034115

PHOSPHORUS

Lambda User Controlled Infrastructure for European Research

Integrated Project

Strategic objective:
Research Networking Testbeds

Deliverable reference number: D.2.1

The Grid-GMPLS Control Plane architecture

Due date of deliverable: 2007-02-28
Actual submission date: 2007-03-04
Document code: Phosphorus-WP2-D2.1

Start date of project:
October 1, 2006

Duration:
30 Months

Organisation name of lead contractor for this deliverable: **neXtworks (NXW)**

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract

This report details the Grid-GMPLS Control Plane Architecture that will be implemented and demonstrated by the PHOSPHORUS WP2 research activities.

G^2MPLS is a Network Control Plane (NCP) architecture that implements the concept of Grid Network Services (GNS). It is aimed to provide part of the functionalities related to the selection, co-allocation and maintenance of both Grid and network resources through a set of seamless procedures at the user-to-network and inter-domain boundaries. G^2MPLS functionalities comprise discovery and advertisement of Grid resource capabilities and availabilities, GNS setup including resource localization, advance reservation and management of recovery in case of fault, GNS monitoring.

The G^2MPLS network and service models are presented in this document, as well as the general G^2MPLS functional architecture and the specific routing, signalling and recovery procedures. Interworking with Grid middleware, NRPS layer and GN2-BoD system is an important issue for which appropriate solutions have been proposed.
This deliverable will drive the subsequent low-level design and developments of the G^2MPLS Control Plane.

Project:	Phosphorus
Deliverable Number:	D.2.1
Date of Issue:	04/03/07
EC Contract No.:	034115
Document Code:	Phosphorus-WP2-D2.1

List of Contributors

Nicola Ciulli	NXW	Radoslaw Krzywania	PSNC
Gino Carrozzo	NXW	Damian Parniewicz	PSNC
Giodi Giorgi	NXW	Maciej Strozyk	PSNC
Dimitra Simeonidou	UESSEX	Jarek Nabrzyski	PSNC
George Zervas	UESSEX	Ariel Oleksiak	PSNC
Reza Nejabati	UESSEX	Anna Tzanakaki	AIT
Artur Binczewski	PSNC	Georgios Markidis	AIT
Maciej Stroiński	PSNC	Wes Doonan	ADVA
Jan Węglarz	PSNC	Eduard Escalona	Fi2CAT/UPC
Bartosz Belter	PSNC	Ferdinand Hommes	FHG-IAIS

Table of Contents

List of Figures

List of Tables

0 Executive Summary

This document describes the Grid-GMPLS Network Control Plane architecture that the core research and development activity of PHOSPHORUS WP2.

In section 1 the objectives of the G^2MPLS architecture are stated, as well as the scope of the document in the WP2 framework.

In section 2 the terminology relevant for the G^2MPLS architecture is presented by specifying the main source of information.

Section 3 provides a brief review of current work in relevant SDOs, mainly aimed to establish the concepts and references for a correct positioning of G^2MPLS with respect to the standardization.

In section 4 the requirements towards G^2MPLS are gathered, including Grid application requirements with impact on underlying network services, Grid service plane requirements, Transport Plane requirements, Control Plane requirements and Management Plane requirements.

In section 5 the reference network and service models for G^2MPLS are presented, by positioning the G^2MPLS Control Plane in the overall PHOSPHORUS architecture and describing the new concepts of Grid Network Service transactions, destination endpoints, permeability of the UNI towards Grid users.

Section 6 describes the G^2MPLS specific procedure with an end-to-end and cross-plane scope.

Section 7 provides the relevant G^2MPLS use-cases, aimed to identify the roles and requirements for some system components such as the G.OUNI gateway and the Grid middleware.

In Section 8 the G^2MPLS network reference points are identified and abstract messaging of the interfaces is provided.

Section 9 details the G^2MPLS functional architecture, by assigning roles and responsibilities for the G^2MPLS procedures to different functional entities in the Network Control Plane.

Section 10 provides some specific issues on the G^2MPLS pat computation process.

Section 11 deals with the robustness of the proposed architecture, both in terms of transport service and in terms of Control Plane resiliencies.

Section 12 reports the scenarios identified for the interworking with Network Resource Provisioning Systems, while section 13 focuses on the interworking with GÉANT2 BoD system.

Appendix A contains complementary information needed to refine the G^2MPLS architectural description, e.g. the information model, the address and identifiers spaces, etc.

1 Objectives and Scope

This document specifies the requirements and the architecture of the Grid-enabled GMPLS (a.k.a. G^2MPLS).

G^2MPLS is a Network Control Plane (NCP) architecture that implements the concept of Grid Network Services (GNS). In the PHOSPHORUS framework, GNS is a service that allows the provisioning of network and Grid resources in a single-step, though a set of seamlessly integrated procedures. As per OFG Open Grid Service Architecture (OGSA, [OGF-GFD80]), GNS belongs to the class of base resources, i.e. those physical or logical resources that are out of the context of the OGSA. Examples of such entities include CPUs and memory in the physical case and licenses, contents and OS processes in the logical case.

G^2MPLS architecture is expected to expose interfaces specific for Grids and is made of a set of extensions to the standard ASON/GMPLS architecture. Therefore, G^2MPLS results in a more powerful NCP solution than the standard ASON/GMPLS, because it will comply with the needs for enhanced network and Grid services required by network "power" users/applications (i.e. the Grids). Nevertheless, the requirements of standard users that just require the automatic setup and resiliency of their connections across the transport network are supported by G^2MPLS as well.

G^2MPLS is not conceived to be an application-specific architecture, i.e. it will support any kind of endpoint applications by providing network transport services and procedures that can fall back to the standard GMPLS ones.

G^2MPLS is aimed to provide part of the functionalities related to the selection, co-allocation and maintenance of both Grid and network resources. This goal translates in:

- *Discovery and advertisement* of Grid capabilities and resources of the participating Grid sites (Vsites);
- *Service setup*
  - *Coordination* with those parts of the Grid middleware needed for the local configuration of the Grid job (i.e. the local job scheduler in particular)
  - *Configuration* of the deriving network connections among the Vsites participating to the Grid job. Depending on the Grid application capabilities and requirements, the network attachment endpoints could be specified or not: e.g. in case of distributed computing and visualization, the network attachment endpoints could be declared, in case of distributed storage they could not. Moreover,

depending on the administrative partitioning of the network, the configured network service might span multiple domains.
   - *Management of resiliency* for the installed network services and, depending on the specification of the network attachment endpoints, possible escalation to the Grid middleware components that could be responsible for check-pointing and recovering the whole job (e.g. by changing the involved Vsite or pausing/postponing the job, etc.)
   - *Advanced reservation*s[1] of Grid and network resources, aimed to guarantee connection availability at job execution time by providing users with a priori information about start, wait and completion times.
- *Service monitoring*
   - Retrieving of the status of a Grid job and the related network connections.

The main advantage of co-allocating jointly Grid and network resources is to configure network connections in the same tier of Grid resources, by guaranteeing service availability and tailoring to the user requirements. In state of the art, co-allocation of Grid and network resources is obtained through cross-layer cooperation between a Grid Resource Management System (RMS, e.g. GLOBUS GRAM) and a Network Resource Provisioning Systems (NRPS). This is a dual-approach and the two layers are reciprocally un-aware, because the RMS lies in the Grid Service Plane and owns just Grid resources, while the NRPS lies in the Network Service Plane and is specialized in managing just network resources (in some case with advance reservation).

The G^2MPLS NCP can bring to an innovation in this field, because of its

- Faster dynamics for service setup in the same time-scale of the NCP ones
- Availability of well-established procedures for traffic engineering, resiliency and crankback
- Uniform interface (G.OUNI) for the Grid-user to trigger Grid & network transactions not natively dependent on a specific Grid middleware.

Moreover, the compliance of the G^2MPLS to the ASON/GMPLS architecture foster for the possible integration of Grids in real operational networks, by overcoming the current limitation of Grids operating as stand-alone networks with their own administrative ownership and procedures.

The definition of the G^2MPLS architecture takes into account a number of factors:

- the optical transport network technologies mostly required by Grids and in the evolution path of NRENs,
- the coexistence and interworking with NRPS-es or Network Management Systems such as the GÉANT2 Inter-Domain Manager (IDM),
- the direct interfacing of Grid middleware and applications.

[1] As defined in OGF GRAAP-WG, advance reservation is a possibly limited or restricted delegation of a particular resource capability over a defined time interval, obtained by the requester from the resource owner through a negotiation process.

Most of the references in this document are to those elements and applications that are in the scope of the PHOSPHORUS project and run in the Grid layer and/or in the Network Management Plane (i.e. NRPS-es such as UCLP, DRAC, ARGON; Grid middlewares such as UNICORE, GLOBUS; etc.). Nevertheless, the G^2MPLS architectural specification is independent of the specific implementations and it has been designed in order to provide interoperation with those external entities through some adaptation modules.

This document will not redefine concepts already part of the current ASON and/or GMPLS requirements, architectures and protocols. The main goal here is to provide a comprehensive and consistent architectural description of the specific G^2MPLS NCP extensions.

This document describes the reference network and service models which are the rationale for the G^2MPLS deployment. Deriving from these models, a set of network reference points is identified and characterized in terms of available functionalities, along with those NCP components which are needed to provide the one-step control of Grid and network resources.

For the purposes of this architectural document, components, interfaces and protocols are described at a high level of abstraction, i.e. neither detailed software architecture nor specific syntax elements for protocols (e.g. definition of object formats, messages, etc.) are provided since they will be part of other specific documents.

2 Terminology

In this section some definitions particularly relevant in the scope of G^2MPLS architecture are provided in case with reference to the originator document.

Keyword	Source	Definition
Data model	**[IETF-RFC3444]**	A mapping of the contents of an information model into a form that is specific to a particular type of repository, protocol, platform, etc. It is a rendering of an information model according to a specific set of mechanisms for representing, organizing, storing and handling data. There are typically three parts: • A collection of data structures such as lists, tables, and relations; • A collection of operations that can be applied to the structures such as retrieval, update, and summation; • A collection of integrity rules that define the legal values or changes of state (operations on values). The audience for a data model is implementers.
GLUE schema	**[GLUE]**	An information model that provides a description of core Grid resources at the conceptual level by abstracting real world resources into constructs that can be represented in computer systems (e.g., objects, properties, behaviour, and relationships). The GLUE schema is not tied to any particular implementation and can be profitably used to exchange information among different knowledge domains.

Keyword	Source	Definition
Grid	**[OGF-GFD81]**	A system that is concerned with the integration, virtualization, and management of services and resources in a distributed, heterogeneous environment that supports collections of users and resources (virtual organizations) across traditional administrative and organizational domains (real organizations).
Grid fabric	**[OGF-GFD81]**	The core set of service interfaces that must be implemented in order to realize an OGSA Grid. Also known as the OGSA infrastructure services.
Grid middleware (MW)	**OGF**	Grid technology (a.k.a. middleware) is employed to facilitate formalizing and complying with the Grid context associated with an application execution. Middleware is computer software that connects software components or applications. It is used most often to support complex, distributed applications. It includes web servers, application servers, content management systems, and similar tools that support application development and delivery. Middleware is especially integral to modern information technology based on XML, SOAP, Web services, and service-oriented architecture.
Grid Network Service (GNS)	**[OGF-GNS]**	A network service (e.g. management of QoS classes, policy enforcement points, topology data, network usage metrics, AAA, etc.) with roles and/or interfaces that are deemed to be specific to a Grid infrastructure is a Grid Network Service [OGF-GNS]. Network Services belongs to the class of the base resources of the OGSA Architecture [OGSA]. Base resources are those physical or logical resources that are supported entities out of the context of the OGSA. Examples of such entities include CPUs and memory in the physical case and licenses, contents and OS processes in the logical case.
Grid Resource	**[OGF-GFD81]**	In OGSA, a resource is an entity that is useful in a Grid environment. The term usually encompasses entities that are pooled (e.g. hosts, software licenses, IP addresses) or that provide a given capacity (e.g. disks, networks, memory, databases). However, entities such as processes, print jobs, database query results and virtual organizations may also be represented and handled as resources.

Keyword	Source	Definition
Grid service	**[OGF-GFD81]**	In general use, a Grid service is a Web service that is designed to operate in a Grid environment, and meets the requirements of the Grid(s) in which it participates.
Information model	**[IETF-RFC3444]**	An abstraction and representation of entities in a managed environment including properties, operations, and relationships. An information model is independent of implementation: that is, it is protocol-neutral, repository-independent, and platform-independent. An information model's level of specificity is varied, dependent on need. It can be described in a formal language such as UML or an informal natural language such as English. An information model is useful for designers to describe the managed environment, for administrators to understand the modelled objects, and for implementers as a guide to the functionality that can be described, limited by, and coded in the data models. CIM is an example of an object-oriented information model.
Job	**[OGF-GFD81]**	A user-defined task that is scheduled to be carried out by an execution subsystem. In OGSA-EMS, a job is modelled as a manageable resource, has an endpoint reference, and is managed by a job manager.
Job manager	**[OGF-GFD81]**	In OGSA-EMS, a service that manages a set of one or more job instances, which may be structured (e.g. a workflow or dependence graph) or unstructured (e.g. an array of non-interacting jobs). The job manager encapsulates all aspects of job execution, including interacting with execution planning services, the provisioning system, containers, and monitoring services. It may also deal with failures and restarts, it may schedule jobs to resources, and it may collect agreements, reservations and job service data.
Job Submission Description Language (JSDL)	**[OGF-GFD81], [JSDL]**	A language for describing job submissions, including details of their required execution environments. See https://forge.gridforum.org/projects/jsdl-wg for more information.

Keyword	Source	Definition
Network Resource Provisioning System (NRPS)	**WP1 D1.1**	The module that has the main task of specifying, reserving, allocating and deploying the set of network resources (links, cross-connections, etc.) required to accomplish the task specified by a user.
NRPS driver	**WP1 D1.1**	The NRPS Driver is an adaptor that translates the proprietary interface of each NRPS to a single common interface to interact with the NSP. This NRPS driver will be defined in this project. On top of each NRPS there will be an NRPS Driver that presents to the NSP a common interface independently of the NRPS running underneath.
Network Control Plane (NCP)	**[ASON-ARCH, ASON-DEF].**	The network Control Plane performs the call control and connection control functions. Through signalling, the Control Plane sets up and releases connections, and may restore a connection in case of a failure. The Control Plane also performs other functions in support of call and connection control, such as routing information dissemination.
Network Service Plane	**WP1 D1.1**	The network service plane is the adaptation layer between the Grid layer and the NRPS layer. It deals with inter domain routing issues at the NRPS layer and will coordinate the different domains in the NRPS layer to meet network resources requests coming from the Grid Layer.
TE-link	**[IETF-RFC4201, IETF-RFC4202]**	A traffic engineering (TE) link is a logical construct that represents a way to group/map information about certain physical resources (and their properties) that interconnect LSRs with information that is used by Constrained SPF (for the purpose of path computation) and by GMPLS signalling.
Virtual organization (VO)	**[OGF-GFD81]**	A virtual organization (VO) comprises a set of individuals and/or institutions having direct access to computers, software, data, and other resources for collaborative problem-solving or other purposes. VOs are a concept that supplies a context for operation of the Grid that can be used to associate users, their requests, and a set of resources. The sharing of resources in a VO is necessarily highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs.

Keyword	Source	Definition
Vsite	**UNICORE architecture**	A Vsite identifies a particular set of Grid resources at a UNICORE site (Usite) and is controlled by a Network Job Supervisor (NJS). Vsites may consist of a single supercomputer or a cluster. If more than one resource is operated by an organization there can be one Vsite for each resource inside one Usite.
Workflow	**[OGF-GFD81], The Grid Workflow Forum**	A workflow is a pattern of business process interaction, not necessarily corresponding to a fixed set of business processes. All such interactions may be between services residing within a single data center or across a range of different platforms and implementations anywhere. Grid Workflows differ from "normal" workflows, such as business workflows or IT workflows, mainly in the following sense: • *Stateful/stateless*: Activities of Grid workflows are often related to stateful services (workflow of Grid Services), whereas other workflows are mostly related to the invocation of stateless services (workflow of Web Services) • *Reliability*: In a computing Grid, resources may fail during the runtime of the workflow. This must taken into account by advanced workflow fault management techniques (e.g. workflow check-pointing and recovery, monitoring) • *Performance*: One objective of Grid computing is to provide high performance computing power. Therefore, Grid Workflows have to deal with resource brokerage, scheduling (load balancing), and distributed applications (parallel computing)

2.1 Abbreviations

A full list of the abbreviations used in this document is provided in Section 16.

3 Reference Standards

This chapter provides a summarized analysis of the reference standards or draft specifications that form the basis of the G^2MPLS Control Plane architecture.

Among the SDOs in the scope of this survey, some are acting in the pure-network domain, such as the Internet Engineering Task Force (IETF), the International Telecommunication Union – Telecommunications sector (ITU-T) and the Optical Internetworking Forum (OIF), others are acting in the Grid domain, such as the Open Grid Forum (OGF).

The main goal of this section is to provide the positioning of G^2MPLS with respect to the identified architectures and interfaces/protocol specifications, in order to define what will be imported and used by the original reference architectures, what needs to be adapted or changed, what needs to be defined from scratch.

3.1 Review of current work in relevant SDOs

A summary of the relevant SDO in the scope of PHOSPHORUS Control Plane activities is provided in Figure 3-1, along with the relationships among the different parties.

The Internet Engineering Task Force (IETF) is chartered by the Internet Society to design and develop architectures and technologies for use by the Internet community. The IETF in turn charters various working groups organized in various subject areas to investigate and propose solutions to any Internet-related network engineering tasks which may arise.

The International Telecommunication Union (ITU) is focused on the definition of architectures and requirements. The ITU mission is to deliver formal standards to be used by the different players in the Telecommunication field (i.e. equipment vendors and network operators) for their specific implementations or operations.

Optical Internetworking Forum (OIF) is an industry-oriented forum with the aim of promoting the development and deployment of interoperable optical networking. The OIF work delivers in the form of Implementation Agreements (IAs) for optical networking products, dealing with optical component technologies (e.g. electrical interfaces, optical transponder interoperability, tuneable lasers, etc.) and network routing and signalling procedures. Implementation Agreements are based on the requirements developed cooperatively by end-users, service providers, equipment vendors and technology providers, and are aligned with worldwide standards (basically ITU but IETF to some extents), augmented if necessary.

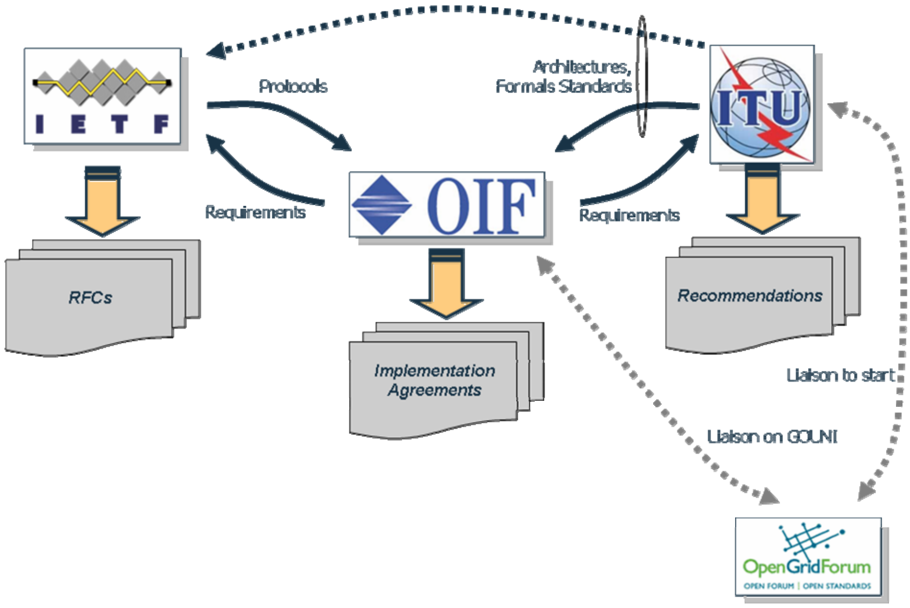


Figure 3-1: Relevant SDOs and their relationships.

3.1.1 ITU-T ASON

Automatically Switched Optical Network (ASON) is the ITU-T SG15 Control Plane framework aimed to provide SDH transport networks (ITU-T Rec. G.803) and Optical Transport Networks (ITU-T Rec. G.872) with:

- Fast and efficient configuration of both switched and soft permanent connections;
- Restoration and traffic engineering functionalities;
- Easy reconfiguration or modification of pre-established connections;

The ASON architecture [ASON-ARCH] describes the set of Control Plane components that are needed to set up, maintain and release connections related to the underlying transport network resources. The basic requirements fulfilled by the ASON Control Plane are:

- support different Transport Planes technologies (but with main focus on the ITU-specific ones , i.e. SDH and OTN),

- be applicable regardless of the used control protocols,
- be applicable regardless of the Control Plane organization (domains / routing areas),
- be applicable regardless of the connection control paradigm (centralized / distributed),
- be reliable, scalable and efficient in all its procedures.

An important feature of ASON is its generic applicability, which allows different packaging of the ASON Control Plane components and functionalities by different vendors.

Despite the detailed architectural work, ASON does not define any protocols, but – on the contrary – it provides a formal standard in which the more protocol-oriented actions by IETF and OIF can take place.

A network conforming to the ASON recommendation is composed of domains which interact in a standardized way, but whose internal operation is protocol-independent and is not subject to standardisation.

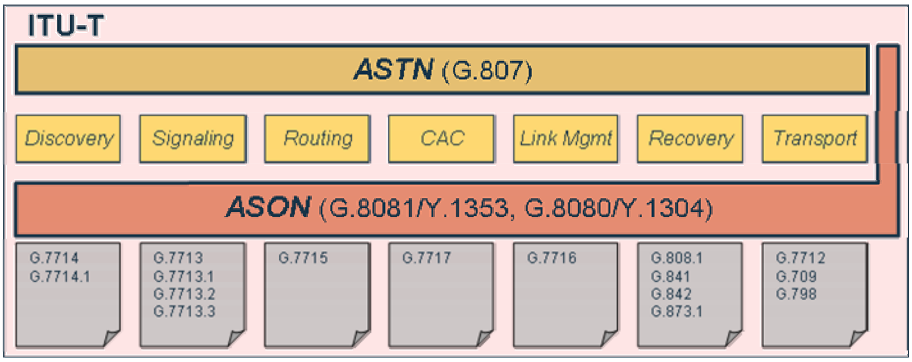


Figure 3-2: ITU-T ASON standardization framework.

The key ASON recommendations can be grouped according to thematic different areas as shown in Figure 3-2. In details, the relevant specifications are:

- Requirements and architecture:
    - ITU-T G.807/Y.1302 “Requirements for the Automatic Switched Transport Network (ASTN)”
    - ITU-T G.8080/Y.1304 “Architecture for the Automatic Switched Optical Network (ASON)”
    - ITU-T G.8081/Y.1353 “Definitions and Terminology for Automatically Switched Optical Networks (ASON)”
- Signalling:
    - ITU-T G.7713/Y.1704 “Generalised Distributed Connection Management”
    - ITU-T G.7713.1/Y.1704 “Distributed Call and Connection Management – PNNI Implementation”
    - ITU-T G.7713.2/Y.1704 “Distributed Call and Connection Management – GMPLS RSVP-TE Implementation”

  - ITU-T G.7713.3/Y.1704 “Distributed Call and Connection Management – GMPLS CR-LDP Implementation”
- Routing:
  - ITU-T G.7715/Y.1706 “Architecture and requirements for routing in automatically switched optical networks”
  - ITU-T G.7715.1/Y.1706.1 “ASON routing architecture and requirements for link state protocols”
- Resource discovery:
  - ITU-T G.7714/Y.1705 ”Generalised automatic discovery techniques”
  - ITU-T G.7714.1/Y.1705.1 “Protocol for automatic discovery in SDH and OTN networks”
- DCN:
  - ITU-T G. 7712/Y.1703 “Data Communication Network“
- Management:
  - ITU-T G.7718/Y.1709 “Framework for ASON management”
- Work in progress
  - ITU-T G.7716/Y.1707 [ASTN link connection status]
  - ITU-T G.7717/Y.1708 [Connection Admission Control]

The functional planes identified in ASON are shown in Figure 3-3. The Transport Plane is the space of those transport resource used for bearing user’s data between locations in a unidirectional or bidirectional flow. Transport Plane can also provide transfer of some control and network management information, depending on the topology of the Data Communication Network (DCN). The Control Plane is the space of those components that control automatically the procedures for setting up and recovering calls/connections, and for distributing routing information needed for call/connection control; moreover, Control Plane is the consumer of state information detected by the Transport Plane (e.g. link fault and signal quality) and used for connection maintenance and recovery. The Management Plane performs management functions for the Transport Plane, the Control Plane and the system as a whole, acting as a coordinator between all the planes. The management functional areas are identified in ITU-T Rec. M.3010 and are often referred to as FCAPS, i.e. fault, configuration, accounting, performance, security. The DCN is a network that provides the communication paths to carry signalling and management information. DCN supports Layer 1 (physical), Layer 2 (data-link), and Layer 3 (network) functionalities for: i) the Management Communication Network (MCN) used by the TMN to deliver management messages; ii) the Signalling Communication Network (SCN) used by the ASON Control Plane to deliver signalling and routing traffic.

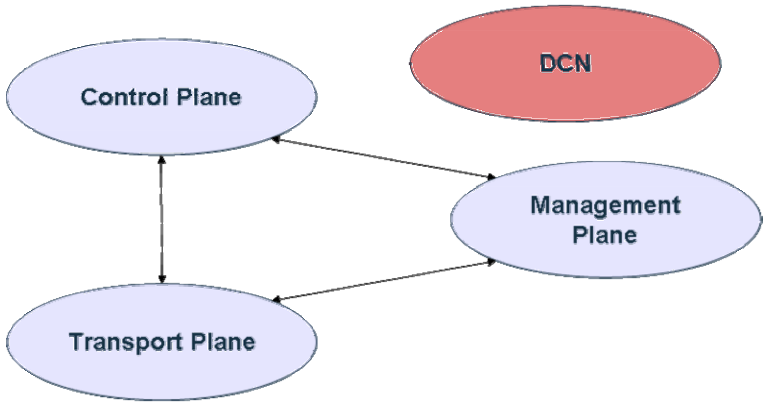


Figure 3-3: Network planes and their relationships in ASON.

The ASON CP functional components are briefly described in Table 3-1 in terms of their primary functions. Since they can be combined in different ways, an example layout of the ASON Control Plane is shown in Figure 3-4.

ASON component	Main purpose
Connection Controller (CC)	• Management and supervision of connection setups, modification, releases • It serves a whole sub-network
Routing Controller (RC)	• Provides the routing function • Provides information to: o CC: path (route) information needed to set up connections o Management: topology information for network management purposes
Link Resource Manager (LRM)	• LRMA (upstream) responsibilities: o allocation and de-allocation of link connections o providing topology and status information • LRMZ (downstream) responsibilities: o providing topology and status information
Traffic Policing (TP)	• Enforcement of user connection parameters • Not needed for Transport Networks based on Deterministic Multiplexing
Call Controller (CallC)	• Calling/called party call controller (CCC) o Located at the ends of a call (e.g. at UNI-C) o Call termination functions • Network call controller (NCC) o Located at intermediate domain boundaries along a call (e.g. at UNI-N, E-NNI)
Signalling Controller (SC)	• CC + CallC
Discovery Agent (DA)	• Implements the separation between Transport and Control Plane name space
Termination and Adaptation Performer (TAP)	• Provides the LRM with a view of the link connection resources • hides any hardware and technology specific details of the resources control

Table 3-1: ASON Control Plane components.

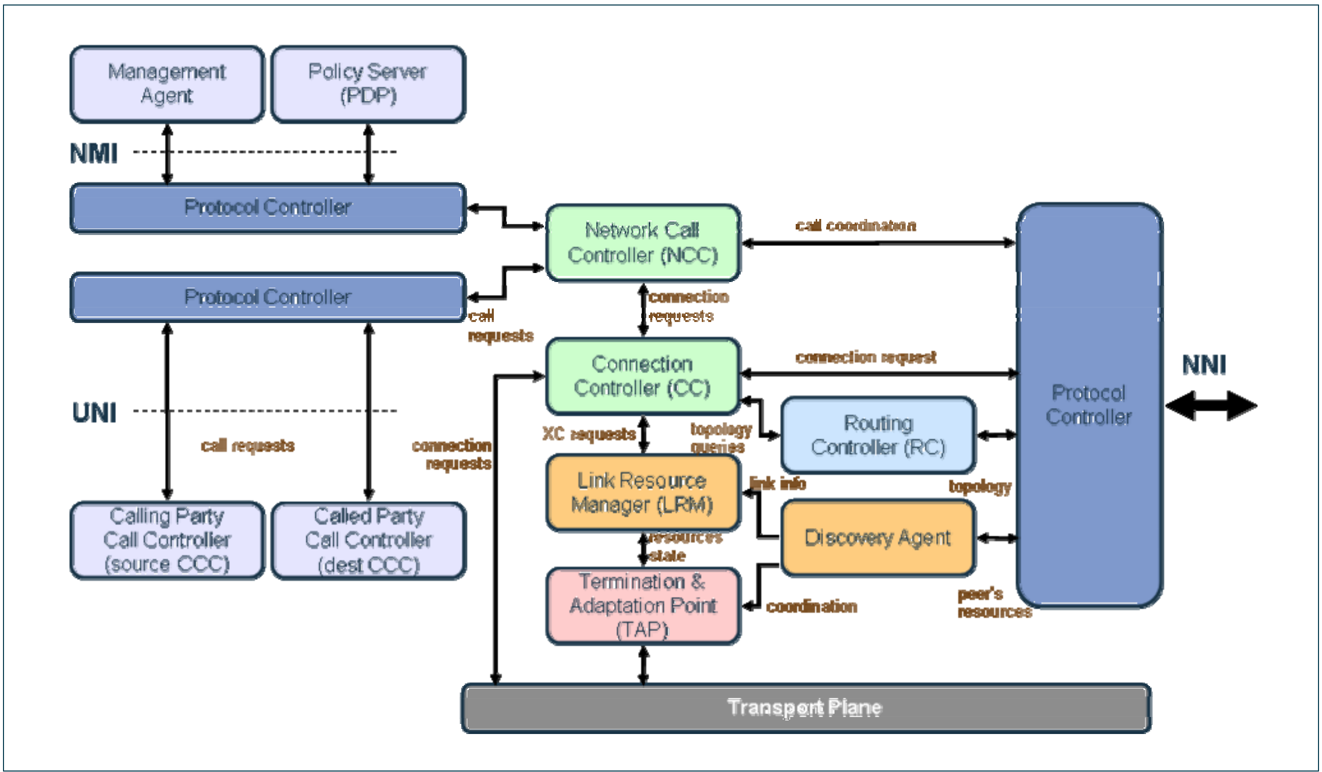


Figure 3-4: Example layout of ASON CP components.

Three types of connections are defined in ASON:

- **Permanent Connection (PC)**, which are provisioned by manual action or via the management system;
- **Switched Connection (SC)**, which are established on a request from the end user using a signalling protocol between connection end-points;
- **Soft Permanent Connection (SPC)**, which are user-to-user connection where the user-to-network portion of the end-to-end connection is set-up by the network management system as a PC, while the network portion of the end-to-end connection is set-up as a switched connection using the Control Plane (i.e. initiated by the Management Plane and set-up by the Control Plane).

All these connection types can support different connection capabilities, e.g. uni-directional point-to-point, bidirectional point-to-point, uni-directional point-to-multipoint, asymmetric (i.e. two unidirectional point-to-point connections with different properties).

The transport resources managed by the ASON are identified in terms of:

- Sub-network point (SNP), a.k.a. label
- SNP Pool (SNPP), a.k.a. component Data-link
- SNP Link connection (LC), i.e. the link between the two SNPs a.k.a. TE-link.

This taxonomy is more formal than in IETF GMPLS, which is based on labels, data links and TE link; however, it serves the abstraction goal of the ASON architecture with respect to some specific “implementation” such as GMPLS.

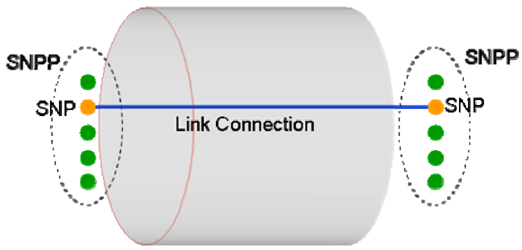


Figure 3-5: ASON representation of the transport network resources.

A basic feature of the ASON architecture is the separation of call control and connection control as shown in Figure 3-6. Call control is only needed at domain boundaries and only procedures for connection control are supported within a domain.

The support of segmented calls implies the identification of specific network reference points, i.e. the logical relationship between ASTN Control Plane entities, defined by the information flow between these entities. In the ASON Control Plane three interfaces are identified:

- **UNI**, a bidirectional signalling interface between service requester and service provider Control Plane entities
- **I-NNI**, a bidirectional signalling interface between Control Plane entities belonging to one or more domains having a trusted relationship
- **E-NNI**, a bidirectional signalling interface between Control Plane entities belonging to different domains

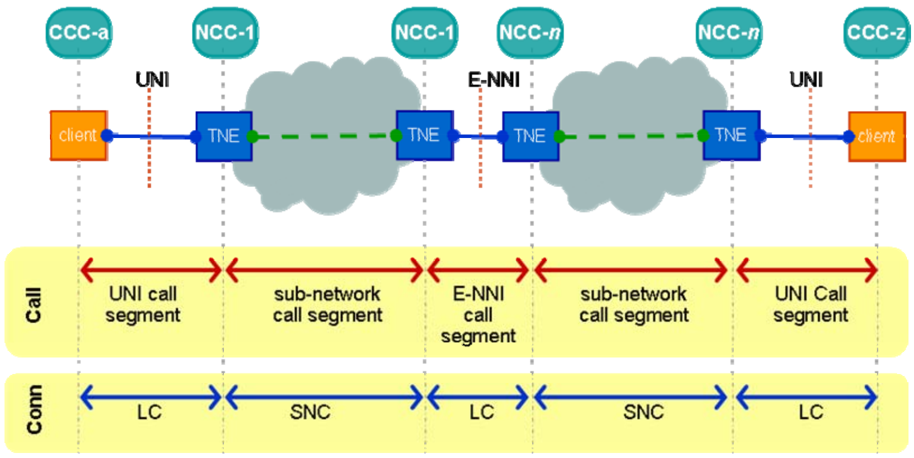


Figure 3-6: Call and connection separation in ASON CP.

The ASON architecture is based on the assumption that the Control Plane partitioning into domains is a best practice mainly related to network operator specific criteria: for example it could aim to protect business practices or to comply with managerial and/or policy issues or to represent the transport network heterogeneity in terms of technologies or control/management models. Based on this assumption, the role of the network reference points (the domain boundary ones in particular, UNI and E-NNI) is vital for the signalling and, consequently, the routing model needs to properly reflect the administrative partitioning. The ASON routing model is based on the concept of Routing Areas (RAs) hierarchically nested in order to reflect the administrative partitioning into domains. An ASON RA is a set of sub-networks, Sub-network Point Pool (SNPP) links that interconnect them, and SNPPs representing the ends of the SNPP links exiting that routing area. The RA services (e.g. path selection) are implemented by Routing Performers (RP), while the coordination and distribution of routing information within a routing area and across its borders are provided by Routing Controllers (RC). Each Routing Control Domain (RCD) knows all the details of its own topology, but it can know but the topology abstraction of other RCD belonging to either the same hierarchical level or lower levels (i.e. the boundary SNP links and SNPPs and – possibly – a summarization of the intra-domain links between SNPPs).

Inter-domain operations in the ASON framework are based on the concept of "federation", i.e. the community of domains that co-operate for the purposes of connection management. Three types of federation are identified in ASON (ref. Figure 3-8):

- **joint federation model**: one CC ("parent") coordinates the operation of the other CCs involved ("children")
- **cooperative model**: no single point coordination; each CC takes care of its part of connection
- **combined federation model**: a mix of the two above, i.e. the CCs are partitioned into groups where the joint federation model is applied; inter-group operations are coordinated according to the cooperative model.

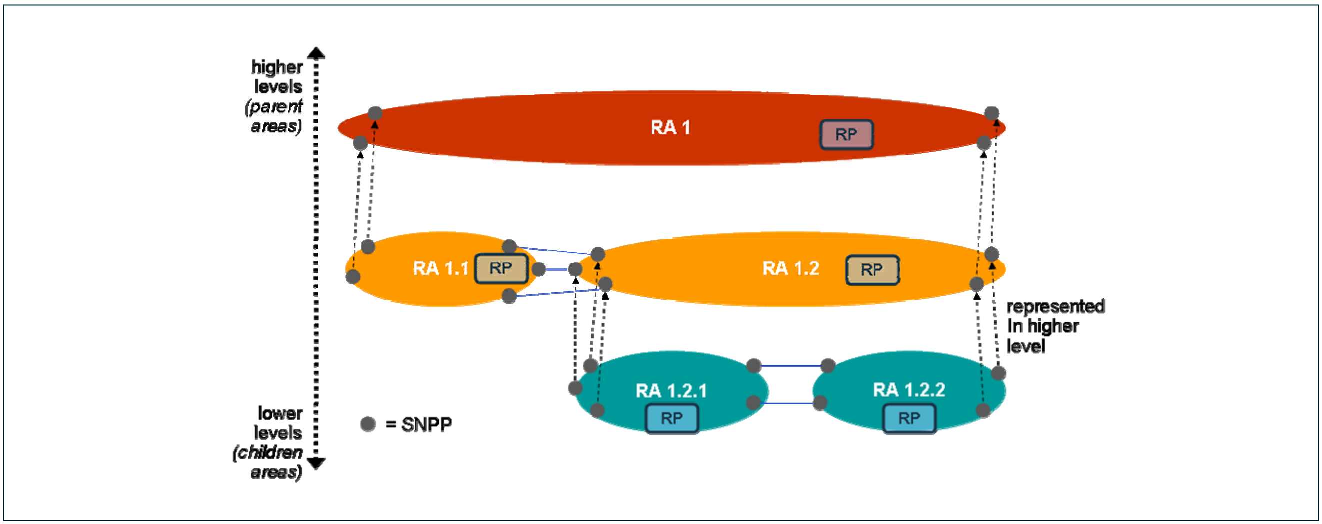


Figure 3-7: ASON routing hierarchies.

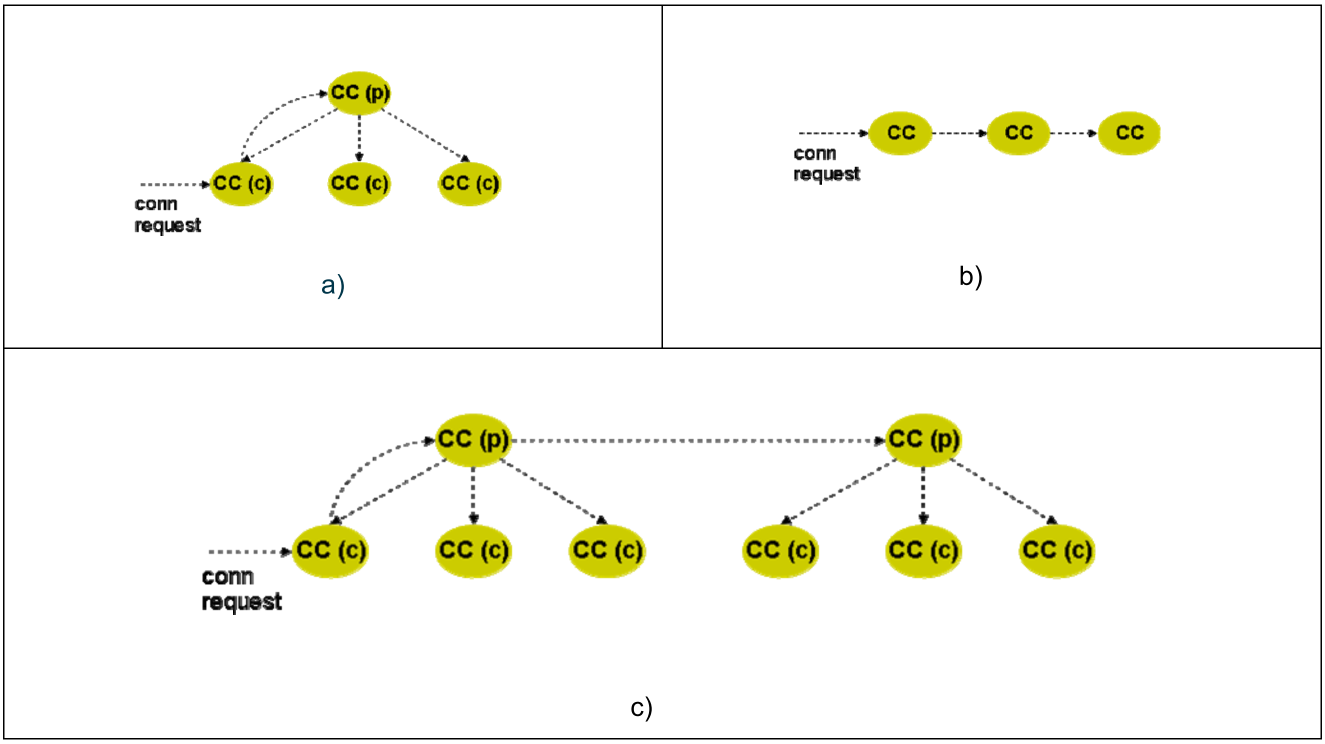


Figure 3-8: ASON federation models: a) joint federation, b) cooperative, c) combined federation.

3.1.2 IETF GMPLS

The IETF provides an open, collaborative forum for the development of standards related to Internet-based technologies. Standards are developed within the IETF through a voluntary, peer-reviewed process, which comprise a broad cross-section of stakeholders from industry, government and academia. Work within the IETF is organized into Areas, which broadly define various high-level categories of Internet-related work. These Areas are further subdivided into Working Groups (WGs), which are chartered to address particular aspects of Internet architecture and functionality within those Areas.

The various standards of possible interest and assistance to the PHOSPHORUS project largely emanate from the Routing Area within IETF, and specifically the CCAMP, L1VPN, L2VPN, MPLS, OSPF and PCE working groups. The following sections describe in some detail the charter of these various groups, and identifies both the adopted standards (referred to as Request for Comment – RFC – documents) and in-progress standards (referred to as Internet-Drafts) of relevance to the PHOSPHORUS effort.

3.1.2.1 *CCAMP Working Group*

The Common Control and Management Plane (CCAMP) working group is chartered to define a common Control Plane architecture suitable for managing the physical path and core tunnelling technologies of Internet and Telecom service providers. The primary technology to emerge from this working group is the Generalized Multi-Protocol Label Switching (GMPLS) architecture.

GMPLS is often misrepresented as being a “superset” of MPLS. On the contrary, GMPLS is a generalization of MPLS, and certain adjustments to the mechanisms defined in MPLS were made in order to broaden the applicability of those mechanisms beyond the single data plane envisioned by the original MPLS specifications. GMPLS can be viewed as a transport-agnostic, IP-inherited common Control Plane architecture, capable of providing standardized, interoperable Control Plane functionality to any form of transport network.

The GMPLS architecture as defined within the IETF is designed to be fully agnostic with regard to specific use models, deployment models, and transport environments. There exist GMPLS specifications describing methods for controlling transport networks as diverse as SDH/SONET, DWDM-based OTNs, OTNs incorporating G.709 encapsulation, and even Ethernet. GMPLS is also agnostic with regard to deployment method, by using well-established procedures for operating with peer-style relationships and with overlay-style relationships amongst domains.

The CCAMP working group has produced and ratified specifications regarding the operation of GMPLS networks in multi-domain scenarios, e.g. those situations where separately-managed transport networks wish to perform end-to-end traffic engineering across their several domains. The approach to handling multi-domain signalling and routing is standardized within the GMPLS framework. Multi-level and multi-region traffic engineering, e.g. those operational cases whereby GMPLS-based LSPs are to be carried within other GMPLS-based LSPs at different levels of switching hierarchy, is already subject to considerable standardization.

The CCAMP working group has also issued standards and drafts regarding the establishment and management of GMPLS-based LSPs incorporating protection attributes. Existing standards define mechanisms necessary to signal protection attributes, establish relationships between LSPs forming protection groupings, and perform automatic recovery/restoration activities.

In general the CCAMP working group has issued many RFCs and Internet-Drafts of relevance to the work of various PHOSPHORUS WPs, ranging from architectural documents to specifications for new protocols to applicability statements of GMPLS technologies to common use scenarios.

The following list provides brief descriptions of the core standards documents directly relevant the PHOSPHORUS G^2MPLS Control Plane architecture.

- *RFC3945: Generalized Multi-Protocol Label Switching (GMPLS) Architecture*
  This document describes the architecture of GMPLS. GMPLS extends MPLS to encompass time-division (e.g., SONET/SDH, PDH, G.709), wavelength (lambdas), and spatial switching (e.g., incoming port or fiber to outgoing port or fiber). The focus of GMPLS is on the Control Plane of these various layers since each of them can use physically diverse data or forwarding planes.

- *RFC3471: Generalized Multi-Protocol Label Switching (GMPLS) Signalling Functional Description*
  This document describes extensions to Multi-Protocol Label Switching (MPLS) signalling required to support Generalized MPLS. This document presents a functional description of the extensions. Protocol specific formats and mechanisms, and technology specific details are specified in separate documents.

- *RFC3473: Generalized Multi-Protocol Label Switching (GMPLS) Signalling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions*
  This document describes extensions to RSVP-TE required to support GMPLS procedures.

- *RFC4202: Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*
  This document specifies routing extensions in support of carrying link state information for GMPLS. This document enhances the routing extensions required to support MPLS Traffic Engineering (TE).

- *RFC4203: Routing OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*
  This document specifies extensions to the OSPF routing protocol in support of carrying link state information for GMPLS. This document defines the enhancements to the Traffic Engineering (TE) properties of GMPLS TE links which can be announced in OSPF TE LSAs.

- *RFC4204: Link Management Protocol (LMP)*
  This document specifies a Link Management Protocol (LMP) that runs between a pair of nodes and is used to manage TE links. Specifically, LMP will be used to maintain control channel connectivity, verify the physical connectivity of the data links, correlate the link property information, suppress downstream alarms, and localize link failures for protection/restoration purposes in multiple kinds of networks. The

management of TE links is not restricted to in-band messaging, but instead can be done using out-of-band techniques.

- *RFC4208: Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model*
  This memo addresses the application of GMPLS to the overlay model. In this model, the core-nodes act more as a closed system. The edge-nodes do not participate in the routing protocol instance that runs among the core nodes; in particular, the edge-nodes are unaware of the topology of the core-nodes. There may, however, be a routing protocol interaction between a core-node and an edge-node for the exchange of reachability information to other edge-nodes..

- *RFC4397: A Lexicography for the Interpretation of Generalized Multiprotocol Label Switching (GMPLS) Terminology within The Context of the ITU-T's Automatically Switched Optical Network (ASON) Architecture*
  This document provides a lexicography for the interpretation of GMPLS terminology within the context of the ASON architecture. It is important to note that GMPLS is applicable in a wider set of contexts than just ASON and the definitions presented in this document do not provide exclusive or complete interpretations of GMPLS concepts.

- *RFC4652: Evaluation of Existing Routing Protocols against Automatic Switched Optical Network (ASON) Routing Requirements*
  This document provides an evaluation of the IETF Routing Protocols against the routing requirements for an Automatically Switched Optical Network (ASON) as defined by ITU-T.

- *RFC4726: A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering*
  This document provides a framework for establishing and controlling Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineered (TE) Label Switched Paths (LSPs) in multi-domain networks. In GMPLS, a domain is considered to be any collection of network elements within a common sphere of address management or path computational responsibility. Examples of such domains include Interior Gateway Protocol (IGP) areas and Autonomous Systems (AS-es).

- *draft-ietf-ccamp-inter-domain-rsvp-te: Inter domain MPLS and GMPLS Traffic Engineering - RSVP-TE extensions*
  This document describes procedures and protocol extensions for the use of RSVP-TE signaling in MPLS packet networks and GMPLS packet and non-packet networks, to support the establishment and maintenance of Label Switched Paths that cross domain boundaries.

- *draft-ietf-ccamp-lsp-stitching: Label Switched Path Stitching with Generalized MPLS Traffic Engineering*
  This document describes a method to combine one or more GMPLS Label Switched Paths to form a single end-to-end LSP. Constituent LSPs are referred to as "LSP segments" (S-LSPs). It may be possible to configure a GMPLS node to switch the traffic from an LSP for which it is the egress, to another LSP for which it is the ingress, without requiring any signaling or routing extensions.

- *draft-ietf-ccamp-inter-domain-pd-path-comp-03.txt: A Per-domain path computation method for establishing Inter-domain Traffic Engineering (TE) Label Switched Paths (LSPs)*
  This document specifies a per-domain path computation technique for establishing inter-domain Traffic Engineering (TE) LSPs. Per-domain computation applies where the full path of an inter-domain TE LSP cannot be or is not determined at the ingress node of the TE LSP, and is not signalled across domain boundaries. This is most likely to arise owing to TE visibility limitations.

- *draft-ietf-ccamp-gmpls-rsvp-te-call-03.txt: Generalized MPLS (GMPLS) RSVP-TE Signaling Extensions in support of Calls*
  This document describes a method for establishing and maintaining associations between endpoints and key transit points in order to support an instance of a service. Such associations are known as Calls. A Call does not provide the actual connectivity for transmitting user traffic, but only builds a relationship by which subsequent Connections may be made. In GMPLS such Connections are known as Label Switched Paths (LSPs). This document specifies how GMPLS RSVP-TE signaling may be used and extended to support Calls. These mechanisms provide full and logical Call/Connection separation.

- *draft-ietf-ccamp-gmpls-ason-routing-ospf-02.txt: OSPFv2 Routing Protocols Extensions for ASON Routing*
  This document provides the extensions of the OSPFv2 Link State Routing Protocol to meet the routing requirements for an ASON as defined by ITU-T. The proposed extensions differ from those defined in the OIF E-NNI routing Implementation Agreement described in the next section.

There exist several more RFCs and Internet-Drafts within the CCAMP that may have some relationship to the PHOSPHORUS work. They deal with:

- Transport-specific extensions for GMPLS signaling and routing: RFC3946, RFC4139, RFC4257, RFC4258, RFC4328, RFC4606, draft-ietf-ccamp-gmpls-vcat-lcas-01.txt
- Transport-specific extensions/profiles for LMP: RFC4207, RFC4209
- Protection/Recovery/Restoration mechanisms: RFC4426, RFC4427, RFC4428, draft-ietf-ccamp-gmpls-recovery-e2e-signaling-04.txt

3.1.2.2 *L1VPN Working Group*

The Layer 1 Virtual Private Network (L1VPN) Working Group is chartered to develop standards and mechanisms for establishing Layer-1 VPN services over GMPLS-enabled transport service provider networks. The L1VPN WG defines the reference network model for delivery of L1VPN services, and provides functional specifications for the operation of path setup signaling as well as methods for limited sharing of information at the CE-PE interface in support of auto-discovery and sharing of reachability information.

The following sections provide brief descriptions of the core L1VPN standards documents which may have relevance to the Phosphorus control plane architecture.

- *draft-ietf-l1vpn-framework-05.txt: Framework and Requirements for Layer 1 Virtual Private Networks*
  This document provides a framework and service level requirements for Layer 1 Virtual Private Networks (L1VPNs). The document examines motivations for L1VPNs, high level (service level) requirements, and outlines some of the architectural models that might be used to build L1VPNs.

- *draft-ietf-l1vpn-basic-mode-01.txt: Layer 1 VPN Basic Mode*
  This draft describes the basic mode of Layer 1 VPNs (L1VPN BM) that is port based VPNs. This draft defines the operational model using either provisioning or a VPN auto-discovery mechanism and the signaling extensions for the L1VPN BM.

- *draft-ietf-l1vpn-ospf-auto-discovery-01.txt: OSPF Based L1VPN Auto-Discovery*
  This document defines an OSPF based layer-1 VPN auto-discovery mechanism. This mechanism enables PEs using the OSPF IGP to dynamically learn about existence of each other, and attributes of currently configured CE-PE links and their associations with L1VPNs.

