

Introduzione al Grid Computing

Paolo Mori

Scaletta

- Definizione di griglia computazionale
- OGSA & OGSF
- Globus toolkit

Griglia Computazionale

- “A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.”

I. Foster and C. Kesselman: *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufman, USA, 1998

Griglia Computazionale (II)

- “...Grid concept is coordinated resource sharing and problem solving in a dynamic, multi-institutional virtual organizations.”

I. Foster, C. Kesselman and S. Tuecke: *The anatomy of the Grid*. International Journal of Supercomputer Applications, 15(3), 2001

Griglia Computazionale (III)

- “A Grid is a system that:
 - Coordinates resources that are not subject to centralized control
 - using standard, open, general-purpose protocols and interfaces
 - to deliver nontrivial qualities of service”

I. Foster: *What is the Grid? A Three Point Checklist*, 2002

Griglia Computazionale (IV)

- Insieme molto ampio e dinamico di entità distinte e geograficamente distribuite che condividono risorse eterogenee per la risoluzione coordinata di problemi

Griglia Computazionale (IV)

- Insieme molto ampio e dinamico di **entità distinte** e geograficamente distribuite che condividono risorse eterogenee per la risoluzione coordinata di problemi

Entità

- Le entità che partecipano alla griglia computazionale sono di vario tipo:
 - Università
 - Istituti di ricerca
 - Aziende
 - Privati
 -
- Le entità appartengono a domini distinti
 - Teams di amministratori distinti
 - Strumenti di gestione delle risorse distinti
 - Politiche di gestione delle risorse distinte

Virtual Organization (VO)

- Insieme dinamico di entità che condividono risorse
 - Diverso scopo
 - Esperimento di fisica/chimica.....
 - Diversa durata
 - Breve
 - Lunga
 - Diversa dimensione
 - Gruppo di molti ricercatori che eseguono un esperimento
 - Gruppo di alcuni studenti che fanno delle prove
- La stessa entità può appartenere a più Virtual Organizations (VOs) simultaneamente

Griglia Computazionale (IV)

- Insieme molto ampio e **dinamico** di entità distinte e geograficamente distribuite che condividono risorse eterogenee per la risoluzione coordinata di problemi

VO = Insieme Dinamico

- Durante la vita della Virtual Organization
 - Nuove entità possono aggiungersi alla VO
 - Alcune entità possono lasciare la VO
- Le entità appartenenti ad una VO sono sconosciute a priori
- Non esistono a priori relazioni di fiducia tra entità

Griglia Computazionale (IV)

- Insieme molto ampio e dinamico di entità distinte e geograficamente distribuite che **condividono** risorse eterogenee per la risoluzione coordinata di problemi

Condivisione

- Ogni entità **mette a disposizione** della griglia computazionale un insieme di risorse locali
- &
- Ogni entità **utilizza** le risorse della griglia computazionale combinandole nel modo più opportuno per risolvere i propri problemi



Ogni entità agisce sia da client che da server
(come nel peer to peer)

Griglia Computazionale (IV)

- Insieme molto ampio e dinamico di entità distinte e geograficamente distribuite che condividono **risorse eterogenee** per la risoluzione coordinata di problemi

Risorse Condivise

- Dispositivi eterogenei:
 - Risorse Computazionali
 - Computers
 - SuperComputers
 - Data Bases
 - Software Repositories
 - Storage
 - Risorse speciali
 - Sensori
 - Centrifughe
 - Simulatori vari (bacino ad onde, tavola sismica)

Risorse Condivise (II)

- Ambienti eterogenei (anche tra dispositivi dello stesso tipo)
 - OS (Unix, Linux, Windows, embedded OSs)
 - Ambienti di sviluppo (J2EE, .NET)
 - Software

Griglia Computazionale (III)

- Insieme molto ampio e dinamico di entità distinte e geograficamente distribuite che condividono risorse eterogenee per la **risoluzione coordinata di problemi**

Risoluzione Coordinata di Problemi

- Alcuni esperimenti richiedono la cooperazione di più scienziati perchè sono
 - Complessi
 - Interdisciplinari
- Le dimensioni di alcuni problemi sono troppo grandi anche per i supercomputers
 - Dimensione dei dati
 - Carico computazionale
- Le risorse necessarie ad alcuni esperimenti sono
 - Molto numerose
 - Fisicamente distanti

Applicazioni

- **Computazionalmente intensive**
 - Simulazioni Interattive (modelli del clima)
 - Simulazioni su larghissima scala (astrofisica, chimica computazionale, terremoti)
 - Ingegneristiche (parameter studies, linked component models)
- **Dimensione dei dati**
 - Analisi di dati sperimentali (fisica delle alte energie)
 - Anlasi di immagini (astronomia, studi climatici, ecologia)
- **Collaborazione**
 - Strumentazioni online (microscopi, dispositivi a raggi X, etc.)
 - Visualizzazione remota (studi climatici, biologia)
 - Ingegneristiche (test strutturali su larga scala, ingegneria chimica)

Evoluzione Storica

- Supercalcolatori



- Cluster di workstations



- Metacomputing

- Grid Computing

Convergenza verso la Griglia di:

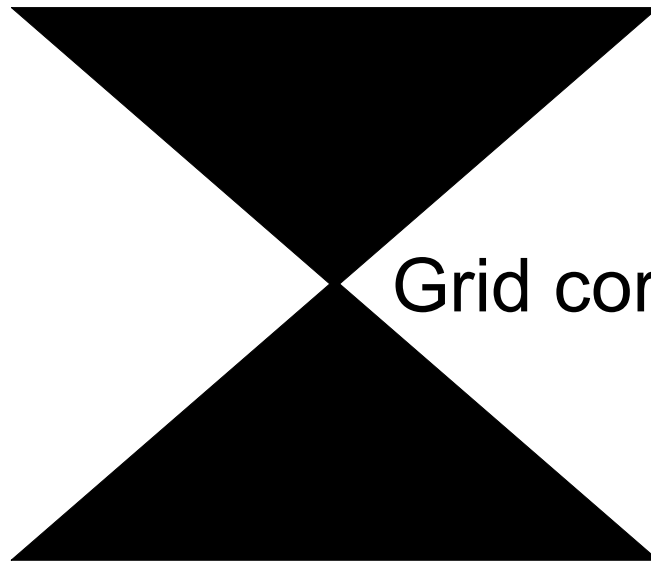
- Web: molti utenti possono sfruttare molte risorse geograficamente distribuite con un protocollo comune (non client-server)
- Peer to Peer: condivisione molti a molti di risorse multiple (non solo uno ad uno e file sharing)
- Calcolo distribuito: sfrutta risorse
 - Differenti
 - Eterogee
 - Geograficamente distribuite

Problemi Implementativi

- Interoperabilità: come garantire che due entità che vogliono condividere risorse riescano a interfacciarsi tra loro?
- Integrazione: come integrare la griglia con i meccanismi e gli strumenti già presenti sulle risorse da condividere?
- Sicurezza: come proteggere
 - le risorse dagli utenti?
 - gli utenti dalle risorse?

Principio della Clessidra

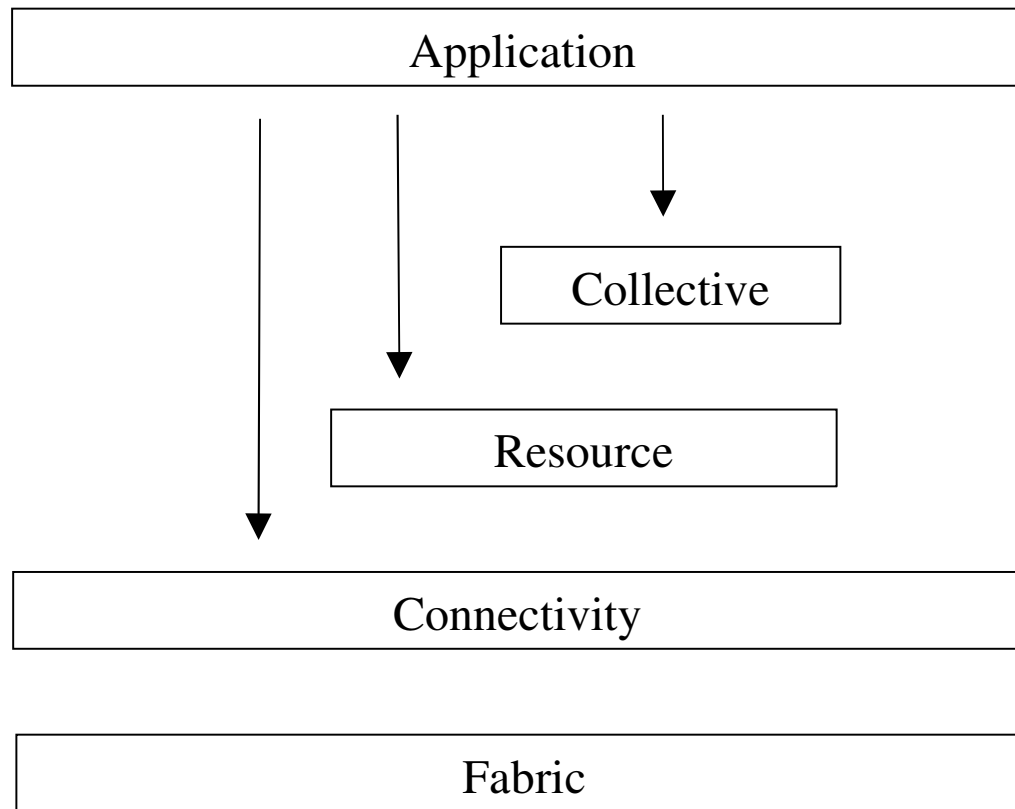
Applicazioni varie



Grid core services

Risorse ed ambienti eterogenei

Architettura Grid



Architettura Grid (II)

