the globus alliance
www.globus.org

How to Build a Grid Service Using GT3

Globus Alliance Staff

Charles Bacon, Lisa Childers, Jarek Gawor,
Joe Insley, Ravi Madduri, Argonne National Laboratory
Ben Clifford, USC/Information Sciences Institute

Table of Contents

How to Build a Grid Service Using GT3

- **Overview of Grid Services and GT3**

- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares

- Publish your Grid Service: The GTR

Web Services

- A Web Service can be viewed as a network-accessible function that can be invoked via a well-defined remote interface

- The interface for Web Services is defined using the standard Web Services Description Language (WSDL)

- Web Services enable service-oriented architectures

  – computational tasks are performed using a loosely-coupled collection of services

Grid Service

- A Grid Service is a standard Web Service, plus extensions

- The extensions are defined in a community standard called the Open Grid Services Infrastructure (OGSI)

- This tutorial is designed to highlight these extensions

OGSI Specification

- The OGSI Specification v1 is a draft in the Global Grid Forum
- It defines basic levels of functionality needed for solving complex problems in distributed computing
  - Atomic, composable patterns in the form of interfaces
  - A model for service composition
- The spec defines a language and building blocks for constructing grid services

Open Grid Services Architecture (OGSA)

- OGSA is built on top of OGSI
- The focus of OGSA is on higher-level constructs, such as archetypal grid services
- Current areas of work on OGSA-level standardization include
  - Resource management
  - Security
  - Workflow
  - Data

Globus Toolkit® 3.0 Components

- A full Java implementation of Version 1 of the OGSI Specification
- A framework for developing and hosting OGSI-compliant grid services
- GT3 security infrastructure
- New services built on OGSI
  - Managed Jobs (akin to gatekeeper/jobmanager)
  - Reliable File Transfer (RFT)
  - Index Service (akin to GIIS)
- A new service not yet OGSI-fied
  - Replica Location Service (RLS)
- A complete GT2.4 distribution

GT3 Components used in this Tutorial

- A full Java implementation of Version 1 of the OGSI Specification
- A framework for developing and hosting OGSI-compliant grid services
- GT3 security infrastructure
- New services built on OGSI
  - Managed Jobs (akin to GT2 GRAM)
  - Reliable File Transfer (RFT)
  - Index Service (akin to GT2 GIIS)
- A new service not yet OGSI-fied
  - Replica Location Service (RLS)
- A complete GT2.4 distribution

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - **Overview**
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
- Publish your Grid Service: The GTR

Hands-On Tutorial Structure

- The hands-on portion of the tutorial is organized as a series of exercises in which students add increasing functionality to a skeletal service implementation

- The exercises demonstrate fundamental interactions using Open Grid Services Infrastructure

- Each exercise includes:
  - A discussion of the concepts behind the exercise
  - Implementation details
  - Step-by-step instructions
  - A view of the finished exercise

Supporting Tutorial Materials

- Each attendee will use
  - This slideset
  - A code bundle, including
    - > A FileShare service implementation with annotations for each exercise
  - A set of exercise notes
  - X.509 certificates
  - Files for sharing
  - The oracle (Ben)
- The instructors will use
  - An index service containing attendee service entries
    - > The index service used in this tutorial will be included in GT 3.2
  - A visualizer for the index service's data

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - **1. Deployment: Stand Up a FileShare Service**
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
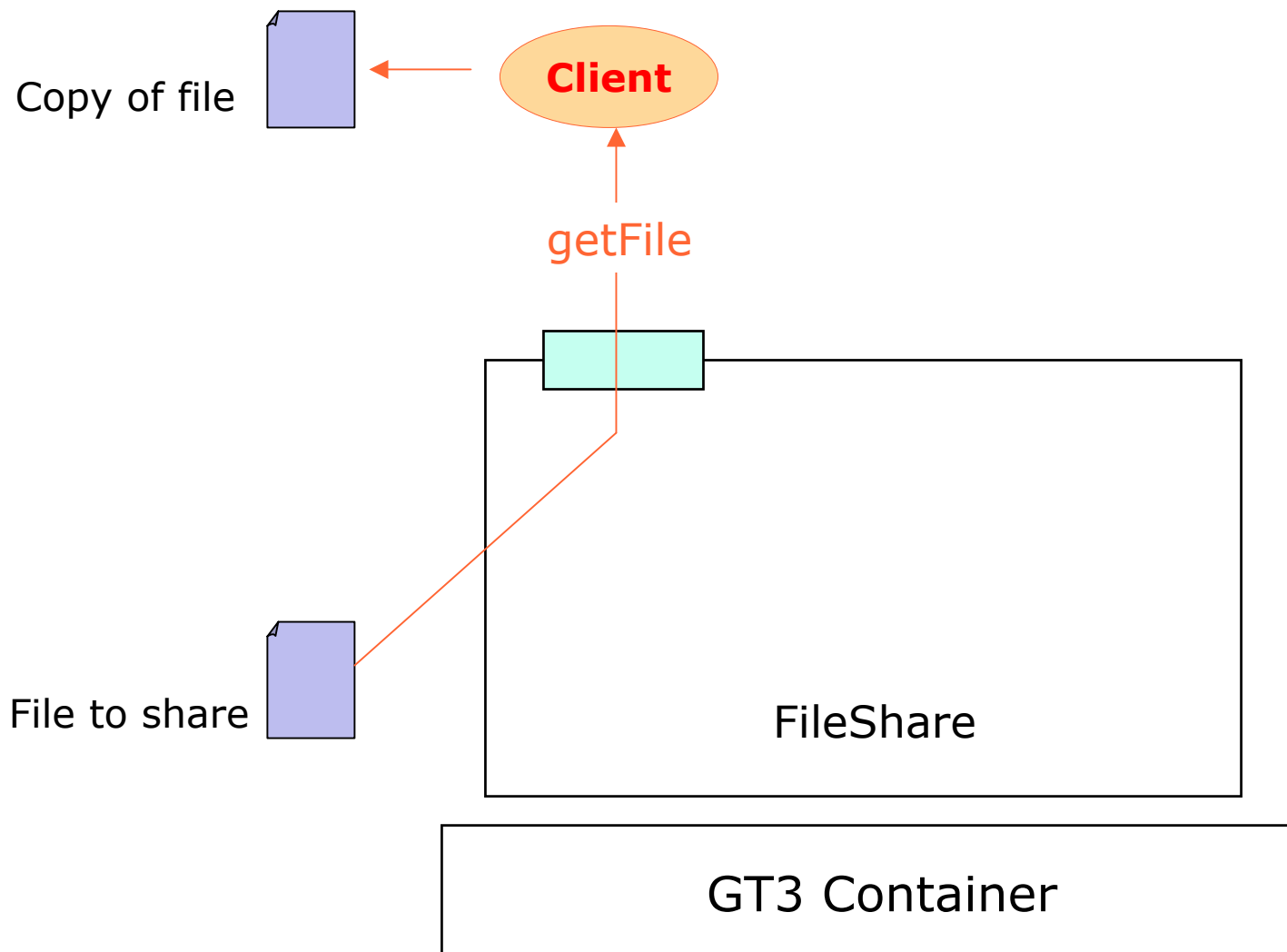- Publish your Grid Service: The GTR

