Motivation

Usually services or resources are used as available, in a best effort approach.

However, guarantees for Quality of Service are required, e.g. if

- the resource or service needs to have some dedicated properties
- a service will be used to perform a task with a fixed critical deadline
- if a service needs to be available during a defined period
- multiple resources or services have to be available in a predefined sequence (workflows) or at the same time (co-allocation)

Complex Service Level Agreements usually may not achieved in a single step, thus

Negotiation between service provider and service consumer is required

Service Level Agreements

Service Level Agreements (SLA) may be used to establish agreements on the quality of a service between a service provider and a service consumer.

Today these service level agreements are often created manually for individual services or are established through a framework contract between service provider and customer.

Given the dynamic nature of the requirements towards service provisioning in Grids and the increased flexibility in selecting the most appropriate service provider for a required service new mechanisms are needed to create Service Level Agreements on the fly and to modify them at a later state.

WS-Agreement was specified aiming to fill this gap.

Web-Services Agreement (WS-Agreement)

WS-Agreement is a specification defining a language and a protocol to create Service Level Agreements (SLA) between a service provider and a service consumer (usually a customer of the provider).

WS-Agreement currently is used in different projects, domains and on different levels to establish electronic agreements on the quality of a service (QoS) between a service provider and a service consumer.

The WS-Agreement specification version 1.0 already includes a protocol for negotiation of agreements. However, this protocol was designed to cover the most simple and general case: an offer for an SLA is made by either of the two parties and the respective other party may accept or reject the offer.

WS-Agreement has been specified by the Grid Resource Allocation Agreement Working Group (GRAAP-WG) and is a proposed OGF recommendation since May 2007.

WS-Agreement Structure and Terminology (1)

WS-Agreement defines a language and a protocol to represent the services of providers, create agreements based on offers and monitor agreement compliance at runtime.

WS-Agreement allows advertising the capabilities of service providers and creating agreements based on creational offers, and for monitoring agreement compliance at runtime.

An agreement defines a relationship between two parties that is dynamically established and dynamically managed. The objective of this relationship is to deliver a service by one of the parties. In the agreement each party agrees on the respective roles, rights and obligations.

A provider in an agreement offers a service according to conditions described in the agreement. A consumer enters into an agreement with the intent of obtaining guarantees on the availability of one or more services from the provider. Agreements can also be negotiated by entities acting on behalf the provider and / or the consumer.

WS-Agreement Structure and Terminology (2)

An agreement creation process usually consists of three steps:

- The initiator retrieves a template from the responder, which advertises the types of offers the responder is willing to accept.
- The initiator then makes an offer
- Which is either accepted or rejected by the responder.

An agreement consists of the agreement name, its' context and the agreement terms.

The context contains information about the involved parties and metadata such as the duration of the agreement.

WS-Agreement Structure and Terminology (3)

