GNS-WSI version 3

(Grid Network Service – Web Services Interface, version 3)

DRAFT

PRELIMINARY INFORMATION

The G-lambda project **http://www.g-lambda.net/**

Oct.27 2008

Table of Contents

1. Introduction

This document specifies GNS-WSI3 (Grid Network Services – Web Services Interface ver.3) protocol for dynamically reserving and allocating network resource. GNS-WSI3 has been discussed and defined in the G-lambda project (http://www.g-lambda.net). In the concept of G-lambda, network resources are basically bandwidths between end-points such as optical paths (lambda), but connectivity such as VLAN can be also supported. Network resources are requested by end-users or middleware such as ASP (Application Service Provider) by reservation requests, and if a request is granted, the resource is automatically provisioned and released at the beginning and end of the reserved period. The G-lambda architecture assumes use of not only network resources but also compute and other resources in a unified manner as shown in Figure 1.

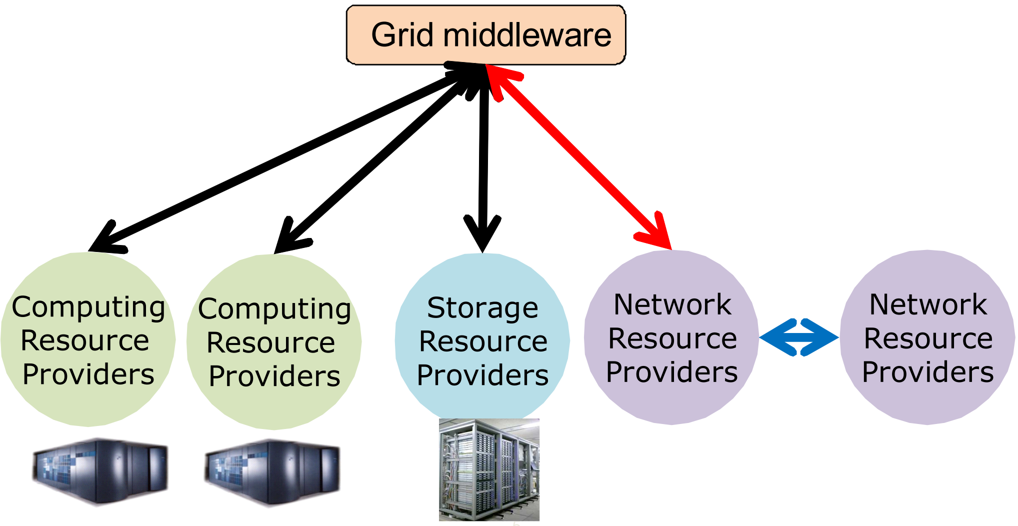
Grid middleware
Computing Resource Providers
Computing Resource Providers
Storage Resource Providers
Network Resource Providers
Network Resource Providers

Figure 1. Unified interface for multiple types of resources.

Requests from end-users or upper layer middleware can be propagated in a tree structured request paths as shown in Figure 2.

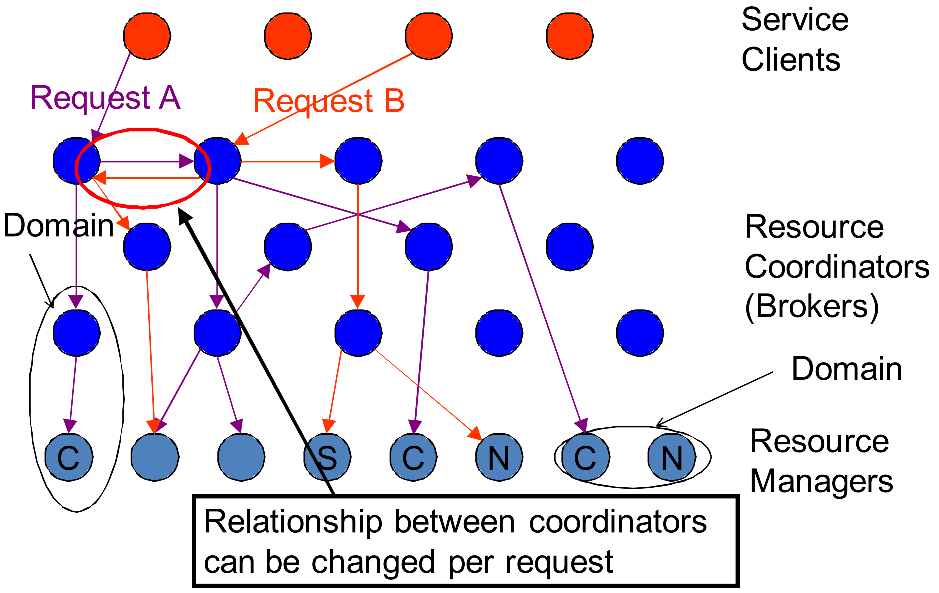
Service Clients
Request A
Request B
Domain
Resource Coordinators (Brokers)
Domain
Resource Managers
C
S
C
N
C
N
Relationship between coordinators can be changed per request

Figure 2. Per-request tree model.

GNS-WSI3 defines messages for checking availability of resources, reserving resources, gathering information about reserved and/or allocated resources, and modify/cancel existing reservations. GNS-WSI3 is defined using Web Services technology. The complete list of supported messages is defined by Web Services Description Language (WSDL). This document is based on3 the WSDL files shown in the appendices and provides additional details on the information elements in each message.

G-lambda project is currently working on defining both WSRF-based and non-WSRF-based GNS-WSI3. In this draft document, the WSRF-based version is explained. Additional information on non-WSRF-based interface will be provided in the future.

1.1. G-lambda project

G-lambda project is a joint research of National Institute of Advanced Industrial Science and Technology (AIST), KDDI R&D Laboratories, NTT, and the National Institute of Information and Communications Technology (NICT).

1.2. References

The following documents contain information that is relevant to this specification:

[1] "Web Services Description Language (WSDL) 1.1", W3C Note. http://www.w3.org/TR/wsdl

[2] OASIS Web Services Resource Framework (WSRF) 1.2 http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrf

[3] IANA-MAU-MIB, Internet Assigned Numbers Authority (IANA), available at http://www.iana.org/assignments/ianamau-mib, April 2007.

[4] Global Information Infrastructure and internet protocol aspects, ITU-T, Recommendation Y.1540, December 2002.

2. Notation Conventions

The key words "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" are to be interpreted as described in RFC-2119 [RFC 2119]. This specification uses namespace prefixes throughout; they are listed in Table 1. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

Table 1. Prefixes and namespaces used in this specification.

Prefix	Namespace
xsd	http://www.w3.org/2001/XMLSchema
rsv	http://schemas.glambda.net/gnswsi3/2008/10/reservation
rsvf	http://schemas.glambda.net/gnswsi3/2008/10/reservationfactory
rsvc	http://schemas.glambda.net/gnswsi3/2008/10/reservationcommand
rdl	http://schemas.glambda.net/gnswsi3/2008/10/rdl
cdl	http://schemas.glambda.net/gnswsi3/2008/10/cdl
ndl	http://schemas.glambda.net/gnswsi3/2008/10/ndl
jsdl	http://schemas.ggf.org/jsdl/2005/11/jsdl

3. Definitions

Client: An application, user, middleware or other NRM that requests resources.

ReservationCommandResource: A ReservationCommandResource is used to tread a operation for a reservation as a resource, and to manage a status of a operation. This is required to support the two phase commit protocol.