- Fabric: risorse da condividere
- Connectivity: protocolli di comunicazione e di sicurezza necessari per le transazioni con le risorse
- Resource: gestione della singola risorsa
 - Gestione allocazione risorsa
 - Accesso e trasferimento dati
 - Stato della risorsa
- Collective: coordinamento delle risorse
 - Ricerca delle risorse con certe proprietà
 - Co-allocazione (allocazione di più risorse)
 - Replica dati
- Application: applicazioni finali

Open Grid Service Architecture

- Documento redatto da:

Global Grid Forum (<http://www.ggf.org>)



- Scaricabile da:

<http://forge.gridforum.org/projects/ogsa-wg>

- Descrive l'architettura di riferimento per la definizione di griglie computazionali

Requisiti OGSA:

Interoperabilità e supporto per ambienti dinamici ed eterogenei

- Virtualizzazione delle risorse: su ogni risorsa si costruisce una risorsa virtuale (servizio)
- Strumenti di gestione comuni: permettono di gestire qualsiasi risorsa nello stesso modo
- Ricerca di risorse: sono necessari meccanismi per localizzare le risorse a partire dalle loro caratteristiche
- Protocolli e schemi standard: permettono l'intefacciamento tra le varie risorse

Requisiti OGSA:

Condivisione risorse tra organizzazioni distinte

- Spazio dei nomi globale: permette di individuare univocamente una risorsa sulla griglia
- Servizio metadati: permette di trovare e invocare le entità
- Autonomia dei singoli nodi: i meccanismi di condivisione devono rispettare le impostazioni locali
- Dati sull'utilizzo delle risorse: meccanismi standard per rilevare e gestire i dati relativi all'uso delle risorse per pagamenti, tracciabilità delle azioni e/o reputazione

Requisiti OGSA:

Qualità del Servizio (QoS)

- **Accordo sul livello di servizio:** occorrono meccanismi per la contrattazione della qualità del servizio richiesto tra utente e fornitore della risorse
- **Rispetto del livello di servizio:** occorrono meccanismi per monitorare e garantire l'effettiva qualità del servizio erogata dal fornitore delle risorse
- **Migrazione dei lavori:** occorrono meccanismi per migrare i lavori da una risorsa all'altra

Requisiti OGSA:

Esecuzione dei lavori (jobs)

- **Supporto per vari tipi di job:** occorre poter eseguire sia job semplici che composti, come i workflows
- **Gestione dei job:** occorrono meccanismi per gestire i vari step dell'esecuzione di un job
- **Schedulazione dei job:** occorrono meccanismi per decidere quali job eseguire e su quali risorse eseguirli
- **Preparazione della risorsa:** occorre automatizzare l'allocazione della risorsa e la sua configurazione per l'esecuzione del job

Requisiti OGSA:

Data Services

- **Accesso ai dati:** occorrono meccanismi per garantire l'accesso ai dati indipendentemente dalla posizione fisica e dalle caratteristiche della risorsa che li ospita
- **Consistenza dei dati:** occorrono meccanismi per garantire la consistenza dei dati in cache o replicati tra più risorse
- **Persistenza dei dati:** i dati devono essere mantenuti sulle risorse fino a che necessario
- **Integrazione dei dati:** occorrono meccanismi per integrare dati eterogenei e distribuiti
- **Gestione della posizione dei dati:** occorrono meccanismi per trasferimento, copia, o caching dei dati

OGSA capability:

Service Oriented Architecture

- Le risorse implementano **servizi**
- Ogni servizio ha una interfaccia
- Ogni servizio ha uno stato
- Interazioni tra i servizi



- Web Services
 - Web Service Description Language (interfaccia)
 - Web Service Resource Framework (stato)

OGSA capability:

Service Oriented Architecture

Web Services



- **Servizi dinamici:** nuovi grid services possono aggiungersi e alcuni grid services esistenti possono scomparire dinamicamente
- **Stato:** i grid service hanno uno stato che deve essere conservato nel tempo
- **Sottoscrizioni e callbacks:** gli utenti possono sottoscrivere i servizi e devono essere notificati dagli stessi quando terminano

Open Grid Service Infrastructure

- Documento redatto da:

Global Grid Forum (<http://www.ggf.org>)



- Scaricabile da:

www.ggf.org/documents/GWD-R/GFD-R.015.pdf

- Specifiche per creazione, gestione, ricerca ed interazione dei/tra grid services
 - linguaggio per descrivere i grid services (GWSDL) estensione di quello per i web services (WSDL)

Grid Toolkits

- Globus (<http://www.globus.org>)
- Legion (<http://legion.virginia.edu>)
- WebOS (<http://www.cs.duke.edu/ari/issg/webos/>)
- PlanetLab (<http://www.planet-lab.org/>)
- Unicore (<http://www.unicore.org/>)
-

Globus toolkit

- The open source Globus Toolkit ® is a fundamental enabling technology for the "Grid," letting people share computing power, databases, and other tools securely online across corporate, institutional, and geographic boundaries without sacrificing local autonomy.
- The toolkit includes software services and libraries for resource monitoring, discovery, and management, plus security and file management.
- Versione corrente: GT4

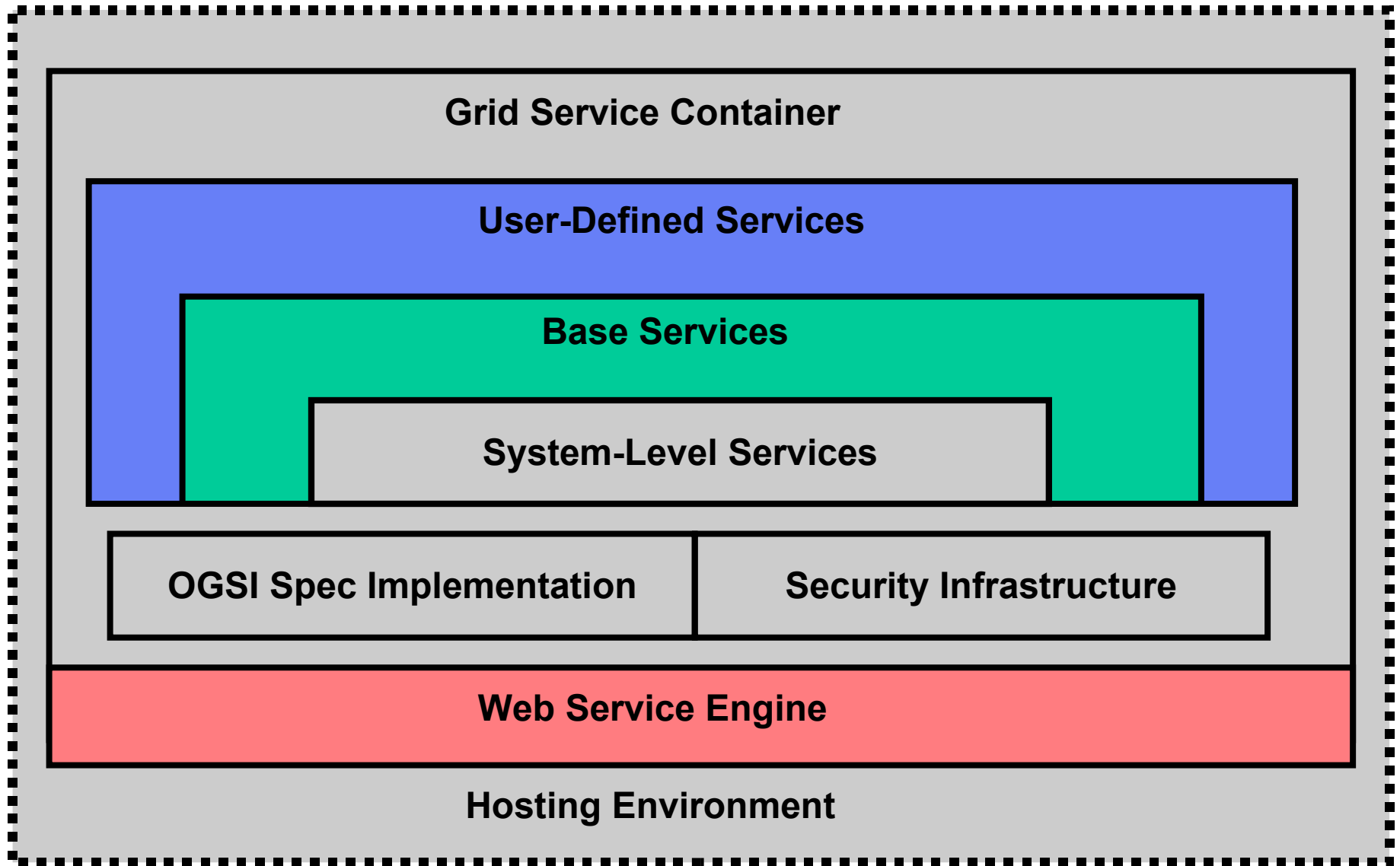
<http://www.globus.org/toolkit/about.html>

