

Grid Middleware and Globus Toolkit Architecture

Lisa Childers

Argonne National Laboratory
University of Chicago



Overview

- Grid Middleware
 - ◆ The problem: supporting Virtual Organizations
 - ◆ Requirements
 - ◆ Capabilities
- The Globus Approach

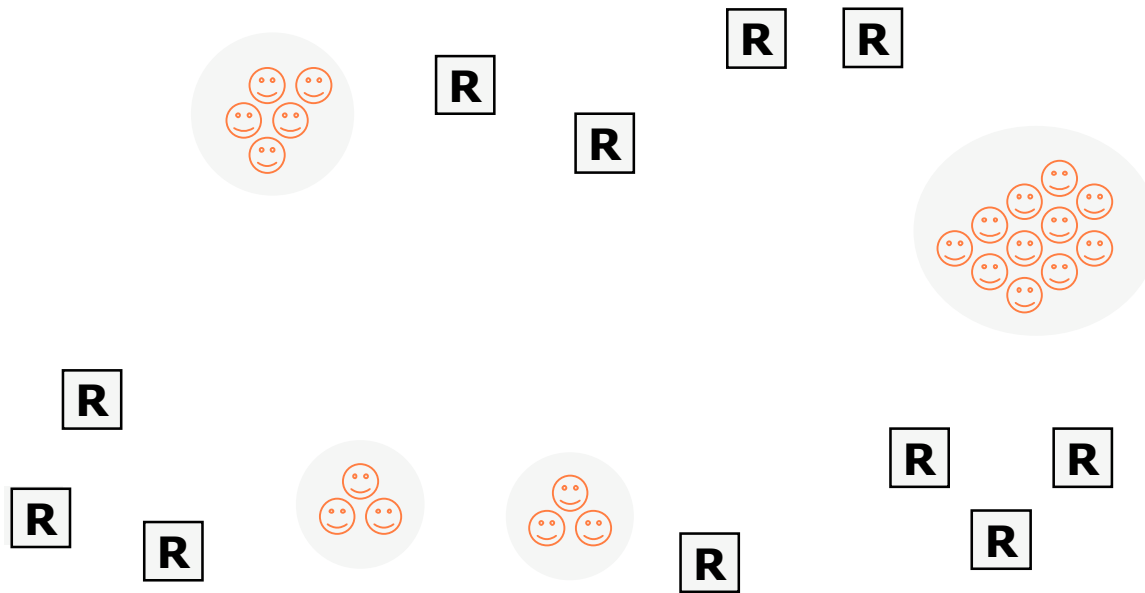
A fundamental problem of Grid Computing: *supporting virtual organizations*

