G^2MPLS in a nutshell

Nicola Ciulli, Gino Carrozzo
Nextworks
ASON/GMPLS as a promising starting point

- ASON/GMPLS is an interesting solution in support of dynamic circuit provisioning for advanced research high-end applications
  - Distributed (either in signalling or routing, or both)
  - Faster restoration
  - Scalable solution
  - Built on a lot of lesson learnt from tlc history, both in terms of protocols (e.g. MPLS RSVP, OSPF) and architectures (e.g. ATM P-NNI)
  - Strong standardization and industry support
    - Standardized UNI and E-NNI interfaces (as a result of OIF IAs)
  - Flexible enough to migrate to new architectures / protocols
Major limitations in the near future

- ASON is for SDH and G.709, whereas the optical new trends are WSONs and Ethernet switching services.
- The OIF UNI is still a user-network to provider-network interface, not an application-to-network interface.
- And it does not allow routing info (e.g. endpoints info) to flow across.
- Applications have relationships beyond the single domain; the OIF E-NNI is just an inter-vendor i/f (not inter-carrier).
- No support for scheduled circuit reservations.
- Point-to-multipoint e2e circuits are not clearly supported.
- An application might want to interconnect remote endpoints between them, not just itself with some other endpoint.
- Applications might want to specify endpoints not just in terms of network names/addresses.
G²MPLS main features

- Lambda Switching Capable, with some WSON extensions
- G.UNI gateways to allow applications to “drive” its reference G.UNI-C and get dedicated dynamic circuits
- Support for application-specific information across the Control Plane
  - I-NNI, E-NNI and UNI routing extensions to distribute application node resources (*grid*)
  - I-NNI, E-NNI and UNI signalling extensions, for both Call and LSP signalling
- Support for advance reservations (in a distributed way)
- G.E-NNI as an inter-carrier i/f
  - Integration with AuthN/AuthZ frameworks
- Anycast circuits, i.e. “connect this local TNA with this amount of app (*grid*) resources”
- Indirect calls, i.e. “connect this TNA / amount of app (*grid*) resources (maybe elsewhere) to this TNA / amount of app (*grid*) resources (anywhere)”
What is G²MPLS?

- G²MPLS is ...
  - a Network Control Plane architecture that implements the concept of **Grid Network Services**
    - *GNS is a service that allows the provisioning of network and Grid resources in a single-step, through a set of seamlessly integrated procedures.*
  - expected to expose **interfaces specific for Grid services**
  - made of a set of **extensions to the standard GMPLS protocols**
    - provide enhanced network and Grid services for “power” users / apps (the Grids)
  - Basically, a superset of ASON/GMPLS
  - A set of Controller prototypes developed in Phosphorus WP2
    - Supports ADVA FSP300RE-II ROADM, Calient DiamondWave FiberConnect
    - Planned demonstrations at SC’08 and ICT-2008
    - Final prototype planned for public delivery Jan/Feb 2009

- G²MPLS is not ...
  - an application-specific architecture; it aims to
    - support **any kind of end-user applications** by providing network transport services and procedures that can fall back to the standard GMPLS ones
    - provide automatic setup and resiliency of network connections for “standard” users
G²MPLS positioning w.r.t. standards [1]

- OIF Network Interfaces (UNI and E-NNI)
- IETF (CCAMP) Protocol Controllers + some architectural aspects
- ASON architecture
G²MPLS positioning w.r.t. standards [2]
Why $G^2$MPLS?

- **uniform interface for the Grid-user** to trigger Grid & network resource actions
- **single-step provisioning** of Grid and network resources (w.r.t. the dual approach Grid brokers + NRPS-es)
- adoption of **well-established procedures** for traffic engineering, resiliency and crankback
- possible **integration of Grids in operational/commercial networks**, by overcoming the limitation of Grids operating on dedicated, stand-alone network infrastructures

Grid nodes can be modelled as network nodes with node-level grid resources to be advertised and configured (this is a native task for GMPLS CP)
G²MPLS functional scope

- G²MPLS will provide part of the functionalities related to the selection and co-allocation of both Grid and network resources.