Exercise 1: Deployment

**1. *Deployment:*
Stand up a
FileShare
service on
your laptop**

FileShare

Build a Grid Service Using GT3

The FileShare Service

Copy of file

Client

getFile

File to share

FileShare

GT3 Container

Fileshare Service Overview

- Interface Description
  - .gwsdl files

- Service and Client Implementation
  - .java files

- Build Instructions for Ant
  - build.xml (like a Makefile)

- Deployment Description
  - .wsdd files

Pieces to Install

- ## GT3 core
  - Includes the container
    - > globus-start-container – starts container

- ## Fileshare service
  - Includes the service and client
  - Automated build/deploy using ant
    - > cleanAll – similar to "make clean"
    - > deployGar – compiles .java, installs into container
    - > undeployGar – uninstalls from container

What Attendees Should Do

- Install GT3 core, fileshare service

- Start the GT3 container
  - Will start FileShare service automatically

- Run the GetFile client
  - Get a file (for example, LICENSE)

What Attendees Should See

- When GT3 container is started, a list of deployed services

- bin/globus-start-container

  org.globus.ogsa.server.ServiceContainer [run:569] INFO: Starting SOAP server at: http://140.221.11.99:8080/ogsa/services/

  With the following persistent services:

  http://140.221.11.99:1888/ogsa/services/core/admin/AdminService

  [and many more]

What Attendees Should See

- After client is run, the file you requested is copied into your current directory

java org.globus.ogsa.impl.samples.fileshare.client.getFile LICENSE

- Receive file 'LICENSE' of size 6615

Exercise 1 Review

- $GLOBUS_LOCATION
  - Place to run globus-start-container
- $TUTORIAL_LOCATION
  - Contains the WSDD file and ant build.xml
  - src/org/globus/ogsa/impl/samples/fileshare
  - schema/samples/fileshare/
  - src/org/globus/ogsa/impl/samples/fileshare/client
- $CLIENT_LOCATION
  - Location for receiving files

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - **2. Naming: Share Files using Identifiers**
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
- Publish your Grid Service: The GTR

Exercise 2: Service Naming

**1. *Deployment:*
Stand up a
FileShare
service on
your laptop**

FileShare

**2. *Service Naming:*
Share files using identifiers**

Client
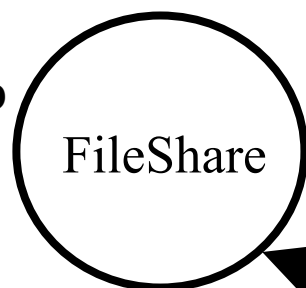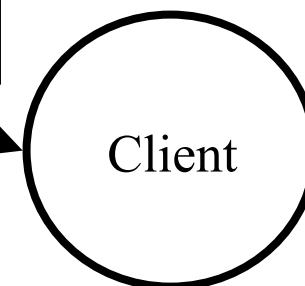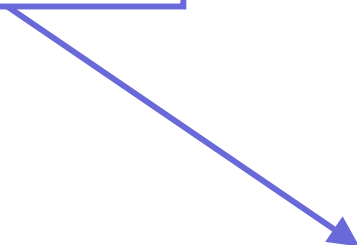
Grid Service Naming

- Grid Services can be stateful.   One implication of this is that one instance is potentially quite different from another
  - Grid Services must be uniquely identifiable
- Grid Service handles are the way in which services (and their associated state) are uniquely identified
- In one sense, a grid service handle can be thought of as representing a particular combination of behavior + state

Naming: Grid Service Handles

A GSH is a network-wide name for a service
- A GSH refers to one specific service

Grid Service Handle

FileShare

Naming: Grid Service Handles

Grid Service Handle

Grid Service Handle

A GSH is a network-wide name for a service

- A GSH refers to one specific service
- More than one GSH can refer to the same service