- *draft-ietf-l1vpn-applicability-enhanced-mode-00.txt: Applicability analysis of Generalized Multiprotocol Label Switching (GMPLS) protocols for the Layer 1 Virtual Private Network (L1VPN) Enhanced Mode*
  This document provides an applicability analysis on the use of Generalized Multiprotocol Label Switching (GMPLS) protocols and mechanisms to satisfy the requirements of the Layer 1 Virtual Private Network (L1VPN) Enhanced Mode. L1VPNs provide customer services and connectivity at layer 1 over layer 1 networks.

- *draft-ietf-l1vpn-applicability-basic-mode-00.txt: Applicability Statement for Layer 1 Virtual Private Networks (L1VPNs) Basic Mode*

This document provides an applicability statement on the use of Generalized Multiprotocol Label Switching (GMPLS) protocols and mechanisms to support Basic Mode Layer 1 Virtual Private Networks (L1VPNs).

3.1.2.3 *L2VPN Working Group*

The Layer 2 Virtual Private Network (L2VPN) Working Group is chartered to specify solutions for supporting provider-provisioned Layer 2 Virtual Private Networks (L2VPNs). This WG is largely responsible for standardizing the Virtual Private LAN Service (VPLS), an L2 service which emulates a shared LAN across an MPLS-enabled IP network, and the Virtual Private Wire Service (VPWS), an L2 service which emulates point-to-point service across similar networks.

PHOSPHORUS users may wish to request more than just simple point-to-point service, at which point the PHOSPHORUS Control Plane will need to use an emulation method for implementing a shared transport media; the VPLS service defined by this working group could be a solution for this issue.

The following list details the standards documents applicable to establishment of L2VPN services.

- *RFC4664: Framework for Layer 2 Virtual Private Networks*
  This document provides a framework for Layer 2 Provider Provisioned Virtual Private Networks (L2VPNs). This framework is intended to aid in standardizing protocols and mechanisms to support interoperable L2VPNs.

- *RFC4665: Service Requirements for Layer 2 Provider Provisioned Virtual Private Networks*
  This document provides requirements for Layer 2 Provider-Provisioned Virtual Private Networks (L2VPNs). It first provides taxonomy and terminology and states generic and general service requirements. It covers point-to-point VPNs, referred to as Virtual Private Wire Service (VPWS), as well as multipoint-to-multipoint VPNs, also known as Virtual Private LAN Service (VPLS). Detailed requirements are expressed from both a customer as well as a service provider perspectives.

- *draft-ietf-l2vpn-vpls-ldp-09.txt: Virtual Private LAN Services Using LDP*
  This document describes a VPLS solution using pseudo-wires, a service previously implemented over other tunnelling technologies and known as Transparent LAN Services (TLS). A VPLS creates an emulated LAN segment for a given set of users, i.e. it creates a Layer 2 broadcast domain that is fully capable of learning and forwarding on Ethernet MAC addresses that is closed to a given set of users. Multiple VPLS services can be supported from a single PE node.

- *draft-ietf-l2vpn-signaling-08.txt: Provisioning, Autodiscovery, and Signaling in L2VPNs*
  This document specifies a number of L2VPN provisioning models, and further specifies the semantic structure of the endpoint identifiers required by each model. It discusses the distribution of these identifiers by the discovery process, especially when discovery is based on the Border Gateway Protocol (BGP). It then specifies how the endpoint identifiers are carried in the two signaling protocols that are used to set up PWs, the Label Distribution Protocol (LDP) and the Layer 2 Tunneling Protocol (L2TPv3).

3.1.2.4 *MPLS Working Group*

The Multi-Protocol Label Switching (MPLS) Working Group is chartered to standardize the base MPLS architecture, protocols and data plane specification. The MPLS standards are the foundation from which the GMPLS work within the CCAMP working group is derived. The MPLS WG has defined a broad range of documents; those included here are perhaps the most relevant to GMPLS implementations as a whole.

- *RFC3031: Multiprotocol Label Switching Architecture*
  This document specifies the architecture for Multiprotocol Label Switching (MPLS).

- *RFC3209: RSVP-TE: Extensions to RSVP for LSP Tunnels*
  This document describes the use of RSVP to establish label-switched paths (LSPs) in MPLS. Since the flow along an LSP is completely identified by the label applied at the ingress node of the path, these paths may be treated as tunnels.

- *RFC4377: Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)*
  This document defines procedures and extensions to RSVP-TE that are needed in order to support unnumbered links, needed fro improving scalability.

- *RFC4206: Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)*
  This document describes basic mechanisms for establishing hierarchies of TE LSPs. To improve scalability of GMPLS, it may be useful to aggregate LSPs by creating a hierarchy of such LSPs. A way to create such a hierarchy is by (a) a Label Switching Router (LSR) creating a Traffic Engineering Label Switched Path (TE LSP), (b) the LSR forming a forwarding adjacency (FA) out of that LSP (by advertising this LSP as a Traffic Engineering (TE) link into the same instance of ISIS/OSPF as the one that was used to create the LSP), (c) allowing other LSRs to use FAs for their path computation, and (d) nesting of LSPs originated by other LSRs into that LSP (by using the label stack construct).

- *RFC4420: Encoding of Attributes for Multiprotocol Label Switching (MPLS) Label Switched Path (LSP) Establishment Using Resource ReserVation Protocol-Traffic Engineering (RSVP-TE)*
  This document defines a new object for RSVP-TE messages that allows the signaling of attribute bits beyond those already defined within the SESSION_ATTRIBUTE object. This document also describes the carriage of arbitrary attribute parameters to make RSVP-TE easily extensible to support new requirements.

3.1.2.5 *OSPF Working Group*

The Open Shortest Path First (OSPF) Working Group is responsible for designing and developing the OSPF protocol, its extensions and profiles, and documenting proper scenarios for deployment and use. As with the work product of the MPLS working group, the work product of this working group form the foundation from which the GMPLS work within the CCAMP working group is derived, and thus is relevant background material for architects and implementers of GMPLS-enabled systems.

The OSPF WG has defined a broad range of documents; those included here are perhaps the most relevant to GMPLS:

- *RFC2328: OSPF Version 2*
  This memo documents version 2 of the OSPF protocol. OSPF is a link-state routing protocol, which is designed to be run internal to a single Autonomous System. Each OSPF router maintains an identical database describing the Autonomous System's topology. From this database, a routing table is calculated by constructing a shortest-path tree.

- *RFC2370: The OSPF Opaque LSA Option*
  This memo defines enhancements to the OSPF protocol to support a new class of link-state advertisements (LSA) called Opaque LSAs. Opaque LSAs provide a generalized mechanism to allow for the future extensibility of OSPF. Opaque LSAs consist of a standard LSA header followed by

application-specific information. The information field may be used directly by OSPF or by other applications. Standard OSPF link-state database flooding mechanisms are used to distribute Opaque LSAs to all or some limited portion of the OSPF topology.

- *RFC3630: Traffic Engineering (TE) Extensions to OSPF Version 2*
  This document describes extensions to the OSPF protocol version 2 to support intra-area Traffic Engineering (TE), using Opaque Link State Advertisements.

3.1.2.6 *PCE Working Group*

The Path Computation Element (PCE) Working Group is chartered to develop a standard Path Computation Element (PCE) based architecture for computation of paths used by MPLS and GMPLS Traffic Engineering LSPs. This working group is specifically defining a PCE model for operating within either a single domain or a modest-sized group of domains (where a domain can be a layer, IGP area of Autonomous System with limited visibility).

As the PHOSPHORUS effort also wishes to provide methods for computing end-to-end paths across a modest number of separately-administered domains, the work of PCE WG is particularly relevant.

The following sections describe some of the existing standards documents relevant to this effort.

- *RFC4655: A Path Computation Element (PCE) Based Architecture*
  This document specifies the architecture for a Path Computation Element (PCE)-based model to address this problem space.

- *RFC4657: Path Computation Element (PCE) Communication Protocol Generic Requirements*
  According to the PCE model, path computation requests are issued by Path Computation Clients (PCCs) towards Path Computation Elements (PCEs). This document specifies generic requirements for a communication protocol between PCCs and PCEs, and also between PCEs where cooperation between PCEs is desirable.

- *RFC4674: Requirements for Path Computation Element (PCE) Discovery*
  This document presents a set of requirements for a Path Computation Element (PCE) discovery mechanism that would allow a Path Computation Client (PCC) to discover dynamically and automatically a set of PCEs along with certain information relevant for PCE selection. It is intended that solutions that specify procedures and protocols or extensions to existing protocols for such PCE discovery satisfy these requirements.

- *draft-ietf-pce-pcep-05.txt: Path Computation Element (PCE) Communication Protocol (PCEP) - Version 1*
  This document specifies the Path Computation Element communication Protocol (PCEP) for communications between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between two PCEs. Such interactions include path computation requests and path computation replies

as well as notifications of specific states related to the use of a PCE in the context of MPLS and GMPLS Traffic Engineering.

- *draft-ietf-pce-disco-proto-ospf-01.txt: OSPF protocol extensions for Path Computation Element (PCE) Discovery*
  This document describes a method by which a Path Computation Client (PCC) can be enabled to dynamically and automatically discover a set of Path Computation Element(s) (PCE), along with some of information that can be used for PCE selection. When the PCE is a Label Switch Router (LSR) participating to the IGP, or even a server participating passively to the IGP, a simple and efficient way for PCE discovery consists of relying on IGP flooding. For that purpose this document defines OSPF extensions for the advertisement of PCE Discovery information within an OSPF area or within the entire OSPF routing domain.

- *draft-ietf-pce-pcep-interarea-reqs-05.txt: PCE Communication Protocol (PCECP) Specific Requirements for Inter-Area Multi Protocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering*
  This document describes the operation of Path Computation Elements in networks comprised of multiple Interior Gateway Protocol (IGP) areas. An inter-area Traffic Engineered-Label Switched Path (TE-LSP) is an LSP that transits through at least two IGP areas. In a multi-area network, topology visibility remains local to a given area, and a head-end Label Switching Router (LSR) cannot compute an inter-area shortest constrained path. One key application of the Path Computation Element (PCE) based architecture is the computation of inter-area TE-LSP paths.

- *draft-ietf-pce-brpc-03.txt: A Backward Recursive PCE-based Computation (BRPC) procedure to compute shortest inter-domain Traffic Engineering Label Switched Paths*
  This document defines a method for computing constrained shortest Traffic Engineering LSPs in MPLS and GMPLS networks across multiple domains. The procedure relies on the use of multiple Path Computation Elements (PCEs) in order to compute such inter-domain shortest constraint paths along a determined sequence of domains, using a backward recursive path computation technique while preserving confidentiality across domains.

3.1.3 OIF UNI and E-NNI

Within the Optical Internetworking Forum (OIF), the User Network Interface (UNI) [OIF-UNI-01.0] and External Network Node Interface (E-NNI) [OIF-E-NNI-Sig-01.0, OIF-E-NNI-Rtr-1.0] are necessitated interfaces to develop interoperable procedures for requesting and establishing dynamic connectivity across heterogeneous networks defined by the automatic switched transport network (ASTN).

3.1.3.1 *UNI*

UNI is defined as the service control interface between the user devices and the transport network equipment from different vendors. It addresses the definitions of connectivity services offered by the transport network, the signalling protocols used to invoke the services, the mechanisms used to transport signaling messages, and the auto-discovery procedures that aid signalling.

OIF UNI 1.0 is scoped to allow an early implementation based on reusing existing signalling protocols and auto-discovery mechanisms taking into consideration work of other standards bodies such as IETF proposed standards. It is a result of cooperation between network and equipment vendors to improve carriers' ability to provide new services while reducing operating cost. OIF UNI 1.0 scope is only on the signalling for service invocation; routing, reachability, and address resolution protocols are outside the scope. It particularly focuses on the ability to create and delete point-to-point transport network connections on-demand including bandwidth, signal type, and routing constraints (node, link or SRLG diversity). OIF UNI 1.0 also specifies methods for neighbour and service discovery, provides policy and security considerations. UNI signalling can be between the network elements. It is also possible for one or both of them to be represented by a proxy.

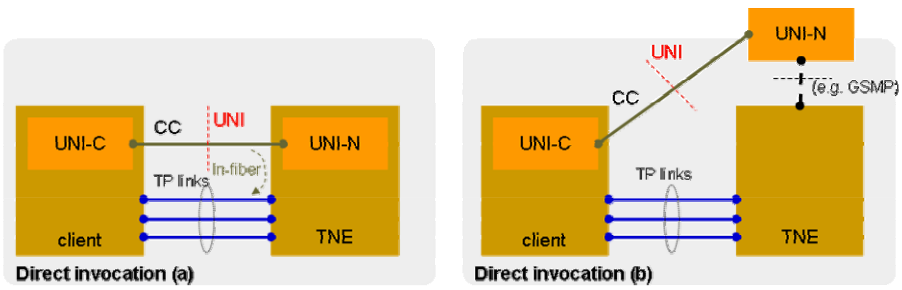


Figure 3-9: OIF direct invocation model: without (a) and with UNI-N proxy (b).

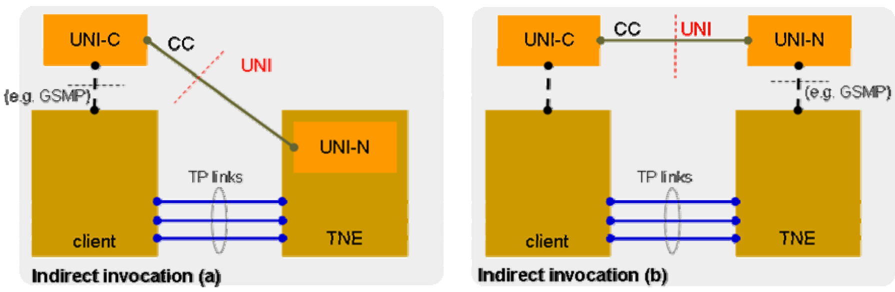


Figure 3-10: OIF indirect invocation model: with (a) and without UNI-C proxy (b).

The procedures supported across the UNI are listed below:

- Association/Connection establishment (signaling): It identifies the creation of a connection
- Association/Connection deletion (signalling): It represents the destruction removal of a connection.
- Status exchange (signalling): the result of status exchange is the discovery of connection status.
- Auto-discovery (signalling): It provides the discovery of connectivity between the client and the network and the services available from the network
- Use (traffic): It is used to exchange user information over a connection (i.e. traffic exchange)

The OIF-UNI-1.0 Implementation Agreement deals with the following issues:

- UNI signalling reference configurations
- Services offered over the UNI
  - Signalling actions
    - Connection Creation
    - Connection Deletion
    - Connection status enquiry
  - Connection types which is defined by:
    - Framing (e.g. SONET/SDH)
    - Transparency
      (1) Section transparency
      (2) Line transparency
      (3) Path transparency
    - Signal Types
    - Concatenation
      (1) Contiguous concatenation
      (2) Virtual concatenation
  - Supporting Procedures
  -
    - Neighbour Discovery, i.e. one of the two parties initiates a procedure of link connectivity verification based on LMP.
    - Service Discovery, i.e. the UNI-C indicates the capabilities of the client device that represents to, and obtains information concerning transport network services from the UNI-N by utilizing a number of service attributes.
    - Signalling Control Channel Maintenance

UNI procedures are divided into mandatory and optional and thus a UNI compliant device must implement all the mandatory features and their corresponding protocol implementations. These procedures are categorized on the following table.

UNI 1.0 Procedure	Status	Minimum Acceptable Implementation
Signaling	Mandatory	RSVP-TE or LDP
Control Channel Realization	Mandatory	Out-of-fiber, IP network
Framing for in-fiber packet data communication over DCC	Mandatory with In-fiber Section or Line DCC-based control channel realization	At least one of these
Control Channel Maintenance	Mandatory	LMP procedure for control channel maintenance for the selected control channel realization
In-fiber neighbor discovery	Optional	Manual configuration
Out-of-fiber neighbor discovery	Optional	Manual configuration
Service Discovery	Optional	Manual configuration

Table 3-2: OIF UNI 1.0 mandatory and optional procedures.

Allowed configurations of signalling transport can be in-fiber or out-of-fiber.

A detailed description of UNI abstract messages is also provided by OIF-UNI-1.0. The level of description is abstract due to the two possible signalling protocols (LDP and RSVP) that can be used being properly extended for UNI signalling.

The second implementation agreement (IA) of UNI1.0 is an update [OIF-UNI1.0R2-COMM, OIF-UNI1.0R2-RSVP] that address mainly extensions to RSVP-TE signalling protocols. Two more IAs the CDR-01 and the SEP-0.1.1 have been released to support OIF UNI 1.0. The first one outlines Call Detail Records (CDR) for OIF UNI 1.0 Billing as an agreement on CDR procedures and formats to allow both carriers and suppliers to capture usage records on optical connection thus offering usage-based billing optical services. On the other hand, OIF-SEP-0.1.1 is focused on Security Extensions for UNI and NNI in order to provide a common set of security mechanisms required to protect the UNI signalling control channel(s).

OIF work has been continued towards UNI 2.0, which is still in the OIF acceptance stage [OIF-UNI2.0-COMM, OIF-UNI2.0-RSVP]. OIF UNI 2.0 definition is planning to include OTN (G.709-based) and Ethernet connections. The notable features under consideration of UNI 2.0 include separation of Call and Connection Control, sub STS-1 Rate Connections, transport of Ethernet services, transport of G.709 interfaces and enhanced security. UNI 2.0 will follow the call and connection architecture from ASON as described in ITU-T G.8080 and G.7713.

The UNI could legitimately be extended in a number of other dimensions. Within the OIF, for example, proposals have been made to allow the client to specify the routing to use and to pass some topology information from the network to the client. The applications in mind were for a private UNI (in the sense discussed above). A limited NNI capability, suitable for a "private NNI" such as might be needed for a L1VPN, was identified as an application of this sort of UNI.

IETF GMPLS UNI Model

IETF GMPLS can operate in an overlay-style model, in which the core nodes act more as a closed system and the edge-nodes do not participate in the routing of the core nodes. Despite IETF-RFC3945 provides a number of reachability models at the UNI, ranging from "Configuration-based", to "Partial peering", "Silent Listening" and "Full Peering", the IETF UNI model implements just the "Configuration-based" and edge-nodes are unaware of the topology of the core-nodes. The main focus of the IETF GMPLS UNI is the RSVP-TE procedures between edge-node and core-node in the overlay model For the purpose of end-to-end reachability some work has been started in IETF under the L1VPN scope.

3.1.3.2 *E-NNI*

E-NNI signalling [OIF-E-NNI-Sig-1.0, OIF-E-NNI-Sig-2.0]

The overall mission of the OIF E-NNI Signalling IA is the implementation of an open inter-domain signal protocol that enables dynamic setup and release of various services. The focus of the E-NNI IA is the specification of abstract signalling messages, attributes and flows for the purpose of enabling end-to-end dynamic establishment of transport connections across multiple control domains which applies to SDH/SONET connection services. E-NNI supports connectivity services, architectures and reference configurations as well as Control Plane addressing and distributed signalling communications. Routing, reachability, address resolution protocols, auto-discovery, policy and security are outside the scope of this IA.

E-NNI signalling is required to support Switched Connection (SC) services as well as Soft Permanent Connection (SFC) services. Service-related events as a call attribute should are supported by the E-NNI signalling messages. Explicit route information associated with E-NNI gateways within the signalling message should be passed across E-NNI. In latter case where a SPC should be established E-NNI, the call parameters originating from the source network must be preserved and passed across the E-NNI.

E-NNI exists between two signalling control domains. The upstream protocol which transmits a call request and the downstream protocol controller that serves as receiver for the request. The upstream protocol controller is referred to as eNNI-C and the downstream protocol controller as eNNI-D. Such a controller could include both network call controller (NCC) and connection controller (CC) functionality defined in G.8080.

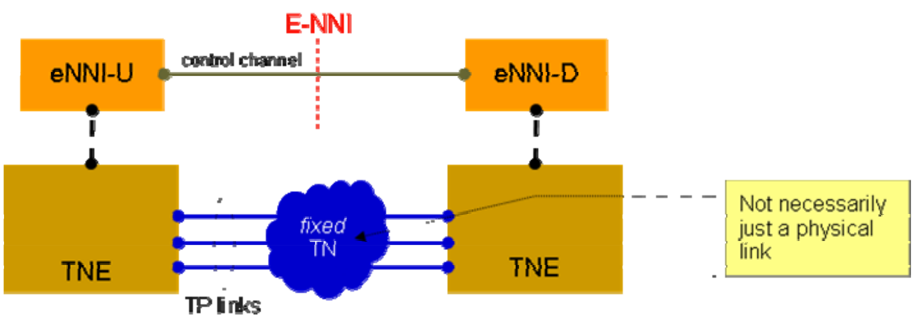


Figure 3-11: OIF E-NNI signalling reference model.

Signalling identifiers are presented in three different categories such as Transport Plane names, Control Plane identifiers for components and Signalling Communication Network (SCN) addresses. The Transport Plane names utilises two separate names; a) the SNPP (Sub Network Point Pool) names provide a link context for SNPs (Sub Network Point) and are used by the Control Plane to identify Transport Plane resources, b) Transport Network Assigned (TNA) names which are used to identify transport resources at UNI reference point if transport resources exist and represent the clients of the transport network, not the transport network endpoints. In addition to these, Signalling Protocol Controller Identifiers are used to describe Control Plane resources. Separate identifiers are used for Connection Controllers (CCs) and Network Call Controllers (NCCs). Signalling Protocol Controller SCN Addresses are also identified for Control Plane components use to communicate.

The issues addressed and reference configurations described in OIF-E-NNI signalling IAs incorporate the followings:

- Supported services over the E-NNI
  - Switched Connections (SCs)
  - Soft Permanent Connections (SPCs)
  - Call and connection separation
- Signalling reference configurations
- Signalling Identifiers
- Control Plane Transport Network described the need for a resilient signalling communication network (SCN) architecture that enable interactions of the protocols running on communication Control Plane nodes
- Compatibility with call/connection control originating from or destined to a UNI 1.0r2/2.0 interface.
- E-NNI signal flow that describes two cases
  - Signal flow for normal setup and release requests, which includes setup and release for both switched connections and soft permanent connections.
  - Signal flow for exception and defect handling as a result of a decision to reject a request.

E-NNI routing [OIF-E-NNI-Rtr-1.0]

In order to support E-NNI signalling with explicit routes including inter-domain paths and to comply with ASON routing defined in ITU-T G.7715 and G.7715.1, OIF produced a specific IA that defines the framework for inter-domain routing and new OSPF-TE routing information exchanges.

The E-NNI routing IA focuses on intra-carrier operations, because the inter-carrier support raises critical issues on AAA and security that are still under discussion.

The routing model that inspires the E-NNI routing is the ITU G.8080 and G.7715 hierarchical model, in which I-NNI routing control domains represent the base layer and each subsequent E-NNI layering above constitutes a hierarchical level. At each hierarchical level, one OSPF instance and area is defined (as per G.7715).

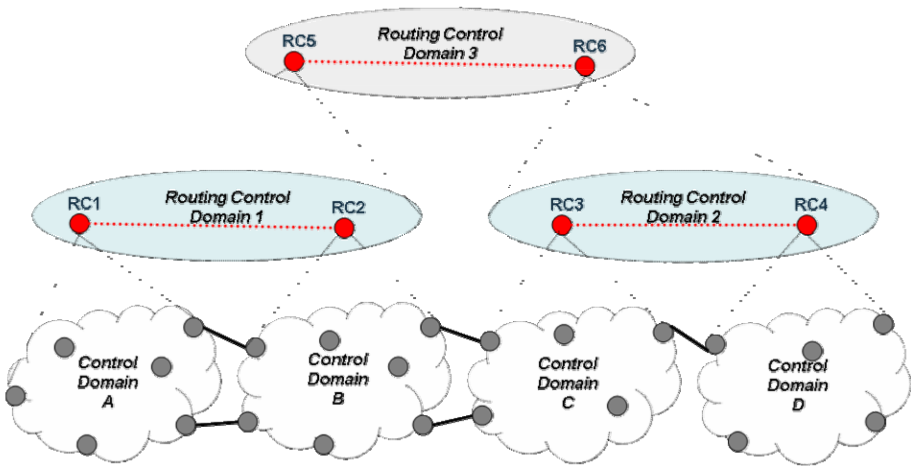


Figure 3-12: OIF E-NNI hierarchical routing.

In each Routing Control Domain (RCD) just one Routing Controller (RC) is defined; the RCD is identified by its RC ID. Neighbouring RCDs are identified through their RC ID and the related (IPv4) SCN address.

E-NNI routing IA has the main purpose of flooding information about:

- inter-domain TE links, between Area Border Routers (ABR) belonging to different base Control Domains (i.e. OSPF-TE areas)
- TNA addresses attached to the RCD
- General domain capabilities (transit or not, what technology, etc.)
- Optionally, summarized intra-domain TE-links, i.e. virtual connections abstractions between Area Border Routers (ABR) belonging to the same base Control Domains (i.e. OSPF-TE areas)

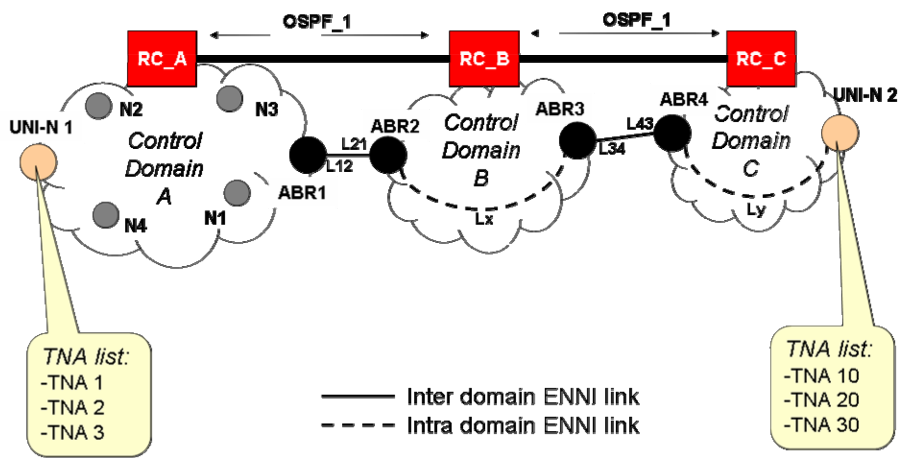


Figure 3-13: OIF E-NNI routing components.

In hierarchical routing the info can flow within a hierarchical level upward / downward across routing hierarchical levels. It order to avoid possible advertisement loops, i.e. the info learnt at level N from a higher level (N+1) can be circulated within level N or propagated down to lower layer, specific OSPF-TE protocol elements (sub-TLVs) have been defined:

- A Hierarchy List with the RC IDs of the visited levels downward
- An Ancestor RC element in TE link advertisements, specifying the level where the referred link is an intra-domain link: lower than that level, it is a border link, higher than that level it is a intra-domain link (summarized)

In E-NNI routing no OSPF multi-area operations are used and the routing information exchanged over the E-NNI routing adjacency should only concern the Transport Network topology. This implies that IP-specific LSAs (e.g. Router LSAs, Summary Network LSAs, etc.) are forbidden and just opaque LSA of type 10 (area-scoped) are flooded.

Unlike traditional OSPF routers that are usually physically adjacent, the E-NNI RCs that would like to form adjacencies are most likely not topologically adjacent within the control plane. In order to let them create one-hop adjacencies, a variety of methods could be used, such as tunnelling (e.g. GRE, IP-in-IP and IPSec), Layer 2 VLANs, and OSPF virtual links. Each method expose its own limitations: for example, VLANs can only be applied within SCNs consisting of a single Ethernet broadcast domain; virtual links are an optional capability of OSPF restricted to the OSPF backbone area, etc. OIF E-NNI promotes the use of the OSPF point-to-multipoint method, which consists in creating adjacencies by configuration by shutting down the automatic OSPF Hello mechanism.

3.1.4 OGF OGSA

Within the Open Grid Forum (OGF), the Open Grid Services Architecture (OGSA) is a service-oriented framework for distributed, heterogeneous, dynamic "virtual organizations". "*Building on concepts and technologies from the Grid and Web Services communities, this architecture defines a uniform exposed service semantics (the Grid Service); defines standard mechanisms for creating, naming and discovering transient Grid service instances; provides location transparency and multiple protocol bindings for service instances; and supports integration with underlying native platform facilities*" [GRID-PHYSIOLOGY]. The fundamental OGSA concept is that of the Grid Service by building on Web Services. Web Service framework technologies provide a simple way to describe, encapsulate, advertise and access a service.

Web Services implement standard interfaces, behaviours, and conventions that collectively allow for services that can be transient (i.e., can be created and destroyed) and stateful (i.e., service instances are distinguished). The Web Services model is based on two simple technologies:

- The Web Services Description Language (WSDL) [W3C-WSDL] network end-points or ports. WSDL addresses this need by defining an XML schema for describing network services as collections of communication end points capable of exchanging messages.
- The Universal Description, Discovery and Integration (UDDI) [UDDI] and the Web Services Inspection Language (WSIL) [WSIL] provide the mechanism needed to discover WSDL documents. UDDI is a specification for a registry that can be used by a service provider as a place to publish WSDL documents. WSIL provides a simple way to find WSDL documents on a web site.

In addition to these standards there are also several others that have been proposed to provide additional features which are listed below:

- Web Services Flow Language (WSFL) proposed by IBM [WSFL] as a mechanism for scripting the workflow for integrating multiple services together.
- Web Services Invocation Framework (WSIF), which can dynamically generate service proxies as objects that may reference with the language of the client application.

3.1.4.1 *Roadmap for OGSA*

The Open Grid Forum[2] (OGF) has embraced the Open Grid Services Architecture (OGSA) as the blueprint for standards-based grid computing. Open Grid Service Architecture areas are distinguished to architectural process, specifications and profiles and software.

- OGSA architectural process works to collect requirements, evaluate the maturity of specifications and produce periodic updates to the following OGSA informational documents:

[2] Formerly Global Grid Forum (GGF).

- OGSA Use Case document
  - End-user application scenarios are provided
- Open Grid Service Architecture, Version 1.0
  - Framework, taxonomy, and functionality to address use case requirements are described
- Service Description document
  - Services in the area of natural language, listing the interfaces and operations defined by each service are analyzed
- Scenario documents involves
  - Service implementation based on use cases combining natural language and UML
- OGSA Roadmap shows
  - Views on the likely future evolution of itself to address unmet requirements and/or respond to technology evolution.

- OGSA specifications and profiles are normative documents.
  - Specification documents describe precise technical requirement (e.g. interfaces, protocols and behaviours, for a hardware of software component)
  - OGSA Profile identifies a set of broadly adopted normative technical specifications that collectively capture current understanding of what software must do to operate and manage interoperable grid environments.
- OGSA software remains to the OGSA normative specification and profiles.
  - Enables customers to deploy grid solutions that interoperate even when based on different open-source or commercial software vendors' implementations

3.1.4.2 *OGSA functional requirements*

The analysis of a number of uses cases such as Commercial Data Center, National Fusion Collaboratory, Severe Storm Prediction, etc. and of other OGS and non-OGF documents helped identify the functional requirements that are needed to support grid environments. Below the specification landscape of OGSA at April 2006 is presented, in which existing, evolving and required services to support Grids are illustrated.

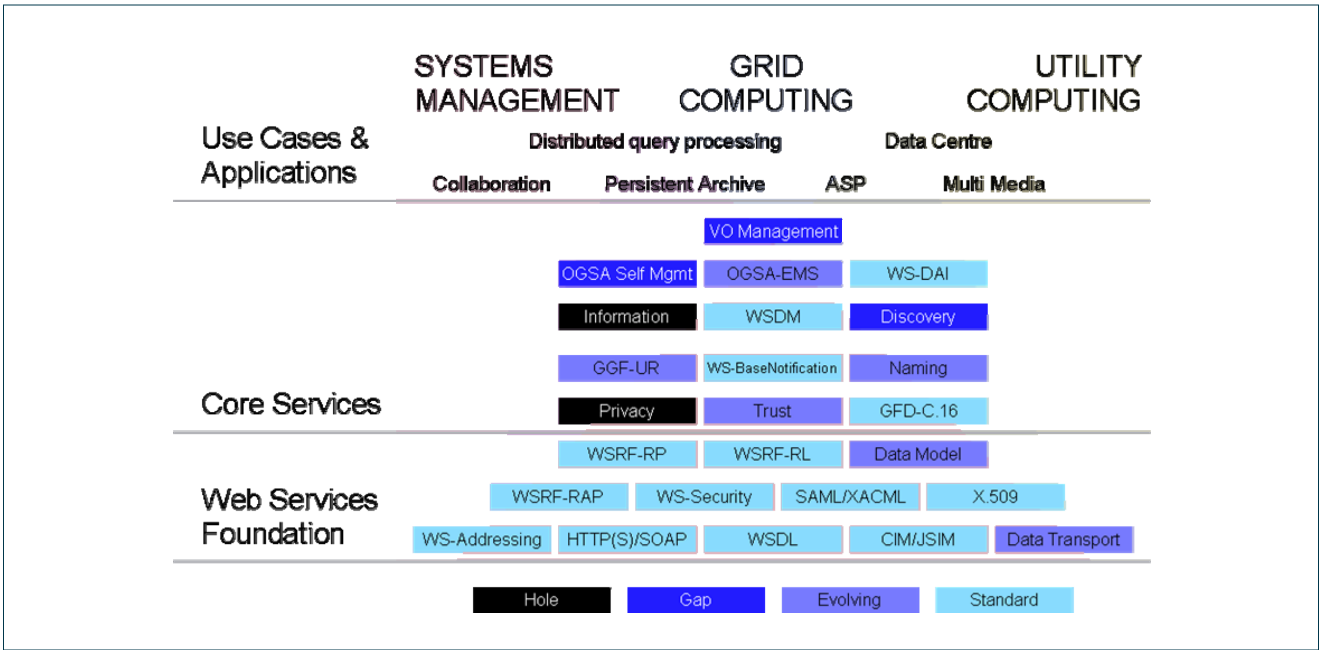


Figure 3-14: Specification landscape of OGSA at April 2006.

3.1.4.3 *OGSA Platform*

The OGSA platform is made up of three components: the Open Grid Service Infrastructure (OGSI), the OGSA Platform interfaces (OGSA services), and the OGSA Platform Models (OGSA schemas). Figure 3-15 illustrates the functional representation of OGSA platform.

OGSI represents the convergence of Web Services and grid technologies. It defines the underlying mechanisms for managing Grid Service instances. OGSA Platform Interfaces are OGSI-compliant Grid Services that are not defined within OGSI. Examples of such services are registries, data access and integration, resource manager interfaces, monitoring, authorization, etc. OGSA Platform Models (OGSA schemas) are representing real entities for Grid.

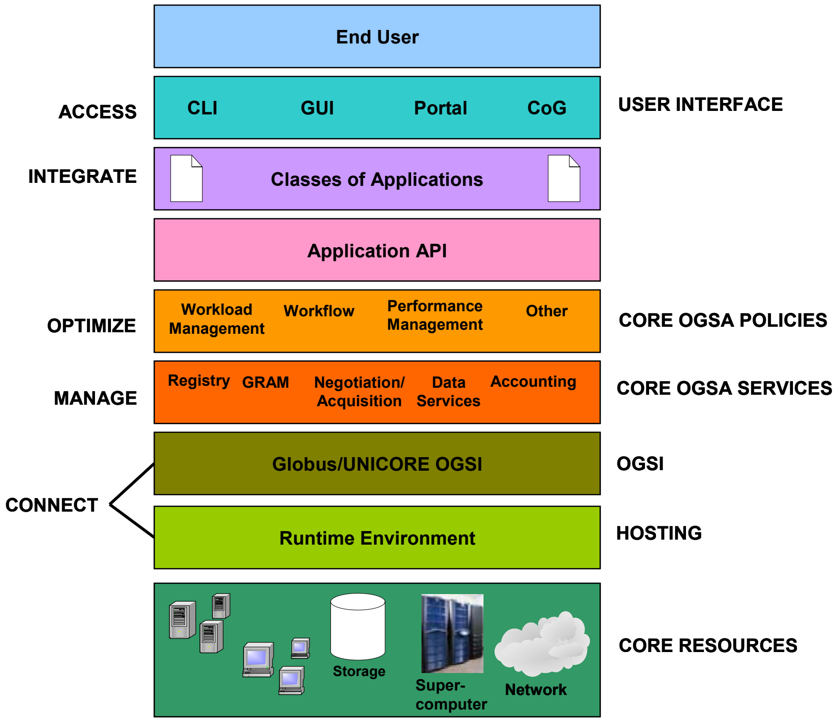


Figure 3-15: OGSA platform functional representation.

OGSA Platform Interfaces (Grid Services)

The OGSA Platform interfaces define a number of functions as Grid Services that commonly occur within Grid systems. The categories that divide these functions are listed and described below.

- *Service Groups and Discovery Interfaces*: The two-level naming defined by OGSI based on GSHs and GSRs provide a way of accessing a known service, but for a number of reasons, a higher level abstraction is necessary to allow clients to find services.
- *Service Domain Interfaces*: A common usage pattern in grid solutions is to create Grid Service collections that together produce a higher order Grid Service interface. This is usually done in order to implement a domain-specific solution based on more generic lower level services.
- *Security*: Security is wide-reaching, encompassing issues relating the management and verification of credentials; privacy and integrity and policy.
- *Policy*: Many Grid Services will need some form of policy management in order to help direct their actions. Policies are needed to guide resources usage as well as incorporate business logic and service level agreements. Policies are implemented or executed within a particular context, such as security, workload manage, and control access to Grid resources.
- *Data Management Systems*: Grid environment has a great variance of data sources, their location relative to consuming services, their types and their lifetimes. This diversity adds much complexity when

Data

Servers

Workstations

trying to integrate multiple data sources for processing or for management. Data management interfaces can be used to provide abstracted details of particular set of data.

- *Messaging and Queuing*: Messaging and queuing functions must be exposed via appropriate OGSA interfaces, which will presumably include a large range of semantic for messaging that are not covered by OGSI.
- *Events*: An event represents a state change in a system that might be of interest of some other party. Standard means of representing, communicating, transforming, reconciling, and recording events are important for interoperability. OASIS Management Protocol TC is working on these mechanisms.
- *Distributed Logging*: Distributed logging can be viewed as a special case of messaging, where one entity generates "log artifacts".
- *Metering and Accounting*: Accounting messages are used to collect, deliver, reconcile, store securely, and manage information about resource usage and/or charges.
- *Transactions*: As grid solutions become deployed to support applications with requirements for the coordination of services and service state, there will be a requirement for a transaction service to do this coordination. The Web Services community is looking at this, and WS-Transactions have been proposed.
- *Grid Service Orchestration*: This refers to the coordination of a set of interacting services in pursuit of performing a single "task". This is also commonly known as workflow. The OGSA Platform is intending to define standard portTypes for launching workflows, as opposed to defining a "one size fits all" workflow language standard. Grid Service factories for different workflow systems can then be defined, allowing clients to use the most appropriate mechanism for their problem domain.

The OGSI and OGSA Platform interfaces provide basic mechanisms fro managing Grid Services and specify some useful patterns for developing grid solutions. However, they do not actually provide any capability for dealing with the real entities (both virtual and physical), which are composed together to solve a particular computing problem. Standard representations of the entities on the Grid need to be defined in order to have some real application. The OGSA Platform then needs some common models for describing and manipulating the real entities that are represented through Grid Services. Common Resource Model (CRM) is currently working to define a model for representing manageable IT resources as Grid Services. Manageable resources can range from hardware, to software systems, to individual batch jobs. CRM describes how management interfaces to these entities are represented as OGSA services. Globus as well as UNICORE can be used to implement OGSA-based Grids.

Open Grid Service Infrastructure (OGSI)

Standard interfaces and semantics are required for common service interactions. Without these it is impossible to build interoperable and reusable components. Such interfaces are provided in OGSA by the OGSI. "The OGSI defines mechanisms for creating, naming, managing lifetime, monitoring, grouping and exchanging information among entities called Grid Services" as mentioned in OGSI document. The features that represent OGSI concepts and interactions are portrayed in Figure 3-15 and explained below.

- *Grid Service Description and Service Instance*: OGSI distinguishes the description of a Grid Service and instances which implement the Grid Service.
- *Service State, Metadata and introspection*: Service data mechanisms are representing metadata and state data related to a service as part of its description, and for accessing that information from a service instance are defined by OGSI. A client application can ask a Grid service instance to return information describing its self (e.g. collection of interfaces that it implements, interface-specification information).
- *Naming and name resolution*: OGSI defines two-level naming scheme for locating Grid Services. An instance may have one or more Grid Service Handle (GSH). The GSH names only one Grid Service instance, globally and for all time, though it is not enough to access the actual Grid Service. This can be resolved into a Grid Service Reference (GSR). GSR is specific to the binding mechanism used by a client to access the service.
- *Grid service lifecycle*: A number of mechanisms for managing the life cycle of a Grid service instance are defined by OGSI. These mechanisms include both explicit destruction and soft-state factories that can be used to create instances implementing specified interfaces.
- *Handling of operation faults*: OGSI defines an XML base type which must be returned in all fault messages from Grid Services. Information defined in this base type includes a description, an originator, a timestamp, a fault cause, a fault code and an extension element for providing any additional information.
- *Service Groups*: Mechanisms of representing and managing groups of service instances may be used for a variety of purposes – e.g. collective information management, collective operations and are all described by OGSI.

3.1.4.4 *Job Submission Description Language (JSDL)*

Job submission Description Language is a language for describing the requirements of computational jobs for submission to resources, particularly in Grid environments, though not restricted to the latter. JSDL wants to provide a standardized language in order to accommodate organizations with a variety of job management systems, which each system has its own language for describing job submission requirements. Automatic interactions between an initial job submitter and intermediary that may further refine the job submission information can be also facilitated by JSDL as shown in Figure 3-16.

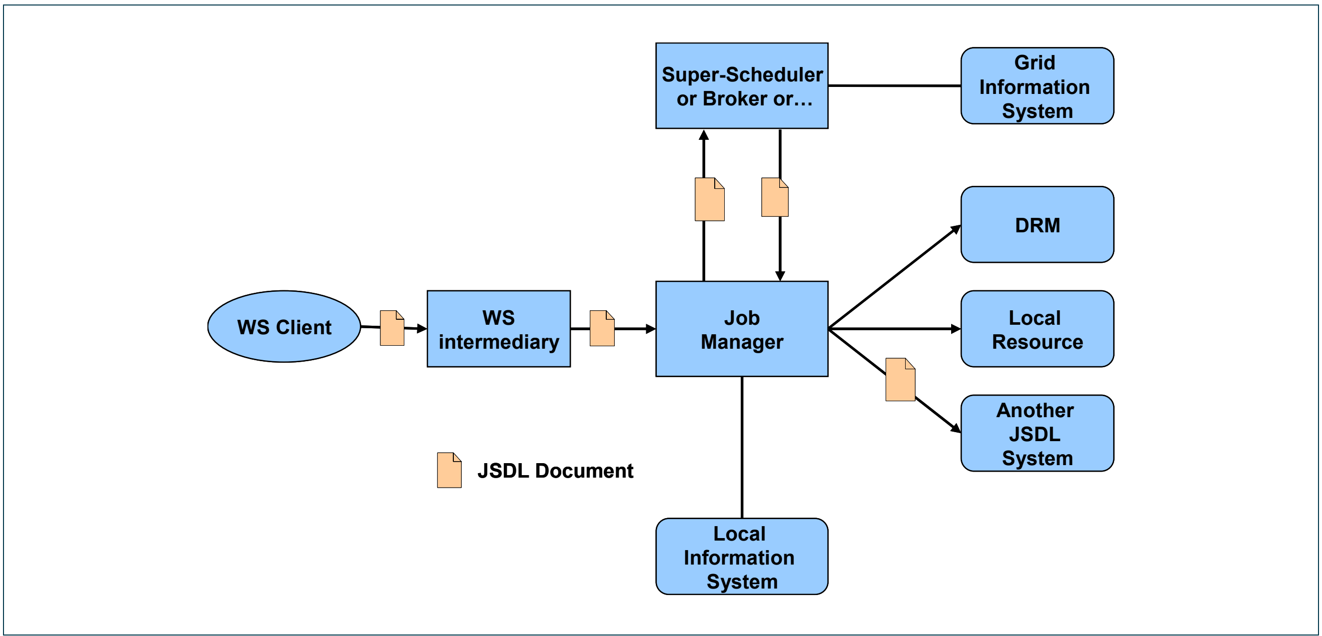


Figure 3-16: JSDL possible positioning on a Grid environment.

JSDL 1.0 provides a core vocabulary for describing job submission of Grid environments which is informed by a number of existing systems such as: Condor, Globus Toolkit, Load Sharing Facility (LSF), Portable Batch System (PBS), (Sun) GridEngine (SGE), Uniform Interface to Computing Resources (Unicore).

Moreover, JSDL is referenced to some standards as well, such as Distributed Resource Management Applications API Specification 1.0 (DRMAA), Common Information Model (CIM), and Portable Operating System Interface (POSIX).

JSDL though does not attempt to address the entire lifecycle of a job or the relationship between individual jobs. Thus a number of other languages and protocols are also required to manage and running jobs that can be described and submitted using JSDL. For example, a workflow language should be used to describe the relationship between individual jobs described using JSDL, and in turn, the relationship of those jobs with the data they consume and produce. The elements which can be described as languages and protocols that are not yet available and are either currently being developed or need to be specified and standardised and described in JSDL document are the following:

- Resource Requirements Language (RRL): Description of resource requirements for a job.
- Scheduling Description Language (SDL): A number of categories that may for example describe scheduling requirements and scheduling of jobs are listed below:
    - Temporal scheduling
    - Data-dependent scheduling
    - Workflow-dependent scheduling
- Web Service Agreement (WS-AG): Agreement between a resource's Web service front-end and a job submission.

- Job Policy Language (JPL): Policy constraints that are applied to any aspect of job execution
- Job Lifetime Management Language (JLML): Describes the state of the job and job control directives

A JSDL document is organized as follows.

- Job Definition
  - Job Description
    - Job Identification
    - JobName
    - Application
      (1) ApplicationName
    - Resources
    - DataStaging
  - any##other Job Description
- any##other Job Definition

3.1.4.5 *GLUE schema*

A key component of the Grid is its Information Service (GIS), which is aimed to offer discovery of existence and characteristics of resources available in a certain moment for subsequent management or use. Description of resources should be abstracted and, thus, applicable to different Grid infrastructures in order to contribute to interoperability among them.

The GLUE Schema (Grid Laboratory Uniform Environment) is an abstract modelling for Grid resources, whose definition started in April 2002 as a collaboration effort between EU-DataTAG and US-iVDGL projects. Current projects that are participating in this activity are EGEE, LCG, Grid3/OSG, Globus and NorduGrid.

GLUE schema provides a description of core Grid resources at the conceptual level by defining an information model that is an abstraction of the real world into constructs that can be represented in computer systems (e.g. objects, properties, behaviour, and relationships). The latest version of the Glue Schema is the v1.3 and it will be used as a starting point for the work of a brand new OGF WG on the definition of a GLUE Schema 2.0 (https://forge.gridforum.org/sf/projects/glue-wg).

The core entities of the GLUES schema include the site concept, an abstraction for the service concept, the Computing Element and the Storage Element as shown in Figure 3-17.

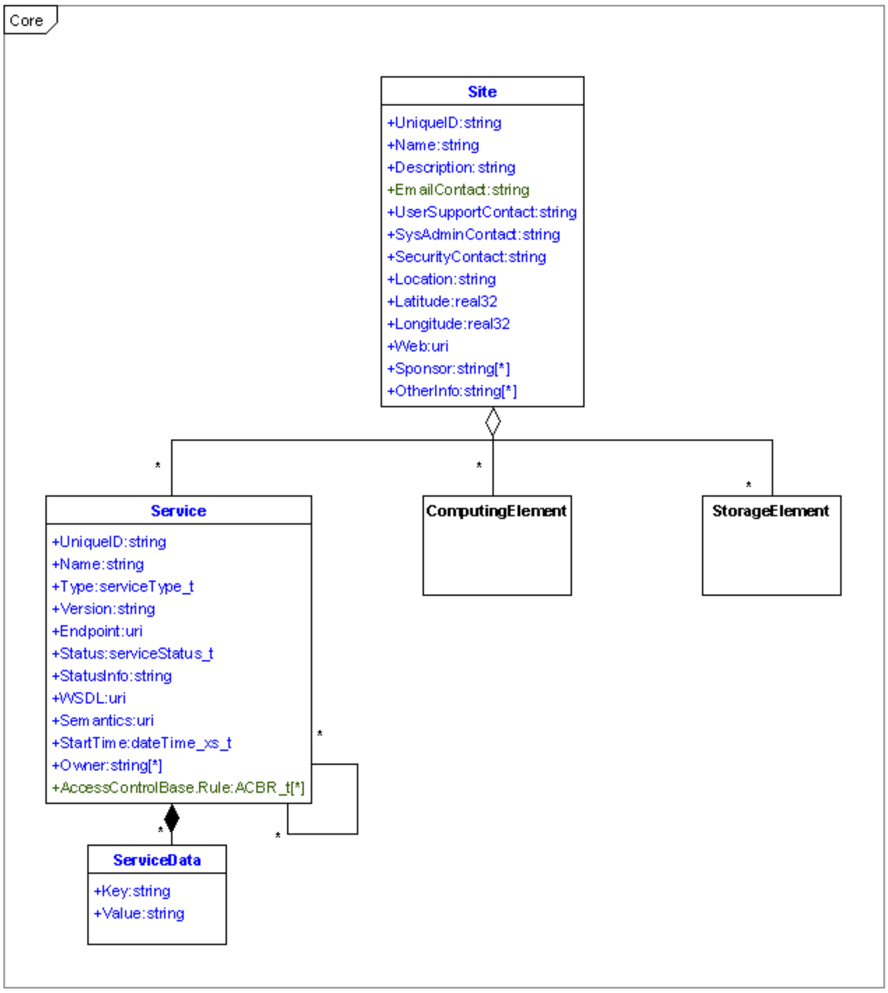


Figure 3-17: GLUE schema core entities.

The site is an administrative concept used to aggregate a set of services and resources that are installed and managed by the same set of persons.

The Service entity captures all the common attributes to Grid Services and should be used as a base entity for the creation of service-specific schemas.

The Computing Element is a base Grid resource possibly bundled in a set of machines locally managed by systems such as the Portable Batch System (PBS), Load Sharing Facility (LSF) or Condor. Computing Elements use queues to stage the requests and can have sets of policies associated. Moreover, they offer scheduling functionalities to impose the fair share of the resources against the set of requests.

Storage resources can vary from simple disk servers managed via GridFTP to complex massive storage systems managed via SRM.

The rationale behind GLUE Schema and its wide adoption in operational Grid infrastructures makes it an attractive information model for the description of Grid resources in the PHOSPHORUS network routing.

3.2 Positioning of G^{2}MPLS

With respect to the standardization framework proposed in Sec. 3.1, G^{2}MPLS Control Plane architecture will tend to adopt an approach with the following main features:

- be a superset of the GMPLS functions and procedures; i.e. some protocol behaviour / procedure available in ASON and/or GMPLS should be available in G^{2}MPLS, as well
- be down-gradable to the behaviour of a legacy ASON and/or GMPLS Control Plane
- be backward compatible with existing ASON and/or GMPLS Control Planes across the UNI and E-NNI

The major architectural reference for G^{2}MPLS Control Plane is the ASON architecture plus OIF IAs, since the administrative partitioning of the networks, the identification of the network reference points (i.e. UNI, I-NNI, E-NNI) and the call-connection procedures are the best suiting the deployment requirements of the NRENs and a Grid-enabled GMPLS.

Obviously, IETF work on GMPLS/MPLS and on the PCE is of utmost interest, since most of the protocol-specific issues were defined and solved in that framework.

In particular, the PCE model and procedures seem suited for a possible implementation of the advance reservations. Similarly, IETF L1VPN and L2VPN work provide an interesting framework for the injection of information from the user side towards the network and vice versa.

The positioning of G^{2}MPLS with respect to OGF work is of reciprocal completion, being the OGF in the study and standardization phase for a possible awareness of Grid and network. OGF pursues this objective through the Grid High Performance Networking Research Group (GHPN-RG), and the G^{2}MPLS architecture is aimed to become one of the main contributions to the activities of GHPN-RG.

4 G^{2}MPLS Requirements

In this chapter the requirements for the G^{2}MPLS NCP are provided, by analysing the possible general needs of the Grid applications and service plane (i.e. the Grid middleware) and the subsequent functionalities expected by the network transport, control and Management Planes. For each requirement a short description is provided along with an identification of its originator and importance for the implementation of the G^{2}MPLS NCP.