Domain: A network administrated by a telecommunication carrier, internet service provider, organization, group, or some other type of authority.

EndPoint Reference (EPR): A EndPoint Reference is used to address and access according to a resource, and represent a address of a resource at a given network endpoint.

Path: A pair of logical network endpoints. When a path allocates over different domains, a path may contain sub-paths.

Port-Type: A port type is a named set of abstract operations and the abstract messages involved.

ReservationResource: A ReservationResource is used to tread a reservation as a resource, and to manage a state of a reservation.

Web Services Resource Framework (WSRF): The WSRF is a set of six Web services specifications that define what is termed the WS-Resource approach to modeling and managing state in a Web services context.

WS-Addressing: WS-Addressing provides transport-neutral mechanisms to address Web services and messages.

WS-BaseFault: WS-BaseFaults defines an XML Schema type for base faults, along with rules for how this base fault type is used and extended by Web services.

WS-ResourceProperties: WS-ResourceProperties is used to read and write properties for a stateful resource.

4. GNS-WSI Services

GNS-WSI provides the following services:

- ReservationFactoryService: Receives registration requests to book Grid resources. It also returns information on resources available on the Grid.
- ReservationService: Receives reservation, modification, and release requests. It also manages current status of reserved resources.
- ReservationCommandService: Supports two-phase commit. It manages the status of pre-reserve, -modify, and -release processes, and abort or commit for each process on order of users.

ReservationResource and ReservationCommandResource are service instances for ReservationService and ReservationCommandService for each user request, respectively.

5. State Model

5.1. ReservationStatus

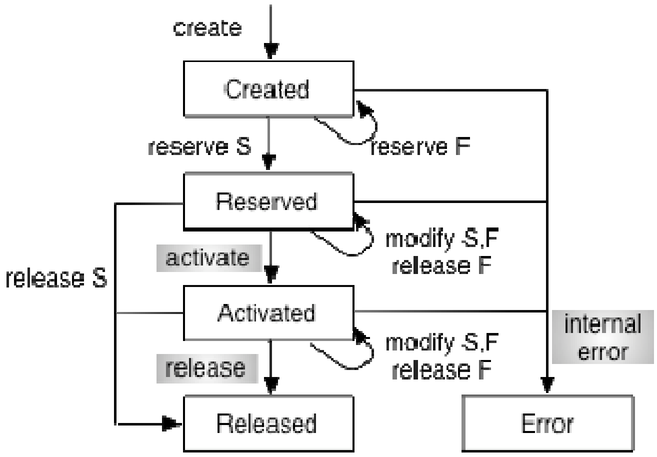

Figure 3. The ReservationStatus transition process.

ReservationStatus is an attribute of ReservationService and represents the current reservation status for each reservation request. The ReservationStatus transition process is shown in Figure 3. The ReservationStatus transition process consists of the following:

- ***Created***: A registration request is accepted (ReservationResource is created).
- ***Reserved***: Requested resources are booked.
- ***Activated***: The resources are activated.
- ***Released***: The resources are released.
- ***Error***: Errors have occurred.

create, **reserve**, **modify**, and **release** in Figure 3 indicate operations of ReservationFactoryService and ReservationService invoked by a client. **S** and **F** represent success and failure or destruction by the client of each command. The gray squares represent status changes at the server side.

5.2. CommandStatus

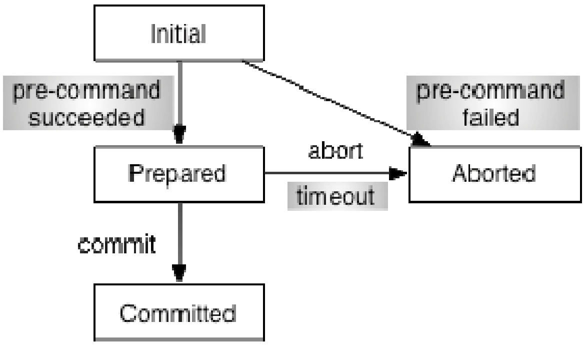

Figure 4. The CommandStatus transition process.

CommandStatus is an attribute of ReservationCommandService and represents the current command status of each ReservationCommandResource created by a reservation-related operation such as reserve, modify, or release.

The CommandStatus transition process shown in Figure 4 consists of the following:

- ***Initial***: reserve / modify / release command has been sent to an actual resource manager, but the request has not been completed yet.
- ***Prepared***: The requested command has been prepared.
- ***Committed***: The command has been completed.
- ***Aborted***: The requested resources are not available or the pre-command has expired.

commit and **abort** in Figure 4 are invoked by the client, and the gray squares also represent status changes at the server side. After CommandStatus has changed to *Prepared*, the client invokes commit or abort.

We use a modified two-phase commit protocol. Fundamentally, a two-phase commit is a blocking protocol. If a coordinator fails after a reserve request, CommandStatus may be left in the *Prepared* state until the coordinator is repaired and the requested resources are blocked for that duration. Moreover, a coordinator and its cohorts are loosely coupled on the Grid, and the coordinator may not issue a commit or abort request. We applied an automatic ``time out'' to the transit from *Prepared* to *Aborted*.

In our system, *Prepared* waiting for a commit or abort request times out at $T_{timeout}$ as follows:

$$T_{timeout} = T_{transit} + \varepsilon$$

$T_{transit}$ indicates the state transit time from *Initial* to *Prepared*.

6. Protocol Sequence of Advance Reservation Process

6.1. State Transitions in the two-phase commit operation

Table 2 shows the state transition of command status corresponding to results of each phase.

From **Initial**, state transits to **Prepared** or **Aborted** according to the result of 1st phase operation. From **Prepared**, state transits to **Committed** or **Aborted** according to the result of 2nd phase (reception of commit command).

From **Aborted** and **Committed**, no state transition will happen since those are finalized state.

Table 2. State transition of command status.

Origin state Condition	Initial	Prepared	Aborted	Committed
1st phase success	Prepared	-	-	-
1st phase fail	Aborted	-	-	-
Reception of abort command	-	Aborted	-	-
Reception of commit command	-	Committed	-	-
Timeout	-	Aborted	-	-

Table 3 shows the state transition of reservation status corresponding to results of each phase.

Initial state of reservation status is **Created**. 1st phase triggers no state transition, therefore the status is kept unchanged. In the success of 2nd phase, state transition occurs when operation is **Reserve** or **Release**. 2nd phase success of **Modify** just overwrite parameters of ReservationResource. When StartTime of a request comes (Activation phase), **Reserved** state transits to **Activated** or **Error** according to the result of activation operation. When EndTime of a request comes (Reservation expiration phase), **Activated** state transits to **Released** or **Error** according to the result of release operation.

