
HP OpenView Service Quality Manager



VPN Services over MPLS Value Pack User's Guide

Edition: 1.4

December 2006

© Copyright 2006 Hewlett-Packard Company, L.P.

Legal Notices

Warranty

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.

The information contained herein is subject to change without notice.

Restricted Rights 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 2004, 2005-2006 Hewlett-Packard Development Company, L.P.

Trademark Notices

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

HP-UX Release 10.20 and later and HP-UX Release 11.00 and later (in both 32 and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

Java™ and all Java based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Microsoft®, Windows®, Windows NT® and Windows® XP are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered US trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of The Open Group.

Contents

Preface	5
Chapter 1	7
OV SQM VPN/MPLS Value Pack description	7
1.1 OpenView Service Quality Manager	7
1.2 What is a value pack	8
1.2.1 Objective	8
1.2.2 OV SQM Value Pack approach	8
1.2.3 Value pack typical content	12
1.2.4 An open solution	13
1.3 Introduction to the OV SQM VPN/MPLS Value Pack	13
1.3.1 Multiservice VPN solution	13
1.3.2 The advantages of MPLS-based VPN	14
1.3.3 VPN Service management challenges	15
1.3.4 The OpenView SQM VPN/MPLS Value Pack scope	15
1.3.5 Use cases	18
1.3.6 OV SQM VPN/MPLS Value Pack software add-on	18
1.3.7 OV SQM VPN/MPLS Value Pack: Key Quality Indicators	25
1.3.8 Solution in use	32
Chapter 2	41
Setting up the OV SQM VPN/MPLS Value Pack	41
2.1 Software and hardware requirements	41
2.2 Value Pack contents	41
2.2.1 HP-UX installation kit portion	42
2.2.2 Windows NT/2000/XP installation kit portion	42
2.3 Installation	43
2.3.1 Installation on an HP-UX platform	43
2.3.2 Installation on a Windows NT, 2000, or XP platform	44
2.3.3 Installation of the VPN/MPLS value pack specific reports	45
2.4 Setting up the Value Pack	45
2.4.1 Creating the mediation database	46
2.4.2 Populating the mediation database	47
2.4.3 Setting up the VPN/MPLS Value Pack	47
2.4.4 Configuring the VPN/MPLS Pack Runtime license	48
2.4.5 Running the VPN/MPLS Value Pack	48
2.4.6 Configuring the HP OpenView SQM platform to run the VPN/MPLS Value Pack	48
2.5 Cleaning up the VPN/MPLS Value Pack	49
2.6 Uninstalling the VPN/MPLS Value Pack	49
2.6.1 Uninstalling the VPN/MPLS Value Pack from HP-UX	49
2.6.2 Uninstalling the VPN/MPLS Value Pack from Windows	50

Chapter 3	53
VPN/MPLS Value Pack model design.....	53
3.1 Introduction	53
3.2 Service Designer project structure.....	53
3.2.1 Package description.....	53
3.3 Key points of the design.....	55
3.3.1 VPN Value Pack service.....	55
3.4 SQL Service Adapter	56
3.4.1 Data sources, and KPIs	56
3.4.2 SQL SAs deployed by the Value Pack	56
Chapter 4	58
Solution building guidelines	58
4.1 Introduction	58
4.2 Modifying the VPN/MPLS Value Pack SQL SA	58
Troubleshooting Guide	59
Installation	59
Service Repository Manager.....	59
VPN/MPLS Value Pack Service Adapters	60
Mediation database.....	60
Appendix A – Example of VPN Service Objectives.....	61
Appendix B – OV PI/mediation DB integration	63
Cisco SAA NRT Datapipe 2.0	63
SNMP Tables	63
OV PI DB Tables	63
Mediation DB Tables.....	64
MPLS-VPN Datapipe	64
SNMP and OV PI tables.....	64
Mediation DB Tables.....	64

Preface

This document describes the VPN/MPLS Value Pack for OpenView Service Quality Manager.

Intended Audience

This document is intended for OpenView SQM integrators, users and administrators.

Required Knowledge

It is assumed that the reader is familiar with the functionality of OpenView SQM and has previous experience of the following:

- System administration and operations
- Service level management.

It is assumed that the reader is familiar with the concepts described in the following books:

- HP OpenView Service Quality Manager Overview.
- HP OpenView Service Quality Manager Administrators Guide.
- HP OpenView Service Quality Manager Information Modeling Reference Guide.

Goal of this document is not to describe Mobile Device Management domain; it is assumed that the reader is familiar with this domain.

Software Versions

The software versions referred to in this document are specified in Section [#Software and hardware requirement](#).

Typographical Conventions

The following typographical conventions are used in this book:

Bold	New terms and to emphasize important words.
<i>Italic</i>	File names, programs, and parameters. The names of other documents referenced in this manual.
Monospace	Source code and examples of file contents. Commands that you enter on the screen. Pathnames. Keyboard key names.

Associated Documents

For a full list of OpenView SQM user documentation, refer to the *HP OpenView Service Quality Manager Product Family Introduction*.

Support

You can visit the HP OpenView support web site at:

<http://www.hp.com/managementsoftware/support>

This Web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

<http://www.managementsoftware.hp.com/passport-registration.html>.

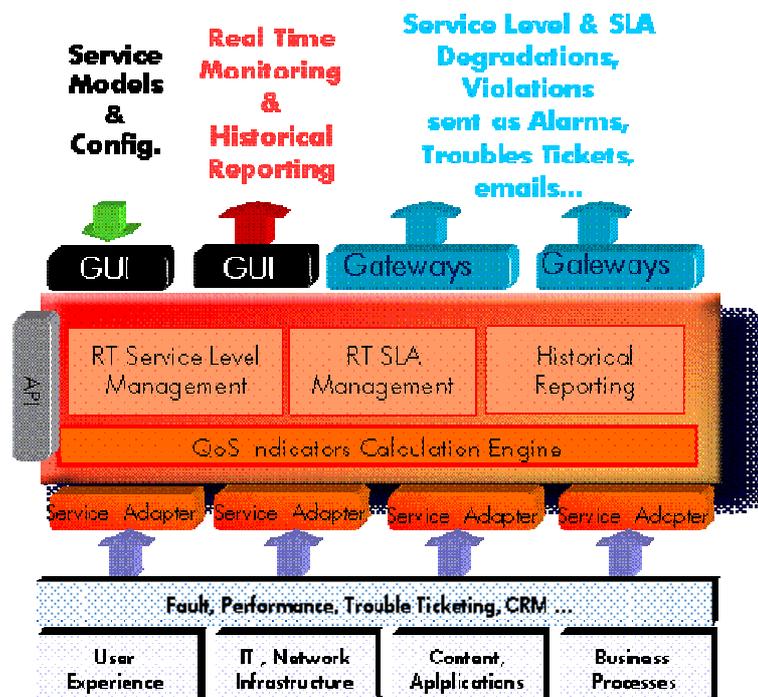
OV SQM VPN/MPLS Value Pack description

1.1 OpenView Service Quality Manager

OpenView SQM provides a complete service quality management solution. It consolidates quality indicators across all domains — telecom, IT networks, servers, and applications — providing end-to-end visibility on service quality. OpenView SQM links service quality degradations to potential effects on business, allowing network support personnel to address problems and prioritize actions proactively.

OpenView SQM monitors the service quality by aggregating information coming from all data sources, such as the network, the IT infrastructure, and the service provider's business processes. Using this information, service operators can pinpoint infrastructure problems and identify their potential affect on customers, services, and Service Level Agreements (SLAs).

Figure 1 - OpenView SQM Main Components



For a detailed description of OpenView SQM, refer to the *HP OpenView Service Quality Manager Overview*.

1.2 What is a value pack

1.2.1 Objective

The HP OpenView Service Quality Manager Value Pack software extends the value and benefits of the HP OpenView Service Quality Manager.

The objective of such value packs is to:

- Provide quick time to value with pre-configured platforms
 - Pre-implemented best practices to improve time to deploy
 - Facilitate integration with customer environment
- Leverage from projects, partners and industry standards
 - Include subject matter experts know-how and recommendations in Service Management into the product
- Improve the value delivered by the OV SQM product
 - Enrich engine capabilities with pre-defined specialized added value with customer focus
 - Provide an operational foundation for a solution that can easily be customized and extended to map customer contexts
- Secure cost of operations
 - Multi-technology / vendor catalogue of solutions and components that keep developing in line with new services.

1.2.2 OV SQM Value Pack approach

1.2.2.1 Defining Service model and KQIs

The methodology that is being used in general to develop SLA and service management solutions involves the following steps that will be re-iterated as a life-cycle to keep improving a solution:

1. Define the scope of the requirements by looking at the different use cases. For example, key operational processes as described in the eTOM (Enhanced Telecom Operation Map) from the Telemangement Forum (www.tmfcentral.com) can be used as a reference to analyze what would be the right metrics that best characterize those processes.

A use case would then be about:

- Describing standard activities of the people or group of people that will use the system
- Identifying the key metrics and policy that describe the activity of that operator (process oriented) but also that describe the behavior of the domain is operating (technology oriented).
- Defining a set of KQIs (Key Quality Indicators) that would keep giving the current and historical status of those metrics.
- Defining how those KQIs will be used by which other system(s).

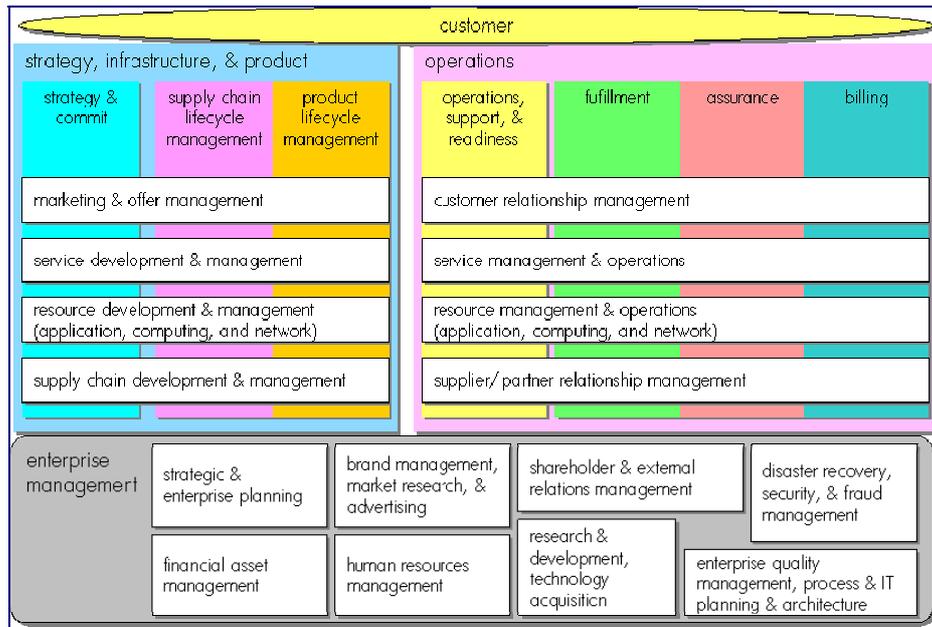


Figure 2 - eTOM – TeleManagement Forum

2. As suggested by the figure below that represents how KQIs can be defined at the top of a hierarchy of indicators, the second step is about analyzing all available data sources and defining how KQIs will be computed and / or aggregated from raw data.

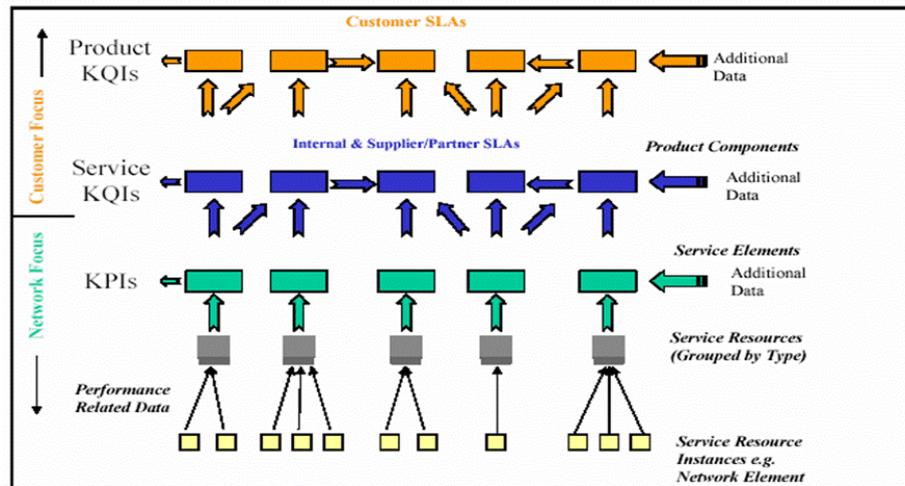


Figure 3 – Key Indicator Hierarchy (from TeleManagement Forum)

3. The 3rd step is about designing a service model that will structure the KQIs around service components into a service to provide a consistent and comprehensive view of that service.

The purpose of a service model is to accommodate many different domains, from access devices, service access points, transport networks, applications to activation and maintenance processes that all contribute to a service and its management. A service model ensures consistency between all possible views that will be provided to various interested groups or parties.

4. From the model that has been designed, the last step is about:

- Deploying real services as individual instances of the service model

- Defining service levels (“class of service”) that will correspond to the objectives of the different types of users (Customer facing people, planning people, partner managers, service managers...)
- Defining SLAs as contracts binding a particular service instance, a particular Service Level for a customer.
- Defining reports as particular and specific views about services, service availability, SLA status...
- Deploying data collection and integration to third party applications.

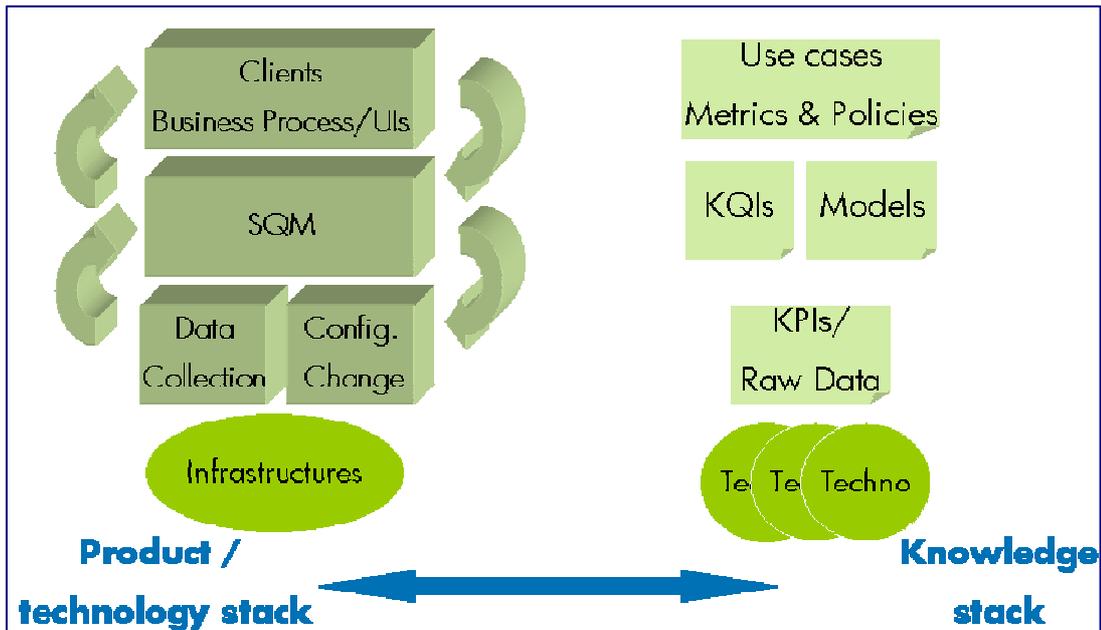


Figure 4 – OV SQM Value Pack solution analysis

1.2.2.2 Analysis of Data sources

There are many types of data sources that can be consider to provide raw data from which KQIs can be computed. Those data sources with typical raw data can be categorized as follows:

- Fault Management systems
 - Statistics on faults against (group of) Network Elements and Service Resources, based on severity, ages ...
 - State, state changes of Network Elements and Service Resources
- Performance management systems
 - Performance data from all domains involved in service delivery
 - Delivered performance against committed performance
- Mediation / usage systems
 - Information about traffic / volumes, based on users, applications and services profiles
- Probes
 - End to end performance metrics from active probes (user simulation)

- Detailed user experience (service access and utilization) from passive probes (decoding signaling protocols)
- Applications log files
 - Information about application sessions and service consumption (error codes, volumes)
- Trouble Ticketing systems
 - Information about customer outages, problem resolution delays to further compute services MTTRs, MTBFs...
- Service Provisioning systems
 - Information about service ordering and activation to further compute Mean Time To Deliver ...

The purpose of Data Source Analysis is to elect, combine and aggregate the most significant and economical metrics that are needed to compute the required KQIs.

1.2.2.3 Data collection

Integration to data sources to collect service related metrics always being a complex issue, one of the objectives of the OV SQM value pack is to provide a quick and easy way to facilitate that integration by:

- Normalizing the interfaces through a mediation database that is included in the OV SQM core product
- Providing a set of tables to load the pre-defined collected metrics.
- Providing the appropriate dedicated Service Adapters that will connect to the mediation DB and retrieve the pre-defined metrics to feed the service model.

In some cases, additional tools are included in the OV SQM value pack to help loading data from well known third party product.

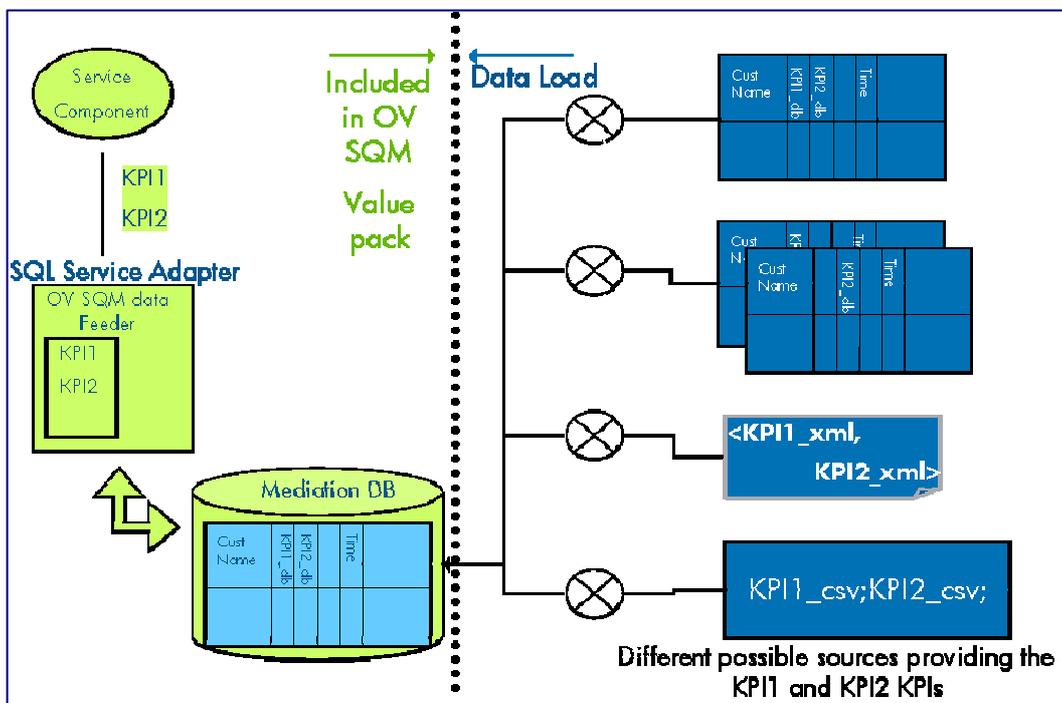


Figure 5 – OV SQM Value Pack: data collection through a mediation database

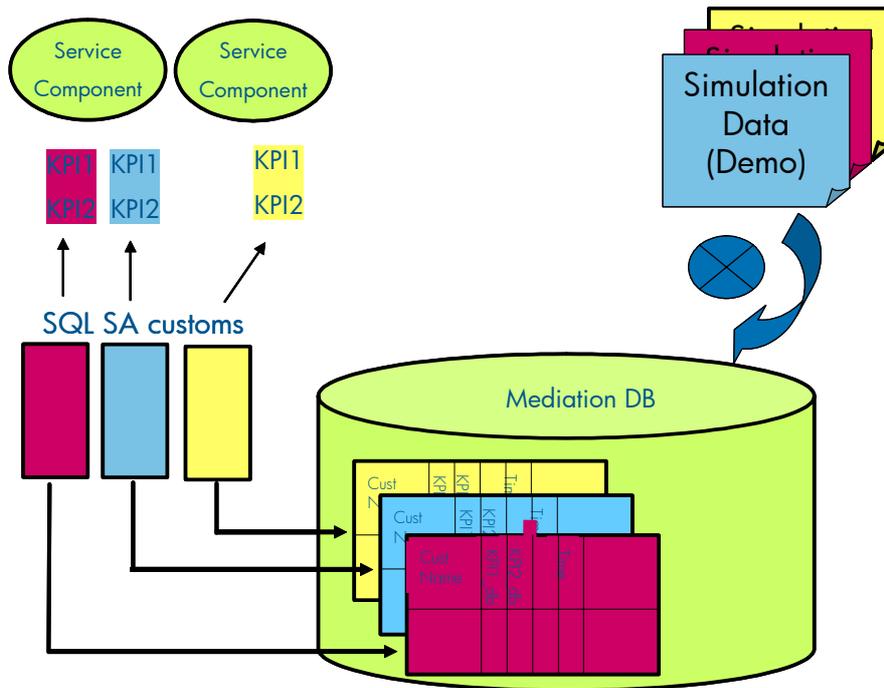
1.2.2.4 Validation / simulation data

Once the OV SQM value pack has been installed, a set of parameter values can be loaded in the different tables of the mediation database.