Globus toolkit

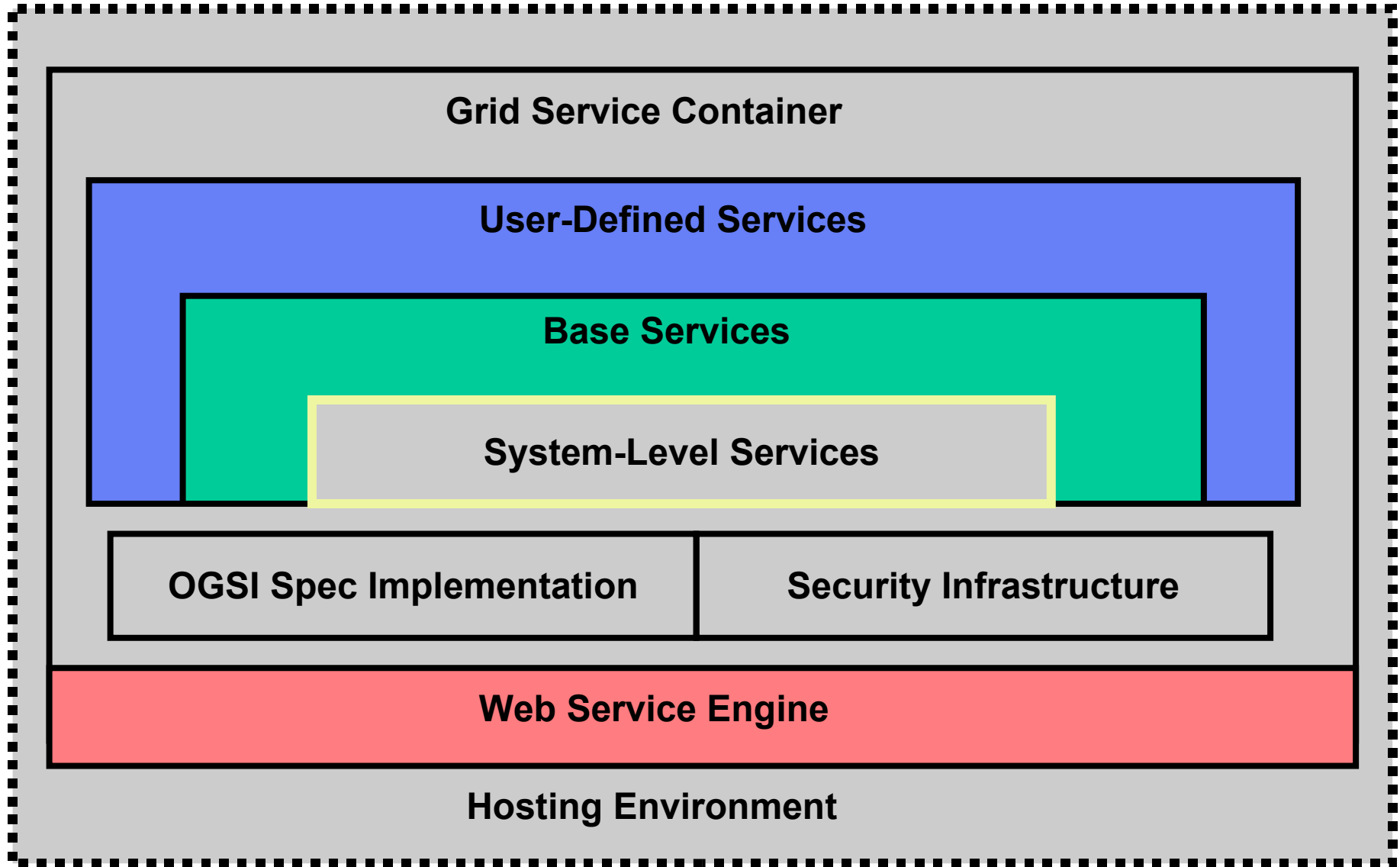
- Strumento per lo sviluppo delle applicazioni di griglia composto da:
 - Servizi di base e di sistema
 - Strumenti per sviluppare nuovi servizi
 - Infrastruttura di sicurezza
- Rispetta gli standard:
 - OGSA
 - OGSF

I. Foster: *Globus Toolkit Version 4: Software for Service-Oriented Systems*. IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, 2005

Architettura dei servizi Globus



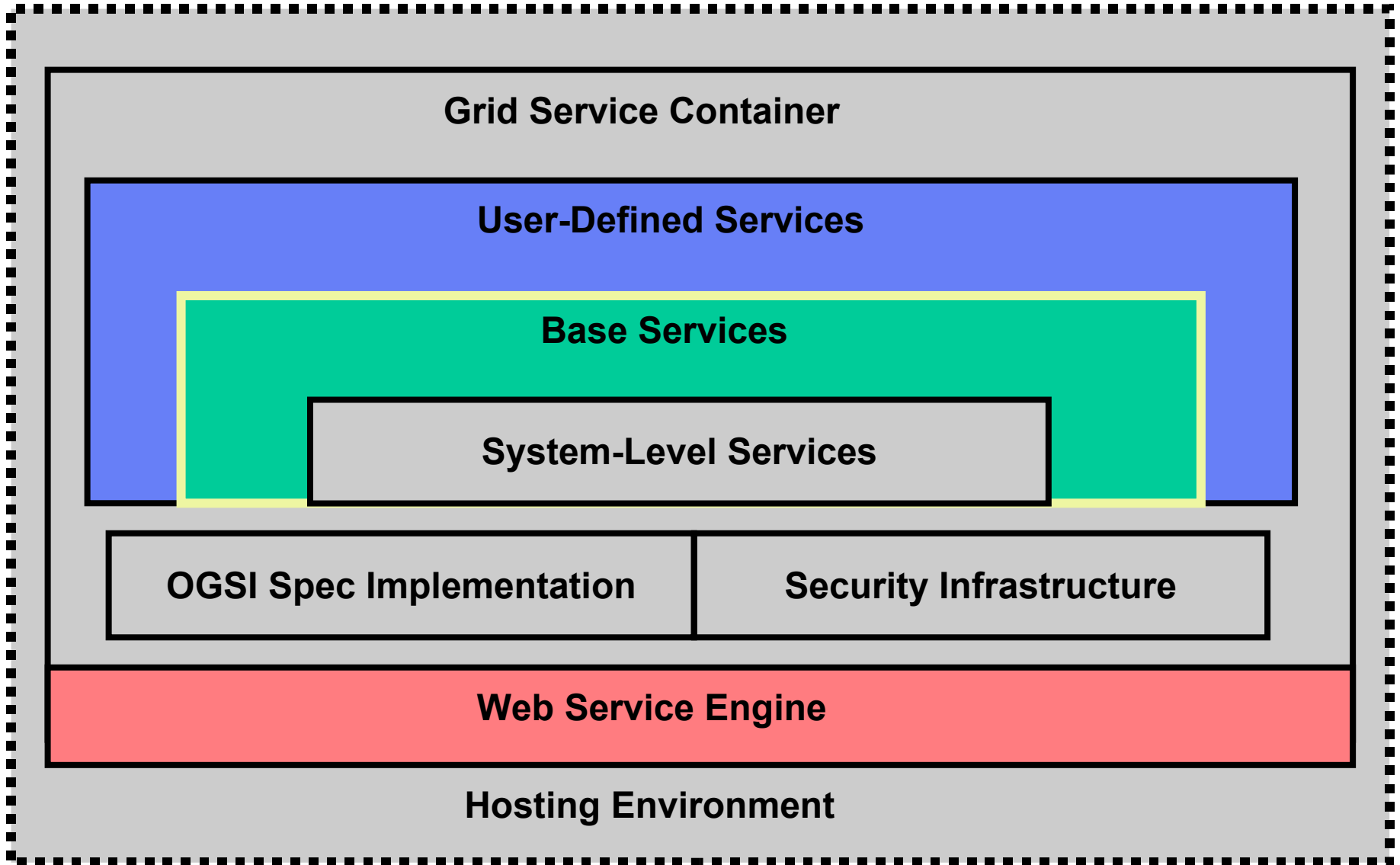
Architettura dei servizi Globus



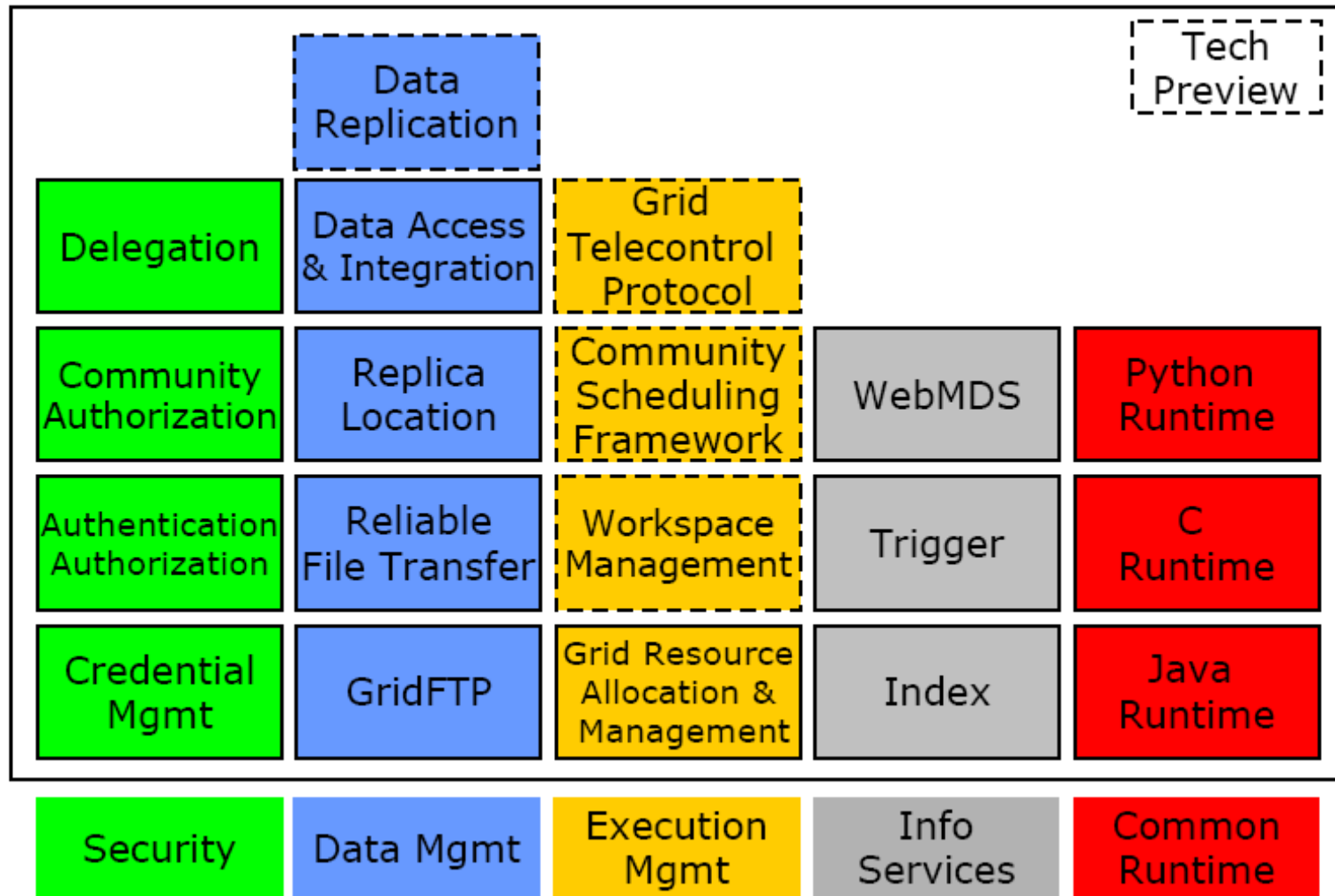
Servizi di sistema

- Servizi di carattere generale che facilitano l'uso dei servizi di griglia di livello superiore
 - Servizio di amministrazione
 - Ping e arresto del container
 - Servizio di log
 - Controllo degli eventi sul servizio
 - Servizio di gestione
 - Monitoraggio dello stato e del carico
 - Attivazione e stop di istanze di servizi