Table 3. State transition of reservation status

Origin state Condition	Created	Reserved	Activated	Released	Error
1st phase success (for Modify and Release)	-	Reserved	Activated	-	-
1st phase fail (for Modify and Release)	-	Reserved	Activated	-	-
2nd phase success (for Reserve)	Reserved	-	-	-	-
2nd phase success (for	-	Reserved	Activated	-	-

Modify)					
2nd phase success (for Release)	-	Released	Released	-	-
2nd phase fail	Error	Error	Error	-	-
Activation success	-	Activated	-	-	-
Activation fail	-	Error	-	-	-
Reservation expiration (Release successful)	-	-	Released	-	-
Reservation expiration (Release fail)	-	-	Error	-	-

6.2. Reservation Diagrams

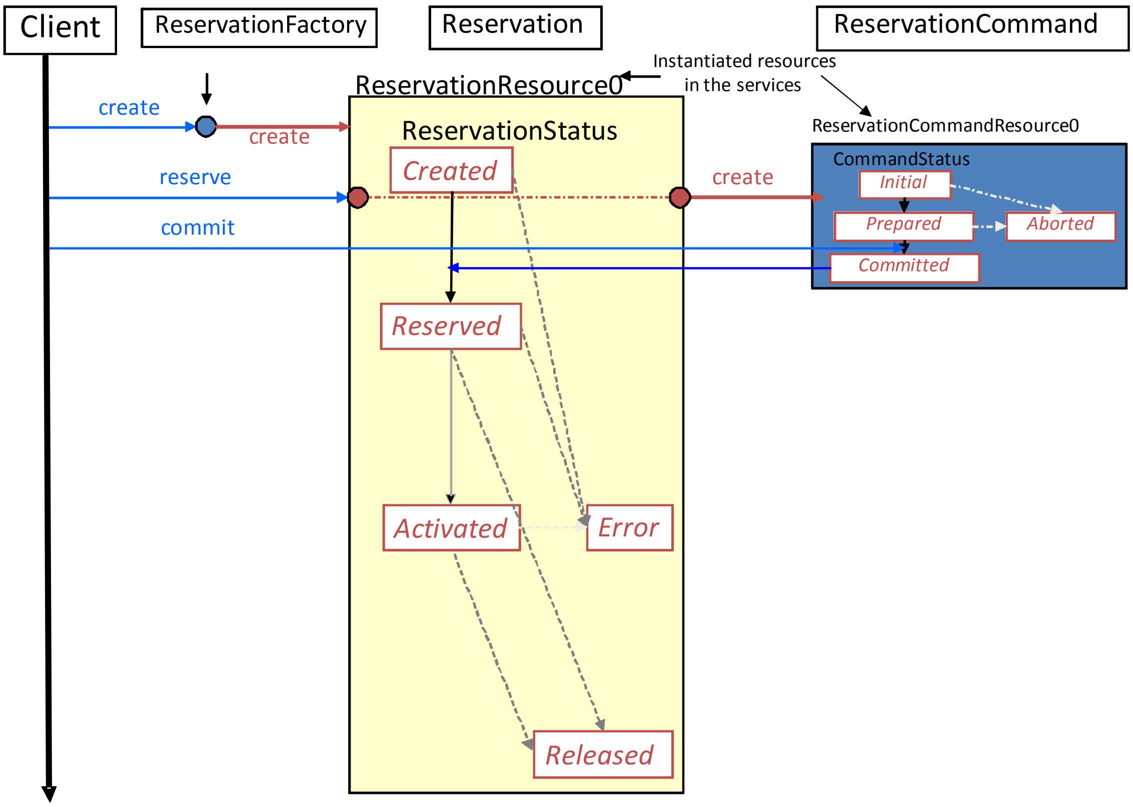

Figure 5. Protocol sequence of reservation process.

Figure 5 shows protocol sequence of reservation process.

1. ReservationResource is created by operation "create" of ReservationFactory service.
2. ReservationCommandResource is created by executing operation "reserve" of Reservation service on ReservationResource of state "Created".
3. ReservationCommandResource status transits from "Initial" to "Prepared" if the requested resource is available.

4. ReservationCommandResource status transits to “Committed” if operation “commit” of ReservationCommand service is conducted when the status is “Prepared”.
5. ReservationResource status becomes “Reserved” if the transition of ReservationCommandResource status to “Committed” is detected.

ReservationResouce status transits to “Activated” at StartTime of the request.

In process 3, if there is no resource available for a request, ReservationCommandResource status transits from “Initial” to “Aborted” and ReservationResource status is kept unchanged (“Created”). In the case that “commit” is not correctly conducted or is not conducted before timeout in process 4, ReservationCommandResource status also transits to “Aborted” and ReservationResource status is kept unchanged (“Created”).

ReservationResource status transits to “Error” if release process becomes not able to be proceeded by any exception (ex. internal error).

6.3. Modification Diagrams

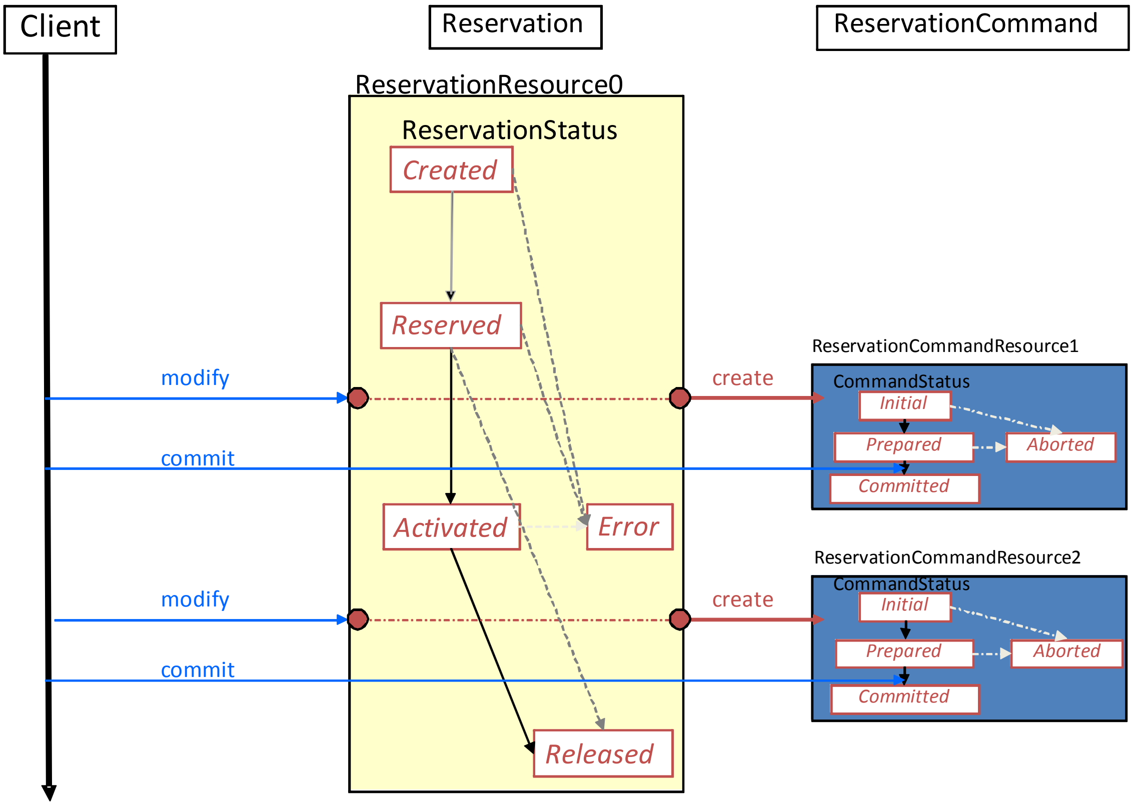

Figure 6. Protocol sequence of modify process.

Figure 6 shows protocol sequence of modify process.

1. ReservationCommandResource is created by executing operation “modify” of Reservation service on ReservationResource of state “Reserved” or “Activated”.
2. ReservationCommandResource status transits from “Initial” to “Prepared” if the

requested resource is available.