FileShare

What Attendees Should Do

- Modify the GetFile client to use a GSH

- Look at the container output for the GSH of the FileShareService

- Use the local service's GSH explicitly with the client

- Exchange GSHs with a neighbor and retrieve from their service

What Attendees Should See

- java org.globus.ogsa.impl.samples.fileshare.client.GetFile uniq1 GSH1

  – Received uniq1 from yourself

- java org.globus.ogsa.impl.samples.fileshare.client.GetFile uniq2 GSH2

  – Received uniq2 from your neighbor

Exercise 2 Review

FileShareServiceGridLocator locator =
  new FileShareServiceGridLocator();

FileSharePortType portType =
  locator.getFileSharePort(new
  HandleType(this.handle));

FileType file =
  portType.getFile(this.filename);

Build a Grid Service Using GT3

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
    - Overview
    - 1. Deployment: Stand Up a FileShare Service
    - 2. Naming: Share Files using Identifiers
    - **3. Inspection: Add Service Data**
    - 4. Virtual Organization: Register with a Community Index
    - 5. Lifetime Management: Maintain service registration
    - 6. Discovery: Find a File
    - 7. GT3 Security: Share Files Securely

    Time permitting:
    - 8. Transience: Create and Destroy FileShares
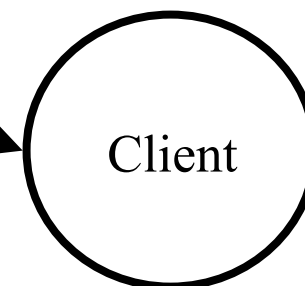- Publish your Grid Service: The GTR

Exercise 3: Inspection

**1. *Deployment:*
Stand up a
FileShare
service on
your laptop**
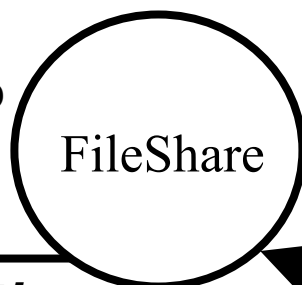
FileShare

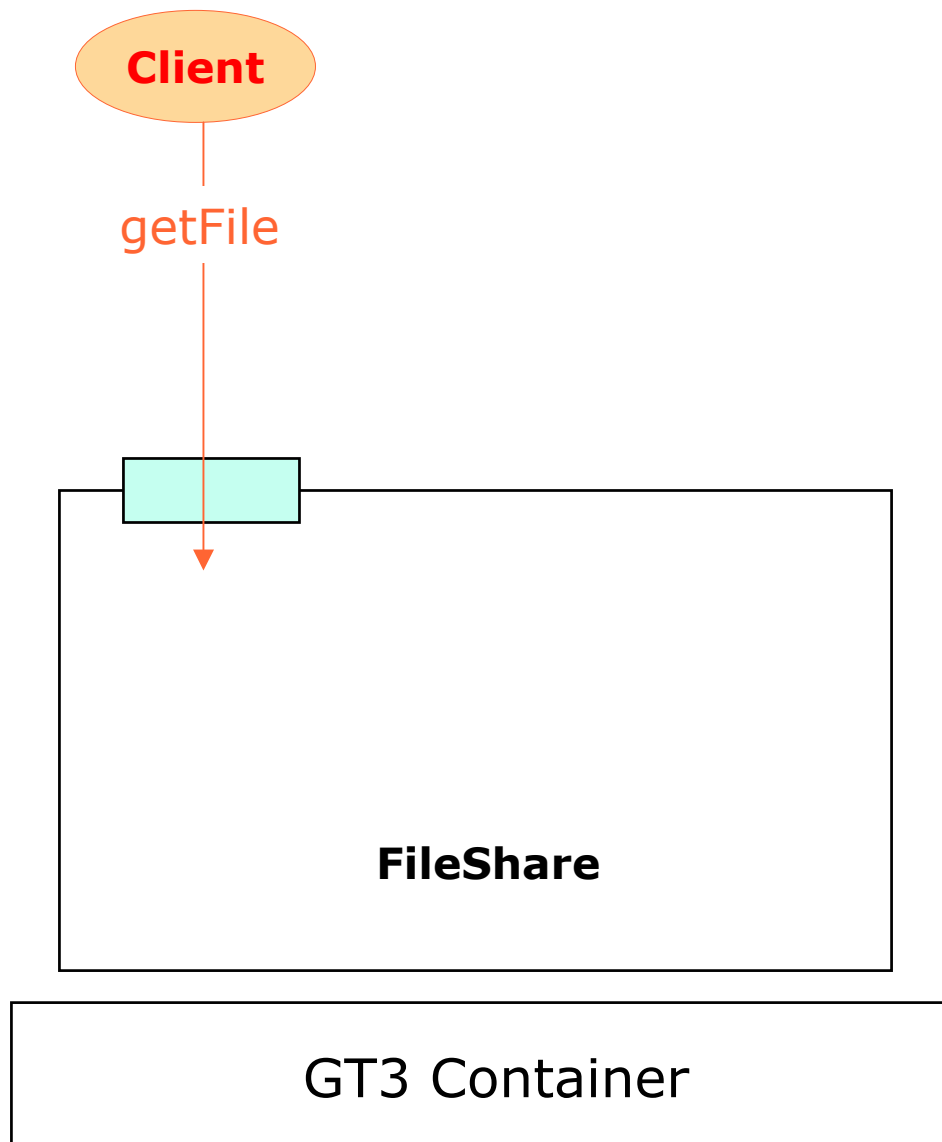**3. *Inspection:*
Add service
data to your
service**