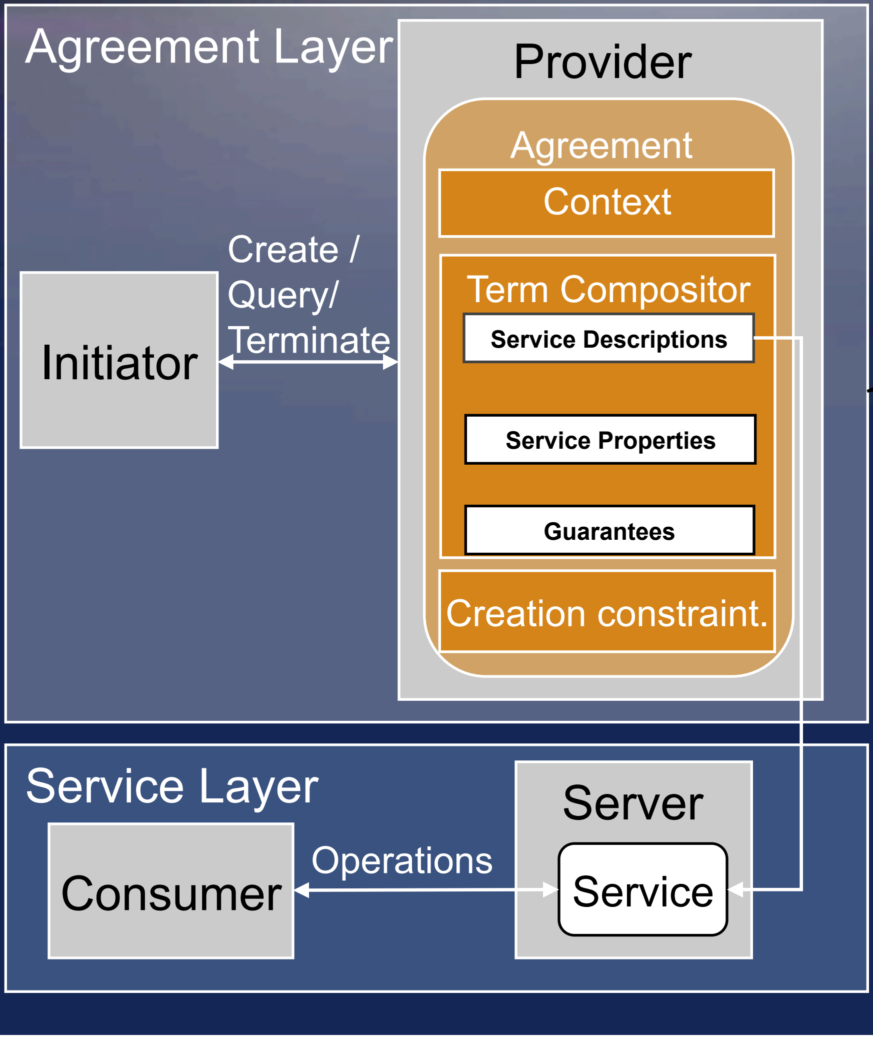
Agreement terms define the content of an agreement:

- Service Description Terms (SDTs) define the functionality that is delivered under an agreement. A SDT includes a domain-specific description of the offered or required functionality (the service itself).
- Guarantee Terms define assurance on service quality of the service described by the SDTs.
  - SDTs relate to Service Level Objectives (SLOs), which describe the quality of service aspects of the service that have to be fulfilled by the provider.

The Web Services Agreement Specification allows the usage of any domain specific or standard condition expression language to define SLOs.

The specification of domain-specific term languages is explicitly left open.

WS-Agreement Layered Model

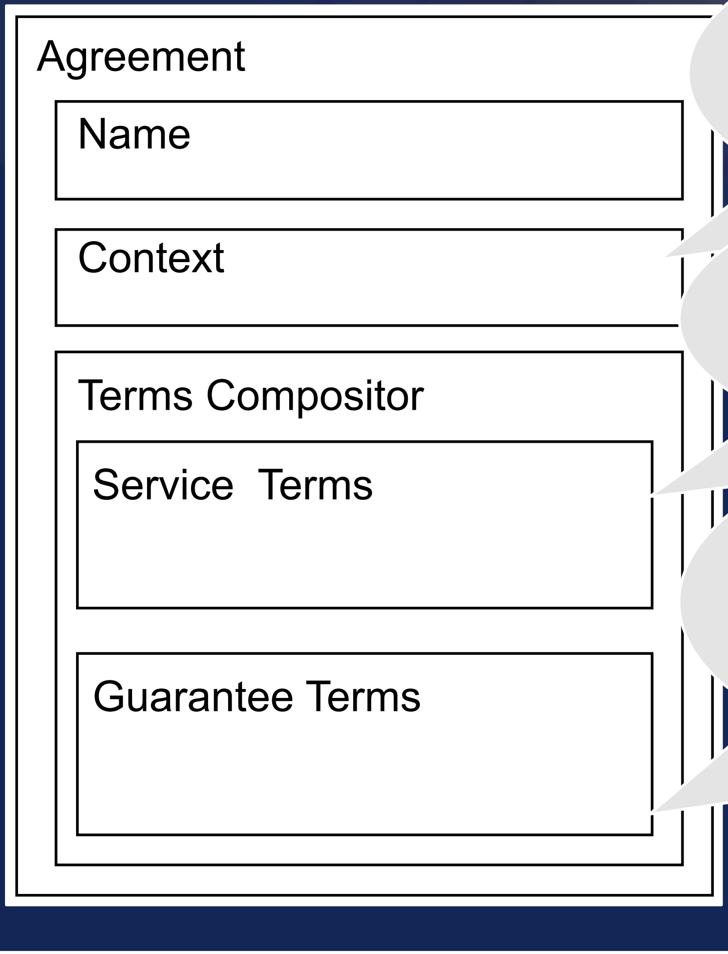


Agreement Layer: Provides a Web service-based interface that can be used to create, represent and monitor agreements with respect to provisioning of services implemented in the service

Service Layer provides the real service. It might be Web Service based or non Web Service based

Whether an Agreement Initiator is a Service Consumer or Service Provider (i.e. Agreement Responder becomes a Service Provider or Service Consumer) is domain and application dependent.