3. ReservaionCommandResource status transits to “Committed” if operation “commit” of ReservationCommand service is conducted when ReservationCommandResource status is “Prepared”.
4. Parameters of ReservationResource are modified.

In process 2, if there is no resource available for a request, ReservationCommandResource status transits from “Initial” to “Aborted” and ReservationResource status is kept unchanged (“Reserved” or “Activated”). In the case that “commit” is not correctly conducted or is not conducted before timeout in process 3, ReservationCommandResource status also transits to “Aborted” and ReservationResource status is kept unchanged (“Reserved” or “Activated”).

ReservationResource status transits to “Error” if release process becomes not able to be proceeded by any exception (ex. internal error).

6.4. Release Diagrams

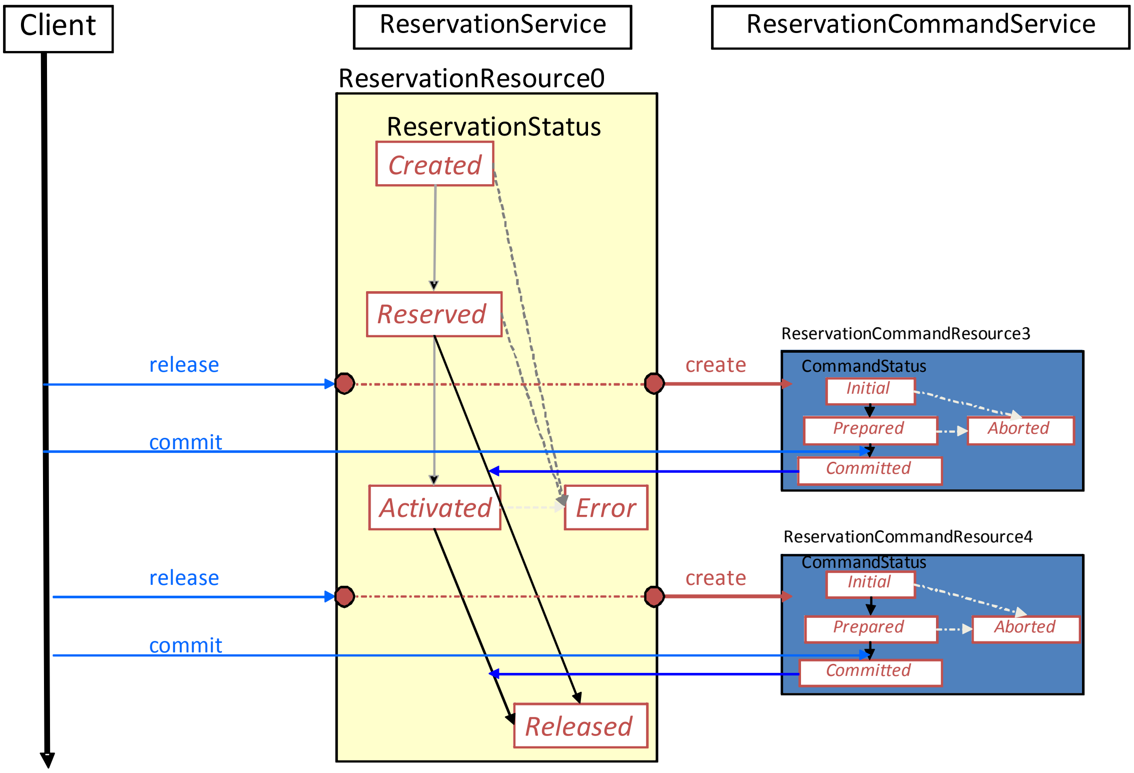

Figure 7. Protocol sequence of release process.

Figure 7 shows protocol sequence of release process.

1. ReservationCommandResource is created by executing operation “release” of Reservation service on ReservationResource of state “Reserved” or “Activated”.

2. ReservationCommandResource status transits from “Initial” to “Prepared” if the resource is able to be released.
3. ReservaionCommandResource status transits to “Committed” if operation “commit” of ReservationCommand service is conducted when ReservationCommandResource status is “Prepared”.
4. ReservationResource status transits to “Released” if the transition of ReservationCommandResource status to “Committed” is detected.

In process 2, if there is no resource available for a request, ReservationCommandResource status transits from “Initial” to “Aborted” and ReservationResource status is kept unchanged (“Reserved” of “Activated”). If “commit” is not conducted until timeout in process 3, ReservationCommandResource status also transits to “Aborted” and ReservationResource status is kept unchanged (“Reserved” of “Activated”).

ReservationResource status transits to “Error” if release process becomes not able to be proceeded by any exception (ex. internal error).

7. Port-types

Table 4 shows the summary of the GNS-WSI port-types and their operations. The details of each port-type are described in Section 8, 9, and 10.

Table 4. Summary of the GNS-WSI port-types and their operations.

ReservationFactory Port-Type	
create	Creates reservation resource
getAvailableResources	Discovery of available resources
checkResourceAvailability	Checks availability of specified resources
Reservation Port-Type	
reserve	Makes advance reservation
modify	Modifies specified reservation resources
modifyAll	Modifies reservation time of all the reserved resources
release	Releases specified reservation resources
releaseAll	Releases all the reserved resources
getReservationStatus	Returns the reservation status
getResourceProperty	Returns specified ReservationResource property
ReservationCommand Port-Type	
commit	Commits the request
abort	Aborts the request
getCommandStatus	Returns the command status
getResourceProperty	Returns specified ReservationResource property

8. ReservationFactory Port-Type

The ReservationFactory service creates a ReservationResource instance in the Reservation service.

Elements defined in this section have a namespace prefix *rsvf*.

8.1. create

The operation is used to create ReservationResource.

8.1.1. Input(s)

Entity	Inherits from			Description
createRequest				Request for creating a resource
Property	**Type**	**Multi.**	**Unit**	**Description**
DistinguishedName	string	0..1		A character string containing a user name who reserved this resource. When the DistinguishedName is null, the operation is processed as an anonymous user

8.1.2. Output(s)

Entity	Inherits from			Description
createResponse				Response against a createRequest
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationEPR	EndpointReferenceType	1		An end point reference for the ReservationResource

8.1.3. Fault

Entity	Inherits from	Description
reservationFactoryFault	BaseFaultType	Fault type of ReservationFactory service

8.2. getAvailableResources

The operation is used to discover available resources.

8.2.1. Input

<table>
<tr><th>Entity</th><th>Inherits from</th><th colspan="3">Description</th></tr>
<tr><td>getAvailableResourcesRequest</td><td></td><td colspan="3">Request for discovering available resources</td></tr>
<tr><th>Property</th><th>Type</th><th>Multi.</th><th>Unit</th><th>Description</th></tr>
<tr><td>DistinguishedName</td><td>string</td><td>0..1</td><td></td><td>A character string containing a user name who reserved this resource. When the DistinguishedName is null, the operation is processed as an anonymous user</td></tr>
<tr><td>AvailableResourceQuery</td><td>AvailableResourceQuery_Type</td><td>1..*</td><td></td><td>Query to discover available resources</td></tr>
</table>

8.2.2. Output

<table>
<tr><th>Entity</th><th>Inherits from</th><th colspan="3">Description</th></tr>
<tr><td>getAvailableResourcesResponse</td><td></td><td colspan="3">Response against a getAvailableRequest</td></tr>
<tr><th>Property</th><th>Type</th><th>Multi.</th><th>Unit</th><th>Description</th></tr>
<tr><td>AvailableResourceQueryResult</td><td>AvailableResourceQueryResult_Type</td><td>0..*</td><td></td><td>Result of getAvailableResourceQuery, which includes available reservation resources</td></tr>
</table>

8.2.3. Fault

Entity	Inherits from	Description
reservationFactoryFault	BaseFaultType	Fault type of ReservationFactory service

8.3. checkResourceAvailability

The operation is used to check availability of specified resources.