Architettura dei servizi Globus



Servizi base di Globus



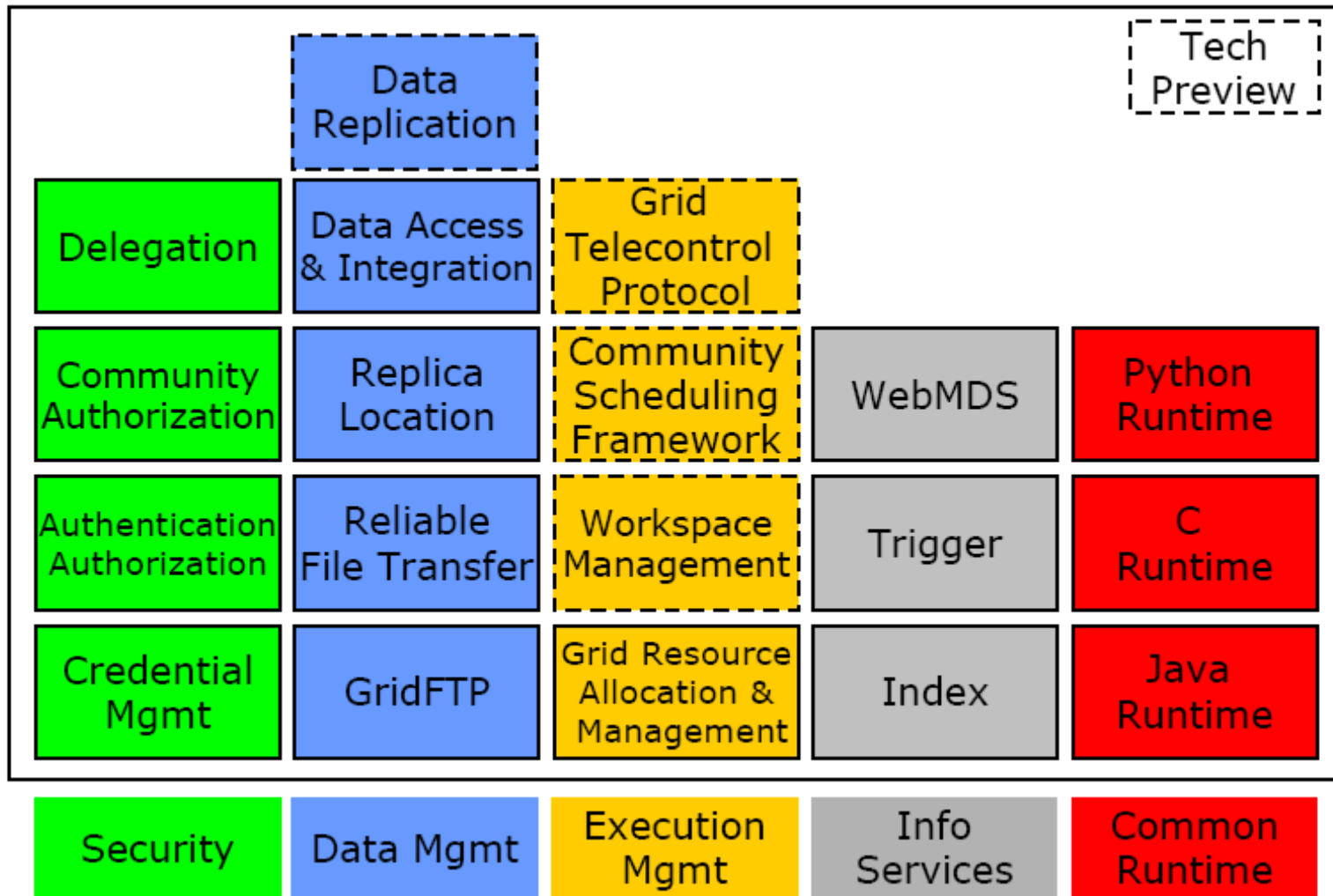
Information Service

- Monitor and Discovery System
 - Ricerca dei servizi sulla griglia: identifica i servizi sulla griglia con le proprietà specificate
 - Monitoring dei servizi sulla griglia: controlla lo stato dei servizi sulla griglia diagnosticando i problemi
- Fornisce due tipi di servizio:
 - Index service: raccoglie informazioni dai servizi e le pubblica per rispondere alle query
 - Trigger service: raccoglie informazioni dai servizi e quando un insieme di condizioni prestabilite sono verificate, esegue delle azioni prestabilite

Esempi di informazioni

- Caratteristiche della risorsa computazionale
 - OS
 - Tipo e numero CPU disponibili
 - Software installati
 - Carico computazionale
 - Memoria disponibile
- Caratteristiche di una rete
 - Banda, latenza
 - Protocolli supportati

Servizi base di Globus



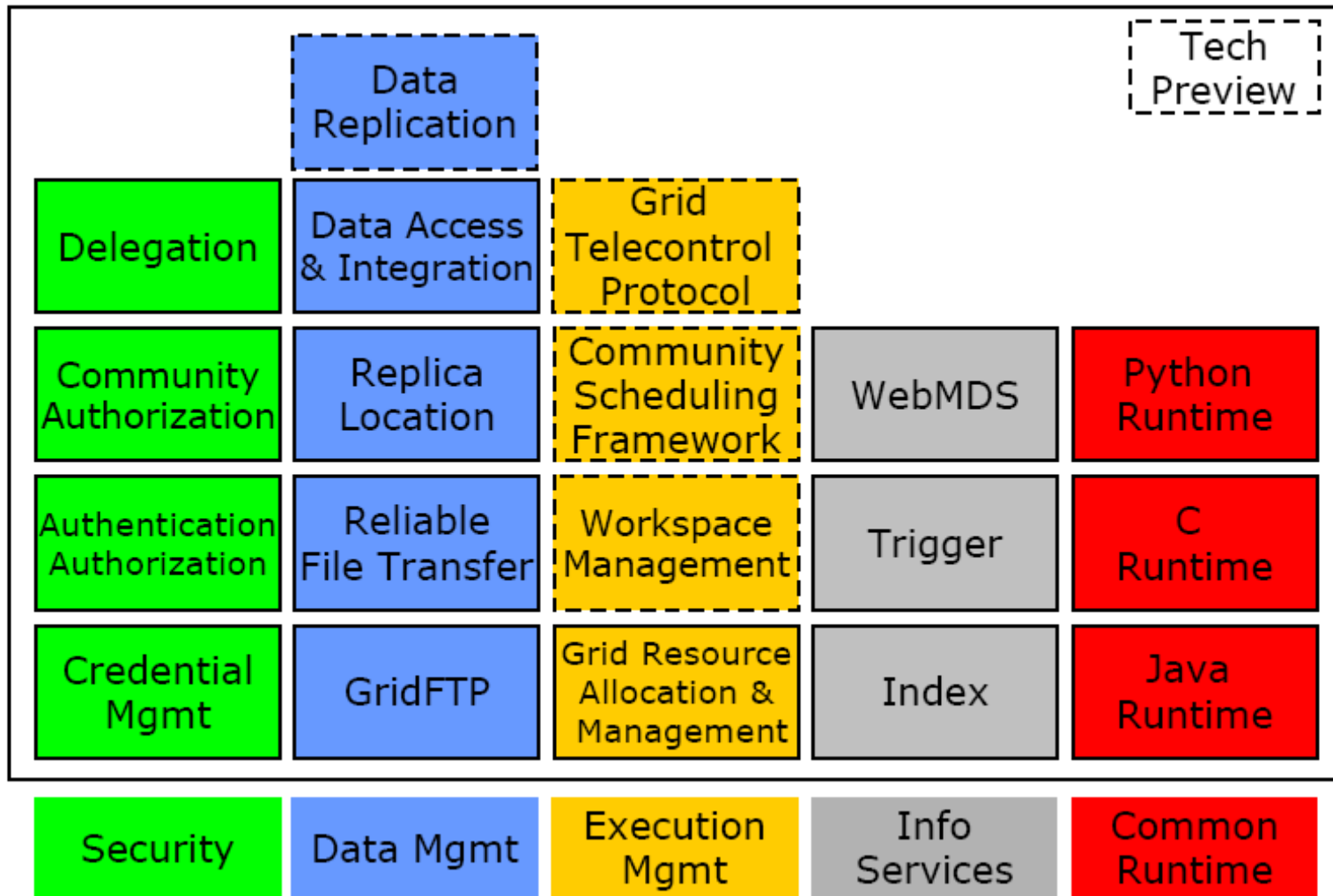
Data Management

- GridFTP: protocollo di trasferimento dati tra risorse di griglia
 - Alte prestazioni
 - Sicuro
 - Affidabile
 - Basato su FTP
- Reliable File Transfer: servizio di trasferimento dati tra risorse di griglia
 - basato su Web Services
 - Più robusto di GridFTP