**2. *Service Naming:*
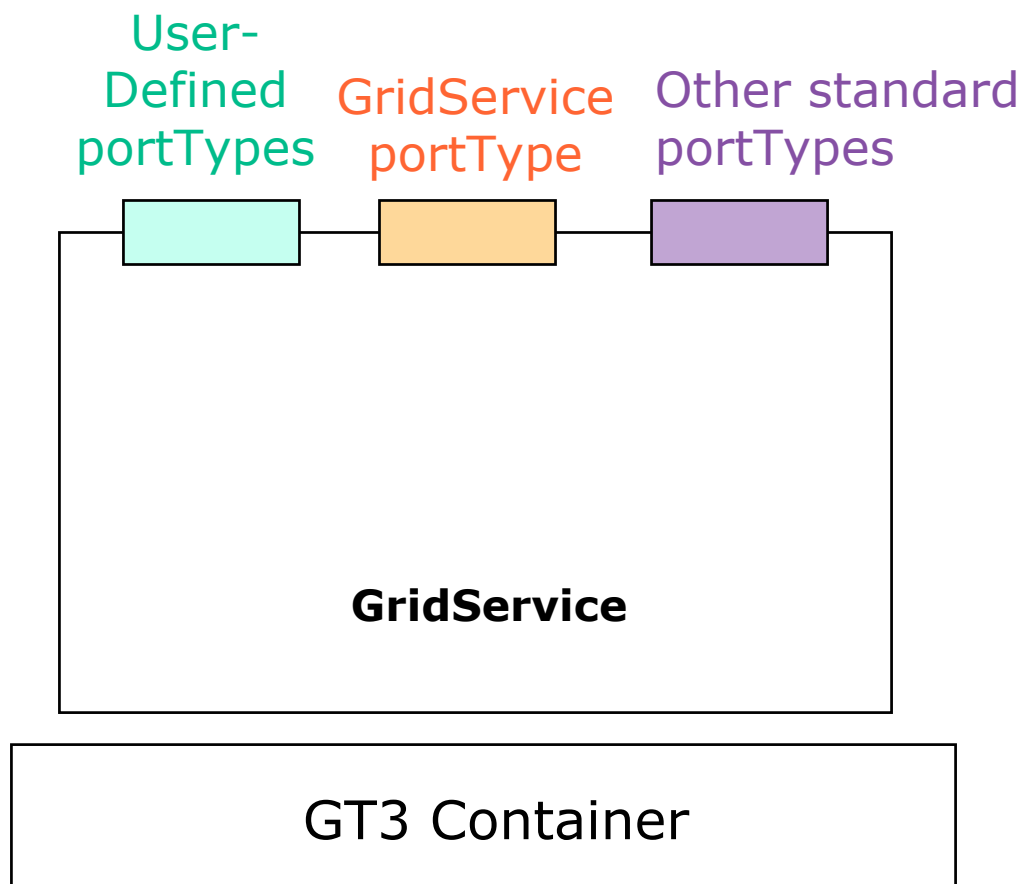Share files using identifiers**

Client

Service Data

- Any Grid Service can expose internal state as Service Data Elements

  – An XML element of arbitrary complexity

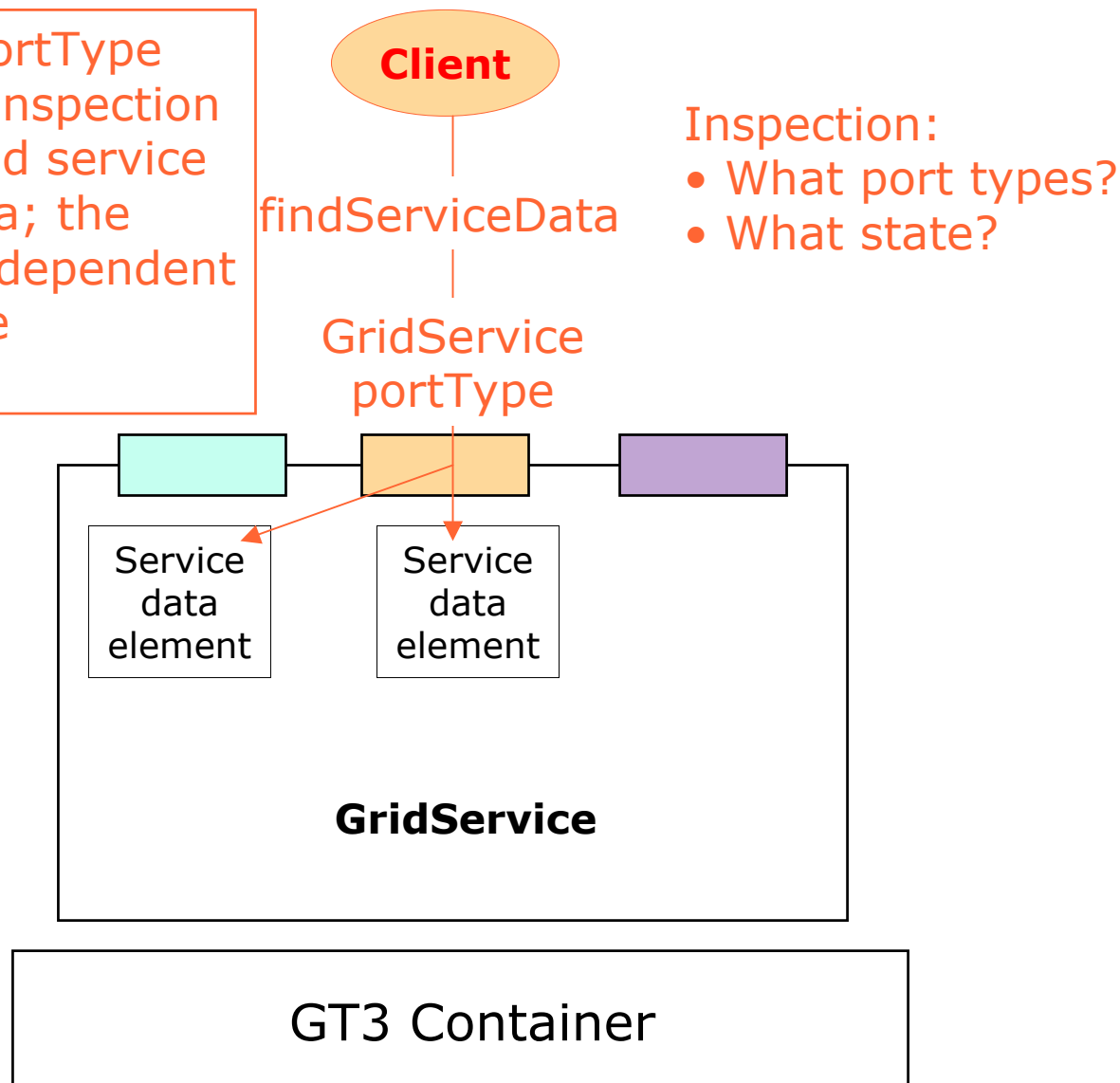- Each service has a set of Service Data Elements