- Co-allocation functionalities
  - **Discovery and Advertisement** of Grid + network capabilities and resources of the participating virtual sites (GridSites)
  - **Service setup / teardown**
    - *coordination* with local job scheduler in middleware
    - *configuration* of the involved network connections among the participating GridSites
      
      *(The network end-point – TNA – might not be specified, if Grid resources are specified)*
    - *resiliency* mgmt for the installed network connections and possible recovery escalation to the Grid MW for job recovery
    - *advanced reservations* of Grid and network resources
  - **Service monitoring**
    - retrieving the status of a job *(Grid transaction)* and of the related network connections
G²MPLS steps beyond state-of-the-art

- w.r.t. ASON/GMPLS
  - OTN extensions (in sig. and routing)
  - Grid extensions (in sig. and routing)
  - support for additional procedures (advanced reservation, anycasting, Grid + network coordinated recovery, “grid” crankback, etc.)
  - Grid-aware interfaces (G.OUNI, G.E-NNI)

- w.r.t. NRPS-es (UCLP, DRAC, ARGON)
  - native multi-domain and multi-tech support in a single CP
  - seamless escalation of recovery procedures, possibly with the MW
  - uniformed user-to-network transactions (G.OUNI), not dependent on specific Grid MW

- w.r.t. GN2 BoD system
  - a control plane solution instead of a mgmt one for BoD services
  - support of “power” users (Grids) and one-step co-allocation of Grid and network

- w.r.t. other projects in the GMPLS field (e.g. NSF-DRAGON, IST-MUPBED)
  - GNS compliancy
  - direct interface with Grid MW and peering with NRPS/IDM
  - demonstration of seamless coexistence in a structured distributed test-bed
Architectural insight
G²MPLS Control Plane models [1]

- Two models for the layering between Grid and Network resources

**G²MPLS overlay model**

**G²MPLS integrated model**

different scope with respect to the IETF GMPLS Overlay & Peer
**G²MPLS Control Plane models [2]**

- **G²MPLS Overlay**
  - Grid scheduler:
    - traditional configuration / monitoring of Grid resource
    - configuration / monitoring of network resource
  - **G²MPLS**
    - e2e information bearer for network and Grid resources information
    - configuration of just network service

- **G²MPLS Integrated**
  - **G²MPLS**
    - e2e information bearer for network and Grid resources information
    - selection of the job segment providers (Grid and network)
    - co-allocation of Grid and network
  - **Grid scheduler:**
    - traditional workflow and information services
G²MPLS and its neighbourhood

Inter-domain layer

AAA, [ID TE routing]
ID setup/tear-down

G.OUNI, G.E-NNI

East-westbound interfaces
Northbound Interface
Southbound Interface
Network Management Interface
G²MPLS Overlay: routing vision

Grid Scheduler topological view

Domain A

Domain B

Grid Site A

G²MPLS domain A

G²MPLS domain B

Grid Site B
G²MPLS Integrated: routing vision
GNS transaction is a set of network calls deriving from the same job specification.
- related to the definition of a job [GFD.81], i.e. a manageable resource with endpoint references and managed by a job manager
- needed to manage multiple calls between different end-points, requested for the execution of the unique job (e.g. refer to compute-store-visualize)
- shareable in a distributed way in the NCP, in order to enable different invocation models

Network call is an extension of the ASON call with further attributes such as temporal specifications
G²MPLS main use cases: Loopback GNS

<table>
<thead>
<tr>
<th>Use case</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>UC#1</td>
</tr>
<tr>
<td>Name</td>
<td>Loopback GNS transaction</td>
</tr>
<tr>
<td>Goal</td>
<td>A Grid User configures and requests the execution of a job that involves Grid resources located only at his local GridSite</td>
</tr>
<tr>
<td>Actors</td>
<td>Grid User at GridSite A</td>
</tr>
<tr>
<td>System components</td>
<td>GridSite A</td>
</tr>
<tr>
<td></td>
<td>- Client Application</td>
</tr>
<tr>
<td></td>
<td>- Grid middleware</td>
</tr>
<tr>
<td></td>
<td>- Local Grid resources</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Grid Middleware (local Grid scheduler in particular) is aware of the types and amount of Grid resources located at its GridSite A.</td>
</tr>
<tr>
<td>Postconditions</td>
<td>Grid Middleware (local Grid scheduler in particular) advertise the new status/availability of the resources under their control.</td>
</tr>
<tr>
<td>Trigger</td>
<td>The Grid User defines the job in the Client Application and issues the request.</td>
</tr>
</tbody>
</table>
### G²MPLS main use cases: GNS – direct invocation

