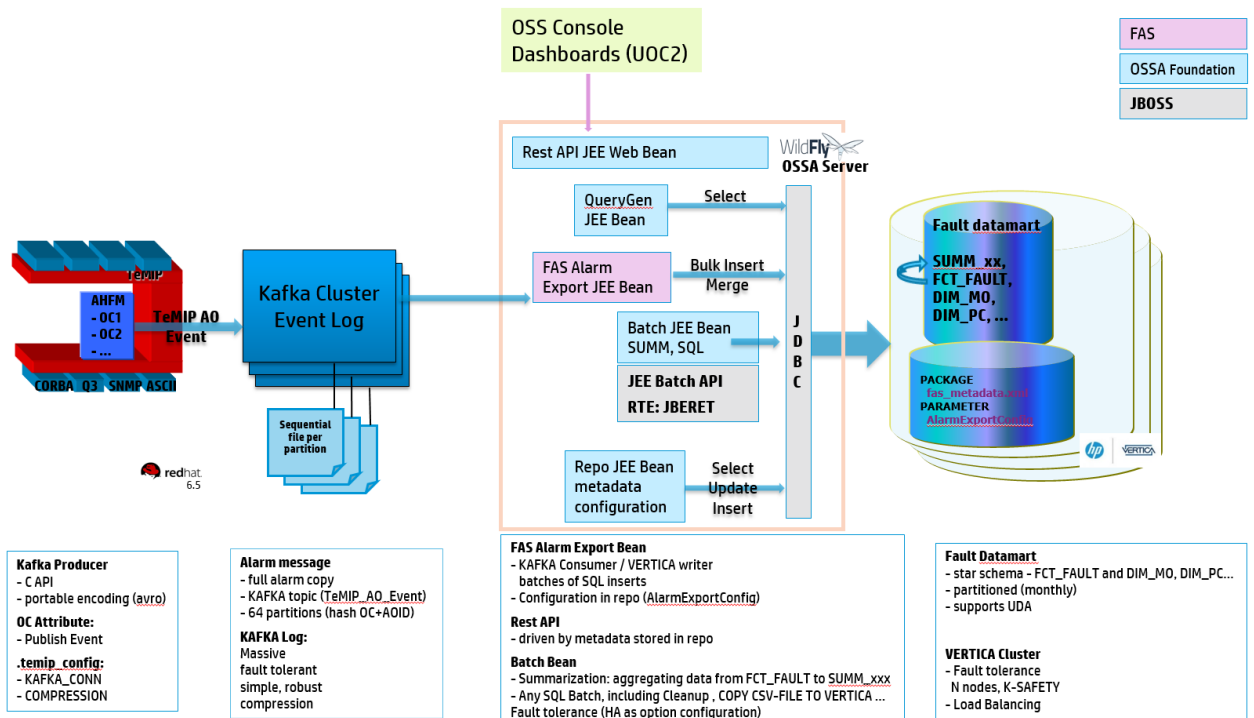


Figure 1 – OSS Fault Analytics & Statistics architecture

In the following chapters, the FAS datamart, metadata and default reports are presented, before demonstrating some possible customization of FAS.

Datamart

Before explaining how to customize the HP OSS Fault Analytics & Statistics solution, we must introduce with some concepts. This document starts with **Datamart**.

A Datamart is a database containing historical and aggregated data stored in a multidimensional schema, generally implemented as a 'star' schema, where **facts** are surrounded by **dimensions** tables. Thus, this database is optimized for reporting and analysis; facts can be analyzed from different axis, the dimensions.

2.1 FAS datamart

The main fact table in the FAS datamart is the **FCT_FAULT** table. The alarms coming from TeMIP via the Kafka bus are stored over time within this fact table. **FCT_FAULT** table will be continually updated overtime, adding new row for each new alarm.

There are 11 dimension tables associated with this fact table. They represent the different types of analysis that can be done on the alarms. Obviously, the dimension tables are also continuously updated overtime depending on the alarm received.

See the diagram below showing the 'star': with FCT_FAULT table in the middle surrounded by the dimension tables.

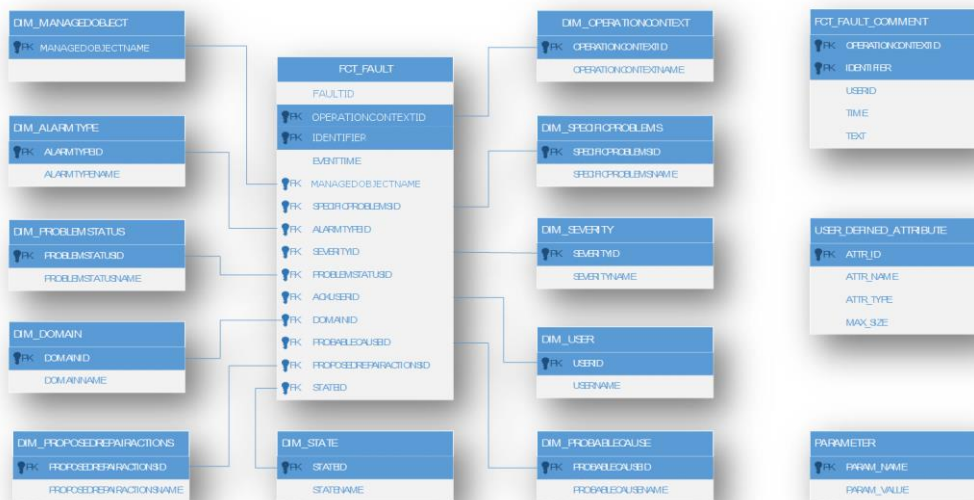


Figure 2 – FCT_FAULT table with dimensions

2.1.1 Dimension tables

Here are described the FAS datamart dimensions tables:

DIM_ALARMTYPE table

COLUMN NAME	COLUMN TYPE
ALARMTYPEID	INTEGER
ALARMTYPENAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_DOMAIN table

COLUMN NAME	COLUMN TYPE
DOMAINID	INTEGER
DOMAINNAME	VARCHAR(700)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_MANAGEDOBJECT table

COLUMN NAME	COLUMN TYPE
MANAGEDOBJECTNAME	VARCHAR(700)
CLASSNAME	VARCHAR(700)
GLOBALMONAME	VARCHAR(700)
GLOBALCLASSNAME	VARCHAR(700)
TNSNAME	GLOBALCLASSNAME
UPDATE_TIMESTAMP	TIMESTAMP

DIM_OPERATIONCONTEXT table

COLUMN NAME	COLUMN TYPE
OPERATIONCONTEXTID	INTEGER
OPERATIONCONTEXTNAME	VARCHAR(700)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_PROBABLECAUSE table

COLUMN NAME	COLUMN TYPE
PROBABLECAUSEID	INTEGER
PROBABLECAUSENAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_PROBLEMSTATUS table

COLUMN NAME	COLUMN TYPE
PROBLEMSTATUSID	INTEGER
PROBLEMSTATUSNAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_PROPOSEDREPAIRACTIONS table

COLUMN NAME	COLUMN TYPE
PROPOSEDREPAIRACTIONSID	INTEGER
PROPOSEDREPAIRACTIONS	VARCHAR(65000)
PROPOSEDREPAIRACTIONSNAME	VARCHAR(500)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_SEVERITY table

COLUMN NAME	COLUMN TYPE
SEVERITYID	INTEGER
SEVERITYNAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_SPECIFICPROBLEMS table

COLUMN NAME	COLUMN TYPE
SPECIFICPROBLEMSID	INTEGER
SPECIFICPROBLEMS	VARCHAR(65000)
SPECIFICPROBLEMSNAME	VARCHAR(500)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_STATE table

COLUMN NAME	COLUMN TYPE
STATEID	INTEGER
STATENAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

DIM_USER table

COLUMN NAME	COLUMN TYPE
USERID	INTEGER
USERNAME	VARCHAR(100)
UPDATE_TIMESTAMP	TIMESTAMP

2.1.2 FCT_FAULT table

Here is the description of the *FCT_FAULT* table:

COLUMN NAME	COLUMN TYPE	DESCRIPTION
FAULTID	INTEGER	An identifier of FCT_FAULT. Obsolete. Do not use it.
ALARMTYPEID	INTEGER	Alarm Type Identifier (Foreign Key)
DOMAINID	INTEGER	Domain Identifier (Foreign Key)
OPERATIONCONTEXTID	INTEGER	Operation Context Identifier (Foreign Key)
IDENTIFIER	INTEGER	Uniquely identifies each Alarm Object belonging to a particular Operation Context
MANAGEDOBJECTNAME	VARCHAR	Managed Object Name (Foreign Key)
PROBABLECAUSEID	INTEGER	Probable Cause Identifier (Foreign Key)
SPECIFICPROBLEMSID	INTEGER	Specific Problems Identifier (Foreign Key)
PROPOSEDREPAIRACTIONSID	INTEGER	Proposed Repair Actions Identifier (Foreign Key)
SEVERITYID	INTEGER	Severity Identifier (Foreign Key)
STATEID	INTEGER	State Identifier (Foreign Key)
PROBLEMSTATUSID	INTEGER	Problem Status Identifier (Foreign Key)
ACKUSERID	INTEGER	Identifier of the user who acknowledged the alarm (Foreign Key)
CLOSEUSERID	INTEGER	Identifier of the user who closed the alarm (Foreign Key)
HANDLEUSERID	INTEGER	Identifier of the user who handled the alarm (Foreign Key)