Data Management

- Replica Location Service: servizio di gestione di copie multiple dei dati
 - Ogni file (file logico) è replicato su più risorse (file fisici)
 - Implementa la corrispondenza file logico -> file fisici
 - Risponde alle query per individuare la copia più vicina di un file logico

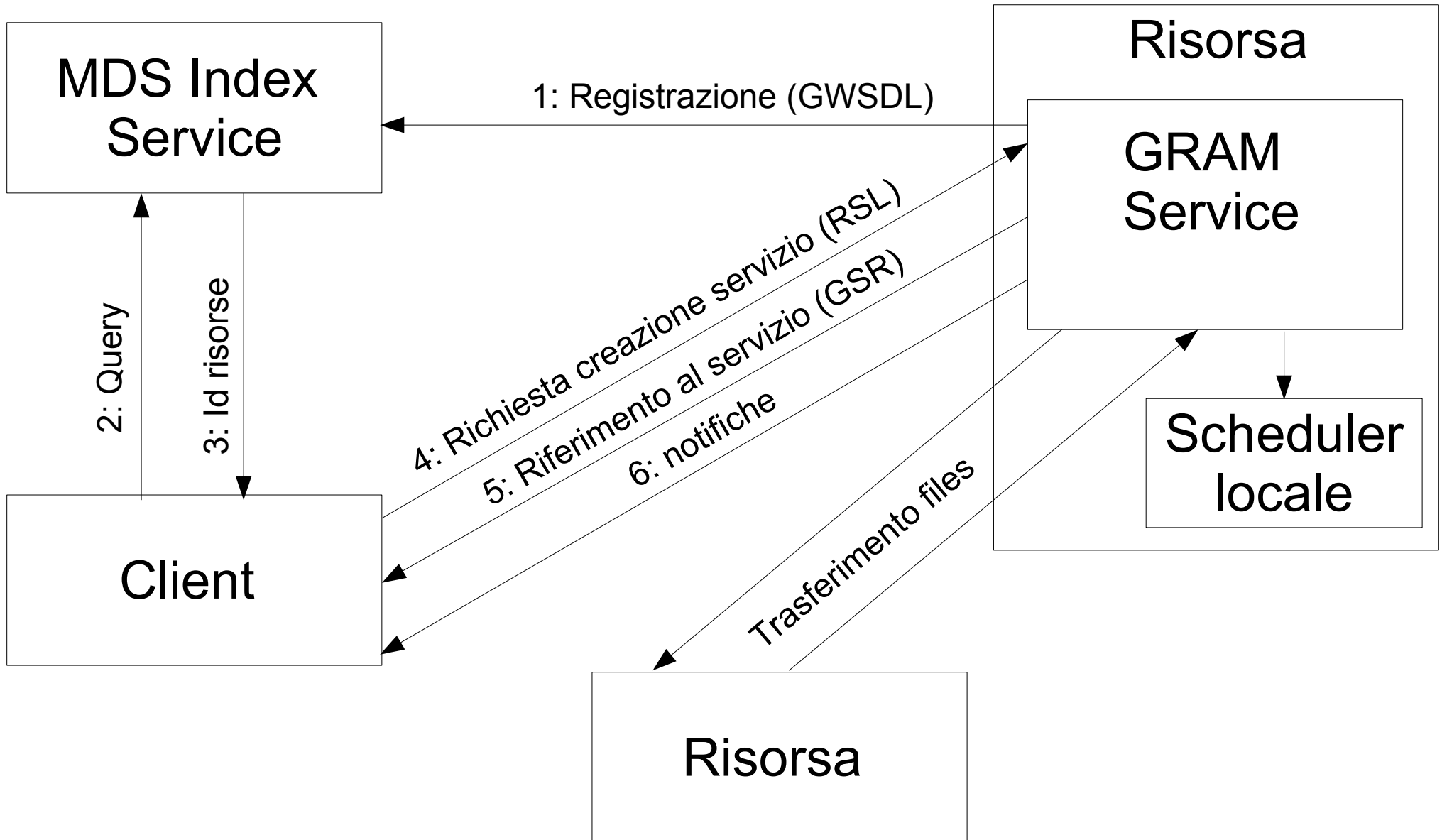
Servizi base di Globus



Grid Resource Allocation & Management

- Servizio che gestisce le risorse computazionali
- Data una richiesta di esecuzione job (in RSL)
 - Crea un ambiente per il nuovo job
 - Trasferisce i files nel / dal nuovo ambiente
 - Sottomette il job allo scheduler della risorsa locale
 - Monitora lo stato di avanzamento del job
 - Spedifica le notifiche al client
 - Gestisce gli stream di I/O del job

Sottomissione di un Job



Resource Specification Language

- Linguaggio per descrivere la richiesta di servizio
- Definito da uno schema XML
- Insieme di attributi che descrivono:
 - Configurazione del job
 - Limiti utilizzo della risorsa

Resource Specification Language (II)

- Configurazione del job (alcune)
 - <gram:executable>
 - Applicazione da eseguire
 - <gram:directory>
 - Directory dove eseguire l'applicazione
 - <gram:arguments>
 - Parametri da passare all'applicazione
 - <gram:environment>
 - Variabili di ambiente
 - <gram:stdin>, <gram:stdout>, <gram:stderr>
 - Input, output e error output dell'applicazione
 -

Resource Specification Language (III)

- Limiti utilizzo della risorsa (alcuni)
 - <gram:maxMemory>
 - Massima quantità di memoria utilizzabile
 - <gram:maxWallTime>
 - Tempo massimo di esecuzione (totale)
 - <gram:maxCPUTime>
 - Tempo massimo di CPU
 -

Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<rsl:rsl xmlns:rsl="http://www.globus.org/namespaces/2004/02/rsl"
.....
<gram:job>
  <gram:executable> <rsl:path>
    <rsl:stringElement value="prova_eseg"/> </rsl:path>
  </gram:executable>
.....
  <gram:arguments>
    <rsl:stringArray>
      <rsl:string> <rsl:stringElement value="dato_mio"/> </rsl:string>
    </rsl:stringArray>
  </gram:arguments>
.....
  <gram:maxCpuTime> <rsl:long value="60"/> </gram:maxCpuTime>
.....
</gram:job>
</rsl:rsl>
```

Esempio (II)