The requirement's originator or source can be distinguished in:

- **Users**, i.e. the individual using the Grid through an application for a demanding job,
- **Network operator**, i.e. the group of individuals owning a network infrastructure hosting the Grids and other connection services.

Different levels of importance are identified for each requirement with an obvious impact on the relevance of the provided overall system functionality. The identified levels of importance are:

- **Essential**, used for those requirements implying mandatory functionalities;
- **Preferred**, used for those requirements implying functionalities whose lacking implies a reduced functionality of the system
- **Nice to have**, used for those requirements implying optional functionalities whose lacking impacts the system but does not hinder a reduced operation;
- **Not essential**, used for those requirements implying completely optional functionalities whose lacking has a very minor impact of the system.

4.1 Grid application requirements with impact on underlying network services

A grid application uses services and functions defined by OGSA and specified by OGSI along with Grid infrastructure to accomplish specific work-related tasks that solve business and technical problems. In general, a Grid application is a collection of work items or jobs that carry out a complex computing task by using Grid network resources. It usually remains private and largely under the developers control. A Grid-enabled applications runs in a dynamic, sometimes loosely defined and heavily networked environment. The integration plan to enable an application to run on a Grid is very depended on the type of application. Applications can be classified based on a number of elements such as parallelism, granularity, communication and dependency. These subcategories of these elements are listed below:

- Parallelism has a significant impact on how the application is integrated to run on a Grid network environment. there are a number of possible examples that represent parallelism
  - Single Program, Single Data: Represents sequential programs that take a single input set and generate a single output set.
  - Single Program, Multiple Data: the input data can be partitioned and processed concurrently using the same program and comprises the majority of applications utilized the Grid computing.
  - Multiple Program, Multiple Data: Represents the broadest category of parallel/distributed applications where both the program and the data can be partitioned and processed concurrently
  - Multiple Program, Single Data applications require different transformation to be applied to the same set of input data
- Communication is associated with the frequency and size of communication required
  - The initial movement of data and programs to grid resources prior to computation
  - The communication while the application execution
- Granularity specifies how long a program can execute before it needs to communicate with other programs in the application
- Dependency between programs with an application

In greater detail, the elements required in designing or enabling a Grid application on such heavily networked environment is a collection of job, data and environmental considerations. Job relates to the collection of work units that make up the application such as data relates to input, storage, and output of data for the grid application. Environment represents the hardware, software and networking and dependencies of an application that run on a Grid environment. Job can be determined by a number of parameters such as:

- Job flow
- Types of jobs
- Number of different jobs
- Depth of sub-jobs
- Job topology

Data management such as data topology and amount of data has an impact on underlying network. Environmental considerations such as time constraints (e.g. sensitivity, difference between on-peak and off-peak network availability) have to be considered.

All above Grid application characteristics lead to a set of requirements that generically affect Grid applications.

- Network Requirements
  - Traffic characteristics (e.g. large data transfers, over long distance)
    - Volume requirements
    - Managed bandwidth requirements (e.g. peak and average bandwidth)
    - Real time requirements (e.g. delay and delay variation)
    - Quality of Service requirements (e.g. guaranteed/assured delivery)
    - Fast optical network re-configurability
    - Connectivity requirements (e.g. point-to-point, point-to-multipoint, multipoint-to-multipoint)
  - monitoring and forecasting requirements prior to and during application execution including relevant requirements for
    - network bandwidth
    - latency and jitter
    - multicast performance
- Application timing formats and timing requirements for performance tuning
- Security Requirements with respect to network, including information on
  - confidentiality of data during communication and computation
  - need for digital signatures, encryption and/or authorization
  - requirements for public vs. private information on application status and execution

Details about these application requirements are presented in the following table.

Requirement		Source	Importance	Description
R1.	Timing	Users	Essential	Application may have strict or loose requirement on application timing requirements (e.g. strict start time and loose end time, or vice versa).
R2.	Security	Users	Essential	Confidentiality of data during communication and computation.
R3.	Network	Users / Network Operator	Essential	Specify network requirements (Large data transfer, peak and average bandwidth requirements, connectivity requirements, e.g. multicast).
R4.	Dependency	Users	Nice to have	Level and type of interaction between programs and applications.
R5.	Communication	Users / Network Operator	Nice to have	Frequency, size of communication required between remote Grid sites.

	Requirement	Source	Importance	Description
R6.	Granularity	Users	Nice to have	how long a program can execute before it needs to communicate with other programs in the application.

Table 4-1: Grid application requirements.

4.2 Grid service plane requirements

The Grid Service that commonly occur within Grid systems are presented below. The categories that divide these functions are listed and described below.

- Scheduling, coordination and brokering requirements, including information on
  - needs for resource reservation prior to or during application execution
  - needs for co-allocation prior to or during application execution
  - Application components which must be scheduled (computation, data, remote instruments, intermediate files, communication, etc.)
  - any real-time requirements of the application
  - tolerance of application to delays during execution
- Remote Data Access requirements , including information on
  - Requirements for publication of data results (e.g. application preference for file-based access, collection-based access, etc.)
  - requirements for streaming vs. batch processing of data sets
  - needs for data staging prior to or during application execution
  - needs for management of distributed data (including preferences for single physical collection vs. distributed logical collection of data)
  - needs for access to heterogeneous storage systems; preferences for interface format to such systems
- Accounting requirements, including information on
  - needs for dealing with multiple accounts and/or accounting systems
  - system components which must be accounted for
- Information Service requirements, including information on
  - information relevant to application which must be accessed from IS
  - frequency of information access
  - metadata relevant to application information (including format of metadata -- single values, distributions, etc.)
  - application preferences on location, structure, representation, and format of IS information
- User Services requirements, including information on
  - needs for viewing system status prior to, during, and after execution
  - preferred format for system status information
  - application needs/preferences for system services and tools

Details about services required and interactions between them are presented in the following table.

Requirement		Source	Importance	Description
R7.	Co-allocation service	User	Essential	Co-allocation of Grid (e.g. data, computation, remote instruments) and network (e.g. data paths, bandwidth) resources on a single step prior or during application execution.
R7.1.	Execution Planning Service (Selection Services)	User	Essential	It builds “schedules”, where a schedule is a mapping (relation) between services and resources, possibly with time constraints. A schedule can be extended with a list of alternative “schedule deltas” that basically describe “if this part of the schedule fails, try this one instead”. EPS typically attempts to optimize some objective function such as execution time, cost, reliability, etc.
R7.2.	Candidate Set Generator	User	Essential	CSG determine the set of resources on which a unit of work can execute – “where is it possible to execute?”, rather than “where will it execute?”
R7.3.	Reservation service	User	Essential	It manages reservations of resources, interact with accounting services (there may be a charge for making a reservation), revoke reservations, etc. this may not be a separate service, rather an interface to get and manage reservation from containers and other resources. The reservation itself is likely to be an agreement document that is signed. A reservation service presents a common interface to all varieties of reservable resources on the grid.
R7.4.	Deployment & Configuration Service	User	Essential	Before a service or data container can be used by a unit of work, it must be configured or provisioned with additional resources.
R8.	Information Service	User	Nice to have	Information services are metadata about resources.
R9.	Monitoring	User	Preferred	This includes mechanisms to describe views on the resource state. Monitoring of applications for fault-tolerance reasons and QoS reasons.

Requirement		Source	Importance	Description
R10.	Fault-Detection and Recovery Services	User	Preferred	Fault-detection and recovery services may or may not be a part of monitoring, and may include support for managing simple schemes for stateless functions that allow trading off performance and resource usage. This service is required to handle more complex schemes such as managing check-pointing and recovery of single-threaded jobs, or even manage applications with distributed state, such as MPI jobs.
R11.	Security	User	Preferred	Security is wide-reaching (i.e. not local), encompassing issues relating the management and verification of credentials; privacy and integrity and policy.
R12.	Authentication, Authorization and Accounting	User	Preferred	Provide means to authenticate, check authorization and account for the service usage.
R12.1.	Policy	User	Preferred	Policies are needed to guide resources usage as well as incorporate business logic and service level agreements. Policies are implemented or executed within a particular context, such as security, workload manage, and control access to Grid resources.
R12.2.	Metering and Accounting	User	Nice to have	Accounting messages are used to collect, deliver, reconcile, store securely, and manage information about resource usage and/or charges.

Table 4-2: Grid service requirements.

4.3 Transport Plane requirements

G^{2}MPLS defines a new networking infrastructure which will be built on the top of a standard GMPLS architecture. This implies that the Transport Plane has to be GMPLS compatible. It means the ability to apply generalized labels for data links. The goal is not only to transport amount of data from one particular side to another, but also adaptively connect any two locations in the PHOSPHORUS test-bed. To this purpose it should be possible to create cross-connections on links provided by lower layer and use switching devices for path provisioning. These pieces of equipment will be controlled by the G^{2}MPLS Control Plane.

In GMPLS advanced mechanisms are used, e.g. for link aggregations and bidirectional connections. So, all these mechanisms should be supported by G^{2}MPLS Transport Plane too. For example there should be possible to identify all aggregated connections – in order to know what is inside when "unpacking" an LSP.

An important issue is how to provide monitoring information to G^2MPLS plane. This includes state of the transport network, alarms and network resources. For example, the Control Plane needs to know when the path goes down to react by establishing a backup path. On the other hand, there is a need to know when the primary path goes up again to delete the backup path and switch to primary again.

Each Grid application may have different requirements for the network. For example a computationally-intensive Grid application may be situated with a best-effort IP service for data movement as opposed to a visualization Grid application which may require high bandwidth, low latency connectivity for storage access. The resource selection and allocation algorithm for a Grid job cannot achieve the possible QoS objectives required by an application if its computations are completely unaware of the underlying network service(s). The required network service involves the network resources at the Grid locations and the interconnecting network domain(s). QoS-parameters of particular interest for the Grid applications typically include:

- Throughput;
  - from grid point of view, throughput, it is an amount of grid data that can be transmitted between two locations in time unit
- Delay; it can be defined as one-way delay or roundtrip delay
  - One way delay is important for a number of reasons:
    - Asymmetric paths
    - Different QoS in two directions
    - Application dependence on one direction more that both directions.
  - It has been defined in RFCs 2330, 2679, 2681 and as mentioned in GFD.23 is important because:
    - Some applications do not perform well (or at all) if end-to-end delay between hosts is large relative to some threshold value
    - Big variation in delay makes it difficult to support many real-time applications
    - Transport-layer protocols cannot sustain high bandwidths during big delay variations.
    - Values of this characteristic at the minimum level provide an indication of the propagation and transmission delay and at a higher level the path congestion.
  - Roundtrip Delay can be composed from the one-way delay measurements in both directions or by noting the time when the packet is sent and comparing this with the time when the response packet is received back from the destination.
- Jitter; it is described in RFC 3393 as it is used in different ways by different groups but particularly for the IP community is defined as:
  - The IP Packet Delay Variation (IPDV) for a selected pair of packets in a stream of packets and is very important in sizing output buffers for applications requiring regular delivery of packets (e.g. voice or video).
- Availability; it is applied to:
  - Network availability can be defined as “down” or “up” or even “congested”
  - Network node availability even if it frequently difficult to differentiate between a node being unavailable or the path to that node being unavailable.
- Packet loss

- Loss of a single packet often has a little impact on applications but repeated loss can have a significant effect. RFC 2680 describes loss characteristics such as one-way loss, roundtrip loss and statistical loss and is important since:
  - Some applications do not perform well if end-to-end loss between hosts is large relative to some threshold value.
  - Excessive packet loss may make it difficult to support certain real-time applications.
  - Transport-layer protocol difficulty to sustain high bandwidths under high packet loss.
  - Impact the QoS provided by Grid applications which may be sensitive to individual packets.

The Transport Plane requirements are gathered in the following table:

Requirement		Source	Importance	Description
R13.	Cross-connect capability	User	Essential	A user wants to create a data path using available links.
R13.1.	Nesting of LSPs	Network Operator	Essential	Different types of transport network technologies in testbed require to build the GMPLS hierarchy of LSPs.
R13.2.	Device management interface	Network Operator	Essential	Control Plane must handle the path request and changes the configuration on devices used for the LSP creation, modification and termination.
R14.	QoS in data path	User	Preferred	Different types of Grid application require different types of QoS
R14.1.	Throughput	User	Preferred	Grid applications may require transferring large amount of data.
R14.2.	Delay	User	Preferred	One-way delay or roundtrip delay.
R14.3.	Jitter	User	Nice to have	IP Packet Delay Variation (IPDV)
R14.4.	Availability	Network Operator	Preferred	Conditions determining operability
R14.5.	Packet loss	User	Preferred	Grid applications may be sensitive to packet loss.
R14.6.	Resilience	Network Operator	Preferred	Applied as link protection mechanism

Table 4-3: Transport Plane requirements.

4.4 Control Plane requirements

The ASTN/ASON requirements specified in [ASON-REQ] are assumed to be the primary reference for the implementation of the G^{2}MPLS Network Control Plane. Further implications deriving from the GNS support in a Control Plane are listed in the following Table 4-4 and must be intended as a complement to the requirements for signalling, routing, link management, addressing and Signalling Communication Network specified in [ASON-REQ].

Requirement		Source	Importance	Description
R15.	Grid service discovery	User	Essential	The G^2MPLS network Control Plane must provide mechanisms for the negotiation of the Grid and network services configurable across the interface between the Grid user/site and the network. This service discovery mechanism will include both network specific resources and operation modes (e.g. types of signals, protocols, routing diversity, permeability modes for the information coming from the network, etc.) and Grid specific capabilities (e.g. types of CPU, storage, OS, etc.)
R16.	Grid resource discovery	User	Preferred	The G^2MPLS network Control Plane must provide mechanisms for learning and advertisement of the Grid and network resource availability at the Grid user site. This neighbour discovery mechanism will include both network resources (e.g. amount of bandwidth, connectivity, etc.) and Grid resources (e.g. amount of CPU, storage, etc.).
R16.1.	Availability of different permeability modes across the UNI	Network Operator	Essential	Depending on the type and amount of routing information that the Grid user and the Network Operator are able to manage for GNS purposes, it is needed that different levels of permeability may be negotiable at the UNI, in order to cope with different scenarios at the Grid layer: • Preconfigured and static reachability information on the remote Grid sites; • Dynamic reachability information on the remote Grid sites with a dynamic learning; • Full Grid routing information and summarized network routing information for resolving the service/job endpoints and some loose connecting route (e.g. in case of inter-domain route), • full Grid and network routing information for resolving the service/job endpoints and the exact connecting routes.

Requirement		Source	Importance	Description
R17.	Connection management	User	Essential	The G^2MPLS network Control Plane will support mainly Switched Connections (SC) through the G.OUNI. Soft Permanent Connections (SPC) could be needed to enable interoperations with some entities in the network Management Plane (e.g. NRPS-es or GÉANT2 IDM). Permanent Connections (PC) are out of scope.
R17.1.	Unidirectional point-to-point connections	User	Essential	These are the majority of the network connections required by Grid applications in state of the art, because the fit better the asymmetric nature if the traffic flowing in the Grid, e.g. upload or download from a set of client towards a server.
R17.2.	Bidirectional point-to-point connections	User	Essential	These connections match the requirement for symmetric bandwidth allocation possibly required by some computational Grid applications in which a main stream direction cannot be identified. This is a mandatory choice for specific Transport Plane technologies, e.g. TDM.
R17.3.	Unidirectional point-to-multipoint connections	User	Nice to have	These connections are not used in state of the art Grid applications, but represent the upcoming enhancement e.g. for faster and effective data replication services.
R17.4.	Anycast point-to-point connections	User	Nice to have	In anycast, data is routed to the "nearest" or "best" destination as viewed by the routing topology. The G^2MPLS implementation of such a service could be provided by processing at the Control Plane layer those GNS requests with implicit resource description (i.e. specification of just the involved Vsites or the amount of Grid resources – e.g. 50 TB – without any info on their network attachment points).
R18.	Flexible bandwidth allocation for GNS services	User	Essential	The GNS(-es) related to the execution of a complex Grid job should be dynamic (i.e. medium lived connections instead of long lived permanent connections) and tailored to the bandwidth needs (i.e. with a guaranteed bandwidth as requested)

Requirement		Source	Importance	Description
R18.1.	Different bandwidth granularities for GNS services	User	Essential	A range of signal types (and thus bandwidth values) needs to be available for building a GNS, ranging from the Gigabit Ethernet up to the SDH-SONET TDM hierarchies and 10 Gigabit Ethernet, in optical or electrical technologies where possible.
R18.2.	Traffic classification / grooming / shaping	Network Operator	Not essential	Introduce some traffic parameters to specific signalling attributes (e.g. in the SENDER_TSPEC and FILTER_SPEC for RSVP), in addition to those related to the interface switching capability and encoding type for the required network connection in order to assist possible efficient traffic grooming / trunking within the transport network (i.e. from the ingress UNI-N downstream)
R19.	Advance reservations for GNS services	User	Essential	The Grid layer may ask at a given time for a future service setup of Grid and network resources, by specifying start time and duration of the required service. This implies: • an immediate processing and reservation of the selected Grid and network resources for that task in that timeframe • a later "service activation" tier just before the execution of the task.
R20.	Traffic Engineering for GNS services	Network Operator	Essential	The standard rationale behind the TE adoption in transport networks (i.e. resource optimization) here applies, because of the impact flexible bandwidth allocations might have on the resource consumption. The main consumer of the TE information is the constraints-based Path Computation Element (PCE), implementing some Constrained Based Routing (CBR) algorithms.
R20.1.	Enhanced TE based on optical network impairments	Network Operator	Nice to have	In the optical network it could be valuable to optimize the CBR outputs by taking into account the impairments deriving from the transmission of the information in the optical mediums. For example, low OSNR values or high polarization mode dispersions could determine performance degradation on the built end-to-end connection.

Requirement		Source	Importance	Description
R21.	Service resiliency	User	Essential	The guarantee of the required service during the execution of a task might be compromised by some possible faults of the involved network or Grid resources. The Control Plane should provide means for faulty condition detection and reaction, as well as mechanism for diverse routing between the failing path and its backups.
R21.1.	Resiliency for inter domain network services	Network Operator	Essential	As a consequence of the above root requirement the Network Operator might need to provide the means for mitigating faulty network condition even when inter-domain connections are involved.
R21.2.	Escalation of network recovery strategies	User / Network Operator	Nice to have	Fixed a hierarchy of recovery procedures (e.g. protection, fast restoration, end-to-end restoration), it could be possible to escalate this hierarchy for recovery purposes (e.g. from protection to restoration). The user should accept the possible degraded network resiliency and, thus, this behaviour should be part of a proper agreement at the service discovery phase.
R21.3.	Coordinated recovery strategies among the different network layers	Network Operator	Essential	In case of network connections implemented through hierarchical LSPs, the cross-layer recovery support is needed.
R21.4.	Coordinated recovery strategies between the G^2MPLS layer and the NRPS layer	Network Operator	Essential	In case a network service is obtained by the interaction between G^2MPLS domains and NRPS-controlled domains, coordination between the respective recovery strategies need to be implemented. This could be a special case of inter-domain recovery coordination.

Requirement		Source	Importance	Description
R21.5.	Coordinated recovery strategies between the Grid layer and the network layer	User	Nice to have	Depending on the seriousness and impact of the occurring network fault, it could be impossible for the Network Control Plane to recover the service. An escalation from the network layer to the Grid layer recovery could be needed in this case, triggered by timely fault notifications e.g. across the G.OUNI. The consumer of notifications should be those entities of the Grid middleware responsible for job monitoring and workflow check-pointing, which could react e.g. by re-scheduling the failing job in a different time or on different Grid resources.
R22.	Crankback signalling for network connections	Network Operator	Nice to have	Network Operator might need some means for lowering the blocking probability possibly occurring in source routed signalling. Depending on the crankback-ing node (i.e. core node or ingress node) and on the selected addressing mode (with Grid network attachment points specified or not), LSP rerouting could be tried: • to the same egress/destination network node • to an alternative egress/destination node supporting the same type and amount of Grid resources In the latter case, Grid knowledge must be fully available in the core network, i.e. it could not be treated opaquely by transit LSRs.
R23.	Multi-homing connectivity	User	Nice to have	Multi-homing refers to multiple links between a user end-point and one or more transport networks. Multi-homing may be used, for example, for load balancing or protection via diverse routes, based on the direct action of the user.
R24.	AAA	Network Operator	Essential	Provide means to authenticate, check authorization and account for the service usage. In the Control Plane scope, AAA mechanisms basically focus on • the interfacing means towards external AAA infrastructures • the internal means to forward session credentials along the signalling path.

Requirement		Source	Importance	Description
R24.1.	Authentication and Authorization of the service requests	Network Operator	Essential	Provide means to check and forward service credentials for using grid and network resources.
R24.2.	Accounting of the service requests	Network Operator	Nice to have	Provide means to account for the real resource usage.
R25.	Message security and integrity	Network Operator	Nice to have	Provide some means for guaranteeing the integrity of the communications in the Control Plane in order to avoid corrupted or spoofed signalling by unauthorized parties that can led to denial of service caused by locking up network resources.
R26.	G^2MPLS control of Transport Plane equipments	Network Operator	Essential	G^2MPLS is conceived to implement the network Control Plane for equipments in the optical, TDM and Ethernet switching domain. Therefore, test-bed equipments are assumed to provide just the Transport Plane and proper interface(s) for its configuration. The possible COTS Control Plane instances of these equipments are assumed to be switched off for G^2MPLS testing.

Table 4-4: Control Plane requirements.

4.5 Management Plane requirements

4.5.1 FCAPS requirements

The Management Plane requirements generally are identified as FCAPS requirements. FCAPS is an acronym for Fault, Configuration, Accounting, Performance, Security, which are the management categories into which the ISO model defines network management tasks. FCAPS requirements are gathered in the following table:

Requirement		Source	Importance	Description
R27.	Fault notification	User / Network Operator	Preferred	User and Network Operator must be aware of problems in faulty conditions in the provisioned services. Thos translates into detecting and localizing a disconnection of data and/or control channels and notification to an upper layer or logging.
R27.1.	Fault mitigation	Network Operator	Nice to have	In case of fault, isolation and correction by using an alternative path and/or restarting some part of job.

Requirement		Source	Importance	Description
R28.	Configuration issuing / retrieving	Network Operator	Nice to have	An automatic Control Plane configuration of the network devices. and retrieving of the operational configurations for management purposes (e.g. config backup).
R29.	Accounting	Network Operator	Nice to have	A measurement of the resource utilization by user services. Possible usage parameters could be: job counts, transported amount of data, network service resiliency, etc.
R30.	Performance monitoring	Network Operator	Not essential	Real time monitoring of grid and network resources utilization
R31.	Security	User / Network Operator	Nice to have	An authentication and authorization process for controlling access to assets in the network. A solution could be in-band with signalling protocols (e.g. Policy_Data and Integrity RSVP-TE objects), but other solutions might apply. User to domain and domain to domain interactions are the main framework.

Table 4-5: FCAPS requirements.

4.5.2 NMS-based service provisioning requirements

Network management refers to the maintenance and administration of large-scale computer networks and telecommunications infrastructure at the top level.

A large number of protocols exist to support network and network device management. Common protocols include SNMP, CMIP, WBEM, Common Information Model, Transaction Language 1, Java Management Extensions - JMX, and netconf.

There is a huge list of commercial and open Network Management Systems: Altris management Suite, CA Unicenter, HP OpenView, IBM Tivoli, Nagios, Novell ZENworks, Spiceworks, Zenoss, etc.

NMS issue for network resources: controlling, planning, allocating, deploying, coordinating, monitoring. NMS refers to FCAPS.

5 G^2MPLS Reference Network and Service Models

This chapter provides an introduction to the G^2MPLS network and service reference models. This discussion is mainly focused on G^2MPLS, but needs to embrace also the other relevant components of the overall PHOSPHORUS architecture, in order to define the relationships and interactions among the different components.

G^2MPLS is a Network Control Plane architecture that implements the concept of Grid Network Services (GNS). In the PHOSPHORUS framework, GNS is a service that allows the provisioning of network and Grid resources at the same time and with the same priority (single-step) through a set of seamlessly integrated procedures. Joint co-allocation of Grid and network resources allows to configure network connections in the same tier of Grid resources, by guaranteeing enhanced service availability (advantage for the user) and tailoring to the user requirements (advantage for the user and the network operator).

From a user's perspective, G^2MPLS enables a real node-to-node deployment of on-demand Grid services, because it exposes specific interfaces towards the Grid layer. Part of the middleware functionalities related to selection, co-allocation and maintenance of both Grid and network resources are provided through these interfaces.

From a network operator perspective, G^2MPLS is a means for the integration of Grids and automated network control plane technologies in real operational networks. This allows on one side to use well established solutions for setup, recovery and crankback of network connections spanning multiple domains and matching traffic engineering objectives; on the other side, G^2MPLS allows to overcome the current limitation of Grids that operate as stand-alone overlaid infrastructures upon research networks with different administrative ownership and uncorrelated procedures.

5.1 Network concept

In the PHOSPHORUS framework, different layers are identified as shown in Figure 5-1:

- Grid layer,
- Network Service Plane,
- G^2MPLS Network Control Plane,
- Transport Plane.

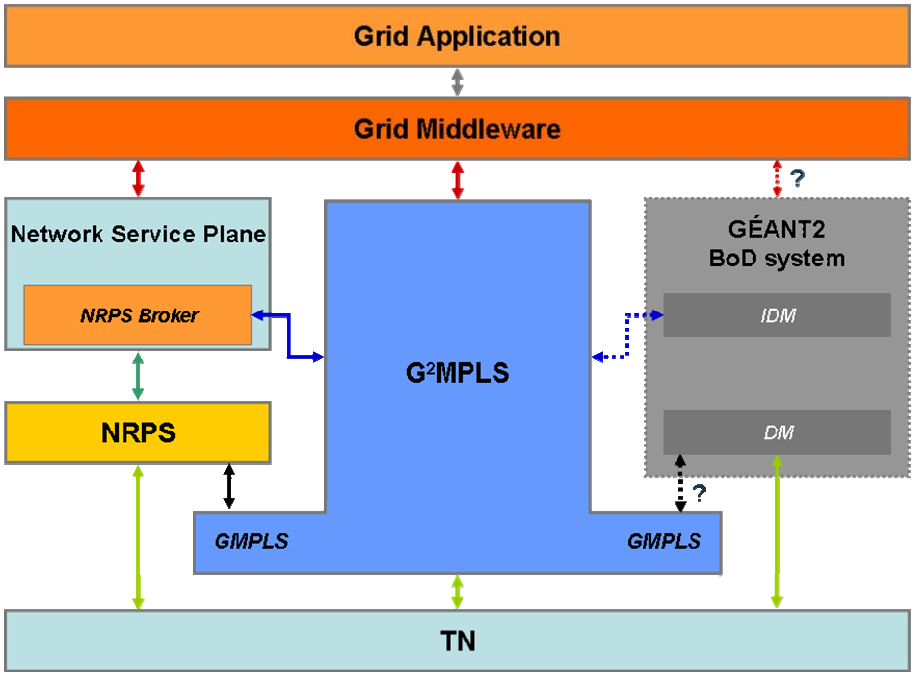


Figure 5-1: G^2MPLS positioning in the PHOSPHORUS framework including GÉANT2 BoD system.

The Transport Plane is the basic layer comprising all the data bearing equipments and their configuration interfaces.

G^2MPLS Network Control Plane is aimed to provide:

- *Discovery and advertisement* of Grid capabilities and resources of the participating Grid sites (*Vsites*);
- *Grid and Network Service setup* including:
    - *Coordination* with the Grid local job scheduler in the middleware responsible for the local configuration and management of the Grid job;
    - *Configuration* of the network connections among the Vsites participating to the Grid job;
    - *Management of resiliency* for the installed network services and possible escalation to the Grid middleware components that could be responsible for check-pointing and recovering the whole job;
    - *Advanced reservation*s of Grid and network resources;
- *Service monitoring* both for the Grid job and the related network connections.

The Grid layer is intended to comprise both Grid application and Grid middleware. Discussion on the architecture of the Grid layer is out of the scope in this document. The relevant aspect of this layer in a G^2MPLS perspective is the functionalities exported to/by the underlying network Control and Management Planes. G^2MPLS is primarily intended to interconnect remote instances of the Grid middleware, which are responsible for managing Grid resources localized in different sites (ref. Figure 5-1). In a more visionary scenario and in support of possible future applications, G^2MPLS could also interconnect directly Grid users/applications and Grid resources. Moreover, the southbound interface of the Grid layer represents a technological boundary, not necessarily an administrative boundary. This reference point cannot be addressed with standard IETF peer/integrated models, because any Grid component in the application or even in the middleware cannot peer with any Control Plane instance running on a network node.

Nevertheless, G^2MPLS is required to be deployed in structured, heterogeneous and multi-domain networks, possibly using other non-GMPLS technologies. Network Resource Provisioning Systems (NRPS) and GÉANT2 BoD system are relevant examples for these neighbouring relationships. The inter-domain natural bent of G^2MPLS raises the issue of the interaction and cooperation with these technologies, both in a peer-style model (blue arrows in Figure 5-1) and in an overlay-style (black arrows).

Network Service Plane (NSP) is the adaptation layer between the Grid and the NRPS-es managing different Transport Network domains. NSP deals with inter domain routing issues at the NRPS layer and coordinates the different NRPS domains (e.g. UCLP, DRAC, ARGON). NSP is designed according to a client-server model and its Network Broker acts as the client of NRPS-es acting as servers. In both cases of interaction, i.e. peer-style and overlay-style, G^2MPLS NCP can progress and receive only network service requests, due to the NSP knowledge of just network resources. In the case of peer-style interaction, G^2MPLS can deploy its Grid control capabilities only in its own domain. On the contrary, this functionality is neither meaningful nor possible in the case of overlay-style interaction.

GÉANT2 BoD system is depicted in dotted lines in Figure 5-1 because it is not in the core bulk of the PHOSPHORUS integration and development activities. However, two possible interoperation scenarios can be adopted by G^2MPLS towards GN2-BoD and they are similar to the approach adopted for NRPS-es and the Network Service Plane.

Further discussions on the G^2MPLS interworking with the NSP and GÉANT2 BoD system are provided in sections 12 and 13.

Two Control Plane models are defined for the G^2MPLS architecture:

- G^2MPLS Overlay
- G^2MPLS Integrated

These models concern the layering of grid and network resources. Thus, they have a different meaning and scope with respect to the IETF definitions for GMPLS Overlay, Augmented and Peer/Integrated models. A description of G^2MPLS Control Plane models is provided in the following subsections.

5.1.1 G^{2}MPLS Overlay

In the G^{2}MPLS Overlay model, the Grid layer is supposed to have Grid and network routing knowledge in order to provide Grid resource configuration and monitoring (as in its standard behaviour) plus network resource configuration and monitoring. G^{2}MPLS acts as an information bearer of network and Grid resources and as a configuration “arm” just for the network service part (ref. Figure 5-2).

Most of the computational and service intelligence is maintained on the Grid layer and, upon the occurrence of a Grid job request from a user, it is up to the Grid scheduler to initiate and coordinate the reservation process through a sequence of phases, including:

- localization of resources in an Index Service available at the Grid middleware;
- negotiation with the selected resources (local and remote Grid plus network in the middle) of a common timeframe for allocating the respective job parts;
- initial advance reservation of the agreed service segments, which implies a temporary booking of the service segment;
- final commit of the advance reservation, which results in the final booking of resources on the different service segments after all the initial booking have been positively acknowledged and correlated.

Each involved Grid site (Vsite) is responsible for scheduling, configuring and monitoring of the job part to be run on its resources. This functionality is provided by the Local Resource Management System (LRMN).

The G^{2}MPLS is responsible for scheduling, configuring and monitoring of the job part related to the network, thus implementing advance reservations, recovery and connection monitoring mechanisms.

These steps are briefly shown in Figure 5-3, in which the Grid user issues a job request towards the Grid scheduler that localizes site A, site B and the G^{2}MPLS network between them as service segments to configure. As detailed in Figure 5-3, the Grid scheduler initiates a negotiation phase (Negotiation request/response) with all the involved job providers (Sites A and B and the G^{2}MPLS domain) aimed to find a common timeframe for the execution of the job complying with the Grid user requirements (time, bandwidth, etc.). If such an intersection exists, the Grid scheduler sends initial advance reservation requests towards the involved sites, in order to book temporarily those resources (e.g. for 60 secs) and procure their availability for the final booking commit/confirmation. This operation is generally for free, i.e. the job provider (both Grid and network) cannot account for any service instantiation until a final commit is received. The final booking of resources is issued by the Grid scheduler once all the advance reservations have been successful. The final advance reservation commit request/response phase is aimed to book definitively (Grid and network) resources for that job in the selected timeframe and it is an accounted operation.

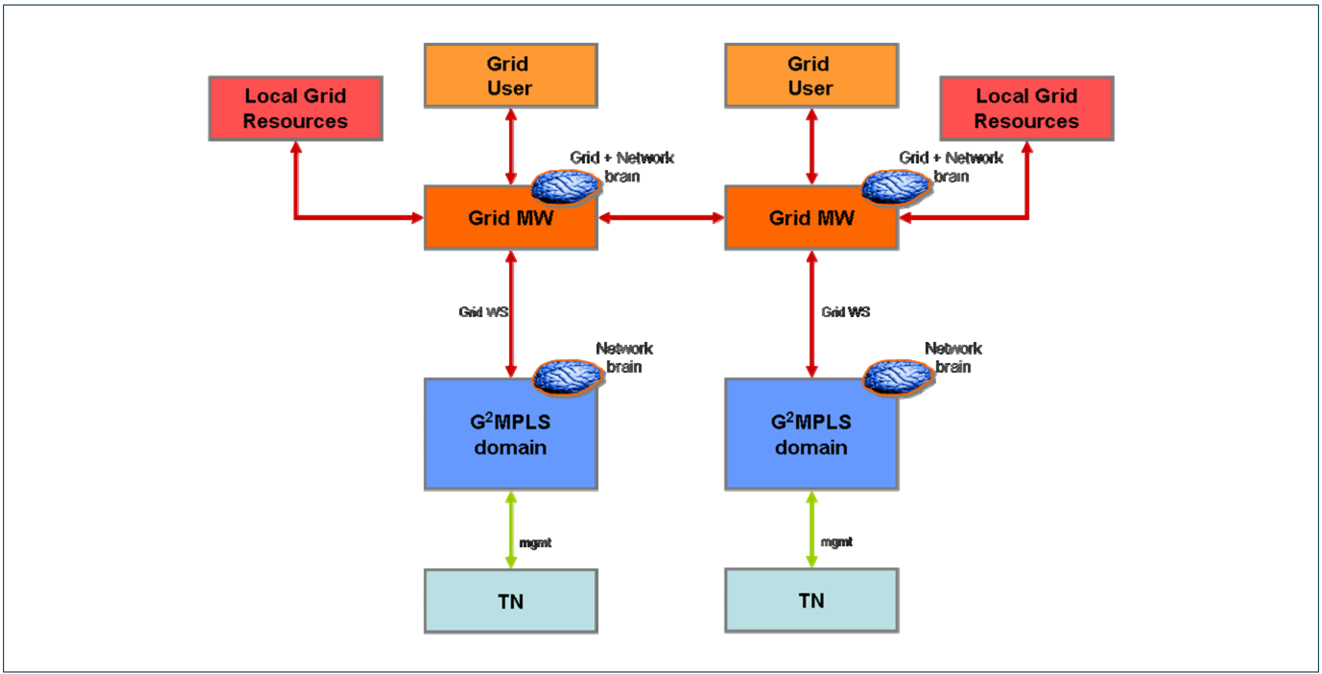


Figure 5-2: G^{2}MPLS overlay model.

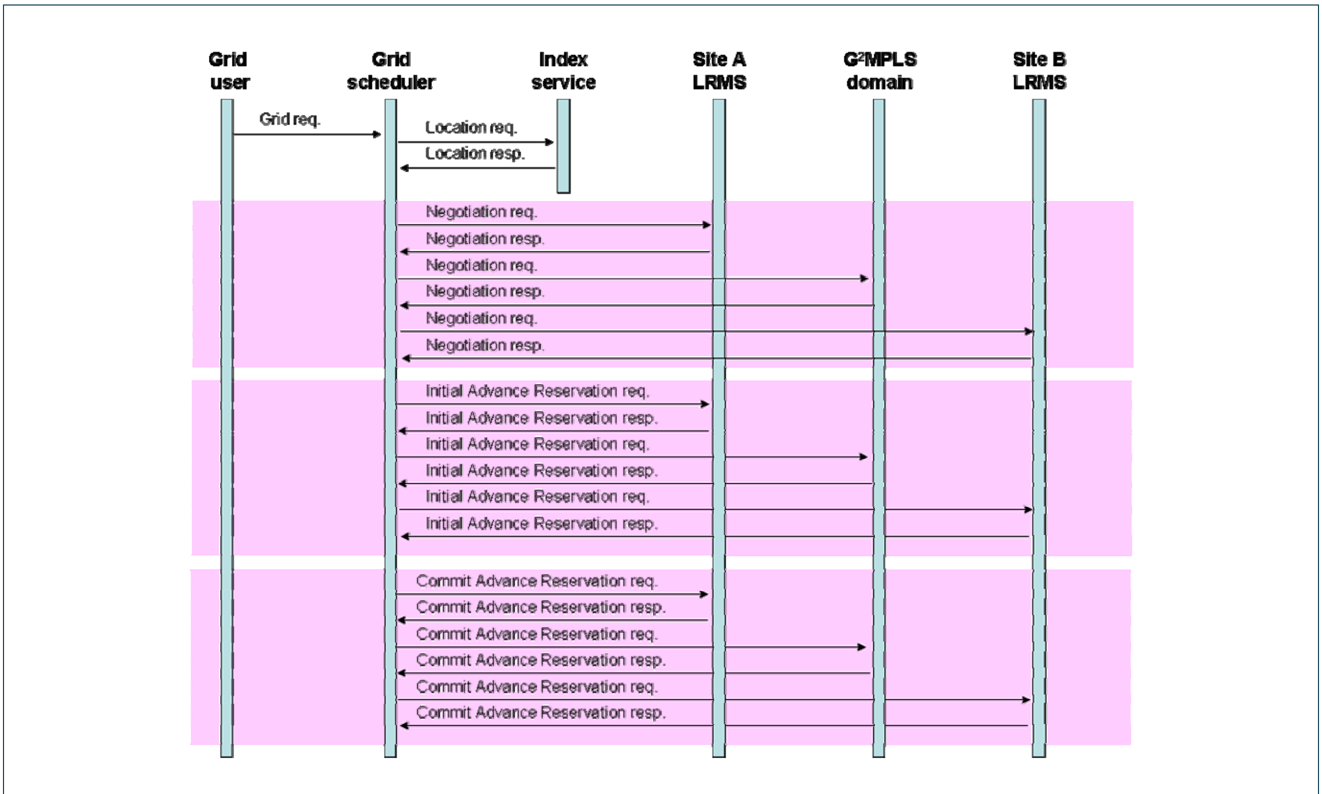


Figure 5-3: Grid and Network service setup in G^{2}MPLS overlay model.

5.1.2 G²MPLS Integrated

In the G²MPLS Integrated model, most of the functionalities for resource advance reservation and commit are moved to the G²MPLS Network Control Plane and G²MPLS is responsible for selecting the job segment providers and coordinating the co-allocation process (ref. Figure 5-4).

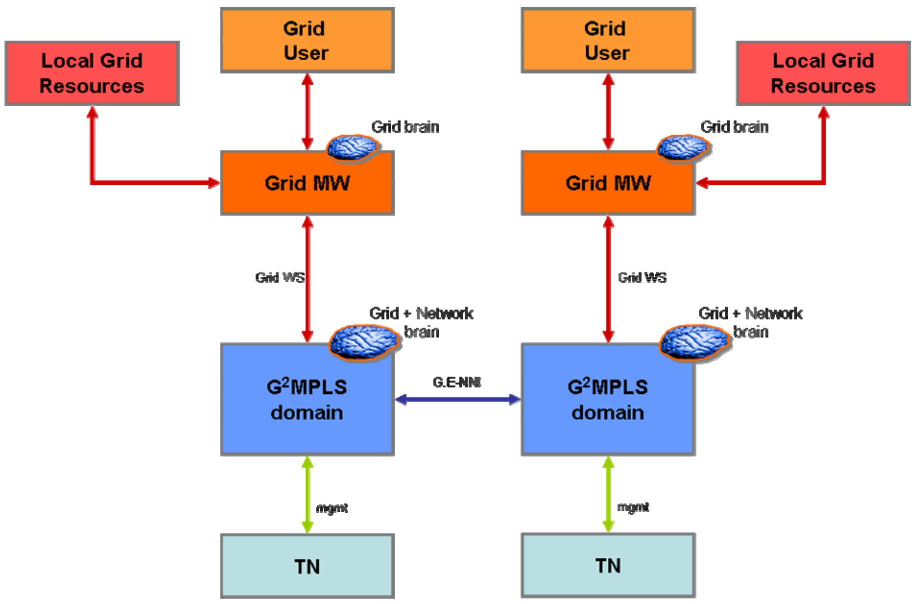


Figure 5-4: G²MPLS integrated model.

Each involved Grid site (Vsite) is still responsible for scheduling, configuring and monitoring of the job part to be run on its resources. This functionality continues to be provided by the Local Resource Management System (LRMN).

On the contrary, G²MPLS is responsible for scheduling and configuring all the job parts, those related to the Grid sites and those related to the network. Therefore, Grid and network resources are seamlessly allocated in one step, under the coordination of the G²MPLS NCP.

For G²MPLS purposes, Grid sites are modelled as network nodes with specific additional Grid resource information. Therefore, these nodes are available in the network topology and can be used for the localization of resources (network and Grid remote) upon the occurrence of a service request. Different routing behaviour could be possible for these Grid end-nodes, i.e. they could not participate in the routing protocol instance or be silent listeners or be full peering entities. This behaviour depends on the network model used while running the G²MPLS.

The Grid and network localization service is included in G^{2}MPLS integrated model, in order to have knowledge of the capability and availability of all the resources that can be involved in Grid Network Services instantiations. The Grid middleware/application could just declare the identity of the Grid site involved in the job, while it is up to the G^{2}MPLS to resolve the network attachment end-points for these sites. However, it could be possible for some applications to declare just an anonymous service requirement (e.g. an amount of storage or CPU) despite its location. In this case G^{2}MPLS is responsible also for localizing the best resource candidate for that service and the best network attachment point.

For all these purposes, routing computations and advertisements are basic enablers of the G^{2}MPLS integrated model, as well as all the signalling procedures for the different network reference points (G.OUNI, G:I-NNI and G.E-NNI). The two schemas that can be used as reference for the Grid-extension of the G^{2}MPLS NCP can be identified in:

- JSDL, for the job submission and description
- GLUE for the resources description.

Te GNS service setup phase in a G^{2}MPLS integrated model is briefly sketched in Figure 5-5.

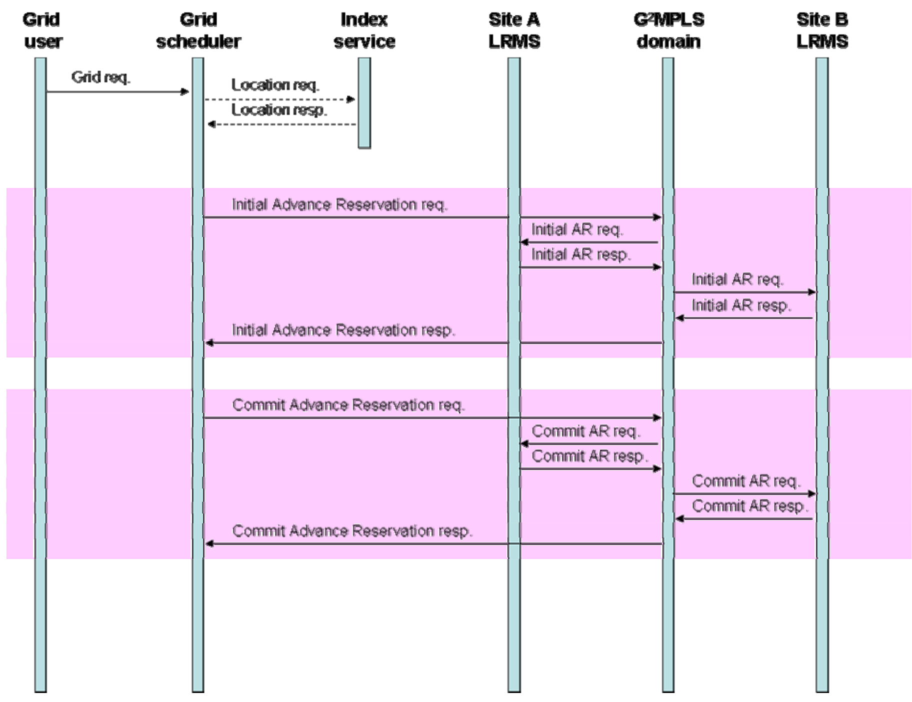


Figure 5-5: Grid and Network service setup in G^{2}MPLS integrated model.

The Grid scheduler functionality is still needed to support the many user applications that rely on specific Grid infrastructure. Moreover, the Grid scheduler is needed to support workflow services. In fact, the elementary unit of service managed by the G^2MPLS NCP is the Grid job, which is a component of the workflow.

The index service at the Grid middleware could be needed for Grid performance monitoring and administration, as could be used by the Grid scheduler to retrieve the identities of the Grid sites to be involved in a job. For this reason its related transactions are drawn by dotted lines.

The negotiation phase previously mastered by the Grid scheduler is now merged with the G^2MPLS-centric initial advance reservation phase. This advance reservation could benefit from the GMPLS crankback procedures. As detailed in Figure 5-5, the G^2MPLS domain receives an "Initial Advance Reservation" request from the Grid scheduler and may be is responsible for finding the job providers (site A and B) in case of implicit service request (no site detail but just service, e.g. an amount of CPU, or storage, etc.). Once G^2MPLS finds common timeframe for the execution of the job complying with the Grid user requirements (time, bandwidth, etc.), it sends a response to the Grid scheduler. The positive response implicitly means availability for the final booking commit/confirmation. The final booking of resources is issued by the Grid scheduler once the advance reservation positive response is issued and it is still G^2MPLS-centric.

An important feature provided by the use of G^2MPLS integrated model is the availability of recovery functionalities, which are the native realm of GMPLS. Current Grid middlewares provide just check-pointing for jobs execution in the framework of a workflow. The G^2MPLS integrated model allows the definition of new escalation strategies, which could start recovering the network connection, then (in case of failure) move to the selection of new job performers – if not strictly specified by the application/middleware – and/or move to the notification towards the Grid layer/user that could take the final decision on pausing re-scheduling the job on other resources. Recovery definitely benefits of the integration of the resource selection and reservation mechanisms into just one decisional entity, i.e. the G2MPLS.

5.1.3 Summary of interactions among the layers

The main interactions among the identified PHOSPHORUS layer are summarized in this section for the G^2MPLS overlay and integrated models.

The functional blocks identified in Figure 5-6 are:

- **Grid User**, the user that request for a Grid job execution on a Grid infrastructure
- **Local Grid resources**, a set of resources that are pooled (e.g. hosts, software licenses, IP addresses) or that provide a given capacity (e.g. disks, network interfaces, memory, databases) localized in a Virtual Site.
- **Grid Middleware**, the bundle of software components that support Grid application through jobs/workflow management.

- **G²MPLS NCP**, the network Control Plane that performs the GNS transaction control functions (setup, releases, restore, monitor, advance reservation) and the related support functions (i.e. grid and routing information dissemination, recovery).
- **Transport network**, the set of transport resources that are grouped according to operator policies aimed to provide network connectivity.

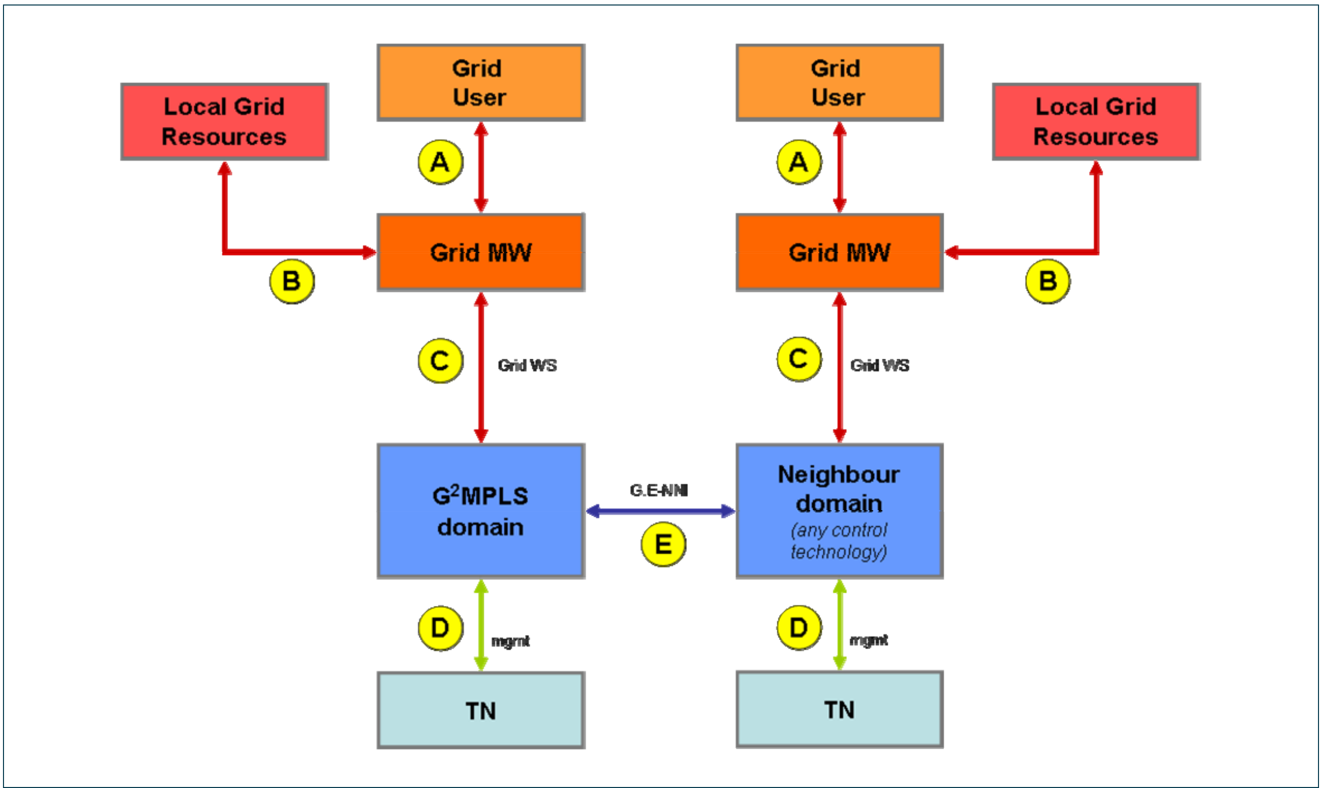


Figure 5-6: Interactions among the PHOSPHORUS layers.

The main interactions among these macro functional block are described in the following table, with X → Y indicating an action in the downstream direction, i.e. from X to Y, and X ← Y indicating an action in the upstream direction, i.e. from Y to X.

Interaction	Actors	Description
A	**Grid User, Grid MW**	→ User issues Grid service requests (Grid resources and/or network connections) ← user retrieves Grid capabilities info (in case of explicit resource selection) ← user retrieves Grid resources info (in case of explicit resource selection)

B	**Grid MW, Local Grid Resources**	← MW retrieves local Grid resource capabilities ← MW retrieves local Grid resource availability → MW issues local Grid resource configuration and performs job supervision
C	**Grid MW, G^2MPLS NCP**	← MW retrieves remote Grid resource capabilities ← MW retrieves remote Grid resource availability **Case A: G^2MPLS Overlay** ← MW retrieves network resource availability (at least in terms of attachment points for the Vsites) → MW issues network resource configuration and performs connection monitoring (advance reservation for inner network resources need to be provided by G^2MPLS) **Case B: G^2MPLS Integrated** → MW issues Grid transaction setup requests and G^2MPLS performs all the transaction related management.
D	**G^2MPLS NCP, Transport Network**	← G^2MPLS retrieves transport network capabilities ← G^2MPLS retrieves transport network resource availability ← G^2MPLS catches transport network resource alarms → G^2MPLS issues transport network configuration
E	**G^2MPLS NCP, peering domain**	← → retrieval of peering domains capabilities (Grid and network or just network info depending on the domain, e.g. G^2MPLS-based or NRPS-based) ← → retrieval of peering domains resource availability (Grid and network or just network info depending on the peering domain, e.g. G^2MPLS-based or NRPS-based) ← → Grid and network or just service configuration

Table 5-1: Description of the interactions among the PHOSPHORUS macro functional entities.