COLUMN NAME	COLUMN TYPE	DESCRIPTION
RELEASEUSERID	INTEGER	Identifier of the user who released the alarm (Foreign Key)
TERMUSERID	INTEGER	Identifier of the user who terminated the alarm (Foreign Key)
ORIGINALEVENTTIME	TIMESTAMP	Time at which the original Alarm Object was generated
EVENTTIME	TIMESTAMP	If there is no associated similar alarm, it is the time at which the alarm occurred. If there are associated similar alarms, it is the time of the most recent similar alarm.
CREATIONTIMESTAMP	TIMESTAMP	Time of the creation of the alarm in TeMIP
ALARMORIGIN	INTEGER	Defines the origin of this Alarm Object as: an incoming alarm – <i>IncomingAlarm</i> or the result of a directive - <i>Create, Demote</i> or <i>Promote</i> Directive.
ESCALATEDALARMFLAG	BOOLEAN	Indicates if this alarm has been escalated or not
CLEARFLAG	BOOLEAN	Indicates if this alarm has been cleared or not
OUTAGEFLAG	BOOLEAN	Specifies if the alarm is from out of service equipment
ACKTIMESTAMP	TIMESTAMP	Time of acknowledgement of the alarm
CLEARTIMESTAMP	TIMESTAMP	Time of clearance of the alarm
CLOSETIMESTAMP	TIMESTAMP	Time of closure of the alarm
HANDLETIMESTAMP	TIMESTAMP	Time at which the alarm was handled
RELEASETIMESTAMP	TIMESTAMP	The Date of Release of the Alarm
TERMTIMESTAMP	TIMESTAMP	The Date of Termination of the Alarm
NOTIFICATIONIDENTIFIER	INTEGER	Provides unique identification for all alarms concerning a particular Managed Object. The identifier can be used for correlation purposes
ADDITIONALTEXT	VARCHAR (65000)	A free form text description of the problem reported
ALARMCLASS	INTEGER	Indicates if it is an original alarm (value = 0) or a similar alarm (value = 1)
ALARMOBJECTOPERATORNOTE	VARCHAR (128)	A free form text that allows the operator to write comments concerning the alarm
BACKEDUPSTATUS	BOOLEAN	A parameter that specifies whether the Managed Object is backed up by another entity or not
BACKUPOBJECT	VARCHAR (200)	Specifies the entity that backs up the Managed Object concerned by the alarm
ORIGINALSEVERITY	VARCHAR (13)	Defines the original severity of an Alarm Object
PARENTALARMOBJECT	INTEGER	The name of the parent Alarm Object of this Alarm Object
RULENAME	VARCHAR (100)	Specifies the Alarm FM name for the rule that caused the alarm to fire
SECURITYALARMDETECTOR	VARCHAR (300)	Defines the Object Detector of a security alarm
TRENDINDICATION	VARCHAR (10)	Indicates whether a problem is becoming more or less severe
USERIDENTIFIER	INTEGER	A free form identifier defining (and defined by) the user that created the Alarm Object
USERTEXT	VARCHAR (128)	A free form text string defined by the user that created the Alarm Object
AGENTALARMDIDENTIFIER	VARCHAR (65000)	Agent Alarm Identifier is used with the Agent Entity attribute to uniquely identify a 3GPP alarm in the scope of a TeMIP Platform

COLUMN NAME	COLUMN TYPE	DESCRIPTION
AGENTENTITY	VARCHAR (65000)	Agent Entity is used with the Agent Alarm Identifier attribute to uniquely identify a 3GPP alarm in the scope of a TeMIP Platform
ACKFLAG	BOOLEAN	Indicates if this alarm has been acknowledged or not
CLOSEFLAG	BOOLEAN	Indicates if this alarm has been closed or not
HANDLEFLAG	BOOLEAN	Indicates if this alarm has been handled or not
TERMFLAG	BOOLEAN	Indicates if this alarm has been terminated or not
PROBLEMOCCURRENCES	INTEGER	The total number of alarm occurrences received for the same problem
CLEAROCCURRENCES	INTEGER	Number of cleared similar alarms
CRITICALOCCURRENCES	INTEGER	Number of critical similar alarms
INDETERMINATEOCCURRENCES	INTEGER	Number of indeterminate similar alarms
MAJOROCCURRENCES	INTEGER	Number of major similar alarms
MINOROCCURRENCES	INTEGER	Number of minor similar alarms
WARNINGOCCURRENCES	INTEGER	Number of minor warning alarms
SATOTAL	INTEGER	Total number of similar alarms for this Alarm Object
ACKDURATION	INTEGER	Time needed to acknowledge the alarm
CLEARDURATION	INTEGER	Time needed to clear the alarm
CLOSEDURATION	INTEGER	Time needed to close the alarm
HANLEDURATION	INTEGER	Time needed to handle the alarm
TERMDURATION	INTEGER	Time needed to terminate the alarm
COLLECTIONDURATION	INTEGER	Time needed to collect the alarm
INFALTDURATION	INTEGER	Difference between the close (or clear) timestamp and the original event time
INMGMTDURATION	INTEGER	Difference between the close (or clear) timestamp and the creation timestamp
HANDLEDBY	VARCHAR (65000)	Specifies the list of Trouble Tickets that currently handle this alarm
CLOSEDBY	VARCHAR (65000)	Specifies the list of Trouble Tickets for which a CLOSE directive has been received for this alarm
PARENTS	VARCHAR (65000)	Specifies the Parent Alarm identifier string of the Group to which the Alarm belongs. Can have multiple Parent Alarms identifier separated by " , " .
CHILDREN	VARCHAR (65000)	Specifies the Children Alarm identifier string of the Group to which this Alarm belongs. Can have multiple Children Alarms Identifier separated by " , " .
LASTMODIFICATIONTIMESTAMP	TIMESTAMP	Latest time at which the Alarm Object was modified
UPDATE_TIMESTAMP	TIMESTAMP	Time at which this record has been persisted

The *FCT_FAULT* primary key is: *OPERATIONCONTEXTID, IDENTIFIER*.

2.1.3 Summarization tables

As part of the Fault Analytics & Statistics product there is a mechanism that regularly fills other Datamart facts tables: this is the **Summarization**.

Summarization is a process that regularly fills **summarization tables** with aggregated values from fact table.

The aggregations are done on different dimensions and on different time granularities (hourly, daily, weekly, monthly).

FAS comes with 11 default summarizations divided on 3 categories:

Category of summarizations	Time aggregation	Usage of summarized data
Management	hourly, daily, weekly, monthly	Analysis of the network management activity Multi dimensional analysis of alarm indicators per <i>Operation Contexts/Users</i>
Network	hourly, daily, weekly, monthly	Analysis of the global activity of the network in term of fault Multi dimensional analysis of alarm indicators per <i>Operation Contexts/Domains/Managed Objects</i>
Equipment	daily, weekly, monthly	Analysis of the faults in term of alarm type Multi dimensional analysis of alarm indicators per <i>Managed Objects /Alarmtype/SpecificProblem/ProblemStatus/State/Severity</i>

11 default summarization tables are thus populated by FAS:

SUMM_HOURLY_MANAGEMENT,
SUMM_DAILY_MANAGEMENT,
SUMM_WEEKLY_MANAGEMENT,
SUMM_MONTHLY_MANAGEMENT,

SUMM_HOURLY_NETWORK,
SUMM_DAILY_NETWORK,
SUMM_WEEKLY_NETWORK,
SUMM_MONTHLY_NETWORK,

SUMM_DAILY_EQUIPMENT,
SUMM_WEEKLY_EQUIPMENT,
SUMM_MONTHLY_EQUIPMENT.

Those summarized tables are also facts tables and thus are also associated to dimension tables (same ones than previously).

The benefit of those tables is that they contains pre-calculated aggregated fact values which can be requested directly for statistical analysis.

For optimization purpose, FAS reports are relying on those tables.

Here are the descriptions of three of those summarized tables:

SUMM_HOURLY_MANAGEMENT, SUMM_HOURLY_NETWORK and
SUMM_DAILY_EQUIPMENT.

Note that the descriptions for the other time granularities in the same categories are equivalent to the ones below.