Virtual Organizations



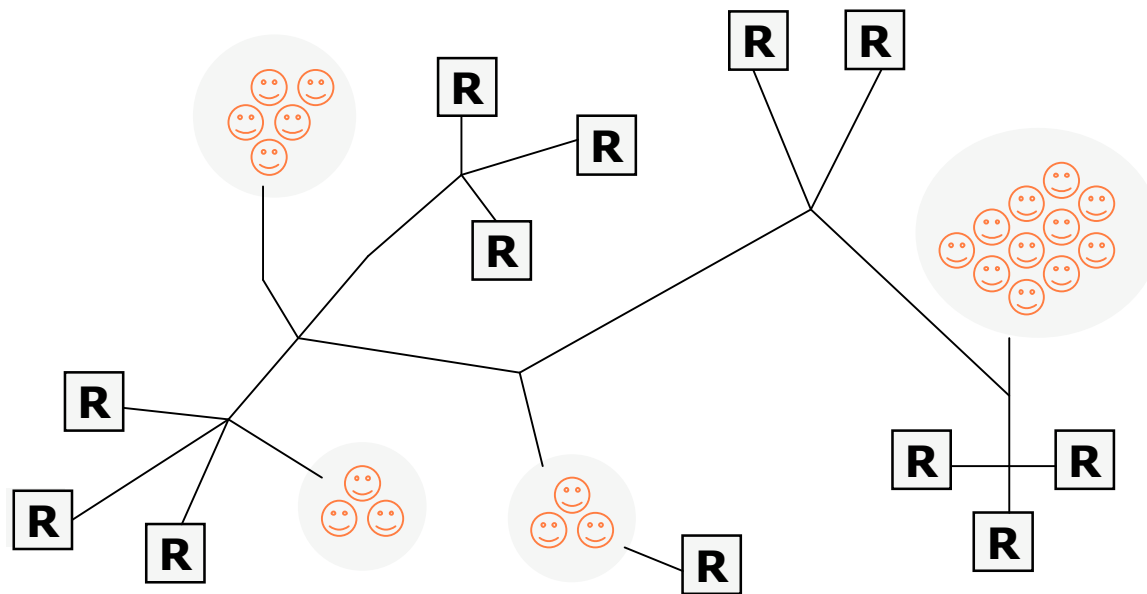
Virtual Organizations

- Distributed resources and people



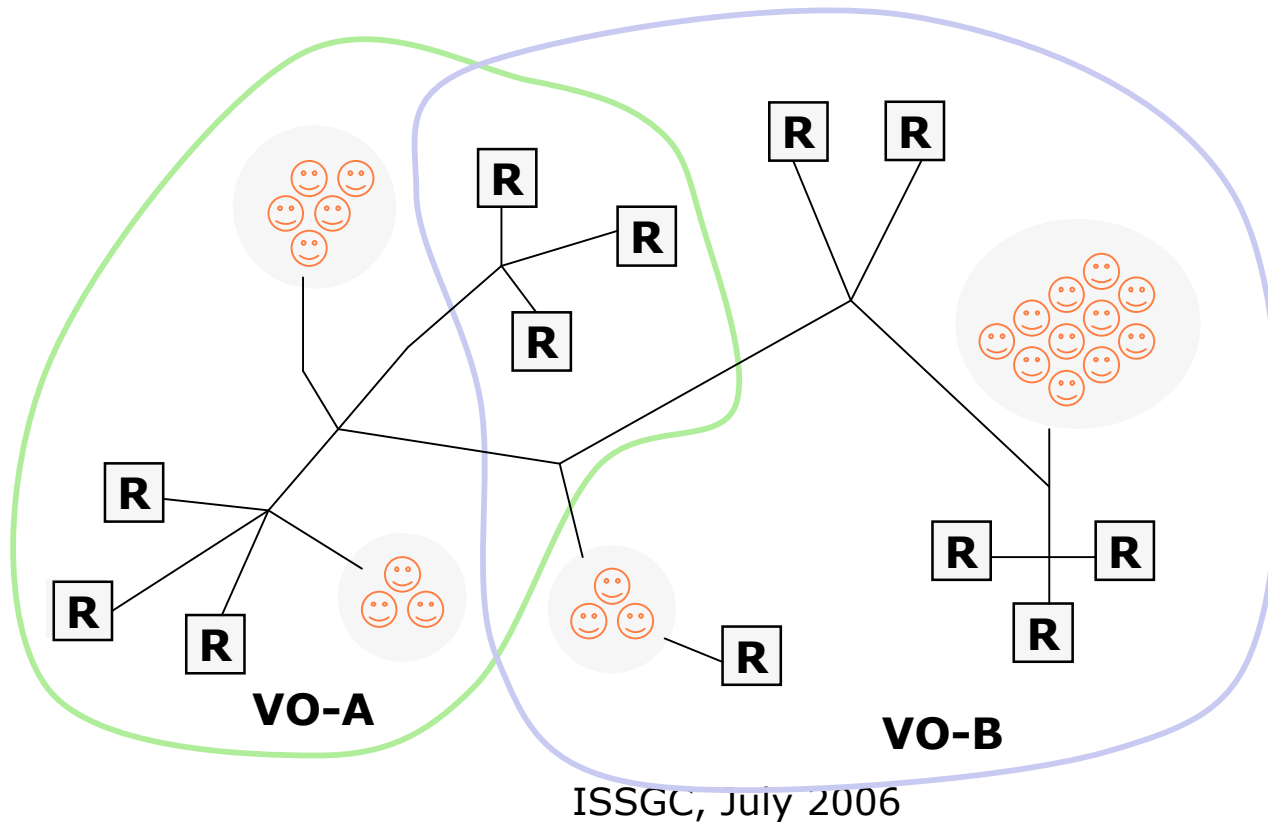
Virtual Organizations

- Distributed resources and people
- Linked by networks, crossing admin domains



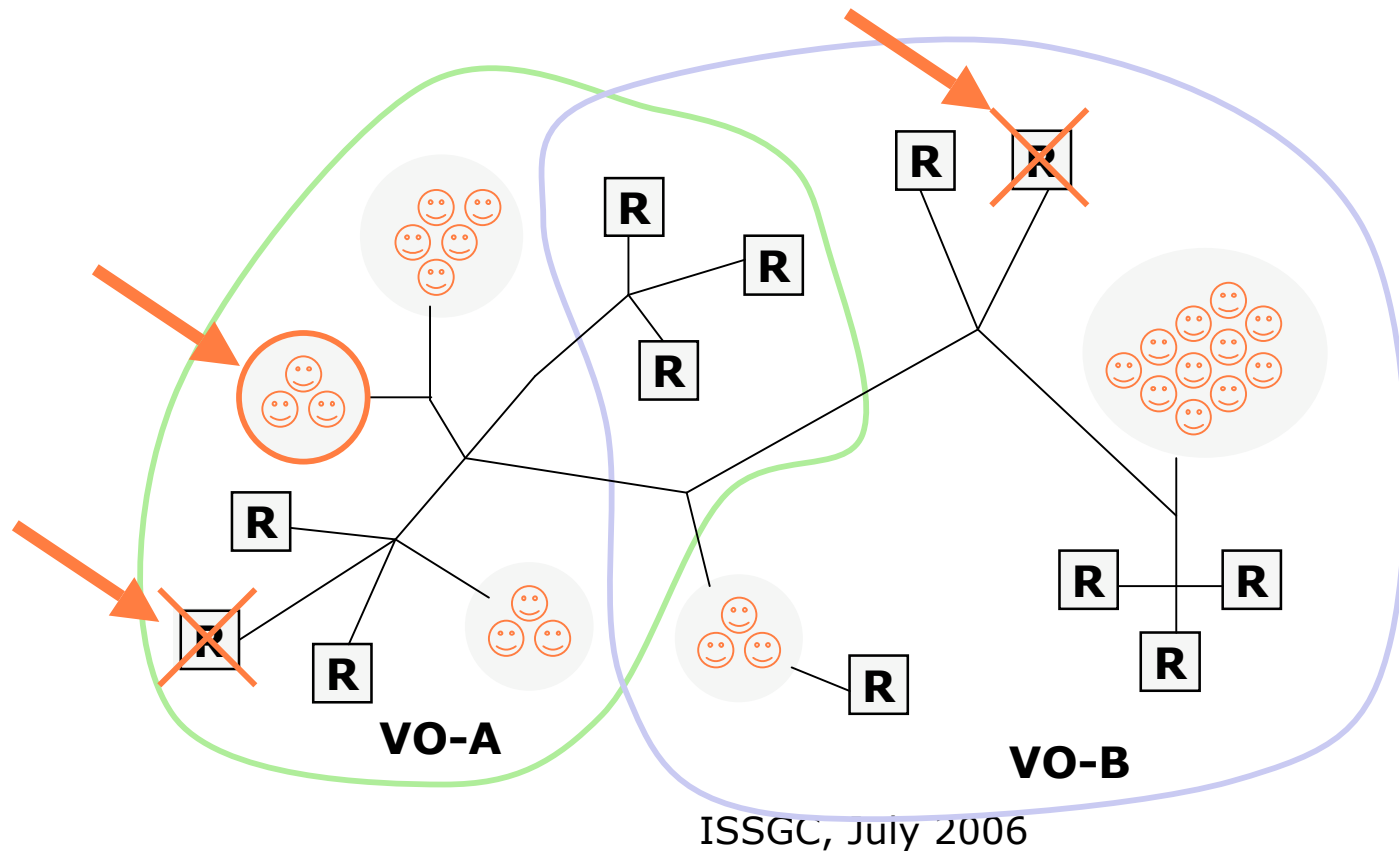
Virtual Organizations

- Distributed resources and people
- Linked by networks, crossing admin domains
- Sharing resources, common goals



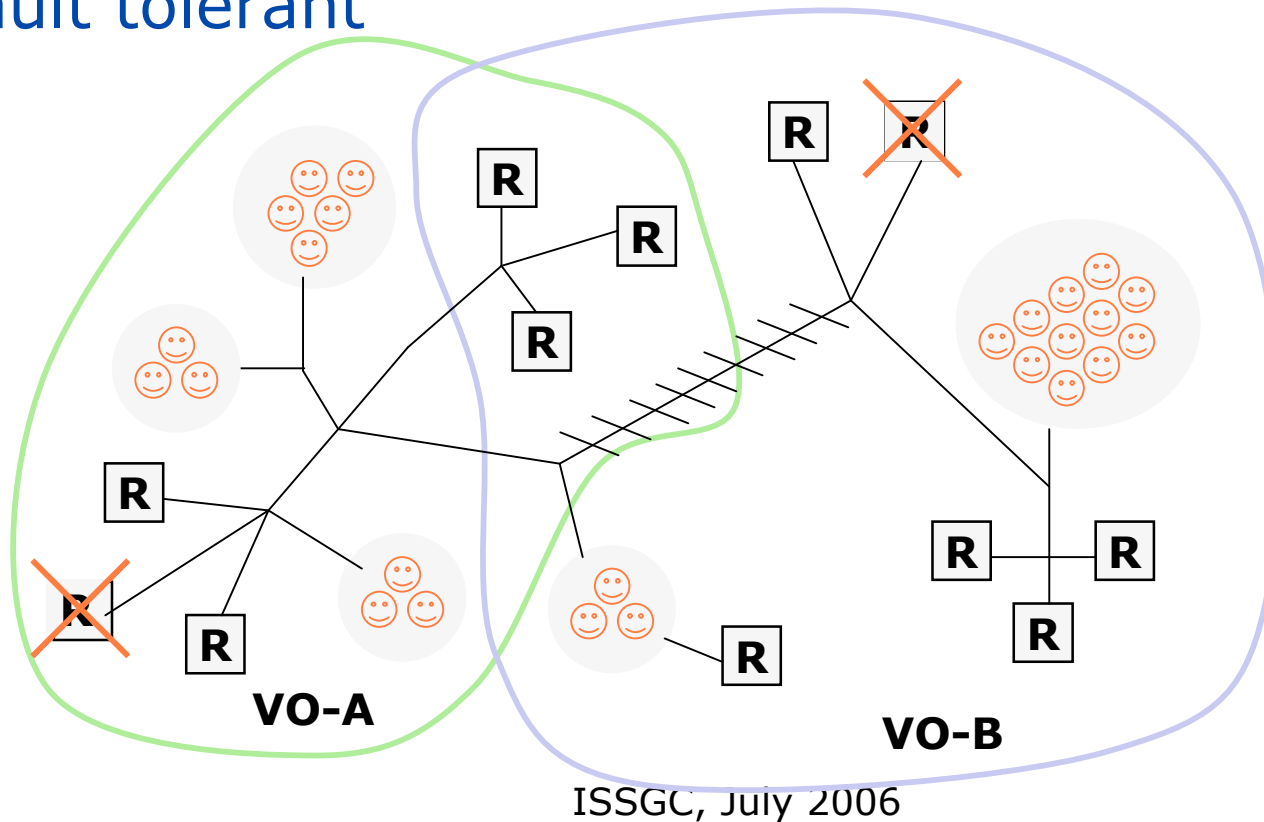
Virtual Organizations

- Distributed resources and people
- Linked by networks, crossing admin domains
- Sharing resources, common goals
- Dynamic