Agreement Document Structure



Agreement

Name

Context

Terms Compositor

Service Terms

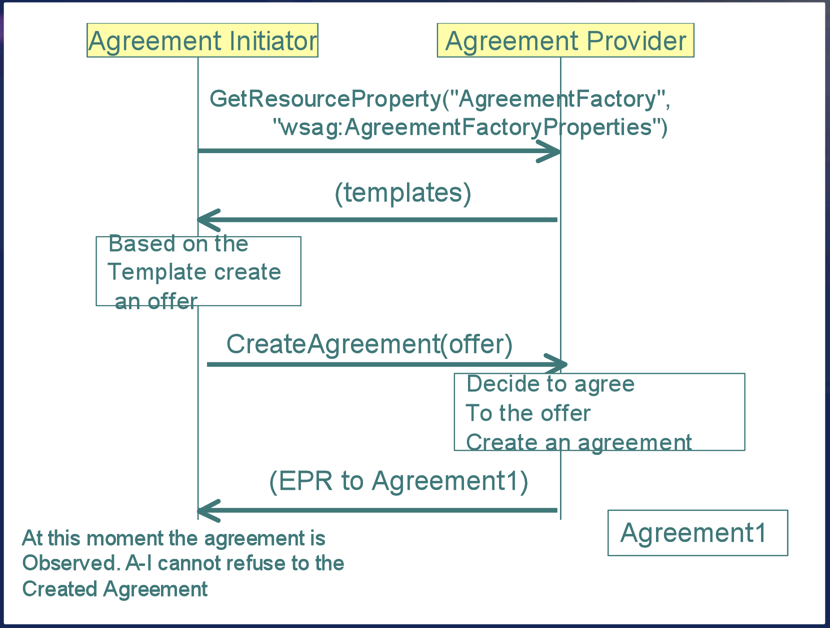
Guarantee Terms

Information about the Agreement Document
AgreementInitiator
AgreementResponder
ExpirationTime

The structure of an ***Agreement Template*** is the same as that of an Agreement, but an Agreement template MAY also contain a ***creation constraint*** section, i.e. a section with constraints on possible values of terms for creating an agreement. The constraints make it possible to specify the valid ranges or distinct values that the terms may take.

the requests can be submitted: Weekdays, etc)
ServiceLevelQbjective - the condition that must be met to satisfy the guarantee. E.g. Needs 128 MB of memory available

Agreement Template



Agreement Initiator

Agreement Provider

GetResourceProperty("AgreementFactory", "wsag:AgreementFactoryProperties")

(templates)

Based on the Template create an offer

CreateAgreement(offer)

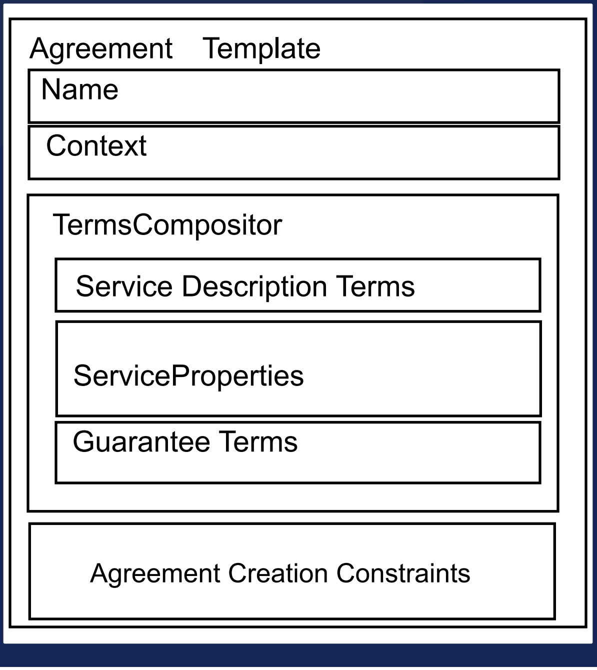
Decide to agree To the offer Create an agreement

(EPR to Agreement1)

Agreement1

At this moment the agreement is Observed. A-I cannot refuse to the Created Agreement

To create an agreement, a client makes an offer to an agreement factory. An agreement creation offer has the same structure as an agreement. The agreement factory advertises the types of offers it is willing to accept by means of agreement templates.