8.3.1. Input

Entity	Inherits from			Description
checkResourceAvailabilityRequest				Request for checking resource availability
Property	**Type**	**Multi.**	**Unit**	**Description**
DistinguishedName	string	0..1		A character string containing a user name who reserved this resource. When the DistinguishedName is null, the operation is processed as an anonymous user
ReservationResources	ReservationResources_Type	1..*		Reservation resources, each of which includes the reservation ID and the time specification

8.3.2. Output

Entity	Inherits from			Description
checkResourceAvailabilityResponse				Response against a checkResourceAvailabilityRequest
Property	**Type**	**Multi.**	**Unit**	**Description**
ResourceStatus	ResourceStatus_Type	1..*		Status of a specified resource. The order should be the same of ReservationResources

8.3.3. Fault

Entity	Inherits from	Description
reservationFactoryFault	BaseFaultType	Fault type of ReservationFactory service

9. Reservation Port-Type

The Reservation service provides general reservation operations (e.g., reservation, modification, release). The Reservation service creates ReservationCommandResource instance in ReservationCommandService.

Elements defined in this section have a namespace prefix *rsv*.

9.1. reserve

The operation is used to create a reservation of some resources.

9.1.1. Input(s)

Entity	Inherits from	Description		
reserveRequest		Request for reserving a resource		
Property	**Type**	**Multi.**	**Unit**	**Description**
ResourceRequirements	ResourceRequirements_Type	1		Information including resource requirements, time resources, and DistinguishedName

9.1.2. Output(s)

Entity	Inherits from	Description		
reserveResponse		Response against a reserveRequest		
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationCommandEPR	EndpointReferenceType	1		An end point reference for the ReservationCommandResource

9.1.3. Fault

Entity	Inherits from	Description
reservationFault	BaseFaultType	Fault tyep of Reservation service

9.2. modify

The operation is used to modify an existing reservation.

9.2.1. Input(s)

Entity	Inherits from	Description		
modifyRequest		Request for modifying specified reservation resources		
Property	**Type**	**Multi.**	**Unit**	**Description**
ResourceRequirements	ResourceRequirements_Type	1		Information including resource requirements, time resources, and DistinguishedName

9.2.2. Output(s)

<table>
<tr><th>Entity</th><th colspan="2">Inherits from</th><th colspan="3">Description</th></tr>
<tr><td>modifyResponse</td><td colspan="2"></td><td colspan="3">Response against a modifyRequest</td></tr>
<tr><th>Property</th><th>Type</th><th colspan="2">Multi.</th><th>Unit</th><th>Description</th></tr>
<tr><td>ReservationCommandEPR</td><td>EndpointReferenceType</td><td colspan="2">1</td><td></td><td>An end point reference for the ReservationCommandResource</td></tr>
</table>

9.2.3. Fault

Entity	Inherits from	Description
reservationFault	BaseFaultType	Fault tyep of Reservation service

9.3. modifyAll

The operation is used to modify all existing reservations.

9.3.1. Input(s)

<table>
<tr><th>Entity</th><th colspan="2">Inherits from</th><th colspan="3">Description</th></tr>
<tr><td>modifyAllRequest</td><td colspan="2"></td><td colspan="3">Request for modifying all reservation resources</td></tr>
<tr><th>Property</th><th>Type</th><th colspan="2">Multi.</th><th>Unit</th><th>Description</th></tr>
<tr><td>Timeframe</td><td>Timeframe_Type</td><td colspan="2">1</td><td></td><td>Time frame of a reservation</td></tr>
</table>

9.3.2. Output(s)

<table>
<tr><th>Entity</th><th colspan="2">Inherits from</th><th colspan="3">Description</th></tr>
<tr><td>modifyAllResponse</td><td colspan="2"></td><td colspan="3">Response against a modifyAllRequest</td></tr>
<tr><th>Property</th><th>Type</th><th colspan="2">Multi.</th><th>Unit</th><th>Description</th></tr>
<tr><td>ReservationCommandEPR</td><td>EndpointReferenceType</td><td colspan="2">1</td><td></td><td>An end point reference for the ReservationCommandResource</td></tr>
</table>

9.3.3. Fault

Entity	Inherits from	Description

reservationFault	BaseFaultType	Fault tyep of Reservation service

9.4. release

The operation is used to release an active resource.

9.4.1. Input(s)

Entity	Inherits from	Description
releaseRequest		Request for releasing specified reservation resources

Property	Type	Multi.	Unit	Description
ReservationID	string	1..*		ReservationID is used to identify a set of requested or reserved resources which is requested by a reserve operation. ReservationID is generated by the requestee of the reserve operation.

9.4.2. Output(s)

Entity	Inherits from	Description
releaseResponse		Response against a releaseRequest

Property	Type	Multi.	Unit	Description
ReservationCommandEPR	EndpointReferenceType	1		An end point reference for the ReservationCommandResource

9.4.3. Fault

Entity	Inherits from	Description
reservationFault	BaseFaultType	Fault tyep of Reservation service

9.5. releaseAll

The operation is used to release all active resources.

9.5.1. Input(s)

Entity	Inherits from			Description
releaseAllRequest				Request for releasing all reservation resources
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			An end point reference for the ReservationCommandResource

9.5.2. Output(s)

Entity	Inherits from			Description
releaseAllResponse				Response against a releaseAllRequest
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationCommandEPR	EndpointReferenceType	1		An end point reference for the ReservationCommandResource

9.5.3. Fault

Entity	Inherits from	Description
reservationFault	BaseFaultType	Fault tyep of Reservation service

9.6. getReservationStatus

The operation is used to get the status of a reservation.

9.6.1. Input(s)

Entity	Inherits from			Description
getReservationStatusRequest				Request to check a specified ReservationResource property
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			An end point reference for the ReservationCommandResource

9.6.2. Output(s)

Entity	Inherits from			Description
getReservationStatusResponse				Response against a getReservationStatusRequest
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationStatus	ReservationStatus_Type	1		Current reservation status for a reservation request

9.6.3. Fault

Entity	Inherits from	Description
reservationFault	BaseFaultType	Fault tyep of Reservation service

9.7. getResourceProperty

The operation is used to get the ResourceProperty.

9.7.1. Input(s)

Reservation attribute name (javax.xml.namespace.QName).

9.7.2. Output(s)

Entity	Inherits from			Description
ReservationResourceProperties				Properties of reserved resource
Property	**Type**	**Multi.**	**Unit**	**Description**
DistinguishedName	string	0..1		A character string containing a user name who reserved this resource. When the DistinguishedName is null, the operation is processed as an anonymous user
ReservationStatus	ReservationStatus_Type	1		Current reservation status for a reservation request
ResourceInformation	ResourceInformation_Type	0..*		Category for resources, for example computing or network resource

ReservationCommandHistory	ReservationCommandHistory_Type	0..*		History of user request information
ErrorInfo	BaseFaultType	0..1		Reason when ResourceStatus / CommandStatus becomes an error

10. ReservationCommand Port-Type

The Reservation service realizes two-phase commit and non-blocking operation.