Those parameter values representing one day of likely data and will help:

- Validating the installation and showing live screens and reports.
- Demonstrating the product for education or training purposes

Figure 6 – OV SQM Value Pack: quick data integration



1.2.2.5 Pre-defined reports

OpenView SQM produces a variety of reports on service quality, service metrics, trends, and SLA status. OpenView SQM uses Business Objects to report the service quality information.

OpenView SQM contains a collection of predefined reports, including:

- Reports on OpenView SQM data, such as service definitions, SLAs, and service levels.
- Reports representing the mean time between faults, the mean time to repair the faults, service degradation and the availability of each service and service component.

1.2.3 Value pack typical content

An OV SQM value pack contains:

- Software add-ons to the OV SQM core products,
- Tools to install and run the value pack on top of a running OV SQM system,
- Documentation to help customers understanding the content of a value pack and augmenting the solution when needed.

1.2.3.1 Software Add-ons

- Service Model and KQIs with calculation expressions
- Service Levels and basic SLAs
- BO Universe/Reports (Optional). These are provided as examples for development of additional customized reports.
- SQL Service Adapters Customs

1.2.3.2 Tools

- Scripts and 1 day of data to populate the mediation DB
- Scripts to setup the Solution Set and to load the sample in the OV SQM repository

1.2.3.3 Documentation

- Value Pack user guide (this document)
- Service Model reference manual (HTML)
- Value Pack release notes

1.2.4 An open solution

The OV SQM value packs provide a quick way to deploy a solution in the context of a specific use case. Nevertheless, it can also be considered as a starter for a customer willing to deploy a more comprehensive and specialized solution. The solution has been design to facilitate different ways to expand the provided packaged by leveraging standard OV SQM features.

Among others, the customer will be able to:

- create new service instances, new service levels based on the pre-defined service model and KQIs to address different use cases, thanks to the graphical administration user interface,
- easily add new KQIs based on the pre-defined collected data thanks to the graphical OV SQM service designer,
- add new tables with new raw data from a new source in the mediation database and create a new Service Adapter thanks to the OV SQL SA toolkit. From this new raw parameters, new KQIs for existing or new service components can be defined,
- load real data or collection similar raw data but from another source of data by modifying the data load scripts,
- deploy new ad-hoc service adapters directly connecting to a data source, removing the mediation tables and replacing the SQL Service Adapter.
- create new reports

1.3 Introduction to the OV SQM VPN/MPLS Value Pack

1.3.1 Multiservice VPN solution

Service providers are actively seeking new, high-margin revenue streams, both to capitalize on their packet networks and to reduce customer churn by providing “sticky” services. VPN (Virtual Private Network) offerings are very attractive for

organizations with multiple sites to be interconnected. They function so well and are so cost-effective compared to their predecessors based on dedicated circuits that very competitive products can be marketed profitably by providers. This causes the VPN market to grow for both multi-site intranets and extranets which allow multiple organizations to exchange information, share services and conduct electronic business.

The most profitable voice and data services share two attributes. One is a “success-based” capital expenditure model, which accelerates return on investment (ROI) because the service provider makes most of the capital investment after acquiring a customer. The other attribute is an infrastructure that can be used to advantage to deliver multiple services, so that the service provider can increase revenue per customer over time with little incremental investment.

IP based VPNs are rapidly becoming the foundation for delivery of value-added services, and many service providers are offering customer specific applications on top of their VPN infrastructure. Emerging services such as Intranet, Extranet, e-commerce, application hosting, and multimedia applications will enable service providers to generate incremental revenue and maintain long-term competitive advantage.

Two unique and complementary VPN architectures based on IP Security (IPsec) and Multiprotocol Label Switching (MPLS) technologies are predominant foundations for delivering secure services.

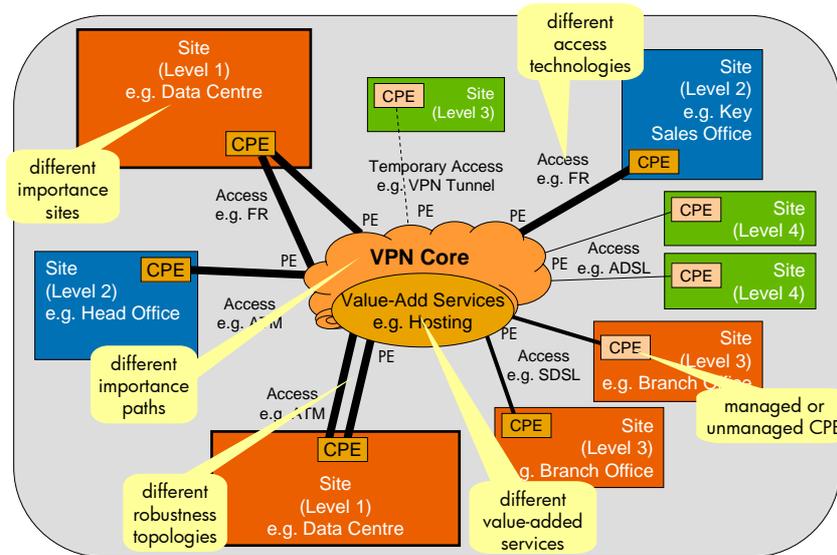
1.3.2 The advantages of MPLS-based VPN

MPLS blends the intelligence of routing with the performance of switching, providing significant benefits to networks with a native IP architecture as well as those with IP, ATM, or a mixture of other Layer 2 technologies. MPLS technology is critical for creating scalable VPNs and delivering end-to-end quality of service (QoS). It enables service providers to make efficient use of existing networks to meet future growth and support rapid fault correction of link and node failures. The inherent MPLS traffic engineering and fast reroute capabilities can significantly improve network service quality by providing the ability to maximize network bandwidth and reroute traffic rapidly in failure conditions within the core network.

MPLS technology helps to deliver highly scalable and differentiated IP services end to end with simpler configuration, management, and provisioning. When deploying MPLS technology, service providers gain immediate benefits, such as the following:

- Highly scalable routing and optimal use of network resources for any-to-any IP connectivity, encompassing multiple customers
- Integration of data, voice, and video networks in one converged infrastructure
- Differentiated, end-to-end IP services that are simpler to configure, manage, and provision and that support SLAs
- A "build once, sell many times" model for delivering network services
- The ability to provide advanced QoS features that help ensure network priority and help provide guaranteed bandwidth for mission-critical traffic
- Centralized service configuration, provisioning, and management

Customer VPN Service



4-Jan-06

HP Confidential

21

Figure 7– customer VPN Service example

1.3.3 VPN Service management challenges

Even though new technologies are the key enabler of value-added services, the complexity arises when service providers need to deliver on service quality guarantees while ensuring that the intelligent network is optimally utilized. To do this successfully, the service provider must understand service usage and service quality trends on the network, proactively detect abnormalities in service delivery that impact specific customer experience, and identify opportunities for additional revenue through better utilization of network resources.

Current OSS tools, however, typically only provide network-element oriented views and the trouble-shooting efforts tend to be network focused, domain-by-domain, and network element by network element. Using such fragmented views, a few operations experts use heuristics and their experience to perform end-customer impact analysis.

In order to proactively monitor service quality and detect/rectify abnormalities, it is necessary to collect data from a variety of sources across the service delivery chain (multiple network elements, domains, technologies and vendors), compute service quality trends, intelligently analyze service quality alerts and provide a service-oriented view for subsequent troubleshooting.

The **VPN/MPLS Value Pack** provides a logical representation that hides the complexities of the underlying network and provides a simplified customer/service oriented view is necessary for management purposes. However the quality of these services will be influenced by the underlying service components. Services are modeled at the network edge and categorized into either ‘customer- oriented’, if they can be represented on a per-customer basis, or ‘network -oriented’.

1.3.4 The OpenView SQM VPN/MPLS Value Pack scope

A service provider must be able to handle the required network scalability and complexity of a large enterprise, as well as seamless integration of the enterprise's existing network technologies and applications. Certain service-level agreement

(SLA) parameters are critical: service availability, network latency, packet loss, mean time to recovery (MTTR), and jitter.

Recently, more and more Service Providers deliver robust Classes of Service options. VPN QoS enables businesses to prioritize mission-critical traffic such as voice, video or financial transactions over non-urgent traffic such as email. Through MPLS routing technology, corporate network managers can group certain types of traffic, so that each has its own Class of Service. By prioritizing traffic, enterprises can safeguard high-priority traffic and low-priority traffic can be assigned to whatever capacity is available, while not having to increase bandwidth purely to support non-critical traffic.

The **VPN/MPLS Value Pack** has been designed to measure and monitor the Quality of Service and associated customer SLAs for five pre-defined VPN Classes of Service:

- Voice-CoS (voice)
- Video-CoS (interactive video, visio conference)
- Critical Data-CoS (call signaling, Systems Network Architecture [SNA], Oracle, PeopleSoft, SAP, Telnet, and others)
- Bulk Data-CoS (streaming video, Internet Web, IBM Lotus Workplace, Novell Groupwise, and others)
- Best Effort-CoS (Simple Mail Transfer Protocol [SMTP], FTP, Internet Web, and others)

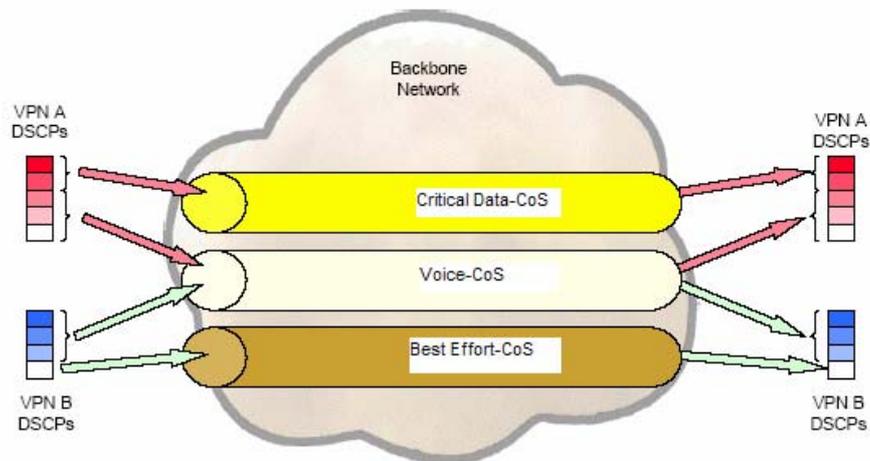


Figure 8– VPN Classes of Service

Each Class of Service has its own service level targets with respect to availability, latency, jitter, bandwidth and packet loss, giving customers a more robust solution than other services that only deliver "best effort" assurances.

Interactive voice and video conferencing	High priority, low latency, low jitter, controlled bandwidth
Client-server applications	High priority, low latency, low loss
Streaming audio and video	Medium priority, low jitter
Network control traffic	High priority, controlled bandwidth
Potentially malicious traffic	Isolated, controlled bandwidth, aggressively discarded
Circuit emulation	Guaranteed, but controlled bandwidth, low jitter
Everything else -- Best effort	Low priority, long queues

Figure 9– VPN CoS requirements

In addition to VPN CoS monitoring, the VPN/MPLS Value Pack **provides with overall performance and availability of the Service Provider and Customer network infrastructure**. Satisfactory network performance and operational support is essential to anticipate network problems and improve planning capacity.

The VPN/MPLS Value Pack comes with a set of SLA examples encompassing availability, latency, packet loss and jitter performance for each Class of Service, along with guarantees for service provision and assurance performance.

1.3.4.1 How does the OpenView SQM VPN/MPLS Value Pack work?

In this first version, the VPN/MPLS Value Pack is focusing on CISCO based VPNs.

The VPN/MPLS Value Pack collects and consolidates:

- Network Performance indicators collected between Routers and Switches (e.g. through CISCO Service Assurance Agent). Various performance metrics are measured - including round trip response time, connect time, packet loss, application performance, and inter-packet delay variance (jitter)... – providing End-to-End (CE-CE) or segmented network performance view (PE-PE, CE-PE).

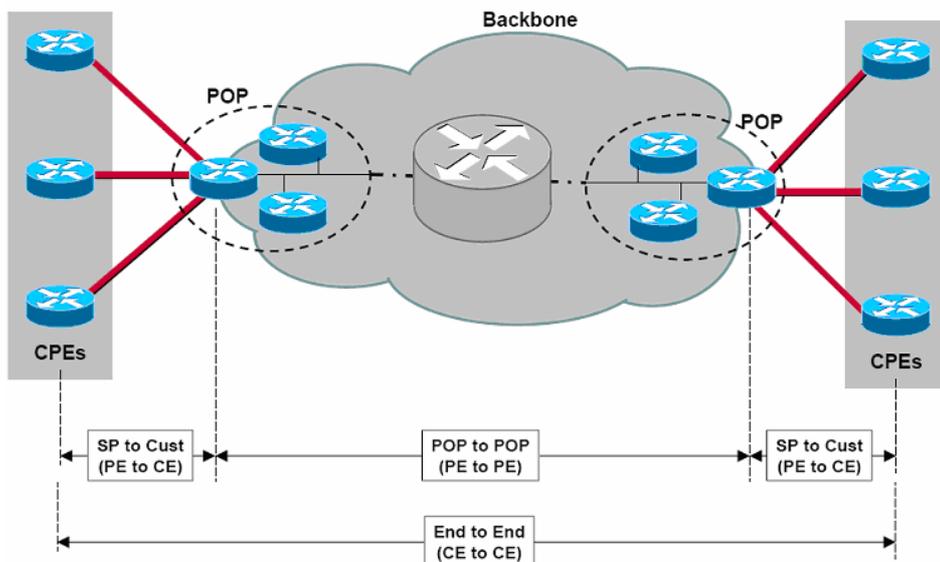


Figure 10– OV SQM solution users

- Network Fault indicators (from Fault Management system like - OpenView TeMIP, NNM, RAMS) to measure the availability of a router, tunnel or access path.
- Service Activation performance to measure efficiency when creating, deploying, modifying or expanding VPN Services (e.g. collected from OpenView Service Activator)
- Customer Care indicators – Mean Time To Repair – to measure the Service Provider operation capabilities.

These Key Performance Indicators, collected from various relevant sources, are aggregated per VPN Class of Service, with differentiated Service Level definitions.

1.3.5 Use cases

1.3.5.1 eTOM and QoS management

The following table illustrates how a solution build on a product like OV SQM help addressing different needs from different organizations in Service Provider companies.

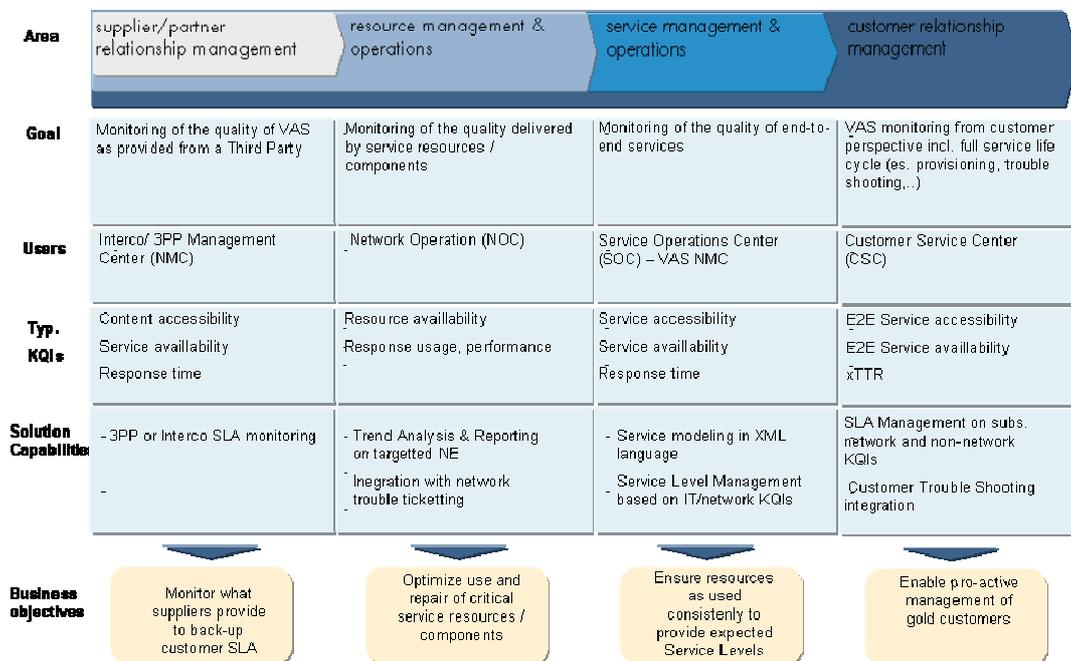


Figure 11– OV SQM solution users

1.3.6 OV SQM VPN/MPLS Value Pack software add-on

The OV SQM VPN/MPLS Value Pack comes with the following software add-on:

1.3.6.1 Service Models

The VPN/MPLS value pack contains a service model that can be deployed for the monitoring of SLAs for:

- Managing differentiated VPN Classes of Service
- Measure the efficiency and availability of the Service Provider VPN infrastructure.

The Service model combines and organizes a set of service components that all together provide an end-to-end view of VPN Service Quality offered to the customer.