Agreement Template

Name

Context

TermsCompositor

Service Description Terms

ServiceProperties

Guarantee Terms

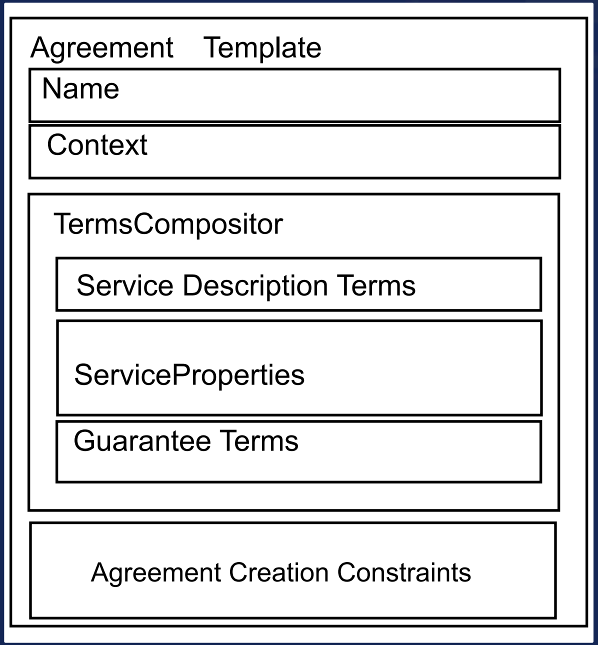
Agreement Creation Constraints

The structure of an agreement template is the same as that of an agreement, but MAY also contain a creation constraint section, i.e. a section with constraints on possible values of terms for creating an agreement.

Agreement Creation Constraints

Creation Constraints are composed of a number of offer items and constraints:

```
<wsag:template>
               …
  <wsag:CreationConstraints> ?
    <wsag:Item>…</wsag:Item> *
    <wsag:Constraint>…</wsag:Constraint> *
  </wsag:CreationConstraints>
```



Agreement Template

Name

Context

TermsCompositor

Service Description Terms

ServiceProperties

Guarantee Terms

Agreement Creation Constraints

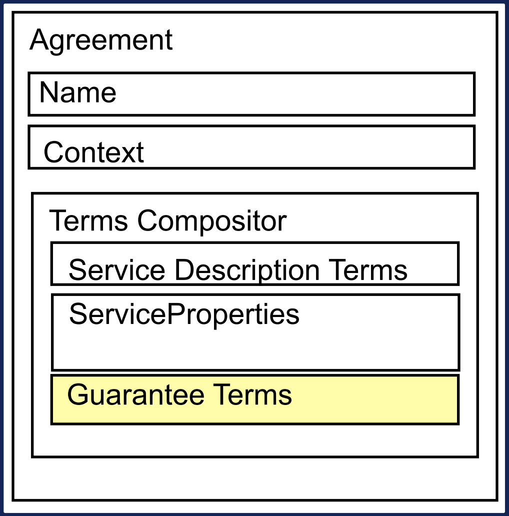
Item: Specifies that a particular field of the agreement must be present with a value in the agreement offer, and which values are possible.
Constraint: Restrict the possible values of a term.

Guarantee Terms

Define the assurance on service quality, associated with the service described by the service definition terms.