Here is the description of the ***SUMM_HOURLY_MANAGEMENT*** table:

COLUMN NAME	COLUMN TYPE	DESCRIPTION
TIME	TIMESTAMP	Timestamp of the Hour for which the aggregation occurred
OPERATIONCONTEXTID	INTEGER	Operation Context Identifier (Foreign Key) for the aggregated alarms during this hour

COLUMN NAME	COLUMN TYPE	DESCRIPTION
PROBLEMSTATUSID	INTEGER	Problem Status Identifier (Foreign Key) for the aggregated alarms during this hour
ACKUSERID	INTEGER	Identifier (Foreign Key) of the user who acknowledged the alarms aggregated during this hour
CLOSEUSERID	INTEGER	Identifier(Foreign Key) of the user who closed the alarms aggregated during this hour
HANDLEUSERID	INTEGER	Identifier (Foreign Key) of the user who handled the alarms aggregated during this hour
RELEASEUSERID	INTEGER	Identifier (Foreign Key) of the user who released the alarms aggregated during this hour
TERMUSERID	INTEGER	Identifier (Foreign Key) of the user who terminated the alarms aggregated during this hour
SUMMARIZED_COUNT	INTEGER	Number of FCT_FAULT rows aggregated for this hour
ALARM_OBJECTS_COUNTER_SUM	INTEGER	Number of alarm objects during this hour
SIMILAR_ALARM_COUNTER_SUM	INTEGER	Number of similar alarms during this hour
ALARM_COUNTER_SUM	INTEGER	Number of alarms (AO and similar) during this hour
CRITICAL_OCCURENCES_SUM	INTEGER	Number of critical occurrences during this hour
MAJOR_OCCURENCES_SUM	INTEGER	Number of major occurrences during this hour
MINOR_OCCURENCES_SUM	INTEGER	Number of minor occurrences during this hour
WARNING_OCCURENCES_SUM	INTEGER	Number of warning occurrences during this hour
INDETERMINATE_OCCURENCES_SUM	INTEGER	Number of indeterminate occurrences during this hour
CLEAR_OCCURENCES_SUM	INTEGER	Number of clear occurrences during this hour
CLEAR_COUNTER_SUM	INTEGER	Number of clearance during this hour
ESCALATED_COUNTER_SUM	INTEGER	Number of escalated alarms during this hour
HANDLED_COUNTER_SUM	INTEGER	Number of handled alarms during this hour
CLOSED_COUNTER_SUM	INTEGER	Number of closed alarms during this hour
ACK_COUNTER_SUM	INTEGER	Number of acknowledged alarms during this hour
TERMINATED_FLAG_SUM	INTEGER	Number of terminated alarms during this hour
ACK_BY_COUNT	INTEGER	Number of user who acknowledged alarms during this hour
CLOSED_BY_COUNT	INTEGER	Number of user who closed alarms during this hour
TERMINATED_BY_COUNT	INTEGER	Number of user who terminated alarms during this hour
RELEASED_BY_COUNT	INTEGER	Number of user who released alarms during this hour
HANDLED_BY_COUNT	INTEGER	Number of user who handled alarms during this hour
CLEAR_DURATION_SUM	INTEGER	Total clear duration for alarms of this hour
HANDLE_DURATION_SUM	INTEGER	Total handle duration for alarms of this hour
CLOSE_DURATION_SUM	INTEGER	Total close duration for alarms of this hour
ACK_DURATION_SUM	INTEGER	Total acknowledgment duration for alarms of this hour
TERM_DURATION_SUM	INTEGER	Total termination duration for alarms of this hour

COLUMN NAME	COLUMN TYPE	DESCRIPTION
COLLECTION_DURATION_SUM	INTEGER	Total collection duration for alarms of this hour
IN_FAULT_DURATION_SUM	INTEGER	Total in fault duration for alarms of this hour
IN_MANAGEMENT_DURATION_SUM	INTEGER	Total in management duration for alarms of this hour
COLLECTION_COUNTER_SUM	INTEGER	Total counter of collection duration greater than 0 for alarms of this hour
IN_FAULT_COUNTER_SUM	INTEGER	Total counter of in fault duration greater than 0 for alarms of this hour
IN_MANAGEMENT_COUNTER_SUM	INTEGER	Total counter of in management duration greater than 0 for alarms of this hour
CLEAR_DURATION_AVG	FLOAT	Average of clear duration for alarms of this hour
HANDLE_DURATION_AVG	FLOAT	Average of handle duration for alarms of this hour
CLOSE_DURATION_AVG	FLOAT	Average of close duration for alarms of this hour
ACK_DURATION_AVG	FLOAT	Average of acknowledgment duration for alarms of this hour
TERM_DURATION_AVG	FLOAT	Average of termination duration for alarms of this hour
COLLECTION_DURATION_AVG	FLOAT	Average of collection duration for alarms of this hour
IN_FAULT_DURATION_AVG	FLOAT	Average of in fault duration for alarms of this hour
IN_MANAGEMENT_DURATION_AVG	FLOAT	Average of in management duration for alarms of this hour
UPDATE_TIMESTAMP	TIMESTAMP	Time at which this record has been persisted

Here is the description of the ***SUMM_HOURLY_NETWORK*** table:

COLUMN NAME	COLUMN TYPE	DESCRIPTION
TIME	TIMESTAMP	Timestamp of the Hour for which the aggregation occurred
DOMAINID	INTEGER	Domain Identifier (Foreign Key) for the alarms aggregated during this hour
OUTAGEFLAG	BOOLEAN	Specifies if the alarms aggregated on this hour are from out of service equipment or not
OPERATIONCONTEXTID	INTEGER	Operation Context Identifier (Foreign Key) for the alarms aggregated during this hour
MANAGEDOBJECTNAME	VARCHAR	Managed Object Name (Foreign Key) for the alarms aggregated during this hour
SUMMARIZED_COUNT	INTEGER	Number of FCT_FAULT rows aggregated for this hour
ALARM_OBJECTS_COUNTER_SUM	INTEGER	Number of alarm objects during this hour
SIMILAR_ALARM_COUNTER_SUM	INTEGER	Number of similar alarms during this hour
ALARM_COUNTER_SUM	INTEGER	Number of alarms (AO and similar) during this hour
CRITICAL_OCCURENCES_SUM	INTEGER	Number of critical occurrences during this hour
MAJOR_OCCURENCES_SUM	INTEGER	Number of major occurrences during this hour
MINOR_OCCURENCES_SUM	INTEGER	Number of minor occurrences during this hour
WARNING_OCCURENCES_SUM	INTEGER	Number of warning occurrences during this hour

COLUMN NAME	COLUMN TYPE	DESCRIPTION
INDETERMINATE_OCCURENCES_SUM	INTEGER	Number of indeterminate occurrences during this hour
CLEAR_OCCURENCES_SUM	INTEGER	Number of clear occurrences during this hour
CLEAR_COUNTER_SUM	INTEGER	Number of clearance during this hour
ESCALATED_COUNTER_SUM	INTEGER	Number of escalated alarms during this hour
HANDLED_COUNTER_SUM	INTEGER	Number of handled alarms during this hour
CLOSED_COUNTER_SUM	INTEGER	Number of closed alarms during this hour
ACK_COUNTER_SUM	INTEGER	Number of acknowledged alarms during this hour
TERMINATED_FLAG_SUM	INTEGER	Number of terminated alarms during this hour
ACK_BY_COUNT	INTEGER	Number of user who acknowledged alarms during this hour
CLOSED_BY_COUNT	INTEGER	Number of user who closed alarms during this hour
TERMINATED_BY_COUNT	INTEGER	Number of user who terminated alarms during this hour
RELEASED_BY_COUNT	INTEGER	Number of user who released alarms during this hour
HANDLED_BY_COUNT	INTEGER	Number of user who handled alarms during this hour
CLEAR_DURATION_SUM	INTEGER	Total clear duration for alarms of this hour
HANDLE_DURATION_SUM	INTEGER	Total handle duration for alarms of this hour
CLOSE_DURATION_SUM	INTEGER	Total close duration for alarms of this hour
ACK_DURATION_SUM	INTEGER	Total acknowledgment duration for alarms of this hour
TERM_DURATION_SUM	INTEGER	Total termination duration for alarms of this hour
COLLECTION_DURATION_SUM	INTEGER	Total collection duration for alarms of this hour
IN_FAULT_DURATION_SUM	INTEGER	Total in fault duration for alarms of this hour
IN_MANAGEMENT_DURATION_SUM	INTEGER	Total in management duration for alarms of this hour
COLLECTION_COUNTER_SUM	INTEGER	Total counter of collection duration greater than 0 for alarms of this hour
IN_FAULT_COUNTER_SUM	INTEGER	Total counter of in fault duration greater than 0 for alarms of this hour
IN_MANAGEMENT_COUNTER_SUM	INTEGER	Total counter of in management duration greater than 0 for alarms of this hour
CLEAR_DURATION_AVG	FLOAT	Average of clear duration for alarms of this hour
HANDLE_DURATION_AVG	FLOAT	Average of handle duration for alarms of this hour
CLOSE_DURATION_AVG	FLOAT	Average of close duration for alarms of this hour
ACK_DURATION_AVG	FLOAT	Average of acknowledgment duration for alarms of this hour
TERM_DURATION_AVG	FLOAT	Average of termination duration for alarms of this hour
COLLECTION_DURATION_AVG	FLOAT	Average of collection duration for alarms of this hour
IN_FAULT_DURATION_AVG	FLOAT	Average of in fault duration for alarms of this hour
IN_MANAGEMENT_DURATION_AVG	FLOAT	Average of in management duration for alarms of this hour