<table>
<thead>
<tr>
<th>Use case</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number</strong></td>
<td>UC#2</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>GNS transaction – Direct Invocation</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>A Grid User configures and requests the execution of a job that involves Grid resources located at his local GridSite A and at GridSite B</td>
</tr>
<tr>
<td><strong>Actors</strong></td>
<td>Grid User at GridSite A</td>
</tr>
</tbody>
</table>

#### System components

- **GridSite A**
  - Client Application
  - Grid middleware
  - Local Grid resources
  - G.OUNI gateway
- **GridSite B**
  - Grid middleware
  - Local Grid resources
  - G.OUNI gateway
- G²MPLS network control plane
  - G².LERs with G.OUNI-N functionality
  - G².LSRs

#### Preconditions

Grid Middlewares at GridSite A and GridSite 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.
## G²MPLS main use cases: GNS – indirect invocation

<table>
<thead>
<tr>
<th>Use case</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>UC#3</td>
</tr>
<tr>
<td>Name</td>
<td>GNS transaction – Indirect Invocation</td>
</tr>
<tr>
<td>Goal</td>
<td>A Grid User at GridSite C configures and requests the execution of a job that involves Grid resources located remotely at GridSite A and GridSite B</td>
</tr>
<tr>
<td>Actors</td>
<td>Grid User at GridSite C</td>
</tr>
</tbody>
</table>
| System components | GridSite A, GridSiteB  
  - Grid middleware  
  - Local Grid resources  
  - G.OUNI gateway  
  GridSite C  
  - Client Application  
  - Grid middleware  
  - G.OUNI gateway  
  G²MPLS network control plane  
  - G².LERs with G.OUNI-N functionality  
  - G².LSRs |
| Preconditions | Grid Middlewares at GridSite A, GridSite B and GridSite 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. |
Example of direct invocation [1]
Example of direct invocation [2]
Example of direct invocation [3]
## G²MPLS extensions to GMPLS

<table>
<thead>
<tr>
<th>Services at reference point</th>
<th>ASON / GMPLS</th>
<th>G²MPLS Overlay</th>
<th>G²MPLS Integrated</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>G.I-NNI</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Routing</strong></td>
<td>(TE) topology resource publication and discovery</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Opaque piggybacking of discovered Grid resources</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td><strong>Signalling</strong></td>
<td>Connection (LSP) setup/tear-down/crankback</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Connection (LSP) status inquire/notification</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Opaque piggybacking of Grid job (JSDL) data</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Handling of Grid job (JSDL) data (anycast GNS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>G.OUNI</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Routing</strong></td>
<td>Grid resources publication and discovery</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td><strong>Signalling</strong></td>
<td>NS (call) setup/tear-down</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>NS (call) status inquire/notification</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Transparent handling of Grid job (JSDL) data</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>(Advance) GNS setup/tear-down</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Advance) GNS status inquire/notification</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>G.E-NNI</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Routing</strong></td>
<td>Network (TE) inter-domain resources publication and discovery</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Inter-domain TE information feed-up/feed-down</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Grid inter-domain resources publication and discovery</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Inter-domain Grid information feed-up/feed-down</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td><strong>Signalling</strong></td>
<td>NS (call) setup/tear-down</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>NS (call) status inquire/notification</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Handling of Grid job (JSDL) data (anycast GNS)</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>(Advance) GNS setup/tear-down</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Advance) GNS status inquire/notification</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Grid resource modelling in G²MPLS routing

Mapping from GLUE schema v1.3

- Site
  - Service
    - Service Data
  - Cluster
    - Computing Element
      - VOview
      - Job
    - Sub-Cluster
      - Software
      - Host
      - Software Data
  - Storage Element
    - Access Protocol
    - Control Protocol
    - Storage Area
      - VOview
G\(^2\) extensions to OSPF @ G.I-NNI & G.UNI

- **Network-related extensions**
  - new sub-TLVs for optical constraints in Link TLV in the TE LSA

- **Grid-related extensions**
  - new opaque LSA: **Grid LSA**
  - + top-level TLVs in it
  - + sub-TLVs in each of them
G²MPLS routing extensions: G.E-NNI considerations

- based on the standard OIF E-NNI OSPF extensions + G².OSPF-TE extensions
- G²MPLS extensions for TE-link top-level TLV → optional parameters for both inter-domain and intra-domain TE links
- Grid Site TLV may be filtered (site ID is contained in other top-level TLVs)
- Grid Sub-Cluster and Storage Element TLVs natively summarize information related to the Grid site
  - to publish a minimum homogeneous bundled set is a configuration policy
- Grid Service and Computing Element TLVs could/should be summarized during the feed-up at RCs
  - Strategy: bundle homogeneous mandatory information (e.g. LRMS Info for Computing Element, Service Info for Service TLVs)
Immediate GNS transaction setup