A guarantee term is composed of

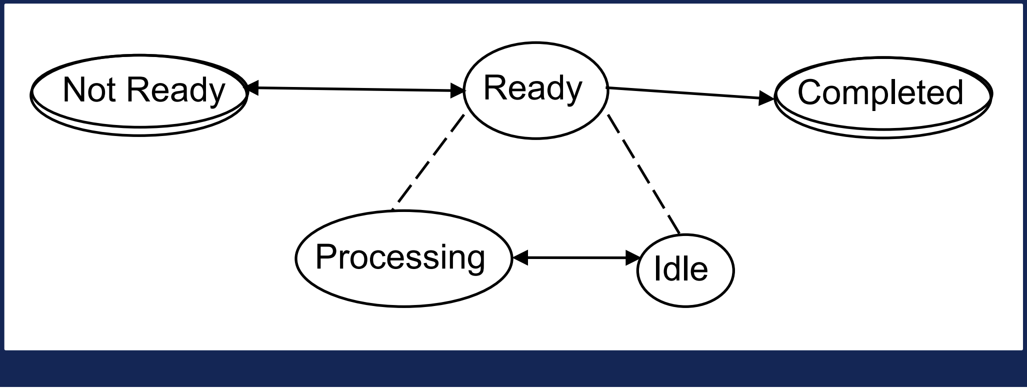
- Obligated
  the obligated party (ServiceConsumer or ServiceProvider)
- ServiceScope
  the list of services this guarantee applies to.
- QualifyingCondition
  an optional condition that must be met (when specified) for a guarantee to be enforced.
- ServiceLevelObjective
  an assertion expressed over service descriptions.
- BusinessValueList
- one or more business values associated with this objective



Monitoring Service Runtime States

Service Runtime States

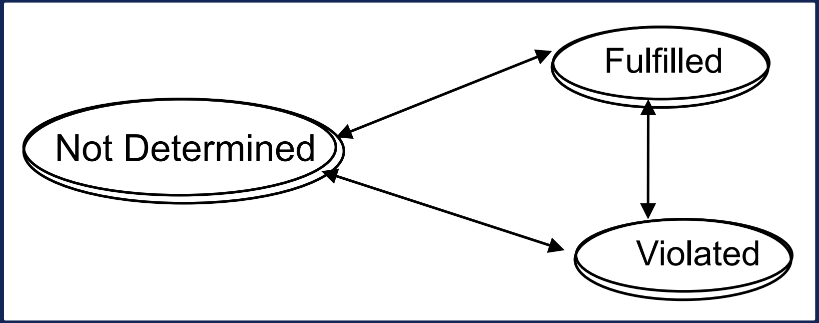
- Not Ready, Ready and Completed are the normative primary states of a service description term.
- Not Ready – The service cannot be used yet.
- Ready – The service can start now to be used by a client or to be executed by the service provider.
- Processing – The service is ready and currently processing a request or is otherwise active.
- Idle – The service is ready, however currently not being used.
- Completed – The service cannot be used any more and any service provider activity performing a job is finished. This state does not express whether an execution of a job or service was successful.



Monitoring Guarantee Runtime States

Guarantee States

- Fulfilled – Currently the guarantee is fulfilled.
- Violated – Currently the guarantee is violated.
- NotDetermined – No activity regarding this guarantee has happened yet or is currently happening that allows evaluating whether the guarantee is met.



Using SLA for Resource Management and Scheduling - CoreGRID 2007 Survey

Systems using WS-Agreement

- VIOLA MetaScheduling Service (VIOLA project)
- AssessGrid Broker (AssessGrid project)
- ASKALON (Uni. of Innsbruck)
- Community Scheduler Framework (Platform; Jilin Uni.)
- AgentScape (Vrije Uni. Amsterdam)
- CATNETS (CATNETS project)
- Job Submission Service (Umeå University)