COLUMN NAME	COLUMN TYPE	DESCRIPTION
UPDATE_TIMESTAMP	TIMESTAMP	Time at which this record has been persisted

Here is the description of the **SUMM_DAILY_EQUIPMENT** table:

COLUMN NAME	COLUMN TYPE	DESCRIPTION
TIME	TIMESTAMP	Timestamp of the Day for which the aggregation occurred
ALARMTYPEID	INTEGER	Alarm Type Identifier (Foreign Key) for the alarms aggregated for this day
SEVERITYID	INTEGER	Severity Identifier (Foreign Key) for the alarms aggregated for this day
OUTAGEFLAG	BOOLEAN	Specifies if the alarms aggregated on this day are from out of service equipment or not
STATEID	INTEGER	State Identifier (Foreign Key) for the alarms aggregated for this day
MANAGEDOBJECTNAME	VARCHAR	Managed Object Name (Foreign Key) for the alarms aggregated for this day
PROBABLECAUSEID	INTEGER	Probable Cause Identifier (Foreign Key) for the alarms aggregated for this day
PROBLEMSTATUSID	INTEGER	Problem Status Identifier (Foreign Key) for the alarms aggregated for this day
SPECIFICPROBLEMSID	INTEGER	Specific Problems Identifier (Foreign Key) for the alarms aggregated for this day
SUMMARIZED_COUNT	INTEGER	Number of FCT_FAULT rows aggregated for this day
ALARM_OBJECTS_COUNTER_SUM	INTEGER	Number of alarm objects during this day
SIMILAR_ALARM_COUNTER_SUM	INTEGER	Number of similar alarms during this day
ALARM_COUNTER_SUM	INTEGER	Number of alarms (AO and similar) during this day
CRITICAL_OCCURENCES_SUM	INTEGER	Number of critical occurrences during this day
MAJOR_OCCURENCES_SUM	INTEGER	Number of major occurrences during this day
MINOR_OCCURENCES_SUM	INTEGER	Number of minor occurrences during this day
WARNING_OCCURENCES_SUM	INTEGER	Number of warning occurrences during this day
INDETERMINATE_OCCURENCES_SUM	INTEGER	Number of indeterminate occurrences during this day
CLEAR_OCCURENCES_SUM	INTEGER	Number of clear occurrences during this day
CLEAR_COUNTER_SUM	INTEGER	Number of clearance during this day
ESCALATED_COUNTER_SUM	INTEGER	Number of escalated alarms during this day
HANDLED_COUNTER_SUM	INTEGER	Number of handled alarms during this day
CLOSED_COUNTER_SUM	INTEGER	Number of closed alarms during this day
ACK_COUNTER_SUM	INTEGER	Number of acknowledged alarms during this day
TERMINATED_FLAG_SUM	INTEGER	Number of terminated alarms during this day
ACK_BY_COUNT	INTEGER	Number of user who acknowledged alarms during this day
CLOSED_BY_COUNT	INTEGER	Number of user who closed alarms during this day
TERMINATED_BY_COUNT	INTEGER	Number of user who terminated alarms during this day
RELEASED_BY_COUNT	INTEGER	Number of user who released alarms during this day

COLUMN NAME	COLUMN TYPE	DESCRIPTION
HANDLED_BY_COUNT	INTEGER	Number of user who handled alarms during this day
CLEAR_DURATION_SUM	INTEGER	Total clear duration for alarms of this day
HANDLE_DURATION_SUM	INTEGER	Total handle duration for alarms of this day
CLOSE_DURATION_SUM	INTEGER	Total close duration for alarms of this day
ACK_DURATION_SUM	INTEGER	Total acknowledgment duration for alarms of this day
TERM_DURATION_SUM	INTEGER	Total termination duration for alarms of this day
COLLECTION_DURATION_SUM	INTEGER	Total collection duration for alarms of this day
IN_FAULT_DURATION_SUM	INTEGER	Total in fault duration for alarms of this day
IN_MANAGEMENT_DURATION_SUM	INTEGER	Total in management duration for alarms of this day
COLLECTION_COUNTER_SUM	INTEGER	Total counter of collection duration greater than 0 for alarms of this day
IN_FAULT_COUNTER_SUM	INTEGER	Total counter of in fault duration greater than 0 for alarms of this day
IN_MANAGEMENT_COUNTER_SUM	INTEGER	Total counter of in management duration greater than 0 for alarms of this day
CLEAR_DURATION_AVG	FLOAT	Average of clear duration for alarms of this day
HANDLE_DURATION_AVG	FLOAT	Average of handle duration for alarms of this day
CLOSE_DURATION_AVG	FLOAT	Average of close duration for alarms of this day
ACK_DURATION_AVG	FLOAT	Average of acknowledgment duration for alarms of this day
TERM_DURATION_AVG	FLOAT	Average of termination duration for alarms of this day
COLLECTION_DURATION_AVG	FLOAT	Average of collection duration for alarms of this day
IN_FAULT_DURATION_AVG	FLOAT	Average of in fault duration for alarms of this day
IN_MANAGEMENT_DURATION_AVG	FLOAT	Average of in management duration for alarms of this day
UPDATE_TIMESTAMP	TIMESTAMP	Time at which this record has been persisted

2.1.4 FCT_FAULT_COMMENT

The table FCT_FAULT_COMMENT is another fact table. It contains the comments associated to alarms. The comment is uniquely identified by the operation context of the alarm, the identifier of the alarm, the user who set the comment and the time at which the comment was set.

FCT_FAULT_COMMENT table

COLUMN NAME	COLUMN TYPE
OPERATIONCONTEXTID	INTEGER
IDENTIFIER	INTEGER
USERID	VARCHAR(100)
TIME	TIMESTAMP
TEXT	VARCHAR(1024)
UPDATE_TIMESTAMP	TIMESTAMP

Metadata

3.1 Metadata concept

The first part of this document dealt with the FAS Datamart.

This chapter deals with the **FAS metadata**.

Metadata is an .xml file that represents an abstraction of the datamart.

Thus, metadata describes:

- the definitions of Dimensions
- the definitions of Facts
- the relations between Facts & Dimensions
- the display names, descriptions, units for OSS Console
- the datasource, tables and columns where to retrieve data from the Datamart
- how to retrieve fact values, and/or aggregate values (depending on the granularity requested)
- the roles allowed for accessing an object (package, dimension, fact)

3.2 FAS metadata description

After installing the FAS kit you will find the FAS metadata at this location:

/opt/ossa/repo-fas/FAS/metadata/ossa_fault_metadata.xml

It is important to understand that this FAS metadata contains identifiers, and display names of Facts and Dimensions which are directly used by the OSS Console FAS reports for determining data to display.

Here is the description of the FAS metadata.

Metadata header:

```
<OSSAPackage id="fault"
  displayName="OSS Analytics -- Fault" version="0.1"
  xsi:noNamespaceSchemaLocation="OSSA_packageDesc.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
  <Description>OSS Analytics -- Fault metadata to query TeMIP Fault data mart through the Rest API</Description>
```

This section contains:

- the name of the FAS metadata package: *fault*
- its display name: *OSS Analytics – Fault*
- its description

Natural dimensions declarations:

```
<Dims>
<Dim id="dimAlarmType" displayName="Dimension Alarm Type" schemaName="ossa_fault" tableName="DIM_ALARMSTYPE" category="Fault">
<Dim id="dimDomain" displayName="Dimension Domain" schemaName="ossa_fault" tableName="DIM_DOMAIN" category="Fault">
<Dim id="dimOC" displayName="Dimension Operation Context" schemaName="ossa_fault" tableName="DIM_OPERATIONCONTEXT" category="Fault">
<Dim id="dimMO" displayName="Dimension Managed Object" schemaName="ossa_fault" tableName="DIM_MANAGEDOBJECT" category="Fault">
<Dim id="dimPC" displayName="Dimension Probable Cause" schemaName="ossa_fault" tableName="DIM_PROBABLECAUSE" category="Fault">
<Dim id="dimSP" displayName="Dimension Specific Problems" schemaName="ossa_fault" tableName="DIM_SPECIFICPROBLEMS" category="Fault">
<Dim id="dimPRA" displayName="Dimension Proposed Repair Actions" schemaName="ossa_fault" tableName="DIM_PROPOSEDREPAIRACTIONS" category="Fault">
<Dim id="dimSev" displayName="Dimension Severity" schemaName="ossa_fault" tableName="DIM_SEVERITY" category="Fault">
<Dim id="dimState" displayName="Dimension State" schemaName="ossa_fault" tableName="DIM_STATE" category="Fault">
<Dim id="dimPbState" displayName="Dimension Problem Status" schemaName="ossa_fault" tableName="DIM_PROBLEMSTATUS" category="Fault">
<Dim id="dimUser" displayName="Dimension User" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
<Dim id="dimUserAck" displayName="Dimension User Ack" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
<Dim id="dimUserClose" displayName="Dimension User Close" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
<Dim id="dimUserHandle" displayName="Dimension User Handle" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
<Dim id="dimUserRelease" displayName="Dimension User Release" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
<Dim id="dimUserTerm" displayName="Dimension User Term" schemaName="ossa_fault" tableName="DIM_USER" category="Fault">
</Dims>
```

This section presents all the FAS natural dimensions, with their display name (names that will be displayed in the OSS Console) and with their corresponding datamart schema names and datamart table names.