Service Interfaces

Client

getFile

FileShare

GT3 Container

Types of Interfaces

User-Defined portTypes   GridService portType   Other standard portTypes

GridService

GT3 Container

Grid Service portType

The GridService portType
is mandated in OGSI

GridService
portType

Service
data
element

Service
data
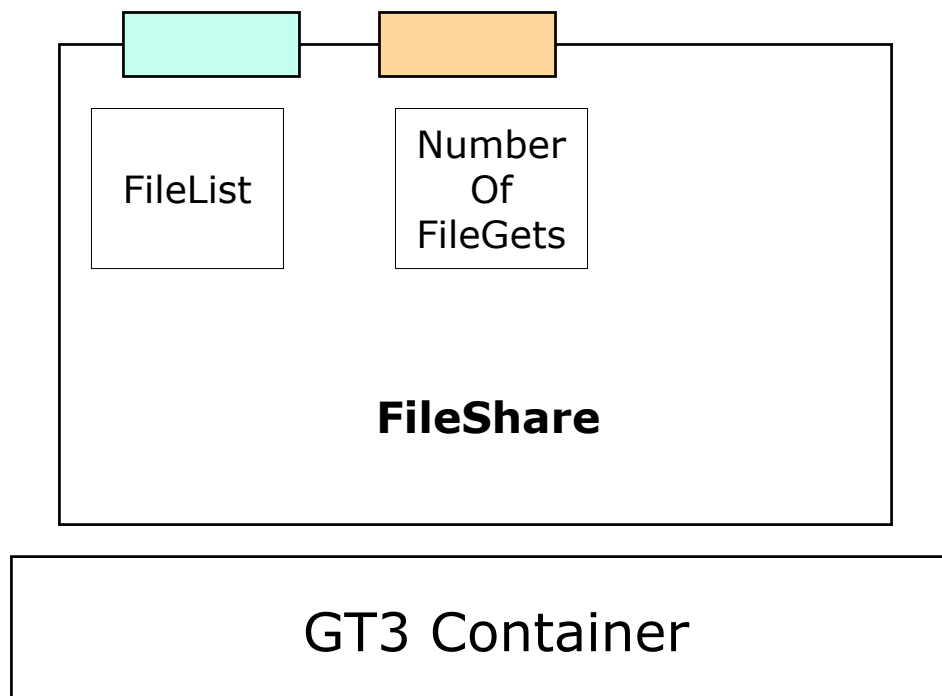element

GridService

GT3 Container
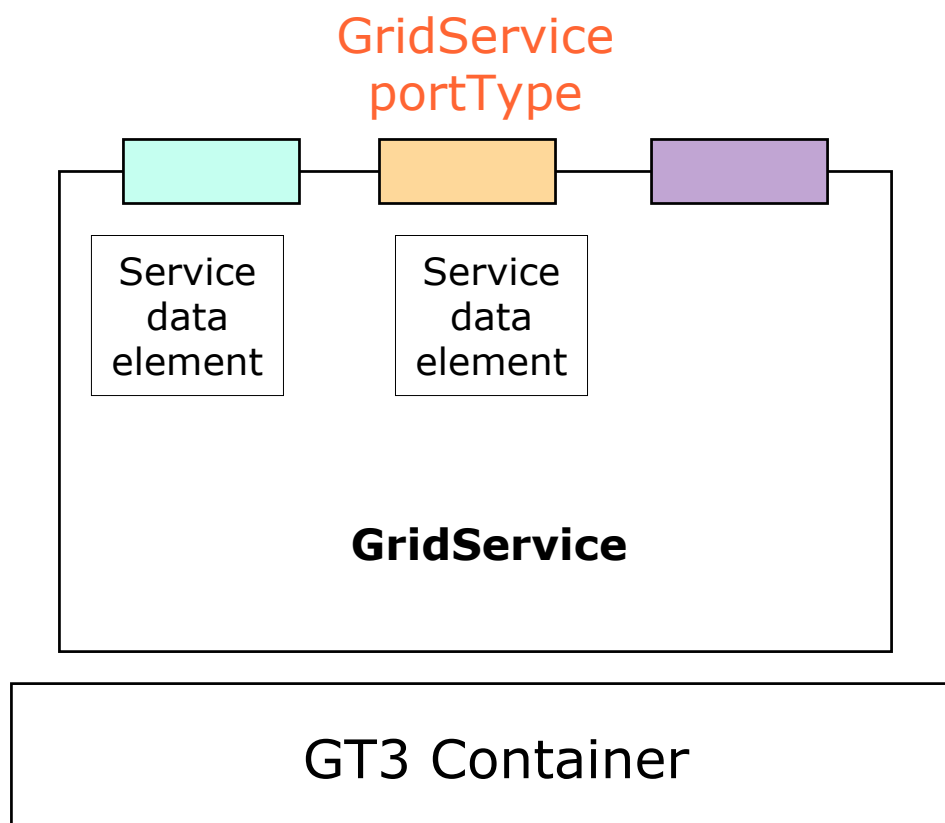
Inspection

The Grid Service portType provides standard inspection mechanisms for grid service operations and data; the mechanisms are independent of a specific service implementation

Client

findServiceData

Inspection:
• What port types?
• What state?

GridService
portType

Service data element

Service data element

GridService

GT3 Container

FileShare Service Data

Build a Grid Service Using GT3

WSDL

- Web Service Description Language
- XML-based language for:
  - Abstractly describing message exchanges between clients and services
    - > Types defined using XML Schema
    - > Message comprising one or more parts of XML Schema types/elements
    - > Operation = input/output or input only messages
    - > Interface = named group of operations
  - Binding the interfaces to concrete protocols
    - > E.g. Soap/http
- It says nothing about what messages are sent

GWSDL

- OGSI requires interface extension/composition

- We worked within W3C WSDL working group to define standard interface extension in WSDL 1.2 that meets OGSI requirements

- But could not wait for WSDL 1.2

- So defined gwsdl:portType that extends WSDL 1.1 portType with:
  - WSDL 1.2 portType extension
  - WSDL 1.2 open content model

- Define GWSDL → WSDL 1.1 & 1.2 mappings