Virtual Organizations

- Distributed resources and people
- Linked by networks, crossing admin domains
- Sharing resources, common goals
- Dynamic
- Fault tolerant



movie time!
(a short tour of the accelerator at CERN)



**10,000
Encyclopedia Britannica's
per second**

Overview

- Grid Middleware
 - ◆ The problem: supporting Virtual Organizations
 - ◆ Requirements
 - ◆ Capabilities
- The Globus Approach

Support for Heterogeneous Environments

- Resource virtualization
- Common management capabilities
- Attribute-based resource discovery
- Standard protocols and schemas

Cross-Organizational Resource Sharing

- Global namespace
- Metadata services
- Site autonomy
- Resource usage data

Optimization of Resource Usage

- Dynamic resource allocation
(supply-side management)
- Dynamic workload prioritization
(demand-side management)

Quality of Service (QoS) Assurance

- Service-level agreement and attainment
- Migration
- ...

Administration

- Policy-based management mechanisms
- Problem-determination mechanisms
- Scalable management architecture

High Availability

- Disaster recovery mechanisms
- Mechanisms for fault management

Job Execution

- Support for a variety of remote job types
- Remote job management
- Job scheduling
- Resource provisioning

Data Services

- Abstractions that provide uniform access and integration to various types of data
- Mechanisms to keep data consistent across replicas and caches
- Mechanisms to persist data
- Mechanisms for data location management

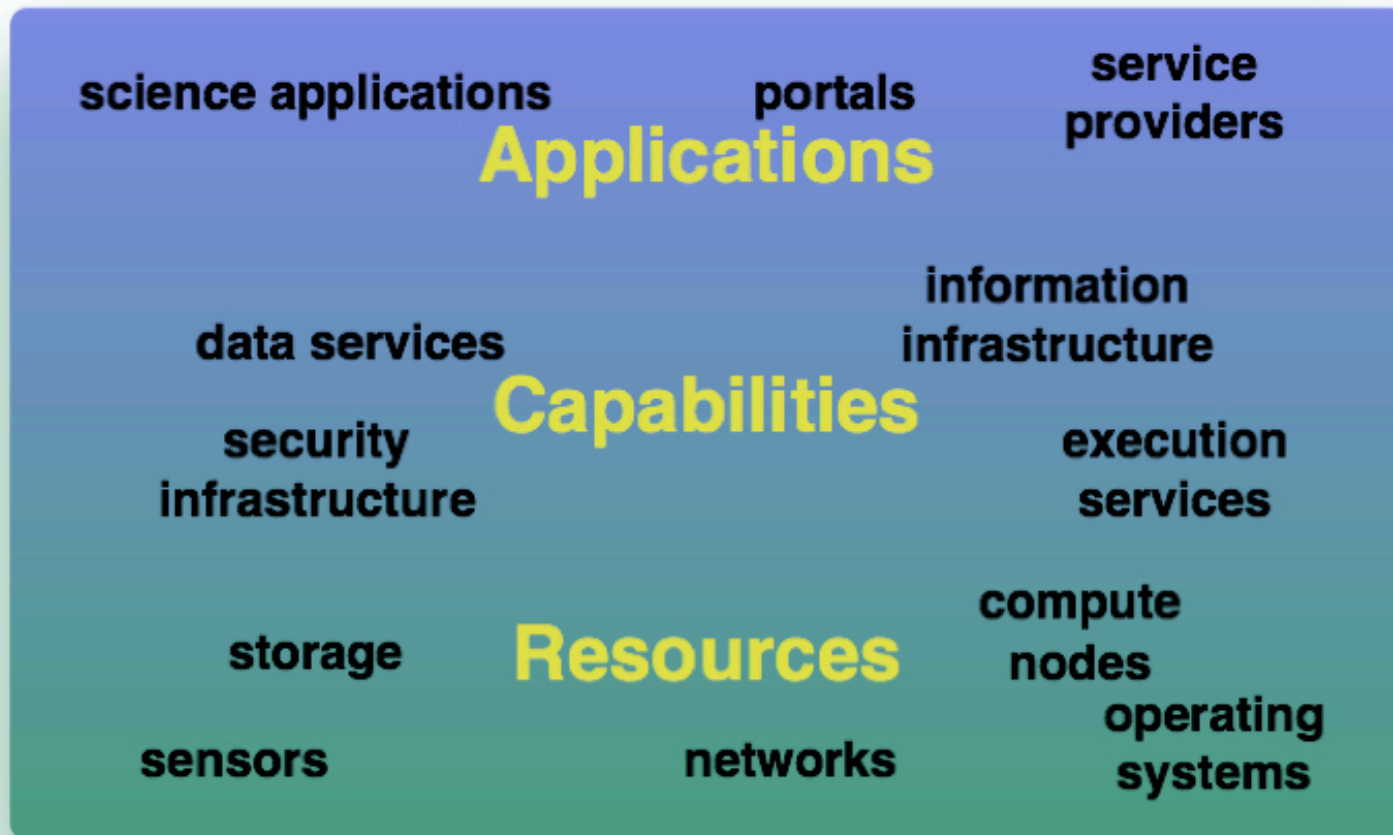
Security

- Authentication and authorization mechanisms
- Support for multiple security infrastructures
- Perimeter security mechanisms that support local infrastructure
- Isolation
- Delegation of access rights
- Support for dynamic negotiation of security policies
- Monitoring in support of intrusion detection and secure logging

Overview

- Grid Middleware
 - ◆ The problem: supporting Virtual Organizations
 - ◆ Requirements
 - ◆ Capabilities
- The Globus Approach

Grid Infrastructure: a Conceptual View



Execution Services

To instantiate and manage “units of work”

- Behavior includes
 - ◆ Finding and selecting execution candidate locations
 - ◆ Preparing for execution
 - ◆ Initiating execution
 - ◆ Managing execution

Data Services

To move, access and manage data resources

- Behavior includes

- ◆ Move data
- ◆ Manage replicated copies
- ◆ Run queries and updates
- ◆ Transform data into new formats
- ◆ Maintain metadata

Resource Management Services

- Management of low-level resources
 - ◆ Monitoring, setup and control, discovery
- Management of the capabilities
 - ◆ Functional interface management (e.g. create and destroy jobs)
- Domain-independent management
 - ◆ System management of the many services on the Grid (e.g., system-wide backups)

Security Services

To facilitate the enforcement of security-related policy within Virtual Organizations

- Behavior includes
 - ◆ Verifying proof of an asserted identity
 - ◆ Identity mapping
 - ◆ Policy-based access control decision-making
 - ◆ Audit and secure logging
 - ◆ Privacy