```
<stdout>/my_directory/stdout</stdout>
```

```
.....
```

```
<fileStageIn>
```

```
<transfer>
```

```
<sourceUrl>gsiftp://gridnode2:2811/my_dir/programma1</sourceUrl>
```

```
<destinationUrl>file:///my_directory/prova_eseg</destinationUrl>
```

```
<sourceUrl>gsiftp://gridnode2:2811/my_dir/dati1</sourceUrl>
```

```
<destinationUrl>file:///my_directory/dato_mio</destinationUrl>
```

```
</transfer>
```

```
</fileStageIn>
```

```
.....
```

```
<fileStageOut>
```

```
<transfer>
```

```
<sourceUrl>file:///my_directory/stdout</sourceUrl>
```

```
<destinationUrl>gsiftp://gridnode2:2811/my_dir/stdout</destinationUrl>
```

```
</transfer>
```

```
</fileStageOut>
```

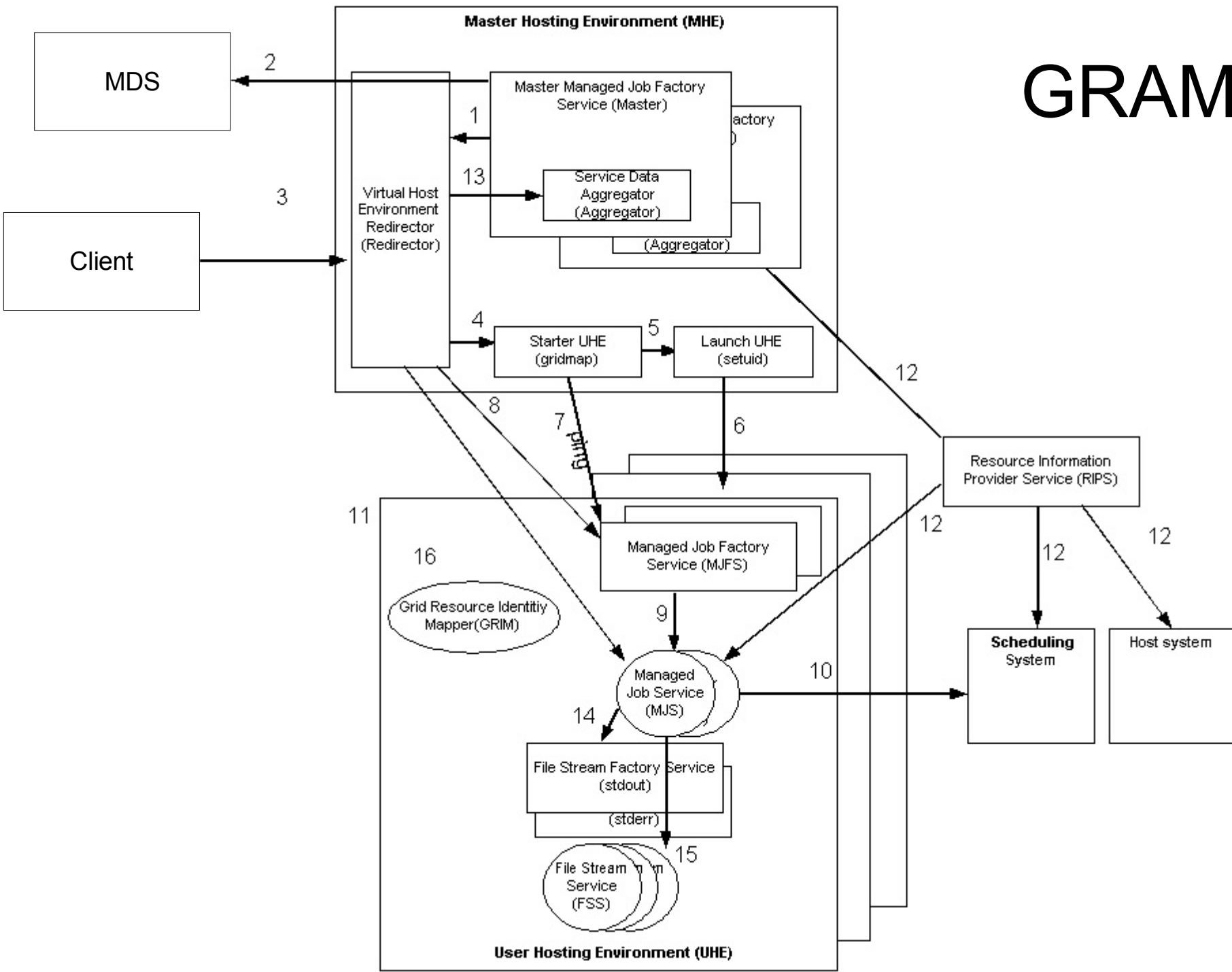
Grid Resource Allocation & Management

- Componenti:
 - Master Hosting Environment /Master Managed Job Factory Service
 - Si registra sul Grid Information Service (WSDL)
 - Riceve una richiesta da un client (RSL)
 - Crea un User Hosting Environment per ogni utente
 - User Hosting Environment: contiene:
 - Managed Job Factory Service
 - Grid Resource Identity Mapper
 - Resource Information Provider Service
 - Raccoglie dati dalla risorsa locale

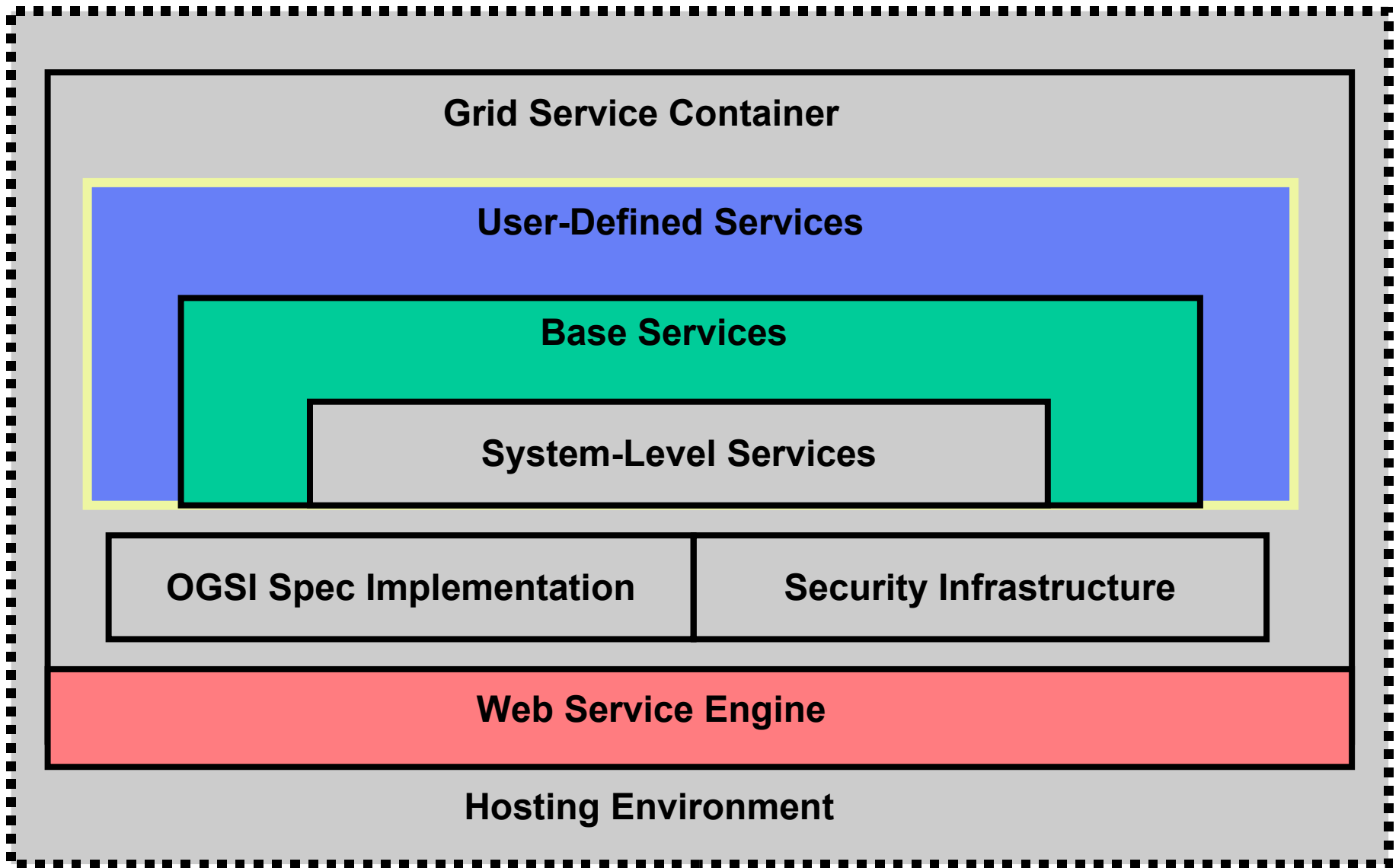
Grid Resource Allocation & Management

- Managed Job Factory Service
 - Crea tanti Managed Job Service per eseguire i jobs sottomessi
- Managed Job Service
 - Sottomette il lavoro alla risorsa locale
 - Crea i file stream services per l'I/O del job
 - Monitora lo stato dell'esecuzione
 - Spedisce le notifiche al client

GRAM



Architettura dei servizi Globus



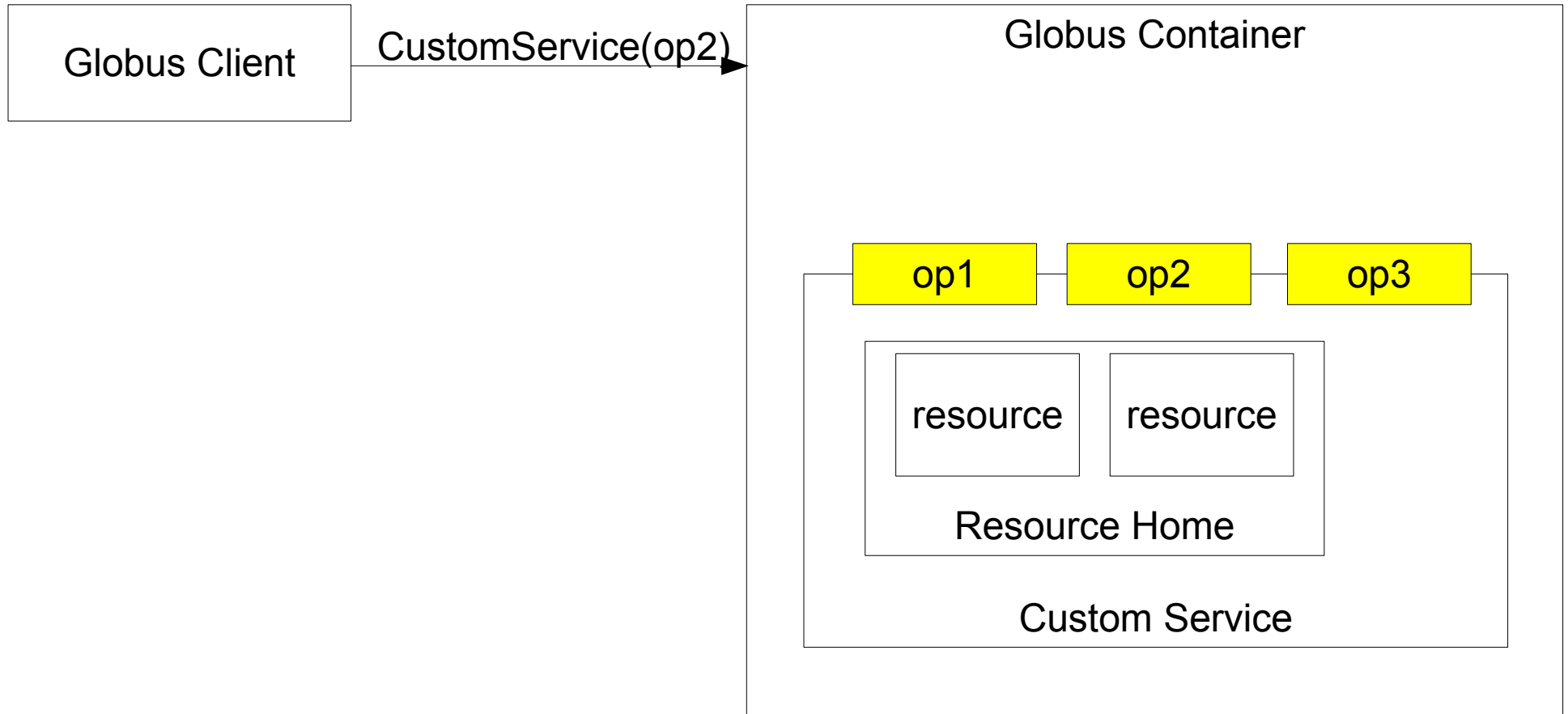
Servizi creati dall'utente

- Globus è un “Grid Service Development Kit” che fornisce allo sviluppatore:
 - Primitive per sviluppare servizi OGSI-Compliant
 - Primitive di sicurezza
 - Servizi di base su cui costruire i servizi di alto livello

Servizi creati dall'utente

- I servizi definiti dall'utente vengono eseguiti all'interno del Globus Container
 - Globus supporta servizi sviluppati in java
- I servizi hanno:
 - Operazioni: definiscono le interazioni che gli utenti possono avere con tali servizi
 - Stato (Resources): memorizza informazioni in modo persistente, che possono essere accedute da una invocazione all'altra del servizio

Servizi Creati dall'utente

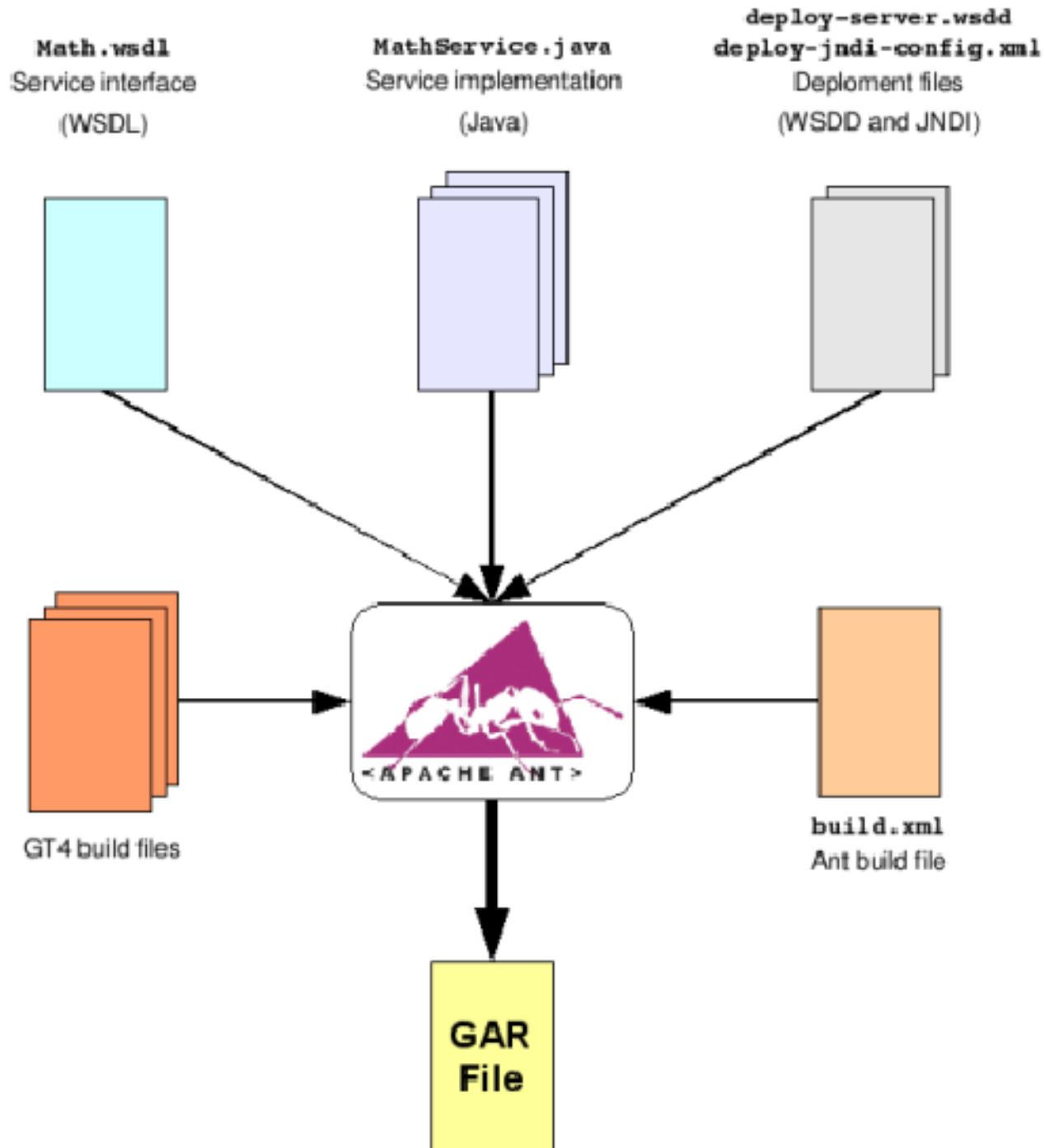


Creare un nuovo servizio Globus

Passi fondamentali:

- Definire l'interfaccia del servizio (WSDL)
 - Operazioni definite dal servizio
- Sviluppare il servizio (java)
 - Implementazione delle operazioni
- Definire i parametri di installazione del servizio in globus (WSDD e JNDI)
- Generare un gar file (Ant, build.xml)

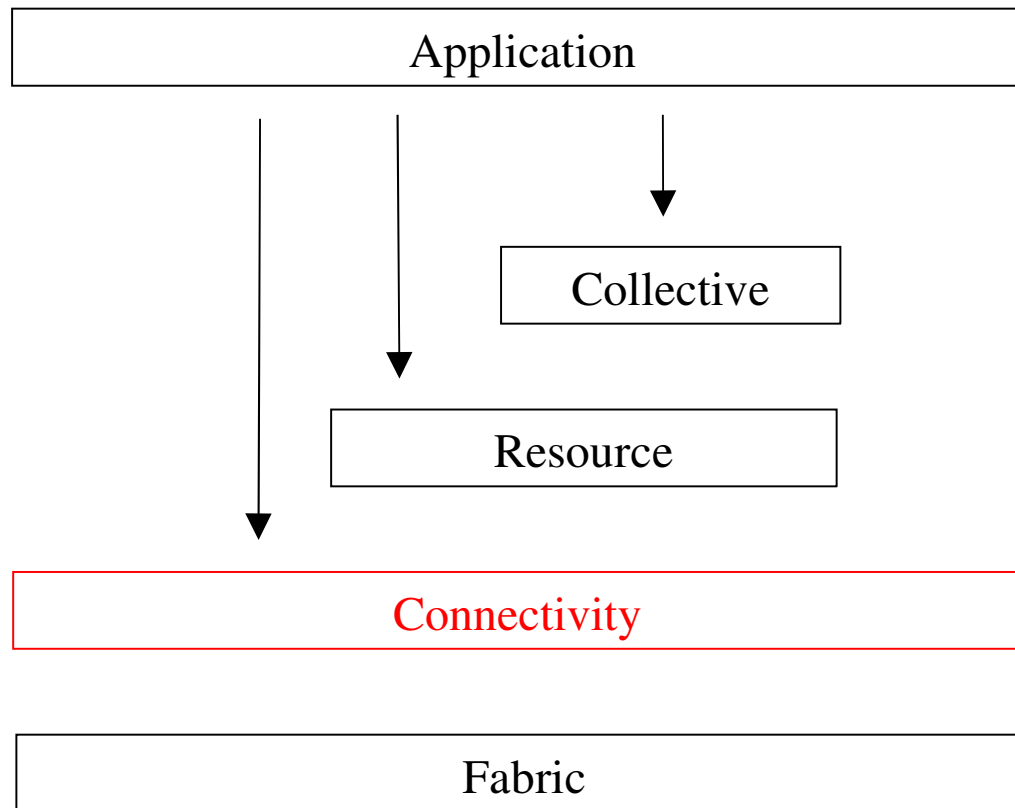
Borja Sotomayor: The Globus Toolkit 4 Programmer's Tutorial



Introduzione alla Sicurezza nel Grid Computing

Paolo Mori

Architettura Grid



Architettura Grid (II)

- Fabric: risorse da condividere
- Connectivity: **protocolli di** comunicazione e di **sicurezza** necessari per le transazioni con le risorse