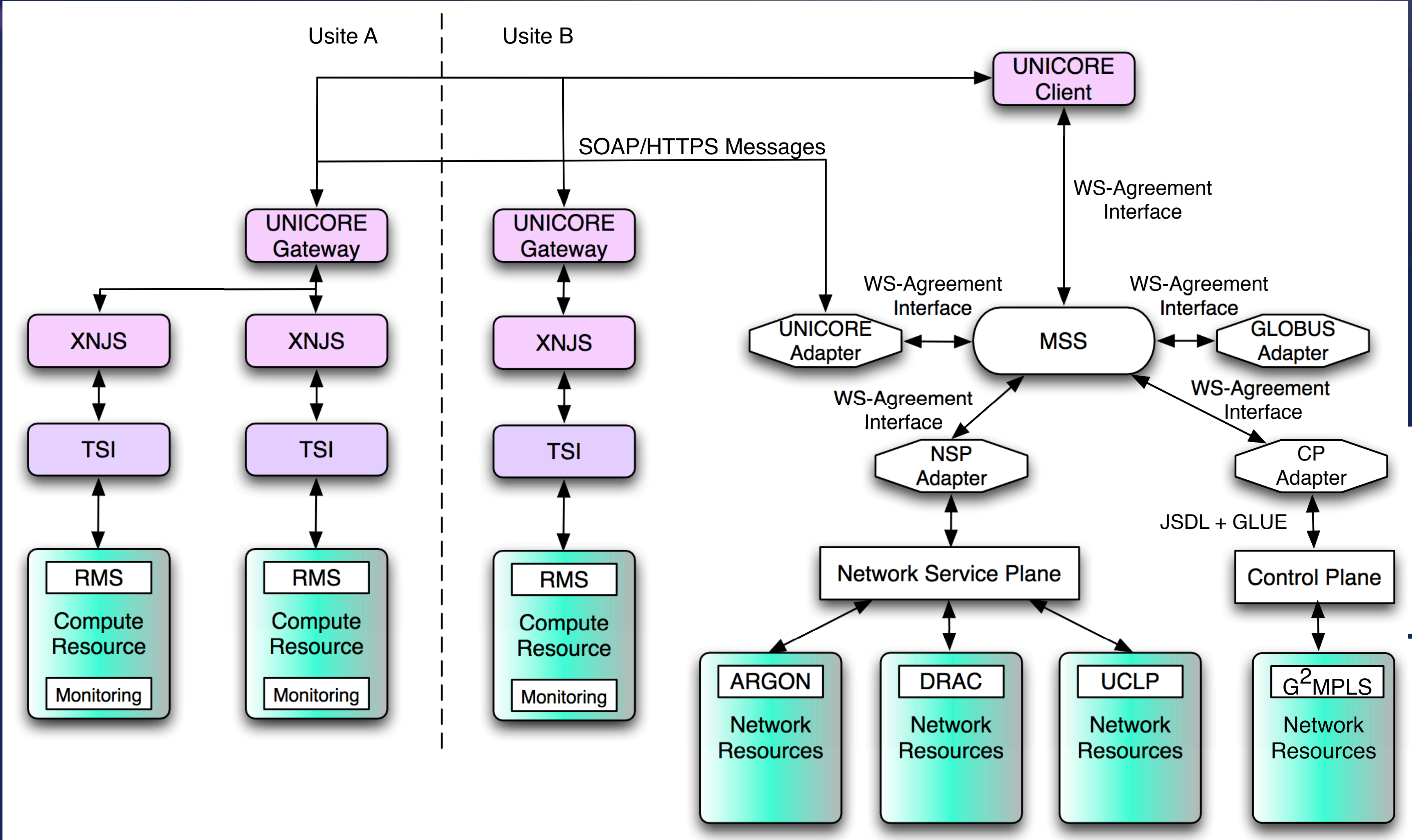
Systems planning to use WS-Agreement

- Grid Resource Management System (PSNC)
- GridWay (Uni. of Madrid)

Systems not yet decided on the SLA technology

- eNanos (BSC)
- Grid superscalar (BSC)

PHOSPHORUS MetaScheduling Service



Evaluation of usage of WS-Agreement

SLA description format

- Seems to fulfil most of the use cases' requirements
- Standardisation of domain-specific attributes would be beneficial (interoperability)
- Obviously missing:

SLA negotiation protocol

- Too simple for many scenarios (re-negotiation, bidding, ...)
- Many efforts and different approaches
- Task for GRAAP-WG: Co-ordinate negotiation discussion

Potential approach for WS-Agreement > V1.0

- Separation of SLA description and protocol?

Use-Cases for Negotiation and Re-Negotiation

Agreement on multiple QoS Parameters

- SLAs might be used for co-allocation or the resource allocation for workflows or distibuted applications
  - Allocation of multiple computational resources together with the network links between them
  - Workflow distributed across several resources or composed of services hosted at different sites

Grid Scheduler interoperation

- No single orchestration service or Grid Scheduler across multiple administrative domains
  - Multiple instances needing to interact

Existing Agreement needs to be modified

- Need of more or less services during run-time
- Service usage time to reduced or extended

Conclusion

WS-Agreement