Self-Management Services

To reduce the cost and complexity of owning and operating IT infrastructure

- Behavior includes
 - ◆ Negotiating mechanisms for resource usage

Information Services

To efficiently access information about applications, resources and services

- Behavior includes
 - ◆ Monitor services and resources
 - ◆ Log system activities
 - ◆ Cache and publish metadata

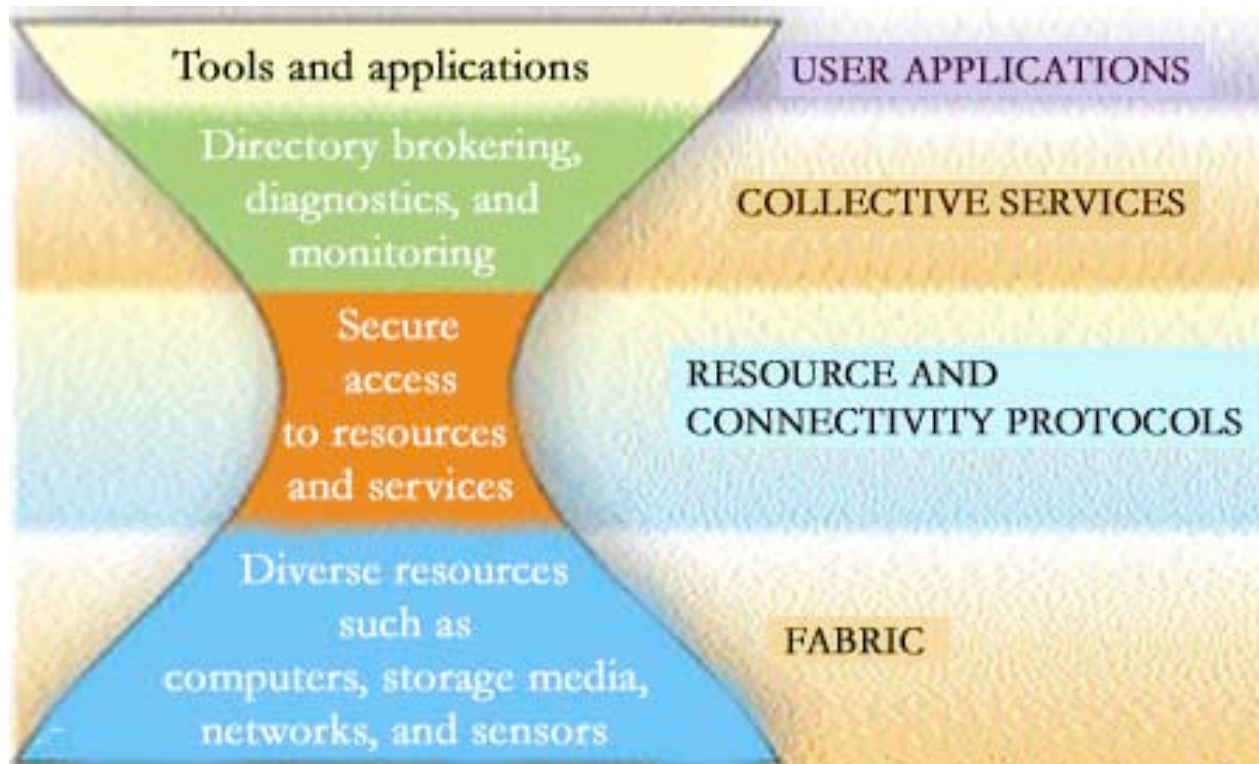
Overview

- Grid Middleware
 - ◆ The problem: supporting Virtual Organizations
 - ◆ Requirements
 - ◆ Capabilities
- The Globus Approach

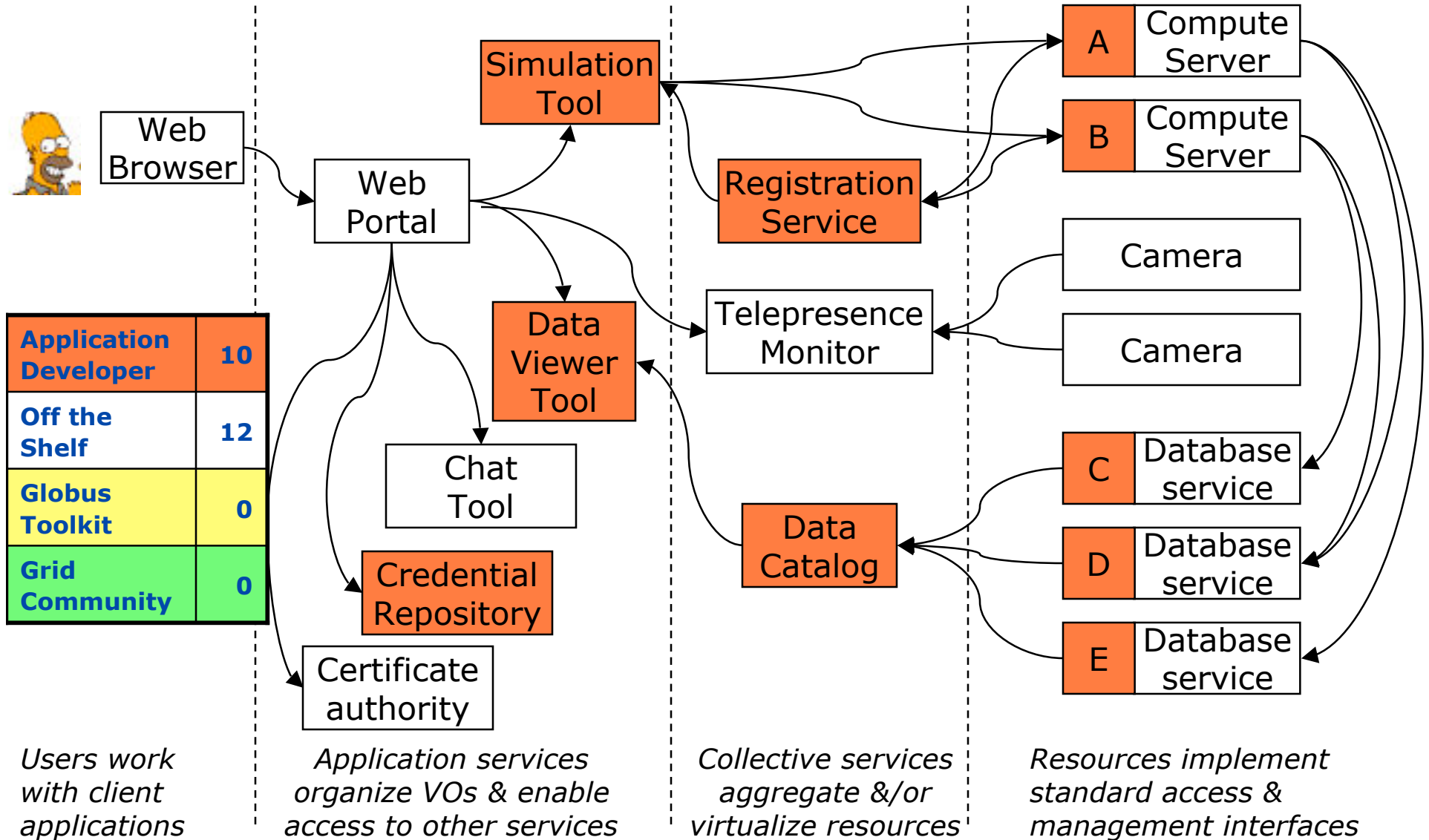
The Role of the Globus Toolkit

- The Globus Toolkit is a collection of solutions to problems commonly found in collaborative distributed applications
- Heterogeneity
 - ◆ A focus, in particular, on wrapping heterogeneity for application developers
- Abstractions
 - ◆ Supporting general-case patterns and interactions, not specific to a particular application domain
- Standards
 - ◆ We capitalize on and encourage use of existing standards (IETF, W3C, OASIS, GGF)
 - ◆ GT also includes reference implementations of new/proposed standards in these organizations