The interactions involving directly G^2MPLS NCP are C, D and E. Interaction D is mandatory for each NCP implementation and it is equipment dependent. Interactions C and E are G^2MPLS specific, i.e. they extend similar interactions just defined for ASON/GMPLS to the Grid semantics and framework.

5.1.4 Routing domains and hierarchies

G^2MPLS architecture complies with the hierarchical routing defined in ITU-T G.8080, ITU-T G.7715, and OIF E-NNI routing specification (OIF-E-NNI-Rtr-1). Therefore, the routing interaction between neighbouring domains will be possible through the communications of Routing Performers (or Routing Controllers in the OIF components namespace) participating to the OSPF area (or Routing Area in the ITU-T ASON namespace) defined for the specific hierarchical level.

According to ITU-T G.805, there are two kinds of domain identified for ASON purposes: an administrative and management domain. The administrative domain represents the extent of resources which belong to a single network operator, a service provider or an end-user. Administrative domains do not overlap amongst themselves. On the other hand, management domains represent set of managed objects, grouped according to the policy, technology or geographical specific requirements. Management domains may overlap amongst themselves, however a management domain should not cross the border of administrative domain.

The ASON hierarchical routing model enables the sharing of automated signalling procedures across the different Control Plane sections envisaged for the PHOSPHORUS G^{2}MPLS architecture.

The different network technologies (optical switching, SDH switching, Ethernet switching) are intended to be partitioned according to administrative partitions, identified by means of some network reference points (G.OUNI, G.I-NNI, G.E-NNI). Therefore, a network provider operating with more than one technology is expected to represent a single administrative entity owning a number of administrative network domains, one per each technology. The interaction between the users and these domains is expected to happen according to the ASON Overlay Model via G.OUNI interface. The interactions between neighbouring domains are expected to happen according to a partial peering model across the G.E-ONNI interface.

In the context of inter-carrier functionality in support of Grids, a further entity needs to be introduced for G^{2}MPLS. This entity might be put on the top of many network administrative domains and may cross the network administrative boundaries, by composing a virtual entity spread among multiple administrative domains. Following a concept defined for Grids, this entity may be identified in a Virtual Organization (VO) comprising a set of Grid sites (Vsites). The following picture presents the concept of Virtual Organisations (VO) on the top of a G^{2}MPLS network, in which a structured administrative partitioning has been applied.

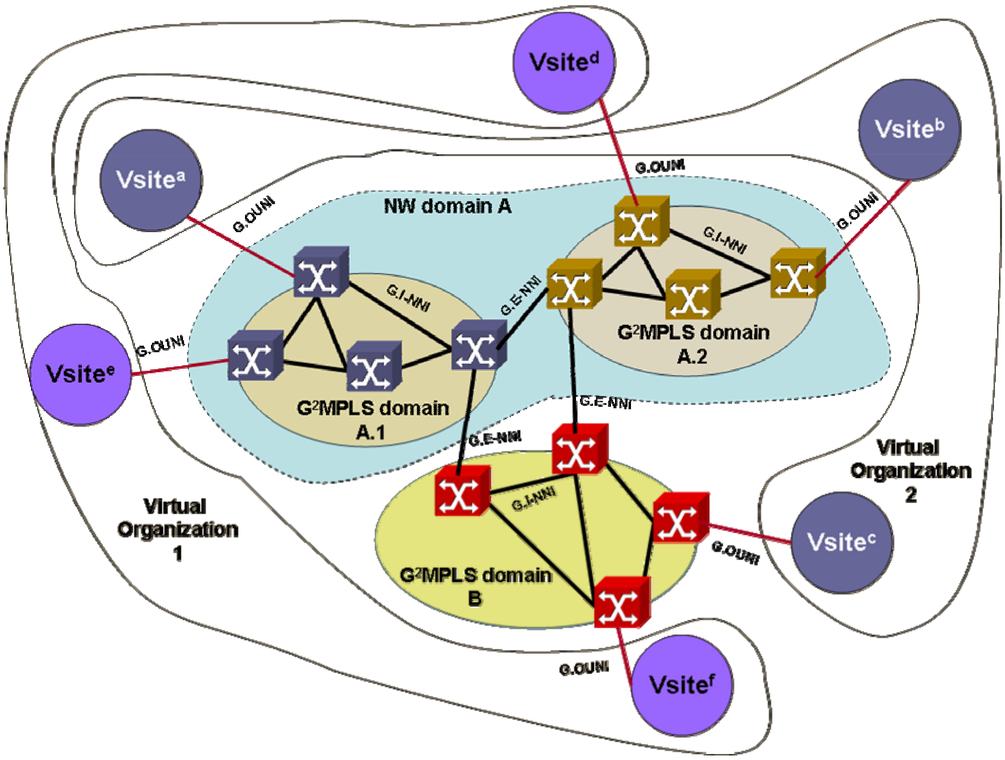


Figure 5-7: G^{2}MPLS routing domains and reference points.

5.1.5 Separation of Signalling Communication Network (SCN) and Transport Network (TN)

As per IEFT RFC 3945, the GMPLS architecture clearly separates the Control Plane and the forwarding plane and it also further separates the Control Plane in two parts, the signalling plane containing the signalling protocols and the routing plane containing the routing protocols.

As per ITU G.7712, ASON requires a communications network, the Signalling Control Network (SCN), to transport signalling messages between its components. The connectivity, i.e. signalling channels, among the SCN components is provided by the Data Communication Network (DCN), which a separate network with respect to the Transport Network with its own communication stack and means.

G^2MPLS architecture complies with this separation statement and the support of Grids does not impose requirements that imply a denial for this compliance.

5.1.6 G^2MPLS gateways

G^2MPLS/GMPLS NCP is structured in a set of different L4/L3 protocols for signalling and routing. Since the Grid framework is mainly WS-based and similar choices have been adopted for the Network Service Plane and the GÉANT2 BoD system, some transition means from one context to the other need to be provided at the external network reference points of the G^2MPLS NCP, i.e. the Grid-capable Optical User-Network Interface (G.OUNI) and the Grid-capable External Network-Network Interface (G.E-NNI). For this purpose two additional architectural elements are part of the G^2MPLS network model (ref. Figure 5-8):

- The G.OUNI gateway
- The G.E-NNI gateway

The gateways are aimed to provide the needed bridging functionality between the two frameworks and preserve the core G^2MPLS/GMPLS signalling and routing procedures by concentrating in single points the adaptation functions. An example of their deployment in an interoperation scenario with NRPS-es is provided in Figure 5-9.

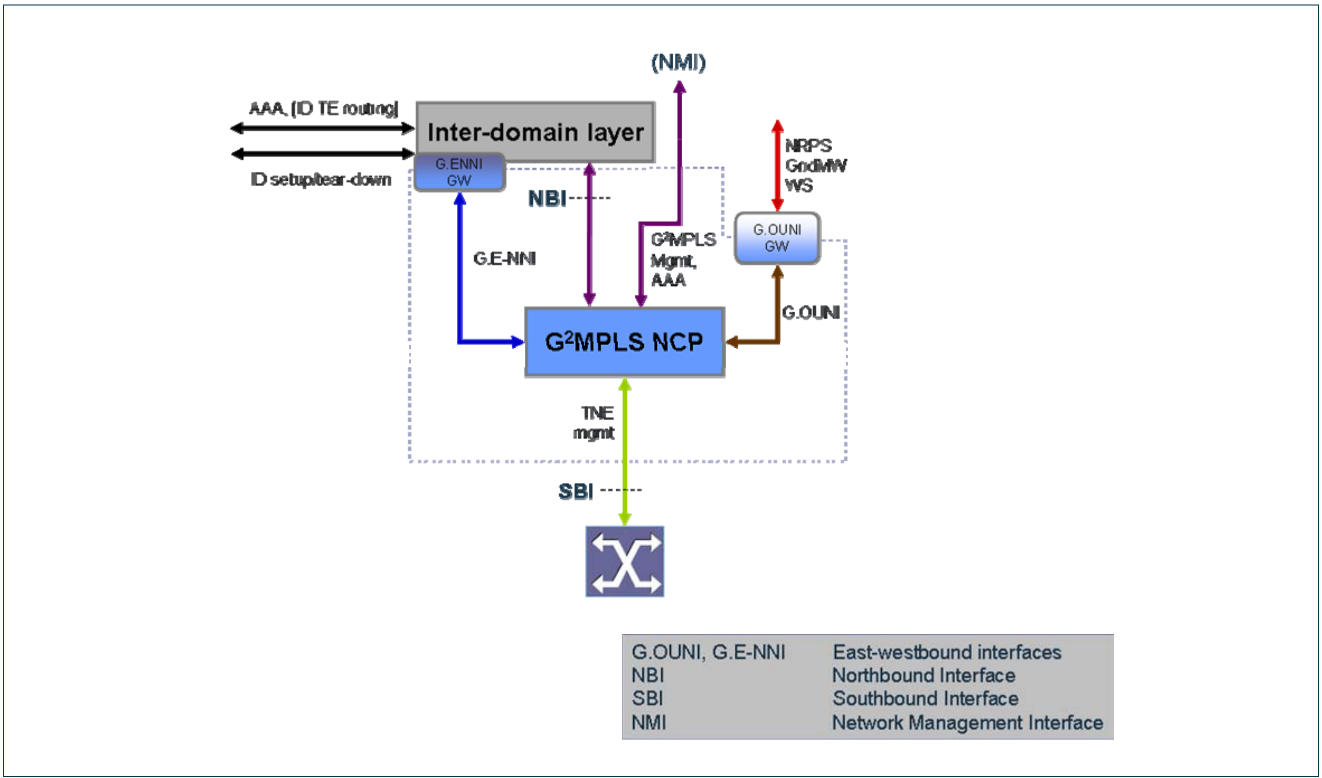


Figure 5-8: G^{2}MPLS NCP with gateway functional elements.

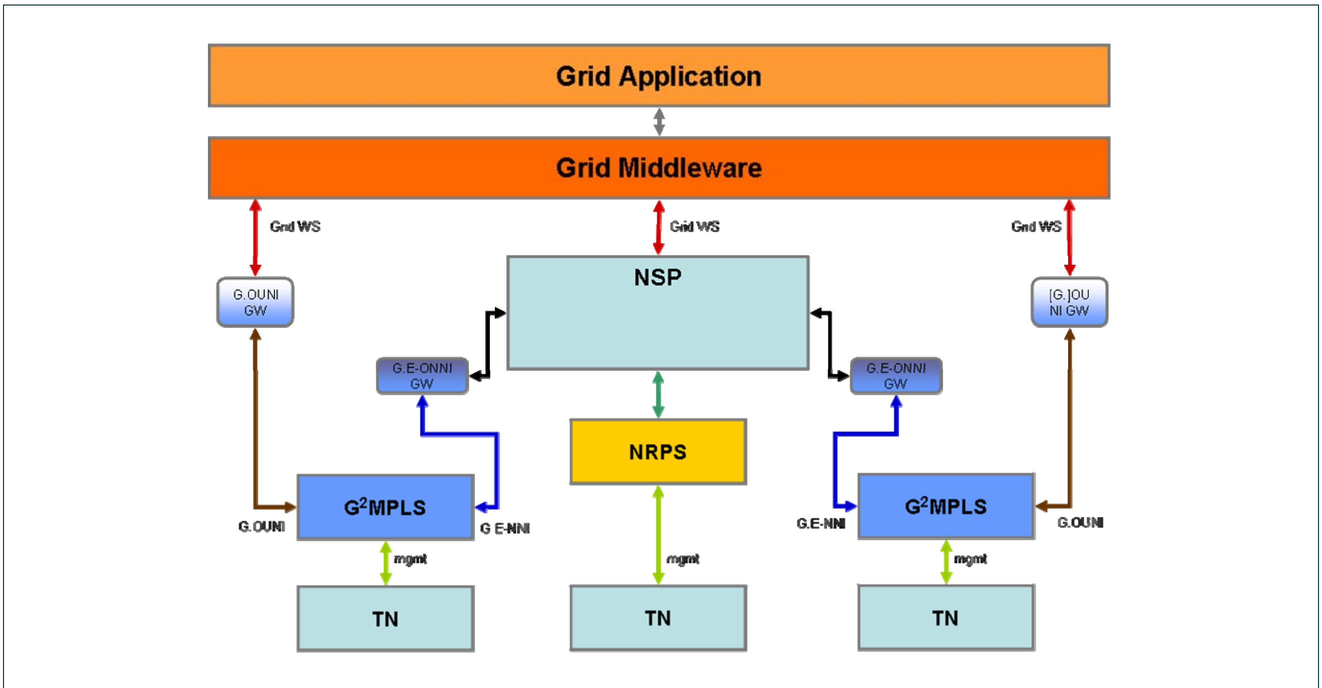


Figure 5-9: G^{2}MPLS interoperation with NRPS-es through gateways.

5.2 Service concept

G^2MPLS architecture exposes interfaces specific for Grids and is made of a set of extensions to the standard ASON/GMPLS architecture. Therefore, G^2MPLS results in a more powerful NCP solution than the standard ASON/GMPLS, because it will comply with the needs for enhanced services required by the Grids.

G^2MPLS is not conceived to be an application-specific architecture, i.e. it will support any kind of endpoint applications by providing network transport services and procedures that can fall back to the standard GMPLS ones. Therefore, two types of network users are identified and supported in the G^2MPLS architecture:

- Grid users (i.e. “power users”), requiring network and Grid services
- Standard users, requiring the automatic setup and resiliency of just network connections across the transport network are supported by G^2MPLS as well.

Services supported by G^2MPLS towards the Grid layer comprise the localization and selection of resources for the execution of a job. They include:

- Resources indexing (processing, storage, executables, resource management and provisioning)
- Resource selection (to decide where to execute a unit of work)
- Reservation

For the purpose of GNS setup and maintenance a new concept of destination endpoint needs to be introduced in G^2MPLS, as well as a configurable permeability for the services at the UNI. These concepts are detailed in the following sub-sections.

AAA issues for the Grid and the network layer impact service concept as well and they framework for their solution is sketched as well.

5.2.1 GNS transactions

A cornerstone of the G^2MPLS NCP architecture is the GNS transaction. A GNS transaction is a set of network calls deriving from the same job specification.

GNS transactions are related to the definition of a Grid job as per [OGF-GFD81], i.e. a manageable resource with endpoint references and managed by a job manager.

The GNS transaction is the container that provides a common root to different network call/connections traversing the same G^2MPLS User to Network reference point. It is shareable in a distributed way in the G^2MPLS NCP, in order to enable different invocation models.

The identified invocation models for the G²MPLS architecture are:

- *direct invocation model*, in which user is co-located at the same Grid site of the some Grid resources be involved in the job and issues a GNS transaction creation including also remote Grid sites;
- *indirect invocation model*, in which user is located remotely with respect to all the Grid sites to be involved in the job execution and issues a remote GNS transaction creation between them.

Some example of this invocation models are detailed in section 7 on G²MPLS use-cases.

For G²MPLS purposes a job executed among Grid sites can be decomposed into different job segments (parts) each with its own description of the requested/involved resources. An extension to this description could be the network resource specification, e.g. comprising:

- adjacency between Vsites (e.g. from site A to site B);
- network QoS specification
  - bandwidth requirements (e.g. 1 Gbps throughput)
  - delay
  - etc.

Each network description is similar to a job segment and is implemented through G²MPLS calls. The bundle of the calls related to the execution of a single job is the GNS transaction.

GNS transaction applies to the G²MPLS calls the ASON concept of call for connection segments. Further operative uses of the GNS transaction could include the GNS service tear-down when the job execution time occurs, or the coordination and escalation of recovery strategies in case of failure.

As depicted in Figure 5-10, the Grid job segmentation leads the establishment of two calls, the first bidirectional from Vsite A to Vsite B, the latter unidirectional from Vsite A to Vsite C. The resulting GNS transactions comprise two G²MPLS calls for Vsite A and just 1 call for Vsite B and C.

The G²MPLS call is an extension of the ASON call (an association between two or more users and one or more domains that supports an instance of a service through one or more domains) with further attributes, e.g. for temporal specifications.

The introduction of the GNS transaction concept allows fitting the typical Grid application use-cases in which multiple connections between different Vsites (end-points) are requested for the execution of the unique job. For example, in a distributed computing and visualization application (e.g. Kodavis) it might be possible to have one connection between a data storage resource/location and a computing resource/location and another connection between the computing resource/location and a visualization client. Similar examples can be described for distributed storage experiments.

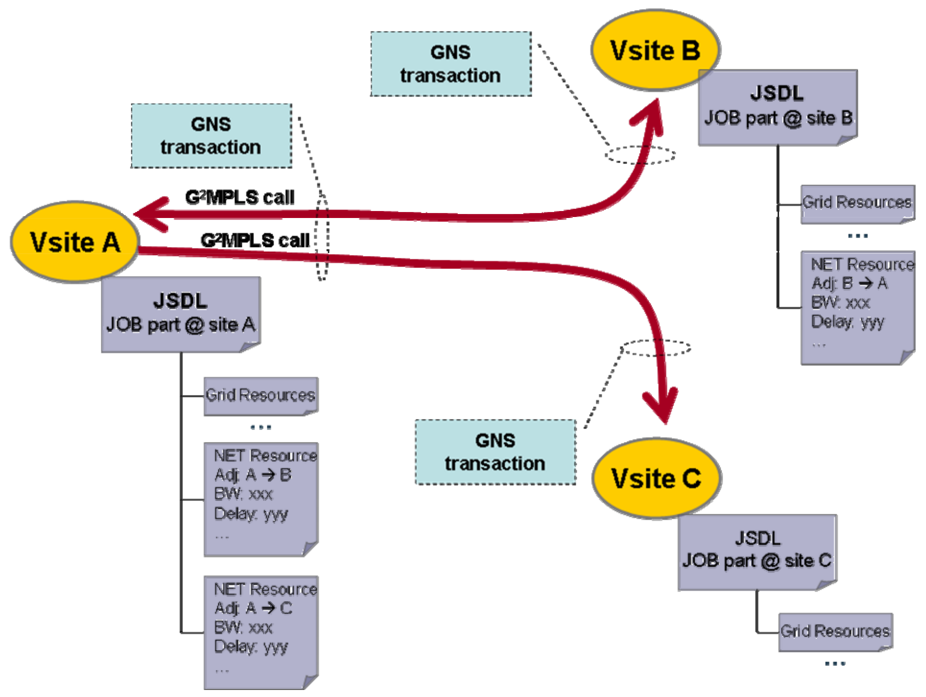


Figure 5-10: Grid job breakdown into GNS transaction and G^{2}MPLS calls.

5.2.2 A new concept of destination end-point

Depending on the specific application, the network attachment point could be part or not of the calls composing the GNS transaction. For example, in case of distributed computing and visualization (Kodavis use-case) network attachment points need to be declared (explicit declaration), in case of distributed storage they could not (implicit declaration).

In case of explicit declaration, the participating Vsites are specified along with the respective network attachment points (i.e. TNAs). For this purpose, these information need to be available at the Grid layer and the resource allocation algorithm at this layer should adopt some mechanism for selecting the best match in a set of choices (i.e. the number of network attachment points for a Vsite could be more than one, e.g. for resiliency purposes)

In case of implicit declaration, two cases may occur:

- participating Vsites are specified, but the network attachment point are implicit, which implies that the G^{2}MPLS NCP will choose the best match in the set of the available network attachment points for the selected Vsite;

- some of the participating Vsite are implicitly declared in the GNS transaction by asking for a service they can provide (e.g. an amount of CPU or storage), which implies that the G^2MPLS NCP will pick at first the best Vsite match in the set of the Vsites for that application and its GNS transaction requirements, and then the best match for the network attachment in the set of those available for the selected Vsites.

5.2.3 "Permeability services" offered to Grid users

The support of GNS, i.e. controlling the network resource in the same context of grid resources (e.g. CPU, storage, etc.), implies the permeability of the G^2MPLS Control Plane reference points to both types of information (i.e. network and grid).

In the User to Network direction, just information on grid capabilities and resources' availability need to be injected by the Grid layer (User) into the G^2MPLS cloud (Network). This service applies only to the G^2MPLS integrated model.

In the Network to User side network, different permeability behaviours could be configured depending on the capabilities of the involved layers. The identified levels of information permeability are:

a) No permeability, which implies the whole Grid and network co-allocation "brain" in on G^2MPLS. This model is similar to the configuration based reachability at UNI [IETF-RFC3945] and applies only to the G^2MPLS integrated model.

b) Permeability of grid resource information, which implies some part of the "grid brain" is in the Grid layer. This model could be considered a full peering reachability for grid resources only and applies only to the G^2MPLS integrated model.

c) Permeability of both grid and network resource information, which implies some part of the "network brain" is on the Grid layer for the network attachment endpoint selection. This model is similar to a silent listening because the Grid layer does not take part in the routing process of the network. It applies to both the G^2MPLS integrated and overlay models. Network information feed up from the Network to the Grid layer could be contain:

    - Endpoint reachability (mandatory)
    - Inner network topology information (optional)
        - Just inter-domain links between reachable domains
        - Inter-domain links and a summarized view of intra-domain links for each reachable domain

- Full topology information

Therefore, in the Network to User direction, the extent of permeability at the G.OUNI may range from pure grid (extreme summarization, e.g. a UNI-N advertising a UNI-C of the amount of storage and processor power reachable across the network) up to network information (i.e. topological information with different levels of details). In this latter case, the proposed permeability is not compliant with ASON, but it is still compliant with IETF [IETF-RFC3945] since it can be interpreted as a full peering or silent listening "routing model".

5.2.4 AAA services

Authentication, Authorization and Accounting is a fundamental service for the secure and trusted use of both Grid and network infrastructures. Access to the Grid computational resources and to the network transport connections need to be regulated and authorized, in order to preserve the final service integrity and the robustness of the various control plane procedures.

The management of network as well as grid resources in a multi-domain heterogeneous environment with different administrative entities is a non-trivial task and is matter of investigation in many research projects. For example, an Authentication and Authorization Infrastructure (AAI) has been defined in GÉANT2 project called eduGAIN ("educational" GÉANT Authorisation INfrastructure) to address the issue of ubiquity and roaming access to its connection services (including BoD). Other similar infrastructures are Shibboleth (Internet2), VOMS (EGEE), GLOBUS (GSI), OGSA AuthZ, etc. All these infrastructures define a set of interactions between an identity provider and a service provider to facilitate user access through a single sign-on mechanism according to a federation model for attribute exchange. The deployment of these infrastructures differs among the different Grid middlewares and network infrastructures, and a standard solution for the AAA problem for both Grids and networks does not exist yet. PHOSPHORUS WP4 is focused on this objective and is aimed to deliver an AAA model and infrastructure that aims to integrate the two dimensions, i.e. Grid and network.

G^2MPLS architecture raises natively the requirement for Grid and network AAA integration, because GNS transactions need to be seamlessly authenticated and authorised along the chain of Grid sites and network domains. The solution of this problem may require to interface the G^2MPLS Network Control Plane to an Authorization entity (AAA server), probably located out of the Control Plane and lying in the Service Plane. For this purpose additional architectural elements are needed as depicted in Figure 5-11 to:

- Forward the AAA request to the Authorization entity
- Enforce the policy decisions at each domain boundary, i.e. the UNI and the E-NNI.

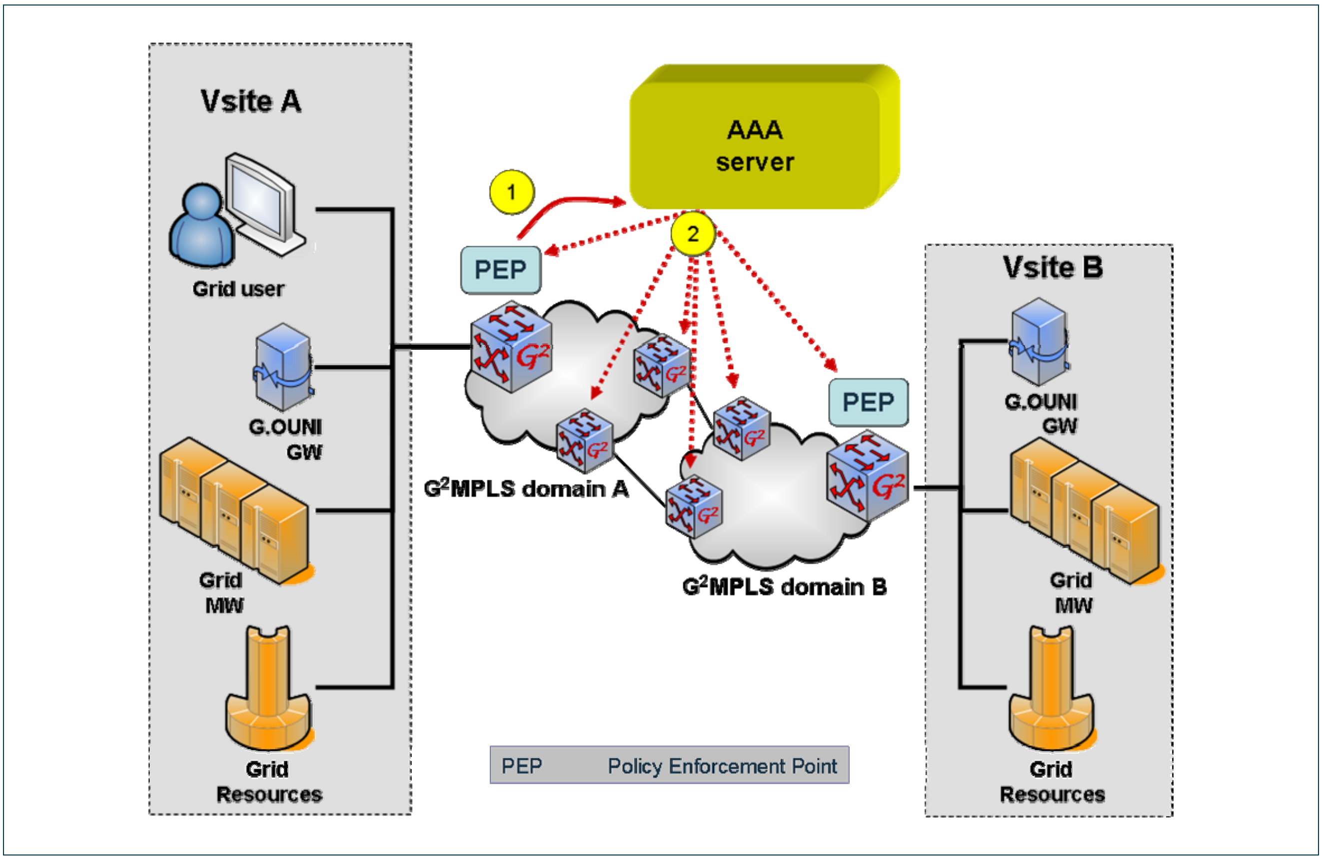


Figure 5-11: G^{2}MPLS and AAA infrastructure.

Further details fro the AAA integration with G^{2}MPLS Network Control Plane are under investigation and rely on a joint work between PHOSPHORUS WP2 and WP4 teams.

6 G²MPLS Procedures

This chapter covers a comprehensive description of Grid-GMPLS procedures with an end-to-end and cross-plane scope.

6.1 G²MPLS Transport Services setup/teardown

In this section a high level description of the special procedures for the setup/tear/down in the grid optical networking frameworks is provided.

6.1.1 Advance reservations

Advance reservation is a reservation scheduled for future (non-instantaneous) execution. Advance reservations in Grids are needed to cope with a guaranteed service at the time of execution of the job. Therefore, the requestor of the connection service (e.g. a Path Computation Client as referred in IETF RFC4655) is obliged to specify information when to create and discard the circuit. This can be provided in two ways, depending on access interface as:

- Start time and duration of the reservation, which guarantees that reservation will be available for specified period of time
- Start time and end time, which will make the system to tear down the circuit at specified time in the future.

Also the start and end time predefines the reservation type. In Table 6-1 there is a set of values that can be assigned for the time attributes:

Time specification	Start time	End time
1	Now	Defined for the future
2	Defined for the future	Defined for the future

Time specification	Start time	End time
3	Now	Unspecified
4	Defined for the future	Unspecified
5	Unspecified	Defined for the future
6	Unspecified	Unspecified

Table 6-1 Potentially allowed start and end time values

The reservation can start immediately (when start time is “now”) or in precisely defined moment in the future. It can be then finalized in the precisely defined moment in the future or remain opened until cancelled. It is also possible that neither start not end time are specified. When start time value is missing, it may be interpreted as if resources should be booked, but the reservation itself will be initialized after additional signalling. This implies that resources must be kept booked for all the time, since reservation is scheduled till infinite. If the reservation end time is specified at a specific moment in the future, then resources may be released at that moment if no activation signal arrives. Only the two first scenarios from Table 6-1, with all times defined as exact moments in the future, are in scope of PHOSPHORUS project. The very first case is in fact a special case of advance reservation, where the circuit creation should start immediately. This kind of reservation may have slightly different reservation process due to optimization issues.

In order to provide feasibility of advance reservations in G^2MPLS, a partitioning of the Transport Network resources is needed, by distinguishing resources to be used for bookings from those that could be used for immediate reservations (e.g. by standard ASON/GMPLS users). This approach is needed because it is impossible to check bookings on resourced used by immediate reservation connections for which the operational timeframe is not declared and should be assumed infinitive.

Since reservation request arrives to an ingress domain, it is expected that circuit feasibility check will be performed basing on the available network knowledge at current moment. This implies that booked resources and defined paths are loaded with the risk of failure at circuit creation time. In order to prevent such situations and minimize such risk, network monitoring is a strong requirement for advance reservation. As reservation may involve multi-domain and heterogeneous aspects, the failure prevention activity will include policy issues (hiding domain internals), efficient communication protocols (for notifications), and reliable issues solving actions (path recalculation, resources re-negotiations, crankback, etc.).

Because reservations are scheduled for the future, it is required to have a calendar instance for maintenance of resource bookings. Such calendars must be scalable and open solution in order to schedule various types of resources, including not only network parameters (like bandwidth, VLAN ids, SDH time slots, etc.) but also typical GRID attributes (number of processors, amount of memory, etc.). Each new incoming reservation must be faced against a calendar to avoid cross-booking of the same resources at the same time. The calendar may be kept centralized for single domain, or even for multi-domain environment, but it may be also distributed. The most common and easier choice to maintain scenario is to have centralized calendar for single domain purposes, and distributed approach for inter-domain purposes. Thus each domain has central reference point for available resources, but each domain can hide these details from other domains.

Since reservation is successfully scheduled and the reservation start time arrives, the circuit should be created on the data plane. The signalling process for that can be initialized in:

- reservation originating domain (which may be out of the reservation path)
- any domain on the reservation path (particularly the first domain on reservation path)
- each domain simultaneously

The most common approach for that is to use the second option, where initializing domain is the domain which is placed at the beginning of the reservation path (where reservation start point belongs). The third option of the above requires synchronization efforts, so that all the domains execute reservation at the same moment on time axis. Otherwise it will be difficult to guarantee that reservation is ready at a time specified by a user.

For PHOSPHORUS project, advance reservations may be performed in two ways:

- basing on domain centralized Path Computation Elements (PCE), or
- basing on domain distributed PCE.

According to IETF RFC4655, a Path Computation Element "is an entity that is capable of computing a network path or route based on a network graph, and of applying computational constraints during the computation". The PCE then integrates the calendar, topology discovery and routing functions enhanced with user constraints and policy rules appliance. The centralized PCE means that there is single entity responsible for all decisions, having all knowledge and enough computational power to fulfil path requests within single domain. It is possible to have a single PCE for multiple domains; however this implies domains to advertise their internal details. Therefore for PHOSPHORUS project, it is assumed that each domain is independent and PCE can be centralized only at single domain level (as per chapters 5.2 in RF4655). PCE can be also distributed within domain, so that multiple PCEs cooperate to fulfil user request (ref. chapters 5.3 and 5.4 in RFC 4655). Each PCE then has partial knowledge of the network, which may benefit in faster computation for large, complex networks.

Since domain is managed by single PCE, it is expected that this entity is well informed about network state and has access to calendar functionality in order to perform advance reservations. The local PCE may exchange information with neighbour PCEs for multi-domain path computation. It is advised that PCE has limited knowledge about all domain internals supported by overall control plane, at least as abstracted information. This is required for multi-domain path computation. In case PCE has full knowledge of neighbour topologies, it is possible to perform path searching instantly. Such path is then called "explicit path", as it contains all hoops from source to destination points. If PCE is allowed to see only abstracted topologies of neighbour domains, it is possible to find "strict/loose path". The "strict" term refers to a local domain, where all links and nodes are known, while "loose" refers to other domain on path, where only abstract nodes or LERs are known. The strict/loose paths must be sent to neighbour domains for further negotiation regarding domain internals. This can be performed either by PCC or PCE, while the second choice will be transparent and more convenient for PCC entity.

The computed reservation, including calendar related information and ERO may be stored on a PCC, PCE itself or LER. As LER is more a part of a Data Plane it should be aware of the reservation not earlier than after signalling, which takes place at reservation start time. A PCE entity should store scheduled reservation due to calendar functionality (unless calendar is separated entity for resource monitoring). As PCC is a client for path, it may also store scheduled path depending on its functionality (if it is just a simple user or advanced GRID application).

For multi-domain purposes calendar synchronization between peering domain is required. As domain may be represented by abstracted topology, its internals should be also abstracted in calendar. At least resources for inter-domain links should be advertised for neighbouring calendar entities. However, it is possible that calendars are not synchronized in any way, and it is a PCE instance which is responsible to validate resource availability on peer-to-peer basis.

The above scenario assumes that an advance reservation is performed at control plane, and signalling and data plane configuration is performed only at service start time. Therefore network devices are not aware of future resource bookings and this functionality is delegated to control plane software. The other possible scenario is that PCE is part of network devices or is strongly integrated with them, and advance reservation path searching is performed through signalling. In such approach the centralized calendar entity may be provided, or each network device has its own calendar functionality. This scenario equals to multiple PCE instances among single domain, where each of them is responsible only for a part of a network. In this particular case, the corresponding part is a single network device (as referred by RFC4655 in chapters 5.1 and 5.3).

For this scenario the resource calendars located at network nodes may be synchronized in the same way as routing is advertised. The most common way for that are OSPF-TE advertisements, with additional usage of Opaque LSA fields. The amount of calendar information, that is synchronized, directly affects the amount of communication messages that needs to be exchanged for reservation. If each device keeps only its own resource bookings and does not advertise it, the circuit scheduling process is expected to retrieve that information at reservation time. The communication process will continue until nodes with available resources will be found. On the other hand, if a network device shares its calendar with all other nodes, the circuit scheduling process can explicitly retrieve the needed information and directly select candidates with free resources for reservation. The overall conclusion from that is that the more resource information from calendars is distributed, the fewer amounts of communication messages are required to establish a reservation. It is advised that the exchanged amount of resource calendar information should be kept as minimum at the beginning. As the project will evolve, the selection of advertised resources should be constantly increased in order to achieve a balance between routing information exchange and communication overhead.

6.1.2 “Grid-fast” circuits set-up

Besides advance reservation setup procedures there are also on-the-fly path establishing procedures. These mechanisms have to create a path between grid resources as fast as possible. It is important in the optical data network because optical switches have a long switching time. “Grid-fast” circuit setup is done by the standard GMPLS exchange of the path setup messages:

- an ingress node receives setup request by G.OUNI, ask PCE for Explicit Routes object (ERO) and include this ERO in Path message sent to egress node,
- Path message travels through all intermediate nodes to egress node; in case of optical data network each switch is informed about planed switching in the data plane and immediately starts doing cross-connection,
- the egress node sends back Resv message when it prepared his own cross-connection - all intermediate nodes do the same,
- when the ingress node receives Resv message it sends to the egress node a PathConf message which indicates that path is ready for the traffic data.

6.1.3 Beyond point-to-point services

A typical usage of bandwidth reservation is the creation of point-to-point (p2p) network connections, but there is an emerging request from specific group of users to progress beyond such a service. For example, in the VLBI project a number of distributed radio-telescopes are simultaneously sending the large amount of data to single computation point for hardware correlation. Also GRID tasks that are executed on separated cluster environments may require high bandwidth connections between each other to synchronize computation data. Therefore, p2p service may be insufficient for some final user groups of PHOSPHORUS G^2MPLS, above all under the requirement of more flexible services, adjustable for needs of more demanding GRID users.

In G^2MPLS architecture, there are three types of connections with more than two end points:

- **Point-to-multipoint**
  P2MP is the connection type where single source of information is sending data to multiple destination point (ref. Figure 6-1). The special case of such situation is mentioned before as VLBI project, where data are send in reverse direction from multiple sources to single destination point. The point-to-multipoint connection can be easily represented as network star topology model.

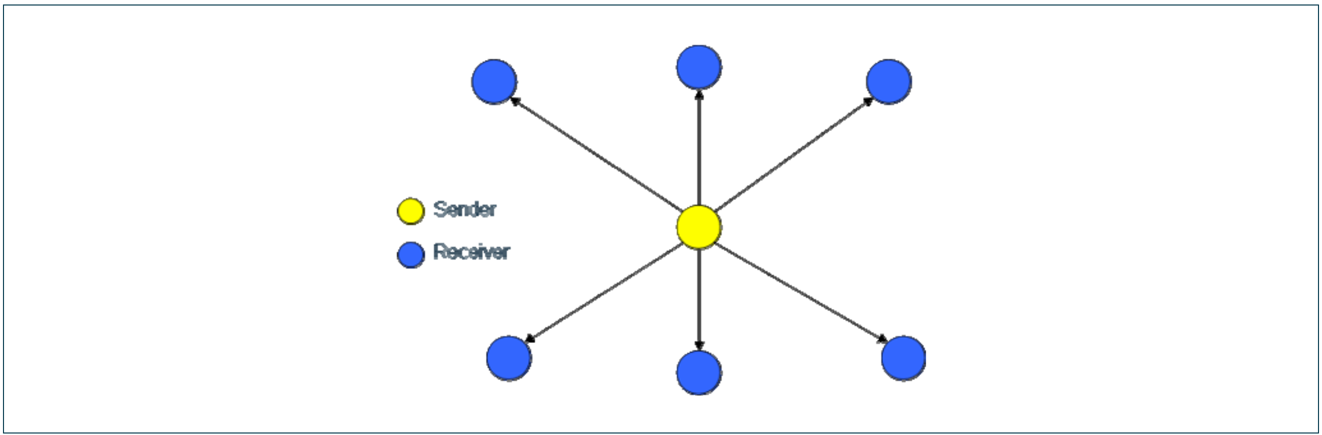


Figure 6-1: Point-to-multipoint connection.

- **Multicasting**
  Multicasting is the connections type where single source of information distributes the same data to multiple destination point simultaneously, optimizing link usage. Optimization here means that data are transferred over each link only once and are copied only where links splits to deliver content to single receivers (ref. Figure 6-2). The main idea of multicasting is to deliver the same content, at the same time to multiple users with minimum link utilization. Therefore multicasts transmission is often used for multimedia data streaming but also for computation data distribution (ftp) to multiple cluster infrastructures.

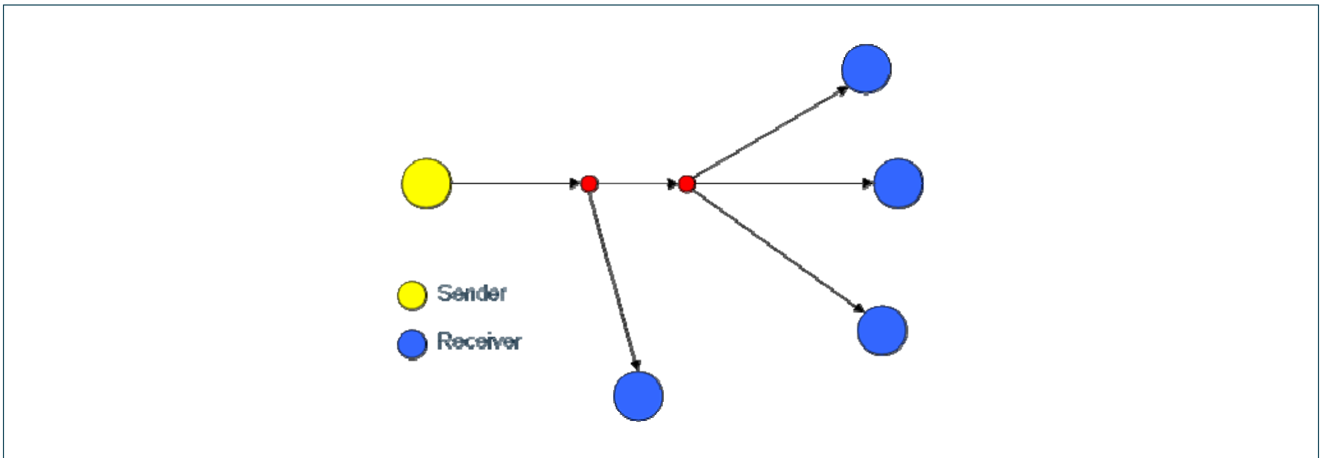


Figure 6-2: Point-to-multipoint connection.

- **Anycasting**
  Anycast is a type of data transmission where data are sent from single source to the nearest or best destination point. The group of receivers is identified with single routing address, however only one is receiving the data at the same time (ref. Figure 6-3). The G^2MPLS implementation of such a service

could be provided by processing at the Control Plane layer those GNS requests with implicit resource description (i.e. specification of just the involved Vsites or the amount of Grid resources – e.g. 50 TB – without any info on their network attachment points).

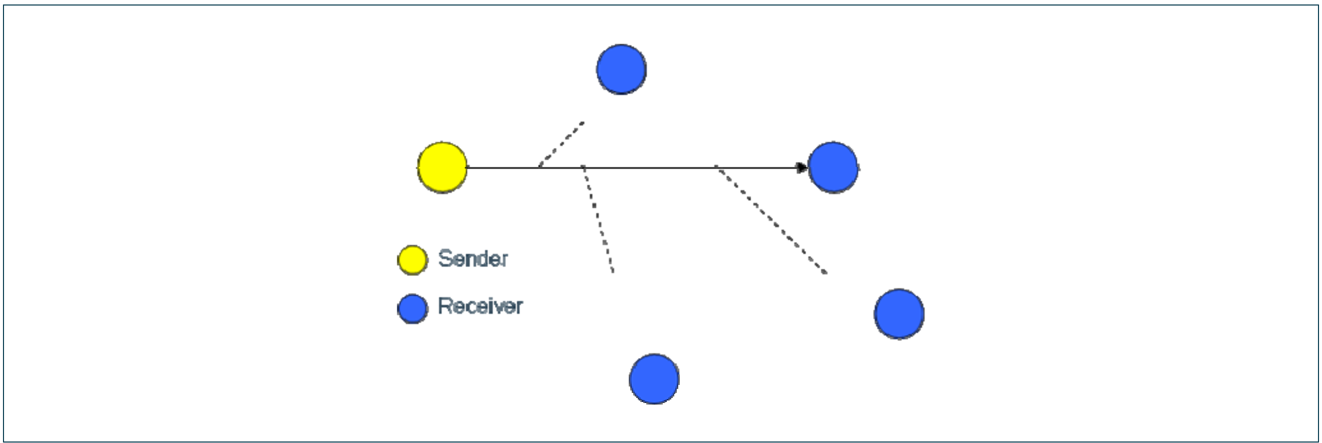


Figure 6-3: Anycasting.

These communication models require more investigation in context of G^{2}MPLS user requirements. At current view of the problem the most suitable implementation is point-to-multipoint connection, as this is already commonly used by GRID environments. Multicasting and anycasting are more experimental and are not likely to meet in operational computation infrastructures. Moreover, although the idea of multicasting is independent from network ISO/OSI layer model, this type of transmission is usually implemented in IP layer, which is out of scope for G^{2}MPLS network control.

6.1.4 Claiming of existing circuits

The implementation of new high performance networks using G^{2}MPLS Control Plane may be considered as a challenging solution. There are several reasons for that, but one of the most important is related to the problem of using the existing infrastructure and Management Planes. In many cases it may be required to run the G^{2}MPLS in an existing networking infrastructure, which is already populated with permanent connections managed by other entities.

The typical use case for this work is an introduction of G^{2}MPLS Control Plane in the domain managed by NRPS. In order to run G^{2}MPLS smoothly, some requirements must be met. This chapter focuses on the requirements for LSP conversion from NRPS to G^{2}MPLS Control Plane. Although it would be possible to make the conversion in an opposite way, i.e. from G^{2}MPLS Control Plane to NRPS, this type of conversion is out of the scope of this document.

From end user's perspective the possibility of turning a connection created and managed by particular NRPS (UCLPv2, DRAC or ARGON) or NMS in general into a connection fully managed by G^{2}MPLS is a valuable option. However, as it may happen that the connection under ownership modification is currently carrying end user's traffic, it is very important to achieve this goal without actually affecting the traffic. In other words, it should be possible to transfer the ownership from the Management Plane to the Control Plane, leaving untouched the actual state of Data Plane.

According to the draft specified by CCAMP Working Group draft-ietf-ccamp-pc-and-sc-reqs-00.txt, the following semantics of LSP is introduced:

- **LSP in Data Plane** – an LSP in Data Plane defines the forwarding or switching operations at each network entity. It is a sequence of data plane resources that achieves end-to-end data transport.
- **Management Plane LSP** – an LSP in Management Plane is the management state information (such as the connection attributes and path information) associated with and necessary for the creation and maintenance of a Data Plane connection.
- **Control Plane LSP** – an LSP in the Control Plane is the Control Plane state information (such as Path and Resv state) associated with and necessary for the creation of a Data Plane connection.

The draft states the most important requirements for conversion procedures, which must be fulfilled to make the process of transformation quick and transparent. The list includes the following:

Requirements	
Req. A	Data Plane LSP Consistency
Req. B	No Disruption of User Traffic
Req. C	Transfer from Management Plane to Control Plane
Req. D	Transfer from Control Plane to Management Plane
Req. E	Synchronization of state among nodes during conversion
Req. F	Support of Soft Permanent Connections
Req. G	Failure of Transfer

Table 6-2: Claiming of existing circuits – the requirements.

It should be ensured that the Data Plane LSP follows exactly the same path through the network and uses the same network resources (Req. A). Additionally, it should be ensured, that the conversion does not disrupt any user's traffic of the LSP being transferred or any other LSP in the network (Req. B). According to Req. C and Req. D, it must be possible to transfer the ownership from Control Plane to Management Plane and vice versa, from Management Plane to Control Plane. Before the conversion is started it should be assured that the state of the LSP is synchronized among all the nodes traversed by it (Req. E). Because it may happen that the path is a combination of permanent connection segments (usually at the source user-to-network side) and a switched connection segments (usually in the core of the network), it must be ensured that such hybrids Soft Permanent Connections are supported (Req. F). If during the transfer process some issues arise causing an incomplete result, it must be ensured, that at the end, the ownership is kept untouched and the traffic in Data Plane has not been affected.

6.1.5 Wavelength-agility

Wavelength agility provides very fast mechanism to select or change wavelength used to carry service between endpoints. Using reconfigurable optical add/drop multiplexers (ROADM) and tuneable laser devices it is possible to terminate, insert or switch a specific wavelength at a certain site.

Wavelength agility also significantly reduces the operational cost of optical networks because it provides increased flexibility and scalability of existing fiber links. In an all optical domain, the network topology can be dynamically and rapidly changed without integration in physical infrastructure. Logical connectivity can be updated in response to changing traffic patterns or triggers from failure/recovery procedures.

There are two possible scenarios for wavelength agility:

- one wavelength corresponds one TE link – each wavelength (lambda) is treated as single, separated TE link. This approach is very flexible and useful when the whole complete OXC switching matrix is available and manageable, i.e. no blocking conditions are declared for the switching matrix. Many single connections in different directions can be serving but management and monitoring issues are more complicated.
- many wavelengths correspond one TE link – the lambdas are grouped into one bundle TE link. In this case we can manage and monitor only the whole bundle interface and do not have access to parameters other than lambdas. The bundle TE link is useful when very high capacity is required in one direction and fine granularity is not so important. Since we have bundle links flexibility of this approach is limited but management is less complex.

6.1.6 Hierarchy of transport services

MPLS was designed to support only Packet Switching Capable (PSC) interfaces and the data forwarding based on a labels. GMPLS generalize the MPLS approach to facilitate the establishment of LSPs in variety of physical technologies.

The GMPLS hierarchy of transport services was defined basing on the devices interfaces switching capabilities, multiplexing mechanisms and concept of nested LSP. The hierarchy is created by multiplexing one or many lower layer LSPs into higher layer LSP. For example, several PSC layer LSP can be transported via L2SC layer LSP. To simplify structure or to reach higher efficiency some layers of the hierarchy model can be omitted e.g., L2SC layer LSPs can be directly multiplexed into LSC or FSC interfaces skipping TDM layer.

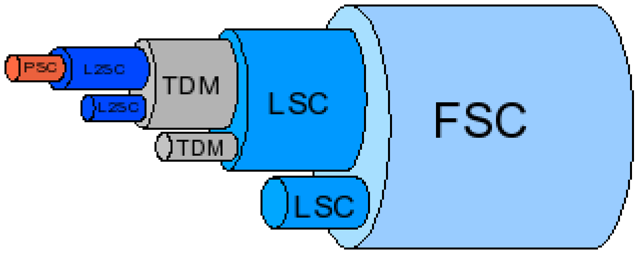


Figure 6-4: LSP hierarchy.

There are five classes of interfaces defined in [IETF-RFC 3945]:

- **Packet Switching Capable (PSC)** - interfaces that recognize packet boundaries and can forward data based on the content of the packet header. Examples include interfaces on routers that forward data based on the content of the IP header and interfaces on routers that switch data based on the content of the MPLS "shim" header.
- **Layer Two Switching Capable (L2SC)** - Interfaces that recognize frame/cell boundaries and can switch data based on the content of the frame/cell header. Examples include interfaces on Ethernet bridges that switch data based on the content of the MAC header and interfaces on ATM-LSRs that forward data based on the ATM VPI/VCI.
- **Time Division Multiplexing Capable (TDM)** - Interfaces that switch data based on the data's time slot in a repeating cycle. An example of such an interface is that of a SONET/SDH Cross-Connect (XC), Terminal Multiplexer (TM), or Add-Drop Multiplexer (ADM). Other examples include interfaces providing G.709 TDM capabilities (the "digital wrapper") and PDH interfaces.
- **Lambda Switching Capable (LSC)** - Interfaces that switch data based on the wavelength on which the data is received. An example of such an interface is that of a Photonic Cross-Connect (PXC) or Optical Cross-Connect (OXC) that can operate at the level of an individual wavelength. Additional examples include PXC interfaces that can operate at the level of a group of wavelengths, i.e., a waveband and G.709 interfaces providing optical capabilities.
- **Fiber Switching Capable (FSC)** - interfaces that switch data based on a position of the data in the (real world) physical spaces. An example of such an interface is that of a PXC or OXC that can operate at the level of a single or multiple fibers.

A range of signal types (and consequently bandwidth values) needs to be available in the G^2MPLS Network Control Plane, ranging from the Gigabit Ethernet up to the SDH-SONET TDM hierarchies and 10 Gigabit Ethernet, in optical technologies where possible. This broad range of granularities might require for the use and control of hierarchical and stitched Label Switched Paths across the different technology domains. The use by client LSP of higher granularity bi-directional LSP is going to be standardized in IETF for standard GMPLS through the binding of carrier LSPs with dynamical Forwarding Adjacency LSP (FA-LSP) or Routing Adjacency (RA) [draft-ietf-ccamp-lsp-hierarchy-bis-01.txt]. This approach is applicable in the G^2MPLS Network Control Plane in support of hierarchical network transport services.

6.1.7 Optical constraints

The optical networks generally deployed in operational networks are "opaque", i.e. each link is optically isolated by transponders performing optoelectronic (O/E/O) conversions. Transponders provide 2R/3R regeneration (re-amplification, reshaping and in case retiming), which eliminates transparency to bit rates and frame format. O/E/O conversions are quite expensive, and tend to be bit rate and format specific, thus reducing the flexibility of the optical network and its capability for a fast evolution of the relevant services. This supports the migration towards all-optical sub-networks, also called "domains of transparency”.

In all-optical networks the signals are transported end-to-end optically, without being converted to the electrical domain along their path. This reduces complexity and overheads and offers reduction of unnecessary and expensive optoelectronic conversions. However, due to the analogue nature of the optical networks as the optical signals propagate through the fibers, they experience several impairments degrading their performance. This has a direct impact on the dimensions that an all-optical network can support.