Focusing on one natural dimension, you can see its attributes declarations, for example, the *Domain* dimension is declared like:

```
<Dim id="dimDomain" displayName="Dimension Domain" schemaName="ossa_fault" tableName="DIM_DOMAIN" category="Fault">
<Description>...</Description>
<DimCol type="Integer" displayName="DOMAINID" colName="DOMAINID" lowCardinalityFlag="true" hiddenFlag="true"/>
<DimCol type="String" displayName="DOMAINNAME" colName="DOMAINNAME" lowCardinalityFlag="true"/>
<SurrogateKey colName="DOMAINID"/>
</Dim>
```

This sub-section declares the association between dimension attributes and their corresponding columns in the datamart.

The *SurrogateKey* of the dimension table is also declared, and will be referenced later in the metadata within the Fact section (<*FactPatterns*>) in order to associate a fact with a dimension.

The *lowCardinalityFlag* allows the OSS Console to behaves differently depending on the number of different values for the dimension attribute:

- *lowCardinalityFlag* = true : very few number of values will be stored in datamart for this dimension attribute
- *lowCardinalityFlag* = false : high number of values will be stored in datamart for this dimension attribute.

The *category* allows to organize the dimensions presentation within the OSS Console.

The *hiddenFlag* allows the OSS Console to determine whether to expose or not the dimension attribute.

Degenerated dimensions declarations:

```
<DegeneratedDims>
<DegeneratedDim id="ddimFault" displayName="Dimension Fact Fault" category="Fault">
<DegeneratedDim id="ddimFaultComment" displayName="Dimension Fact Fault Comment" category="Fault">
<DegeneratedDim id="ddimSummMgmt" displayName="Dimension for Summarization Management Tables" category="Fault">
<DegeneratedDim id="ddimSummEq" displayName="Dimension for Summarization Equipment Tables" category="Fault">
<DegeneratedDim id="ddimSummNw" displayName="Dimension for Summarization Network Tables" category="Fault">
</DegeneratedDims>
```

This section presents the all the FAS degenerated dimensions.

Degenerated dimensions are group of degenerated dimension attributes which act are standard dimensions from user perspective (from OSS Console) but which are stored in FACTS tables within datamart (whereas natural dimensions are stored in specific dimension tables). That is why their declarations are a bit different.

Focusing on one degenerated dimension, you can see its attributes declarations. For example, here is an extract of the *ddimFault* degenerated dimension:

```

<DegeneratedDim id="ddimFault" displayName="Dimension Fact Fault" category="Fault">
<Description>Dimension for FCT_FAULT</Description>
<DimCol type="Integer" displayName="FAULTID" colName="FAULTID" lowCardinalityFlag="false" hiddenFlag="true"/>
<DimCol type="Integer" displayName="IDENTIFIER" colName="IDENTIFIER" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="EVENTTIME" colName="EVENTTIME" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="CREATIONTIMESTAMP" colName="CREATIONTIMESTAMP" lowCardinalityFlag="false"/>
<DimCol type="Integer" displayName="ALARMORIGIN" colName="ALARMORIGIN" lowCardinalityFlag="true"/>
<DimCol type="Boolean" displayName="ESCALATEDALARMSFLAG" colName="ESCALATEDALARMSFLAG" lowCardinalityFlag="true"/>
<DimCol type="Boolean" displayName="CLEARFLAG" colName="CLEARFLAG" lowCardinalityFlag="true"/>
<DimCol type="Boolean" displayName="OUTAGEFLAG" colName="OUTAGEFLAG" lowCardinalityFlag="true"/>
<DimCol type="Timestamp" displayName="ACKTIMESTAMP" colName="ACKTIMESTAMP" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="CLEARTIMESTAMP" colName="CLEARTIMESTAMP" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="CLOSETIMESTAMP" colName="CLOSETIMESTAMP" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="HANDLETIMESTAMP" colName="HANDLETIMESTAMP" lowCardinalityFlag="false"/>
<DimCol type="Timestamp" displayName="RRRASRTTIMESTAMP" colName="RRRASRTTIMESTAMP" lowCardinalityFlag="false"/>

```

Those degenerated dimensions are referenced later in the metadata file, at the place where FACTS are declared (*colName* declared here corresponds to column of the datamart FACT table referencing degenerated dimension later).

Facts declarations:

Here is an extract of the FAS Facts declarations:

```

<Facts>
<Fact id="PROBLEMOCCURRENCES" displayName="PROBLEMOCCURRENCES" type="Integer" unit="#"/>
<Fact id="CLEAROCCURRENCES" displayName="CLEAROCCURRENCES" type="Integer" unit="#"/>
<Fact id="CRITICALOCCURRENCES" displayName="CRITICALOCCURRENCES" type="Integer" unit="#"/>
<Fact id="INDETERMINATEOCCURRENCES" displayName="INDETERMINATEOCCURRENCES" type="Integer" unit="#"/>
<Fact id="MAJOROCCURRENCES" displayName="MAJOROCCURRENCES" type="Integer" unit="#"/>
<Fact id="MINOROCCURRENCES" displayName="MINOROCCURRENCES" type="Integer" unit="#"/>
<Fact id="WARNINGOCCURRENCES" displayName="WARNINGOCCURRENCES" type="Integer" unit="#"/>
<Fact id="SATOTAL" displayName="SATOTAL" type="Integer" unit="#"/>
<Fact id="CNTID" displayName="CNTID" type="Integer" unit="#"/>

<Fact id="ACKDURATION" displayName="ACKDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="CLEARDURATION" displayName="CLEARDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="CLOSEDURATION" displayName="CLOSEDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="HANDLEDURATION" displayName="HANDLEDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="TERMDURATION" displayName="TERMDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="COLLECTIONDURATION" displayName="COLLECTIONDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="INFAULTDURATION" displayName="INFAULTDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>
<Fact id="INMGMTDURATION" displayName="INMGMTDURATION" type="Integer" unit="ms" worstOrdering="Asc"/>

<Fact id="ACKDURATION_AVG" displayName="ACKDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="CLEARDURATION_AVG" displayName="CLEARDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="CLOSEDURATION_AVG" displayName="CLOSEDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="HANDLEDURATION_AVG" displayName="HANDLEDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="TERMDURATION_AVG" displayName="TERMDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="COLLECTIONDURATION_AVG" displayName="COLLECTIONDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="INFAULTDURATION_AVG" displayName="INFAULTDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>
<Fact id="INMGMTDURATION_AVG" displayName="INMGMTDURATION_AVG" type="Float" unit="ms" worstOrdering="Asc"/>

```

This section only lists all the Facts which are stored in the FAS datamart. They are referenced later in the metadata, within the *FactCalculation* section.

Fact Patterns declarations:

Here are the FAS Fact Patterns:

```

<FactPatterns>
<FactPattern id="fctFault">
<FactPattern id="fctFaultComment">
<FactPattern id="summMgmt">
<FactPattern id="summEq">
<FactPattern id="summNw">
</FactPatterns>

```

A *FactPattern* represents a set of Facts which can be found in different Datamart tables having the same table definition, but in which the fact values are time-aggregated differently, for example: hourly, daily, weekly, monthly aggregation. You can see above that FAS metadata exposes the facts concerning the raw alarms (*id="fctFault"*), the facts containing the alarm comments (*id="fctFaultComment"*), and the 3 facts containing the summarized data, on the 3 kind of summarizations: Management (*id="summMgmt"*), Equipement (*id="summEq"*), Network (*id="summNw"*).