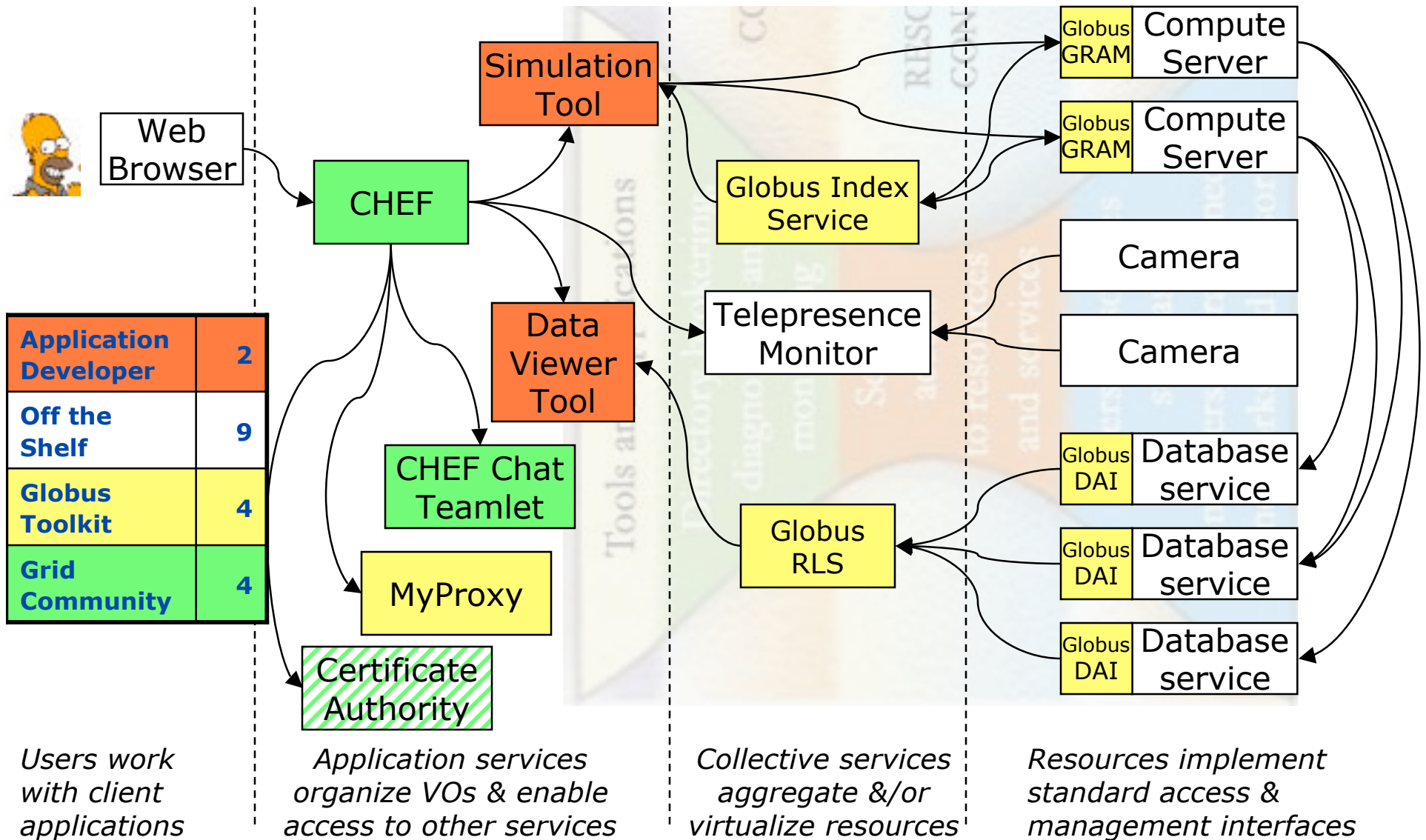
Layers in the Grid



Without the Globus Toolkit



With the Globus Toolkit



The Globus Toolkit: “Standard Plumbing” for the Grid

- Today the majority of the GT public interfaces are usable by application developers and system integrators
 - ◆ Relatively few end-user interfaces
 - ◆ In general, not intended for direct use by end users (scientists, engineers, marketing specialists)
- *Not turnkey solutions, but **building blocks & tools** for application developers & system integrators*
 - ◆ Some components (e.g., file transfer) go farther than others (e.g., remote job submission) toward end-user relevance
- **Better to reuse than reinvent!**
 - ◆ Plenty of interesting unsolved problems to work on
 - ◆ Compatibility with other Grid systems comes for free

Sidebar: The Globus Commitment to Open Source

- To allow for inspection
 - ◆ for consideration in standardization processes
- To encourage adoption
 - ◆ in pursuit of ubiquity and interoperability
- To encourage contributions
 - ◆ harness the expertise of the community