GWSDL (Cont.)

- All operations on our service are defined in our GWSDL

- We provide a definition of our custom portTypes

- We obtain standard Grid Service operations by extending the GridService portType

    - <grid:portType name="FileSharePortType" extends="ogsi:GridService">

SDEs in GWSDL

- In addition to operations, SDEs are defined in GWSDL

```
<gwsdl:serviceData name="SDEName"
                type="xsd:int"
                maxOccurs="m"
                minOccurs="n"
                mutability="mutable">
</gwsdl:serviceData>
```

Namespaces

- All our operations are defined inside of XML namespaces

- `<definitions`
  `xmlns:tns="http://ogsa.globus.org/`
  `samples/2003/09/fileshare">`

- Our SDEs are also defined inside of namespaces. We will need to know those namespaces when we query

- For example:

  http://ogsa.globus.org/samples/2003/09/fileshare:FileList

XML Types

- Several types are predefined
  - xsd:int
- Also can specify more complex datatypes
  - <xsd:complexType name="foo">
- Creating new types is out of scope for this tutorial. The types you need for your SDEs have been defined in the GWSDL

What Attendees should Do

- Uncomment serviceData from fileshare_port_type.gwsdl

- Uncoment SDE update code in FileShareImpl.java

- Verify your work by using handy client: ogsi-find-servicedata-by-name
  - Use the pre-built client to inspect service data by name
  - Observe how service data changes over time

What Attendees Should See

- Output of ogsi-find-service-data-by-name

```
    […]
 <ns3:NumberOfFileGets
    […]
   xsi:type="xsd:int">
 0
 </ns3:NumberOfFileGets>
    […]
```
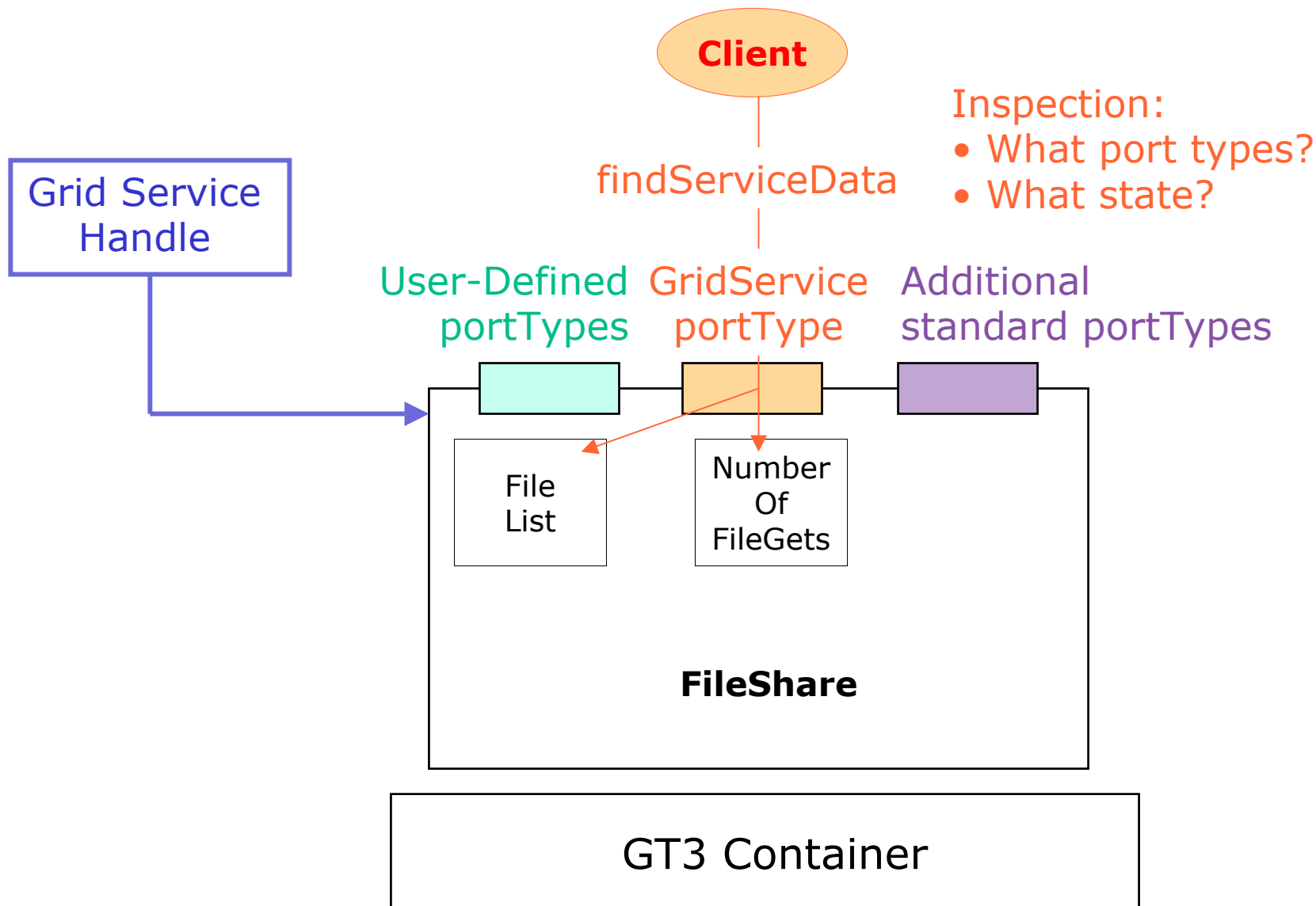
Exercise 3 Review