Here is an extract of the fact pattern *summEq* describing the facts issued from summarized tables dealing with Equipment analysis:

```

<FactPattern id="summEq">
  <FactCalculation factIdRef="ALARM OBJECTS COUNTER SUM" colName="ALARM OBJECTS COUNTER SUM">
  <FactCalculation factIdRef="SIMILAR ALARM COUNTER SUM" colName="SIMILAR ALARM COUNTER SUM">
  <FactCalculation factIdRef="ALARM COUNTER SUM" colName="ALARM COUNTER SUM">
  <FactCalculation factIdRef="CRITICAL OCCURRENCES SUM" colName="CRITICAL OCCURRENCES SUM">
  <FactCalculation factIdRef="MAJOR OCCURRENCES SUM" colName="MAJOR OCCURRENCES SUM">
  <FactCalculation factIdRef="MINOR OCCURRENCES SUM" colName="MINOR OCCURRENCES SUM">
  <FactCalculation factIdRef="WARNING OCCURRENCES SUM" colName="WARNING OCCURRENCES SUM">
  <FactCalculation factIdRef="INDETERMINATE OCCURRENCES SUM" colName="INDETERMINATE OCCURRENCES SUM">
  <FactCalculation factIdRef="CLEAR OCCURRENCES SUM" colName="CLEAR OCCURRENCES SUM">
  <FactCalculation factIdRef="CLEAR COUNTER SUM" colName="CLEAR COUNTER SUM">
  <FactCalculation factIdRef="ESCALATED COUNTER SUM" colName="ESCALATED COUNTER SUM">
  <FactCalculation factIdRef="HANDLED COUNTER SUM" colName="HANDLED COUNTER SUM">
  <FactCalculation factIdRef="CLOSED COUNTER SUM" colName="CLOSED COUNTER SUM">
  <FactCalculation factIdRef="ACK COUNTER SUM" colName="ACK COUNTER SUM">
  <FactCalculation factIdRef="TERMINATED FLAG SUM" colName="TERMINATED FLAG SUM">
  <FactCalculation factIdRef="ACK BY COUNT" colName="ACK BY COUNT">
  ...
  <FactCalculation factIdRef="IN MANAGEMENT DURATION MIN AVG" colName="IN MANAGEMENT DURATION AVG">
  <FactCalculation factIdRef="IN MANAGEMENT DURATION HR AVG" colName="IN MANAGEMENT DURATION AVG">
  <DegeneratedDim ddimIdRef="ddimSummEq"/>
  <DataSources>
  <Joins>
</FactPattern>

```

Within a *FactPattern*, are declared the:

- *<FactCalculation>* : describing how to retrieve and calculate fact values (for facts referencing already defined *<Fact>* entries)
- *<DegeneratedDim>* : degenerated dimensions associated to the fact. This attribute *DegeneratedDim ddimIdRef* just references *<DegeneratedDim>* already defined entries from the first sections of the metadata
- *<DataSources>* : the location of the facts within the FAS datamart
- *<Joins>* : the associated natural dimensions of the fact

Fact Pattern FactCalculation declarations:

Here is a *FactCalculation* declared in the *FactPattern summEq*:

```

<FactCalculation factIdRef="ALARM_COUNTER_SUM" colName="ALARM_COUNTER_SUM">
  <MultiRowSQLExpression>(sum (ALARM_COUNTER_SUM) )</MultiRowSQLExpression>
</FactCalculation>

```

- *factIdRef* : this *FactCalculation* refers to the *Fact Id ALARM_COUNTER_SUM* (previously declared in *<Fact>*)
- *colName* : the fact retrieval is done from the column named "ALARM_COUNTER_SUM"
- *MultiRowSQLExpression* : defines the SQL expression to be used when time aggregation is requested. In this case, when requesting the counter of alarms for a granularity higher than the time granularity defined for this table, the *sum(ALARM_COUNTER_SUM)* is applied on the related fact rows.

Fact Pattern DataSources declarations:

Here are the *DataSources* declared in the *FactPattern summEq*:

```
<DataSources>
  <DataSource schema="ossa_fault" table="SUMM_DAILY_EQUIPMENT" timestampCol="TIME" timePeriodDuration="1440"/>
  <DataSource schema="ossa_fault" table="SUMM_WEEKLY_EQUIPMENT" timestampCol="TIME" timePeriodDuration="10080"/>
  <DataSource schema="ossa_fault" table="SUMM_MONTHLY_EQUIPMENT" timestampCol="TIME" timePeriodDuration="43200"/>
</DataSources>
```

This section declares 3 possible datasources for *summEq* fact pattern.

They reference the same schema (*schema="ossa_fault"*), but different tables (*SUMM_DAILY_EQUIPMENT*, *SUMM_WEEKLY_EQUIPMENT*, *SUMM_MONTHLY_EQUIPMENT*) having the same table definition.

The difference is that they do not store the same aggregation level of fact values:

- *timePeriodDuration = 1440* minutes
=> which represents daily aggregation, in *SUMM_DAILY_EQUIPMENT* table
- *timePeriodDuration = 10080* minutes
=> which represents weekly aggregation, in *SUMM_WEEKLY_EQUIPMENT* table
- *timePeriodDuration = 43200* minutes
=> which represents monthly aggregation, in *SUMM_MONTHLY_EQUIPMENT*

Depending on the granularity used by an OSS Console report for requesting a fact, the relevant datasource will be used for retrieving the data.

Fact Pattern Joins declarations:

Here are the *Joins* declared in the *FactPattern summEq*:

```
<Joins>
  <Join joinType="InnerJoin" dimIdRef="dimMO" fkColInFactTable="MANAGEDOBJECTNAME"/>
  <Join joinType="InnerJoin" dimIdRef="dimAlarmType" fkColInFactTable="ALARMTYPEID"/>
  <Join joinType="InnerJoin" dimIdRef="dimPC" fkColInFactTable="PROBABLECAUSEID"/>
  <Join joinType="LeftOuterJoin" dimIdRef="dimSP" fkColInFactTable="SPECIFICPROBLEMSID"/>
  <Join joinType="InnerJoin" dimIdRef="dimSev" fkColInFactTable="SEVERITYID"/>
  <Join joinType="InnerJoin" dimIdRef="dimState" fkColInFactTable="STATEID"/>
  <Join joinType="InnerJoin" dimIdRef="dimPbState" fkColInFactTable="PROBLEMSTATUSID"/>
</Joins>
```

They declare all the associations from the *FactPattern* to its natural dimensions in term of type of join (*joinType*), natural dimension referenced (*dimIdRef* previously declared in *<Dims>*), and foreignKey (*fkColInFactTable*) used for the association.

Visualization declaration:

Here is the *Visualization* declaration:

```
<Visualization>
  <Service name="Alarm Facts">
    <Fact idRef="PROBLEMOCCURRENCES"/>
    <Fact idRef="CLEAROCCURRENCES"/>
    <Fact idRef="CRITICALOCCURRENCES"/>
    <Fact idRef="INDETERMINATEOCCURRENCES"/>
    <Fact idRef="MAJOROCCURRENCES"/>
    ...
  </Service>

  <Service name="Alarm Summarizations">
    <Fact idRef="ALARM_COUNTER_SUM"/>
    <Fact idRef="ALARM_OBJECTS_COUNTER_SUM"/>
    <Fact idRef="SIMILAR_ALARM_COUNTER_SUM"/>
    <Fact idRef="ESCALATED_COUNTER_SUM"/>
    ...
  </Service>
</Visualization>
```

This *Visualization* section is used by the OSS Console in order to organize the facts visualization per service and possible subfolders.

With this *FAS Visualization*, in OSS Console the facts concerning the raw alarms will be put in a folder named "Alarm facts" and the aggregated facts (coming from summarization) will be put in a folder named "Alarm Summarizations".

FAS default reports

Please refer to the *HP Fault Analytics & Statistics user guide* for a presentation of the FAS default reports.

FAS Customization

5.1 Mapping rules from TeMIP alarms to FAS datamart

5.1.1 Specific Problems configuration

Fault Analytics & Statistics let the integrator choosing whether he needs to support TeMIP alarm Specific Problem or not.

The choice is driven by the configuration of FAS as introduced in the “*Configuring FAS alarm export application*” section of the *HP Fault Analytics & Statistics – Installation and Administration guide*.

The command for setting the FAS parameter values is:

```
`${OSSA_HOME}/bin/ossa-repo.sh setParam OSSA_FAULT AlarmExportConfig <XX>
```

where <xx> is a json description of the configuration.

Within this json description you can define which alarm Specific Problems should be exported to Vertica or not. The setting is done per TeMIP Global Class.

See the following example of FAS configuration json description:

```
{
  "consumerGroupConfigs": [
    {
      "consumerGroupProperties": {
        "group.id": "st-01",
        "zookeeper.connect": "myhost.com:2181",
        "auto.commit.enable": "true",
        "auto.offset.reset": "smallest",
        "rebalance.max.retries": "20",
        "rebalance.backoff.ms": "10000",
        "zookeeper.session.timeout.ms": "60000",
        "zookeeper.connection.timeout.ms": "60000"
      },
      "consumerThreadNumber": 16
    }
  ],
  "temipClassSpecificProblems": [
    {
      "temipClass": "OSI_SYSTEM",
      "supportsSP": true
    }, {
      "temipClass": "",
      "supportsSP": false
    }
  ]
}
```

The configuration example above defines that Specific Problems attribute of alarms which TeMIP Global Class starts with *OSI_SYSTEM* will be persisted into *DIM_SPECIFICPROBLEMS* table. Obviously, those alarms stored into *FCT_FAULT* will have their *SPECIFICPROBLEMSID* referencing the corresponding *DIM_SPECIFICPROBLEMS*.