- Resource: gestione della singola risorsa
 - Gestione allocazione risorsa
 - Accesso e trasferimento dati
 - Stato della risorsa
- Collective: coordinamento delle risorse
 - Ricerca delle risorse con certe proprietà
 - Co-allocazione (allocazione di più risorse)
 - Replica dati
- Application: applicazioni finali

esempio: Risorse Computazionali

- La risorsa condivisa è un computer (anche virtuale)
- Ogni utente può eseguire le proprie applicazioni utilizzando le risorse computazionali condivise
 - Utenti sconosciuti (senza relazioni di fiducia prestabilite)
 - Applicazioni sconosciute



- L'ambiente di griglia deve garantire:
 - Protezione delle risorse condivise dalle applicazioni
 - Protezione delle applicazioni dalle risorse condivise

Problemi

- Le risorse condivise appartengono a domini amministrativi diversi (siti)
- Ogni sito ha proprie impostazioni locali:
 - Politiche di sicurezza
 - Meccanismi di sicurezza (per applicare la politica)
- Non esistono relazioni di fiducia a priori tra utenti o siti
- Le VOs impongono propri requisiti di sicurezza agli utenti e siti

Security Architecture for Open Grid Services

- Documento redatto da:

Global Grid Forum (<http://www.ggf.org>)



- Scaricabile da:

http://forge.gridforum.org/projects/ogsa-sec-wg/document/Security_Architecture_for_Open_Grid_Services/en/2

- Descrive i problemi, i requisiti ed una architettura di riferimento per la sicurezza nelle griglie computazionali

Requisiti OGSA sec:

Autenticazione

- Verifica dell'identità dell'utente
- Meccanismi esistenti:
 - Login e passwd
 - Certificati digitali chiave pubblica
 - Biometrici
 -

Requisiti OGSA sec:

Single Sign-on

- L'autenticazione dell'utente deve essere eseguita una volta sola (anche se vengono acceduti più servizi)
- Il framework deve eseguire automaticamente eventuali successive autenticazioni su altre risorse
 - Politica di sicurezza
 - Limiti di tempo

Requisiti OGSA sec:

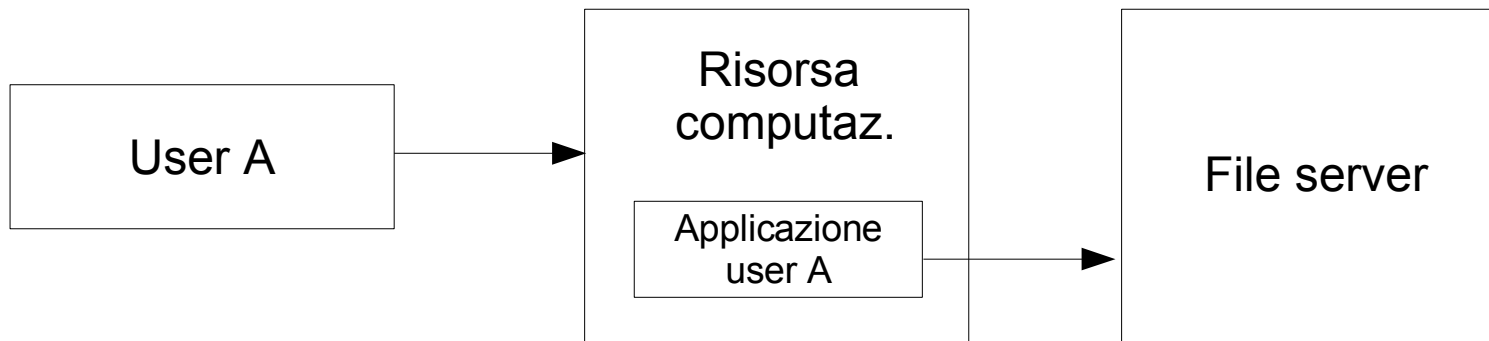
Autorizzazione

- Definisce i diritti degli utenti sulle risorse
- Meccanismi esistenti:
 - Permessi del sistema operativo
 - Permis (Role based)
 - Akenti
 - Permessi gestiti dal servizio specifico (ad esempio un data base management system)
 -

Requisiti OGSA sec:

Delega

- Trasferisce i diritti di accesso di un utente ad un altro (o ad un servizio)
- Deve essere limitata
 - Al task da eseguire per conto dell'utente delegante
 - La durata nel tempo
- Necessaria per eseguire applicazioni che interagiscono con altri servizi



Requisiti OGSA sec:

Comunicazioni

- Integrità messaggi
 - Il messaggio ricevuto è identico a quello spedito
- Confidenzialità messaggi
 - Il messaggio deve essere letto solo da utenti che ne hanno diritto
- Non Ripudio:
 - Il mittente non deve poter rinnegare il messaggio spedito
- Meccanismi esistenti:
 - Crittografia
 - Firma digitale

Requisiti OGSA sec:

Auditing e Accounting

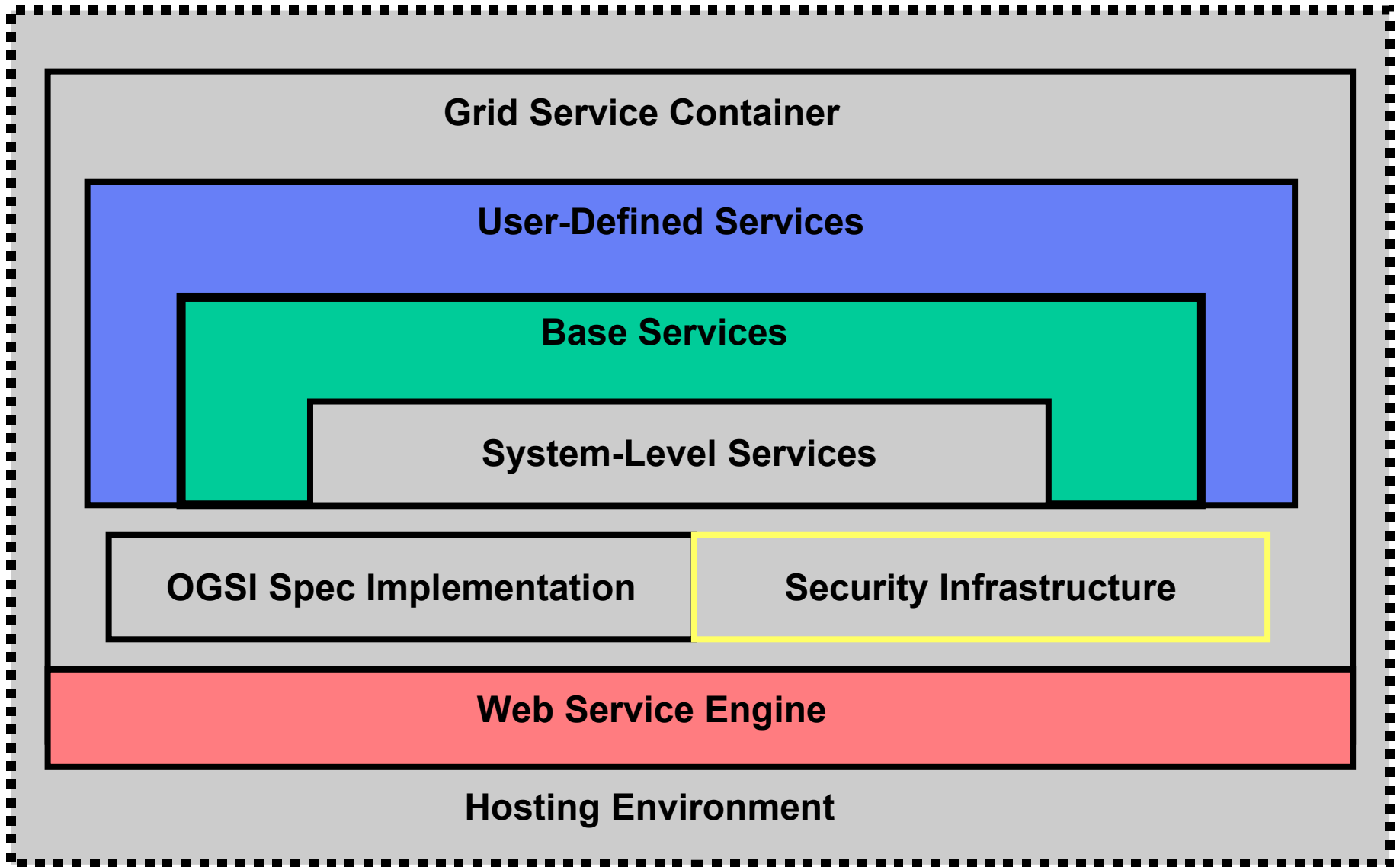
- Auditing