Domains of transparency can be used as a compromise between all-optical and opaque networks. In these networks selective regeneration is used at specific network locations as needed in order to maintain acceptable signal quality from source to destination. It is important to note that the issues associated with signal impairments in optical networks become significantly more challenging for higher data rates.

Exploiting impairments in routing

There are many types of physical optical impairments. The number of parameters that needs to be managed at the Control Plane level through on-line distributed routing depends on many variables, including:

- Equipment design choices (e.g. terminal and node architectures, device performance characteristics),
- Fiber characteristics (e.g. chromatic dispersion and polarization mode dispersion parameter, attenuation),
- Service characteristics (e.g. circuit speeds),
- Network size, (e.g. link lengths, span lengths, number of cascaded nodes)
- Network operator engineering and deployment strategies(e.g. dispersion map applied)

Optical impairments can be classified into two categories, linear and nonlinear. Linear impairments are independent of signal power and affect wavelengths individually. Some examples of linear impairments are amplified spontaneous emission (ASE), polarization mode dispersion (PMD), crosstalk and chromatic dispersion. On the other hand nonlinear impairments are significantly more complex and affect not only each individual optical channel through self-phase modulation (SPM), but they also cause disturbance and interference between them through cross-phase (XPM) and four wave mixing (FWM) effects.

To overcome the problems caused by the impairments at the physical layer, dynamic impairment management techniques may be implemented in-line (e.g. optical means of impairments compensation) or at the optical transponder interfaces (e.g. electronic mitigation of impairments). From the network layer view, the implementation of certain RWA algorithms that consider signal impairments and constraint the routing of wavelength channels according to the physical characteristics of the optical network paths can further improve the performance and minimize the blocking probability of connection requests. One approach that can be followed towards this direction is to identify a set of impairments that are considered independently and a path is assumed to be feasible if a set of criteria reflecting the signal quality in terms of the different impairments are satisfied. An alternative and more accurate approach is to identify a specific metric that considers all the physical impairments experienced by the optical signal and their interplay such as e.g. the performance factor Q as it is formed in the presence of a variety of impairments and consider this a routing parameter.

Polarization Mode Dispersion (PMD)

Polarization Mode Dispersion is the most important polarization effect for high capacity, long haul systems with high bit rates. PMD arises from the birefringence in the fiber that gives rise to the differential group delay between the two principal states of polarization. PMD is manifest as a time varying and statistical pulse broadening and pulse distortion and it is also dependent on environmental variations, particularly temperature. Because of the statistical nature of PMD, the differential group delay increases with the square root of the length of the fiber and is expressed in units of pico-seconds (ps) per sqrt(km) which is the measured unit of the PMD parameter (Dpmd). For PMD management the following formula should be true [STRAND-01]:

$$B\sqrt{\sum_{k=1}^{M} D_{PMD}^{2}(k)\cdot L(k)} < a$$

where **M** is the total number of spans in the circuit, **B** is the bit rate of the channel in sec-1, $\mathbf{D_{PMD}(k)}$ is the fiber PMD parameter in ps per sqrt(km) of the k-th span in the circuit, **L(k)** is the length in km of the k-th span. **a** is a design parameter whose typical value is around 0.1 (10%).

If all the spans have the same $\mathbf{D_{PMD}}$ parameter, then the equation becomes:

$$BD_{PMD}\sqrt{L} < a$$

where L is the total length of the circuit.

Typical values for $\mathbf{D_{PMD}}$ range from 0.1 ps / per sqrt(km) for newer types of fiber, to 0.5 ps / per sqrt(km) for older fibers. Thin means that for newer fibers with a PMD parameter of 0.1 picosecond per square root of km, the maximum length of the transparent segment (without PMD compensation) is limited to 10000km at 10Gb/s bit-rate and 625km for 40Gb/sec transmissions

In general, the PMD requirement is not an issue for most types of fibers at 10Gb/s or lower bit rate. But it will become an issue at bit rates of 40Gb/s and higher. However, the availability of PMD compensation devices reset the PMD constraint that disappears from the routing perspective.

Amplified Spontaneous Emission

Optical amplifiers are used periodically within fibre links and at the various nodes present in the network to compensate for fibre and other optical component related attenuation. Unfortunately, optical amplification is not possible without the generation of amplified spontaneous emission (ASE) noise. Each optical amplifier contributes ASE, and these contributions add cumulatively along the amplifier chain. This accumulated ASE gives rise to signal-spontaneous beat noise at the receiver, which is the fundamental noise limit in an optically amplified transmission system and degrades the optical signal to noise ratio (OSNR), which depends on the bit rate, the transmitter-receiver technology (e.g., FEC), and margins allocated for the impairments, and needs to be maintained at the receiver. In order to satisfy this requirement, vendors often provide some general engineering rule in terms of maximum length of the transparent segment and number of spans. For larger transparent domains, more detailed OSNR computations are needed to determine whether the OSNR level through a domain of transparency is acceptable. This would provide flexibility in provisioning or restoring a lightpath through a transparent sub-network.

If the path traverses **M** optical amplifiers each one with gain **G(k)** (**k=1..M**), then the maximum value for **M** is obtained by the following relationship:

$$M \le \left\lfloor \frac{P_L}{2h\nu B_0 \cdot SNR_{\min} \cdot \sum_{k=1}^{M} n_{SP}(k)(G(k)-1)} \right\rfloor$$

where $\mathbf{P_L}$ is the average power launched at the transmitter, $\mathbf{n_{SP}(k)}$ is the spontaneous emission factor of the k-th amplifier, $\mathbf{h = 6.63 \cdot 10^{-34}}$ **J/Hz** is the Planck's constant, ν is the carrier frequency and $\mathbf{B_0}$ is the optical bandwidth. If all the spans have the same **G(k)** and $\mathbf{n_{SP}(k)}$ the formula becomes:

$$M \le \left\lfloor \frac{P_L}{2h\nu B_0 \cdot SNR_{\min} \cdot n \cdot (G-1)} \right\rfloor$$

Assuming P=4dBm, SNRmin=20dB with FEC, B=12.5GHz, n=2.5, G=25dB, based on the constraint, the maximum number of spans is at most 10. However, if FEC is not used and the requirement on SNRmin becomes 25dB, the maximum number of spans drops down to 3.

Most impairments generated at OXCs or OADMs, including polarization dependent loss, coherent crosstalk, and effective passband width, could be approximated with a domain-wide margin on the OSNR, plus in some cases a constraint on the total number of networking elements. However, detailed models can be also used to accurate calculate the impact of these impairments on the overall signal quality.

Polarization-Dependent Loss (PDL)

PDL accumulates in a system when a number of components with polarization dependent loss or gain performance are cascaded along the transmission path. The state of polarization fluctuates with time and its distribution is very important. It is generally required to maintain the total PDL on the path to be within some acceptable limit, potentially by using some compensation technology. Generally, this impairment translates in a maximum allowable number of spans declared by the equipment vendor.

Chromatic Dispersion

Chromatic dispersion or group velocity dispersion (GVD) has been considered for many years to be the most serious linear impairment for systems operating at bit rates from 2.5 Gbps to 10Gbps and is causing different frequencies of light to travel at different speeds. This linear process causes broadening of the optical pulses, resulting in inter-symbol interference which impairs system performance and imposes limitation of the maximum attainable transmission distance.

In general this impairment can be adequately (but not optimally) compensated for on a per-link basis, and/or at system initial setup time, e.g. through the use of Dispersion Compensating Fibre (DCF). It is noteworthy that the effect of GVD combined with fibre nonlinearities, such as self- and cross-phase modulation (SPM/XPM) and four-mixing (FWM), is much more complicated since GVD can increase or alleviate the effects of fibre nonlinearities. Chromatic Dispersion can be handled semi-analytically together with Self Phase Modulation and evaluate an eye closure penalty induce from the combination of the two impairments.

Crosstalk

Optical crosstalk refers to the effect of unwanted leakage of an optical signal transmitted through optical filters, switches and other optical components introducing interference to itself or other signals. It includes both coherent (i.e. intrachannel) crosstalk and incoherent (i.e. interchannel) crosstalk. Main contributors of crosstalk are the OADM and OXC sites that use a DWDM multiplexer/demultiplexer (MUX/DEMUX) pair.

For a relatively sparse network where the number of OADM/OXC nodes on a path is low, crosstalk can be tolerated and ignored at routing level. But for some relatively dense networks the per-link crosstalk information should be propagated to make sure that the end-to-end path crosstalk (sum of the crosstalks on all the component links) is within some limit, e.g. -25dB threshold with 1dB penalty. Another way to treat this impairment without flooding of routing information is to have a routing algorithm that adds extra-costs for each traversed OADM/OXC nodes that has a MUX/DEMUX pair: this can ensure the selection of the best (i.e. lower crosstalk) route in the transparent domain.

Effective Passband

The effective passband of a transparent domain narrows and spectral clipping may be introduced due to filter narrowing effects introduced when cascading a number of optical filters along the optical path. In general, this is a system design issue, i.e., the system is designed with certain maximum bit rate using the proper modulation format and filter spacing. Traffic at lower bit rates can tolerate a narrower passband, thus the maximum allowable number of narrow filters will increase as the bit rate decreases.

Nonlinear Impairments

Fiber nonlinearities fall into two categories. One is stimulated scattering (Raman and Brillouin), and the other is the optical Kerr effect due to an harmonic motion of bound electrons in the material resulting in an intensity dependent refractive index with optical power [Agrawal]. While stimulated scatterings are responsible for intensity dependent gain or loss, the nonlinear refractive index is responsible for intensity dependent phase shift of the optical signal. Nonlinearities stemming from the Kerr effect occur due to the nonlinear relationship between the induced polarization P and the applied electric field E when higher powers and /or bit rates are applied as shown in :

$$P = \varepsilon_0 (\chi^{(1)} \mathrm{E} + \chi^{(3)} \mathrm{E}^3 + ...)$$

where ε_0 is the permittivity of vacuum and $\chi^{(j)}$ the j-th order susceptibility. The linear susceptibility $\chi^{(1)}$ is the dominant contribution to the polarization ***P*** and its effects are included through the refractive index n [Agrawal]. The cubic term $\chi^{(3)}$ is responsible for phenomena like third-harmonic generation, four wave mixing and nonlinear refraction. The first two processes (processes that generate new frequencies) are usually not important unless phase matching conditions are satisfied. Nonlinear refraction instead is always present and deeply affects the propagation of intense light in an optical fibre. The electromagnetic wave passing along the optical fibre induces a cubic polarization which is proportional to the third power of the electric field. This is equivalent to a change in the effective value of $\chi^{(1)}$ to $\chi^{(1)} + \chi^{(3)}E^2$. In other words the refractive index is changed by an amount proportional to the optical intensity :

$$\tilde{n}(I) = n + n_2 I$$

where *n* is the linear part , *I* is the optical intensity and n_2 is the nonlinear-index coefficient related to $\chi^{(3)}$ by

$$n_2 = \frac{2}{\varepsilon_0 c n} \frac{3}{8n} \chi^{(3)}$$

This intensity dependence of the refractive index (optical Kerr effect) is responsible for numerous nonlinear effects. To note that even if the value of the nonlinear coefficient n_2 is quite small, nonlinear effects in optical fibres assume a relevant importance due to that the magnitudes of these effects depend on the length of the fiber along which the signal travels and on the ratio n_2/A_{eff}, where A_{eff} is the effective area of the lightmode. Despite the intrinsically small values of the nonlinear coefficient for silica, the nonlinear effects in optical fibers can be observed even at relatively low powers considering that the light is confined in a relative small area over long interaction lengths due to the extremely low attenuation coefficient and the presence of optical amplifiers.

One manifestation of the intensity dependence of the refractive index occurs through SPM, a phenomenon that leads to spectral broadening of optical pulses travelling along a fiber. XPM is similar to SPM but in this case the induced phase change depends not only on the intensity of the transmitted signal itself, but also on the other co-propagating WDM channels. Another nonlinear effect that can be relevant in optical fibers is FWM.Due to this phenomenon new signal frequencies are generated that coincide with some of the already existing the transmission channels inducing interferences which degrade the transmitted signal quality. Fast and accurate analytical models can be developed to calculate the nonlinear impairments and incorporate them in the path computation process. These models would likely require very detailed knowledge of the physical infrastructure, including measured dispersion values for each span, fiber core area and composition, as well as knowledge of subsystem details such as dispersion compensation technology. This information would need to be combined with knowledge of the current loading of optical signals on the links of interest to determine the level of nonlinear impairment.

It could be assumed that nonlinear impairments are bounded and result in 1 or 2 dB margin in the required OSNR level for a given bit rate, consequently setting a limit on the maximum number of spans.

An Alternative Approach – Maximum Distance as the only Constraint

A rule of thumb for operational all-optical networks is to limit the maximum distance of point-to-point connections, given a fixed per-span length based on the OSNR constraint for a given bit rate. This translates impairments in lengths contributions per each span and the constraint to be matched by routing is that the sum of all the link-distance over all links of a path should be less than the maximum-path-distance

A basic issue – wavelength availability

Routing in an all-optical network without wavelength conversion implies that the chosen wavelength is available on all links. This information needs to be advertised by the routing process. Surely, advertising detailed wavelength availabilities on each link raises scalability issues, but depending on the size of the network this burden could be acceptable.

Implications on G^2MPLS routing

A number of parameters have been described in this section for managing the optical constraint at the Control Plane level for all-optical networks. For a G^2MPLS TE link belonging to an all-optical domain the following parameters should be specified and flooded via G^2.OSPF-TE:

- D_{PMD}
- Physical length (in Km)
- List of amplifiers (include their gain G and noise figure n_{SP})
- List of available wavelengths.
- Dispersion values
- Input power levels

Impairment should change slowly and they could be measured at the installation time and possibly refreshed as a result of measurement campaigns. This relaxes the requirement for continuous and invasive measurements in the all-optical network.

Further investigations will be provided by joint research activities among WP2, WP5 and WP6 during the project lifetime, in order to identify the relevant optical constraints for the PHOSPHORUS test-bed and some reasonable values for them, e.g. derived by specific simulations.

6.2 G^{2}MPLS transport service modifications

To reach optimal grid and network utilization and fulfil dynamic requirements of some grid tasks G^{2}MPLS has to support transport service modifications.

There are three main scopes of network and grid parameters modifications:

- parameters decrease – it is the easiest modification to handle. In most cases it can be done “on the fly” only by some reconfiguration request and without any connectivity or computation breaks;
- parameters increase – in some cases very difficult to serve. Availability of additional resources have to be checked and, in several cases, decision has to be made with MW consultation:
  - additional network resources – if requested additional bandwidth is available for existing path only some path parameters modifications are required. In case when bandwidth is bigger than link capacity the procedure of new path computation is triggered. If there is an alternative path, new connection is created and whole task communication traffic is switched on the new link without middleware interference. If an alternative path that fulfil modification request does not exist middleware has to make decision about task continuation with current path configuration or task aborting because of lack of available resources.
  - additional grid resources – processes performed in grid domain are quite similar to that in network domain. When new requested resources are available on the same cluster they can be simply allocated for the task. For additional resources accessible on different grid system new communication path is required, it is an indirect additional network resources request. In case, if additional resources can not be found middleware is responsible for future task workflow. MW can break task and dump process image on disk or let task complete computations.
  - additional network and grid resources – network and grid resources are requested jointly. Process of allocation additional resources can be treated as combination of network resources modifications and grid resources modifications with additional dependant constraints.
- relocation – the most complicated case. Practically new resource reservation process must be performed to fulfil the request. When new resources are allocated all previously reservation are removed and no longer supported. Only in case of network path relocation we can try to find new connection and switch the path without interference in grid resources. This modification in some cases is similar to recovery procedure.

6.3 G^2MPLS recovery

To deliver reliable services G^2MPLS, like GMPLS, requires a set of procedures to provide protection or restoration of the data traffic. In case of failure occurrence on working LSP, these recovery procedures have to be fast enough not to disrupt application's data connections. There are many recovery schemes with different levels of pre-provisioning. The recovery types could be categorized by time and resource consumption, setup vulnerability, quality of protection, packet loss, failover coverage [RFC 3469]. A list of some common recovery strategies is provided in Table 6-3:

Recovery type	**Description**	**Recovery time** *[0, 10] scale*	**Resource consumption** *[0, 10] scale*
Full LSP Re-routing	This is a recovery scheme without path pre-computation, which is the most flexible solution.	10	2
Pre-planed LSP Re-routing without Extra-Traffic	This is a recovery scheme resource pre-selection and without resource pre-allocation. It can be referred also as "shared mesh" recovery (there is no resource pre-selection).	6	4
LSP Protection with Extra-Traffic	This is a recovery scheme with resource pre-allocation. Protection LSP can be used for transporting additional low priority traffic. Resource consuming optimization is available by N:M path mapping type.	4	6
Dedicated LSP Protection (1+1)	This is a recovery scheme with resource pre-allocation which doesn't allow sharing of the recovery resources. It is the best time performance and the less flexible option.	2	10

Table 6-3: The mainly used recovery schemes

As it is shown, every recovery model has advantages and disadvantages but from a user point of view protection schemes achieve the best performance. However, using protection schemes, especially in dedicated 1+1 scheme, implies a higher consumption of network resources in entire operator's network.

Recovery types can be also divided by spatial factor. A label-switched path may be subject to local (span), segment and/or end-to-end recovery [RFC 4426]:

- span protection is protection of link between two neighbouring switches,
- segment protection refers to recovery of an LSP segment (Sub-Network Connections in the ITU-T terminology),
- end-to-end protection refers to protection of entire LSP from the ingress to the egress port.

Multiple recovery levels can be used concurrently by a single LSP for added resiliency. This model is useful in the multi-domain environment where an LSP consists of parts in different domains. Every domain's part of the

LSP can be a segment which his own protection type. Unsuccessful segment recovery, policy aspects or an inter-domain link failure should launch an end-to-end recovery. Restoration or protection path may be link and node disjoint with the working path. It may follow by alternative inter-domain links or through different intermediate domains. Inter-domain recovery issue in the context of G^{2}MPLS architecture composes of two particular situations:

- recovery is coordinated between two or more G^{2}MPLS domains by G.E-NNI,
- recovery is coordinated between G^{2}MPLS domain and NRPS-controlled domain by G.E-NNI or another NRPS/G^{2}MPLS interface.

Recovery coordination between NRPS-controlled domains is out of scope of this document.

Crucial issue for the service resiliency is to use multiple links between Grid nodes and G^{2}MPLS network and use more than one network node to end these links. It makes possible to recovery last link or node failure.

G^{2}MPLS takes care about network and grid resources. Therefore, not only network recovery procedures are necessary but also some grid recovery procedures. However, any grid node failure can interrupt the grid workflow. If a network or Vsite failure breaks grid's procedures then recovery has to be escalated to grid middleware triggered by fault notification by G.OUNI. Grid middleware releases all already used resources, re-schedules job and sends new request to G^{2}MPLS Control Plane.

The G^{2}MPLS Control Plane recovery is part of the G^{2}MPLS robustness and it is detailed in the following section 11.

6.4 Crankback applied to set-up and restoration

Path setup can be blocked by links or nodes without sufficient resources. It can happen when information used to compute a path is out of date. In crankback scheme [IETF-CRANKBACK], failure information is returned from the point of failure to some ingress LSR (repair point) where new path setup is made to avoid the blocked resources. In particular conditions, it may be no available way to send a new path message to the same egress node (destination node). In that case we could consider escalation to middleware or selection of new grid node to which path will be send. In the first scenario, the grid middleware probably is responsible for sending a new job request to G^{2}MPLS Control Plane. The second scenario is possible if Grid middleware used the implicit service identification (i.e. without any network attachment point specification) and Traffic Engineering Database (TED) in G^{2}MPLS Control Plane nodes contains alternative grid resources. Crankback operation could be repeated a few times. When the number of retries is exceeded the path-making efforts will be ended and job error response is sent to the middleware.

Crankback is also a very good working mechanism in case of LSP re-routing when a failure occurs, because the crankback error message (RSVP PathErr or Notify) carries precise and fast the failure details to the ingress node, so next computed path should avoid the known failures. This mechanism is known as crankback routing.

However, there is not specification how use the crankback routing for future path computation or to update and flood TED.

In case of non-explicit abstract node or a loose hop in Path ERO (Explicit Routes Object) there is a problem of identifying which hop in the explicit route has failed, because the lack of some topology details in computing PCE. In that case Route Exclusions object should be used in a new computed path setup to avoid some nodes or links.

When crankback operation is repeated all information about experienced blockages or failures should be gathered and used in further re-routing attempts so the resulting path should detour all known network problems. The gathered information should be discarded after some time period at least.

Crankback can be applied to advance reservation mechanism to search advance path within network not fully recognized by routing protocol. Thus PCE can provide ERO for a new Path message. This ERO may contain intermediate nodes which haven't provided future links availabilities for PCE. Because there is a higher possibility of adversity it can be normal to repeat crankback procedure a few times. The crankback operation timeout can be greater in advance reservation that in on-fly path setup. Additionally, the crankback error message should be able to carry the full future links availabilities on taken path (time extended crankback routing). This information can be used in PCE to compute next advance paths.

Crankback could be used in case of grid resource unavailability when Path message come to Vsite. Detailed information about this lack can be sent back to the ingress node and alternative grid resources can be applied for the non-explicit grid resources job request.

6.5 Auto-discovery procedures

Auto-discovery mechanism is a very important feature, which allows finding quickly actually available resources or acquiring updates when resources become available or unavailable in the domain. The result of auto-discovery is the discovery of connectivity between the client and the network and the services available from the network. The following chapter presents the discussion of auto-discovery mechanisms on the background of Data Communication Network (DCN), Transport Network (TN) and G^2MPLS service capability.

6.5.1 DCN related

As per ITU G.7712, the DCN is a network that supports Layer 1 (physical), Layer 2 (data-link), and Layer 3 (network) functionality. A DCN can be designed to support transport of distributed management communications related to the TMN, distributed signalling communications related to the ASTN, and other operations communications (e.g. voice communications, software downloads, etc.).

A DCN may be designed in such way, that only IP layer is supported. A DCN supporting only IP may consist of various sub-networks using different physical and data link layer protocols, however, all sub-networks will support IP as the network layer protocol.

It is assumed, that the network is organized into a hierarchy of Routing Areas. Additionally, each control domain has at least one Routing Controller. Routing Controllers using auto-discovery mechanisms learns the topology and keeps the knowledge about their peer Routing Controllers running within the common parent Routing Area. As per [OIF-E-NNI-Rtr-1.0], it might be possible, that RCs finds the topology via configuration, i.e. network administrator is responsible for establishment of control adjacency between peer RCs.

As per [OIF-E-NNI-Rtr-1.0], each RC should be configured to run as an OSPF-TE node to exchange OSPF-TE messages with its neighbours. Those RCs form routing adjacencies in the Control Plane. Each of them advertises corresponding inter- and intra-domain links and all the reachable TNA addresses for its domain.

6.5.2 TN related

In G^2MPLS architecture it is assumed, that the Network Control Plane is working on the abstraction of the actual network topology. Since the abstraction has to be maintained and updated, a new component is introduced – Transport Resources Controller. Its main responsibility is to build an abstraction of the technology specific details of the switching resources and keep it updated.

From the Network Control Plane point of view, it is important to have an access to information about bindings between technology specific name spaces for transport resources. This includes couples: <port, wavelength> for DWDM equipments, <port, virtual container> for TDM, <port, VLANs> for Ethernet and <data-link, label> for G2MPLS. Since the NCP is responsible for the communication with Transport Plane devices, it should be possible to translate technology specific configurations as for transport as for Grid (G^2MPLS) resources available in the network.

Those requirements presented above form a rationale for the existence of the Transport Resources Controller in G^2MPLS functional architecture.

6.5.3 Service (network & grid) capability related

According to [OIF-UNI1.0R2-COMM] service discovery is the process by which a client device obtains information about available services from the transport network and the transport network obtains information about the client UNI signalling (i.e. UNI-C) and port capabilities. To make a service discovery mechanism useful in G^2MPLS functional architecture, this definition should be extended to support Grid capabilities offered by Vsites attached to the G.OUNI.

In G^2MPLS architecture, the service discovery process will be supported by a component called GNS Service Discovery Agent (G-SDA). Both functions (discovering network and grid capabilities) will be realized here and most important features should cover:

- discovery of signalling protocol capabilities (type of signalling protocol, its version, etc.)
- discovery of network capabilities (routing diversity, transparency in case of SONET/SDH, etc.)
- discovery of Grid capabilities (type of CPUs, storage compilers, etc.)

In addition to the procedures of service discovery, it is also important to consider the procedures of discovery a Grid and network resources (i.e. their amount) available in the network. Both, network and grid resources discovery procedures, allow a transport network elements and directly attached client devices to discover their connectivity automatically and verify connection parameters, as well as discover all Grid resources currently available in the network.

The functionality of discovery network and Grid resources will be supported by a component called Discovery Agent (DA). Because the functionality of DA covers the functionality of the ASON DA, the G^2MPLS DA will be designed to expose the same interfaces. However, in addition to ASON's functionality, G^2MPLS DA will be also responsible for discovering and verification of Grid resources available at Vsites attached to the G.OUNI, thus the exposed interfaces should be extended accordingly.

Although, as defined in [OIF-UNI1.0R2-COMM], the procedures for service and resources discovery are optional, i.e. all compliant with UNI 1.0 signalling implementations are not required to include them, it is recommended to consider them as a mandatory components in G^2MPLS implementation. In the case both procedures are not implemented, the information provided by these procedures must be manually configured.

6.6 Management functionalities

For the network operator the management of G^2MPLS Control and Data Plane is a very important issue. It allows the operator to detect and diagnose the failures in the network. The management platform receives notifications from the devices (network monitoring) and helps to localize the cause of the problem.

From GMPLS perspective the following management functionalities are essential in order to deal with failures in control and data plane:

- detection and diagnosis of broken LSPs
  There should be a mechanism which allows notifying the broken path to the management platform. GMPLS should provide the possibility to detect the failure only from the originating node of that LSP. Manual checking of each LSR on the path is not desirable.
- measurement of SLAs
  Each service realized in the network results in a contract (SLA) between the user of the network and the network provider. It is important to examine whether all the parameters specified in SLAs are satisfied

by underlying network. Although various methods of the measurements may be considered, there should be a minimal impact on the network resources.

However, in Grid-enabled GMPLS Control Plane it is important to extend the management functionalities with Grid resources monitoring capability. It should be possible to integrate Grid monitoring services with G2MPLS Control Plane in that way, that extended SLAs are measured and monitored (by the means of Grid resources, like storage, the CPUs, etc.). Each failure, network- as well as grid-related, should be notified to the management platform and appropriate action should be taken.

6.7 Traffic grooming and shaping

The User-to-Network reference point could be considered the network entry gate, through which user data traffic is mapped into the internal network transmission entity (e.g. a VLAN, a timeslot, a lambda, etc.). Classification and aggregation of user traffic might be implemented at the UNI under supervision of the G^2MPLS Control Plane in order to match the Service Level Agreement negotiated through signalling.

The problem of traffic grooming and shaping at the UNI could be extended to any network boundary in which a technology transition and a possible increase of bandwidth of the bearing transport service could occur.

The problem of traffic grooming and shaping in G^2MPLS architecture is not a primary objective of the PHOSPHORUS network and is left for further studies.

7 G^{2}MPLS Use Cases

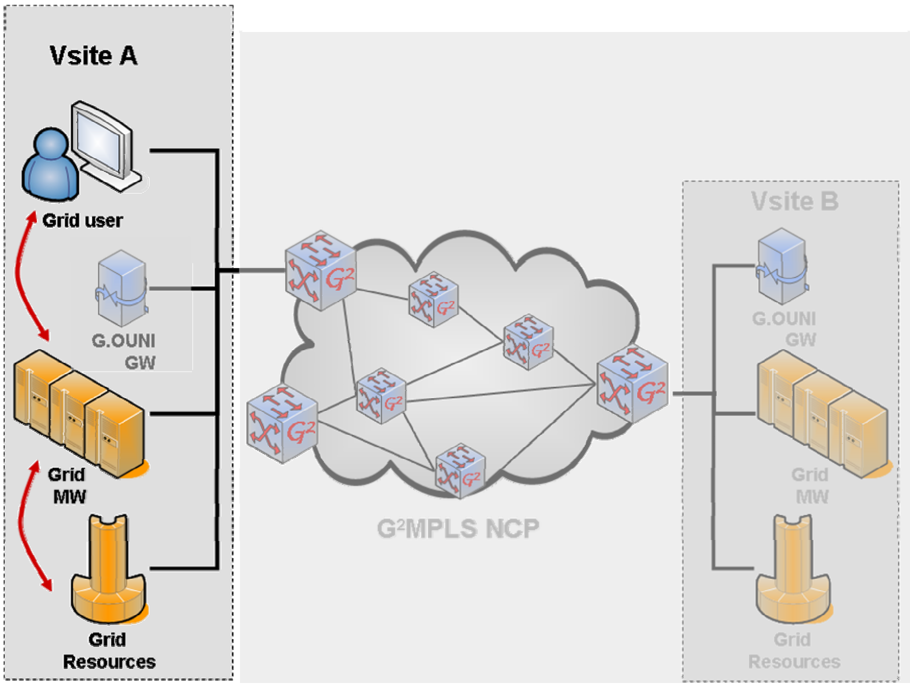
In this chapter some use-cases for the G^{2}MPLS architecture are provided with the aim of identifying the roles and requirements for some system components such as the G.OUNI gateway and the Grid middleware. Further use-cases can be obtained by the compositions or variation of these basic ones.

Those use-cases involving local and remote Grid resources (ref. following sections) are described under the assumption of just one call in one GNS transaction. This does not compromise the scope and general applicability of the architecture, since more complex use-cases can be obtained by the composition of these basic ones.

Each use-case is compliant with a common template describing:

- *Use case identity*
    - **Number**, a sequential reference number,
    - **Name**, a short reference name for its identification,
    - **Goal**, the functionality or behaviour that is expected to be provided by the system once the use-case is executed,
    - **Actors**, the individuals and their means that trigger system reactions
    - **System components**, the system entities involved in the execution of the use-case
    - **Preconditions**, those actions that must occur before the execution of the use-case in order to obtain the described behaviour
    - **Postconditions**, those conditions that could occur once the use-case has been executed and the system continues its operations
    - **Trigger**, the main action that starts the use-case execution.
- *Use case steps*
    - An ordered and hierarchical sequence of events that describe the use case execution.

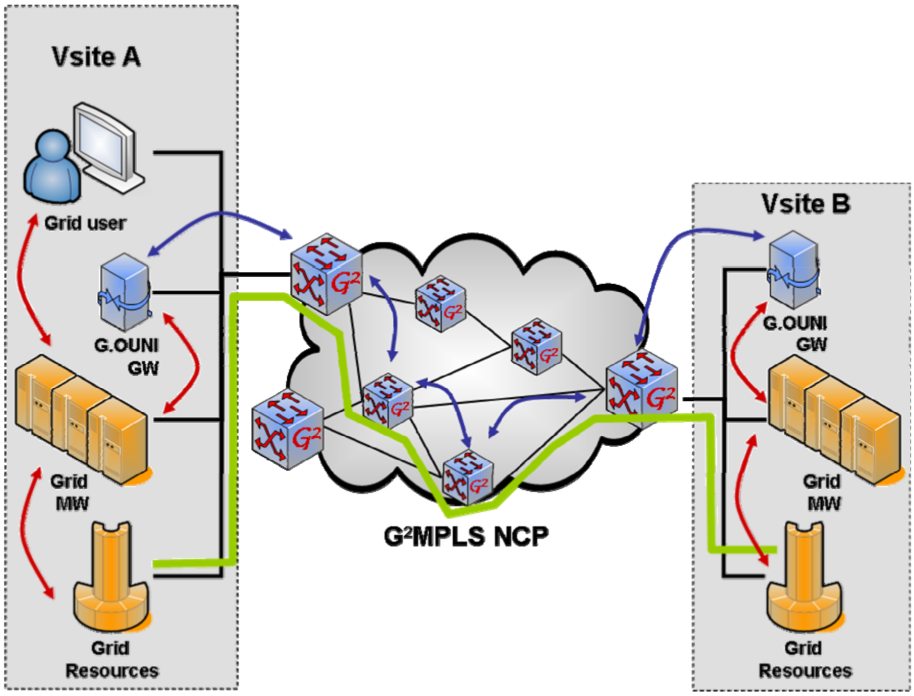
7.1 Loopback GNS transaction

Use case	
Number	UC#1
Name	Loopback GNS transaction
Goal	A Grid User configures and requests the execution of a job that involves Grid resources located only at his local Vsite
Actors	• Grid User at Vsite A
System components	• Vsite A o Client Application o Grid middleware o Local Grid resources
Preconditions	Grid Middleware (local Grid scheduler in particular) is aware of the types and amount of Grid resources located at its Vsite A.
Postconditions	Grid Middleware (local Grid scheduler in particular) advertise the new status/availability of the resources under their control.
Trigger	The Grid User defines the job in the Client Application and issues the request.
Steps	
	
1.	Client Application sends request to local Middleware at Vsite A.
2.	The Grid broker in the middleware resolves the local availability of Grid resources needed for the job
3.	The Grid broker schedules and configures the local Grid resources via LRMS.
4.	The middleware sends job request confirmation to the Client Application.
5.	Client Application notifies the Grid User of the job request confirmation.

6. [optional for advance reservation]	The middleware sends job execution notification to the Client Application.
7. [optional for advance reservation]	Client Application notifies the Grid User of the job execution.
Notes	
No network connection is scheduled on NCP. No interaction is needed with the NCP.	

7.2 GNS transaction – direct invocation

Use case	
Number	UC#2
Name	GNS transaction – Direct Invocation
Goal	A Grid User configures and requests the execution of a job that involves Grid resources located at his local Vsite A and at Vsite B.
Actors	• Grid User at Vsite A
System components	• Vsite A o Client Application o Grid middleware o Local Grid resources o G.OUNI gateway • Vsite B o Grid middleware o Local Grid resources o G.OUNI gateway • G^2MPLS network Control Plane o G^2.LERs with G.OUNI-N functionality o G^2.LSRs
Preconditions	Grid Middlewares at Vsite A and Vsite B (local Grid scheduler in particular) are aware of the types and amount of Grid resources located at their own site (directly) and at remote sites (via G.OUNI service and resource discovery).
Postconditions	Grid Middlewares (local Grid scheduler in particular) advertise the new status/availability of the resources under their control.
Trigger	The Grid User defines the job in the Client Application and issues the request.
Steps	

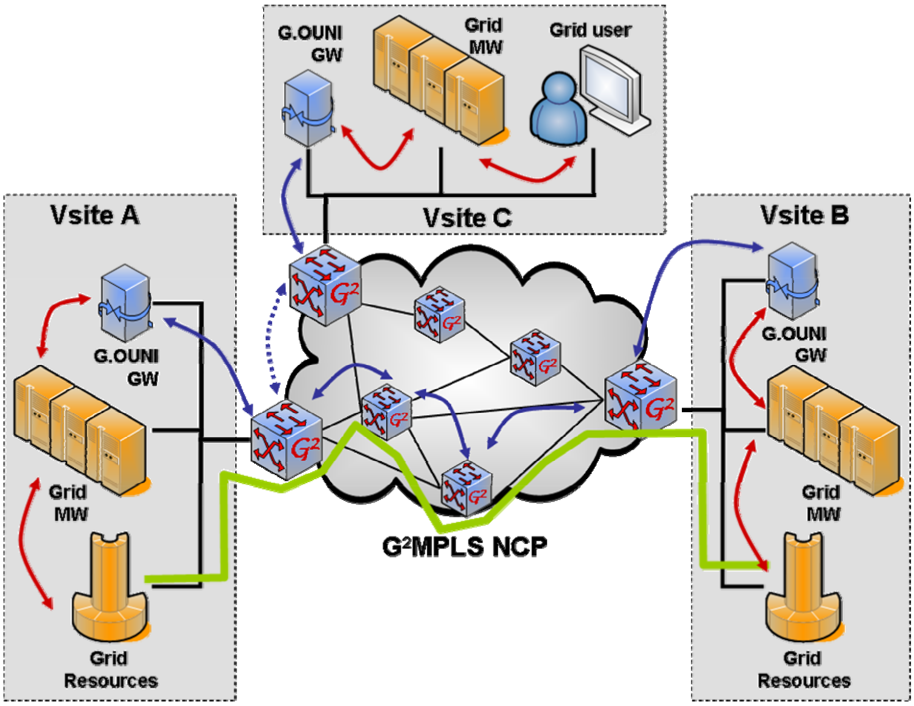


1.	Client Application sends request to local Middleware at Vsite A.
2.	The Grid broker in the middleware resolves the Grid resources needed for the job (local and remote).
3.	The Grid broker requests the local G.OUNI gateway to build a GNS transaction between this Vsite (i.e. the local scheduled resources) and the remote Vsite B, in case with explicit indication of the ingress/egress network attachment points (it depends on the use and details of the network routing information on the middleware).
4.	The G.OUNI gateway schedules and configures the local Grid resources via LRMS on Vsite A.
5.	The G.OUNI gateway sends a GNS transaction request to the G.OUNI-N on the peering G^2.LER.
6.	The GNS transaction request is sent through the G^2.LSRs towards the selected egress G^2.LER.
7.	The egress G^2.LER sends the GNS transaction request to the peering G.OUNI gateway.
8.	The G.OUNI gateway on site B requests its LRMS in the middleware to schedule and configure the local Grid resources.
9.	The LRMS at Vsite B schedules and configures the local Grid resources.
10.	The LRMS at Vsite B sends job request confirmation to the G.OUNI gateway.
11.	The G.OUNI gateway sends a GNS transaction response to the G.OUNI-N on the peering egress G^2.LER.
12.	The GNS transaction response is sent back through the G^2.LSRs towards the selected ingress G^2.LER.
13.	The ingress G^2.LER sends a GNS transaction response to the G.OUNI gateway.

14. [optional 3-tier signalling]	GNS transaction confirm signalling tier among ingress G.OUNI gateway, G^2MPLS NCP and egress G.OUNI gateway.
15.	The GNS transaction response is sent back by the G.OUNI gateway at Vsite A to the Grid broker.
16.	The middleware sends job request confirmation to the Client Application.
17.	Client Application notifies the Grid User of the job request confirmation.
18. [optional for advance reservation]	The G^2MPLS advance reservation manager issues connection activation (despite the distributed or centralized implementation of the module).
19. [optional for advance reservation]	The G^2MPLS NCP notifies of the G.OUNI gateways of the connection activation.
20. [optional for advance reservation]	The G.OUNI gateways notify the middleware of the GNS transaction activation.
21. [optional for advance reservation]	The middleware sends job execution notification to the Client Application.
22. [optional for advance reservation]	Client Application notifies the Grid User of the job execution.
Notes	

7.3 GNS transaction – indirect invocation

Use case	
Number	UC#3
Name	GNS transaction – Indirect Invocation
Goal	A Grid User at Vsite C configures and requests the execution of a job that involves Grid resources located remotely at Vsite A and Vsite B.
Actors	• Grid User at Vsite C
System components	• Vsite A o Grid middleware o Local Grid resources o G.OUNI gateway • Vsite B o Grid middleware o Local Grid resources o G.OUNI gateway • Vsite C o Client Application o Grid middleware o G.OUNI gateway • G^2MPLS network Control Plane o G^2.LERs with G.OUNI-N functionality o G^2.LSRs
Preconditions	Grid Middlewares at Vsite A, Vsite B and Vsite C (local Grid scheduler in particular) are aware of the types and amount of Grid resources located at their own site (directly) and at remote sites (via G.OUNI service and resource discovery).
Postconditions	Grid Middlewares (local Grid scheduler in particular) advertise the new status/availability of the resources under their control.

Trigger	The Grid User defines the job in the Client Application and issues the request.
Steps	
	
1.	Client Application sends request to local Middleware at Vsite C.
2.	The Grid broker in the middleware resolves the Grid resources needed for the job (remote).
3.	The Grid broker requests the local G.OUNI gateway to build a GNS transaction between Vsite B and Vsite C, in case with explicit indication of the ingress/egress network attachment points (it depends on the use and details of the network routing information on the middleware).
4.	The G.OUNI gateway sends a GNS transaction request to the G.OUNI-N (acting as a proxy) on the peering G^2.LER.
5.	The G^2.LER peering with Vsite C sends a remote GNS transaction request to the selected G^2.LER peering with Vsite A. This is a G^2MPLS ISI (Internal Signalling Interface).
6.	The G^2.LER peering with Vsite A sends the GNS transaction request to the peering G.OUNI gateway.
7.	The G.OUNI gateway at Vsite A requests the LRMS in the middleware to schedule and configure the local Grid resources for Vsite A.
8.	The LRMS at Vsite A schedules and configures the local Grid resources.
9.	The LRMS at Vsite A sends job request confirmation to the G.OUNI gateway.
10.	The G.OUNI gateway sends a GNS transaction response to the G.OUNI-N on the peering G^2.LER.
11.	The GNS transaction request is sent through the G^2.LSRs towards the selected egress G^2.LER.
12.	The egress G^2.LER sends the GNS transaction request to the peering G.OUNI gateway.

13.	The G.OUNI gateway requests the LRMS in the middleware to schedule and configure the local Grid resources for Vsite B.
14.	The LRMS at Vsite B schedules and configures the local Grid resources.
15.	The LRMS at Vsite B sends job request confirmation to the G.OUNI gateway.
16.	The G.OUNI gateway sends a GNS transaction response to the G.OUNI-N on the peering egress G^2.LER.
17.	The GNS transaction response is sent back through the G^2.LSRs towards the selected ingress G^2.LER (that peering with Vsite A).
18.	The G^2.LER peering with Vsite A sends a remote GNS transaction response to the G^2.LER peering with Vsite C. This is a G^2MPLS ISI (Internal Signalling Interface).
19.	The G^2.LER peering with Vsite A sends a GNS transaction response to the G.OUNI gateway at Vsite A.
20. [optional 3-tier signalling]	GNS transaction confirm signalling tier among involved G.OUNI gateways and G^2MPLS NCP.
21.	The GNS transaction response is sent back by the G.OUNI gateway at Vsite C to the local Grid scheduler.
22.	The middleware sends job request confirmation to the Client Application.
23.	Client Application notifies the Grid User of the job request confirmation.
24. [optional for advance reservation]	The G^2MPLS advance reservation manager issues connection activation (despite the distributed or centralized implementation of the module).
25. [optional for advance reservation]	The G^2MPLS NCP notifies of the G.OUNI gateways of the connection activation.
26. [optional for advance reservation]	The G.OUNI gateways notify the middleware of the GNS transaction activation.
27. [optional for advance reservation]	The middleware sends job execution notification to the Client Application.
28. [optional for advance reservation]	Client Application notifies the Grid User of the job execution.
Notes	

8 G^{2}MPLS Network Reference Points and Interfaces

In this chapter the G^{2}MPLS network reference points are identified and abstract messaging of the related interfaces is defined.

8.1 Network reference points

The identification of the network reference point in the G^{2}MPLS NCP is inspired by the ASON/GMPLS architecture, which is the best match of the Control Plane requirements declared in the previous sections.

In the G^{2}MPLS NCP five network reference points are identified:

- the Grid-capable Optical User-Network Interface (G.OUNI)
- the Grid-capable External Network-Network Interface (G.E-NNI)
- the Grid-capable Internal Network-Network (or Node-Node) Interface (G.I-NNI)
- the southbound interface (SBI)
- the northbound interfaces (NBI).

In Figure 8-1 all these reference points are represented in an example scenario including two neighbouring G^{2}MPLS domains, interfaced through G.OUNI gateways to the Grid middleware and application respectively. The advance reservation service is provided in one of these domains, i.e. domain on the left in Figure 8-1, through a centralized entity (e.g. a Path Computation Element as per IETF RFC 4655); in the latter case, i.e. the domain on the right in Figure 8-1, this service is distributed on the different NCP instances by means of a pre-planned setup of connections done at request time with a further activation tier at service time (which relies on crankback for optimization mechanism of blocked source routes).

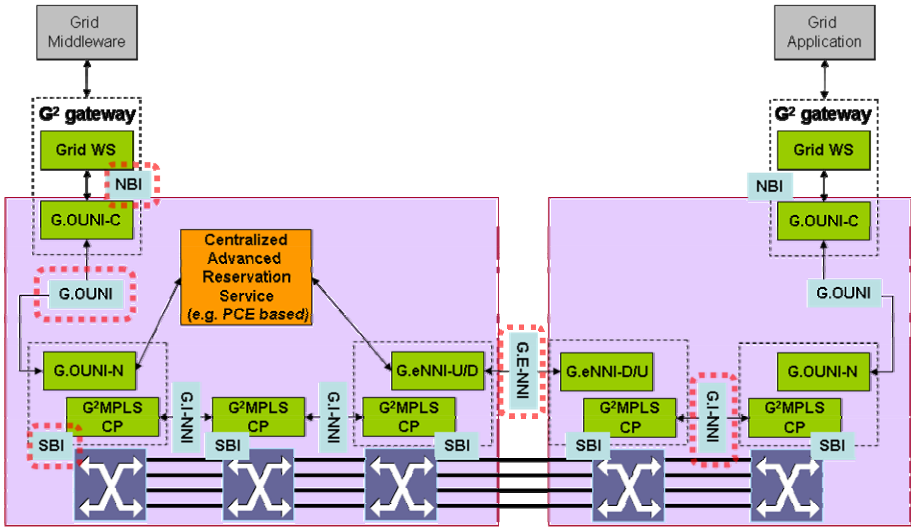


Figure 8-1: G^2MPLS network reference points.

8.2 G^2MPLS interfaces and related functionalities

8.2.1 East-westbound interfaces

8.2.1.1 *G.OUNI*

Grid Optical User Network Interface (G.OUNI) comprises a number of procedures to facilitate on demand as well as in-advance access to Grid services over G^2MPLS. G.OUNI is conceived also to interoperate with current GMPLS transport network in a limited downgraded configuration, by acting as a standard O-UNI. Interoperable procedures between Grid users/resources (e.g. storage, processor, memory) and optical network for agreement negotiation and Grid service activation have to be developed.

The G.OUNI reference point acts as the interface between the client Grid Vsite and the G^2MPLS domain. In general, G.OUNI interconnects the Grid middleware to G^2MPLS, but in a more visionary scenario G.OUNI could also connect directly Grid users/applications and Grid resources to G^2MPLS. In the first case, Grid user/application and Grid resources are mediated by the Grid middleware towards the G^2MPLS network, and, thus the middleware preserves many of its specific functionalities. In the latter case, i.e. Grid endpoints directly connected to the G.OUNI, G^2MPLS should support services/protocols that are currently established between Grid endpoints and Grid middleware. For example, JSDL documents can then be directly sent by the user towards the network through G.OUNI, and WSRF procedures. Job manager, job policy, job lifetime management should be part of G^2MPLS. The following table represents the services standardised by OGF, OASIS and W3C that could possibly be supported by G.OUNI.

Grid User ↔ G.OUNI	Grid Resource ↔ G.OUNI	Grid Middleware ↔ G.OUNI
• JSDL document	• WSRF • GLUE Schema	• WS-Agreement

Table 8-1: Grid services supported by G.OUNI.

The number of procedures supported by G.OUNI depends on the level of Grid network integration and number of Grid services supported by G^2MPLS. The G^2MPLS layering model determines the G.OUNI functionalities to be used.

The G.OUNI interface should be able to support the following:

- Service invocation
    - Direct
    - Indirect
- Neighbour Discovery (Grid and network scoped)
    - Extensions to current OIF UNI v2.0 to support Grid Resource Discovery, i.e. Grid resources availability.
- Service Discovery(Grid and network scoped)
    - Extensions to current OIF UNI v2.0 to support Grid Service Discovery, i.e. Grid resources capability.
- Grid Network Service creation/deletion
    - Support of flexible bandwidth allocation
    - Support of network constraints (e.g. latency, jitter)
    - Support of job/service request
        - e.g. implementation could be based on RSVP-TE extensions to support job/service request.
    - Support of advance reservations
        - Inclusion of service time schedules in the signalling messages
    - Support of Grid resource availability
        - Configurable permeability of Grid and Network resource availability as derived by routing instances.
    - Extension to support internal Grid service/resource failures (e.g. cluster failures) for job recovery purposes
- AAA
    - Extensions to integrate Grid AAA and Security means with the network related means.

The G.OUNI interface has knowledge of the JSDL semantic [GFD.56] and implements its functionalities through the following abstract messaging classes (ref. Figure 8-2 and Table 8-2).

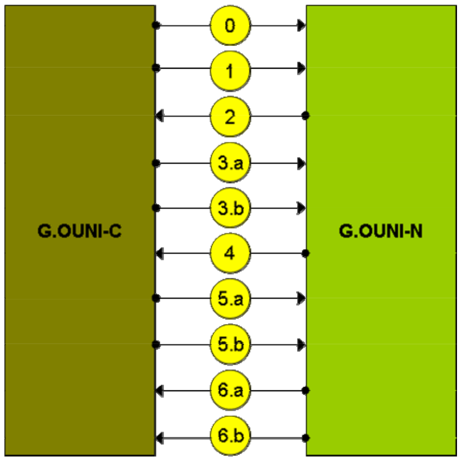


Figure 8-2: G.OUNI abstract messaging classes.

Abstract Messaging Class	Name	Description
0	Grid service discovery	Discovery of Grid capabilities (e.g. CPU types, OS types, etc.)
1	Local Grid resource discovery	Feed-down of Grid resource availability (e.g. how much CPU types, storage, etc.)
2	Remote Grid and Network resource discovery	Grid & network resource availability feed-up (e.g. how much remote CPU types, storage, Vsites attachment points, etc.). The detail of network information depends on the capabilities of the Grid fabric (MW + Application) to specify explicitly the network ingress/egress points or not, i.e. it depends on the permeability level request during the Grid Service Discovery phase.
3.a	NS request	Just network scope requests. These requests are issued in case of explicit specification of the network attachment points. Communications between the MW instances on the remote Grid sites is provided at the MW layer with traditional means. They could be just bridged by G^2MPLS, e.g. through a dedicated Grid signalling path.
3.b	GNS request	Grid and network scoped request. These requests are issued in case of implicit specification of the network attachment points and are possible in the full-fledged G^2MPLS operational framework.
4	Grid resource allocation request	Grid allocation request forwarded from the network to the remote Vsite.
5.a	Network Service status	Status enquiry for a given Network Service
5.b	Grid Network Service status	Status enquiry for a given Grid Network Service

Abstract Messaging Class	Name	Description
6.a	Network Service notify	Notification message sent autonomously by UNI-N to UNI-C to indicate a change in the status of the Network Service (e.g., un-restorable connection failure).
6.b	Grid Network Service notify	Notification message sent autonomously by UNI-N to UNI-C to indicate a change in the status of the Grid Network Service (e.g., un-restorable, connection failure, remote Grid resource failure, etc.).

Table 8-2: G.OUNI abstract messaging description.

In addition to the messages described in Table 8-2, all Grid middleware information and messages described for northbound interface in section 8.2.3 might be supported by G.OUNI.

Some examples of signalling flows for the different messaging classes are shown in Figure 8-4, Figure 8-5, Figure 8-6 and Figure 8-7, including the Centralized Advanced Reservation Service where needed. In Figure 8-4 and Figure 8-6 the ack-ing mechanism is not depicted for the sake of clarity: it is mandatory for the hard signalling mode (i.e. without continuous refreshes of the reservations as per standard RSVP IETF 2205), optional in case of soft signalling mode. Moreover, in these pictures the inner G^2MPLS messaging is provided as well, derived by the G.OUNI classes. ISI in Figure 8-6 stands for Internal Signalling Interface and its specification is out of the scope of this document.