The configuration example above also says that alarms with other TeMIP Global Class will not have their specific problems stored into *DIM_SPECIFICPROBLEMS*.

5.2 FAS solution enrichment

5.2.1 User Defined Attributes example

In order to introduce the possible customizations of FAS, we will take the common example of adding TeMIP User Defined Attributes within FAS solution.

5.2.1.1 User Defined Attributes datamart enrichment

In addition to the standard TeMIP attributes, FAS can handle TeMIP user defined attributes.

In the following, we will take as example one new User Defined Attribute named MYUDA. Obviously, the same steps must be done for all the User Defined Attributes required in Datamart.

First, you need to create your own .ddl file containing the declaration of the additional columns corresponding to the user defined attributes needed in the FAS datamart schema. Then, you will have to load your .ddl file, as FAS database user.

FCT_FAULT table

Start by adding column definition for FCT_FAULT table.

```
ALTER TABLE FCT_FAULT ADD COLUMN <Column Name> <Column Constraint>;
```

For example:

```
ALTER TABLE FCT_FAULT ADD COLUMN MYUDA VARCHAR(1024);
```

Then, enrich the USER_DEFINED_ATTRIBUTE table.

```
INSERT INTO USER_DEFINED_ATTRIBUTE (ATTR_ID, ATTR_NAME, ATTR_TYPE, MAX_SIZE)
VALUES (<Attribute ID>, '<Column Name>', '<Attribute Type>', <Max Size>);
```

For example:

```
INSERT INTO USER_DEFINED_ATTRIBUTE (ATTR_ID, ATTR_NAME, ATTR_TYPE, MAX_SIZE)
VALUES (10056, 'MYUDA', 'String', 1024);
```

The <Attribute ID> must correspond to the ID of the attribute within TeMIP.

The <Column Name> must be the name of the column you just added in FCT_FAULT.

The <Attribute Type> is used for classification, you can specify “String”, “Int” or “Long”.

The <Max Size> is useful only when the type of the column you add is “VARCHAR”. It tells FAS to truncate the extra characters when needed. Please make sure that the value corresponds to the column definition from FCT_FAULT table.

Now, you must restart your OSSA Server:

as *ossa* linux user, in a terminal where you have sourced the *ossa* environment:

```
$ jbossstop
$ jbossstart
```

FAS is now able to export User Defined Attributes to FAS Datamart table FCT_FAULT.

Summarizations tables

Important point: based on the OSS Console Views you want to design (dealing with summarized data), determine the necessary User Defined Attributes to add within the summarized tables definitions for each of the three summarization categories: Network, Management or Equipment.

So, add the column definitions for each required User Defined Attribute within all required summarized tables by executing:

```
ALTER TABLE SUMM_XXX ADD COLUMN <Column Name> <Column Type>;
```

In the following example, we add the MYUDA User Defined Attribute in the 11 existing summarized tables:

```
ALTER TABLE SUMM_DAILY_EQUIPMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_WEEKLY_EQUIPMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_MONTHLY_EQUIPMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_HOURLY_MANAGEMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_DAILY_MANAGEMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_WEEKLY_MANAGEMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_MONTHLY_MANAGEMENT ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_HOURLY_NETWORK ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_DAILY_NETWORK ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_WEEKLY_NETWORK ADD COLUMN MYUDA VARCHAR(200);
ALTER TABLE SUMM_MONTHLY_NETWORK ADD COLUMN MYUDA VARCHAR(200);
```

and, modify the UNIQUE constraints on the tables you have modified, for example:

```
-- first drop existing unique constraints on summarized table
ALTER TABLE SUMM_DAILY_EQUIPMENT DROP CONSTRAINT UQ_SUMM_DAILY_EQUIPMEN;
ALTER TABLE SUMM_WEEKLY_EQUIPMENT DROP CONSTRAINT UQ_SUMM_WEEKLY_EQUIPM;
ALTER TABLE SUMM_MONTHLY_EQUIPMENT DROP CONSTRAINT UQ_SUMM_MONTHLY_EQ;
ALTER TABLE SUMM_HOURLY_MANAGEMENT DROP CONSTRAINT UQ_SUMM_HOURLY_MANAGEME;
ALTER TABLE SUMM_DAILY_MANAGEMENT DROP CONSTRAINT UQ_SUMM_DAILY_MANAGEME;
ALTER TABLE SUMM_WEEKLY_MANAGEMENT DROP CONSTRAINT UQ_SUMM_WEEKLY_MGMT;
ALTER TABLE SUMM_MONTHLY_MANAGEMENT DROP CONSTRAINT UQ_SUMM_MONTHLY_MGMT;
ALTER TABLE SUMM_HOURLY_NETWORK DROP CONSTRAINT UQ_SUMM_HOURLY_NETWORK;
ALTER TABLE SUMM_DAILY_NETWORK DROP CONSTRAINT UQ_SUMM_DAILY_NETWORK;
ALTER TABLE SUMM_WEEKLY_NETWORK DROP CONSTRAINT UQ_SUMM_WEEKLY_NETWORK;
ALTER TABLE SUMM_MONTHLY_NETWORK DROP CONSTRAINT UQ_SUMM_MONTHLY_NETWORK;
-- then create the new unique constraints with new user defined attribute
ALTER TABLE SUMM_DAILY_EQUIPMENT ADD CONSTRAINT UQ_SUMM_DAILY_EQUIPMEN UNIQUE
    (TIME, MANAGEDOBJECTNAME, ALARMTYPEID, PROBABLECAUSEID,
     SPECIFICPROBLEMSID, PROBLEMSTATUSID, SEVERITYID, STATEID, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_WEEKLY_EQUIPMENT ADD CONSTRAINT UQ_SUMM_WEEKLY_EQUIPM UNIQUE
    (TIME, MANAGEDOBJECTNAME, ALARMTYPEID, PROBABLECAUSEID,
     SPECIFICPROBLEMSID, PROBLEMSTATUSID, SEVERITYID, STATEID, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_MONTHLY_EQUIPMENT ADD CONSTRAINT UQ_SUMM_MONTHLY_EQ UNIQUE
    (TIME, MANAGEDOBJECTNAME, ALARMTYPEID, PROBABLECAUSEID,
     SPECIFICPROBLEMSID, PROBLEMSTATUSID, SEVERITYID, STATEID, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_HOURLY_MANAGEMENT ADD CONSTRAINT UQ_SUMM_HOURLY_MANAGEME UNIQUE
    (TIME, OPERATIONCONTEXTID, ACKUSERID, CLOSEUSERID, HANDLEUSERID,
     RELEASEUSERID, TERMUSERID, OUTAGEFLAG, MYUDA);
ALTER TABLE SUMM_DAILY_MANAGEMENT ADD CONSTRAINT UQ_SUMM_DAILY_MANAGEME UNIQUE
    (TIME, OPERATIONCONTEXTID, ACKUSERID, CLOSEUSERID, HANDLEUSERID,
     RELEASEUSERID, TERMUSERID, OUTAGEFLAG, MYUDA);
ALTER TABLE SUMM_WEEKLY_MANAGEMENT ADD CONSTRAINT UQ_SUMM_WEEKLY_MGMT UNIQUE
    (TIME, OPERATIONCONTEXTID, ACKUSERID, CLOSEUSERID, HANDLEUSERID,
     RELEASEUSERID, TERMUSERID, OUTAGEFLAG, MYUDA);
ALTER TABLE SUMM_MONTHLY_MANAGEMENT ADD CONSTRAINT UQ_SUMM_MONTHLY_MGMT UNIQUE
    (TIME, OPERATIONCONTEXTID, ACKUSERID, CLOSEUSERID, HANDLEUSERID,
     RELEASEUSERID, TERMUSERID, OUTAGEFLAG, MYUDA);
ALTER TABLE SUMM_HOURLY_NETWORK ADD CONSTRAINT UQ_SUMM_HOURLY_NETWORK UNIQUE
    (TIME, DOMAINID, OPERATIONCONTEXTID, MANAGEDOBJECTNAME, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_DAILY_NETWORK ADD CONSTRAINT UQ_SUMM_DAILY_NETWORK UNIQUE
    (TIME, DOMAINID, OPERATIONCONTEXTID, MANAGEDOBJECTNAME, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_WEEKLY_NETWORK ADD CONSTRAINT UQ_SUMM_WEEKLY_NETWORK UNIQUE
    (TIME, DOMAINID, OPERATIONCONTEXTID, MANAGEDOBJECTNAME, OUTAGEFLAG,
     MYUDA);
ALTER TABLE SUMM_MONTHLY_NETWORK ADD CONSTRAINT UQ_SUMM_MONTHLY_NETWORK UNIQUE
    (TIME, DOMAINID, OPERATIONCONTEXTID, MANAGEDOBJECTNAME, OUTAGEFLAG,
     MYUDA);
```

Please ensure that your Datamart has been enhanced with new User Defined Attributes within the FCT_FAULT table and within the summarized tables, and that the constraints have been modified accordingly.