Elements defined in this section have a namespace prefix *rsvc*.

10.1. commit

The operation is used to commit a request.

10.1.1. Input(s)

Entity	**Inherits from**			**Description**
commitRequest				A request to commit the operation (reserve/modify/release).
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			Status of a command

10.1.2. Output(s)

Entity	**Inherits from**			**Description**
commitResponse				A response against a commitRequest.
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			Status of a command

10.1.3. Fault

Entity	**Inherits from**	**Description**
reservationCommandFault	BaseFaultType	Fault type of ReservationCommand service

10.2. abort

The operation is used to abort a request.

10.2.1. Input(s)

Entity	Inherits from	Description		
abortRequest		A request to abort the operation (reserve/modify/release).		
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			Status of a command

10.2.2. Output(s)

Entity	Inherits from	Description		
abortResponse		A response against an abortRequest.		
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			Status of a command

10.2.3. Fault

Entity	Inherits from	Description
reservationCommandFault	BaseFaultType	Fault type of ReservationCommand service

10.3. getCommandStatus

The operation is used to get CommandStatus.

10.3.1. Input(s)

Entity	Inherits from	Description		
getCommandStatusRequest		A request to obtain the status of the ReservationCommandResource.		
Property	**Type**	**Multi.**	**Unit**	**Description**
	void			Status of a command

10.3.2. Output(s)

Entity	Inherits from	Description
getCommandStatusResponse		A response against a getCommandStatusRequest.

Property	Type	Multi.	Unit	Description
CommandStatus	CommandStatus_Type	1		Status of a command

10.3.3. Fault

Entity	Inherits from	Description
reservationCommandFault	BaseFaultType	Fault type of ReservationCommand service

10.4. getResourceProperty

The operation is used to get the ResourceProperty.

10.4.1. Input(s)

Reservation attribute name (javax.xml.namespace.QName).

10.4.2. Output(s)

Entity	Inherits from	Description
ReservationCommandResourceProperties		A set of properties related to a ReservationCommandResource.

Property	Type	Multi.	Unit	Description
CommandStatus	CommandStatus_Type	1		Status of a command
ErrorInfo	BaseFaultType	0..1		Reason when ResourceStatus / CommandStatus becomes an error
ReservationResourceKey	string	1		Key of a ReservationResource

11. Data types

This section contains the definition of data types.

11.1. Reservation related data types

Elements defined in this section have a namespace prefix *rdl*.

11.1.1. ResourceStatus_Type

Closed enumeration

Entity	Inherits from	Description
ResourceStatus_Type	string	Status of a specified resource. The order should be the same of ReservationResources
Value	**Description**	
Available	Resource is available	
NotAvailable	Resource is not available	

11.1.2. ReservationStatus_Type

Closed enumeration

Entity	Inherits from	Description
ReservationStatus_Type	string	Current reservation status for a reservation request
Value	**Description**	
Created	A registration request is accepted (ReservationResource is create)	
Reserved	The requested resource is booked	
Activated	The resource is activated	
Released	The resource is released	
Error	Errors have occurred	

11.1.3. CommandStatus_Type

Closed enumeration

Entity	Inherits from	Description
CommandStatus_Type	string	Status of a command
Value	**Description**	

Initial	A reserve / modify / release command has been sent to an actual resource manager
Prepared	The requested command has been prepared
Aborted	The requested resources are not available or the pre-command has expired
Committed	The requested command has been completed

11.1.4. TimeUnit_Type

Entity	Inherits from	Description
TimeUnit_Type	string	A unit of time
Value	**Description**	
MILLISECOND	Millisecond	
SECOND	Second	
MINUTE	Minute	
HOUR	Hour	
DAY	Day	
WEEK	Week	
MONTH	Month	
YEAR	Year	

11.1.5. ResourceInformation_Type

Entity	Inherits from			Description
ResourceInformation_Type				Category for resources, for example computing or network resource
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationResources	ReservationResources_Type	0..*		Reservation resources, each of which includes the reservation ID and the time specification

11.1.6. ReservationResources_Type

Entity	Inherits from			Description
ReservationResources_Type				Reservation resources, each of which includes the reservation ID and the time specification
Property	**Type**	**Multi.**	**Unit**	**Description**

ReservationID	string	0..1		ReservationID is used to identify a set of requested or reserved resources which is requested by a reserve operation. ReservationID is generated by the requestee of the reserve operation.
TimeSpecification	TimeSpecification_Type	1		Time specification

11.1.7. AvailableResourceQuery_Type

Entity	Inherits from			Description
AvailableResourceQuery_Type				Query to discover available resources
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationResources	ReservationResources_Type	1		Reservation resources, each of which includes the reservation ID and the time specification

11.1.8. AvailableResourceQueryResult_Type

Entity	Inherits from			Description
AvailableResourceQueryResult_Type				Result of getAvailableResourceQuery, which includes available reservation resources
Property	**Type**	**Multi.**	**Unit**	**Description**
ReservationResources	ReservationResources_Type	0..*		Reservation resources, each of which includes the reservation ID and the time specification

11.1.9. SLADocument_Type

Entity	Inherits from			Description
SLADocument_Type				Searvice Level Agreement (SLA) document
Property	**Type**	**Multi.**	**Unit**	**Description**
Timestamp	dateTime	1..*		Time stamp

11.1.10. ResourceRequirements_Type

Entity	Inherits from	Description		
ResourceRequirements_Type		Information including resource requirements, time resources, and DistinguishedName		
Property	**Type**	**Multi.**	**Unit**	**Description**
CoallocationTimeframe	CoallocationTimeframe_Type	0..*		Coallocation time frame
ReservationResources	ReservationResources_Type	1..*		Reservation resources, each of which includes the reservation ID and the time specification

11.1.11. CoallocationTimeframe_Type

Entity	Inherits from	Description		
CoallocationTimeframe_Type		Coallocation time frame		
Property	**Type**	**Multi.**	**Unit**	**Description**
Timeframe	Timeframe_Type	1		Time frame of a reservation
CoallocationTimeframeID	string	1		Coallocation time frame ID

11.1.12. ReservationCommandHistory_Type

Entity	Inherits from	Description		
ReservationCommandHistory_Type		History of user request information		
Property	**Type**	**Multi.**	**Unit**	**Description**
Timestamp	dateTime	1		Time stamp
ReservationCommandEPR	EndpointReferenceType	1		An end point reference for the ReservationCommandResource
OperationName	string	1		Name of an operation
OperationArg	string	0..1		Argument of an operation
FinalCommandStatus	CommandStatus_Type	0..1		Final CommandStatus

11.1.13. TimeSpecification_Type

Entity	Inherits from	Description
TimeSpecification_Type		Time specification

Property	Type	Multi.	Unit	Description
Timeframe	Timeframe_Type	0..1		Time frame of a reservation
CoallocationTimeframeID	string	0..1		Coallocation time frame ID

11.1.14. Timeframe_Type

Entity	Inherits from	Description
Timeframe_Type		Time frame of a reservation

Property	Type	Multi.	Unit	Description
Exact	Exact_Type	0..1		Exact time frame type
Range	Range_Type	0..1		Ranged time frame type

11.1.15. Exact_Type

Entity	Inherits from	Description
Exact_Type		Exact time frame type

Property	Type	Multi.	Unit	Description
StartTime	dateTime	1		Start time of a reservation
EndTime	dateTime	1		End time of a reservation

11.1.16. Range_Type

Entity	Inherits from	Description
Range_Type		Ranged time frame type

Property	Type	Multi.	Unit	Description
Duration	Duration_Type	1		Duration
EarliestStartTime	dateTime	0..1		Earliest start time of a reservation
LatestStartTime	dateTime	0..1		Latest start time of a reservation

11.1.17. Duration_Type

Entity	Inherits from	Description		
Duration_Type		Duration		
Property	**Type**	**Multi.**	**Unit**	**Description**
DurationValue	long	1		A value of duration
TimeUnit	TimeUnit_Type	1		A unit of time

11.2. Network resources related data types

Elements defined in this section have a namespace prefix *ndl*.