<http://dev.globus.org>

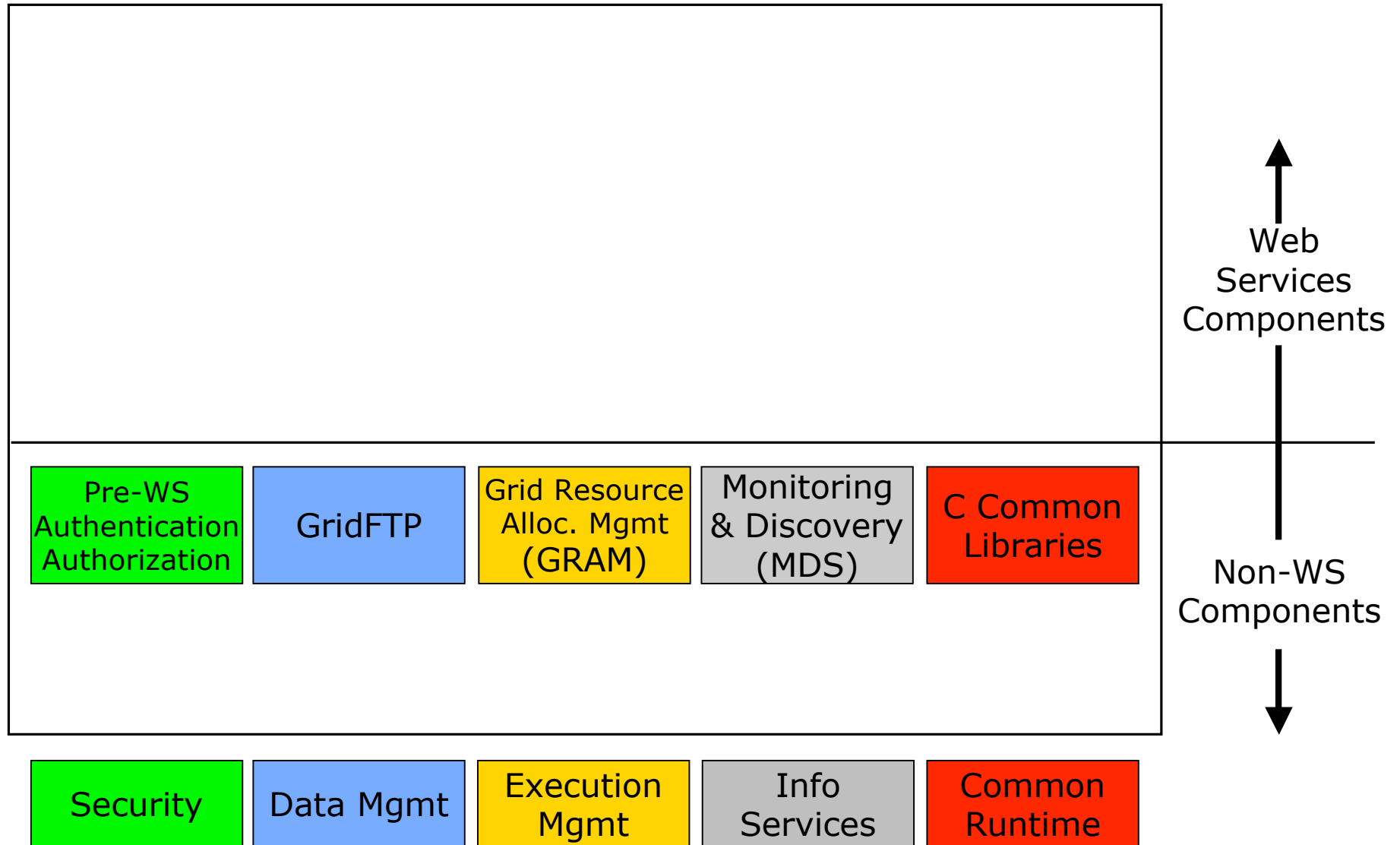
Globus Toolkit by Domain Areas

- Core runtime
 - ◆ Infrastructure for building new services
- Security
 - ◆ Apply uniform policy across distinct systems
- Execution management
 - ◆ Provision, deploy, & manage services
- Data management
 - ◆ Discover, transfer, & access large data
- Information services
 - ◆ Discover & monitor dynamic services

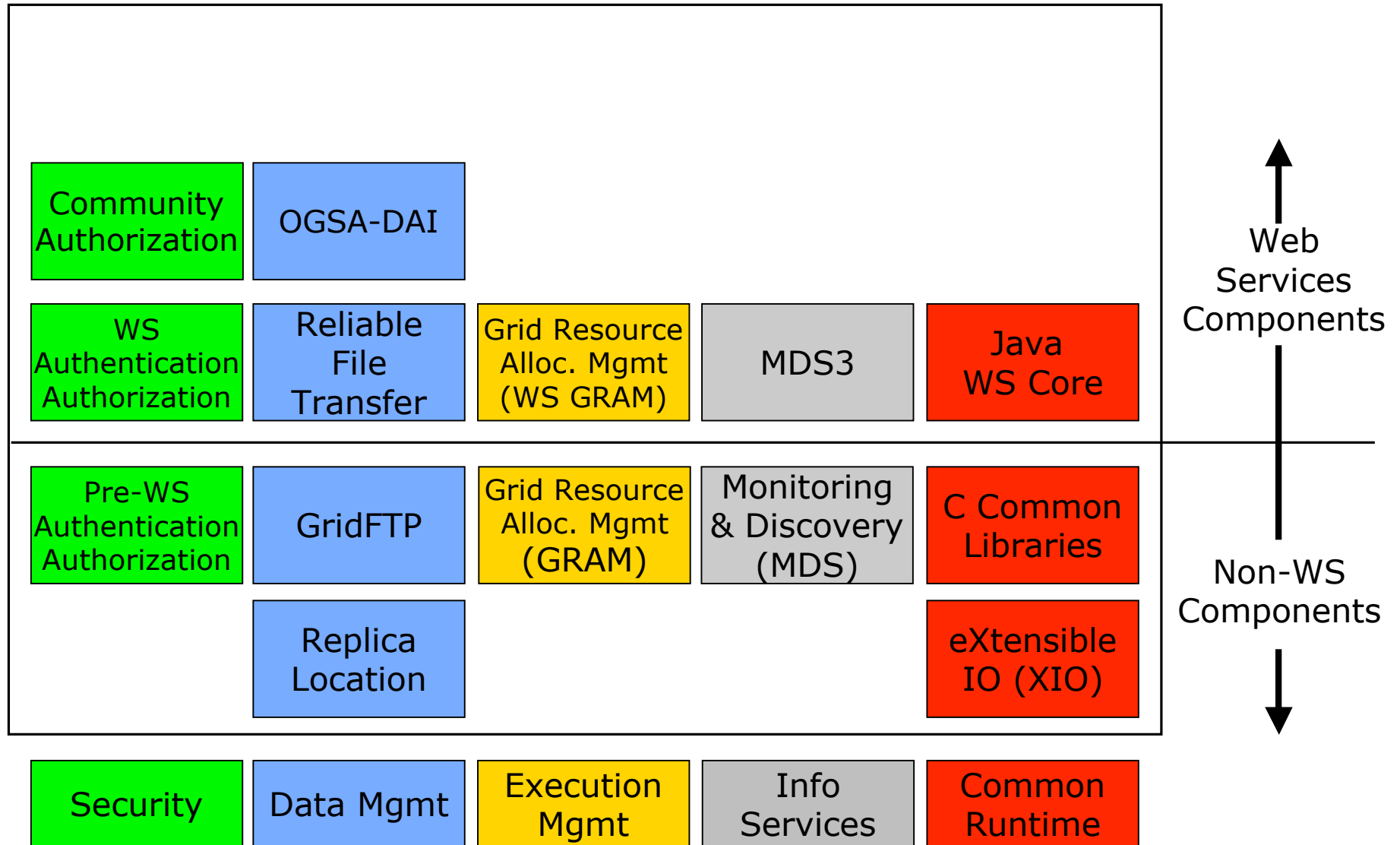
Globus Toolkit by Protocol Type

- Web service protocols
 - ◆ WSDL, SOAP
 - ◆ WS Addressing, WSRF, WSN
 - ◆ WS Security, SAML, XACML
 - ◆ WS-Interoperability profile
- Non Web service protocols
 - ◆ Standards-based, such as GridFTP
 - ◆ Custom