 - Deve essere possibile risalire agli utenti che hanno utilizzato una data risorsa in un dato momento
- Accounting
 - Deve essere possibile misurare l'utilizzo delle risorse per addebitarne il consumo agli utenti

Globus toolkit

- Globus Security Infrastructure (GSI)
- Public Key Infrastructure (PKI)
- Certificati X.509
- Autorizzazione

<http://www.globus.org/toolkit/docs/4.0/security>

Architettura dei servizi Globus



Certificati X.509

- Documenti di identità in formato digitale
- Rilasciati dalla Certification Authorities
- Contengono almeno:
 - Identità del proprietario (DN)
 - Chiave pubblica del proprietario
 - Scadenza del certificato
 - Nome della Certification Authority che l'ha rilasciato
 - Firma digitale della Certification Authority che ne garantisce l'autenticità
- Chiave Privata (segreta) associata al certificato

Esempio di Certificato X.509

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 2 (0x2)

Signature Algorithm: md5WithRSAEncryption

Issuer: O=Grid, OU=GlobusTest, OU=simpleCA-morip.iit.cnr.it, CN=Globus Simple CA

Validity

Not Before: Feb 10 17:15:53 2005 GMT

Not After : Feb 10 17:15:53 2006 GMT

Subject: O=Grid, OU=GlobusTest, OU=simpleCA-morip.iit.cnr.it, OU=iit.cnr.it,
CN=paolo mori

Esempio di Certificato X.509 (cont.)

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public Key: (1024 bit)

Modulus (1024 bit):

00:d0:83:52:d7:59:55:c7:c8:35:5a:76:37:f5:a3:

.....

cb:f6:40:e2:b6:8c:16:ad:39

Exponent: 65537 (0x10001)

X509v3 extensions:

Netscape Cert Type:

SSL Client, SSL Server, S/MIME, Object Signing

Signature Algorithm: md5WithRSAEncryption

2a:d1:ea:43:19:6d:ca:ec:da:f4:d4:b0:df:1c:d6:bf:d5:45:

.....

4d:2d:0c:8a:5f:b3:a1:ad:56:be:d5:e0:a2:ee:47:cb:7f:87:

-----BEGIN CERTIFICATE-----

MIICYDCCAcmgAwIBAgIBAjANBgkqhkiG9w0BAQQFADBjMQ0wCwYDVQQKEwRHcmllk

.....

MA6FwFNrdXKq7bNsb0lhLU0tDIpfs6GtVr7V4KLuR8t/h5O0

-----END CERTIFICATE-----

Certificati Proxy

- Utilizzati al posto dei certificati originali
- Emessi dall'utente (non dalla CA)
 - Stesso Subject Name
 - Diversa coppia chiave pubblica/privata
 - Firmati con la chiave privata del certificato originale
- Limitazioni
 - Durata molto inferiore all'originale (qualche ora invece di 1 anno)
 - Set di diritti ristretto

Utilizzo dei Certificati Proxy

- Single Sign-on
- Delegation

Autorizzazione: Grid Mapfile

- Meccanismo standard di autorizzazione di Globus
- Ogni entry del grid mapfile definisce per un DN l'account locale sul quale viene mappato
 - I diritti assegnati all'utente sono quelli definiti dal Sistema Operativo per quell'account
- Esempio:
"/O=Grid/OU=GlobusTest/OU=simpleCA-gridnode2.iit.cnr.it/OU=iit.cnr.it/CN=Paolo" paolo

Autorizzazione: Custom

- Globus permette di adottare un sistema di autorizzazione personalizzato
 - Interno
 - Classe java che implementa l'interfaccia PDP di Globus
 - Esterno
 - Servizio Globus
 - Sulla risorsa locale
 - Su una risorsa remota
 - Meccanismo di Callout per invocare il sistema esterno
 - SAML

Autorizzazione