- **G.OUNI**
- **G.I-NNI**
- **G.E-NNI**

**Flow:**
- **G.OUNI**: Path
- **G.OUNI**: Resv
- **G.OUNI**: ResvConfirm
- **G.I-NNI**: Path
- **G.I-NNI**: Resv
- **G.I-NNI**: ResvConfirm
- **G.E-NNI**: Path
- **G.E-NNI**: Resv
- **G.E-NNI**: ResvConfirm
Advance reservation of GNS transaction

- G.OUNI
- G.I-NNI
- G.E-NNI

**Negotiation and resource booking**

**Activation**

- G.OUNI: Path
- G.I-NNI: Path
- G.E-NNI: Path

- G.OUNI: Resv
- G.I-NNI: Resv
- G.E-NNI: Resv

- G.OUNI: ResvConfirm
- G.I-NNI: ResvConfirm
- G.E-NNI: ResvConfirm

- G.OUNI: Path Activate
- G.I-NNI: Path Activate
- G.E-NNI: Path Activate
G²MPLS controller functional decomposition

- **NBI**
  - Mant. Protocol Controller (G²-NCC)
  - G²MPLS LSP Controller (G²-LC)
  - G²MPLS Routing Controller (G²-RC)
  - GNS Transaction and G²MPLS Call Network Controller (G²-NCC)

- **GNS Transaction**
  - G²-LSP request
  - G²-LSP action

- **TE-link Manager (TEM)**
  - Label requests
  - Topology queries

- **Transport Resource Controller (TRC)**
  - Label state
  - Link info

- **GNS Service Discovery Agent (G-SDA)**
  - Coordination
  - GNS caps

- **Discovery Agent (DA)**
  - Coordination
  - Peer's resources

- **Transport Plane**

- **Link Layer**
  - G.OUNI
  - G.E-NNI
  - G.I-NNI

- **Grid & Network Topology**
  - Peer's GNS caps

- **Network Interface**
  - Transport Plane
G²MPLS functional components [1]

- GNS Transaction and G2MPLS Call Controller (G²-CC)
  - control and management of both GNS Transactions and the related G²MPLS Calls
  - 2 types: calling/called party G²-CC (G²-CCC), network G²-CC (G²-NCC)

- G²MPLS LSP Controller (G²-LC)
  - management of each Label Switched Path (LSP) that is part of a G²MPLS call

- G²MPLS Routing Controller (G²-RC)
  - stores an updated topology view of Grid and network resource
  - uses the topology for the computation of paths upon a request from G²-LC

- TE-link Manager (TEM)
  - 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
G²MPLS functional components [2]

- GNS Service Discovery Agent (G-SDA)
  - discovers Grid and network capabilities between a GridSite attached to the G.OUNI (client side) and the G²MPLS NCP (network side)

- Discovery Agent (DA)
  - discovers network resources (i.e. by correlation and verification) in the G²MPLS NCP

- Transport Resource Controller (TRC)
  - abstracting the technology specific details of the switching resources for NCP

- Protocol Controllers (PC)
  - maps the contents and actions of the interfaces into objects and messages of the G²MPLS protocol instances
    - 3 types
      - Signalling (SPC) → G².RSPV-TE
      - Routing (RPC) → G².OSPF-TE
      - Link management (LPC) → LMP (G.LMP?)
Most of the G²MPLS protocols start from the GNU QUAGGA routing suite v0.99.7 (http://www.quagga.net/)
Some info about developments

- Based on Quagga stack
  - with some non-Quagga protocols
  - Internal CORBA (omniORB) middleware

- Migration of QUAGGA to GMPLS
  - “Skeleton” GMPLS stack components
    - Link Resource Manager (per ASON: a manager of TE links, data links and labels)
    - SCN Gateway client/server (to support the decoupling of TN and SCN)
    - cross-protocol utilities identified and implemented
    - a tool for the automated code generation for Finite State Machines
  - Migration of QUAGGA protocols to GMPLS
    - GMPLS extensions to the QUAGGA OSPFv2 daemon
    - From-scratch implementation of a GMPLS RSVP-TE integrated in QUAGGA