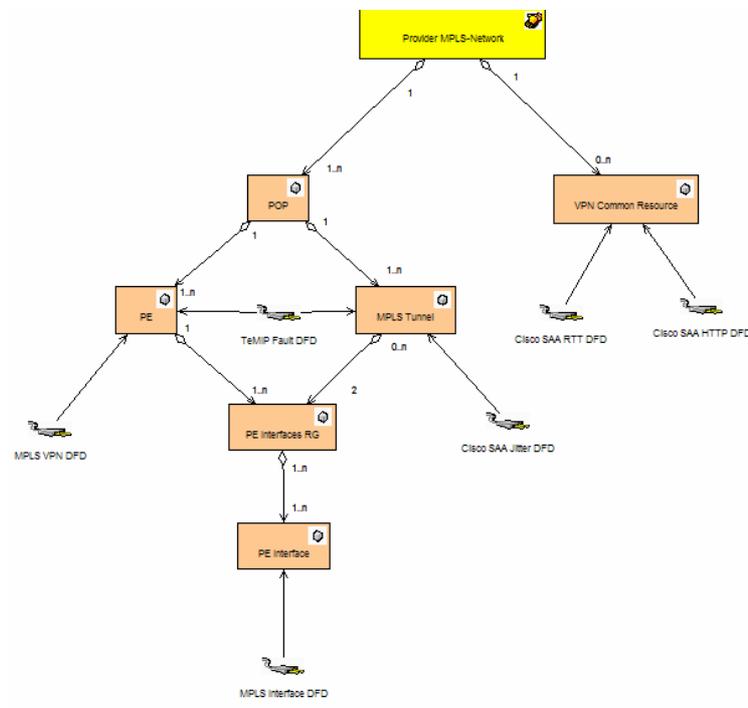
There are different Service definitions in the VPN/MPLS Value Pack to provide different aspects of the Service Quality:

1) The Service Provider Network infrastructure definition

This Service Definition provides a global view (performance and availability) of the Network Backbone. The following components are monitored:

- **POP** (Point of Presence): Access Point to the Service Provider network. \
- **MPLS Tunnel**: point to point connection between core network edge routers.
- **PE interfaces Redundancy Group**: group of PE interfaces that has a spare and one or more primary interfaces
- **PE Interface**: connection from the Provider Edge router to the Customer Edge router.
- **VPN Common Resource**: VPN Infrastructure component supporting additional VPN services (DNS, HTTP, ICPM...).

with their respective KQIs, further explained in 1.3.7.

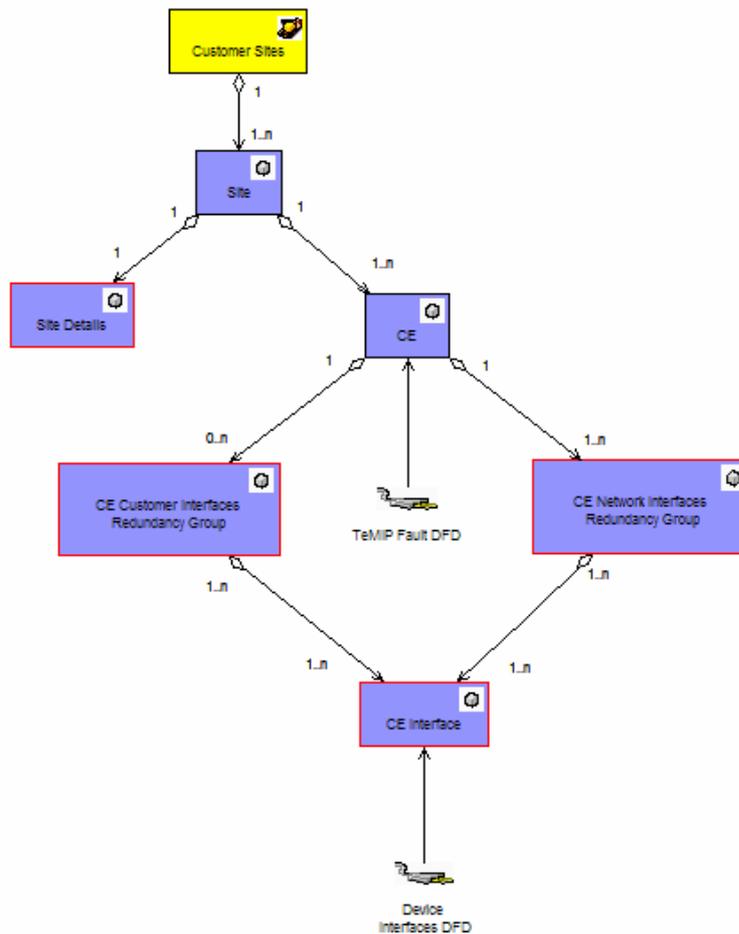


2) The Customer Network infrastructure definition

This Service Definition provides a global view (performance and availability) of the Customer VPN infrastructure. The following components are monitored:

- **VPN Site**: Customer office to be connected to other offices through WAN using the VPN.
- **Customer Edge (CE)**: Customer Edge Router equipment.
- **CE interfaces Redundancy Group**: group of CE interfaces that has a spare and one or more primary interfaces
- **CE Interface**: connection from the Customer Edge router to the Provider Edge router.

with their respective KQIs, further explained in 1.3.7.

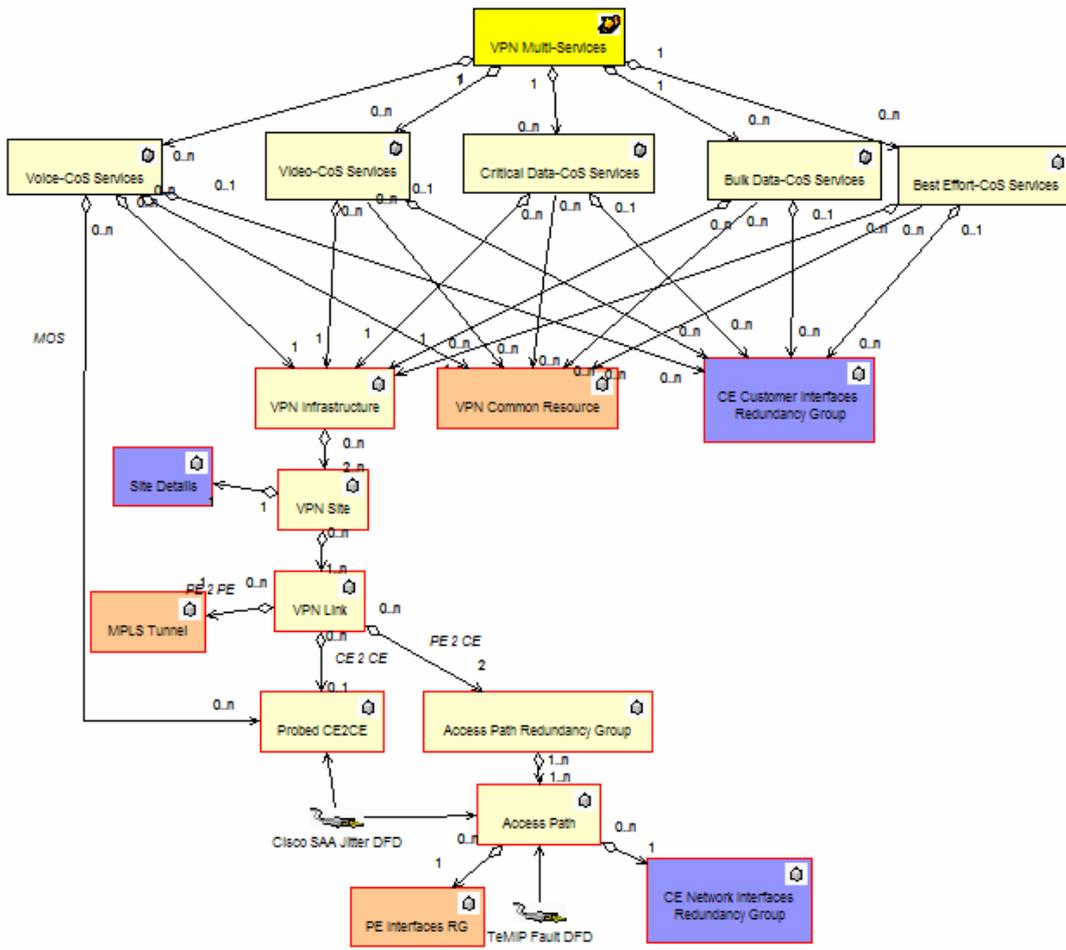


3) The VPN MultiService definition

This Service is designed to monitor the different VPN Classes of Service (CoS). The following components are monitored:

- **XXX-CoS:** - represents the service differentiation levels within the Service Provider backbone. Five VPN CoS are available (Voice, Video, Critical Data, Bulk Data, Best Effort)
- **VPN Infrastructure:** list of Service Components, or equipments entering into the delivery of a VPN Service.
- **VPN Site:** Customer Site entering into the VPN Service.
- **VPN Link:** List of VPN Links (CE-CE, CE-PE, PE-PE) entering into the VPN Service delivery.
- **Access Path Redundancy Group:** Groups of Access Paths (CE-PE), interconnecting a PE to a CE. This group contains spares and one or more primary access paths.
- **Access Path:** Connection between a group of PE Interfaces and CE Interfaces.
- **Service Desk:** service component representing VPN service help desk – customer care management.
- **Service ordering:** service component representing the VPN Service activation – measure the performance and status of VPN service activation.

with their respective KQIs, further explained in 1.3.7.



1.3.6.2 Service Levels and basic SLAs

The VPN/MPLS value pack allows the definition of Service Levels that correspond to the various use cases described below.



SLAs: examples for eTOM

operations

operations, support, & readiness	fulfillment	assurance	billing
----------------------------------	-------------	-----------	---------

customer relationship management

service management & operations

resource management & operations (application, computing, and network)

supplier/partner relationship management

	Operational SLA	Customer SLA
<ul style="list-style-type: none"> • Customer Relationship Management 	Understand general service availability, per region, per service. <i>Purpose: answer customer requests/complains</i>	Understand Service availability with agreed performance from specified locations for Gold customers. <i>Purpose: figures to compute SLA penalties.</i>
<ul style="list-style-type: none"> • Service management & operations 	General service performance against design goals (QoS policies monitoring). <i>Purpose: service capacity planning</i>	Monitor customer SLAs for each services <i>Purpose: understand business impact and prioritize service failures</i>
<ul style="list-style-type: none"> • Resource management & operations 	General resource performance against design goals (MTBF... monitoring). <i>Purpose: network capacity planning</i>	General resource performance against vendors commitments goals. <i>Purpose: NE vendor SLA monitoring</i>
<ul style="list-style-type: none"> • Supplier/partner relationship management 	Monitor content provider performance against objectives <i>Purpose: content suppliers scoring</i>	<i>Monitor wholesale – retail relationships</i> <i>Purpose: support SLAs for multi-players service value chain</i>

June 10, 2005 17

Figure 12 – OV SQM Value Pack: data collection through a mediation database

Some typical SLAs for VPN Services management are provided as examples with the value pack. Those sample SLAs with associated demo data can be used for demonstration or education purposes:

- Those scenarios demonstrate service quality management of VPN Services using HP OV Service Quality Manager VPN/MPLS value pack
- The scenarios are based on VPN Services —IP TV, IP Telephony, Internet...-- that are delivered by the Service Providers to end users
- They highlight, with simple business model scenarios, the different types of SLAs that can be deployed (customer SLAs, Service Provider internal SLA ...).
- They highlight the need for the overall integration within a full OSS environment (e.g. integration with troubleshooting and Trouble Ticket applications).

Additional example of VPN Service Levels and SLAs are provided in this document Appendix A.

Example 1: Sample business model for User/Corporate VPN Services SLA

The service provider wants to offer a SLA to large corporations who want to connect geographically dispersed branch offices, teleworkers, business partners, and customers to network resources over flexible and secure connections

The requirement to manage and report on multiple customer enterprise SLAs is necessary.

SLA will support several types of VPN Services (IP-Telephony, internet...) with associated objectives for the following metrics: service availability, delay, jitter, MTTR, MTBF...

SLA name: HP VPN Multi Service SLA

- Service: **HP Multi Services VPN**
Containing the following VPN Services:
 - Interactive Video
 - Telephony
 - Internet
- Service level metrics (D for Degradation, V for Violation), given here for a *Gold* Service Level:

Quality Metrics	Video Service	Telephony	Internet
Availability	D: 98% - V: 96%	D: 99% - V: 96%	Not Applicable
Delay	D: 200 ms - V: 400 ms	D: 150 ms - V: 400 ms	Not Applicable
Jitter	D: 4 ms - V: 10 ms	D: 2 ms - V: 4 ms	Not Applicable
Loss Rate	D: 1% - V: 2%	D: 1% - V: 3%	Not Applicable
MOS	Not Applicable	D: 3.5 - V: 2.5	Not Applicable

- SLA results need to be reported monthly

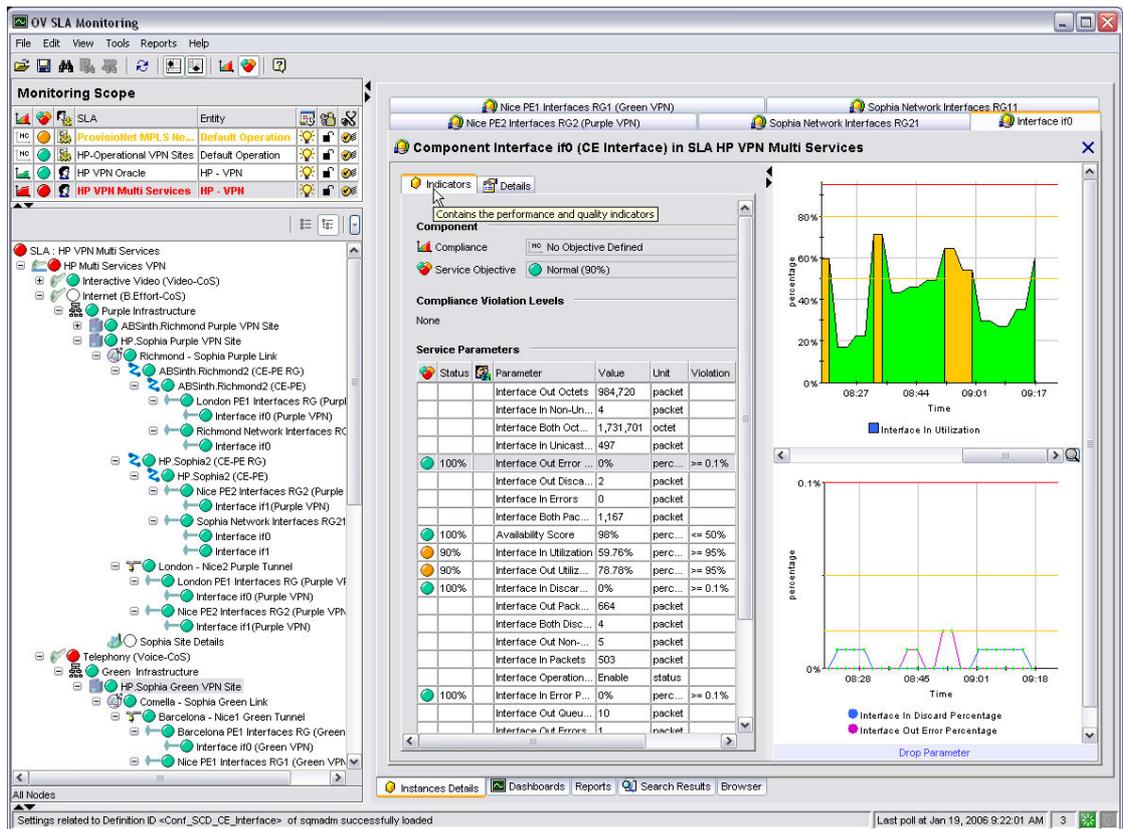


Figure 13 – OV SQM real time monitoring user interface

Example 2: Sample business model for service provider Internal SLA

- Service Provider has expectations on service quality of their overall network.
- Service provider is not bound legally to pay penalties if objectives are not met
- Business requires that an internal user SLA be set and measured to ensure customers are satisfied
- Key component layers to manage include: POP, MPLS Tunnels, PE Routers interfaces, DNS, HTTP services...
- The network SLA combines Network/equipment performance & NE availability metrics.

SLA name: VPN Oracle SLA

- Service Name: HP Oracle VPN
- Service levels
 - Availability
 - Max Jitter, Mean Jitter
 - Max Delay, Mean Delay
 - Packet Loss
 - HTTP/DNS RTT (Round Trip Time)
- SLA results need to be reported monthly and weekly



Figure 14 – OV SQM real time monitoring user interface: VPN Operational SLA

Additional examples of Customer SLAs are delivered (with associated dashboards).
 Examples of SLA objectives are also provided in this document Appendix A.

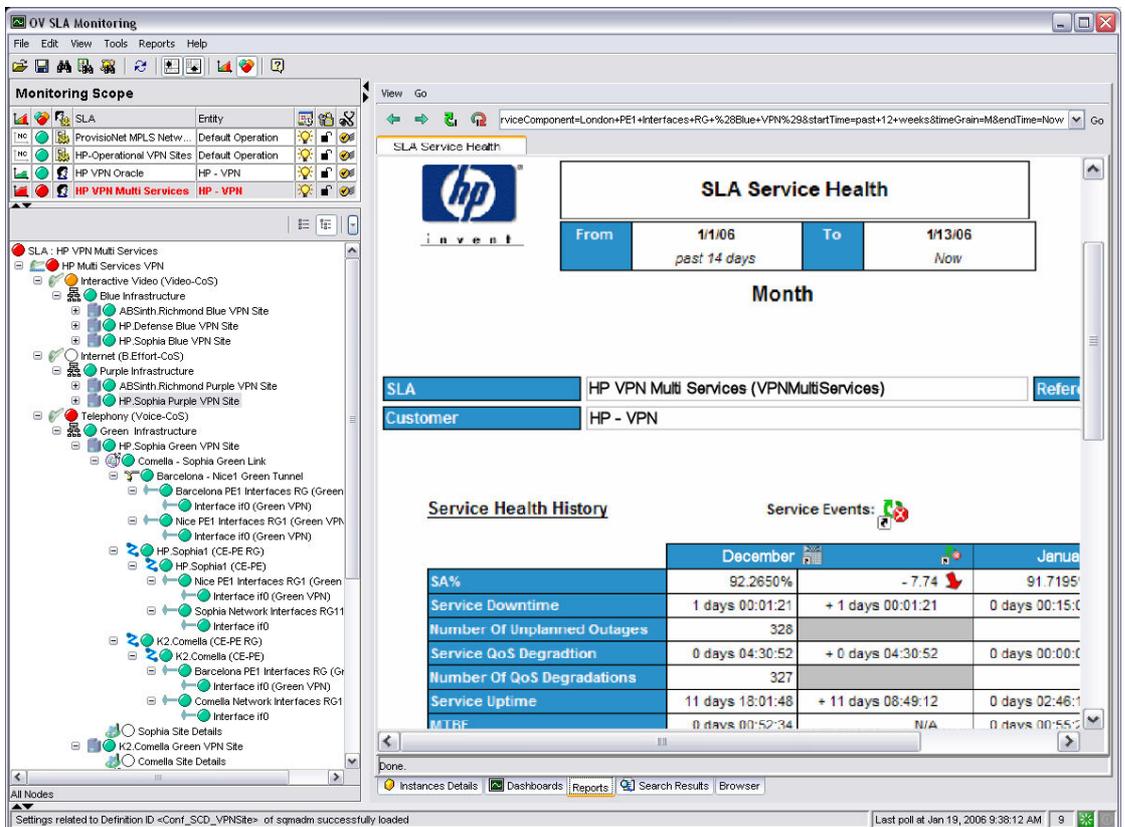


Figure 15 – OV SQM user interface: SLA off-the shelf reports

1.3.6.3 BO Universe/Reports

1.3.6.3.1. OV SQM standard reports

The OpenView Service Quality Manager produces a variety of predefined reports on historical quality of service and statistical information. Reports creation can be scheduled or on-demand.