- SDEs are defined in GWSDL

- You can extend GWSDL to get pre-defined operations, like findServiceData

- SDEs use XML types

- SDEs are namespace qualified
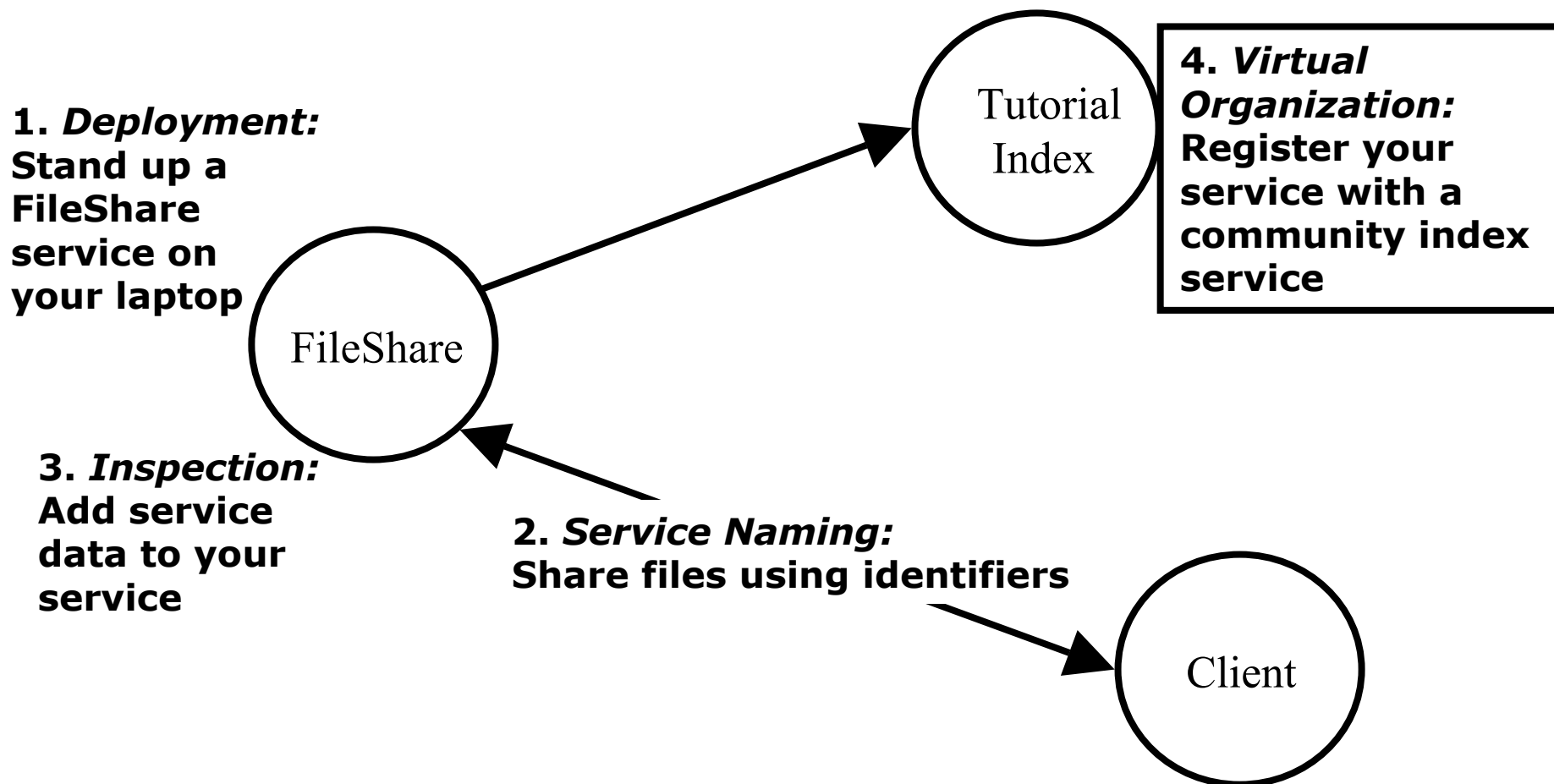
- The GridService portType is mandated by OGSI

What We've Covered So Far

Client

findServiceData

Inspection:
- What port types?
- What state?