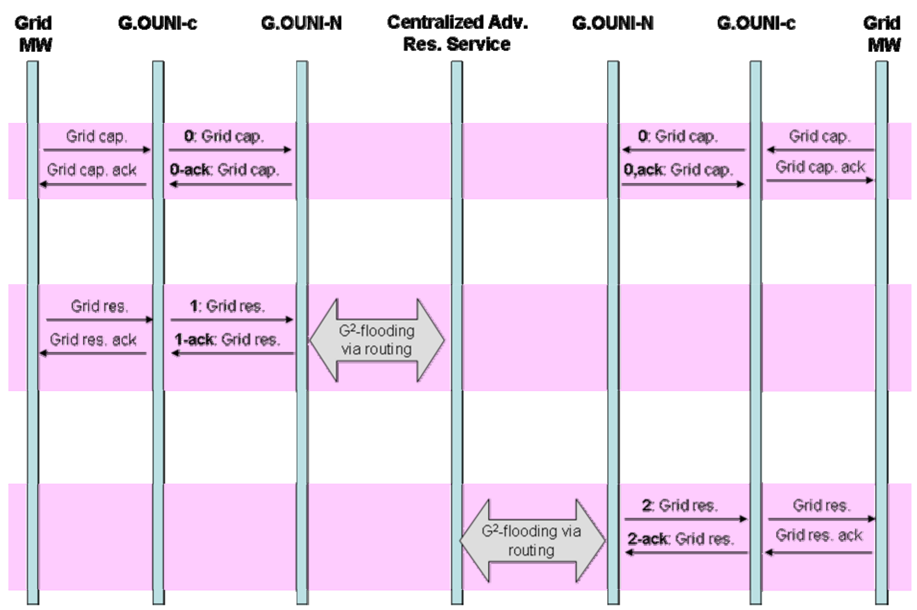


Figure 8-3: Abstract messaging for Grid service and resources discovery at G.OUNI.

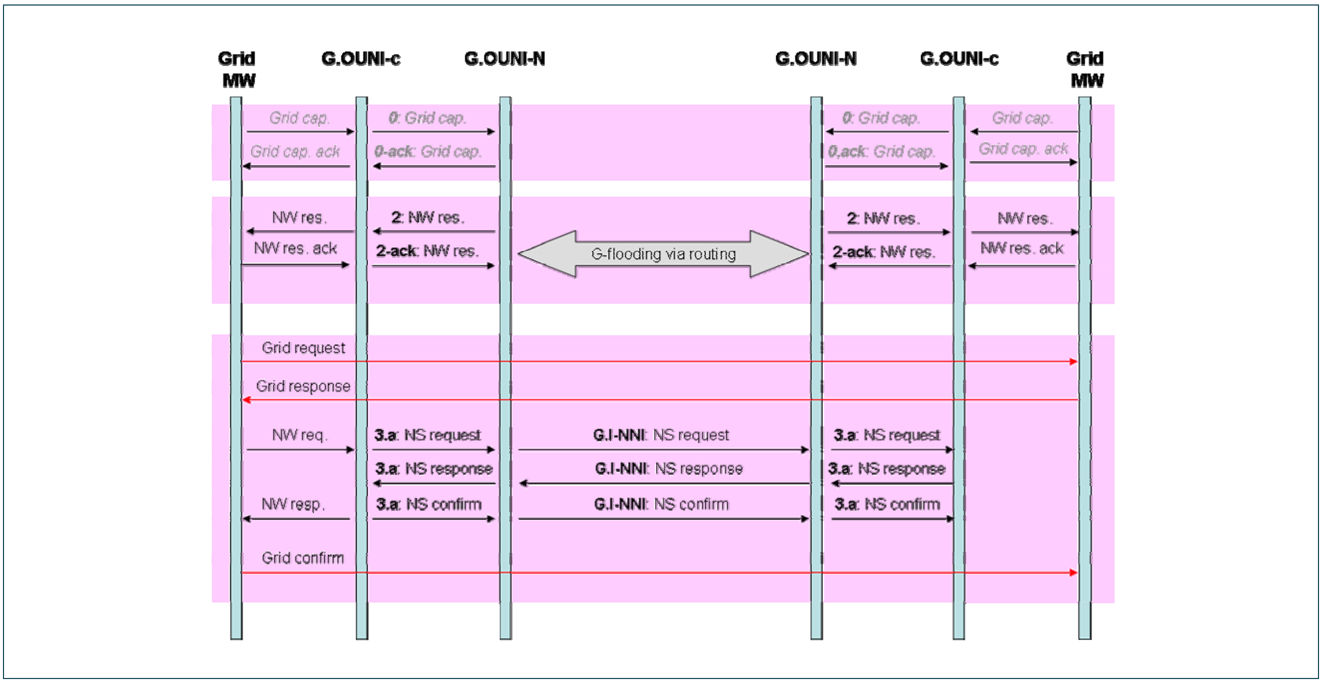


Figure 8-4: Abstract messaging for a NS request at G.OUNI.

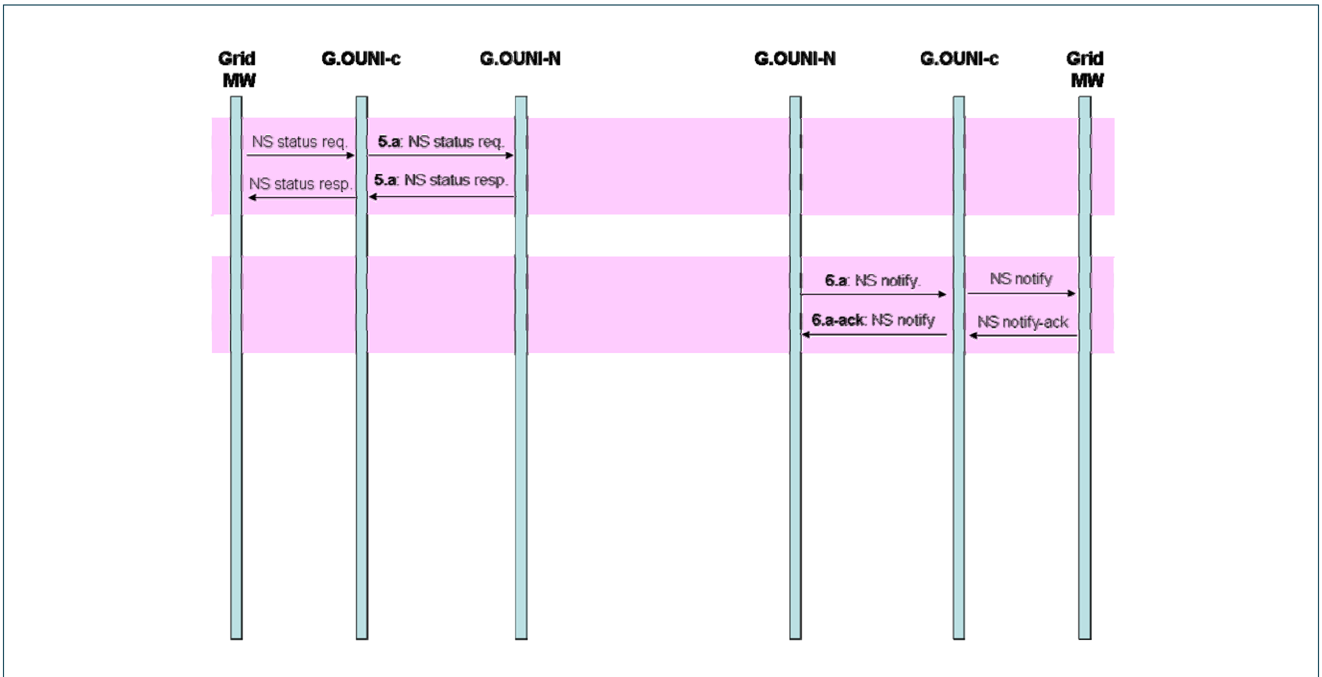


Figure 8-5: Abstract messaging for a NS status and notify at G.OUNI.

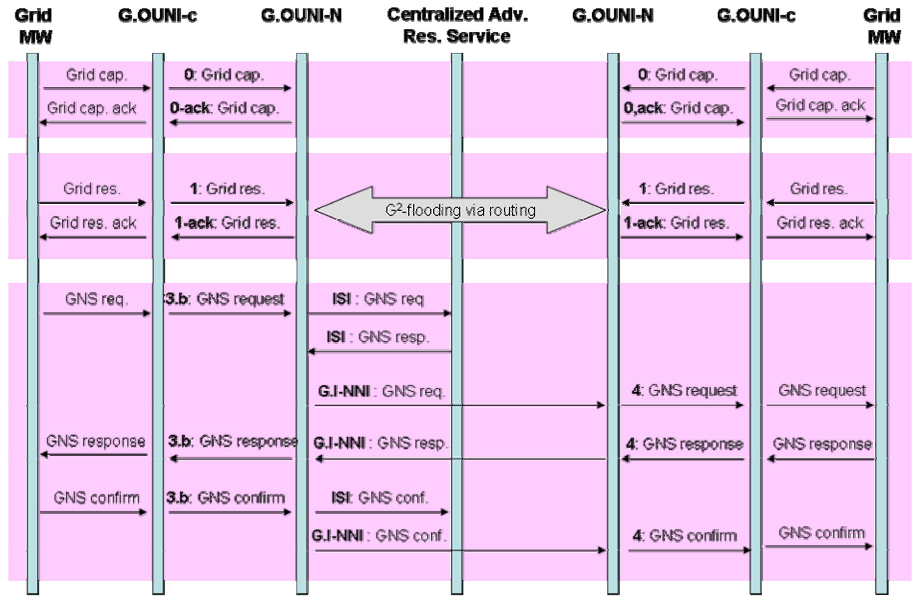


Figure 8-6: Abstract messaging for a GNS request at G.OUNI.

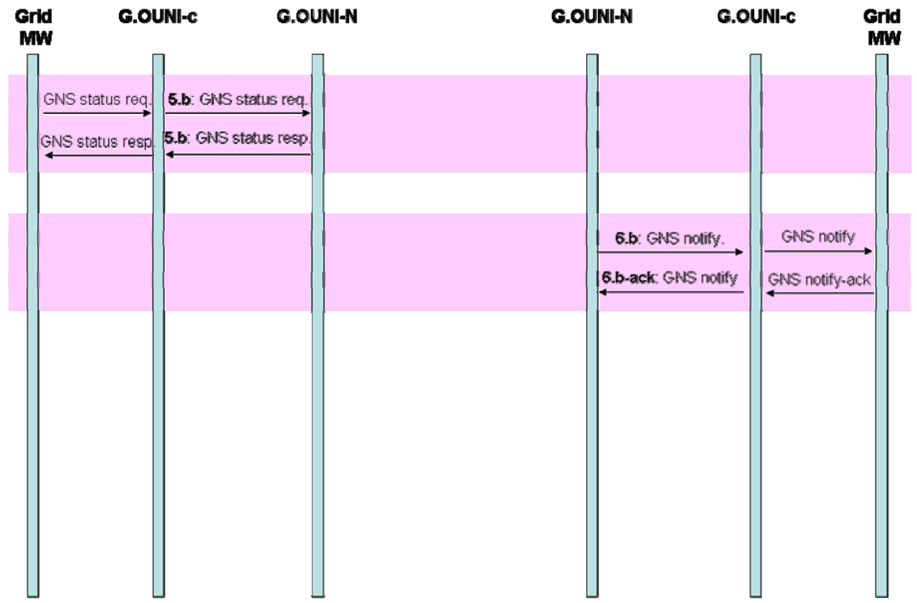


Figure 8-7: Abstract messaging for a GNS status and notify at G.OUNI.

8.2.1.2 *G.I-NNI*

The G.I-UNI interface has knowledge of the GNS semantic and implements its functionalities through the following abstract messaging classes (ref. Table 8-3 and Figure 8-8).

Abstract Messaging Class	Name	Description
11	Grid & network resource announcement	Flooding of local and learned Grid resource availabilities (e.g. how much CPU types, storage in Vsite x, etc.) and network resources (TE links, etc.)
12	Grid & network resource discovery	Flooding of local and learned Grid resource availabilities (e.g. how much CPU types, storage in Vsite x, etc.) and network resources (TE links, etc.).
13.a	NS request	Just network call/connections requests. Issued in case of explicit specification of the network attachment points.
13.b	GNS request	Grid + network requests. Issued in case of implicit specification of the network attachment points.
15	Network Service status	Status enquiry for a given network connection.
16	Network Service notify	Notification message sent autonomously to indicate a change in the status of a given network connection (e.g., un-restorable connection failure).

Table 8-3: G.I-NNI abstract messaging description.

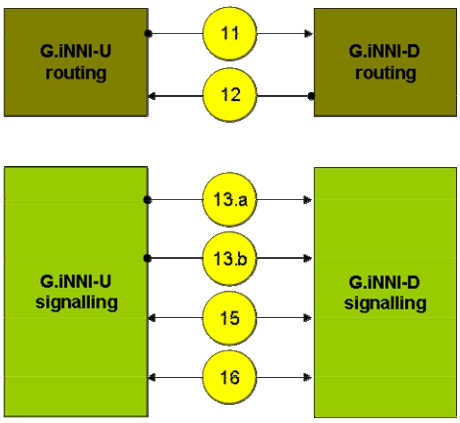


Figure 8-8: G.I-NNI abstract messaging classes.

The deriving signalling flows for the different messaging classes are shown in Figure 8-9, Figure 8-10 and Figure 8-11, by skipping the ack-ing mechanism in Figure 8-10 and Figure 8-11 for the sake of clarity.

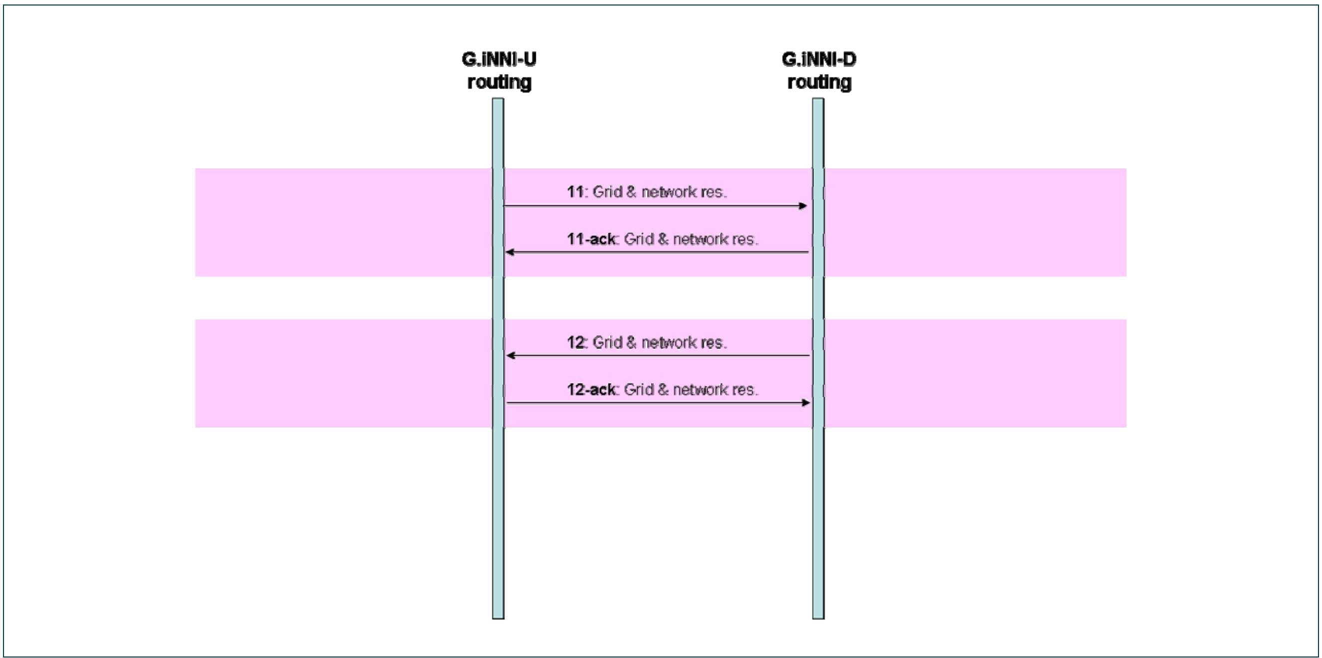


Figure 8-9: Abstract messaging for Grid and network resource discovery at G.I-NNI.

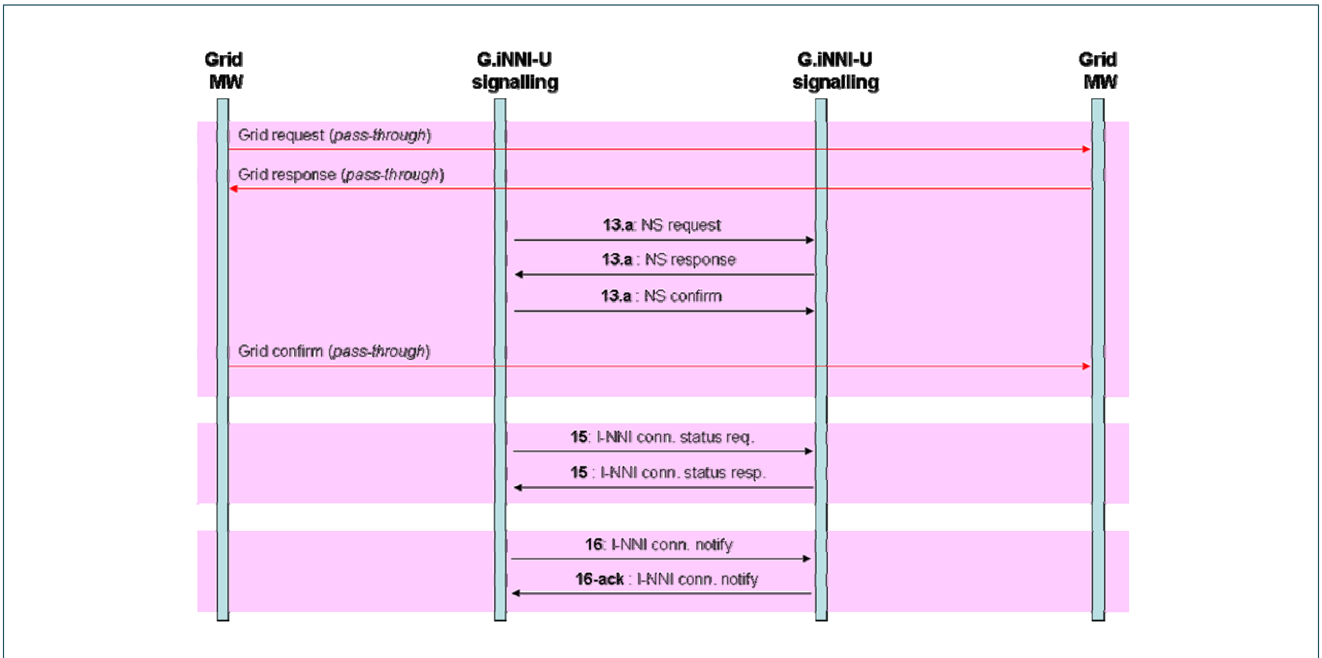


Figure 8-10: Abstract messaging for NS request and connection status/notify at G.I-NNI.

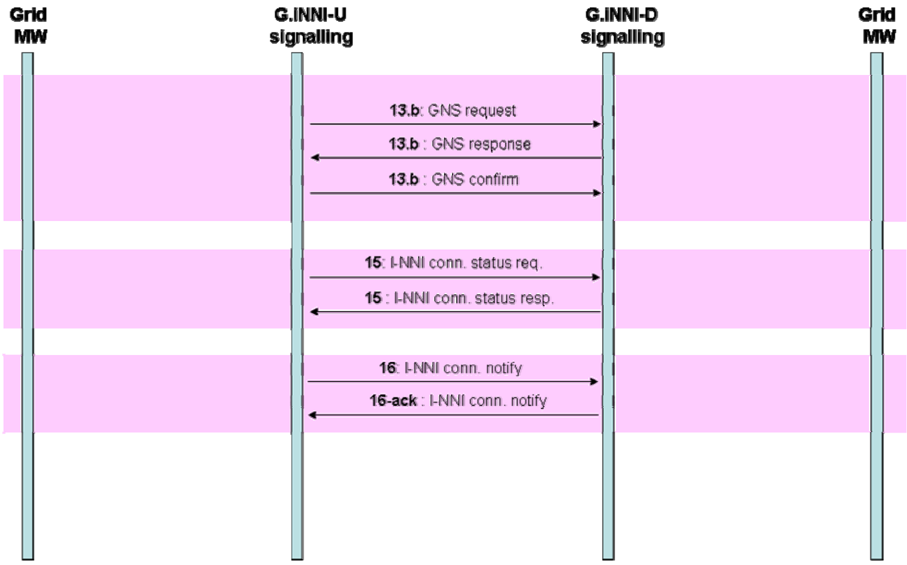


Figure 8-11: Abstract messaging for GNS request and connection status/notify at G.I-NNI.

8.2.1.3 *G.E-NNI*

The G.E-UNI interface has knowledge of the GNS semantic and implements its functionalities through the following abstract messaging classes (ref. Figure 8-12). Just one layer of hierarchy is considered for the inter-domain routing.

Abstract Messaging Class	Name	Description
101	Grid & network resource announcement	Inter-domain flooding of local + learned Grid resource availabilities (e.g. how much CPU types, storage in Vsite x, etc.) and network resources (TE links, etc.)
102	Grid & network resource discovery	Inter-domain flooding of local + learned Grid resource availabilities (e.g. how much CPU types, storage in Vsite x, etc.) and network resources (TE links, etc.).
103.a	NS request	Just network E-NNI call/connections requests. Issued in case of explicit specification of the network attachment points.
103.b	GNS request	Grid + network E-NNI requests. Issued in case of implicit specification of the network attachment points.
105	Network Service status	Status enquiry for a given E-NNI network connection.

Abstract Messaging Class	Name	Description
106	Network Service notify	Notification message sent autonomously to indicate a change in the status of a given E-NNI network connection (e.g., un-restorable connection failure).

Table 8-4: G.E-NNI abstract messaging description.

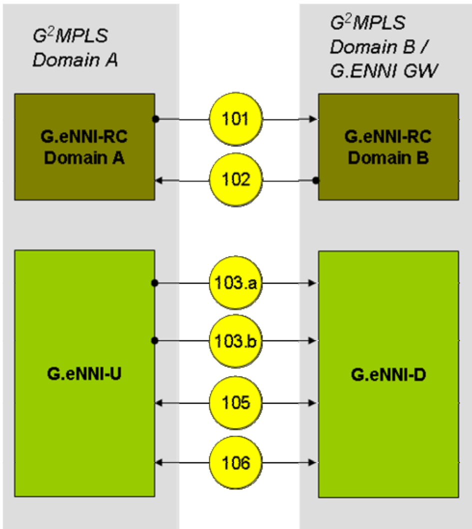


Figure 8-12: G.E-NNI abstract messaging classes.

The deriving signalling flows for the different messaging classes are shown in Figure 8-13, Figure 8-14 and Figure 8-15, by skipping the ack-ing mechanism in NS and GNS requests for the sake of clarity.

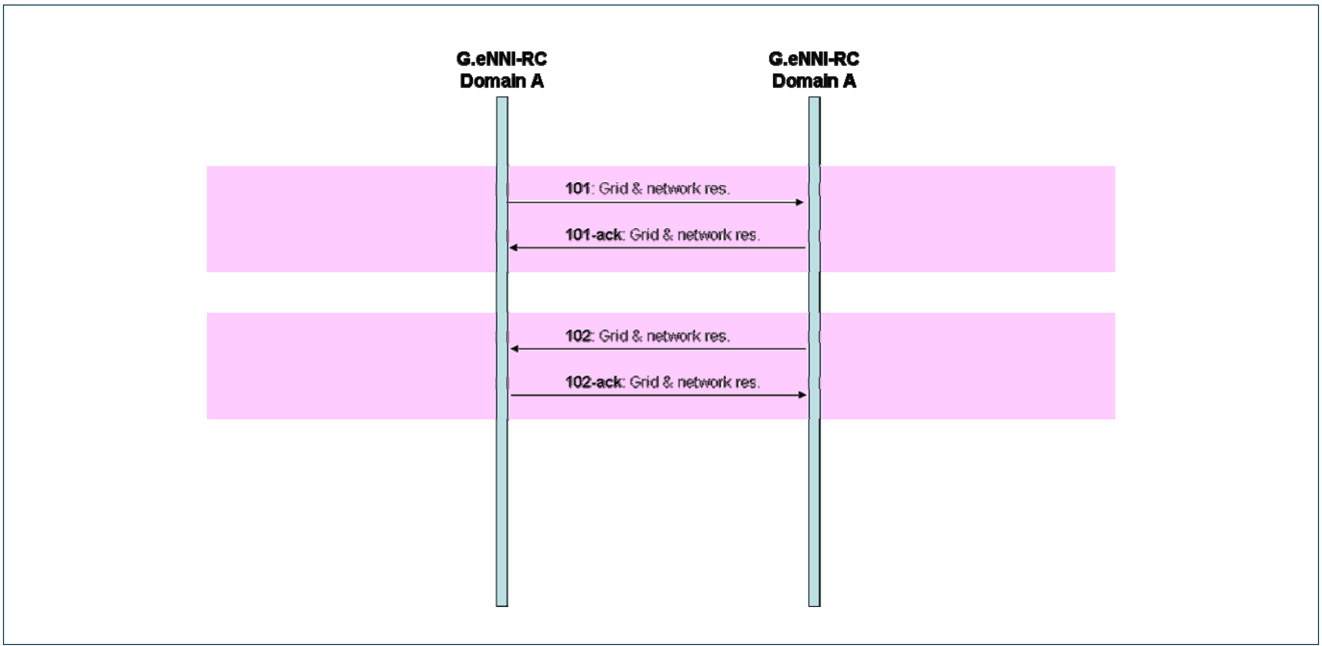


Figure 8-13: Abstract messaging for Grid and network resource discovery at G.E-NNI.

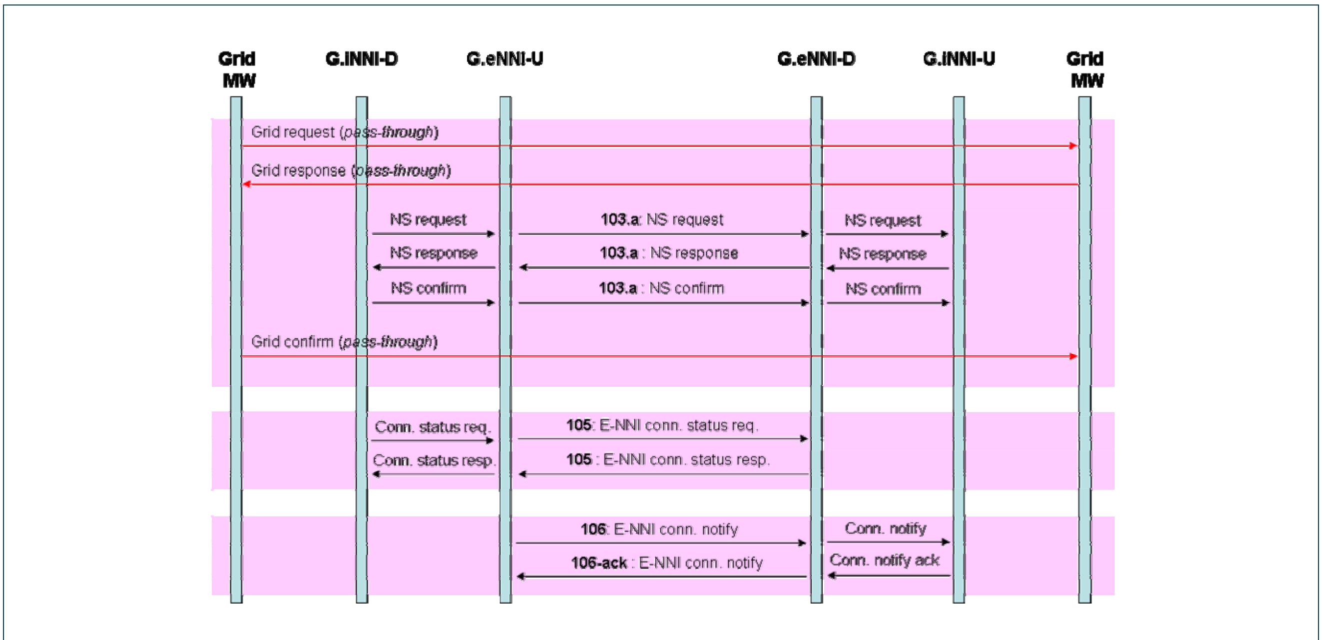


Figure 8-14: Abstract messaging for NS request and connection status/notify at G.E-NNI.

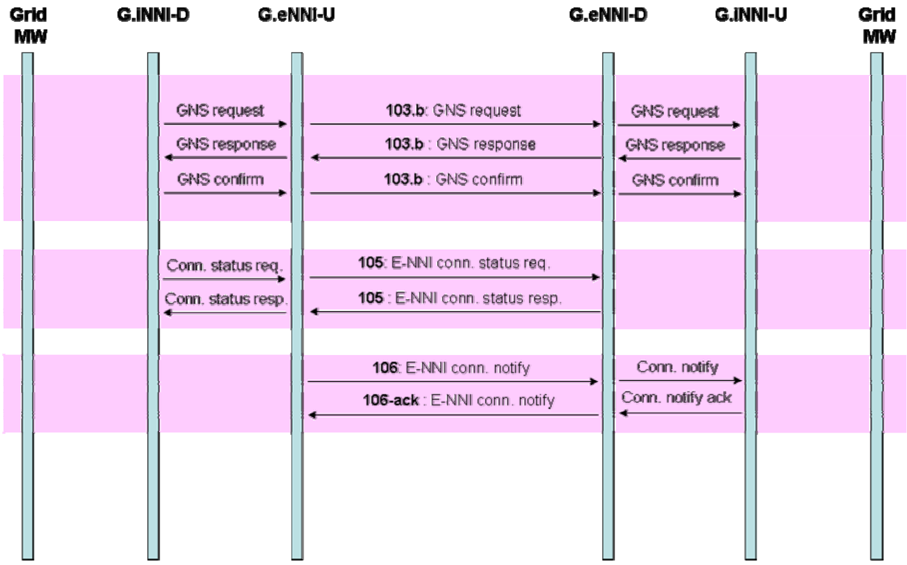


Figure 8-15: Abstract messaging for GNS request and connection status/notify at G.E-NNI.

8.2.2 Southbound interface (SBI)

The SBI interface allows the semantic translation of Control Plane actions into specific actions to be carried out by underlying Transport Network Equipment. Depending on the interface exported by the equipment (e.g. TL1, SNMP, CLI, etc.), the SBI syntax adapts and implements the abstract messaging classes here described (ref. Table 8-5 and Figure 8-16). The SBI interface is conceived to have a client side on the Control Plane and a server side on the management agent of the Transport Network Equipment. Most of the transactions are initiated by the client towards the server. Just the notification is asynchronously initiated by the server upon the occurrence of an event in the set of monitored resources (e.g. link down/up, etc.).

Abstract Messaging Class	Name	Description
1001	Transport Network resource discovery	Retrieving of transport resource availabilities (e.g. how many ports and signals are available on the TNE).
1003	Transport Network resource configuration request	Request for cross-connecting resource corresponding to label X to resource corresponding to label Y. Depending of the technology these labels identify lambdas, timeslots, VLANs, etc.
1005	Transport Network resource status	Status enquiry for a given Transport Network resource, i.e. a lambda, a timeslots, VLANs, etc. Status is expected to be returned as free, busy, and locked.

Abstract Messaging Class	Name	Description
1006	Transport Network resource notify	Notification message sent autonomously by the server to indicate a change in the status of a given Transport Network resource (e.g. a connection or signal failure).

Table 8-5: SBI abstract messaging description.

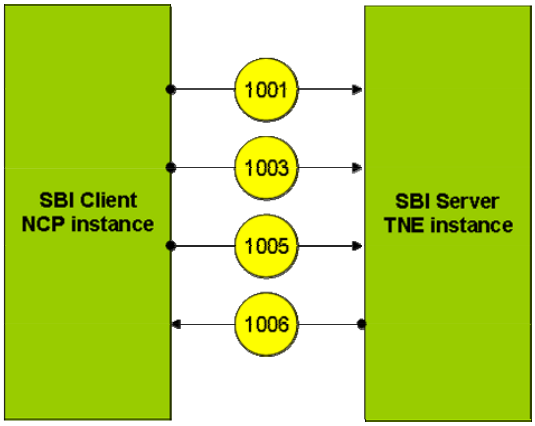


Figure 8-16: SBI abstract messaging classes.

The deriving signalling flows for the different messaging classes are shown in Figure 8-17.

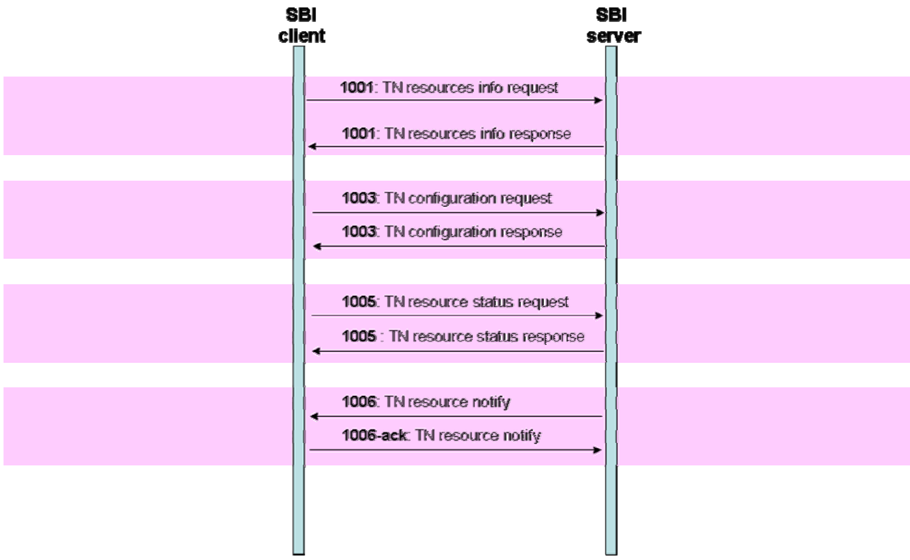


Figure 8-17: Abstract messaging for resource discovery, configuration and notification at SBI.

8.2.3 Northbound interfaces (NBI)

G^{2}MPLS exposes two kind of northbound interfaces: one towards the Grid layer and one towards the Network Service Plane (including NRPS for the purposes of this discussion).

8.2.3.1 *Grid Northbound interface (G.NBI)*

Services supported by NBI towards the Grid layer are under the root of Execution Management Services, which is aimed to find execution candidate locations and selecting execution location. They comprise:

- Resources indexing (processing, storage, executables, resource management and provisioning)
    - Service Container contains running entities, whether they are jobs or running WS. Containers have resources properties that describe static information (capability) such as what kind of executables they can take (e.g. OS version, libraries installed, policies, and security environment) as well as current load and QoS information
- Resource selection (to decide where to execute a unit of work)
    - Execution Planning Services (EPS)
      EPS is a service to provide a mapping (relation) between services and resources based on schedules. EPS should attempt to optimize some objective function such as execution time, reliability, etc. EPS just generate the schedule which will be enacted by Job Manager (part of Grid MW). EPS will likely use information services and Candidate Set Generators (CSG).
    - Candidate Set Generators

CSG is called by EPS to get a set of resources (capability) and then get updated information on those resources from an information service (availability) and then execute the optimization function to build the schedule. CSG simply determines the set of resources on which a unit of work can execute (where is it possible to execute rather than where will it execute)

- Reservation
  To manage reservations of resources, interact with accounting services, revoke reservations, etc. It is a common interface to all varieties of reservable resources (e.g. computing, network, special purpose instruments) on the Grid.

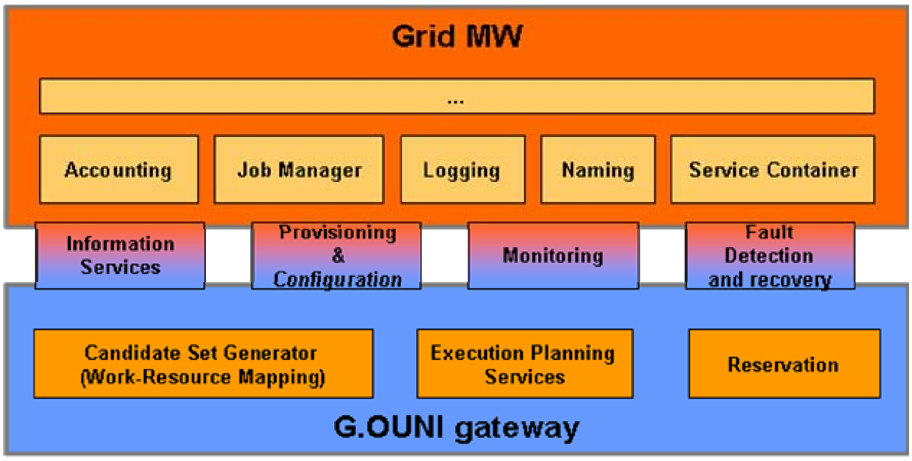


Figure 8-18: Services supported by NBI towards the Grid layer.

The G.NBI interface has knowledge of the Grid semantic and implements its functionalities through the following abstract messaging classes (ref. Table 8-6).

Abstract Messaging Class	Name	Description
A	Job manager call (from Job Manager to EPS)	A legacy Grid job with appropriate job description written in JSDL is being processed. Job Manager calls the EPS to get a schedule.
B	Information service call (from CSG to Information services)	The CSG that determines the set of resources on which a unit of work can execute calls the information services (databases of attribute metadata about resources) to evaluate the suitability of a container (resource) for a job. This is based on binary availability and policy settings.
C	Schedule return (from EPS to Job Manager)	CSG informs EPS and EPS in turn selects the service container. Then it returns the schedule to the JM.

Abstract Messaging Class	Name	Description
D	Resource Reservation request (interaction between JM and Resource)	Resource reservation creation by JM for a group of jobs that are being managed. EPS use reservations to guarantee the execution plan for a particular job. Reservation to all varieties of reservable resources on the Grid.
E	Create reservation	Reservation creation message
F	Revoke reservation	Reservation revocation message
G	Reservation account interaction	in case of charges applied to a specific reservation, reservation service should interact with account services
H	Deployment and Configuration request (between resource service and deployment and configuration service)	Before a service can be used by a unit of work it must be configured (e.g. accessibility of configuration files and executables) or provisioned (e.g. network path establishment)
I	Monitoring interaction (Grid and network monitoring)	Monitoring of applications services for both fault-tolerance reasons and QoS reasons.
L	Fault detection and recovery interaction	Grid and network interaction to allow trading off performance and resource usage.

Table 8-6: G.NBI abstract messaging description.

8.2.3.2 *NSP Northbound interface (N.NBI)*

NRPS and the Network Service Plane (NSP) operate just in the network domain and are used by the Grid layer as a meta-Grid service. The use of the G^2MPLS northbound interface by an NRPS and/or the NSP limits to network only the services exportable by that interface. Therefore, in this context G^2MPLS is used as standard GMPLS Control Plane.

The N.NBI is used only to request/tear down network connections and to retrieve topology and connection status information. It has been defined in the PHOSPHORUS WP1 framework, in the form of a WS-based interface [PHOSPHORUS-D1.1]. This interface is based on a simple request/response model, which abstracts and generalizes transactions between NRPS/NSP and GMPLS.

Abstract Messaging Class	Name	Description
1	Path Creation Service	It provides a basic point-to-point LSP creation between two ports specified by TNA addresses. In case of advance reservation request at the NRPS, the message will include also a path specification in the form of a strict explicit route object (traversed nodes, ports and possibly labels).
2	Path Termination Service	It tears down LSPs previously set up by an NRPS/NSP.
3	Topology Discovery Service	It exports the OSPF-TE LSA database containing all link and node information within the controlled domain.

Abstract Messaging Class	Name	Description
4	Path Discovery Service	It retrieves information about the status (i.e. up, down, recovering) of an established point to point connection in the controlled network.

Table 8-7: N.NBI abstract messaging description.

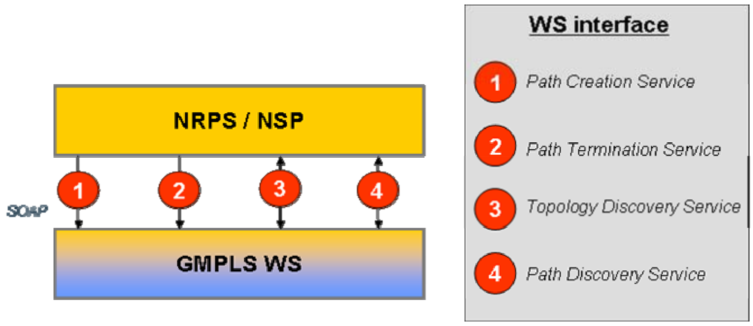


Figure 8-19: G.NBI abstract messaging classes.

Further details on the N.NBI message format can be found in [PHOSPHORUS-D1.1]. Some deployment scenarios for N.NBI are described in the remainder of this document as specific interworking issues between G^2MPLS and NRPS-es (ref. section 12).

9 G^{2}MPLS Functional Architecture

The G^{2}MPLS architectural breakdown in main functional entities is described in this chapter.

An overall picture of these components and the interactions among them is provided in Figure 9-1. For each component a description is provided in the following sub-sections, by specifying

- the main functionality
- the interfaces exposed to the other components, along with their directionality and main action.

The directionality of the interfaces is distinguished in input and output directions, as usual.

The identified functional entities refer to the similar components of the ASON architecture [ASON-ARCH], but in most of the cases they provide the extended functionalities needed to support Grid Network Services. Details on the protocol extensions are out of the scope of this section, because the primary goal of this specification is to provide the G^{2}MPLS architectural concept in terms of abstract messaging and high level semantic for the G^{2}MPLS services. Protocol details are part of the network reference point specifications that will be provided by specific companion documents.

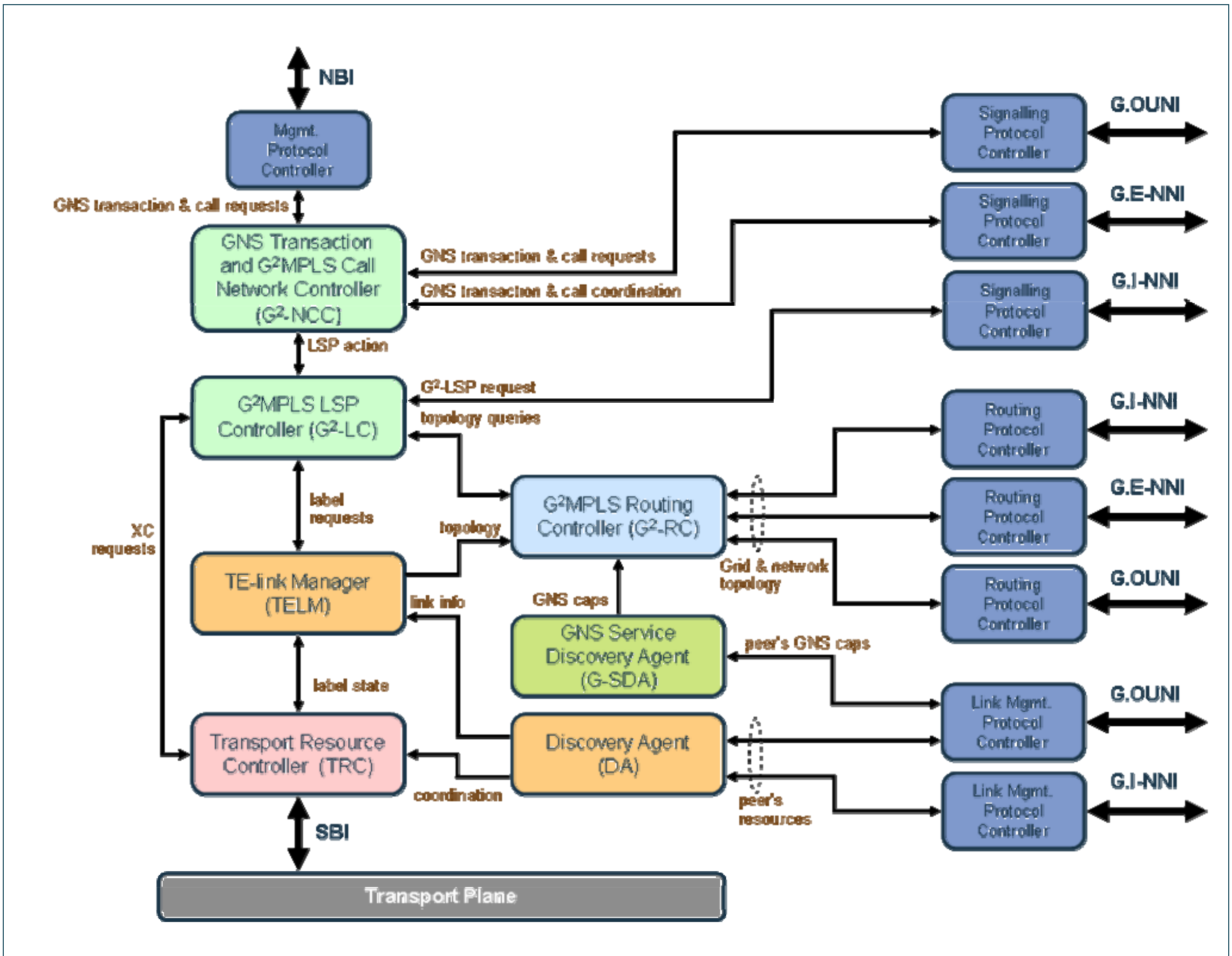


Figure 9-1: G²MPLS functional architecture.

9.1 GNS Transaction and G²MPLS Call Controller (G²-CC)

The GNS Transaction and G²MPLS Call Controller is the functional entity responsible for the control and management of both GNS Transactions and the related G²MPLS Calls. Similarly to the ASON CC, the G²-CC needs to be distinguished in:

- calling/called party G²-CC (G²-CCC), which lies in the user side of the G.OUNI and, according to the G.OUNI and UNI models, may be co-located with end systems (direct invocation model) or acts as a proxy (indirect invocation model),

- network G^2-CC (G^2-NCC), which runs at the network domain boundaries i.e. the network side of the G.OUNI and the upstream/downstream G.E-NNI sides.

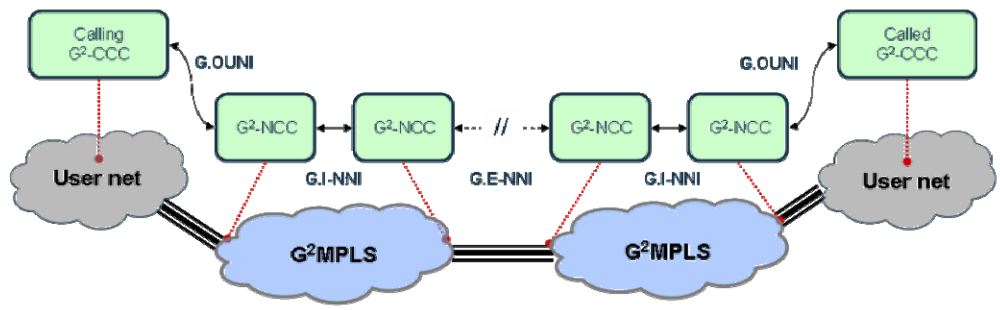


Figure 9-2: Distinction in calling/called party G^2CC and network G^2CC.

The GNS transaction and the G^2MPLS call have an end-to-end scope, but their control and management is needed only when a domain boundary is traversed, be it between the user network and a G^2MPLS domain or between two G^2MPLS domains (ref. Figure 9-2). In compliance with ASON/GMPLS architecture, the consequent steps for routing and signalling the connections (LSPs) under each call segment do not need any knowledge (processing) of the GNS transaction and G^2MPLS call. However, in the implementation they could convey transparently this information from one side to the other by enabling coordination among the G^2-NCCs.

The main functionalities of the G^2-CCC along with its interfaces are provided in Table 9-1.

Interface	Peer	Directionality	Main action
GNS transaction and G^2MPLS call request	G^2-NCC	in/out	Generate or accept a request for activation/deletion/modification of a GNS transaction and its related G^2MPLS calls
GNS transaction and G^2MPLS call notification	G^2-NCC	in/out	Request and retrieve information on the state of GNS transaction and/or its related G2MPLS calls.

Table 9-1: G^2-CCC interfaces.

The main functionalities of the G^2-NCC along with its interfaces are provided in Table 9-2.

Interface	Peer	Directionality	Main action
GNS transaction and G^2MPLS call request	G^2-CCC	in/out	Accept or generate the request for activation/deletion/modification of a GNS transaction and its related G^2MPLS calls
GNS transaction and G^2MPLS call notification	G^2-CCC	in/out	Retrieve and retrieve information on the state of GNS transaction and/or its related G2MPLS calls.

GNS transaction and G^2MPLS call coordination	G^2-NCC	in/out	Progress or accept the request for activation/deletion/modification of a GNS transaction and its related G^2MPLS calls and call segments.
LSP action	G^2-LC	in/out	Generate requests for LSP actions (setup, release, modify, status) towards the G^2MPLS LSP manager.
GNS transaction AAA	External AAA entity	in/out	Request for the authentication and authorization of the GNS transaction request and provide accounting information to an external AAA infrastructure.
NBI	Management Plane Agent	in/out	Provide the configurations and management (status, notifications) for GNS transaction and/or its related G2MPLS calls according to a Soft Permanent Connection model.

Table 9-2: G^2-NCC interfaces.

9.2 G^2MPLS LSP Controller (G^2-LC)

The G^2MPLS LSP Controller is responsible for management of each Label Switched Path (LSP) that is part of a G^2MPLS call. Possible actions on the LSPs are setup, release and modification.

G^2-LC receives requests for LSP actions through its interface with the G^2-CC, and it coordinates the LSP management in the local scope (i.e. on the NCP instance running upon a TNE) and towards the peering NCP instances through G.I-NNI.

G^2-LC exposes interfaces to peering G^2-LC, G^2-RC, TELM and TRC as detailed in Table 9-3.

Interface	Peer	Directionality	Main action
LSP action	G^2-CC	in	Receive incoming requests for LSP actions (setup, release, modify, status) by a G^2MPLS call manager.
Topology queries	G^2-RC	out	Resolve route fragments (completion of loose ERO parts) to progress in local configurations (label picking and cross-connections) and peering signalling.
Label request	TELM	out	Pick a label and its owning data link in a specified TE-link deriving from route information processing (i.e. ERO)
Cross-connection request	TRC	out	Request for a cross-connection between two labels
Peer coordination	peering G^2-LC	in/out	Receive (in) or forward (out) requests for LSP actions (setup, release, modify)

Table 9-3: G^2-LC interfaces.

9.3 G^{2}MPLS Routing Controller (G^2-RC)

The G^{2}MPLS Routing Controller is the functional entity responsible for storing an updated topology view of Grid and network resources. Topology is detailed for the domain under its ownership and summarized at different extents for the neighbouring domains (i.e. just reachability information; reachability + inter-domain network connectivity; reachability + inter-domain + summarized intra-domain network connectivity; etc.).

The G^2-RC uses the topology for the computation of paths upon a request from G^2-LC (with implicit or explicit declaration of the network attachment points as discussed in the previous sections of this document). The computed path scope may range from portions of a route (at the minimum extend the next-hop) to the full end-to-end path across a chain of network domains (inter-domain routing). For this reason and for possible scalability issues, G^2-RC may be implemented either as a unique standalone module either as a distributed set of modules, in both cases complying with the IETF PCE architectural model [IETF-RFC4655].

The detail of the information stored in the G^2-RC topology and, consequently, of the computed routes depends on the adopted routing detail policy, i.e. the amount of information that each network operator configures and publishes internally (i.e. in its domain in case of distributed G^2-RC) and towards in the neighbouring domains.

G^2-RC exposes interfaces to G^2-LC, Routing Protocol Controller (RPC), TELM and G-SDA as detailed in Table 9-4. In case of distributed G^2-RC in a single domain, it exposes also an interface to the federated G^2-RC.

Interface	Peer	Directionality	Main action
Topology queries	G^2-LC or federated G^2-RC	in	Request for route computation (full or fragments.
GNS capabilities retrieval	G-SDA	in	Receive those Grid capabilities of the peering Grid Vsite that are useful for routing.
Local TE-link topology update	TELM	in	Receive updates on the TE-link parameters upon any occurred modification (allocation, de-allocation, fault of resources).
Grid and network topology flooding	RPCs	in/out	Flood local (i.e. TNE internal) and receive remote Grid & network information to feed its own topology and those of the G^2-RCs participating in the domain. Action is mandatory at G.I-NNI and G.E-NNI, while it depends on the permeability level configured at G.OUNI for the “power” Grid user/Vsite. Grid information come from the G.OUNI and has the same end-to-end scope of the network attachment points at the standard UNI (TNAs).

Table 9-4: G^2-RC interfaces.

9.4 TE-link Manager (TELM)

The TE-link Manager is responsible for the management of the TE-links [IETF-RFC4201, IETF-RFC4202] configured under an instance of G^2MPLS NCP. The TE-links are the result of a bundling procedure applied to a number of physical component data-links with the eligibility for being part of the same logical construct[3].