The OV SQM reporting module relies on an open datamart that is pre-integrated with a best-of-breed reporting tool. The objective of the OV SQM Datamart is to store in one place all QoS indicators that have been collected and aggregated from all sources.

Several types of information are available in dedicated reports. Reports are organized in multiple categories:

- SLA Compliance reports
- Service Health reports
- Status History reports
- Inventory reports

1.3.6.3.2. VPN/MPLS Value pack specific reports

In this first version of the OpenView SQM VPN/MPLS Value Pack, no customized reports are delivered.

1.3.7 OV SQM VPN/MPLS Value Pack: Key Quality Indicators

1.3.7.1 Need for high level Key Quality Indicators

Ideally, only a small set of end-to-end historical and real time Key Quality Indicators should be enough to get an instant view of services and SLA health and status. Those high level KQIs are composite indicators build from primary performance or quality indicators each of them coming from various domains and/or different geographies.

Even though the primary input indicators would be different as coming from different contexts (technology, vendors...), whatever the use case, the same categories of KQIs can be defined to characterize the different aspects of VPN Services management. Those KQIs will be either Technical KQIs or Non Technical KQIs.

Technical KQIs:

KQIs	Comment
Availability	VPN Service Availability - overall Operational Status of the service
Packet Loss	Number of packets discarded due to errors divided by the total number of packets transmitted.
Delay	Time it takes a packet to reach its destination from the time that the packet is generated at the source. This delay includes retransmission delay due to packet loss.
Jitter	Variation in the time between packets arriving, caused by network congestion, timing drift, or route changes.
Throughput	Measure in bits per second of the successful traffic being transmitted

	between two devices.
MOS (mean Opinion Score)	Numerical measure of the quality of human speech at the destination end of the circuit, using a 1-5 scale.

Non Technical KQIs

Service provisioning (Time To Activate) KQIs refer to parameters attached to the processed involved in providing services or changing services to customers.

Service Repair (Mean Time To Repair) KQIs refer to parameters attached to the processed involved in restoring services to customers.

1.3.7.2 Data sources

The following tables list most parameters that are collected from the various systems mentioned as data sources in the service models:

- VPN/MPLS Device: collects information on VPN devices.
- VPN/MPLS Interfaces: collects information on MPLS-type interfaces.
- CISCO SAA Jitter: collects latest information of jitter operations.
- CISCO SAA RTT: collects latest information of all other SAA operations (echo, udpEcho, tcpConnect, dns, dls, dhcp, ftp).
- CISCO SAA HTTP: collects latest information of HTTP operations.
- TeMIP Fault: collects service impact faults.
- Service Activation: collects VPN Service Activation performance and status.
- Service Desk: collects customer care efficiency.

1.3.7.2.1. VPN/MPLS Device KPIs

Name	Data Type	Stereotype	Comment
ConfiguredVrfs	Integer	Parameter	[RVpnDeviceStats] The number of VRFs which are configured on this node.
ActiveVrfs	Integer	Parameter	[RVpnDeviceStats] The number of VRFs which are active on this node. That is, those VRFs whose corresponding mplsVpnVrfOperStatus object value is equal to operational (1).
ConnectedInterfaces	Integer	Parameter	[RVpnDeviceStats] Total number of interfaces connected to a VRF.
VpnVrfOperStatus	Integer	Parameter	[RVpnVrf] Denotes whether a VRF is operational or not. A VRF is up(1) when at least one interface associated with the VRF, which ifOperStatus is up(1). A VRF is down(2) when: a) There does not exist at least one interface whose ifOperStatus is up(1), or b) There are no interfaces associated with the VRF.
VpnVrfActiveInterfaces	Integer	Parameter	[RVpnVrf] Total number of interfaces connected to this VRF with ifOperStatus = up(1).
VpnVrfAssociatedInterfaces	Integer	Parameter	[RVpnVrf] Total number of interfaces connected to this VRF (independent of ifOperStatus type).
SecIllegalLblVltns	Integer	Parameter	[RVpnVrf] Indicates the number of illegally received labels on this VPN/VRF.
RoutesAdded	Integer	Parameter	[RVpnVrf] Indicates the number of routes

			added to this VPN/VRF over the course of its lifetime.
RoutesDeleted	Integer	Parameter	[RVpnVrf] Indicates the number of routes removed from this VPN/VRF.
CurrNumRoutes	Integer	Parameter	[RVpnVrf] Indicates the number of routes currently used by this VRF.

1.3.7.2.2. VPN/MPLS Interface KPIs

Name	Data Type	Stereotype	Comment
ifMtu	Integer	Parameter	[SRIRDevPorts] The size of the largest datagram which can be sent/received on the interface, specified in octets.
ifOperStatus	Integer	Parameter	[SRIRDevPorts] The current operational state of the interface, up(1), down(2) or testing(3).
ifLastChange	AbsTime	Parameter	[SRIRDevPorts] The value of sysUpTime at the time the interface entered its current operational state.
ifInOctets	Integer	Parameter	[SRIRDevPorts] The total number of octets received on the interface including framing characters.
ifInUcastPkts	Integer	Parameter	[SRIRDevPorts] The number of subnetwork-unicast packets delivered to a higher-layer protocol.
ifInNUcastPkts	Integer	Parameter	[SRIRDevPorts] The number of non-unicast (ie subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol.
ifInDiscards	Integer	Parameter	[SRIRDevPorts] The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol.
ifInErrors	Integer	Parameter	[SRIRDevPorts] The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.
ifInUnknownProtos	Integer	Parameter	[SRIRDevPorts] The number of packets received via the interface which were discarded because of an unknown or unsupported protocol.
ifOutOctets	Integer	Parameter	[SRIRDevPorts] The total number of octets transmitted out of the interface including framing characters.
ifOutUcastPkts	Integer	Parameter	[SRIRDevPorts] The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address including those that were discarded or not sent.
ifOutNUcastPkts	Integer	Parameter	[SRIRDevPorts] The total number of packets that higher-level protocols requested be transmitted to a non-unicast (ie a subnetwork-broadcast or subnetwork-multicast) address including those that were discarded or not sent.
ifOutDiscards	Integer	Parameter	[SRIRDevPorts] The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted.
ifOutErrors	Integer	Parameter	[SRIRDevPorts] The number of outbound packets that could not be transmitted because of errors.
ifOutQLen	Integer	Parameter	[SRIRDevPorts] The length of the output packet queue (in packets).
sysUpTime	Integer	Parameter	[SRIRDevPorts] The time (in hundredths of a second) since the network management portion of the system was last re-initialized.
AvailabilityScore	Float	Parameter	[SRIRDevPorts] Calculated availability score based on ifLastChange and sysUpTime.

ifInPkts	Integer	Parameter	[SRIRDevPorts] The total number of unicast and non-unicast packets delivered to a higher-layer protocol.
ifOutPkts	Integer	Parameter	[SRIRDevPorts] The total number of unicast and non-unicast packets that higher-level protocols requested be transmitted to a subnetwork-unicast address including those that were discarded or not sent.
ifBothPkts	Integer	Parameter	[SRIRDevPorts] The total number of inbound and outbound packets (ifInPkts+ifOutPkts).
ifBothDiscards	Integer	Parameter	[SRIRDevPorts] The total number of inbound and outbound packets discarded (ifInDiscards+ifOutDiscards).
ifBothOctets	Integer	Parameter	[SRIRDevPorts] The total number of inbound and outbound octets (ifInOctets+ifOutOctets).
ifBothErrors	Integer	Parameter	[SRIRDevPorts] The total number of inbound and outbound packets containing errors (ifInErrors+ifOutErrors).
DuplexSnapshot	Integer	Parameter	[SRIRDevPorts] The interface duplex.
SpeedSnapshot	Integer	Parameter	[SRIRDevPorts] The provisioned interface speed.
InSpeedSnapshot	Integer	Parameter	[SRIRDevPorts] The provisioned inbound interface speed.
OutSpeedSnapshot	Integer	Parameter	[SRIRDevPorts] The provisioned outbound interface speed.
ifInUtil	Float	Parameter	[SRIRDevPorts] The calculated inbound interface utilization.
ifInDiscardPct	Float	Parameter	[SRIRDevPorts] The inbound discard rate as a percentage of total inbound packets.
ifInErrorPct	Float	Parameter	[SRIRDevPorts] The inbound error rate as a percentage of total inbound packets.
ifOutUtil	Float	Parameter	[SRIRDevPorts] The calculated outbound interface utilization.
ifOutDiscardPct	Float	Parameter	[SRIRDevPorts] The outbound discard rate as a percentage of total outbound packets.
ifOutErrorPct	Float	Parameter	[SRIRDevPorts] The outbound error rate as a percentage of total outbound packets.
InLabelsUsed	Integer	Parameter	[RMplsInterfacePerf] This value indicates the specific number of labels that are in use at this point in time on this interface in the incoming direction.
FailedLabelLookup	Integer	Parameter	[RMplsInterfacePerf] This value indicates the number of labeled packets that have been received on this interface and were discarded because there were no matching entries found for them in mplsInSegmentTable.
OutLabelsUsed	Integer	Parameter	[RMplsInterfacePerf] Indicates the number of top-most labels in the outgoing label stacks that are in use at this point in time on this interface.
OutFragments	Integer	Parameter	[RMplsInterfacePerf] This variable indicates the number of outgoing MPLS packets that required fragmentation before transmission on this interface.

1.3.7.2.3. CISCO SAA Jitter KPIs

Name	Data Type	Stereotype	Comment
NumOfRTT	Integer	Parameter	The number of RTT's that were successfully measured.
RTTSum	Integer	Parameter	The sum of Jitter RTT's that are successfully measured.
RTTSum2	Integer	Parameter	The sum of squares of RTT's that are

			successfully measured.
MinRTT	Integer	Parameter	The minimum of RTT's that were successfully measured.
MaxRTT	Integer	Parameter	The maximum of RTT's that were successfully measured.
MinOfPosSD	Integer	Parameter	The minimum of all positive jitter values from packets sent from source to destination.
MaxOfPosSD	Integer	Parameter	The maximum of all positive jitter values from packets sent from source to destination.
NumOfPosSD	Integer	Parameter	The sum of number of all positive jitter values from packets sent from source to destination.
SumOfPosSD	Integer	Parameter	The sum of RTT's of all positive jitter values from packets sent from source to destination.
Sum2PosSD	Integer	Parameter	The sum of square of RTT's of all positive jitter values from packets sent from source to destination.
MinOfNegSD	Integer	Parameter	The minimum of absolute values of all negative jitter values from packets sent from source to destination.
MaxOfNegSD	Integer	Parameter	The maximum of absolute values of all negative jitter values from packets sent from source to destination.
NumOfNegSD	Integer	Parameter	The sum of number of all negative jitter values from packets sent from source to destination.
SumOfNegSD	Integer	Parameter	The sum of all negative jitter values from packets sent from source to destination.
Sum2NegSD	Integer	Parameter	The sum of square of RTT's of all negative jitter values from packets sent from source to destination.
MinOfPosDS	Integer	Parameter	The minimum of all positive jitter values from packets sent from destination to source.
MaxOfPosDS	Integer	Parameter	The maximum of all positive jitter values from packets sent from destination to source.
NumOfPosDS	Integer	Parameter	The sum of number of all positive jitter values from packets sent from destination to source.
SumOfPosDS	Integer	Parameter	The sum of RTT's of all positive jitter values from packets sent from destination to source.
Sum2PosDS	Integer	Parameter	The sum of square of RTT's of all positive jitter values from packets sent from destination to source.
MinOfNegDS	Integer	Parameter	The minimum of absolute values of all negative jitter values from packets sent from destination to source.
MaxOfNegDS	Integer	Parameter	The maximum of absolute values of all negative jitter values from packets sent from destination to source.
NumOfNegDS	Integer	Parameter	The sum of number of all negative jitter values from packets sent from destination to source.
SumOfNegDS	Integer	Parameter	The sum of all negative jitter values from packets sent from destination to source.
Sum2NegDS	Integer	Parameter	The sum of square of RTT's of all negative jitter values from packets sent from destination to source.
PacketLossSD	Integer	Parameter	The number of packets lost when sent from source to destination.
PacketLossDS	Integer	Parameter	The number of packets lost when sent from

			destination to source.
PacketOutOfSeq	Integer	Parameter	The number of packets arrived out of sequence.
PacketMIA	Integer	Parameter	The number of packets that are lost for which we cannot determine the direction.
PacketLateArvl	Integer	Parameter	The number of packets that arrived after the timeout.
Sense	Enum	Parameter	An application specific sense code for the completion status of the latest Jitter RTT operation, which could be: ok(1), disconnected(2), overThreshold(3), timeout(4), busy(5), notConnected(6), dropped(7), sequenceError(8), verifyError(9), applicationSpecific(10), dnsServerTimeout(11), tcpConnectTimeout(12), httpTransactionTimeout(13), dnsQueryError(14), httpError(15) or error(16).
MOS	Integer	Parameter	The MOS value for the latest jitter operation.
ICPIF	Integer	Parameter	Represents ICPIF value for the latest jitter operation.

1.3.7.2.4. CISCO SAA RTT KPIs

Name	Data Type	Stereotype	Comment
CompletionTime	Integer	Parameter	The completion time of the latest RTT operation successfully completed (in milliseconds)
Sense	Enum	Parameter	A sense code for the completion status of the latest RTT operation., which could be: ok(1), disconnected(2), overThreshold(3), timeout(4), busy(5), notConnected(6), dropped(7), sequenceError(8), verifyError(9), applicationSpecific(10), dnsServerTimeout(11), tcpConnectTimeout(12), httpTransactionTimeout(13), dnsQueryError(14), httpError(15) or error(16).

1.3.7.2.5. CISCO SAA HTTP KPIs

Name	Data Type	Stereotype	Comment
RTT	Integer	Parameter	Round Trip Time taken to perform HTTP operation. This value is the sum of DNSRTT, TCPConnectRTT and TransactionRTT.
DNSRTT	Integer	Parameter	Round Trip Time taken to perform DNS query within the HTTP operation. If an IP Address is specified in the URL, then DNSRTT is 0.
TCPConnectRTT	Integer	Parameter	Round Trip Time taken to connect to the HTTP server.
TransactionRTT	Integer	Parameter	Round Trip Time taken to download the object specified by the URL.
MessageBodyOctets	Integer	Parameter	The size of the message body received as a response to the HTTP request.
Sense	Enum	Parameter	An application specific sense code for the completion status of the latest RTT operation, which could be: ok(1), disconnected(2), overThreshold(3), timeout(4), busy(5), notConnected(6), dropped(7), sequenceError(8), verifyError(9), applicationSpecific(10), dnsServerTimeout(11), tcpConnectTimeout(12), httpTransactionTimeout(13), dnsQueryError(14), httpError(15) or error(16).

1.3.7.2.6. TeMIP Fault KPIs

Name	Data Type	Stereotype	Comment
CommunicationsAlarm Number of Outstanding Alarms	Integer	Parameter	Number of Outstanding Service Impacting Alarms
CommunicationsAlarm Problem Severity	Integer	Parameter	Severity of the problem
EquipmentAlarm Number of Outstanding Alarms	Integer	Parameter	Number of Outstanding Service Impacting Alarms
EquipmentAlarm Problem Severity	Integer	Parameter	Severity of the problem

1.3.7.2.7. Service Activation KPIs

Name	Data Type	Stereotype	Comment
NbFltyActiv	Float	Parameter	Number of faulty activations.
NbOfActiv	Float	Parameter	Number of activations.
NbOfModif	Float	Parameter	Number of modifications.
TimeToActivate	Float	Parameter	Time between order and resource activated.

1.3.7.2.8. Service Desk KPIs

Name	Data Type	Stereotype	Comment
MTTR	Integer	Parameter	Mean Time To Repair metric, based on Average open duration metric
Average actual duration	Integer	Parameter	Average actual duration
Average duration over deadline	Integer	Parameter	Average duration over deadline
Average open duration	Integer	Parameter	Average open duration
Maximum duration over deadline	Integer	Parameter	Maximum duration over deadline
Number of closed items	Integer	Parameter	Number of closed items
Number of open items	Integer	Parameter	Number of open items
Number of overdue items	Integer	Parameter	Number of overdue items

1.3.7.3 Additional Computed /aggregated Key Quality Indicators

Example of Key Quality Indicators as aggregated for a MPLS Tunnel (as a collection of 2 PEs) :

Name	Data Type	Stereotype	Comment
Throughput	Float	Parameter	Throughput
Mean jitter SourceToDestination	Float	Parameter	Mean jitter value from destination to source calculated from the jitter parameters provided by SAA. Value in milliseconds.
Mean jitter	Float	Parameter	Mean jitter value from destination to source

DestinationToSource			calculated from the jitter parameters provided by SAA. Value in milliseconds.
Max jitter DestinationToSource	Float	Parameter	The maximum jitter measure for Destination to Source direction.
Max jitter SourceToDestination	Float	Parameter	The maximum jitter measure for Source to Destination direction.
One way latency	Float	Parameter	The half Round Trip Time.
Both directions average Jitter	Float	Parameter	The average between the mean jitter Source to Destination and the mean jitter Destination to Source.
Packet loss rate	Float	Parameter	Packet Loss Rate. An average of the Packet Loss in both directions S to D & D to S.
Both directions Max jitter	Float	Parameter	The maximum between the max jitter Source to Destination and the max jitter Destination to Source.
Total number of packets drop	Integer	Parameter	The total number of packets dropped.
Both directions packets loss	Float	Parameter	The number of loss packet loss, computed on Source to Destination packet loss and Destination to Source packet loss.
Packet Drop Rate	Float	Parameter	Rate of dropped packet, formula being number of dropped packets/total number of packets
Current # Critical Equipment Alarm	Integer	Parameter	Current number of Critical Equipment Alarm
Current # Major Equipment Alarm	Integer	Parameter	Current number of Major Equipment Alarm
Current # Critical Communication Alarm	Integer	Parameter	Current number of Critical Communication Alarm
Current # Major Communication Alarm	Integer	Parameter	Current number of Major Communication Alarm

1.3.8 Solution in use

The following sections show how HP OpenView Service Quality Manager can help its users to be more efficient in their daily activities monitoring network resources, end-to-end services, or customer SLAs.

1.3.8.1 Service Level management

The HP OpenView Service Quality Manager real-time monitoring interface provides a quick and simple way to investigate service problems, as shown in the following figure.