5.2.1.2 User Defined Attributes summarization batch job enrichment

You must now modify the summarization batch job in order to take into account the User Defined Attributes.

As *ossa* linux user, modify the summarization batch job description:

```
`${OSSA_HOME}/repo-fas/FAS/batch/FASsummJob.xml
```

For each summarization on which you want to handle User Defined Attribute, add its column name in the property `src_dimensions`:

```
<property name="src_dimensions" value="...,MYUDA" />
```

For example:

```
<property name="src_dimensions"
value="OPERATIONCONTEXTID,ACKUSERID,CLOSEUSERID,HANDLEUSERID,
RELEASEUSERID,TERMUSERID,OUTAGEFLAG,MYUDA" />
```

If all the summarizations needs to have the `MYUDA` User Defined Attribute, 11 modifications of this kind must be done in the summarization batch job description.

Then, load the new summarization batch job description into the OSSA server: as *ossa* linux user, in a terminal where you have sourced the *ossa* environment:

```
sh `${OSSA_HOME}/bin/ossa-repo.sh loadParam FAS FASsummJob.xml \
`${OSSA_HOME}/repo-fas/FAS/batch/FASsummJob.xml
```

The new summarization batch job will now be scheduled, and the new User Defined Attributes will be populated in the summarized tables.

5.2.1.3 User Defined Attributes FAS metadata enrichment

Start by backing up the original fault metadata:

```
`${OSSA_HOME}/repo-fas/FAS/metadata/ossa_fault_metadata.xml
```

Then, enhance it by adding the necessary User Defined Attributes for the four degenerated dimensions definitions `ddimFault`, `ddimSummMgmt`, `ddimSummEq`, `ddimSummNw`.

For example:

```
...
<DegeneratedDim id="ddimFault" displayName="Dimension Fact Fault" category="Fault">
  <Description>Dimension for FCT_FAULT</Description>
...
  <DimCol type="String" lowCardinalityFlag="false" displayName="My User Defined Attribute"
colName="MYUDA"/>
...

<DegeneratedDim id="ddimSummMgmt" displayName="Dimension for Summarization Management Tables"
category="Fault">
  <Description>Dimension for Summarization Management Tables</Description>
...
  <DimCol type="String" displayName="MYUDA" colName="MYUDA" lowCardinalityFlag="false"/>
...

<DegeneratedDim id="ddimSummEq" displayName="Dimension for Summarization Equipment Tables"
category="Fault">
  <Description>Dimension for Summarization Equipment Tables</Description>
...
  <DimCol type="String" displayName="MYUDA" colName="MYUDA" lowCardinalityFlag="false"/>
...

<DegeneratedDim id="ddimSummNw" displayName="Dimension for Summarization Network Tables"
category="Fault">
  <Description>Dimension for Summarization Network Tables</Description>
...
  <DimCol type="String" displayName="MYUDA" colName="MYUDA" lowCardinalityFlag="false"/>
...
```

Then load the modified metadata (with the FAS views and workspaces also):

```
sh ${OSSA_HOME}/bin/ossa-repo.sh loadMetadataViewsWks \  
    ${OSSA_HOME}/repo-fas/FAS/metadata/ossa_fault_metadata.xml \  
    ${OSSA_HOME}/repo-fas/FAS/ui/views.json \  
    ${OSSA_HOME}/repo-fas/FAS/ui/workspaces.json
```

OSSA Server is now ready to handle requests concerning User Defined Attributes.

5.2.1.4 User Defined Attributes FAS reports

Once the FAS/OSSA Server has been enriched with User Defined Attributes, your OSS Console reports might need to be enhanced or new ones might need to be created.

For this, please refer to *HP Fault Analytics & Statistics User Guide*, and to *HP UOC User Guide*.

5.2.2 Other customizations

5.2.2.1 Other metadata enrichments

In the previous section, what has been highlighted concerning metadata enrichment is about addition of new simple degenerated dimensions.

Obviously, you can also add:

- more complex degenerated dimensions
- natural dimensions
- fact attributes

This is what is presented below.

Addition of SQL defined degenerated dimension:

In the same way that was described previously, a new degenerated dimension attribute can be added within metadata in the `<DegeneratedDims>` section. Here we take as example the addition of the a new `DegeneratedDim` based on the original event time of the alarm and that gives the number of the week in the year. Here is the example of this new degenerated dim named: `ORIGINALEVENTTIME_WEEK`

```
<DegeneratedDims>  
<DegeneratedDim id="ddimFault" displayName="Dimension Fact Fault" category="Fault">  
    ...  
    <DimCol type="Integer" displayName="ORIGINALEVENTTIME_WEEK" colName="ORIGINALEVENTTIME_WEEK" lowCardinalityFlag="true">  
        <RowSQLExpression>EXTRACT(ISO WEEK FROM ORIGINALEVENTTIME)</RowSQLExpression>  
    </DimCol>  
    ...
```

This shows that the retrieval of degenerated dimension value can be done thanks to a SQL clause.

Addition of natural dimensions

Concerning the addition of natural dimension, you just have to declare it in the `<Dims>` section, with the declaration of the corresponding Datamart table, column and declaration of the surrogate key. You can refer to the previous chapter *FAS metadata description | natural dimension declarations* for more details.

Note that your new natural dimension will certainly have to be associated from a fact: thus, the `<Joins>` part of the corresponding `<FactPattern>` will have to be enhanced.

Please refer to the previous chapter *FAS metadata description | Fact Pattern Joins declarations* for more details.

Addition of fact attributes

Concerning the addition of fact attributes, you must declare it in the relevant `<FactCalculation>` section.

In this example, for the *FactPattern summMgmt*, we declare a new fact attribute *CRITICAL_AND_MAJOR_OCCURRENCES_SUM* which is the sum of existing fact attributes *CRITICAL_OCCURRENCES_SUM* and *MAJOR_OCCURRENCES_SUM*.

```
<FactPattern id="summMgmt">
  ...
  <FactCalculation factIdRef="CRITICAL_AND_MAJOR_OCCURRENCES_SUM" colName="CRITICAL_AND_MAJOR_OCCURRENCES_SUM">
    <RowSQLExpression>CRITICAL_OCCURRENCES_SUM+MAJOR_OCCURRENCES_SUM</RowSQLExpression>
    <MultiRowSQLExpression>(sum(MAJOR_OCCURRENCES_SUM+CRITICAL_OCCURRENCES_SUM))</MultiRowSQLExpression>
  </FactCalculation>
  ...

```

The `<RowSQLExpression>CRITICAL_OCCURRENCES_SUM+MAJOR_OCCURRENCES_SUM</RowSQLExpression>` define the value of the fact when no time aggregation is requested.

When time aggregation is requested, the fact rows implied within the time granularity will be aggregated with the formula described in:

```
<MultiRowSQLExpression>(sum(MAJOR_OCCURRENCES_SUM+CRITICAL_OCCURRENCES_SUM))</MultiRowSQLExpression>
```

In this example, we define the fact attribute within the *FactPattern summMgmt*. So, as already explained in the *FAS metadata description / Fact Pattern DataSources declarations*, the fact attribute will be associated to the summarized tables *SUMM_HOURLY_MANAGEMENT*, *SUMM_DAILY_MANAGEMENT*, *SUMM_WEEKLY_MANAGEMENT*, *SUMM_MONTHLY_MANAGEMENT* related to Management analysis.

Finally, in order to have this new fact attribute visualizable in the OSS Console, you must add it in the *Visualization* section, as described below:

```
<Visualization>
  <Service name="Alarm Summarizations">
    ...
    <Fact idRef="CRITICAL_OCCURRENCES_SUM"/>
    <Fact idRef="MAJOR_OCCURRENCES_SUM"/>
    <Fact idRef="WARNING_OCCURRENCES_SUM"/>
    <Fact idRef="MINOR_OCCURRENCES_SUM"/>
    <Fact idRef="CLEAR_OCCURRENCES_SUM"/>
    <Fact idRef="INDETERMINATE_OCCURRENCES_SUM"/>
    ...
    <Fact idRef="CRITICAL_AND_MAJOR_OCCURRENCES_SUM"/>
    ...

```

5.2.2.2 Reports scheduling

The FAS / OSSA server / UOC solution allows the end user to schedule the generation of FAS reports and sending them by mails.

This can be done through the execution of customized scheduled batch jobs loaded into OSSA server.

Those customized batch jobs must use *ossa.ConsoleReport* batchlet (which allows to request the generation of UOC report) and *ossa.Mail* batchlet (which allows to send reports by mail).

FAS kit contains an example of such batch job, it is located at:

```
/opt/ossa/repo-fas/FAS/batch/report_WS-FAS-NETWORK-MGMT-HEALTH.xml
```

For detailed description of such kind of batch job, you can refer to the sections *OSSA Batchlet Library - ConsoleReport* and *OSSA Batchlet Library - Mailer* from *HP OSS Analytics Foundation Integration guide*.