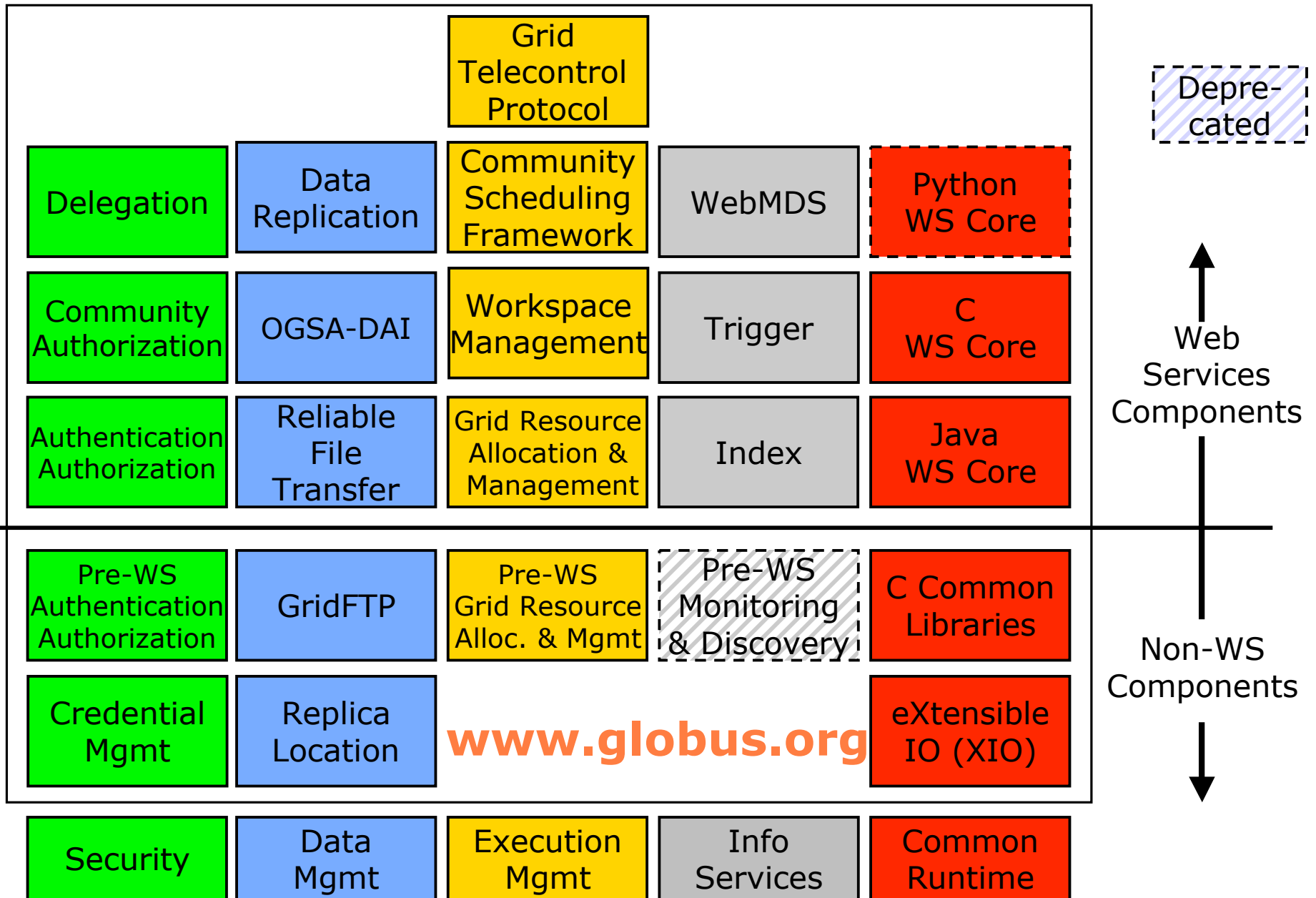
Globus Toolkit version 2 (GT2)



Globus Toolkit version 3 (GT3)



Globus Toolkit version 4 (GT4)



Globus Toolkit: Common Capabilities

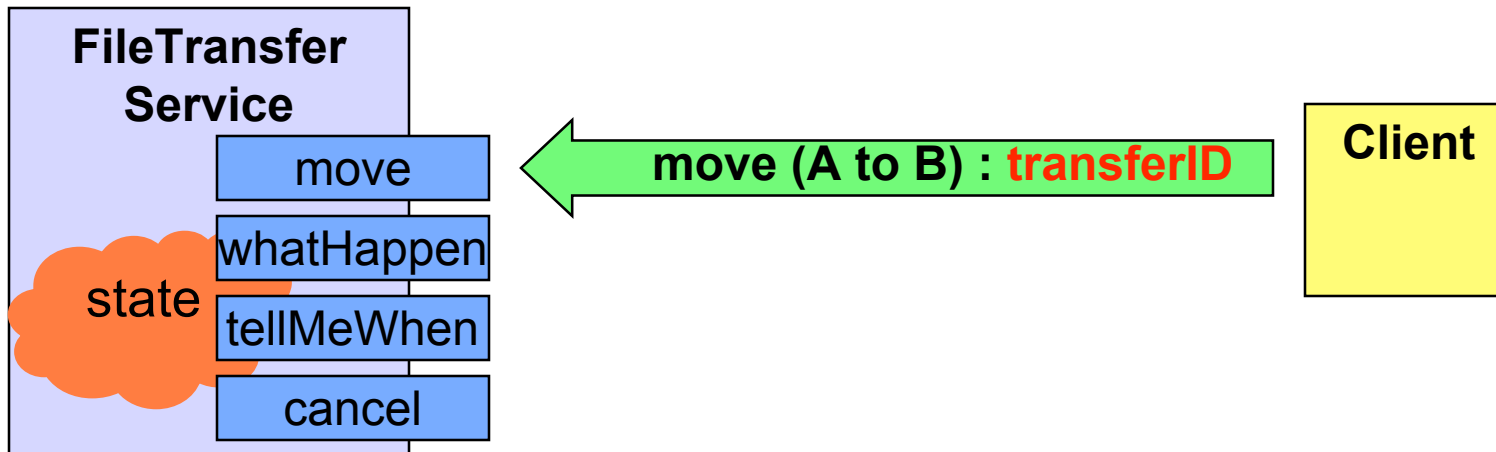
Globus Toolkit v4
www.globus.org

	Data Replication			
Credential Mgmt	Replica Location	Grid Telecontrol Protocol		
Delegation	OGSA-DAI	Community Scheduling Framework	WebMDS	Python Runtime
Community Authorization	Reliable File Transfer	Workspace Management	Trigger	C Runtime
Authentication Authorization	GridFTP	Grid Resource Allocation & Management	Index	Java Runtime
Security	Data Mgmt	Execution Mgmt	Info Services	Common Runtime

GT4 Common Runtime

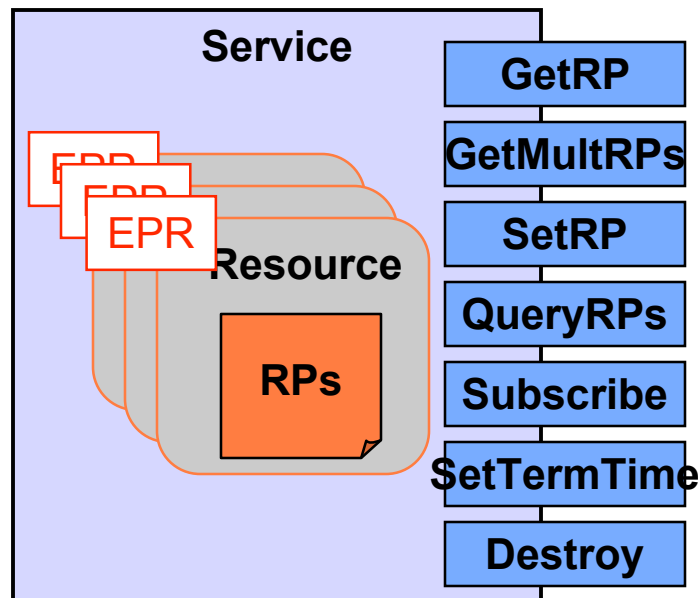
- Providing the common interfaces and capabilities for the toolkit as a whole
- Supports GT services (GRAM, RFT, Delegation, etc.) as well as user-developed services
- Leverages existing WS standards
 - ◆ WS-I Basic Profile: WSDL, SOAP, etc.
 - ◆ WS-Security, WS-Addressing
- Adds support for emerging WS standards
 - ◆ WS-Resource Framework, WS-Notification
- Java, Python, & C hosting environments

FileTransferService (without WSRF)