The functionalities of the TELM entity comprise:

- Selection and allocation/de-allocation of resources (<Data-link, label>) in TE-link for signalling purposes,
- Management of the TE-link status and bundling information for topology purposes.

Since the selection mechanism may be requested at the same time on the head-end and far-end of the TE-link, e.g. because of two signalling sessions in counter-directions, some contention on the same resources (<Data-link, label>) may occur. For this reason, the upstream TELM should provide an interface towards the peering downstream TELM by which negotiating the final selected resource; consequently, the downstream TELM is requested to pick the selected resource in a set of possible choices proposed by the upstream TELM. Generally this procedure is implemented by the signalling protocol instance with proper protocol objects (e.g. the Suggested Label and Label Set in G.RSVP-TE for GMPLS networks)

TELM exposes interfaces to G^2-LC, G^2-RC, peering TELM, TRC and DA as detailed in Table 9-5.

Interface	Peer	Directionality	Main action
Label request	G^2-LC	in	Request for selection and allocation/de-allocation of resources (<Data-link, label>)
Local TE-link configuration	NetAdmin	in	Configure the TE-link information set (i.e. TE metrics, resource classes, etc.)
Local TE-link bundling	DA, TRC	in	Retrieve, correlate and bundle Data-link information by DA and label information by TRC
TE-link resource negotiate	peering-TELM	in/out	Select the resource on downstream TELM by picking it in a set provided by the upstream TELM.
Resource connect	DA, TRC	out	Allocate / de-allocate resources (<Data-link, label>)
Local TE-link topology update	G^2-RC	out	Update the TE-link information (e.g. available bandwidth, etc.) upon an allocation or de-allocation of resources on TE-link accordingly to an advertisement policy.

Table 9-5: TELM interfaces.

[3] As per IETF-RFC4201, all component data-links in a bundled TE-link have the same Link Type (i.e., point-to-point or multi-access), the same Traffic Engineering metric, the same set of resource classes at each end of the links, and must begin and end on the same pair of LSRs.

9.5 GNS Service Discovery Agent (G-SDA)

The GNS Service Discovery Agent is the functional entity responsible for discovering Grid and network capabilities between a Vsite attached to the G.OUNI (client side) and the G^2MPLS NCP (network side). Discovery of capabilities is generally referred to as service discovery, and this functional entity is conceived as an extension of the standard service discovery functionality described in [OIF-UNI1.0].

G-SDA main actions comprise:

- Standard service discovery
  - Negotiation of the signalling protocol and its version to be used across the G.OUNI,
  - Correlation of the service attributes of all the transport links connecting the Vsite to the network (e.g. encoding type, signal types, etc.),
  - Discovery of network capabilities (e.g. transparency in case of SONET/SDH, diversity routing capability)
- GNS specific discovery
  - Discovery of Vsite capabilities (e.g. type of CPUs, storage, OS, etc.)

G-SDA exposes interfaces to G^2-RC and peering G-SDA as detailed in Table 9-6.

Interface	Peer	Directionality	Main action
GNS service discovery	peering G-SDA	in/out	Discover GNS service capabilities with adjacent peers. Action is possible only at G.OUNI.
GNS capability update	G^2-RC	out	Update Grid capabilities of the peering Grid Vsite that are useful for routing.

Table 9-6: G-SDA interfaces.

9.6 Discovery Agent (DA)

The Discovery Agent is the functional entity responsible for discovering network resources (i.e. by correlation and verification) in the G^2MPLS NCP. DA can be referred to as neighbour discovery entity, because it discovers resources (i.e. their amount) and can verify them in a link-local scope (i.e. their mapping in the physical Transport Plane).This functionality is the same as per the ASON Discovery Agent [ASON-ARCH], thus the DA in the G^2MPLS NCP exposes the same interfaces.

DA exposes interfaces to TRC, TELM and peering DA as detailed in Table 9-7.

Interface	Peer	Directionality	Main action
Peer coordination	peering DA	in/out	Discover and verify network resources in the link-local scope with adjacent peers.

			Action is possible at G.I-NNI and G.OUNI with the only scope on network resources.
Transport resources coordination	TRC	out	Update label information on the TRC with the results of coordination with peering DAs.
Data-link coordination	TELM	out	Update data-link information on the TELM with the results of coordination with peering DAs.

Table 9-7: DA interfaces.

9.7 Transport Resource Controller (TRC)

The Transport Resource Controller is the functional entity responsible for abstracting the technology specific details of the switching resources for NCP use. The TRC is the functional entity directly interacting with the Transport Plane via the southbound Interface (SBI).

The main functionalities of the TRC are:

- To provide the translation and maintain the bindings between the technology specific name space for transport resources (e.g. in DWDM equipments: <port, wavelength>; in TDM: <port, virtual container>; in Ethernet: <port, VLANs>) and the G^2MPLS name space (<data-link, label>),
- To provide the translation between the technology specific configurations for transport resources (e.g. cross-connections, protections, etc.) and the G^2MPLS ones triggered by G^{2-}LC
- To maintain the NCP bindings among the resources (e.g. cross-connections, bookings, protections/restorations, etc.).

TRC exposes interfaces to TELM, G^2-LC, DA and Transport Plane Agent via SBI as detailed in Table 9-8.

Interface	Peer	Directionality	Main action
Local TE-link bundling	TELM	out	Update label state information
Resource connect	TELM	in	Allocate / de-allocate resources (<Data-link, label>)
Cross-connection request	G^2-LC	in	Request for a cross-connection between two labels
Transport resources coordination	DA	in	Update label information with the results of coordination between DA and its peers.
SBI	Transport Plane Agent	in/out	Provide the technology specific discovery, updates and configurations for transport resources.

Table 9-8: TRC interfaces.

9.8 Protocol Controllers (PC)

As per ASON architecture [ASON-ARCH], the protocol controllers are responsible for mapping the contents and actions of the interfaces described in the previous sub-sections into objects and messages of the protocol instances running on a G^2MPLS Control Plane network element.

Protocol messages are carried through the Data Communication Network (DCN) and, in particular, they use the set of DCN interfaces configured for being part of the Signalling Communication Network (SCN) [ASON-DCN].

Protocol communications between two G^2MPLS instances are possible through the SCN if an adjacency is established between them. The existence of such adjacency implies that an IP path[4] must exist between the two communicating entities. This can be accomplished in different ways:

- IP tunnelling,
- GRE tunnelling,
- A TE-link with interface switching capability of PSC,
- A bidirectional LSP with interface switching capability of PSC.

A single protocol may receive and map semantics of different functional entities, according to a general classification of the functional realm: signalling, routing, link management.

As a result of use cases analysis a set of additional grid parameters for G^2MPLS protocols is defined:

- grid services
  - computation service
  - data storage service
  - visualization service
- hardware parameters
  - integrated system / Cluster
  - # CPUs
  - storage amount
  - system performance
- software parameters
  - OS type
  - OS instances
  - compilers
  - scheduling system
  - other software, daemons and licenses requirements
- security parameters
  - AAA

[4] Since GMPLS and consequently G^2MPLS are designed natively for IP networks, the SCN is expected to support IP at Layer 3.

- normal user / admin
- special rights
- Service Level Agreement (SLA)

- Time parameters
    - start time
    - duration
    - end time
    - preferred timeslots
    - possible timeslots

9.8.1 Signalling (SPC)

The Signalling Protocol Controller is the hub for all the functional entities related to the configuration and management of GNS transactions, G^2MPLS call, call segments and LSPs. Therefore, SPC may interact with the G^2-CCCs, G^2-NCCs, G^2-LCs. SPC may indirectly interact with the TELM when negotiation is needed on a set of candidate resources (the downstream TELM should pick in the set provided by the upstream TELM).

Different signalling protocols might be used for implementing the SPC and surely they all need some extensions in order to support the G^2MPLS services. The possible alternatives depend on the referred network interface:

- At G.OUNI, UNI-RSVP-TE [OIF-UNI1.0R2-COMM, OIF-UNI1.0R2-RSVP] and UNI-CR-LDP [OIF-UNI1.0R2-COMM],
- At G.I-NNI, G.RSVP-TE [IETF-RFC3473] and CR-LDP [IETF-RFC3472],
- At G.E-NNI, ENNI-RSVP-TE, ENNI-CR-LDP, ENNI-PNNI [OIF-E-NNI-Sig-1.0].

The most common choice in operational GMPLS networks is RSVP protocol. For this reason, the companion specification on protocol objects and mechanism in support of G^2MPLS will provide details on G^2.RSVP-TE.

9.8.2 Routing (RPC)

The Routing Protocol Controller is the entity responsible of exchanging all the routing message to/from the G^2-RCs. The most common choice for routing at the Control Plane is to use Interior Gateway Protocols (IGP), such as OSPF or IS-IS. GMPLS routing as well as OIF E-NNI routing use IGPs and OSPF in particular. Generally routing is an internal task of the network, i.e. the user sections of the network does not participate in the exchange of routing information in an overlay model. Therefore, no routing at the UNI is supported by the current standards, both according to OIF and IETF in case of overlay model [IETF RFC4208].

G^{2}MPLS support implies extensions to the routing protocols in order to allow Grid information flooding. Moreover, depending on the permeability level configured for routing information across the G.OUNI, the G^2-RC may have peers also in the user section of the network.

9.8.3 Link Management (LPC)

The Link Management Protocol Controller is the collector for all the functionalities related to service and neighbour discoveries. Therefore, LPC may interact with the G-SDA and DA.

The unique available reference for the implementation of LPC is the LMP protocol [IETF-RFC4204] with its procedures for control channel management, link property correlation (both TE-link and Data-link), link connectivity verification (Data-link), fault management.

LPC acts in the link-local scope and it is not present at the network-network boundaries (i.e. E-NNI), because two different peering domains may communicate at E-NNI, but they can discover available resources and capabilities reciprocally through routing. Instead, LPC is needed at OIF UNI, because it provides a dynamical negotiation means for the offered service and the involved resource in absence of publishing of routing information by both directions.

G^{2}MPLS support implies extensions to the service discovery and thus to the link management protocol, in order to allow Grid capabilities agreement between the Vsite and the accessed network. No extension is foreseen for the standard neighbour discovery, if the permeability level of the G.OUNI is high, i.e. a G^2-RC is present on the UNI client side. On the contrary extensions are needed if the permeability is low, i.e. LPC is used for neighbour discovery and all the routing information is confined into the G^{2}MPLS Control Plane.

10 Path computation issues in G^2MPLS

The possible path computation procedures identified in the context of PHOSPHORUS should be in compliance with the IETF Path Computation Element *(*PCE) architectural model [IETF-RFC4655] since most of the protocol-specific issues are defined and solved in this framework. This model implies that a separate component responsible for the inter- and intra-domain path computation based on specified constrains should be introduced. This component can be identified as an application (different building block) residing within or externally to a network node, providing optimal routes and interacting with the Control Plane for the establishment of the proposed paths. This section provides a brief analysis and description of the architectural considerations that stem from the use of this component for the path computation in G^2MPLS and identifies additional issues that could be addressed with the integration of grid and network resources within the PCE framework.

10.1 Path Computation Element motivation

The PCE is an entity (component, application or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints during the computations [IETF-RFC4655]. The PCE provides Traffic Engineering (TE) functions to a GMPLS enabled network and several deployment scenarios exist arguing where PCEs may be deployed e.g. a PCE may be placed on each single network node or on a local Autonomous Domain (responsible for a number of nodes) and a single or multiple PCEs may be used to compute a given path, or a centralized PCE may be used to compute all paths.

The deployment of a dedicated PCE will relax the processing power needed by a network node to run constrained based routing algorithms and implement highly CPU-intensive optimization techniques. Also it may eliminate the need for the network nodes to maintain the memory demanding Traffic Engineering Database (TED) by establishing it on a separate node and making it available for path computation through the PCE. Another incentive that makes the solution of a separate PCE attractive is the optimal inter-domain routing which can be handled through distributed computation with cooperation among PCEs within each of the domains or even by a central PCE that has access to the complete set of topology information. In cases where the network elements do not have a Control Plane or routing capability but only have a data plane and a Management Plane, it is desirable to run the path computation on the PCE and to send the cross-connection commands to each node on the computed path through the Management Plane. Additionally the PCE can in an efficient manner consider local policies that impact the path computation and selection, in response to a path computation request, and also it can be used to compute backup paths in the context of fast reroute protection of TE LSPs. Finally the sophisticated constraint routing algorithms utilize by the PCE can in a convenient way address issues like: i) resource coordination (e.g. CPUs, storage) ii) advance reservation iii) physical layer impairments in transparent optical networks iii) different connection types (unicast, multicast or anycast) and iv) QoS.

10.2 Path Computation Element architectural considerations

The path computation in a domain can follow two different models: the Centralized Computation Model and the Distributed Computation Model. According to the former model all the path computations for a domain will be performed by a single centralized PCE which receives all the computation requests. This model has a single point of failure, and therefore a backup PCE can be used to take over the computation responsibility in the event of a failure of the primary PCE. The later model considers a domain that may include multiple PCEs and the computation of paths is shared among the PCEs. Therefore a given path may be computed by a single PCE or multiple PCEs, and a Grid user (i.e. the Path Computation Client (PCC) [IETF-RFC4655]) – may be linked to a particular PCE or may be able to choose among several PCEs.

In order for the PCC to communicate with a PCE, it needs first to learn its location. This can be achieved either through local configuration at the PCC or on a protocol-based discovery mechanism that can be additionally governed by a specific policy that can for example provide load sharing between PCEs. Also the capabilities of each PCE should be available to the PCC through static configuration or even dynamically in order to assist the application PCC chose which PCE to use. Some possible capabilities that may be advertised or configured could include, a set of constraints that the PCE accounts for (like diversity, shared risk link groups (SRLGs), optical impairments, wavelength continuity etc.) and computational capacity considerations (e.g. the number of computations it can perform per second).

The communication between the PCC and the PCE i.e. the path computation requests from the PCC and the return of a path computation response from the PCE can be performed by a response/request protocol. The path computation request may include a set of requirements, such as the following:

- the source and destination of the path
- the bandwidth and other QoS parameters desired
- resources and SRLGs to use or avoid
- the number of disjoint paths required and whether near-disjoint paths are acceptable
- the levels of resiliency , reliability and robustness of the path resources
- policy-related information
- time constraints
- job description

A request/response protocol is also required for a PCE to convey path computation requests to another PCE and for the PCE to return the path computation response. An important feature of the PCEs that are cooperating to compute a path is that they apply compatible or identical computation algorithms and coordinated policies. This requires coordination through the communication between the PCEs. When multiple PCEs cooperate to compute a path it is important that they have a coordinated view of the meaning of constraints particularly when the PCEs are responsible for different domains.

10.3 Path Computation process

Important issues that are addressed by the deployment of PCE deal with heterogeneous technologies which require the development of multi-region path computation techniques and also with the development of scalable distribution of TE data in a multi-domain environment. One of the possible approaches proposed to be followed is described in this section. The PCE in this case, represents a local Autonomous Domain (AD) that acts as a protocol listener to the intra-domain routing protocols e.g. OSPF-TE, and is also responsible for inter-domain routing. PCEs peer across domains and exchange abstract or actual topology information to enable inter-domain path computation and also utilize a modified version of OSPF-TE to share a link state database between domains. PCEs also include advanced algorithms which allow path computation with multiple constraints like physical layer impairments, network resource coordination, advance reservations and different connection types. The constraint path computation process performed by the PCE can be described briefly in the following steps. Prior to path request all necessary information on grid resources are identified and transferred to a MetaSheduling Service (MSS) responsible for the orchestration of resources of different sites belonging to different administrative domains, based on advanced reservations [VIOLA UBonn]. Also the TED is continuously updated with link state information driven from a routing daemon i.e. OSPF. Upon a request arrival the user specified parameters carried by the LSP request are parsed into constraints inside the PCE which takes the responsibility to provide the required end to end path if possible. This process can actually be quite complex and may require multiple interactions with MSS and PCEs in other domains. For example Grid applications may require the allocation of a number of different Grid resources for their execution in various scenarios. These resources usually are distributed on different Grid sites and dynamic optical network connection provisioning must be triggered considering all the resources required by the application. Connections may be initiated for example due to the necessity of code migration to other sites, for data retrieval and storage, for load balancing purposes, when parallel simulations are required or even for visualization of resources with real time characteristics. The coordination of such applications requires the rapid discovery of appropriate connections which are the result of very complex and intensive path computations. PCE can assist to this direction through the abstracted information of the global topology stored in the TED, of each domain, especially in cases where the network Management Plane is not able to provide this functionality.

The situation becomes even more complicated when Grid users request advance reservations to guarantee the availability of network resources. Advance reservation is a mechanism that allows a user to request exclusive access, for a specific time interval in the future, to a set of resources that satisfy specific requirements. Local Grid resources are basically under the control of Middleware's (MW) local schedulers of the Grid sites but the overall advance reservation of Grid and network resources is under the control of the G^2MPLS. Therefore the reservations should be supported in two phases. In the first phase the MSS acting as a global Grid Broker must take into consideration the user time constraint demands and derive a Grid resource schedule by incorporating resources from different Grid sites. In the second phase the PCE of the G^2MPLS Control Plane must discover the available network resources according to the LSP schedule maintained at the PCE, necessary to provide access to the indicated Grid resources. In this way a reduced network topology is created which will be utilized by advance routing algorithms that will be developed in conjunction with WP5 activities for optimal path computation. The LSP schedule information can be exchanged as resource reservation TE attributes in order to be translated into routing constraints understandable by the PCE. In addition the resource reservation states of the LSP schedule table at the PCE will be updated dynamically so that the path computation results can always be reflected in resource states and be disseminated to other domains. Therefore any user requesting an advance schedule service can be provided instantly with a deterministic answer. Upon successful advance reservation, the reserved time slots of related resources will be dedicated to that service and will not be allocated to any other services. Also heuristics supporting malleable reservations can be introduced to the PCE, for cases where the job start time is not strictly fixed in order to avoid resource fragmentation that would lead to utilization degradation. The coexistence of immediate and advance reservations is another challenging task that algorithms implemented at the PCE could deal with in an efficient way. In this case optimal resource partitioning or pre-emption techniques can be implemented that will affect the routing decisions of the PCE.

Except from the Grid constraints (Grid resource scheduling and coordination) described above the PCE constructs the reduced topology of the network considering also QoS issues and policy restrictions. Network QoS provisioning is of great significance in Grid applications which require high throughput (bandwidth), low latency (delay and jitter) and low packet-loss rate (reliability) and can be deployed using various mechanisms. Admission control and scheduling for example can lead to altering the network topology by dynamically disabling or enabling link connections accordingly for satisfying certain service requirements.

Based on the created reduced topology the algorithms that are implemented on the PCE proceed to the path calculation taking into consideration several constraints related with the nature of the submitted job, the network and the available resources. One of the constraints that the PCE must handle deals with physical layer impairments. As signals propagate along the optical paths without OEO regeneration the physical impairments associated with the transmission line and the optical switching nodes accumulate resulting in some cases in unacceptable signal quality. Therefore, optical layer quality monitoring and optical quality–based network service provisioning (routing and resource allocation) become critical for connection SLA assurance. The implementation of certain RWA algorithms that consider signal impairments and constraint the routing of wavelength channels according to the physical characteristics of the optical network paths can improve the performance of connection requests. These algorithms are reported as Impairment Constraint Based Routing (ICBR) algorithms and ensure that connections are feasible to be established considering not only the network level conditions (connectivity, capacity availability etc) but also the equally important physical performance of the connections. The difficulties of a constrained based routing approach stem from the seemingly diverse nature of the networking and physical performance issues that have to be considered. From one hand there are the physical layer impairments (linear like Amplifier induce noise, Polarization Mode Dispersion, Chromatic Dispersion, in-band crosstalk, filter concatenation and nonlinear like Self-Phase Modulation, Cross-Phase Modulation, Four Wave Mixing) that have to be taken into account and on the other hand there are the networking aspects (blocking probability, end-to-end delay, throughput) that capture and describe the overall performance of the optical network. It is evident, that all these heterogeneous issues have to be modelled and unified under a properly designed framework that will provide a solution for the RWA problem, which will be both feasible and efficient. Such algorithms will be developed by WP5 and incorporated to the PCE in order to provide paths with improved quality characteristics. For the realization of these algorithms by the PCE the majority of the impairments must be measured or derived using fast analytical modelling techniques on a link by link basis and therefore the PCE must interact closely with mechanisms that provides optical physical layer monitoring information. The dissemination of the impairment information by the network components is a complicated task that can be performed through the use of appropriate signaling mechanisms and protocols (e.g. RSVP) to guarantee the Service Level Agreements (SLAs). At the final step of the connection discovery impairment constraint based routing algorithms are applied and compute a path which is returned from the PCE in the form of an Explicit Route Object (ERO).

In conjunction with optical impairments, algorithms developed at the PCE should deal with resiliency strategies (protection, restoration) and provide solutions to support different connection types (unicast, multicast, anycast). Integrated approaches that consider multiple parameters can be developed to provide the ability to the PCE to compute optimal paths maintaining high network performance and implementing optimize protection and restoration schemes.

11 G^{2}MPLS Robustness

The G^{2}MPLS objective of controlling different technologies in the Transport Plane (e.g. optical and electrical) under the same Control Plane instance and of supporting seamlessly Grid infrastructures raises important requirements on the recovery of faulty conditions affecting Grid Network Services.

In general, the term recovery refers to the capability of the system to preserve or recover the integrity of an existing transport service in the event of any failures within the Transport Network. GMPLS enables two main classes of recovery procedures: protections and restorations.

However, also the Control Plane infrastructure can be affected by faults (node- or link-scoped), mainly affecting the SCN and the Control Plane communications but possibly resulting in degradations of the operated Grid Network Services.

In this section the two issues are discussed, because both contribute to the robustness of the G^{2}MPLS Control Plane.

11.1 G^{2}MPLS transport service recovery

G^{2}MPLS recovery uses Control Plane mechanisms (i.e. signalling, routing, link management mechanisms) to support Transport Plane fault recovery [IETF-RFC4626, IETF-RFC4627, and IETF-RFC4628]. The focus of the available standards is mainly in the single domain scope, but the more complex inter-domain recovery problem is currently under specification [IETF-IDREC, IETF-SEGREC, and IETF-E2ESIGREC].

A taxonomy of recovery in the Transport Plane services may be done according to area and time scale for the entire operation. The recovery area distinguishes between:

- *path-level* recovery procedures, in which failure notifications are propagated till the end nodes of the affected service and there solved;
- *segment-level* recovery procedures, in which failures are notified and solved over a portion of the network traversed by a segment LSP

- *span-level* recovery procedures, in which failures are notified and solved locally at involved nodes, i.e. those next to the failed resource.

Regarding the recovery time scale, distinction exists between those faster strategies which perform pre-calculation and pre-allocation of a backup connection or set of spans (i.e. protection), and other slower strategies in which a new connection (or set of spans), at the most pre-calculated, are dynamically allocated at time of failure (i.e. restoration).

As protection and restoration techniques may be performed both at path-level and at segment-level and at span-level, an overall taxonomy of recovery may be sketched as in Figure 11-1.

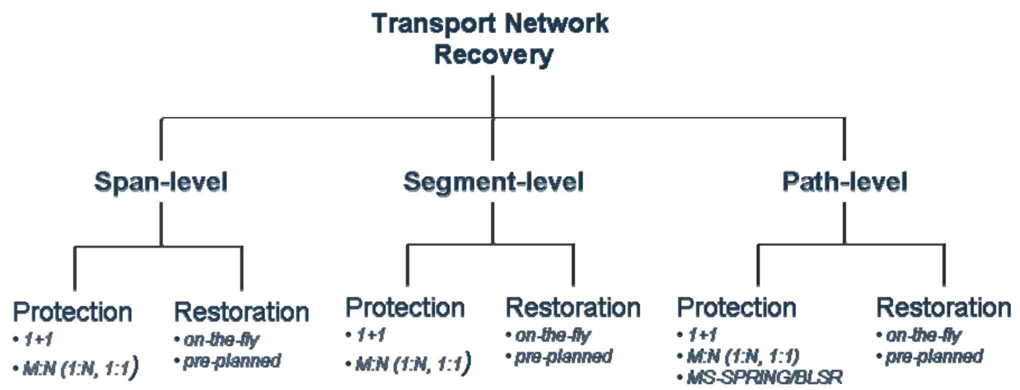


Figure 11-1: A taxonomy of recovery procedures.

An important feature which affects anyway a restoration technique, be it local, segment or end-to-end, is the centralized or distributed approach for the implementation of the recovery manager (path computation and signalling).

Centralized restoration techniques compute primary and backup resources in a central controller processing all the possible fault notifications generated in the domain under its control. This controller might be placed in the Control Plane e.g. in the form of a PCE element [IETF-RFC4655] or in the Management Plane, i.e. in the Network Management System (NMS). In both cases, having the knowledge of the whole topology and the utilization of the network at that time, the central controller can compute primary and/or backup paths and coordinate with the underlying network nodes by signalling (Control Plane case) or direct configurations (Management Plane case), in order to solve the faulty condition. Therefore, network intelligence is concentrated in a single point of the network and most of nodes have limited Control Plane capabilities (e.g. only failure detection and notification capabilities).

Distributed restoration techniques are based on full Control Plane capabilities for each node in the network, in order to provide those nodes with the means for failure management and for the creation of backup paths. This is the GMPLS and G^2MPLS preferred approach.

The restoration time is mostly contributed by the time spent for failure detection, failure localization/isolation (if needed), for failure notification and also for the computation and allocation of the backup connection along the network. All these times are obviously saved in a protection procedure, in which only the traffic switching from the failed path to its backup is needed.

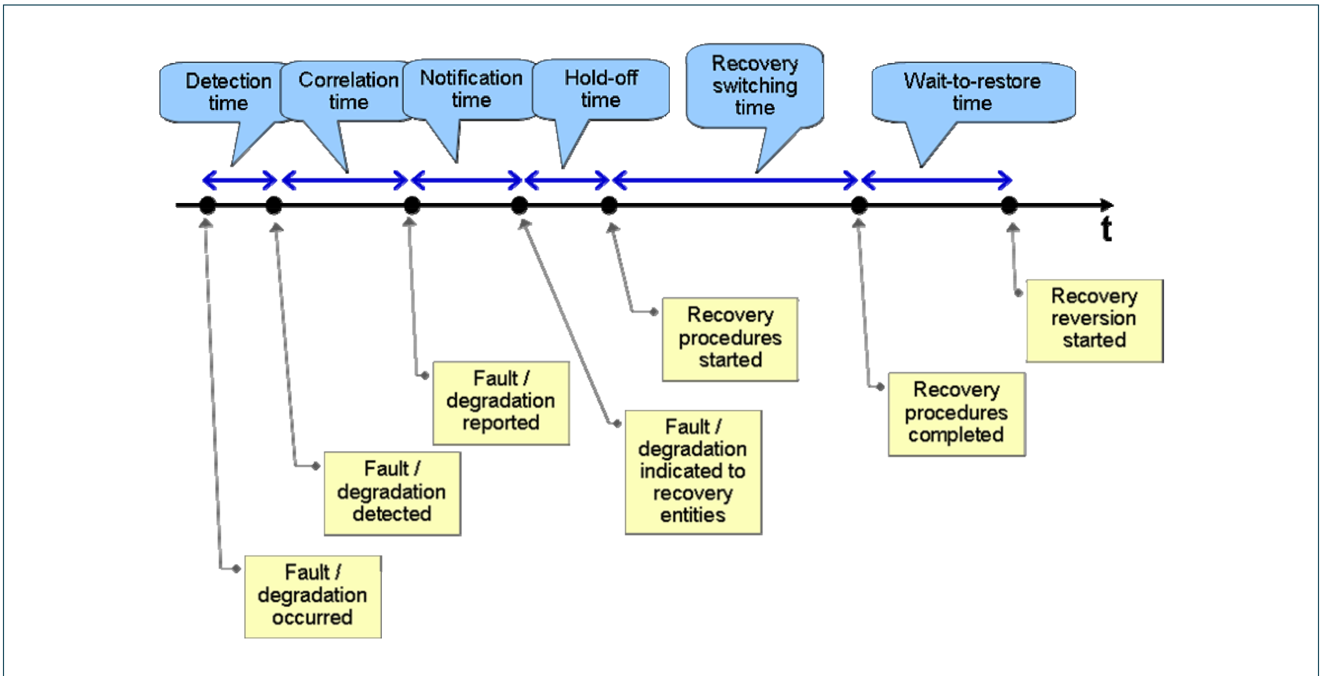


Figure 11-2: Recovery timeframe.

G^2MPLS Transport Service recovery complies with the recovery specifications for GMPLS [IETF-RFC4626, IETF-RFC4627, IETF-RFC4628, IETF-IDREC, IETF-SEGREC, IETF-E2ESIGREC] and exposes the same functional entities as detailed in Table 11-1 and Figure 11-3.

Entity	Actions	Main action
Detecting Entity	Failure detection	Detects a failure or group of failures. The list of failures is non-correlated.
Reporting Entity	Failure correlation and notification	Correlates the received list of failures and reports the resulting fault to the deciding entity.
Deciding Entity	Recovery decision	Makes the recovery decision or selects the recovery resources. It has to be one of the two end-nodes of the LSP/span (the “master”).
Recovering Entity	Recovery activation Recovery reversion	Provides the recovery of the LSPs/spans. It can be any node along the LSP/span (the two end-nodes, at least).

Table 11-1: G²MPLS recovery functional entities.

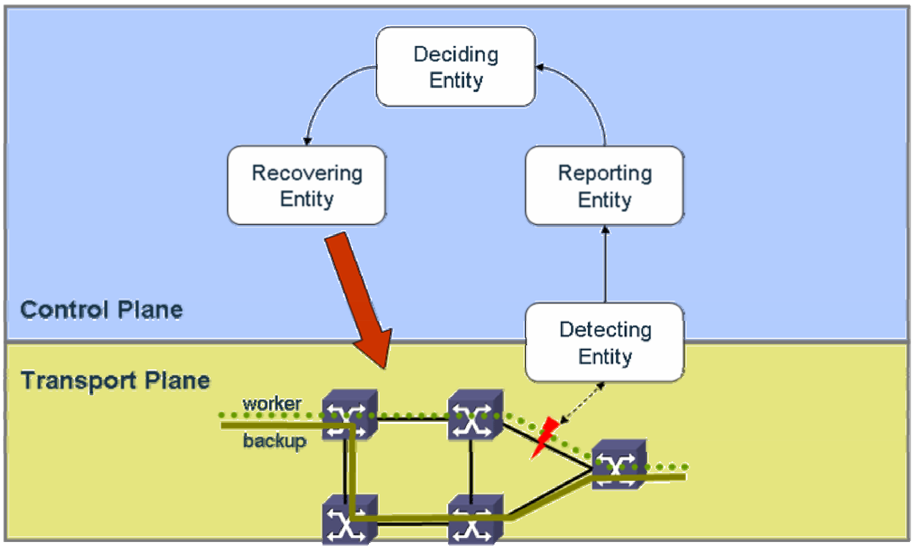


Figure 11-3: Recovery entities.

11.1.1 Fault management

Fault management comprises the tree fault related phases of recovery:

1. fault detection
2. fault correlation (localization and isolation)
3. fault notification

Failure detection for Transport Plane resources is strictly dependent on the Transport Plane technology. It is generally bound to the occurrence of a Signal Degrade (SD) or Signal Fail (SF)/Loss of Light (LOL) for the installed connections. The generation of such a signal depends on the specific technology used in the Transport Plane and, above all, on the availability of signal supervision means such as in the case of SDH/SONET or OTN (i.e. G.709) networks. For G²MPLS purposes the southbound interface towards the transport network equipment must provide some asynchronous mechanisms for signalling the occurrence of a fault on the configured data paths.

Failure correlation is the process by which a list of correlated failures is collected before reporting. This is the case of different alarms affecting the same resource with and increasing level of defect or the case of a single critical failure (e.g. an entire link) that can cause failure conditions to a number of LSP using it. In G^2MPLS this functionality can be provided by the Control Plane in the Transport Resource Controller functional entity.

Failure localization and isolation provides information about the location and identity of the failed Transport Plane resource. It is possible time consuming task, particularly in case of end-to-end recovery in which the deciding entity could be located at the head-end or far-end of the connection and has no information on the fault origin but the LSP ID. In G^2MPLS this functionality is optional and strictly depends on the adopted recovery mechanism, i.e. it could be useless and time consuming in case of end-to-end restoration, it could be very important in case of local repairs.

Failure notification is the process by which the correlated failure is notified to the involved nodes and the recovery decision entity. This is a mandatory trigger for the recovery procedures in each control and Management Plane technology.

11.1.2 Protection procedures

Recovery strategies for transport networks providing commercial services are mostly protection techniques. Examples are SDH/SONET 1+1 and M: N span- and path-protections. A brief summary of the G[^{2}]MPLS support for protection procedures is provided in Table 11-2.

Protection	GMPLS Protection	Description
1+1	Dedicated 1+1	• 1 dedicated protection LSP/span protects 1 working LSP/span • the normal traffic is permanently duplicated at the ingress node on both the working and protection LSPs/spans • No extra traffic can be carried over the protection LSP/span
1:1	Dedicated 1:1	• 1 specific recovery LSP/span protects 1 specific working LSP/span • the normal traffic is transmitted over only one LSP (working or recovery) at a time • Extra traffic can be transported on the recovery LSP/span resources
1:N (N >= 1)	Shared	• 1 recovery LSP/span protects of up to N working LSPs/spans, all ending on the same node pair • Extra traffic can be transported over the recovery LSP/span • The working LSPs/spans should be resource disjoint in the network, thus not sharing any failure probability (not mandatory, but very useful)
M:N (N >= 1, N >= M)	Shared	• M specific recovery LSPs/spans protect N specific working LSPs/spans, all ending on the same node pair • Extra traffic can be transported over the M recovery LSPs/spans when available • The working LSPs/spans should be resource disjoint in the network, thus not sharing any failure probability (not mandatory, but very useful)
BLSR/ MS-SPRING	Enhanced	• Applicable to rings only in 2 or 4 fibers topologies. • Customization to rings of the 1:1 procedure

		• Extra traffic can be transported over the recovery ring side when available • The working LSPs/spans should be resource disjoint in the network, thus not sharing any failure probability (not mandatory, but very useful)

Table 11-2: GMPLS/G^{2}MPLS protection procedures.

11.1.3 Restoration procedures

As transport network clients are supposed to ask not only for bandwidth increase but, above all, for dynamic circuit creation, network utilization will soon become a must. In this scenario restoration techniques will be more suitable, and new transport services for recovery purposes will be set up only when they are needed, thus at failure time.

11.1.3.1 *Transport Network scope only*

A brief summary of the G[^{2}]MPLS support for restoration procedures is provided in Table 11-3.

Restoration	Scope	Description
Pre-Planned	LSP/segment	• an end-to-end restoration path is pre-calculated before failure • a signaling message is sent along this pre-selected path to reserve bandwidth • the resources reserved on each link of a restoration path may be shared across different working LSPs • on failure detection, LSP signaling is performed along the restoration to actuate the cross-connections
Shared-Mesh	LSP/segment	A case of Pre-planned LSP restoration, where the pre-planning of protection LSPs can also include resources already planned for other protection LSPs
On-the-fly	LSP/segment	• an end-to-end restoration path is established after failure • the restoration path may be dynamically calculated after failure (a.k.a. "**full LSP restoration"**), or pre-calculated before failure • no signaling is used along the restoration path before failure • no restoration bandwidth reservation • it may reuse the intermediate node's resources of the working LSP under failure condition (and may also include additional intermediate nodes) • Two variants for reversion: o *Hard LSP Restoration*: the LSP is released before the full establishment of an alternate LSP (*break-before-make*) o *Soft LSP Restoration*: the LSP is released after the full establishment of an alternate LSP (*make-before-break*)

Table 11-3: GMPLS/G^{2}MPLS restoration procedures.

Some further restoration schemes obtained from the combination of the basic ones are listed in the following and are of interest in the G^2MPLS Control Plane architecture:

- **Revertive restoration without working LSP removal**
  - 1 working LSP, restored with a protection LSP created after failure
  - The working LSP is not removed after protecting
  - The traffic is reverted to the working LSP after healing the failed resource, and the protecting LSP is removed
- **1+1 protection with restoration of protecting LSP**
  - If a failure occurs on the protecting LSP, that LSP is restored on-the-fly (be it carrying traffic or not)
  - The former protecting LSP can be removed…
  - … or left in place (e.g. if the protecting LSP is idle when failure occurs, its protection will be reverted to it, if no failure occurs on the worker LSP meanwhile)
- **M:N protection with restoration of shared protecting LSPs**
  - Same as above for failures on the protecting LSP
  - …with one more trigger for restoration: the possibility to add dynamically new LSPs in case of usage of the available M…
  - ,,,a kind of M[+L]:N: M protections + L restorations for N working LSPs
- **Pre-planned restoration with re-planning**
  - On failures of pre-planned protecting resources (if detected!), the protecting LSP is recalculated.

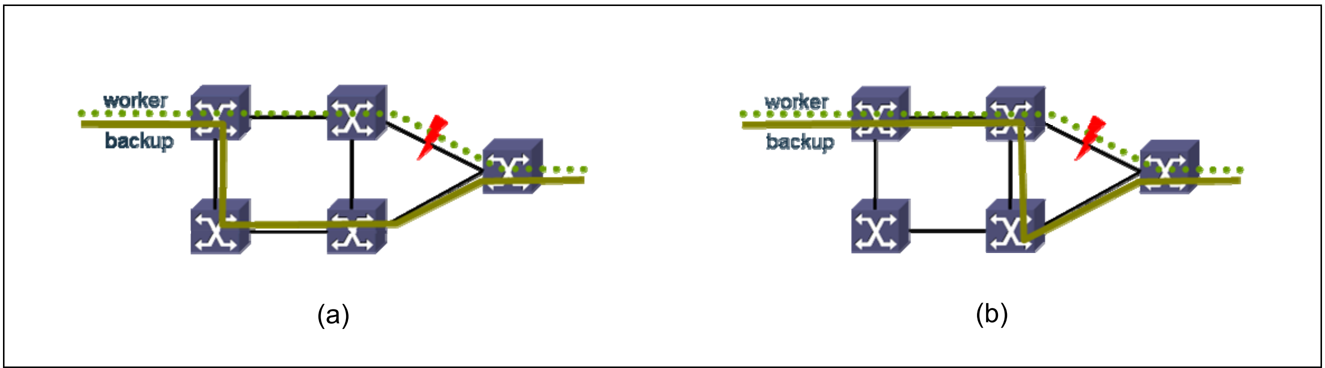


Figure 11-4: Recovery procedures: a) end-to-end; b) segment/local repair.

Pre-provisioning is aimed to lower the recovery time for a failing connection, as well as local repair strategies promise to be more suitable than end-to-end ones for keeping restoration times in few hundreds of milliseconds. However, in both cases a higher signalling load is needed with respect to end-to-end techniques [IETF-SEGREC].

As a matter of fact, local repair strategies are prone to waste resources in the network, because of the sub-optimality of the resulting backup paths, fixed a computation rule (e.g. shortest path, fixed a cost for each link). On the other hand, end-to-end restoration strategies are more efficient, providing the computation of the best end-to-end backup path in the network (e.g. the shortest). However, both strategies are sub-optimal in the pre-planned mode, as pre-planning could be seen as closer to protection than restoration and backup paths do not take into account any modification of the network load occurred in the meanwhile. An exhaustive analysis of pros and cons of the local-repair and end-to-end approaches (e.g. efficiency, speed, Control Plane function requirements, etc.) is beyond the scope of this deliverable and may be found in [IETF-RFC4428].

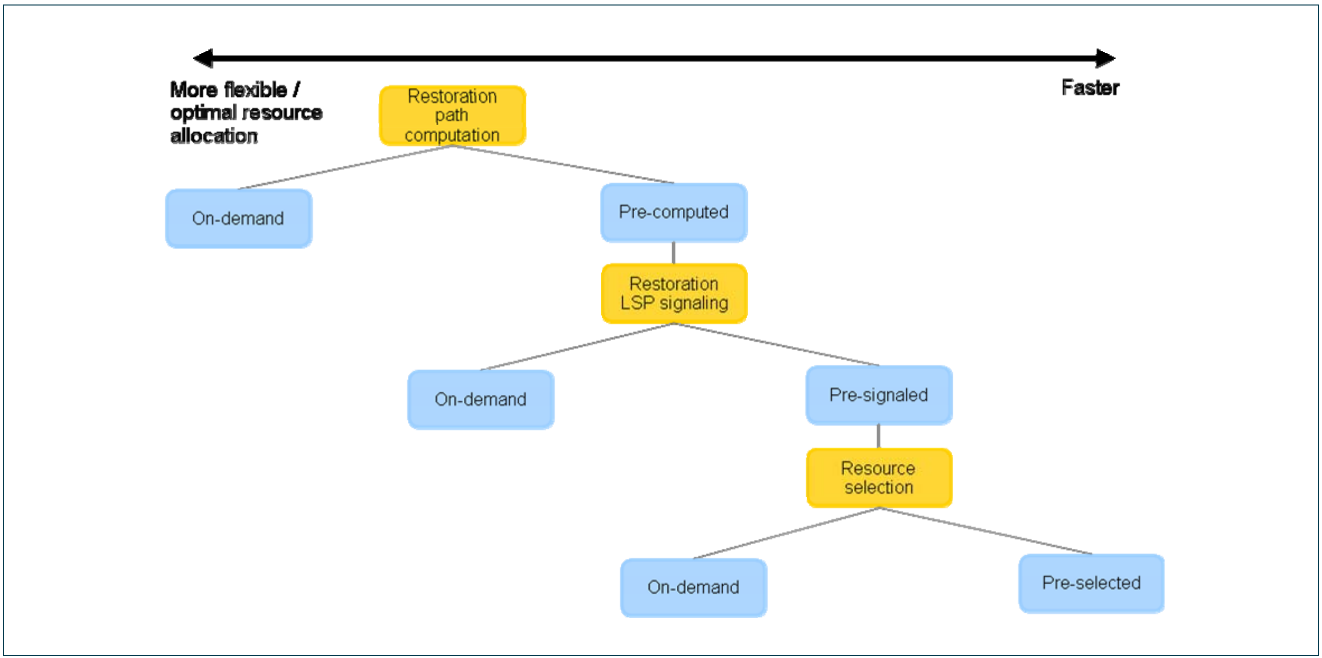


Figure 11-5: Evaluation of pre-provisioning performances in restorations.

The multi-technology and multi-domain nature of the PHOSPHORUS project claims for a multi-layer recovery management based on restorations. PHOSPHORUS hierarchies exist both horizontally (i.e. among similar technologies administratively partitioned) and vertically (i.e. among nested technologies).

In general, different technological layers in a layered network can have different and specific recovery mechanisms. It is expected that the recovery action is taken by the recoverable LSP/span closest to the failure in order to avoid the multiplication and contention of recovery actions. Moreover, it is expected that a failure is properly correlated and isolated, in order to avoid notification on various deciding entities on different layers and consequently a race of concurrent recovery procedures.

The hierarchical restoration problem is faced in G^2MPLS according to four main directions:

- Restoration for inter domain network services, which include those means compliant with [IETF-IDREC] for mitigating faulty network condition when inter-domain connections are involved
- Coordination between the G^2MPLS layer and the Inter-domain layer (Network Service Plane + NRPS), which is need when the network service is built trough the interaction between G^2MPLS domains and

e.g. NRPS-controlled domains capable of proper recovery strategies. This is a special case inter-domain recovery coordination.

- Escalation of network recovery strategies, which makes possible to escalate a hierarchy of recovery procedures (e.g. protection, end-to-end restoration, etc.) in case of blocking in the adopted recovery action. In this case the user should accept the possible degraded network resiliency and, thus, this behaviour should be part of a proper agreement at the service discovery phase.
- Coordination among the different network layers, which includes those means needed to avoid race conditions between communicating layers, e.g. MPLS-TE and G^2MPLS.

11.1.3.2 *Joint (and seamless) network & grid restoration procedures*

The elementary unit at the Grid layer is the single job, intended as a self-consistent task using a certain amount of resources in a specified timeframe within the Virtual Organization. Depending on the definition of the job contents and objectives, a number of network calls and connections are created in the G^2MPLS network and each network connection, segment, span could be recovered in case of fault in the network, according to the procedures described in the previous sections. However, some critical and unrecoverable fault condition might occur in the network, especially in case of limited connectivity and/or meshing degree. Depending on the seriousness and impact of the occurring network fault, it could be impossible for the Network Control Plane to recover the service. In such a case an escalation from the network layer to the Grid layer could be needed, triggered by timely fault notifications across the G.OUNI.

The consumer of fault notifications from the network should be those entities of the Grid middleware responsible for job monitoring and workflow check-pointing. These entities are expected to react at the notification e.g. by re-scheduling the failing/failed job in a different timeframe and/or on different Grid resources.

Mechanism for reversion are obviously not needed in this case, because it is assumed that the Network Control Plane is no more master in the recovery action and there is no foreseen benefit in the Grid layer to revert the job on the previous network resources. However, some mechanisms for the removal of possible orphans of network connection/segments should be implemented in the Control Plane, according to the procedures established by the IETF standards.

11.1.4 Diversity support

The G^2MPLS Routing Controller must compute worker and protection paths in a way which is logically diverse. The different levels of disjointness defined for G^2MPLS are:

- node, in which different nodes (and different links) are crossed by the primary-backup pair of routes;
- link, in which different links are crossed by the routes;
- SRLG, in which the Shared Risk Link Group lists of the two routes have no intersection.

The first two kinds of disjointness are related to the logical layer of the transport network, e.g. the TE-topology. Node disjointness is a more stringent condition than the link one; so, two node disjoint paths are also link disjoint. The two levels are distinguished because a topology might block node disjoint search, but might still provide link disjointness.

Link and node diversity between working and recovery LSPs/spans does not guarantee complete disjointness. Due to the common physical layer topology (passive), additional hierarchical concepts such as the Shared Risk Link Group (SRLG) are used to locate conduits, fibers or general risks for physical resources.

SRLG disjointness is a necessary (but not sufficient) condition to ensure optical network survivability. With respect to the physical network resources, a working-recovery LSP/span pair must be SRLG disjoint in case of dedicated recovery type. On the other hand, in case of shared recovery, a group of working LSP/span must be mutually SRLG-disjoint in order to allow for a (single and common) shared recovery LSP itself SRLG-disjoint from each of the working LSPs/spans.

11.2 G^{2}MPLS Control Plane Resilience

Two major types of faults can impact a Network Control Plane:

- control channel fault, which affects communications between neighbouring nodes,
- nodal fault, which affect the entire network element hosting the Control Plane instance.

Control channel recovery is an issue under the responsibility of the SCN, above all in case of out-band implementation. Procedures for protocol state persistence and message integrity just defined in GMPLS should guarantee that the recovery of a faulty control channel does not imply service disruptions. These procedures are particularly relevant for signalling protocols; in fact, after an SCN disconnect event, the signalling protocol controller should announce its persisted state to the requesting neighbours and start synchronizing on the pending transactions interrupted by the fault.

On the contrary, Control Plane restarts are much more demanding because most of the state information got lost and the realignment procedure should involve the failed node and all its neighbours for a cross-check of the mappings and resource bindings. Most of these procedures are protocol specific and will be formalized in the companion deliverables on G^{2}MPLS protocols. The identification of the persistent data per each involved protocol will be provided in those documents as well.

12 Interworking of G^2MPLS and Network Resource Provisioning Systems (NRPS)

This chapter deals with the problem of coexistence and interworking of G^2MPLS and Network Resource Provisioning Systems. The discussion is mainly focused on the proposition of an overall interworking scenario with a brief overview of the shared interfaces between GMPLS/G^2MPLS and NRPS-es (e.g. DRAC, ARGON or UCLP). The proposed scenario is the result of an agreement between PHOSPHORUS WP1 and WP2 teams and it is aimed to cope with the two phases of the project, i.e. the first phase, in which only GMPLS developments will be available, and the second phase, in which the full G^2MPLS suite will be tested.

The overall PHOSPHORUS architecture with GMPLS/G^2MPLS components and NRPS-es is summarized in Figure 12-1.

From a pure G^2MPLS perspective, interworking needs to be achieved with two main modules:

- the Network Service Plane (NSP), i.e. the adaptation layer between the Grid layer and the NRPS-es that provides inter-domain routing and connection request coordination;
- the NRPS itself.

Consequently, three scenarios may be identified:

- *Case A*: standard GMPLS as the only controller of a domain interfaced to the NSP through an NRPS driver[5] glue module,
- *Case B:* standard GMPLS as "slave" controller of a domain mastered by an NRPS;
- *Case C:* full-fledged G^2MPLS as the only controller of its Grid & network resources and peering with a neighbouring NSP.

[5] According to WP1 deliverable D1.1, the NRPS Driver is an adaptor that translates the proprietary interface of each NRPS to a single common interface to interact with the Network Service Plane. On top of each NRPS there will be an NRPS Driver that presents to the Network Service Plane a common interface, independently of the NRPS running underneath. The same concept can be applied to GMPLS in order to cope with those deployment scenarios (i.e. local test-beds) in which neither an NRPS nor the G^2MPLS is available.

Case A and B involve the G^2MPLS northbound interface, while Case C involves the east-west interface.

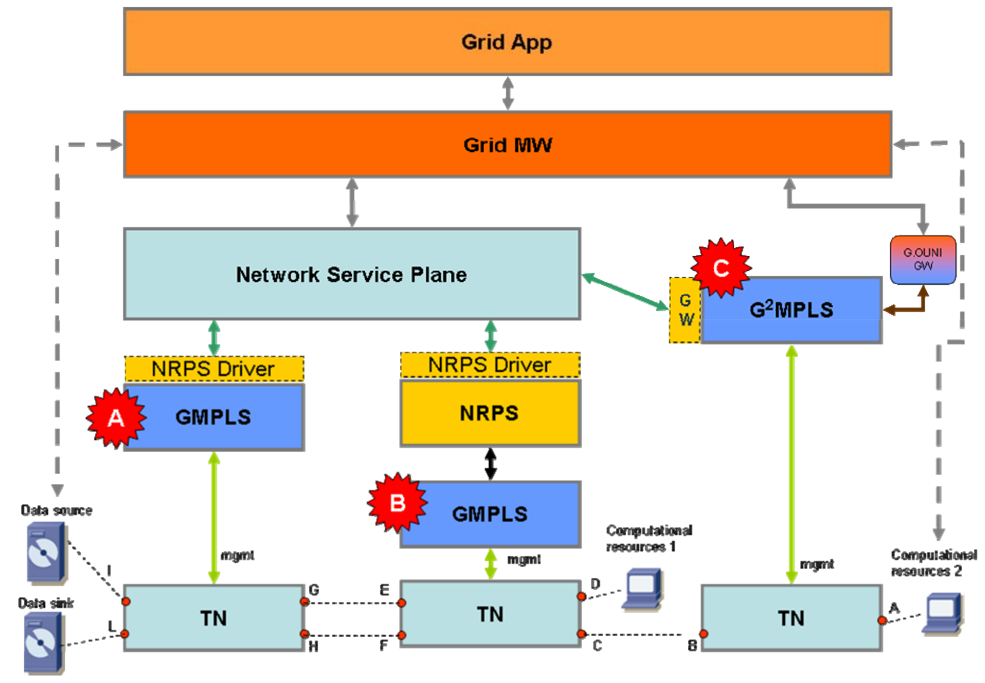


Figure 12-1: GMPLS/G^2MPLS interworking with NRPS.

12.1 Interworking through northbound interface

NRPS owns the control of just network resources within its domain. Therefore, the use of the G^2MPLS northbound interface by an NRPS and/or the NSP limits the types of service configurable at that interface. In this context, the G^2MPLS northbound interface is used only to request/tear down network connections and to retrieve topology and connection status information; therefore, the G^2MPLS is used as standard GMPLS Control Plane.

In Figure 12-2 the two possible options for using GMPLS are shown by means of a use case. At step 1, the Grid application submits a job request to the Grid Middleware, asking for the configuration of a Grid job between “Computational resource 1” and “Data Source”. This job request includes also the configuration of the network resources in the Grid. Once the Grid Middleware (e.g. through its MetaScheduling Service) localizes all the resources, it starts negotiating availabilities by means of steps 2.a, 2.b and 2.c. The NSP is triggered by step 2.b and, in particular, the NRPS Broker in the NSP is responsible for selecting the inter-domain route and coordinating the inter-domain connection establishment. Once an inter-domain route for the required network service is computed, the NRPS Broker forwards the domain-specific requests to the involved underlying resource provisioning systems.