11.2.1. BWUnit_Type

Closed enumeration

Entity	Inherits from	Description
BWUnit_Type	string	An unit of data transfer rate.
Value	**Description**	
Pbps	A unit of data transfer rate equal to 1,000 Tbps.	
Tbps	A unit of data transfer rate equal to 1,000 Gbps.	
Gbps	A unit of data transfer rate equal to 1,000 Mbps.	
Mbps	A unit of data transfer rate equal to 1,000 kbps.	
kbps	A unit of data transfer rate equal to 1,000 bits per seconds.	
bps		

11.2.2. MediaTypeName_Type

Closed enumeration

Entity	Inherits from	Description
MediaTypeName_Type	string	A name of the interface media defined as media attachment units [IANA-MAU-MIB].
Value	**Description**	
bOther	other or unknown	
bAUI	AUI	
b10base5	10BASE-5	

bFoirl	FOIRL
b10base2	10BASE-2
b10baseT	10BASE-T duplex mode unknown
b10baseFP	10BASE-FP
b10baseFB	10BASE-FB
b10baseFL	10BASE-FL duplex mode unknown
b10broad36	10BROAD36
b10baseTHD	10BASE-T half duplex mode
b10baseTFD	10BASE-T full duplex mode
b10baseFLHD	10BASE-FL half duplex mode
b10baseFLFD	10BASE-FL full duplex mode
b100baseT4	100BASE-T4
b100baseTXHD	100BASE-TX half duplex mode
b100baseTXFD	100BASE-TX full duplex mode
b100baseFXHD	100BASE-FX half duplex mode
b100baseFXFD	100BASE-FX full duplex mode
b100baseT2HD	100BASE-T2 half duplex mode
b100baseT2FD	100BASE-T2 full duplex mode
b1000baseXHD	1000BASE-X half duplex mode
b1000baseXFD	1000BASE-X full duplex mode
b1000baseLXHD	1000BASE-LX half duplex mode
b1000baseLXFD	1000BASE-LX full duplex mode
b1000baseSXHD	1000BASE-SX half duplex mode
b1000baseSXFD	1000BASE-SX full duplex mode
b1000baseCXHD	1000BASE-CX half duplex mode
b1000baseCXFD	1000BASE-CX full duplex mode
b1000baseTHD	1000BASE-T half duplex mode
b1000baseTFD	1000BASE-T full duplex mode
b10GbaseX	10GBASE-X
b10GbaseLX4	10GBASE-LX4
b10GbaseR	10GBASE-R
b10GbaseER	10GBASE-ER
b10GbaseLR	10GBASE-LR
b10GbaseSR	10GBASE-SR
b10GbaseW	10GBASE-W

b10GbaseEW	10GBASE-EW
b10GbaseLW	10GBASE-LW
b10GbaseSW	10GBASE-SW
b10GbaseCX4	10GBASE-CX4
b2BaseTL	2BASE-TL
b10PassTS	10PASS-TS
b100BaseBX10D	100BASE-BX10D
b100BaseBX10U	100BASE-BX10U
b100BaseLX10	100BASE-LX10
b1000BaseBX10D	1000BASE-BX10D
b1000BaseBX10U	1000BASE-BX10U
b1000BaseLX10	1000BASE-LX10
b1000BasePX10D	1000BASE-PX10D
b1000BasePX10U	1000BASE-PX10U
b1000BasePX20D	1000BASE-PX20D
b1000BasePX20U	1000BASE-PX20U

11.2.3. SwitchingScheme_Type

Closed enumeration

Entity	Inherits from	Description
SwitchingScheme_Type	string	A switching/routing scheme in a path.
Value	**Description**	
LAMBDA		
ETHERNET		
ATM		
MPLS		
IP		

11.2.4. NetworkResources_Type

Entity	Inherits from	Description
NetworkResources_T	ReservationResource	A set of network resource information.

ype	s_Type			
Property	**Type**	**Multi.**	**Unit**	**Description**
Path	Path_Type	0..1		A pair of logical network endpoints. When a path allocates over different domains, a path may contain sub-paths.
SLADocument	SLADocument_Type	0..1		Searvice Level Agreement (SLA) document

11.2.5. Path_Type

Entity	Inherits from	Description		
Path_Type		A pair of logical network endpoints. When a path allocates over different domains, a path may contain sub-paths.		
Property	**Type**	**Multi.**	**Unit**	**Description**
APoint	Endpoint_Type	1		An endpoint in a path. An APoint must be paired with a ZPoint.
ZPoint	Endpoint_Type	1		An endpoint in a path. A ZPoint must be paired with an APoint.
Route	Route_Type	0..1		A list of subsets of the path.
Bandwidth	Bandwidth_Type	1		An abstracted element of a requesting and assigned bandwidth.
PathProperties	PathProperties_Type	0..1		A set of a path information.

11.2.6. Endpoint_Type

Entity	Inherits from	Description		
Endpoint_Type		A termination site of a path or sub-path. An endpoint may contain termination points of connections.		
Property	**Type**	**Multi.**	**Unit**	**Description**
DomainName	string	0..1		A name of a domain, which is a network administrated by a telecommunication carrier, internet service provider, organization, group, or some other type of authority.

TerminationPoint	TerminationPoint_Type	0..*		A physical or logical termination point of a connection.

11.2.7. Termination_Point_Type

Entity	Inherits from			Description
TerminationPoint_Type				A physical or logical termination point of a connection.
Property	**Type**	**Multi.**	**Unit**	**Description**
TerminationPointName	string	0..1		A name of a termination point.
MediaType	MediaType_Type	0..1		A media type of a termination point.

11.2.8. Route_Type

Entity	Inherits from			Description
Route_Type				A list of subsets of the path.
Property	**Type**	**Multi.**	**Unit**	**Description**
SubPath	SubPath_Type	1..*		A pair of logical network endpoints. An NRM handling a sub-path may be different domain from the domain of the NRM handling a path.

11.2.9. SubPath_Type

Entity	Inherits from			Description
SubPath_Type				A pair of logical network endpoints. An NRM handling a sub-path may be different domain from the domain of the NRM handling a path.
Property	**Type**	**Multi.**	**Unit**	**Description**
APoint	Endpoint_Type	1		An endpoint in a path. An APoint must be paired with a ZPoint.
ZPoint	Endpoint_Type	1		An endpoint in a path. A ZPoint must be paired with an APoint.
Nrm	Nrm_Type	1		An NRM information handling a sub-path.

11.2.10. Nrm_Type

Entity	Inherits from	Description
Nrm_Type		An NRM information handling a sub-path.

Property	Type	Multi.	Unit	Description
NrmName	string	1		A name of NRM.
NrmUrl	string	1		A URL of NRM.