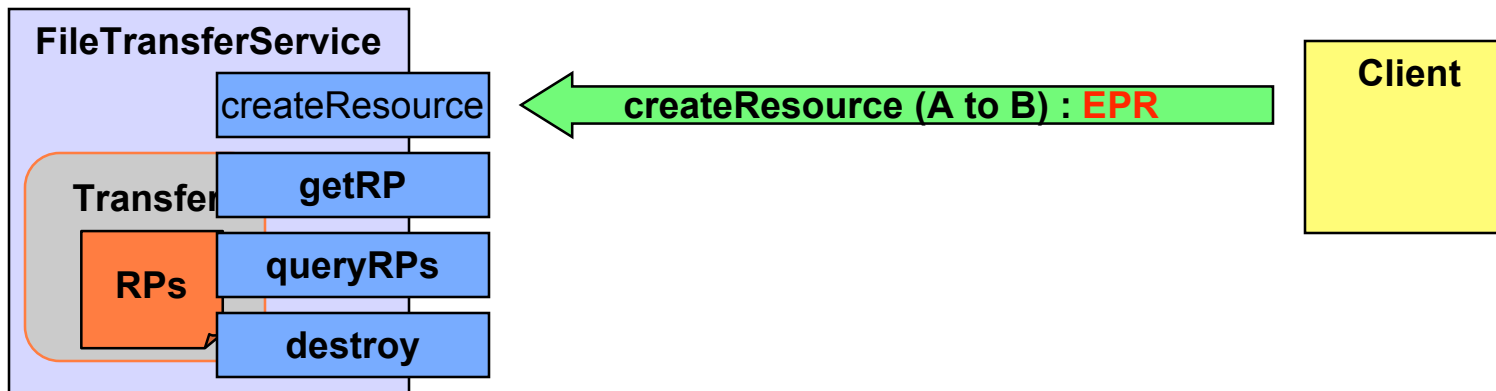
- Developer reinvents wheel for each new service
 - ◆ Custom management and identification of state: **transferID**
 - ◆ Custom operations to inspect state synchronously (**whatHappen**) and asynchronously (**tellMeWhen**)
 - ◆ Custom lifetime operation (**cancel**)

Uniform Interface for Common Interaction Patterns



- State representation
 - ◆ Resource
 - ◆ Resource Property
- State identification
 - ◆ Endpoint Reference
- State Interfaces
 - ◆ GetRP, QueryRPs, GetMultipleRPs, SetRP
- Notification Interfaces
 - ◆ Subscribe
 - ◆ Notify
- Lifetime Interfaces
 - ◆ SetTerminationTime
 - ◆ ImmediateDestruction
- ServiceGroups

FileTransferService (with WSRF)

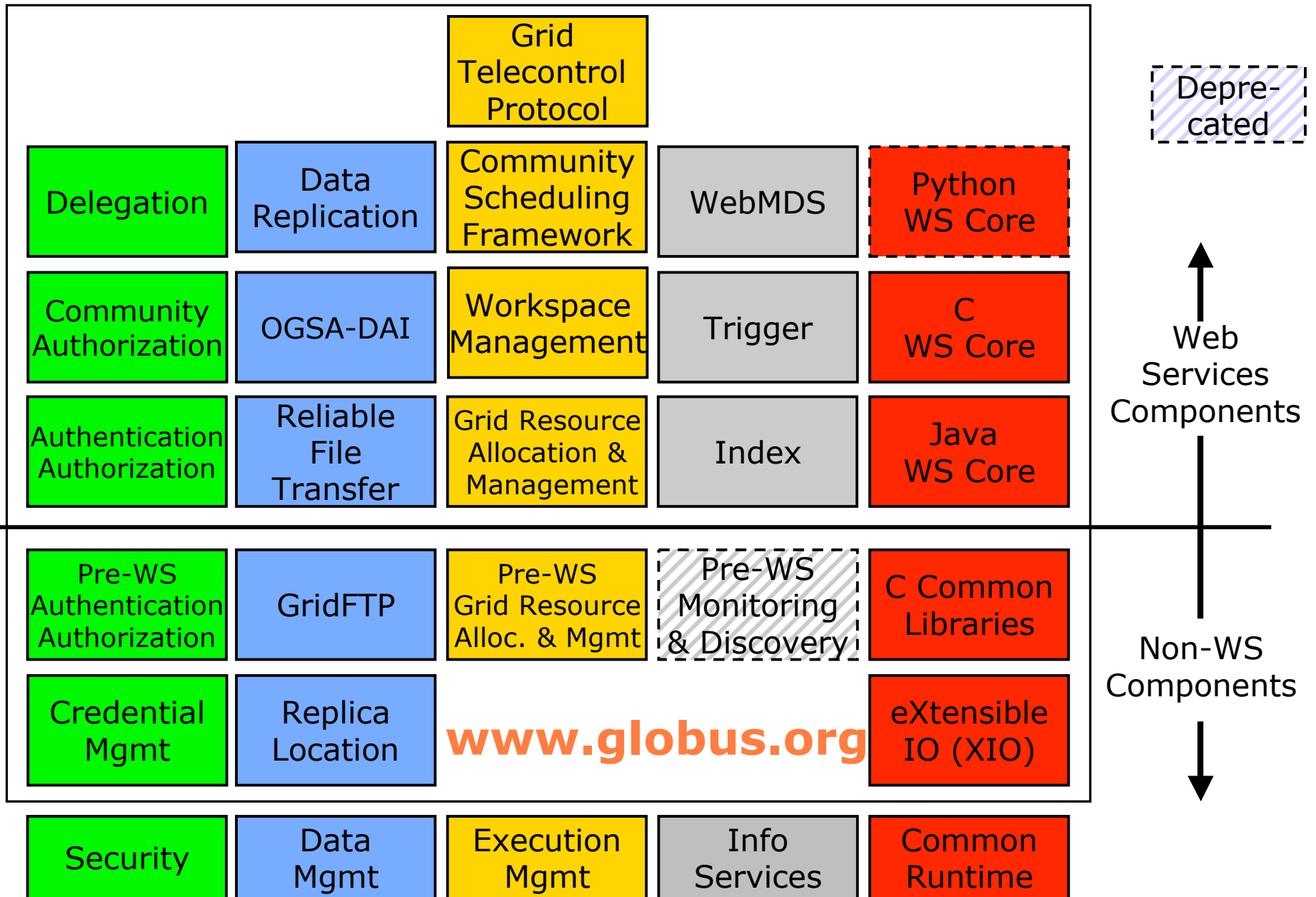


- Developer specifies custom method to createResource and leaves the rest to WSRF standards:
 - ◆ State exposed as Resource + Resource Properties and identified by Endpoint Reference (EPR)
 - ◆ State inspected by standard interfaces (GetRP, QueryRPs)
 - ◆ Lifetime management by standard interfaces (Destroy)

Uniform Interfaces for Common Requirements

- **Naming and bindings** (basis for virtualization)
 - ◆ Every resource can be uniquely referenced, and has one or more associated services for interacting with it
- **Lifecycle** (basis for fault resilient state mgmt)
 - ◆ Resources created by services following factory pattern
 - ◆ Resources destroyed immediately or scheduled
- **Information model** (basis for monitoring, discovery)
 - ◆ Resource properties associated with resources
 - ◆ Operations for querying and setting this info
 - ◆ Asynchronous notification of changes to properties
- **Service groups** (basis for registries, collective svcs)
 - ◆ Group membership rules & membership management
- **Base Fault type**

Globus Toolkit version 4 (GT4)



Wrap-up

- Grid Middleware
 - ◆ The problem: supporting Virtual Organizations
 - ◆ Requirements
 - ◆ Capabilities
- The Globus Approach