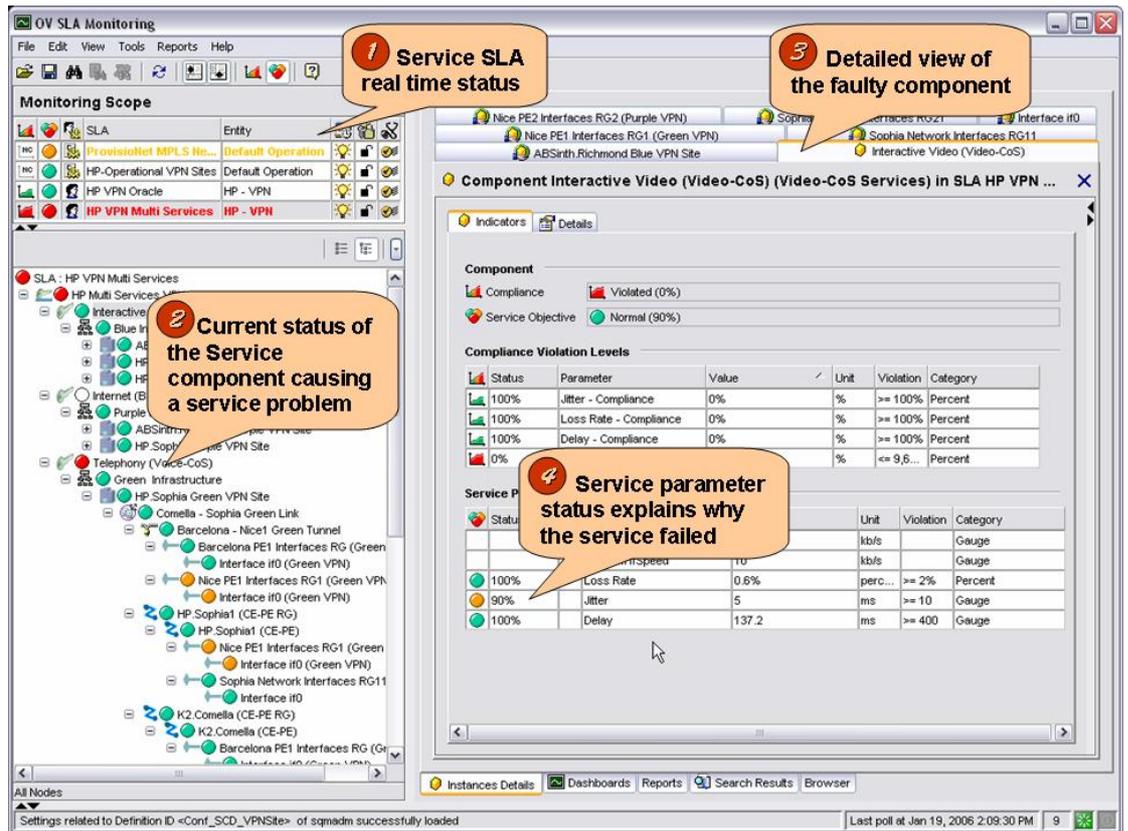


Figure 16 – HP OpenView Service Quality Manager interface

This window displays the following information, as numbered in the figure above:

The Monitoring Scope window showing the current status of each monitored service.

1. By selecting one of the monitored SLAs, Operators see a detailed view of that SLA or service. They can define filters to focus on faulty (degraded or violated) components that impact the SLA or service status.
2. A detailed view of the selected component lists the current values of all indicators attached to that component.
3. Parameter statuses, enabling the Operator to see which conditions caused a service failure.

Once they have used HP OpenView Service Quality Manager to find out which service component is causing the problem, the Operator can use a contextual menu to identify each SLA potentially impacted by the service resource concerned, as shown in the following figure.

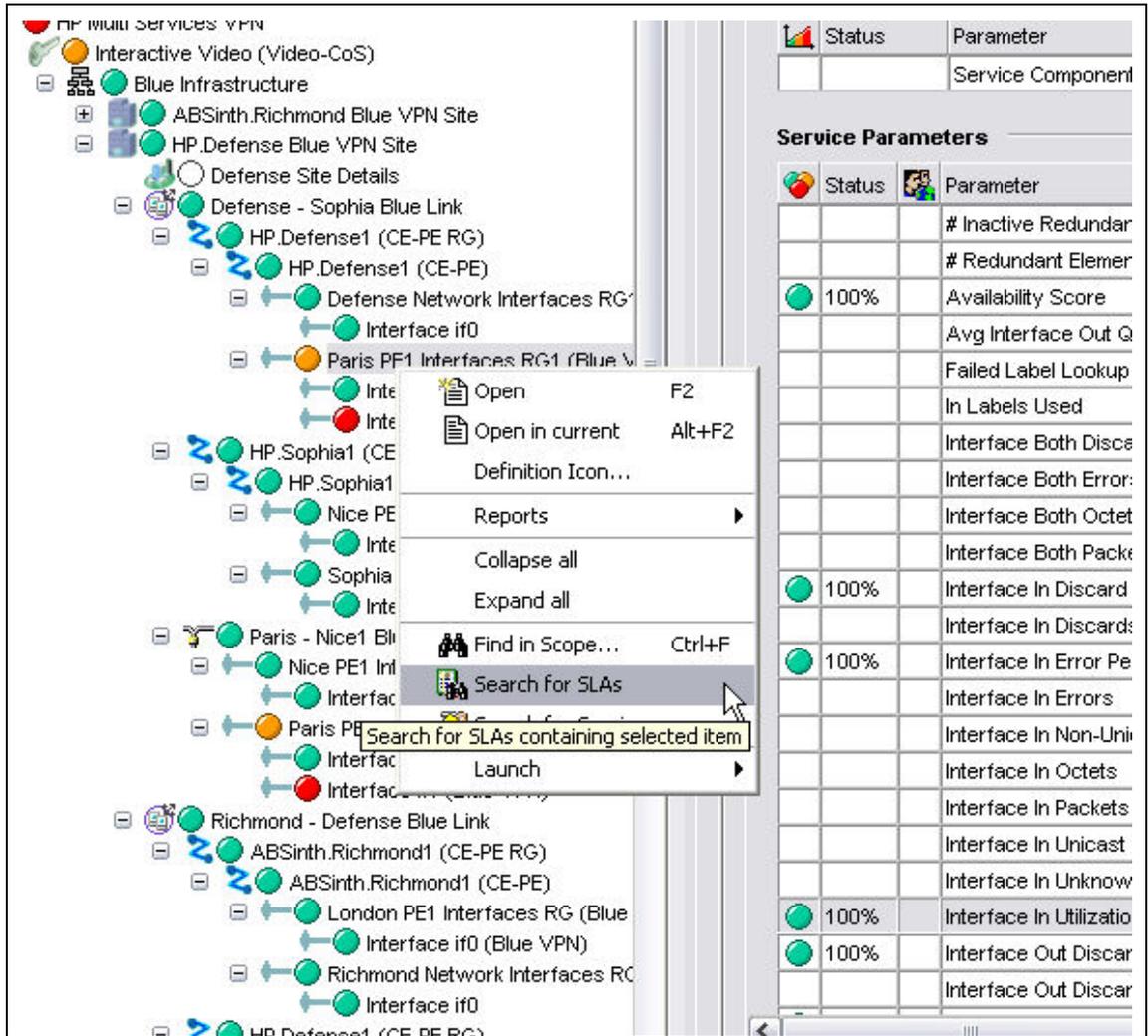


Figure 17 – Service component tree contextual menu

1.3.8.2 Customer SLA management

HP OpenView Service Quality Manager enables those interfacing with customers to meet customer expectations better, by giving them a comprehensive, end-to-end view representing how those customers perceive service utilization, as shown in the following figure.

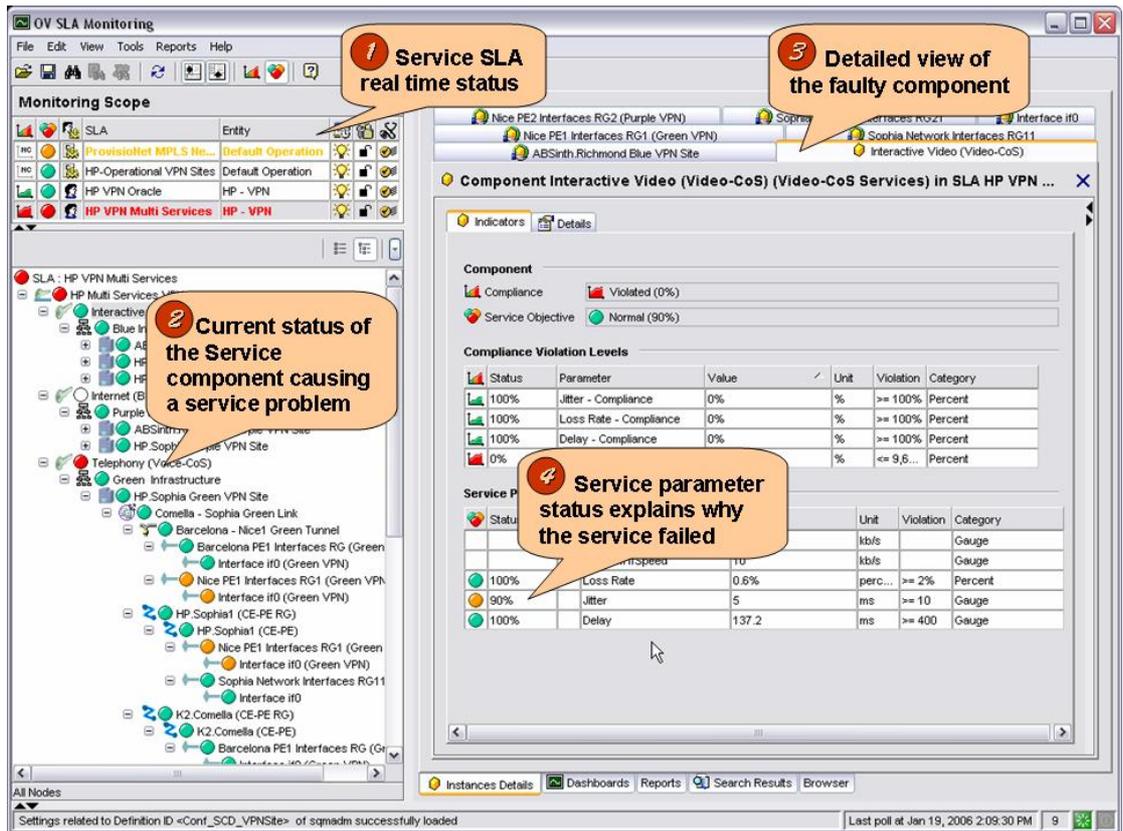


Figure 18 – Monitoring Scope example window

The Monitoring Scope window displays the following information, as numbered in the figure above:

1. The Monitoring Scope window shows the current status of each monitored SLA.
2. A detailed window of the service linked to that customer SLA lists all Key Quality Indicators included in the customer contract and best describing the customer experience regarding the service.
3. The current status of each indicator relative to the objectives set for that customer helps the Operator to understand what problem the service user is currently facing: in this example, there is problem Jitter degradation on the Interactive Video Service.

The Operator can use various dashboards to investigate the details behind the *VPN Multi Services* aggregate indicator, by looking at the history and current status of each primary indicator used in calculating the high-level *VPN Multi Services* aggregate indicator, as shown in the following figure.

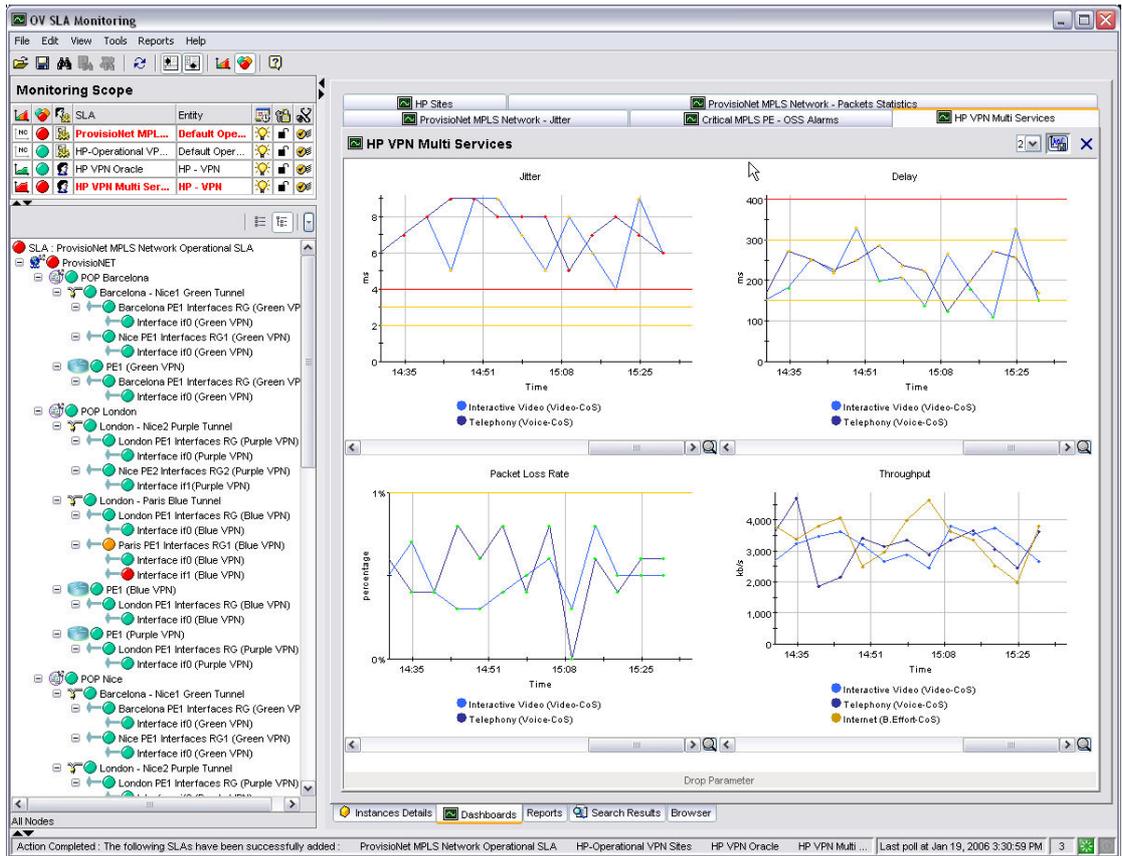


Figure 19 – VPN Multi Services dashboard

Operators can display predefined reports providing summary information on service availability over various timescales. End users can display similar reports using a Web-based interface, as shown in the following figure.

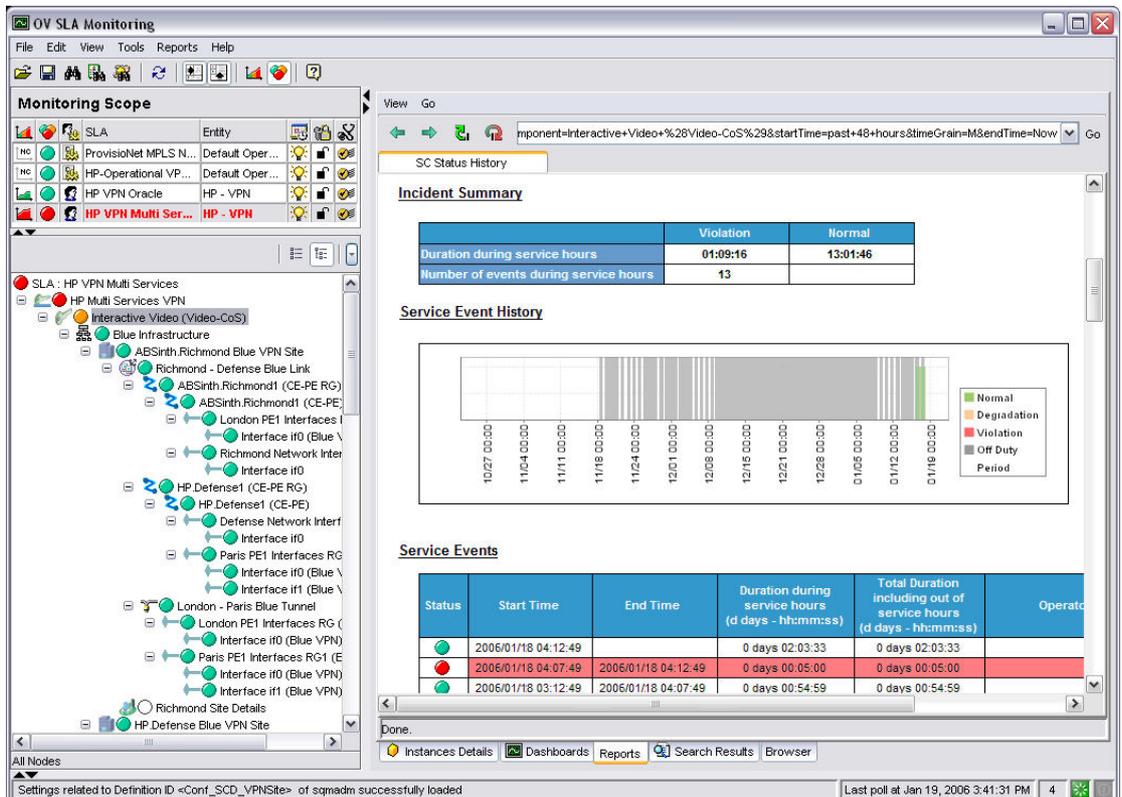


Figure 20 – OV SQM SLA reports window

1.3.8.3 Resource management

HP OpenView Service Quality Manager provides resource managers and network planners with a better understanding of how service resources are performing against expectations, and enables them to proactively plan resource updates to meet business expectations while reducing the effect on the customer experience.

Its Monitoring Scope window shows the current status of each SLA attached to specific monitored resources, as shown in the following figure.

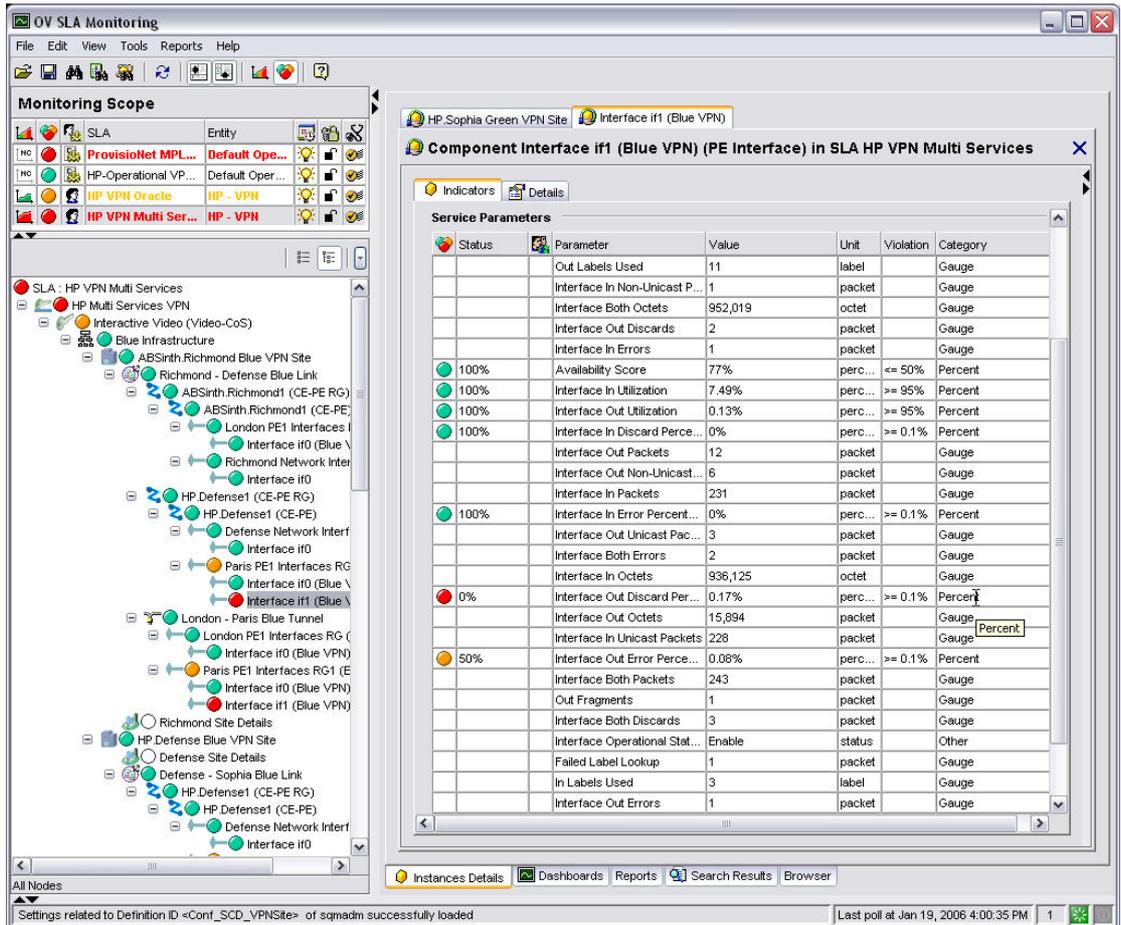


Figure 21 – Service component parameter window

HP OpenView Service Quality Manager’s real-time monitoring dashboards help you analyze or aggregate complex indicators (such as In Utilization or Out Utilization) through changes in the primary indicators (such as CE, for example), as shown in the following figure.