11.2.11. Bandwidth_Type

Entity	Inherits from	Description
Bandwidth_Type		An abstracted element of a requesting and assigned bandwidth.

Property	Type	Multi.	Unit	Description
RequestedBW	RequestedBW_Type	0..1		A requesting bandwidth by an application, user or other NRM to be guaranteed.
ReservedBW	ReservedBW_Type	0..1		An assigned bandwidth by the NRM.

11.2.12. RequestedBW_Type

Entity	Inherits from	Description
RequestedBW_Type		A requesting bandwidth by an application, user or other NRM to be guaranteed.

Property	Type	Multi.	Unit	Description
MinimumBW	GeneralBW_Type	1		An acceptable minimum bandwidth requesting from an application, user or other NRM to be guaranteed. The guaranteed bandwidth must not be assigned below the value of this parameter. When a MaximumBW is given, NRM decides the assigned bandwidth between a MinimumBW and a MaximumBW.
MaximumBW	GeneralBW_Typ	0..1		An acceptable maximum bandwidth requesting

	e			from an application, user or other NRM to be guaranteed. In a commercial network, clients can pay within the value of this parameter. NRM decides the assigned bandwidth between a MinimumBW and a MaximumBW. This parameter must be requested with a MinimumBW.
PeakBW	GeneralBW_Type	0..1		A requesting bandwith without guarantee. The value of this parameter should be equal to or less than the rate of the physical network interface.

11.2.13. ReservedBW_Type

Entity	**Inherits from**	**Description**		
ReservedBW_Type		An assigned bandwidth by the NRM.		
Property	**Type**	**Multi.**	**Unit**	**Description**
GuaranteedBW	GeneralBW_Type	1		An assigned bandwidth with guarantee.
PeakBW	GeneralBW_Type	0..1		A requesting bandwith without guarantee. The value of this parameter should be equal to or less than the rate of the physical network interface.

11.2.14. GeneralBW_Type

Entity	**Inherits from**	**Description**		
GeneralBW_Type				
Property	**Type**	**Multi.**	**Unit**	**Description**
Rate	int	1		The number of bits that are transfered per unit of time.
BWUnit	BWUnit_Type	1		An unit of data transfer rate.

11.2.15. PathProperties_Type

Entity	**Inherits from**			**Description**
PathProperties_Type				A set of a path information.
Property	**Type**	**Multi.**	**Unit**	**Description**
Availability	float	0..1	%	The ratio of the total time that a path is capable of being used in a year.
TransmissionDelay	int	0..1	msec	The acceptable one-way time to transfer between an APoint and a ZPoint. This parameter doesn't include a forwarding delay in switches or routers.
SwitchingScheme	SwitchingScheme_Type	0..1		A switching/routing scheme in a path.
EthernetParameters	EthernetParameters_Type	0..1		A set of Ethernet related information.
MPLSParameters	MPLSParameters_Type	0..1		A set of MPLS related information.
IPParameters	IPParameters_Type	0..1		A set of IP related information.

11.2.16. MediaType_Type

Entity	**Inherits from**			**Description**
MediaType_Type				A media type of a termination point.
Property	**Type**	**Multi.**	**Unit**	**Description**
MediaTypeName <<choice>>	MediaTypeName_Type	0..1		A name of the interface media defined as media attachment units [IANA-MAU-MIB].
OtherMediaType <<choice>>	string	0..1		A name of the other media type.

11.2.17. EthernetParameters_Type

Entity	**Inherits from**	**Description**
EthernetParameters_Type		A set of Ethernet related information.

Property	Type	Multi.	Unit	Description
MTU	int	0..1	Byte	A requested/allowable MTU.
VLANTagID	int	0..1		A parameter for configuring virtual LANs in the IEEE 802.1Q header.
CoS	int	0..1		A parameter to differentiate traffic in the Ethernet frame header.

11.2.18. MPLSParameters_Type

Entity	Inherits from			Description
MPLSParameters_Type				A set of MPLS related information.
Property	**Type**	**Multi.**	**Unit**	**Description**
LSPID	string	0..1		A parameter to create a label switched path (LSP) in MPLS networking.
Exp	int	0..1		A parameter to differentiate traffic in the MPLS header.

11.2.19. IPParameters_Type

Entity	Inherits from			Description
IPParameters_Type				A set of IP related information.
Property	**Type**	**Multi.**	**Unit**	**Description**
PacketLossRate	float	0..1	%	The ratio of total lost IP packet outcomes to total transmitted IP packets in a population of interest. The detailed definition is described at [Y.1540] as IPLR (IP Loss Rate).
PacketErrorRate	float	0..1	%	The ratio of total errored IP packet outcomes to the total of successful IP packet transfer outcomes plus errored IP packet outcomes in a population of intrest. The detailed definition is described at [Y.1540] as IPER (IP Error Rate).
AverageJitter	int	0..1	msec	The average variation of the end-to-end delay from a packet ot the next packet.
MaximumJitter	int	0..1	mse	The maximum variation of the end-to-end delay

			c	from a packet ot the next packet.
Latency	int	0..1	msec	The average one-way IP packet transfer time for a population of interest between an APoint and a ZPoint with forwarding delays at switches and routers. The detailed definition is described at [Y.1540] as IPTD (IP Transfer Delay).
LatencyVariation	int	0..1	msec	The variation of a one-way latency within a stream of packets. The detailed definition is described at [Y.1540] as IPDV (IP Delay Variation).
PacketReordering	boolean	0..1		NRM provides the guarantee to transfer packets in order or not.
ToS	int	0..1		A parameter reserved for the service type in the IPv4 header.

11.2.20. NetworkResourceSLADocument_Type

Entity	Inherits from	Description
NetworkResourceSLADocument_Type	SLADocument_Type	A set of information related to network service level agreement (SLA).

Property	Type	Multi.	Unit	Description
SLAAvailability	float	0..1	%	The ratio of the total time that an NRM is capable of being used in a year.
ProvisioningDelay	Duration_Type	0..1		The maximum time to activate an assigned path.
FailureReportDelay	Duration_Type	0..1		The maximum time to notify clients of failures.
Redundancy	string	0..1		The configuration to ensure the path continues under the failure(s).

11.3. Compute resources related data types

Elements defined in this section have a namespace prefix *cdl*.

11.3.1. ComputeResources_Type

Entity	Inherits from	Description

ComputeResources_Type	ReservationResources_Type	Compute resources		
Property	**Type**	**Multi.**	**Unit**	**Description**
Site	string	0..1		A site name
Resources	Resources_Type	0..1		Resources defined in JSDL v 1.0
ResourceAttribute	ResourceAttribute_Type	0..*		The compute resource is related to zero or more attributes

11.3.2. ResourceAttribute_Type

Entity	Inherits from	Description		
ResourceAttribute_Type		The compute resource is related to zero or more attributes		
Property	**Type**	**Multi.**	**Unit**	**Description**
Key	string	1		An attribute name
Value	string	1		A value for the attribute

11.3.3. ComputeResourcesSLADocument_Type

Entity	Inherits from	Description		
ComputeResourceSLADocument_Type	SLADocument_Type	SLA Document of compute resources		
Property	**Type**	**Multi.**	**Unit**	**Description**
Description	string	0..1		An SLA document description

12. Error Cases

TBD

13. Optional Extensions

13.1. Lifetime

TBD

13.2. Notification

TBD

13.3. SLA

TBD

14. Security Consideration

TBD