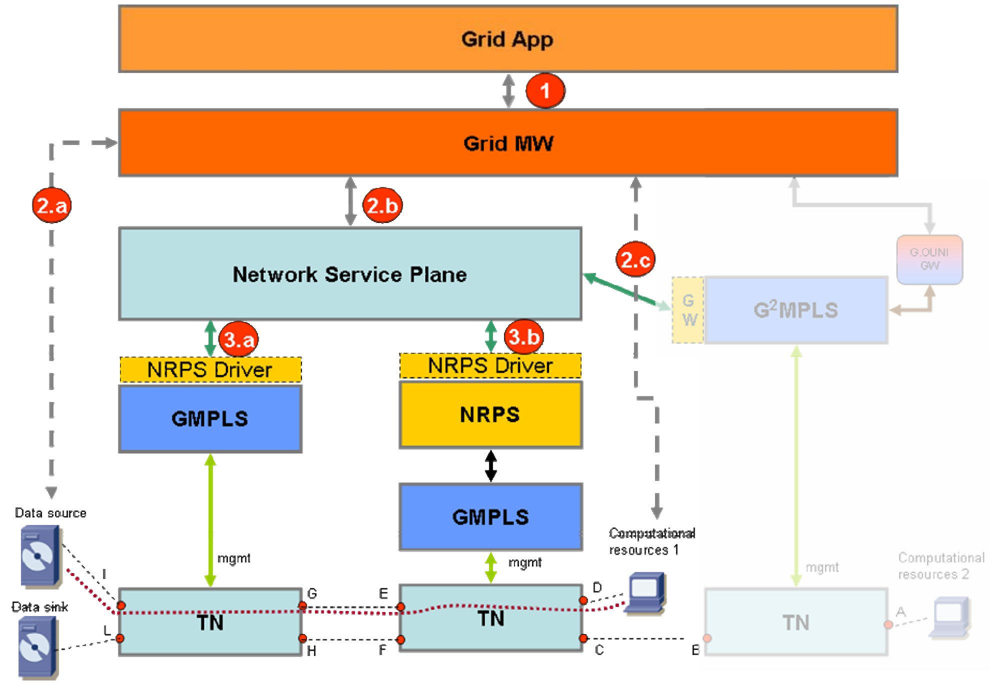


Figure 12-2: GMPLS interworking with NRPS through northbound interface.

In case of GMPLS interfaced directly to the NSP through an NRPS driver (step 3.a), the NRPS Driver on top of the GMPLS can be interpreted as a “thin” NRPS. The interface between the NRPS Driver and GMPLS is based on a kind of management interface. The Transport Network endpoints (G, H, I, L in Figure 12-2) can be interpreted as TNAs and will represent the sole topology knowledge (entered manually) available at the NRPS Driver. The transactions between GMPLS and its NRPS driver are limited to the creation/tear-down and monitoring of Soft Permanent Connections (SPC) into the GMPLS domain. GMPLS is responsible for the computation and signalling of explicit routes matching the request. GMPLS is not required to support advance reservation capabilities; the NRPS Driver checks advance reservation availability just at the external TNAs, under the assumption that a path will always exist in the GMPLS network at execution time. NRPS Driver issues instant connection requests to the underlying GMPLS just before the start time of each advance reservation. NRPS Driver will not be able to guarantee the success of its accepted advance reservations, because it cannot access any topology status and calendar information about the GMPLS domain and its TNAs. No E-NNI signalling is needed in this scenario, since all the inter-domain procedures run at the NSP. This scenario is proposed as an interim solution for interoperability in those domains (i.e. local test-beds) where neither an NRPS nor the G^2MPLS are available.

In case of GMPLS “slave” of a domain mastered by an NRPS (step 3.b), the NRPS will ask the underlying GMPLS Control Plane for network connections starting and ending in the GMPLS domain; therefore, no E-NNI signalling is needed since all inter-domain issues run at the NSP. In this scenario, advance reservations are implemented at the NRPS layer and it is assumed that NRPS owns solely all the transport resources configured by the GMPLS; moreover, GMPLS is used for configuration only by its NRPS.

The interface between the NRPS and GMPLS cannot rely on the standard OIF O-UNI specifications, because the NRPS must know full topology information (not limited to the endpoints) to provide effective advance reservations. Moreover, some mechanisms for communicating a strict explicit route from the NRPS to the GMPLS are needed in order to avoid the misalignment between advanced path computation & bookings at the NRPS and the independent path computation processes at the GMPLS. For this critical reason, the interface between the NRPS and GMPLS is more compliant with a Network Management Interface (NMI) than an O-UNI and case 3.b can be considered as a superset of case 3.a described above.

In the PHOSPHORUS WP1 framework, a proposal for a WS-based interface between NRPS and GMPLS has been provided [PHOSPHORUS-D1.1]. This interface is based on a simple request/response model that does not raise specific implementation issues for ARGON, UCLP and DRAC.

Two possible implementation models are foreseen for this WS interface, as depicted in Figure 12-3 and Figure 12-4.

In the first case (ref. Figure 12-3) a centralized approach is applied and the GMPLS WS servant is placed in the NRPS. It accesses the GMPLS domain via a GMPLS stub element, which is an instance of the GMPLS Control Plane un-correlated from any transport network element and aimed to simulate the calling and called signalling parties. This GMPLS stub element is accessed through an API, participates in the G.OSPF-TE domain and starts closed-loop signalling sessions for LSPs (i.e. starting from it and ending on it). Routing and signalling are both carried on via I-NNI. This solution is particularly effective for a narrow control of the Control Plane dynamics during the LSP signalling and operations (e.g. in case of recovery).

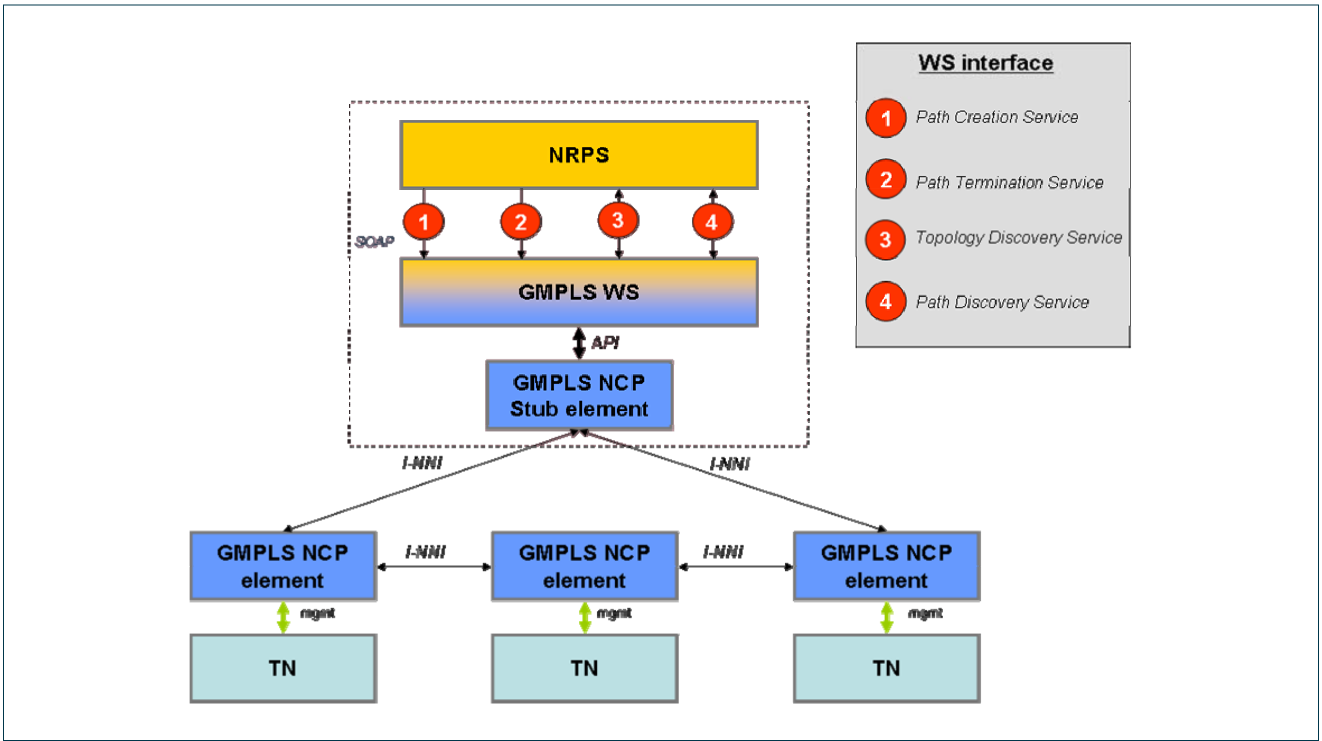


Figure 12-3: GMPLS-NRPS interworking via Web Services: centralized solution.

In the latter case (ref. Figure 12-4) a distributed approach is applied and the GMPLS WS servant is co-located with each edge GMPLS network element. Signalling and routing procedures among the GMPLS instances are implemented via I-NNI and the GMPLS WS servant accesses connection and topology services via an API. This solution replicates the GMPLS WS servants and requires specific procedures on these GMPLS edge elements for implementing the ingress and egress cross-connections.

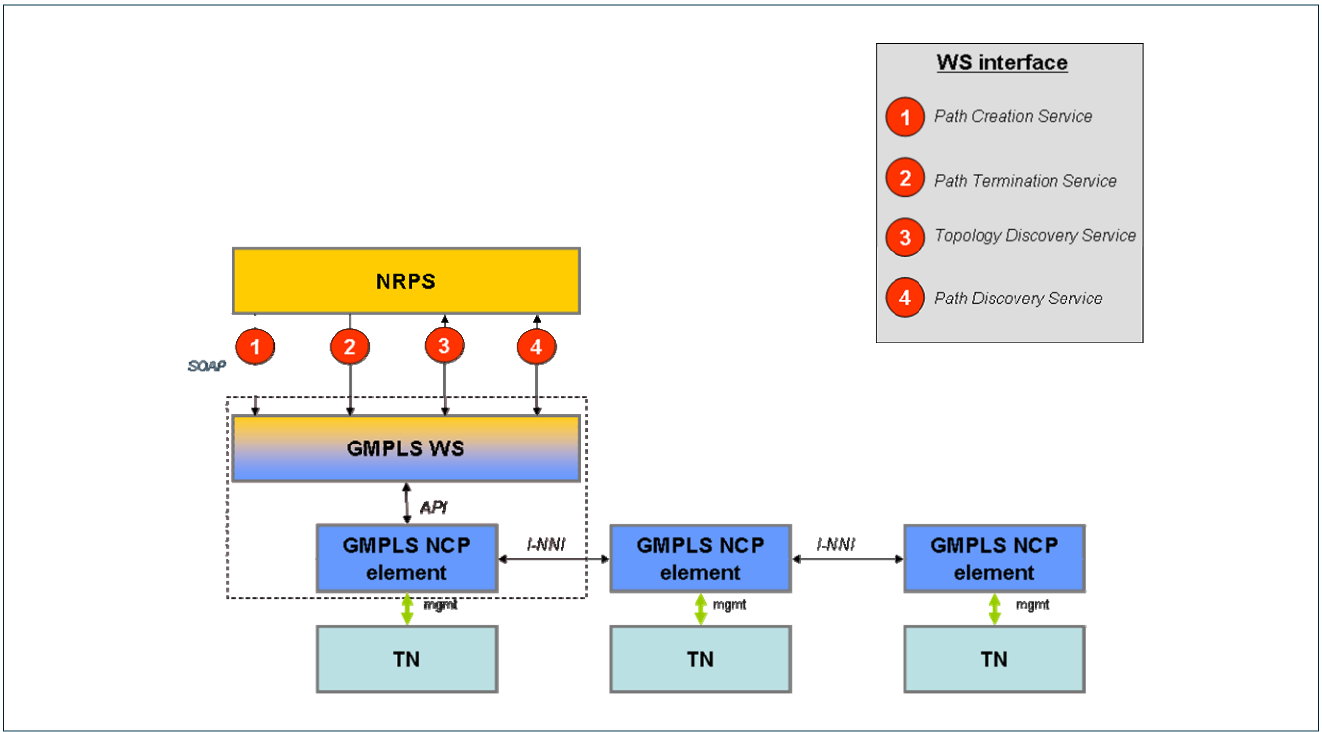


Figure 12-4: GMPLS-NRPS interworking via Web Services: distributed solution.

12.2 Interworking through east/west interfaces

In a G^2MPLS domain Grid and network resources are both under the same control and just peering relationships with other G^2MPLS domains or NRPS-based domains can be established (case C in Figure 12-1). For this reason, G^2MPLS is at the same level of the NSP and any interworking can happen through the G^2MPLS east/westbound interfaces. Due to the network scope limitation of the NSP, the G^2MPLS inter-domain transactions can support only network services. The completion of the Grid transactions is up to the Grid Middleware layer through dedicated communications.

In Figure 12-5 this option is described by means of a use case. At step 1, the Grid application submits a job request to the Grid Middleware, asking for the configuration of a Grid job between "Computational resource 2" and "Data Source", which includes also the configuration of the network resources between them. The Grid Middleware (e.g. through its MetaScheduling Service) starts localizing resources and negotiating their availabilities (step 2.a, 2.b and 2.c). Upon the occurrence of step 2.b, the G^2MPLS domain is activated for network service reservation and setup through the G.OUNI GW (step 3). G^2MPLS is responsible for selecting the inter-domain route and starting the advance reservation negotiation and establishment of the inter-domain connection. Once an inter-domain route for the required network service is computed and G^2MPLS signalling is progressed up to the domain egress point (B in Figure 12-5), G^2MPLS forwards the inter-domain network service request to the NSP (step 4), by specifying the next segment endpoints (from C to I in Figure 12-5). Then, NSP proceeds the network service setup according to the procedures defined [PHOSPHORUS-D1.1].

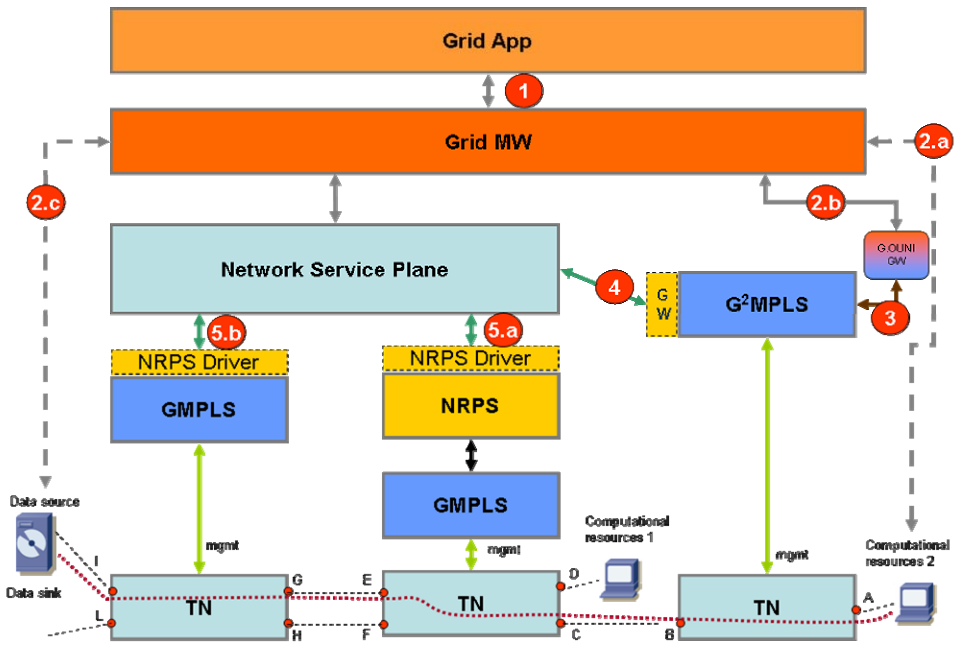


Figure 12-5: G2MPLS interworking with NSP through east-west interface.

13 Interworking of G^{2}MPLS and GÉANT2 BoD system

GÉANT2-JRA3 BoD system provides automatic inter-domain bandwidth reservation in heterogeneous environments through Inter-Domain Manager modules (IDM). Local domain management is secondary objective and is performed by a Domain Manager (DM) with the assistance of Technology Proxy (TP) modules, responsible for raw devices configuration. The BoD architecture allows to freely implement DM and TP accordingly to local domain technology and requirements, including SDH/SONET, Ethernet and GMPLS-controlled transport networks. The exhaustive API for Inter-Domain Manager module (IDM) is delivered to end-users for a full interfacing to the system.

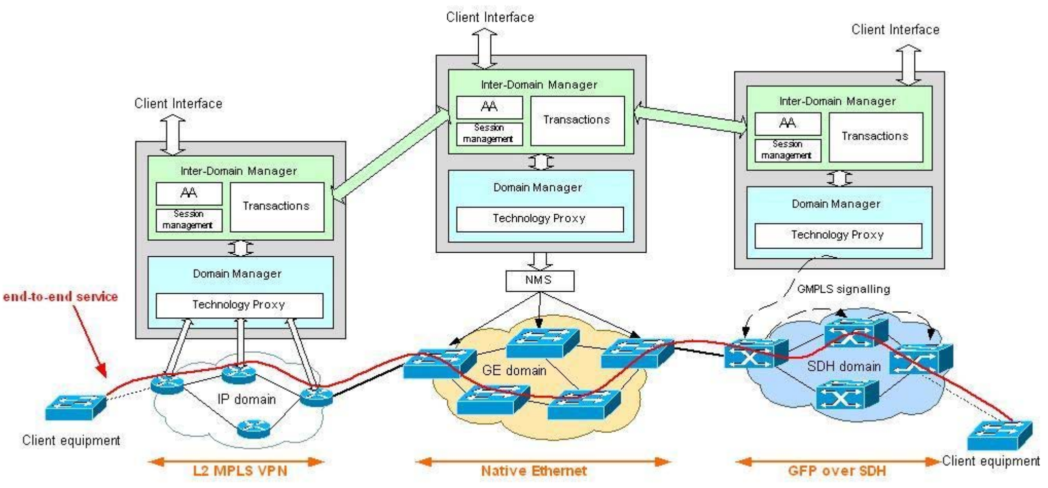


Figure 13-1: GÉANT2 BoD system *(courtesy of GÉANT2 project).*

The IDM has the knowledge of the inter-domain topology with various levels of details depending on the information published by the different domains. Upon receiving a BoD connection request, IDM manages the inter-domain connection setup by:

- interacting with the GN2 Authorization and Authentication Infrastructure (AAI), to authenticate the identity of BoD service requestors and authorize access to the BoD service
- computing an inter-domain path in its Pathfinder module
- issuing a local domain connection request to the underlying (and possibly pre-existent) Domain Manger-Technology Proxy
- forwarding the request for connection setup to the selected neighbouring IDM

Neither DM nor TP exceed domain boundaries to perform day-to-day operations, because only IDM is responsible for inter-domain message exchange and signalling for Control Plane.

This architecture breaks inter-domain connection and signalling down to a number of segments, one for each traversed domain. Therefore, it is similar to the ASON/GMPLS approach for establishing calls: the ingress and egress Network Call Controller roles are collapsed into the single IDM module.

Although GN2-JRA3 is aware of Grid environment requirements, current efforts do not involves any Middleware integration. However, reservation requests are submitted to IDM via Web Services interface, and they are conceived to be generated by both users and applications. The standard API does not include any Grid specific extensions to support G^2MPLS.

GN2 BoD system lies more in the Management Plane than in the Control Plane and even if it can comply with the requirement of automation of the inter-domain connection establishment phase, it cannot completely fulfil the requirements for connection reliability, which are the native realm of network Control Plane technologies such as GMPLS.

Although the interaction of GMPLS enabled networks with JRA3 system is not a primary objective for the GÉANT2 project, some cooperation scenarios can be taken into consideration with a similar approach as described in Sec. 12 for NRPS-es. The main problem with such an interaction is related to the inter-domain interface, which in GÉANT2 is WS-based for signalling and OSPF-TE based for routing, while in G^2MPLS is completely based on E-NNI protocols for both routing and signalling.

Two possible interoperation scenarios can be adopted to converge towards interoperability.

The first option, case A in Figure 13-2, is based on the northbound interface and requires usage of IDM instance above a GMPLS domain. In this case the IDM acts as an NRPS on top of GMPLS Control Plane and continues to be the sole responsible of the IDM-IDM communications.

The latter option, case B in Figure 13-2, is to use G^2MPLS as a peer of the IDM and to interwork through east-west interface. In this case an E-NNI proxy is needed in the IDM, in order to allow the BoD system to communicate directly to G^2MPLS. The limitation to network scope for the inter-domain G^2MPLS transactions apply also in this scenario, due to the unavailability of Grid information in IDM layer.

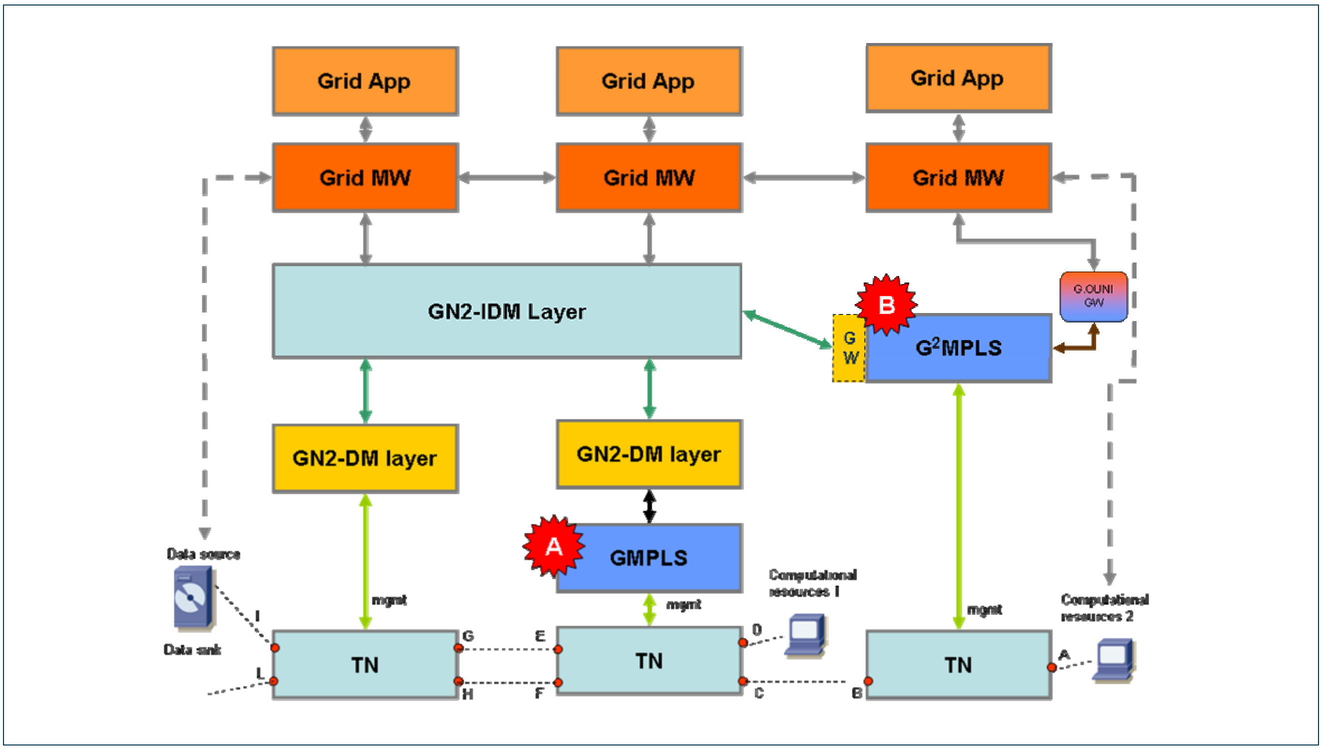


Figure 13-2: GMPLS/G^2MPLS interworking with GÉANT2 BoD system.

Further details are left for further studies and depend on the possible planning of joint demonstration activities between PHOSPHORUS and GÉANT2 projects.

14 Closing notes

This document describes the Grid-GMPLS Control Plane Architecture, part of which will be implemented and demonstrated by the PHOSPHORUS experimental activities on the test-bed.

The network and service models needed to implement the Grid Network Service concept have been presented in this document, and the general G^2MPLS functional architecture as well as specific G^2MPLS routing, signalling and recovery procedures have been defined. The interworking issues with Grid middleware, NRPS layer and GN2-BoD system have been considered and appropriate solutions for the coexistence of these technologies have been proposed. These solutions rely on the technical agreement between the PHOSPHORUS WP2 team and the other PHOSPHORUS research teams responsible for Grid middleware and NRPS. Formal liaison with GN2-JRA3 will assess the feasibility of the proposed interworking between G^2MPLS and the GN2 BoD system.

Specific analysis of G^2MPLS deployment scenarios in NRENs environments will be discussed in PHOSPHORUS deliverable D2.6. Specific protocol enhancements will be detailed in deliverable D2.2, while the G^2MPLS high level system design will be discussed in deliverable D2.3. Further details on the specification of the G^2MPLS interfaces will be provided through deliverable D2.7. The overall set of these deliverables will build the structure of the G^2MPLS Network Control Plane architecture.

This deliverable will drive the subsequent low-level design and developments of the G^2MPLS Control Plane.

15 References

15.1 Normative references

[ASON-ARCH]	ITU-T G.8080/Y.1304 Recommendations, “Architecture for the Automatic Switched Optical Network (ASON)”, 2001.
[ASON-DEF]	ITU-T G.8081/Y.1353 Recommendations, “Definitions and Terminology for Automatically Switched Optical Networks (ASON)”, 2004.
[ASON-REQ]	ITU-T G.807/Y.1302 Recommendations, “Requirements for the Automatic Switched Transport Network (ASTN)”, 2001.
[ASON-DCN]	ITU-T G. 7712/Y.1703 Recommendations, “Data Communication Network”, 2003.
[ASON-DCM]	ITU-T G.7713/Y.1704 Recommendations, “Generalised Distributed Connection Management”, 2001.
[ASON-PNNI]	ITU-T G.7713.1/Y.1704 Recommendations, “Distributed Call and Connection Management – PNNI Implementation”, 2003.
[ASON-RSVP]	ITU-T G.7713.2/Y.1704 Recommendations, “Distributed Call and Connection Management – GMPLS RSVP-TE Implementation”, 2003.
[ASON-LDP]	ITU-T G.7713.3/Y.1704 Recommendations, “Distributed Call and Connection Management – GMPLS CR-LDP Implementation”, 2003.
[ASON-AD]	ITU-T G.7714/Y.1705 Recommendations, “Generalised automatic discovery techniques”, 2001
[ASON-AD-SDH]	ITU-T G.7714.1/Y.1705.1 Recommendations, “Protocol for automatic discovery in SDH and OTN networks”, 2003.
[ASON-ROUT]	ITU-T G.7715/Y.1706 Recommendations, “Architecture and requirements for routing in automatically switched optical networks”, 2002.
[ITU-G.841]	ITU-T G.841, "Types and Characteristics of SDH Network Protection Architectures," 1998.
[ITU-G.808.1]	ITU-T G.808.1, "Generic Protection Switching – Linear trail and subnetwork protection," 2003.
[IETF-RFC3945]	E. Mannie (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Architecture”, IETF RFC 3945, October 2004.
[IETF-RFC3471]	L. Berger (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description”, IETF RFC 3471, January 2003.
[IETF-RFC3472]	P. Ashwood-Smith (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Constraint-based Routed Label Distribution Protocol (CR-LDP) Extensions”, IETF RFC 3472, January 2003.

[IETF-RFC3473] L. Berger (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions”, IETF RFC 3473, January 2003.

[IETF-RFC4257] G. Bernstein (Ed.), “Framework for Generalized Multi-Protocol Label Switching (GMPLS)-based Control of Synchronous Digital Hierarchy/Synchronous Optical Networking (SDH/SONET) Networks”, IETF RFC 4257, December 2005.

[IETF-RFC4606] E. Mannie (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Extensions for Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH) Control“, IETF RFC 4606, August 2006.

[IETF-RFC4328] D. Papadimitriou (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for G.709 Optical Transport Networks Control”, IETF RFC 4328, January 2006.

[IETF-RFC4139] D. Papadimitriou (Ed.), “Requirements for Generalized MPLS (GMPLS) Signaling Usage and Extensions for Automatically Switched Optical Network (ASON)”, IETF RFC 4139, July 2005.

[IETF-RFC4208] G. Swallow (Ed.), “Generalize Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model”, IETF RFC 4208, October 2005.

[IETF-RFC4201] K. Kompella, Y. Rekhter, L. Berger, “Link Bundling in MPLS Traffic Engineering (TE)”, IETF RFC 4201, October 2005.

[IETF-RFC4202] K. Kompella (Ed.), “Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)”, IETF RFC 4202, October 2005.

[IETF-RFC3630] K. Katz (Ed.), “Traffic Engineering (TE) Extensions to OSPF Version 2”, IETF RFC 3630, September 2003.

[IETF-RFC4203] K. Kompella (Ed.), “OSPF Extensions in Support of Generalized Multi-Protocol Label Switching”, IETF RFC 4203, October 2005.

[IETF-RFC4652] D. Papadimitriou (Ed.), “Evaluation of Existing Routing Protocols against Automatic Switched Optical Network (ASON) Routing Requirements”, IETF RFC 4652, October 2006.

[IETF-RFC4258] D. Brungard (Ed.), “Requirements for Generalized Multi-Protocol Label Switching (GMPLS) Routing for the Automatically Switched Optical Network (ASON)”, IETF RFC 4258, November 2005.

[IETF-RFC4204] J. Lang (Ed.), “Link Management Protocol (LMP)”, IETF RFC 4204, October 2005.

[IETF-RFC4207] J. Lang (Ed.), “Synchronous Optical Network (SONET)/Synchronous Digital Hierarchy (SDH) Encoding for Link Management Protocol (LMP) Test Messages”, IETF RFC 4207, October 2005.

[IETF-RFC4209] A. Fredette (Ed.), “Link Management Protocol (LMP) for Dense Wavelength Division Multiplexing (DWDM) Optical Line Systems”, IETF RFC 4209, October 2005.

[IETF-RFC4426] J. Lang (Ed.), “Generalized Multi-Protocol Label Switching (GMPLS) Recovery Functional Specification", IETF RFC 4426, March 2006.

[IETF-RFC4427] E. Mannie (Ed.), “Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS)”, IETF RFC 4427, March 2006.

[IETF-RFC4428] D. Papadimitriou (Ed.), “Analysis of Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery Mechanisms (including Protection and Restoration)”, IETF RFC 4428, March 2006.

[IETF-SEGREC] L. Berger, I. Bryskin, D. Papadimitriou, A. Farrel, "GMPLS Based Segment Recovery", IETF Draft, work in progress, draft-ietf-ccamp-gmpls-segment-recovery-03.txt, October 2006.

[IETF-IDREC] T. Takeda, Y. Ikejiri, A. Farrel, J.P. Vasseur, "Analysis of Inter-domain Label Switched Path (LSP) Recovery", IETF Draft, work in progress, draft-ietf-ccamp-inter-domain-recovery-analysis-00.txt, December 2006.

[IETF-E2ESIGREC] J.P. Lang (Ed.), "RSVP-TE Extensions in support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", IETF Draft, work in progress, draft-ietf-ccamp-gmpls-recovery-e2e-signaling-04.txt, October 2006.

[IETF-RFC4726] A. Farrel (Ed.), "A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering", IETF RFC 4726, November 2006.

[IETF-RFC4783] L. Berger (Ed.), "GMPLS - Communication of Alarm Information", IETF RFC 4783December 2006.

[IETF-RFC3477] K. Kompella, Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol – Traffic Engineering (RSVP-TE)", IETF RFC 3477, January 2003.

[IETF-RFC3480] K. Kompella, Y. Rekhter, A. Kullberg, "Signalling Unnumbered Links in CR-LDP (Constraint-Routing Label Distribution Protocol)", IETF RFC 3480, February 2003.

[IETF-RFC4655] A. Farrel J.-P. Vasseur, J.Ash, "A Path Computation Element (PCE) – Based Architecture", IETF RFC 4655, August 2006.

[OIF-UNI1.0] Optical Interworking Forum, "User Network Interface (UNI) 1.0 Signalling Specification", http://www.oiforum.com/public/documents/OIF-UNI-01.0.pdf.

[OIF-UNI1.0R2-COMM] Optical Interworking Forum, "UNI 1.0 Signaling Specification, Release 2: Common Part", http://www.oiforum.com/public/documents/OIF-UNI-01.0-R2-Common.pdf.

[OIF-UNI1.0R2-RSVP] Optical Interworking Forum, "RSVP Extensions for User Network Interface (UNI) 1.0 Signaling, Release 2", http://www.oiforum.com/public/documents/OIF-UNI-01.0-R2-RSVP.pdf.

[OIF-UNI2.0-COMM] Optical Interworking Forum, "UNI 2.0 Signaling Specification: Common Part", draft OIF2005.204.02, work in progress.

[OIF-UNI2.0-RSVP] Optical Interworking Forum, "RSVP Extensions for User Network Interface (UNI) 2.0 Signaling", draft OIF2005.205.01, work in progress.

[OIF-E-NNI-Sig-1.0] Optical Interworking Forum, "Intra-Carrier E-NNI 1.0 Signaling Specification", http://www.oiforum.com/public/documents/OIF-E-NNI-Sig-01.0-rev1.pdf.

[OIF-E-NNI-Sig-2.0] Optical Interworking Forum, "Intra-Carrier E-NNI 2.0 Signaling Specification", draft OIF2005.381.03, work in progress.

[OIF-E-NNI-Rtr-1.0] Optical Interworking Forum, "External Network-Network Interface (E-NNI) OSPF-based Routing - 1.0", http://www.oiforum.com/public/documents/OIF-ENNI-OSPF-01.0.pdf.

[OGF-GFD80] I. Foster, H. Kishimoto, A. Savva, D. Berry, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, J. Treadwell, J. Von Reich, "The Open Grid Services Architecture, Version 1.5", GFD-I.080 http://forge.gridforum.org/projects/ogsa-wg, July 2006.

[W3C-WDSL] Web Services Description Language (WSDL) 1.1, W3C, http://www.w3.org/TR/wsdl.

[UDDI] UDDI: Universal Description, Discover and Integration of Business for the Web, http://www.uddi.org.

[WSIL] "Web Services Inspection Language (WSIL), http://xml.coverpages.org/IBM-WS-Inspection-Overview.pdf.

[WSFL] Web Services Flow Language (WSFL)”, http://www-4.ibm.com/software/solutions/webservices/pdf/WSFL.pdf .

[GLUE] S. Andreozzi (Ed.), “GLUE Schema Specification – version 1.3”, Draft 3 – 16 Jan 2007, http://glueschema.forge.cnaf.infn.it/Spec/V13.

[JSDL] A. Savva (Ed.), “Job Submission Description Language (JSDL) Specification, Version 1.0”, GFD.056, http://forge.gridforum.org/projects/jsdl-wg, November 2005.

15.2 Informational references

[OGF-GNS] G. Clapp, T. Ferrari, D.B. Hoang, A. Jukan, M.J. Leese, P. Mealor, F. Travostino, “Grid Network Services”, draft-ggf-ghpn-netserv-2, Grid High Performance Networking Research Group working draft, May 2005.

[OGF-GFD81] J. Treadwell (Ed.), “Open Grid Services Architecture Glossary of Terms Version 1.5”, GFD-I.081, http://forge.gridforum.org/projects/ogsa-wg, July 2006.

[IETF-RFC4397] I. Bryskin (Ed.), “A Lexicography for the Interpretation of Generalized Multiprotocol Label Switching (GMPLS) Terminology within The Context of the ITU-T's Automatically Switched Optical Network (ASON) Architecture”, IETF RFC 4397, February 2006.

[VIOLA UBonn] Christoph Barz and Markus Pilz, “Co-Allocation of Compute and Network Resources in the VIOLA-Testbed”, TERENA 2006

[IETF-RFC3444] A. Pras, J. Schoenwaelder, “On the Difference between Information Models and Data Models”, IETF RFC 3444, January 2003

[GRID-PHYSIOLOGY] I. Foster, C. Kesselman, J. Nick, and S. Tuecke, “The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration”, Globus Project, 2002.

[PHOSPHORUS-D1.1] PHOSPHORUS Deliverable D.1.1, “Requirements and specifications of interfaces and architecture for interoperability between NRPS, GMPLS, Middleware”.

[GN2-IDM] GÉANT2 Deliverable DJ.3.3.1, “GÉANT2 Bandwidth on Demand Framework and General Architecture”.

[STRAND-01] J. Strand, A. Chiu, R. Tkach, “Issues For Routing In The Optical Layer”, IEEE Communications Magazine, February 2001, pp.81-87.

[OGF-GDF56] D. Simeonidou, R. Nejabati (Editors), “Optical Network Infrastructure for Grid”, GFD-I-036, http://forge.gridforum.org/projects/ghpn-rg/, August 2004.

[Agrawal] G.Agrawal, “Nonlinear Fiber Optics, Academic Press ,1995

16 Acronyms

AAA	Authentication, Authorisation, and Accounting
AAI	Authentication and Authorization Infrastructure
ANSI	American National Standards Institute
API	Application Programming Interface
ARGON	Allocation and Reservations in Grid-enabled Optical Networks
BB	Bandwidth Broker
BGRP	Border Gateway Reservation Protocol
BoD	Bandwidth on Demand
BR	Border Router
COPS	Common Open Policy Protocol
CORBA	Common Object Request Broker Architecture
CP	Control Plane
CPE	Customer Premises Equipment
DRAC	Dynamic Resource Allocation Controller
DVB	Digital Video Broadcasting
DWDM	Dense Wavelength Division Multiplexing
EGEE	Enabling Grids for E-sciencE
EC	European Commission
E-NNI	Exterior NNI
ETSI	European Telecommunications Standards Institute
EU	European Union
FCAPS	Fault, Configuration, Accounting, Performance, Security
G.OSPF-TE	GMPLS OSPF-TE
G.RSVP-TE	GMPLS RSVP-TE
G^{2}MPLS	Grid-GMPLS (enhancements to GMPLS for Grid support)
GE	Gigabit Ethernet
GÉANT	Pan-European Gigabit Research Network
GGF	Global Grid Forum
GHPN	Grid High Performance Networking
GMPLS	Generalized MPLS
G-OUNI	Grid OUNI

GSMP	General Switch Management Protocol
HW	Hardware
IDM	GÉANT2 Inter-domain Manager
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
I-NNI	Interior NNI
IP	Internet Protocol
IPR	Intellectual Property Right
IPSec	IP security
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ITU	International Telecommunication Union
LAN	Local Area Network
LDP	Label Distribution Protocol
LRMS	Local Resource Management System
LSP	Label Switched Path
LSR	Label Switch Router
MAC	Media Access Control
MAN	Metropolitan Area Network
MP	Management Plane
MPLS	Multi Protocol Label Switching
NMS	Network Management System
NNI	Network to Network Interface
NO	Network Operator
NREN	National Research and Education Network
NRPS	Network Resource Provisioning Systems
NSP	Network Service Plane
OAM	Operations, Administration and Maintenance
OGF	Open Grid Forum
OIF	Optical Internetworking Forum
OSPF	Open Shortest Path First protocol
OSPF-TE	OSPF with Traffic Engineering extensions
O-UNI	Optical UNI
P2MP	Point to Multi Point
PON	Passive Optical Network
QoS	Quality of Service
RSVP	Resource reSerVation Protocol
RSVP-TE	RSVP with Traffic Engineering extensions
RTP	Real-time Transport Protocol
SDO	Standard Developing Organizations
SLA	Service Level Agreement
SLS	Service Level Specification

SME	Small and Medium Enterprise
SNMP	Simple Network Management Protocol
SP	Service Provider
SW	Software
TE	Traffic Engineering
TGC	Trusted Computing Group
TL-1	Transaction Language 1
TLS	Transport Layer Security
TLV	Type-Length-Value protocol fields
TMF	Tele Management Forum
TO	Telecom Operator
TP	Transport Plane
UCLP	User-Controlled Lightpath Provisioning system
UNI	User to Network Interface
VLAN	Virtual LAN
VPN	Virtual Private Network
WAN	Wide Area Network
WP	Work Package
WS	Web Service

Appendix A Additional details on the G^2MPLS Architecture

A.1 Overall G^2MPLS information model

In this section the G^2MPLS information model is defined, obtained as an extension of the ASON/GMPLS one (ref. pink blocks in Figure 16-1).

The main element in this information model for GNS purposes is the GNS transaction.

A GNS transaction is a set of network calls deriving from the same job specification.

The GNS transaction is related to the definition of Job as per GFD.81, i.e. a manageable resource with endpoint references and managed by a job manager.

The introduction of the GNS transaction concept allows to fit the typical Grid application use-cases in which multiple connections between different end-points are requested for the execution of the unique job (e.g. refer to Kodavis application use case). The GNS transaction is the container that provides a common root to different network call/connections traversing the same G^2MPLS User to Network reference point. It is shareable in a distributed way in the G^2MPLS NCP, in order to enable different invocation models.

The network call is an extension of the ASON call (an association between two or more users and one or more domains that supports an instance of a service through one or more domains) with further attributes such as temporal specifications.

All the other elements of the model are aligned with the ASON/GMPLS ones and share the same definitions and properties.

The call segment is the component element to be stitched in order to form the end-to-end call. Call segment is domain scoped, i.e. it is defined in terms of ingress and egress domain Access Point, QoS breakdown for that domain and recovery behaviour.

Each call segment contains a number of GMPLS LSPs, which are specified in terms of explicit route proposed (ERO) and recorded (RRO) by signalling. LSP are a collection of nodes, TE-links link components and labels that are part of the routing section of the information model.

Each GMPLS network is a collection network domains, identified by their Routing Controller ID and domain capabilities.

Each network domain contains a number of network nodes with the typical identifiers (Router ID) and capabilities. For SCN purposes, the network node contains also the set of control interfaces exposed to the DCN that allow the protocols' operations.

Each network node contains the set of established routing adjacencies, identified mainly by the remote network node identifier..

Each adjacency contains a number of TE-links, specified trough their TE parameters, and each TE-link contains a set of data links (or component links). In the case of multiple data links under the same TE-link the bundling procedures and limitations must be applied [IETF-RFC4201]. Each TE-link is related to a set of available control channels (ref. LMP) used for binding the Control Plane to the Transport Plane

Each data link contains a set of labels, i.e. the final transport resource unit, whose format is technology dependent.

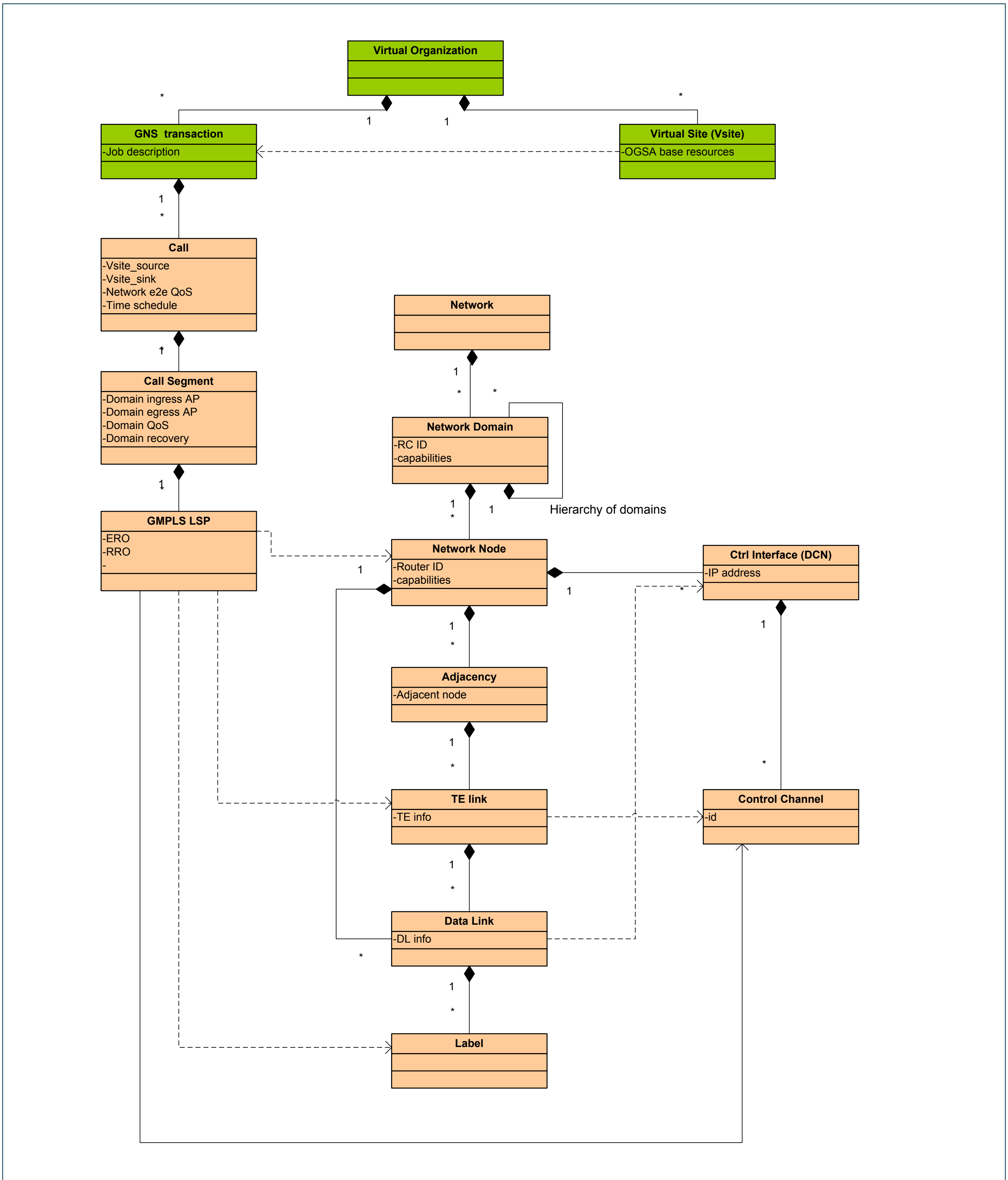


Figure 16-1: G^{2}MPLS information model.

A.2 Address and identifiers spaces

Identifiers and addresses provide the necessary means of identification for Control Plane components and resources.

According to the separation between Control Plane and Transport Plane different spaces can be identified in the G^2MPLS architecture [ASON-ARCH, IETF-RFC3945] as shown in the following Table 16-1.

Space	Description
SCN address space	It contains the unique and global addresses for the SCN interfaces used for the communications among the G^2MPLS Control Plane instances.
Control Plane identifier space	It contains the identifiers (addresses or names) for the components of the Control Plane, being them either abstract functionalities (e.g. G^2-CCC, G^2-RC) or protocol-based functionalities (e.g. G^2.OSPF-TE Router ID).
Transport Plane identifier space	It contains the identifiers for all the transport resources in order to allow resource identification for routing and signalling purposes.

Table 16-1: G^2MPLS address and identifier spaces.

All the network elements that are part of the G^2MPLS Control Plane must have access to the SCN for protocol message exchange; this implies having an interface on SCN and, consequently, a globally unique address from its space. Originating from GMPLS architecture, G^2MPLS is based on the IP addressing model for its network components. Therefore, IPv4 and/or IPv6 addresses are used to identify G^2MPLS network elements. This rule applies to G^2-LSRs, G^2-LERs and those elements that operate entirely in the Control Plane (e.g. PCE-based G^2MPLS routing controllers).

Most of the functional entities of the G^2MPLS Control Plane described in section 9 must be identifiable and/or addressable for their correct operations. They pick an identifier from the Control Plane identifiers space. A mapping of the G^2MPLS functional entities with the corresponding identifiers (if any) is provided in Table 16-2.

Functional entity	Identifier	Space	Type	Scope	Status
GNS Transaction and G^2MPLS Network Call Controller (G^2-CC)	A name that identifies it to peers for the purpose of maintaining G^2-call signalling relationships	Control Plane identifiers	IPv4-like address	Globally unique	Optional
G^2MPLS LSP Controller (G^2-LC)	A name that identifies it to peers for the purpose of maintaining G^2-LSP signalling relationships	Control Plane identifiers	IPv4-like address	Unique within administrative domain	Optional
G^2MPLS Routing Controller (G^2-RC)	A name used to identify the entity that creates the routing announcement.	Control Plane identifiers	IPv4-like address	Globally unique	Mandatory
TE-link Manager (TELM)	N/A				
GNS Service Discovery Agent (G-SDA)	N/A				

Functional entity	Identifier	Space	Type	Scope	Status
Discovery Agent (DA)	N/A				
Transport Resource Controller (TRC)	N/A				
Signalling Protocol Controller (SPC)	An SCN address for exchanging signalling protocol messages (call and LSP cases).	SCN addresses	IP address	Unique within administrative domain	Mandatory
Routing Protocol Controller (RPC)	A name that identifies the speaking node to peers for the purpose of exchanging routing protocol relationships (e.g. OSPF Router ID)	Control Plane identifiers	IPv4-like address	Globally unique	Mandatory
	An SCN address for exchanging routing protocol messages.	SCN addresses	IP address	Globally unique	Mandatory
Link management Protocol Controller (LPC)	An SCN address for each interface used to exchange link management protocol messages.	SCN addresses	IP address	Globally unique	Mandatory

Table 16-2: G^{2}MPLS functional entities and their identifiers.

The Transport Plane resources managed by the G^{2}MPLS Control Plane are identified according to the standard ASON/GMPLS rules and practices, including bundling of resources and unnumbered identifiers [IETF-RFC4201, IETF-RFC3477, IETF-RFC3480]. A summary of the transport resources identified in the G^{2}MPLS information model and their identifiers is provided in Table 16-3.

Transport Plane resource	Identifier	Type	Scope
Control Channel	A value that identifies the control channel in the set of control channels of a node.	Unnumbered	Node local
Network Domain	A value that identifies the domain in a routing hierarchy (e.g. the OSPF Area ID)	IPv4-like address	Globally unique
Network Node	A value that identifies the node belonging to a domain in a routing hierarchy (e.g. the OSPF Router ID)	IPv4-like address	Unique within administrative domain
Adjacency	A value that identifies the remote node sharing the same set of TE-links with this network node.	IPv4-like addresses	Unique within administrative domain
G.OUNI TE-link (TNA)	A value that identifies the Transport Network Assigned address used to identify globally the TE-link between a user node and an ingress node of the transport network.	IPv4 address, IPv6 address, NSAP address	Globally unique

Transport Plane resource	Identifier	Type	Scope
G.E-NNI TE-link	A value that identifies a bundle of Data Links interconnecting two border NEs that belong to neighbouring domains (inter-domain TE-link) or summarizing a reachability between two border NEs that belong to the same domain (intra-domain TE-link).	Unnumbered	Node Local, but the tuple <Domain-Node ID-TE-link> must be globally unique
G.I-NNI TE-link	A value that identifies a bundle of Data Links grouped/mapped in order to improve the scalability of the information flooding.	IP address, unnumbered	Unique within administrative domain in case of IPv4/IPv6. Node local if unnumbered.
Data link	A value that identifies a port containing a number of labels that can be unambiguously identified.	IP address, unnumbered	Unique within administrative domain in case of IPv4/IPv6. Node local if unnumbered.
Label	A value that identify a transport network elementary resource (e.g. a time-slot, a wavelength, etc.). It is not used to identify the "class" to which the label belongs, which is implicit in the multiplexing capabilities of the Data Link on which the label is used.	Unnumbered (32-bit)	Node

Table 16-3: G^{2}MPLS transport resources identifiers.

The possible IPv4/IPv6 format of some addresses does not imply that these addresses need to belong to the same addressing space than public IPv4/IPv6 addresses used for the Internet. Private IP addressing can be used when no exchange with any other operator is needed.

A.3 Coexistence of GMPLS and G^{2}MPLS Control Planes

The G^{2}MPLS Control Plane derives from the GMPLS architecture and has been defined as a superset of the original GMPLS functions and procedures in support of the specific functionalities required by Grids.

The study of the relationships between G^{2}MPLS and GMPLS domains is an interesting use-case, from which an evaluation of the achievable levels of interworking in signalling and routing procedures can be derived. Moreover, the study of the coexistence of the two Control Plane solutions can contribute to the definition of the steps for the introduction of G^2MPLS functionality in an operational NREN infrastructure, both in an intra-domain context and in the more challenging inter-domain context.

Being this issue strictly related to the deployment models and solutions of the G^2MPLS Control Plane, it will be covered by the companion deliverable D2.6.