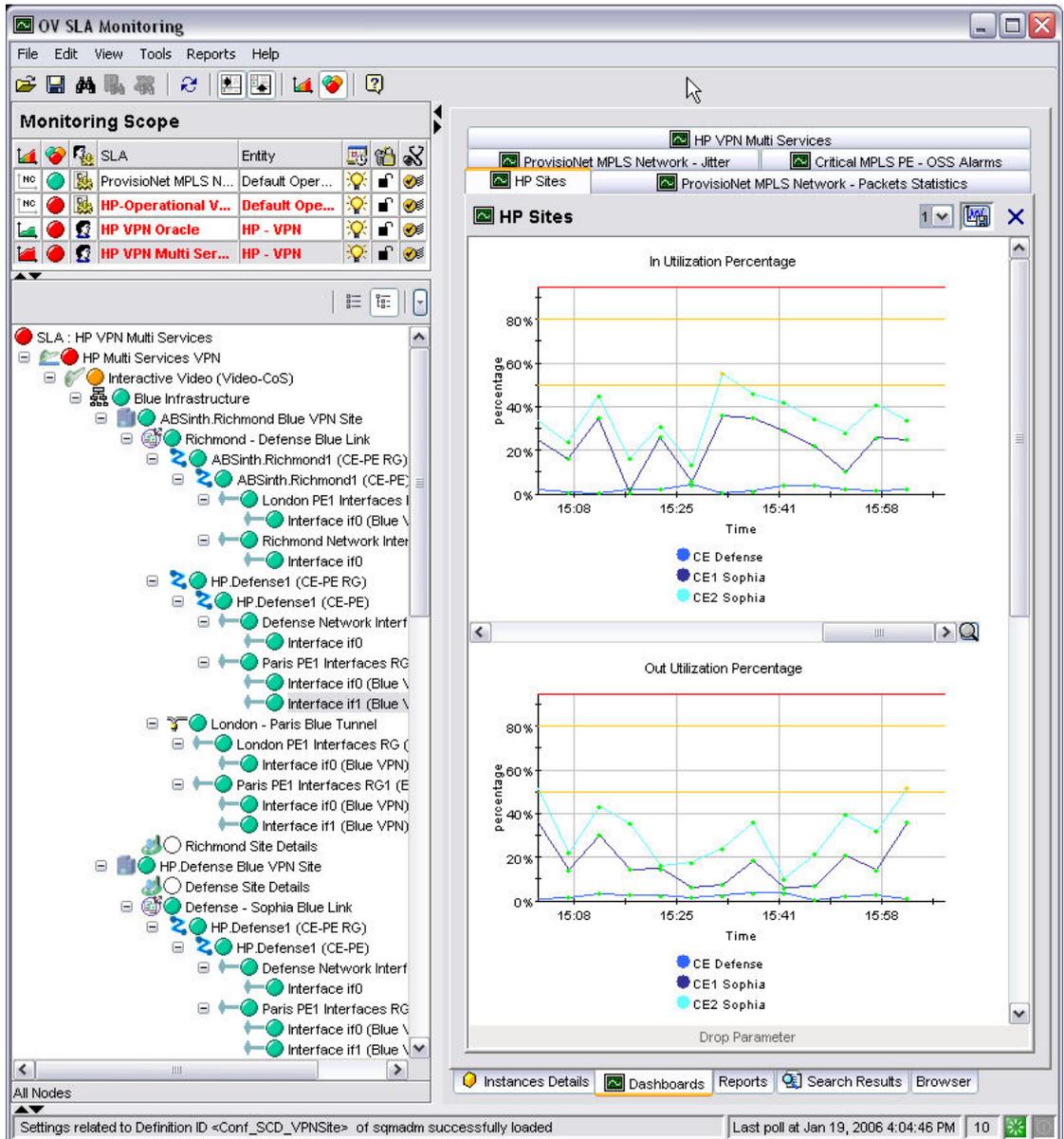


Figure 22 – In and Out Utilization window

Predefined historical reports provide a comprehensive view of the Quality of Service each service component provides. HP OpenView Service Quality Manager’s real-time monitoring interface enables you to access and navigate through reports directly from the service tree structure, as shown in the following figure.

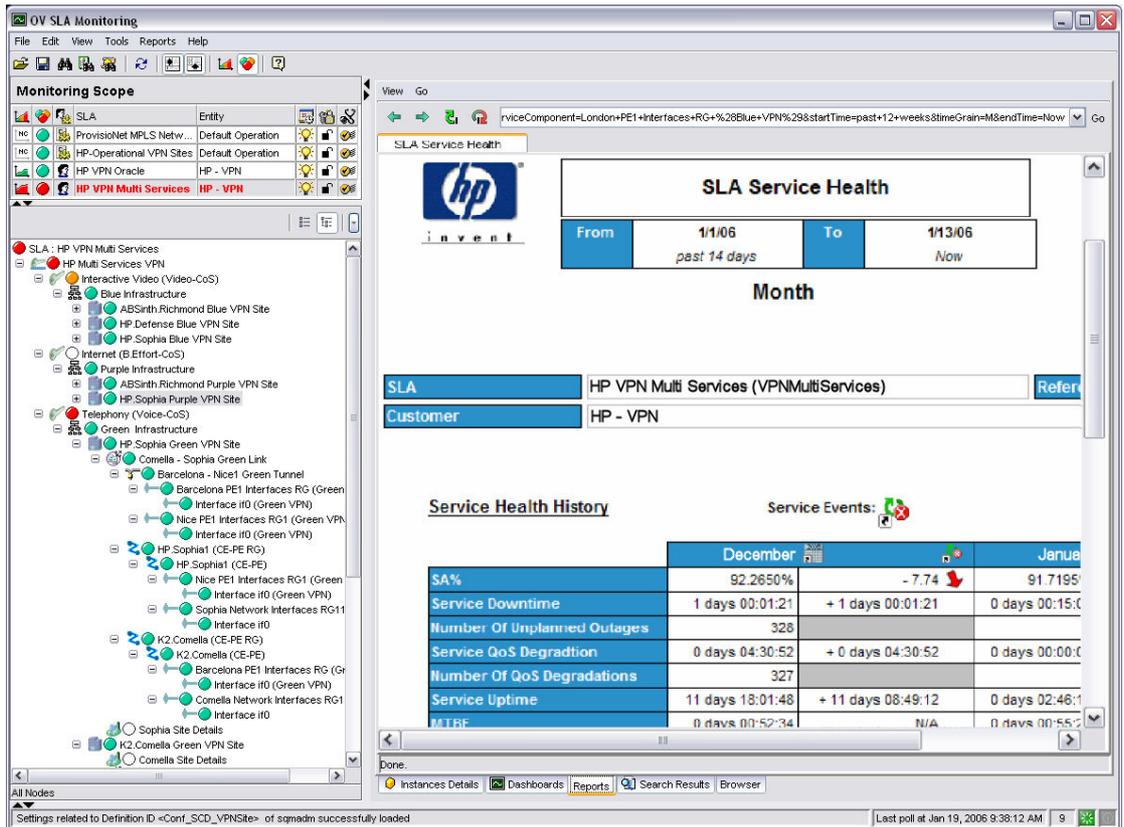


Figure 23 – SLA Service Health window

Additional customized reports can be developed to calculate, for examples, the rebates granted to customers whose services did not reach the expected level of quality.

You can also use additional contextual or user-defined menus to simplify operator related tasks. The following figure shows a user launching tools to view the associated VPN infrastructure schema within HP OpenView Service Quality Manager’s real-time user interface.

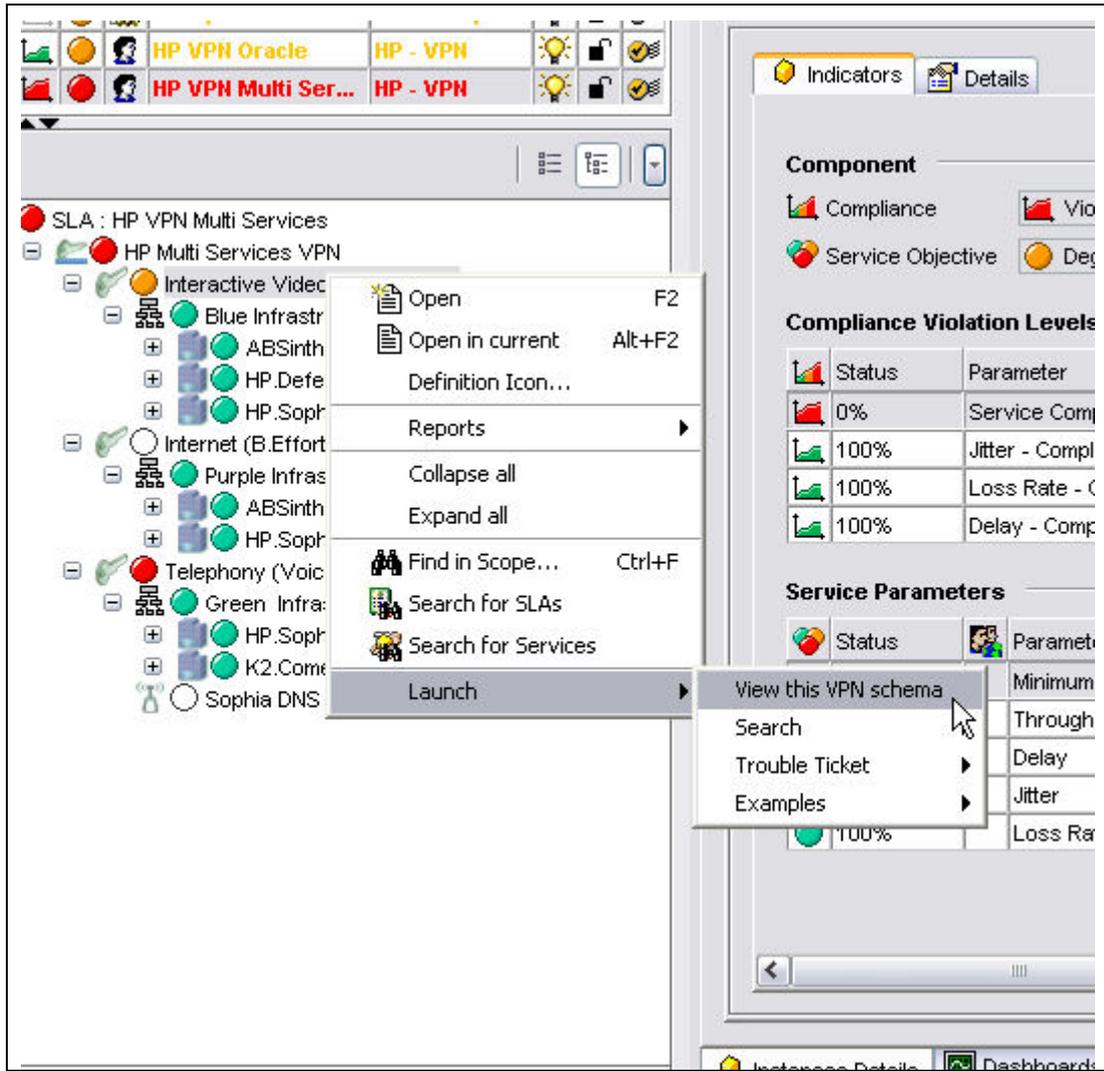


Figure 24 - 'Launch' contextual menu

Chapter 2

Setting up the OV SQM VPN/MPLS Value Pack

2.1 Software and hardware requirements

On HP-UX

The Value Pack must be installed on an HP OpenView SQM 1.3 (with last Service Pack update) software platform running on an HP-UX hardware platform. An Oracle database must be available on the server that is to hold the mediation database. The OpenView SQM platform must include the following elements:

- slmonitoring director
- SA Common V1.3 (SQMSAGTWCOMMON), and the SQL Service Adapter Runtime V1.3 (SQMSASQL)
- slreporting director
- The SQM SNMP Action Executor, if you want to send SNMP traps when an SLA is degraded or violated.

For details of the OpenView SQM software installation procedure, see the *HP OpenView Service Quality Manager Installation Guide*.

On Windows

Once you have installed the Value Pack, you must install at least the SLA Monitoring graphical interface and the SLA Administration graphical interface.

If you wish to develop your own Service Model based on the Service Model defined in the Value Pack, you must also install the Service Designer tool on top of the Rational Rose[®] software.

2.2 Value Pack contents

The Value Pack contains two installation kit portions for installation on the two platforms used jointly:

- Unix kit portion, for installation on an HP-UX platform
- Windows kit portion, for installation on a Windows NT, 2000, or XP platform

The contents of each of these kit portions are detailed in the following paragraphs.

You must install both of these kit portions, each on the appropriate platform, as described in “Installation” paragraph.

2.2.1 HP-UX installation kit portion

The Unix installation kit portion contains the following:

The scripts and data used to populate the mediation database (Mediation DB):

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/populate_sqlsa_db
```

Data and SQL scripts are contained in the following directories:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/Database/VPN_MPLS
```

The SQL SA toolkit projects (*.sqltk) and the generated kits (*.zip) are contained in the following directories:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/SQLSA/VPN_MPLS
```

The scripts that deploy and create the SQL SA, and load definitions and instances into the Service Repository Manager (SRM):

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/setup_solution
```

The scripts and data that only register service definitions and instances, together with the Service Levels and SLAs, into the SRM:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/load_definitions
```

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/load_instances
```

Their input data is contained in the following directories:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/Definition/  
    .../DFD  
    .../ExprDef  
    .../ServiceDef  
    .../ServiceLevel
```

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/Instance/  
    .../Customer  
    .../Profile  
    .../SI  
    .../SIG  
    .../SLA
```

The script that cleans up the Value Pack:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/cleanup_solution
```

The VPN/MPLS Value Pack specific Business Object universe and reports are delivered under directory:

```
$STEMIP_SC_HOME/SolutionSet/VPN_MPLS/Reporting/  
    .../Universe  
    .../Reports
```

Note: For this first version of the VPN/MPLS Value Pack, no customized reporting is delivered.

2.2.2 Windows NT/2000/XP installation kit portion

The Windows installation kit portion contains the following:

Value Pack documentation. This is contained in the following directory:

```
%STEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Doc
```

An HTML image of the Service Design, which does not require the Service Designer for display, is also provided, in the following directory:

```
%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Doc\ModelDesign
```

Service Model project files used as input for the Service Designer are contained in the following directory:

```
%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Model\Designer
```

Source code of the Custom Expressions used by the Service Model, are contained in the following directories:

```
%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Model\CustomExpressions\  
java
```

```
%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Model\CustomExpressions\  
sql
```

The VPN specific Business Object universe and reports are delivered under the following directory. They are the same as those delivered on HP-UX.

```
%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Reporting\  
...Universe  
...Reports
```

Note: For this first version of the VPN/MPLS Value Pack, no customized reporting is delivered.

2.3 Installation

2.3.1 Installation on an HP-UX platform

Logging on

Log on to the HP-UX system where you want to install the HP OpenView Service Quality Manager VPN/MPLS Value Pack, as **root** user.

Installing the kit

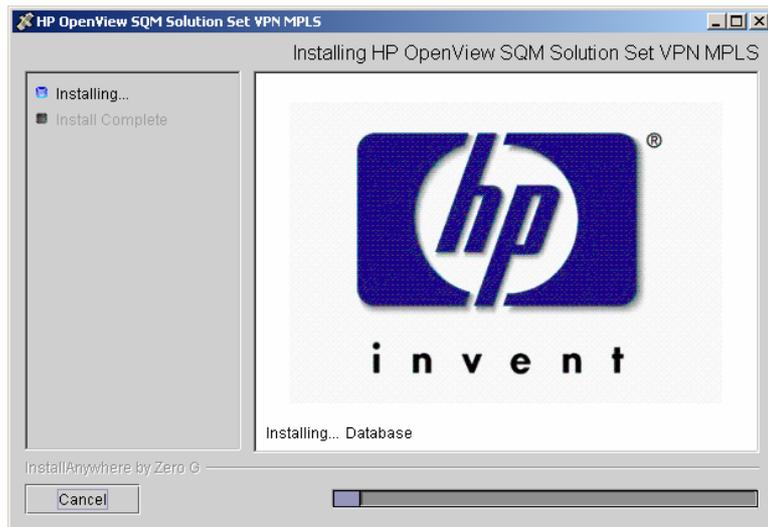
To install the Unix portion of the VPN/MPLS Value Pack kit, do the following.

- If the OpenView SQM kernel is not already installed on the server, do so by following the procedure described in the *HP OpenView Service Quality Manager Installation Guide*.
- If the OpenView SQM Kernel is not already initialized, you must set up the environment variables by entering the following commands. For further details, see the *HP OpenView Service Quality Manager Installation Guide*.

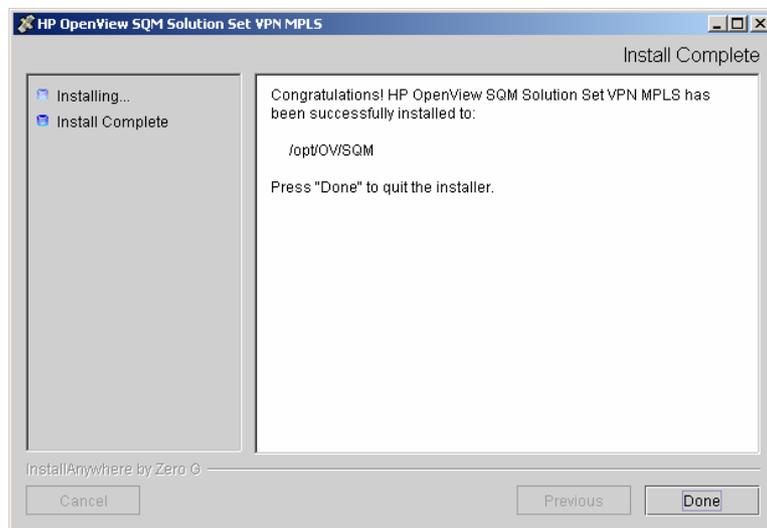
```
# cd $TEMIP_SC_VAR_HOME  
# . $TEMIP_SC_VAR_HOME/temip_sc_env.sh
```

- Insert the VPN/MPLS Value Pack CD-ROM into your CD-ROM drive, and then execute *SQMSSETVPNMPLS-1.30.00.bin*

The following output is displayed.



When the Unix portion of the VPN/MPLS Value Pack has been installed, the following screen is displayed.



Click **Done** to end the installation process on the Unix platform.

2.3.2 Installation on a Windows NT, 2000, or XP platform

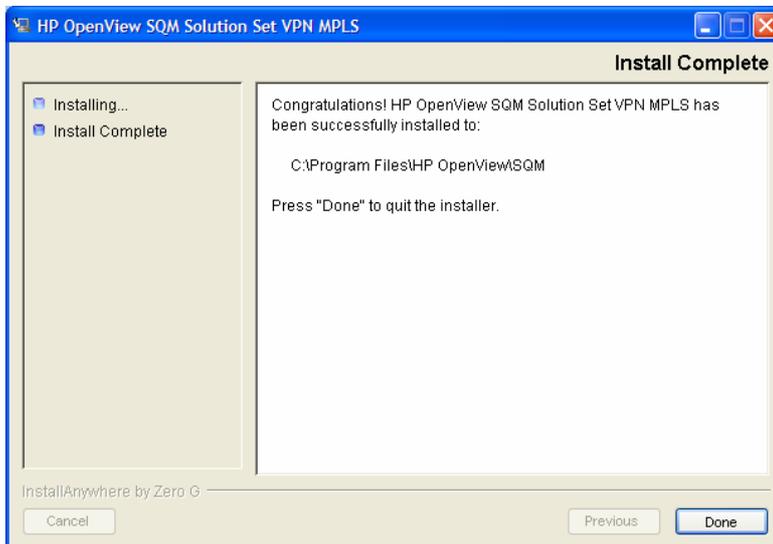
To install the Windows portion of the HP OpenView Service Quality Manager VPN/MPLS Value Pack kit, do the following.

1. Insert the VPN/MPLS Value Pack CD-ROM into your CD-ROM drive, and then open the `SQM-1.30.00\Windows` folder.
2. Run the `SQMSSETVPNMPLS-1.30.exe` installer.

The following output is displayed:



When the Windows portion of the VPN/MPLS Value Pack has been installed, the following screen is displayed.



Click **Done** to end the installation process on the Windows platform.

2.3.3 Installation of the VPN/MPLS value pack specific reports

The installation and setup of VPN/MPLS Value Pack reports, suppose that OV SQM Reporting has been previously configured. Please refer to the OV SQM Reporting Installation and Configuration Guide to perform this pre-requisite setup phase.

2.4 Setting up the Value Pack

To set up the Value Pack, you must execute the three following steps. These have been differentiated so that the mediation database (Mediation DB) can be created if it does not exist, and so that Mediation DB can be populated on a remote HP-UX server.

Note

You must perform each of the following steps under the **root** account. You must do so in order to set up each SQL Service Adapter and set up the **oracle** and **sqmadm** users as necessary.

1. Create the mediation database (i.e. the *sqlsa* database). This step is invoked by executing the *temip_create_database* script.
2. Populate the mediation database with data that has been collected from the different Data Feeder Instances (DFIs) used by the Value Pack. This step is invoked by executing the *populate_sqlsa_db* script.
3. Set up the SQL Service Adapter that collects data from the mediation database, and then registers the Service Definitions, Service Levels, Service Instances, and SLA(s) provided by the Value Pack. This step is invoked by executing the *setup_solution* script.

Each of these steps is described in detail below.

2.4.1 Creating the mediation database

The mediation database, *sqlsa*, must be created before the Value Pack is set up. It can be created either on the local HP-UX server or on a remote HP-UX computer, using the Oracle scripts provided with SQM to deploy and delete the Oracle databases used by the *OpenView SQM* applications. For more details of the installation procedure, see the *Service Adapter for HP OpenView Operations Installation, Configuration and User's Guide*.

2.4.1.1 *temip_create_database sqlsa*

This tool is used to deploy all databases used by the HP OpenView SQM system. You must invoke this tool to create the *sqlsa* database on the HP-UX computer that will host it.

Note

You can display help text for this tool by invoking the tool as follows:
`$TEMIP_SC_HOME/oracle/scripts/temip_create_database -h`

To create the mediation database, run the *temip_create_database* tool by doing the following.

Enter the following command:

```
$ TEMIP_SC_HOME/oracle/scripts/temip_create_database
```

By invoking the tool without specifying parameters, you run the tool in an interactive mode that asks you to choose which database you wish to create and guides you through the installation process by asking you basic questions.

The *temip_create_database* tool first displays a choice of available databases, as shown below.

```
temip_create_database will deploy a database for one of the following OpenView SQM products:
```

```
- TeSC logger                (logger)    [1]
- TeSC logger's archive      (arch)      [2]
- TeSC SPDM                   (spdm)      [3]
- TeSC SRM                     (srm)       [4]
```

```

- TeSC Datamart production (dmprod) [5]
- TeSC Datamart staging (dmstag) [6]
- TeSC SQL SA (sqlsa) [7]
- TeSC BO repository (borepos) [8]
- Exit (terminate the session) [e]

```

Please enter the database you want to install:

1. Select `sqlsa [7]` to create the mediation database.

The `temip_create_database` tool next asks you which predefined size of database you wish to create (*SMALL*, *MEDIUM*, *LARGE* or *USER*).

2. Select *SMALL*. This size is sufficient to hold the mediation database.

2.4.2 Populating the mediation database

Once you have created the mediation database as described above, execute the following script on the HP-UX computer that hosts the `sqlsa` database.

```
# $TEMIP_SC_HOME/SolutionSet/VPN/MPLS/populate_sqlsa_db
```

This script loads the data that has been collected from each DFI used by the VPN/MPLS Value Pack into the mediation database, `sqlsa`. If the `sqlsa` database is not on the same host as the OpenView SQM platform, you must also install the VPN/MPLS Value Pack on the HP-UX computer hosting the `sqlsa` database so that the `populate_sqlsa_db` script is installed on that computer.

The mediation database then contains all data collected from each DFI during one day of operational running, together with a view of each table that updates the record's timestamp to the current date.

2.4.3 Setting up the VPN/MPLS Value Pack

Once you have populated the mediation database as described in the previous paragraph, execute the following script on the HP-UX computer hosting the OpenView SQM platform:

```
# $TEMIP_SC_HOME/SolutionSet/VPN/MPLS/setup_solution
```

This script does the following.

It creates the SQL Service Adapters that collect the data from the mediation database and publish the Key Performance Indicators to the OpenView SQM platform.

At this stage of the setup process, you are asked to enter the Java DataBase Connectivity (JDBC) parameters used to connect the SQL Service Adapters to the mediation database containing their source data. You must enter the following information:

```

Please enter the
database URL [jdbc:oracle:thin:@haltere.vbe.cpqcorp.net:1521:sqlsa]:
    jdbc:oracle:thin:@yourMediationDBhost.vbe.cpqcorp.net:1521:sqlsa
Please enter the database username [sqlsa]: sqlsa
Please enter the database password [*****]: sqlsa
Please enter the database JDBC Driver [oracle.jdbc.driver.OracleDriver]:

```

1. Register the Service Definitions and Service Levels in the Service Repository Manager.
2. Load the Service Instances and SLA into the Service Repository Manager.

2.4.4 Configuring the VPN/MPLS Pack Runtime license

Before you run the HP OpenView Service Quality Manager VPN/MPLS Value Pack, you must configure the **SQM-VPACK-VPN** license.

The VPN/MLPS Value Pack is supplied with a temporary license, under which it can be used for a trial period of 90 days. This license is activated when the application is executed for the first time.

For details of how to use the *temip_sc_license* utility to retrieve HP OpenView SQM licenses and import them into Autopass, see the *HP OpenView Service Quality Manager Administration Guide*.

2.4.5 Running the VPN/MPLS Value Pack

Once you have executed the *setup_solution* script, you can run the acquisition director. The VPN/MPLS Value Pack director is called *acquisitionVPN*.

Note

Before you invoke the *temip_sc_start_director* command described below, you must ensure that the *setup_solution* script has set the JDBC 'CLASSPATH' environment variable in the SQM environment file *\$TEMIP_SC_VAR_HOME/temip_sc_env.sh* if this has not already been done. In this case, you must also source this file and then stop and start the kernel by executing the commands *temip_sc_stop_kernel* and *temip_sc_start_kernel*. You do not need to restart the OpenView SQM platform itself.

The final message displayed when the *setup_solution* command is executed shows whether or not it is necessary to re-source the *\$TEMIP_SC_VAR_HOME/temip_sc_env.sh* (OpenView SQM environment) file before you restart the OpenView SQM kernel and start the *acquisitionVPN* director.

To run the HP OpenView Service Quality Manager VPN/MPLS Value Pack director, do the following.

Log on as **sqmadm** user.

1. Execute the following command.

```
# temip_sc_start_director -plat slmv12 -dir acquisitionVPN
```

This enables you to monitor the statuses of Services and SLAs provided by the VPN/MPLS Value Pack, using the SLA Monitoring graphical interface.

SLAs, Customers, Service Instances, and Service Level management tasks are performed using the SLA Administration graphical interface, however.

2.4.6 Configuring the HP OpenView SQM platform to run the VPN/MPLS Value Pack

2.4.6.1 Predefined monitoring user profile

The VPN/MPLS Value Pack provides a predefined user profile for the HP OpenView Service Quality Manager real-time monitoring interface. The user name associated to this predefined user profile is *sqmopVPN*. The HP_UX administrator has to create

this user before you can login to the HP OpenView Service Quality Manager real-time monitoring interface using this identity.

2.4.6.2 Acquisition period

You do not need to configure the HP OpenView Service Quality Manager platform manually to run the VPN/MPLS Value Pack unless you regenerate the custom SQL Service Adapter (SA) kits. If this is the case, you must then adjust each modified Service Adapter's acquisition period so that it is longer than the data collection period for data provided by the mediation database. The minimum acquisition periods for the VPN Service Adapter provided in the VPN/MPLS Value Pack is 5 minutes.

For details of how to use the TIBCO Designer graphical interface to edit the configuration used by the Central Repository provided with the OpenView SQM Kernel, see the *HP OpenView Service Quality Manager Administration Guide*.

2.5 Cleaning up the VPN/MPLS Value Pack

You can remove the SQL Service Adapter from the HP OpenView SQM platform and deregister the Service Definitions, Service Levels, Service Instances, and SLAs provided with the VPN/MPLS Value Pack from the Service Repository Manager by executing the following HP-UX script:

```
# $TEMIP_SC_HOME/SolutionSet/VPN_MPLS/cleanup_solution
```

2.6 Uninstalling the VPN/MPLS Value Pack

2.6.1 Uninstalling the VPN/MPLS Value Pack from HP-UX

Note:

Before you uninstall the VPN/MPLS Value Pack, you must execute the *\$TEMIP_SC_HOME/SolutionSet/VPN_MPLS/cleanup_solution* script to remove related data from the Service Repository Manager and from the SQM Central Repository. For further details, see “Cleaning up the VPN/MPLS Value Pack”, above.

Logging on

To log on to the HP-UX computer in order to uninstall the VPN/MPLS Value Pack, do the following.

Log on to the HP-UX system as **root** user.

1. Load the OpenView SQM environment variables by entering the following commands.

```
# cd $TEMIP_SC_VAR_HOME  
# . $TEMIP_SC_VAR_HOME/temip_sc_env.sh
```

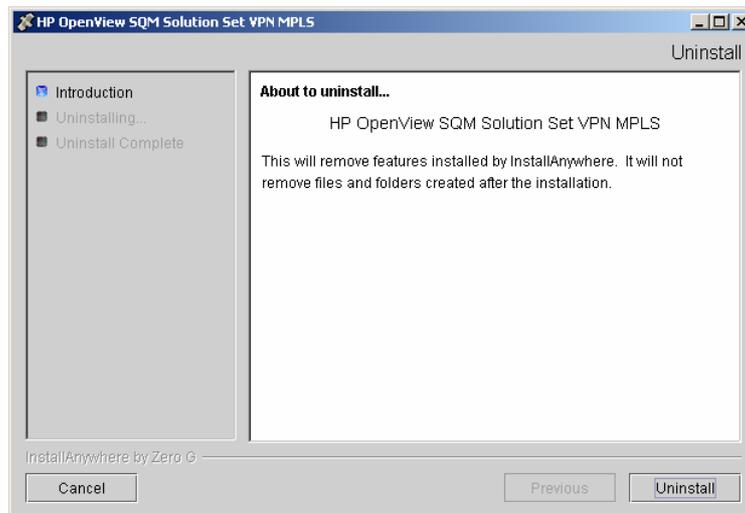
Uninstalling the kit

On the OpenView SQM platform on which the VPN/MPLS Value Pack has been installed, do the following.

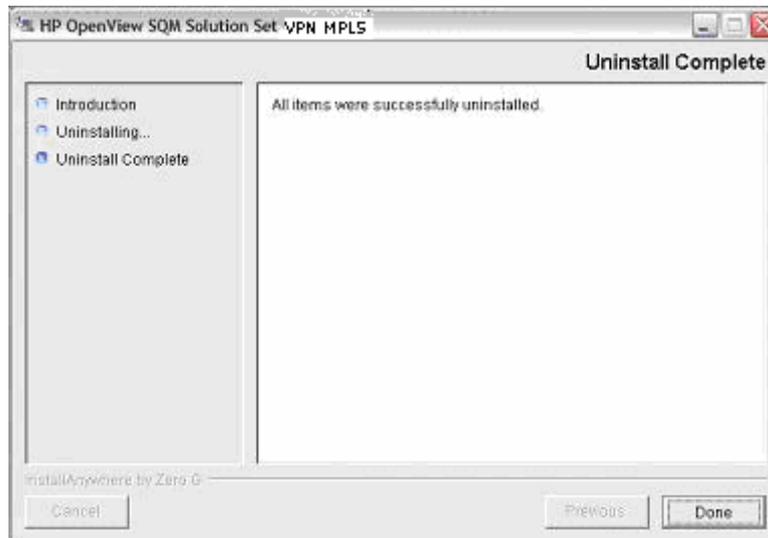
2. Enter the following commands.