- Grid extensions to GMPLS routing and signalling protocols
  - Grid extensions to QUAGGA GMPLS OSPFv2 daemon
  - Grid extensions to the Phosphorus GMPLS RSVP-TE

- Grid Optical User Network Interface (G.OUNI) implementation
  - G.OUNI (both –N and –C) version of the Phosphorus GMPLS RSVP-TE
  - Interfacing to middleware

- Call Controller and Recovery Controller
G²MPLS controllers

- A preliminary prototype publicly delivered as XEN VM (http://www.ist-phosphorus.eu/deliverables.php, ref. D2.5)
  - Supports ADVA FSP300RE-II ROADM, Calient DiamondWave FiberConnect
- Demonstrated at SC’08 and ICT-2008
- Final prototype scheduled for public delivery Jan 2009
Deployment of GMPLS at NRENs
A proposal for GMPLS deployment in NRENs

- User network
  - UNI
  - UNI
  - User network
  - UNI
  - UNI

- Region/Layer A.1.1
  - I-NNI
  - NREN 1
  - Area A.1
  - Region/Layer A.1.2
    - I-NNI
    - Inter-Carrier Layer
    - E-NNI

- Region/Layer B.1.1
  - I-NNI
  - NREN 2
  - Area A.2
  - Inter-Carrier Layer
  - E-NNI

- GEANT2
  - Area A
Deployment of G\textsuperscript{2}MPLS in NRENs

- Some add-ons to the required GMPLS feature list
  - Advance reservations
  - Co-existence with adjacent NRPS
  - Clear resource partitioning and virtualization
  - Path computation for \textit{lambda networks (optical impairments)} + management of advance reservation

- Grid+network specific desiderata
  - Integration of Grids with NRENs infrastructures (a.k.a. add a CP to Grids)
    - No more a “ships-in-the-night” style but reciprocal awareness and seamless one-step configuration
  - Identification of the reference points and CP models
  - Integration and escalation of \textit{recovery} procedures
  - Integration with existing \textit{AAA AuthZ} mechanisms for Grids
  - Integration with existing NMS
Coexistence of GMPLS and G²MPLS [1]

- Mainly an inter-domain problem
  - G²MPLS nodes dispersed in GMPLS domains act as std-GMPLS nodes (routing, signalling)
  - GMPLS nodes merged in G²MPLS domains CANNOT act as G²MPLS nodes

- The most convincing future scenario for intra-domain coexistence seems to be a multi-region domain
  - Phosphorus G²MPLS used to provide CP capabilities to equipments currently lacking in it (e.g. carrier Ethernet switches or similar)
  - Off-the-shelf GMPLS used to control SDH or WDM technologies
Coexistence of GMPLS and G²MPLS [2]

- … but in the inter-domain context
  - G²MPLS domains handle both Grid and network resources
  - ASON/GMPLS domains control only network resources

- This implies a reduction of the information set carried out through the G.E-NNI, both for routing purposes and, above all, for signalling
GMPLS/G²MPLS: 2-parties scenario

Grid User

Local Grid Resources

Grid MW

Std UNI client

NMS

GMPLS domain

Std UNI client

G²MPLS domain

TN

TN

Grid WS, via G.OUNI

GNS transaction

UNI

E-NNI

NMI

mgmt

mgmt

G.OUNI degraded to std UNI

ASON Call
GMPLS/G²MPLS: 3-parties scenario
Support to this analysis [1]

- public questionnaires to NRENs and HPCs

Goals
- To collect requirements,
- To stimulate interest towards our solutions,
- To identify trends and plans for GMPLS/G²MPLS deployment and/or Grid &network integration

Questionnaire for NRENs
- Answers (CESNET2, PIONER, FCCN, DFN, GARR, HEAnet)

Questionnaire for HPCs
- Answers (Barcelona Supercomp. Center, CINECA)
Support to this analysis [2]

- Positive positioning mainly by NRENs who are currently missing an automatic control/management system for network and Grid resources or just have test-beds for that
  - Internal to the project (PIONER, CESNET2)
  - External (GARR, DFN)

- “Conditional & polite” consensus by NRENs and HPC external to this consortium probably due to the lack of running and public demonstrations of the Phosphorus concepts

- Main drivers towards consensus
  - capability of Phosphorus GMPLS/G^2MPLS to control efficiently the optical infrastructures integrated with the switched Ethernet
  - smooth integration of Phosphorus GMPLS/G^2MPLS with the GN2 BoD
  - Support and maintenance of the produced software
  - The wide dissemination of Phosphorus concept