```
# cd $TEMIP_SC_HOME/SolutionSet/VPN_MPLS/Uninstaller  
# ./Uninstall_VPN_MPLS
```

The **Uninstall** window is displayed.



3. Click **Uninstall** to remove the software from your system. When all of the software has been removed, the following window is displayed.



4. To finish, click **Done**.

2.6.2 Uninstalling the VPN/MPLS Value Pack from Windows

To uninstall the VPN/MPLS Value Pack from Windows, do the following.

From the Windows *Start* menu, select

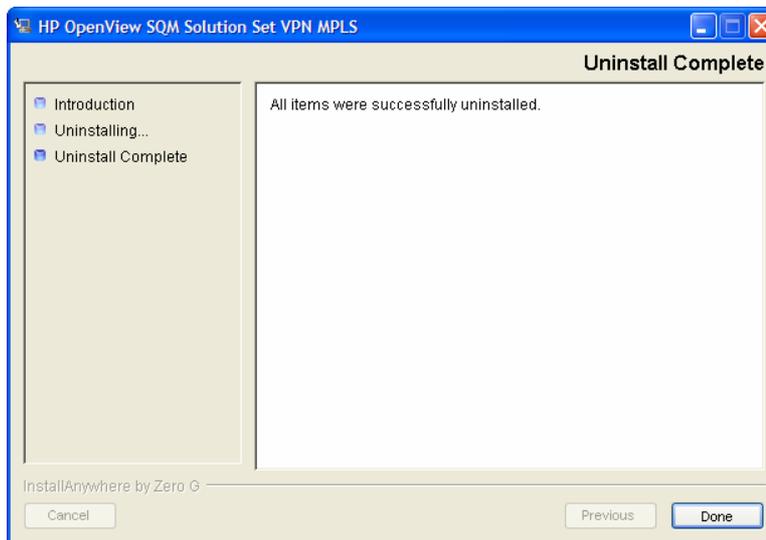
All programs → *HP OpenView* → *SQM* → *SSET* → *VPN_MPLS* → *Uninstall*.

The **Uninstall** window is displayed.



1. Click **Uninstall**.

The software is uninstalled from your system. When all of the software has been removed, the following window is displayed.



2. To finish, click **Done**.

Chapter 3

VPN/MPLS Value Pack model design

The aim of this chapter is to help you to navigate through the VPN Service Definitions provided in the VPN/MPLS Value Pack model. Its intention is not to describe all elements in the model, since these can be easily retrieved and understood using the HP OpenView Service Quality Manager Service Designer.

Instead, it describes the model structure and explains the main Class Diagrams, Sequence Diagrams, and Custom Expressions to help you to understand the key points of the design.

Note

Users that do not intend to customize the Service Definitions provided by the VPN/MPLS Value Pack do not need to read this chapter.

It is assumed that readers of this chapter are familiar with the information modeling concepts described in the *HP OpenView Service Quality Manager Information Modeling Reference Guide*.

3.1 Introduction

The VPN/MPLS Value Pack contains generic Network Provider and Customer infrastructure components that serve as a basis for deployment of VPN Services.

3.2 Service Designer project structure

The model main Service Designer project file has the **.mdl** suffix. The Service Designer project that has been used to design the VPN Service Model that the Value Pack installs in the `%TEMIP_SC_HOME%\SolutionSet\VPN_MPLS\Model\Designer` directory is:

- VPNMultiServices.mdl

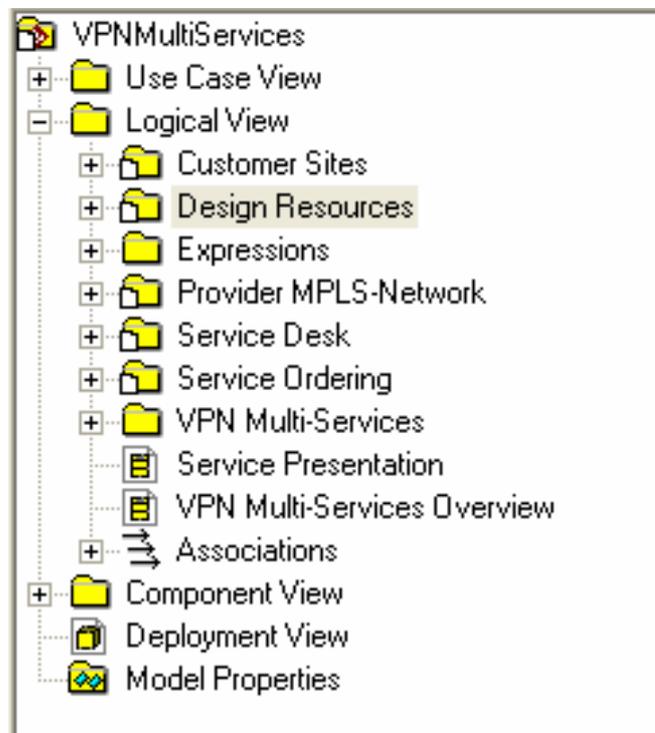
3.2.1 Package description

As shown in the following figure, the upper level of each Service Designer project provided by the VPN/MPLS Value Pack contains the following packages:

- CustomerSites
- DesignResources
- ProviderMPLSNetwork
- ServiceDesk

- ServiceOrdering

Figure 25 - VPN service design package



The Class Diagram View Package provides a simplified view of the generic service component that relate to particular domains:

- Customer Sites: Customer Service components entering into VPN Service delivery (VPN Site, CE Interface Redundancy Group, CE Interface...)
- Provider MPLS-Network: Service Provider components entering into the VPN Service delivery (POP, MPLS Tunnel, PE Interfaces Redundancy Group, PE Interfaces...)
- VPN Multi-Services: VPN CoS definition, using Customer and Provider package components.
- Service Desk: Help Desk component.
- Service ordering: Service Activation components.

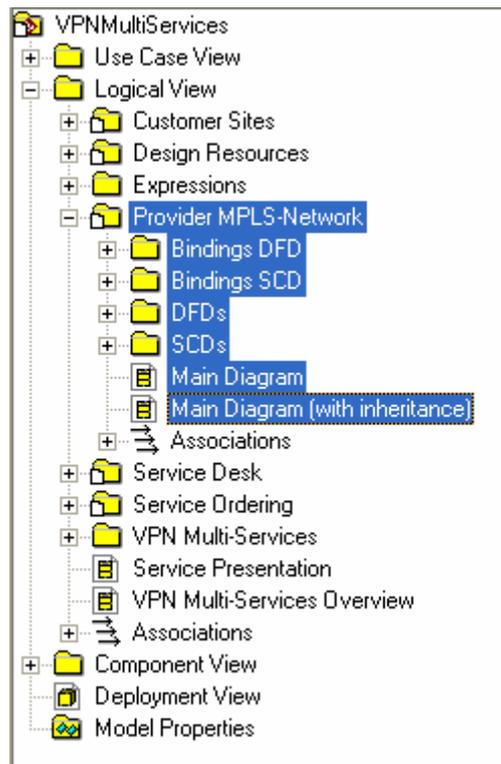
Additional packages are providing global resources to previously described VPN Service packages:

- Design Resources: Group of super class definitions, inherited by other components. General resource definition.
- Expressions: set of Pre-defined and Customized expressions used by the VPN Service packages.

In each package, standard sub-directories are defined:

- Bindings DFD: Package primary bindings.
- Bindings SCD: Package secondary bindings.
- DFDs: List of DFDs used by this package
- SCDs: List of components defined in this package.

Figure 26 - Package sub-packages

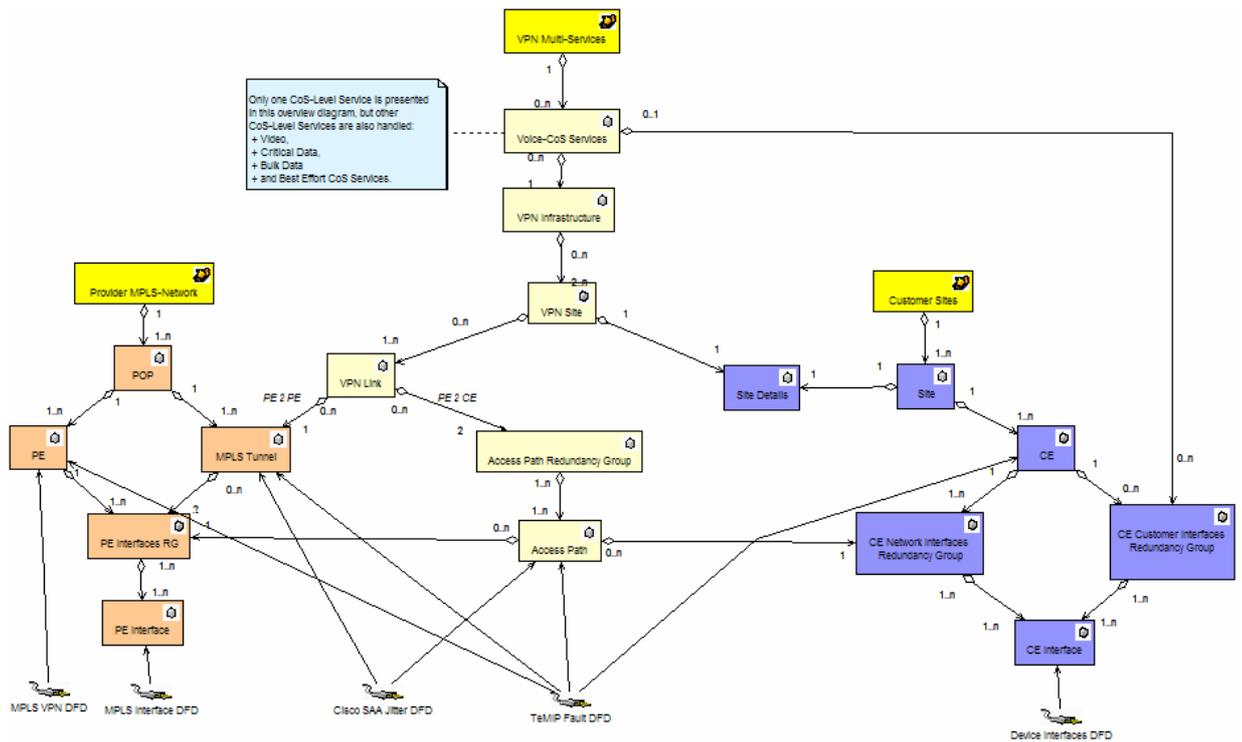


3.3 Key points of the design

3.3.1 VPN Value Pack service

The VPN service has been designed as shown in the following class diagram.

Figure 27 - VPN Multi-Service class diagram



The VPN service comprises the following group of components:

- **Service Provider Network infrastructure:** All VPN Core network components (Orange color).
- **Customer Network/Service infrastructure:** All VPN Customer components (Blue color).
- **VPN Multi-Service:** All components entering into the VPN Service delivery (Yellow color).

3.4 SQL Service Adapter

3.4.1 Data sources, and KPIs

3.4.2 SQL SAs deployed by the Value Pack

The following figure shows how each runtime component provided by the Value Pack is integrated into the SQM platform. These components are:

- Mediation database
- acquisitionVPN director
- SQL Service Adapter (SQL SA) embedded in the acquisitionVPN director

Note

The SQL SA is not the actual SA that the OpenView SQM provides to collect data on your physical service resources. The SQL SA retrieve data from the mediation database into which the Value Pack loads data collected in one day from a predefined set of resources used to build the Value Pack.

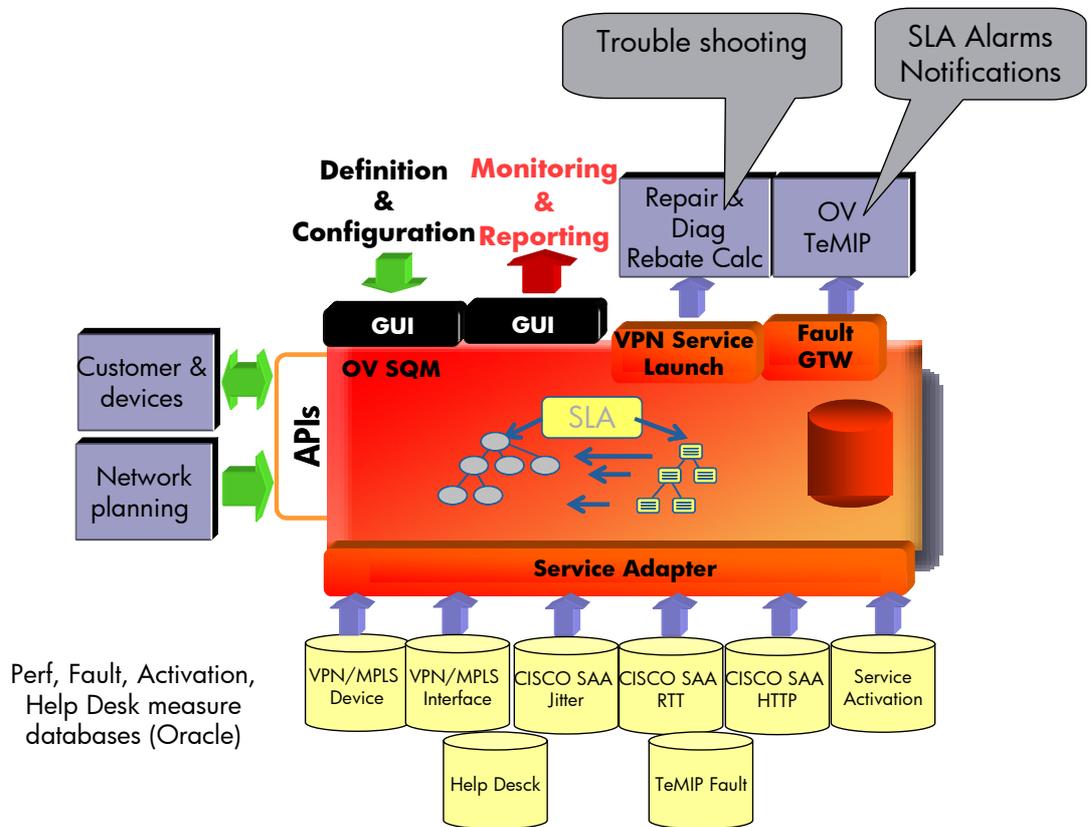
A unique SQL SA is used to collect the KPIs described in the previous paragraph from the Value Pack mediation database in which the live data collected in one day is loaded for the Service Instances and their Service Component Instances deployed by the Value Pack. This SQL SA, which was built using the SQM SQL SA toolkit, is:

VPN_MPLS SA: The custom SQL SA that collects from the mediation database the KPIs provided by the different data sources.

The mediation DB, related to Cisco Service Assurance Agent (Cisco SAA) and to MPLS/VPN, are build on top of OV PI report packs, respectively the "Cisco SAA Datapipe" and "MPLS VPN Datapipe" report packs.

These both report packs also collect KPI directly from respective SNMP agents, and populate retrieved data into specific DB tables, where the mediation DB is plugged. For a complete reference on this OV PI DB tables/mediation DB tables integration, please refer to Appendix B.

Figure 28 - Value Pack runtime components



Solution building guidelines

4.1 Introduction

You can use the Value Pack as a starting point from which to develop your own solution.

In all cases, you must do at least the following when you transform the Value Pack into an operational solution:

- You must replace the SQL Service Adapters provided by the VPN/MPLS Value Pack with Service Adapters that collect KPIs for your own service resources.
- You must use the SLA Administration graphical interface to create the appropriate Service Instances and groups, customers, and SLAs for your own case.

If necessary, you can modify the Service Definitions via the Service Designer, and then load the new model into the SRM before instantiation. You may also need to redefine the Service Levels for your own case, using the SLA Administration graphical interface.

4.2 Modifying the VPN/MPLS Value Pack SQL SA

In most cases, you can use the SQL Service Adapter provided by the VPN/MPLS Value Pack, but you must nevertheless customize so that it can connect to the database in which the KPIs for your own service resources are stored. You can do so using the SQL Service Adapter Toolkit provided with the HP OpenView Service Quality Manager. For details of how to use the SQL SA Toolkit to customize the SQL SA, see the *HP OpenView Service Quality Manager SQL Service Adapter Toolkit Installation, Configuration and User's Guide*. You can use the SQL SA Toolkit's graphical interface to define the mapping between the DFD parameters and the database's TABLE columns. The Value Pack includes the SQL SA toolkit project used to build the initial SQL SA. This project is defined in the file with the *.sqltk* suffix.

Note

When you use the SQL Service Adapter Toolkit's graphical interface to modify the SQL SA projects provided by the Value Pack, you must set up the properties of the project at least to set your own project paths, and specify the JDBC connector parameters enabling you to access your KPI database.

In all other cases, you must replace an SQL Service Adapter provided by the VPN/MPLS Value Pack by a different type of Service Adapter.

Troubleshooting Guide

This appendix describes basic methods you can use to check the condition of Value Pack components.

Installation

You must check that the following Value Pack components are set up correctly on the Service Quality Manager platform:

The mediation database must be populated and accessible.

The Service Adapter provided by the Value Pack must be running.

The Service Definitions and Service Instances must be registered in the Service Repository Manager.

In addition to checking these components, you can verify the contents of the following setup log file created when the Value Pack is set up:

```
$TEMIP_SC_VAR_HOME/trace/Setup_SolutionSet_VPN_MPLS.log
```

Service Repository Manager

Diagnostic check

You can check whether the Service Repository Manager has correctly registered the Service Definitions delivered by the Value Pack, by entering the following commands. Each command must return the required definitions. Alternatively, you can use the SLA Administration graphical interface to browse these definitions.

To check that all definitions required by the VPN Service Definition have been registered, enter the following command:

```
temip_sc_get_sd -sd VPNMultiServices
```

To check whether the Service Repository Manager has correctly registered the Service Instances, Service Levels and SLAs delivered by the VPN/MPLS Value Pack, you can either enter the following commands or use the SLA Administration graphical interface to browse the SLA:

```
temip_sc_get_sla -sd VPNMultiServices
```

This command must return the contents of all SLA defined in the Value Pack demo environment.

VPN/MPLS Value Pack Service Adapters

Diagnostic check

To check that the Service Adapters delivered by the VPN/MPLS Value Pack have been set up correctly, enter the *start director* command as follows:

```
temip_sc_start_director -plat slmv12 -dir acquisitionVPN
```

Next, check that all Service Adapters are running by entering the following command:

```
temip_sc_show_director -plat slmv12 -dir acquisitionVPN
```

The *VPN/MPLS* application must be running.

Mediation database

Diagnostic check

Use an SQL browser to check that the following *VIEWS* are not empty:

SAA_HTTP_DFD_V

SAA_JITTER_DFD_V

SAA_RTT_DFD_V

DEV_IF_DFD_V

MPLS_IF_DFD_V

MPLS_VPN_DFD_V

OVSA_V

OVSDCALLS_V

TEMIP_FAULT_DFD_V

Use an SQL browser to check that the following *TABLES* are not empty:

SAA_HTTP_DFD;

SAA_JITTER_DFD;

SAA_RTT_DFD;

DEV_IF_DFD;

MPLS_IF_DFD;

MPLS_VPN_DFD;

OVSA;

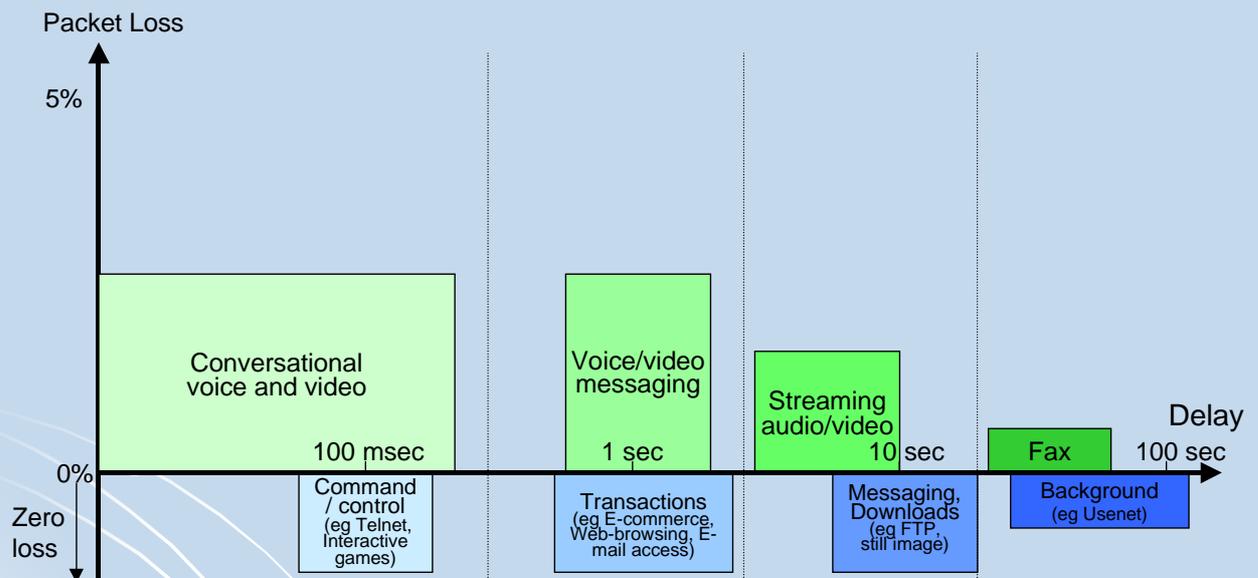
OVSDCALLS;

TEMIP_FAULT_DFD;

Appendix A – Example of VPN Service Objectives

The following tables provide with some example of Objectives associated to the different VPN Classes of Service.

User-centric Delay and Packet Loss requirements – ITU G.1010*



Source: ITU G.1010 [“Draft New Recommendation G.QoSRT – End-user Multimedia QoS Categories”, ITU-T study group 12, contribution 37, August 2001]

Performance targets for Conversational/Real-Time Services (audio and video applications)

- Source: 3GPP TS 22.105 V6.2.0 & G.1010

Medium	Application	Degree of symmetry	Typical Data rates/ Amount of Data	Key performance parameters and target values		
				End-to-end One way Delay	Delay Variation within a call	Information Loss**
Audio	Conversational voice	Two-way	4 - 64 kb/s	<150 msec Preferred* <400 msec limit*	< 1 msec	< 3% Packet Loss Ratio
Video	Videophone	Two-way	16 -384 kb/s	< 150 msec preferred <400 msec limit Lip-synch : < 100 msec		< 1% Packet Loss Ratio
Data	Telemetry - two-way control	Two-way	<28.8 kb/s	< 250 msec	N.A	Zero
Data	Interactive games	Two-way	< 1 KB	< 250 msec	N.A	Zero
Data	Telnet	Two-way (asymmetric)	< 1 KB	< 250 msec	N.A	Zero

* Assumes adequate echo control

** Exact values depend on specific codec, but assumes use of a packet loss concealment algorithm to minimise effect of packet loss

9

Appendix B – OV PI/mediation DB integration

Cisco SAA NRT Datapipe 2.0

SNMP Tables

Cisco SAA NRT Datapipe collects information coming from these SNMP tables:

rttMonCtrlAdminTable (1.3.6.1.4.1.9.9.42.1.2.1), contains information on Round Trip Time (RTT) monitoring definitions.

rttMonEchoAdminTable (1.3.6.1.4.1.9.9.42.1.2.2), contains information on RTT specifics definitions.

rttMonLatestHTTPOperTable (1.3.6.1.4.1.9.9.42.1.5.1), contains the status of latest HTTP RTT operation.

rttMonLatestJitterOperTable (1.3.6.1.4.1.9.9.42.1.5.2), contains the status of latest Jitter operation.

rttMonLatestRttOperTable (1.3.6.1.4.1.9.9.42.1.2.10) contains the status of latest RTT operation.

OV PI DB Tables

These previous SNMP tables are mapped in OVPI DB tables, as follow:

SNMP Table	DB Table
rttMonCtrlAdminTable	RCiscoSAADP_NRTAdmin
rttMonEchoAdminTable	
rttMonLatestHTTPOperTable	R_CiscoSAADP_NRTHttp
rttMonLatestJitterOperTable	R_CiscoSAADP_NRTJitter
rttMonLatestRttOperTable	R_CiscoSAADP_NRTRtt

These DB tables are bound together as follow:

Table 1	Table 2	Join
R_CiscoSAADP_NRTHttp	RCiscoSAADP_NRTAdmin	T1.DSI_KEY_ID = T2.DSI_KEY_ID
R_CiscoSAADP_NRTJitter	RCiscoSAADP_NRTAdmin	T1.DSI_KEY_ID = T2.DSI_KEY_ID
R_CiscoSAADP_NRTRtt	RCiscoSAADP_NRTAdmin	T1.DSI_KEY_ID = T2.DSI_KEY_ID

Mediation DB Tables

The mediation DB tables, and also their respective DFD, are plugged on the last OVPI DB tables as follow:

OV PI DB Table	Mediation DB Table
R_CiscoSAADP_NRTHttp RCiscoSAADP_NRTAdmin	SAA_HTTP_DFD
R_CiscoSAADP_NRTJitter RCiscoSAADP_NRTAdmin	SAA_JITTER_DFD
R_CiscoSAADP_NRTRtt RCiscoSAADP_NRTAdmin	SAA_RTT_DFD

MPLS-VPN Datapipe

SNMP and OV PI tables

The 2 supported SNMP MIBs are dispatched in 5 groups. The following table summarizes them mapping between SNMP tables and OV PI DB tables:

SNMP table	OV PI Raw table	OV PI Property table
mplsInterfaceConfTable (1.3.6.1.3.96.1.1)	xDPMplsInterfaceConf	K_DPMplsInterface
mplsInterfacePerfTable (1.3.6.1.3.96.1.2)	xDPMplsInterfacePerf	K_DPMplsInterface
mplsVpnScalars (1.3.6.1.3.118.1)	xDPVpnDeviceStats	K_DPMplsVpnDevice
mplsVpnInterfaceConfTable (1.3.6.1.3.118.1.2.1)	xDPVpnInterfaceConf	K_DPVpnInterface
mplsVpnVrfTable (1.3.6.1.3.118.1.2.2)	xDPVpnVrf	K_DPVpnVrf
mplsVpnVrfSecTable (1.3.6.1.3.118.1.2.6)		
mplsVpnVrfPerfTable (1.3.6.1.3.118.1.3.1)		

Mediation DB Tables

The mediation DB tables, and also their respective DFD, are plugged on the last OVPI DB tables as follow:

OVPI DB Table	Mediation DB Table
RMplsInterfacePerf K_MplsInterface	MPLS_IF_DFD

RVpnDeviceStats	
K_MplsVpnDevice	
<i>K_VpnInterface</i>	MPLS_VPN_DFD
RVpnVrf	
K_VpnVrf	