Grid Service Handle

User-Defined portTypes

GridService portType

Additional standard portTypes

File List

Number Of FileGets

FileShare

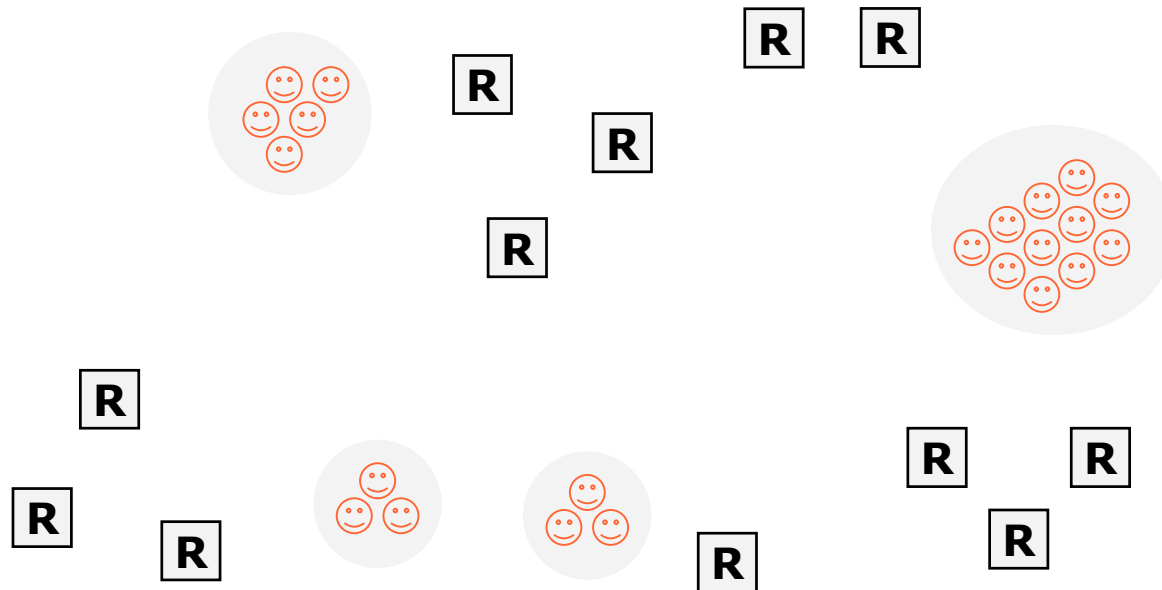GT3 Container

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - **4. Virtual Organization: Register with a Community Index**
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
- Publish your Grid Service: The GTR
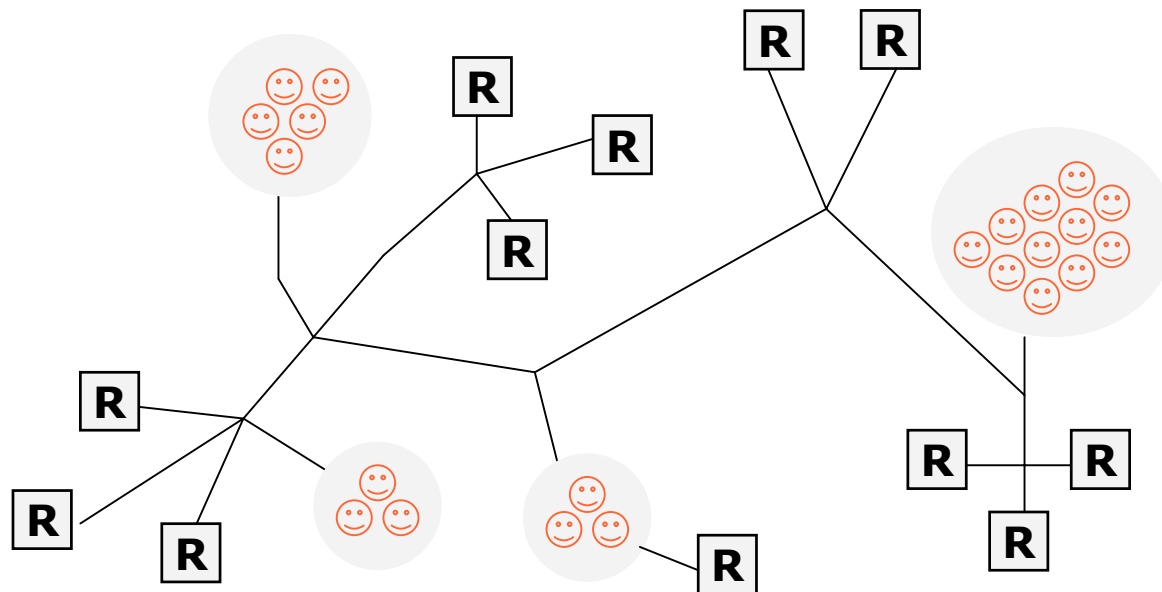
Exercise 4: Virtual Organization

1. *Deployment:*
**Stand up a
FileShare
service on
your laptop**

FileShare

Tutorial
Index

4. *Virtual
Organization:*
**Register your
service with a
community index
service**

3. *Inspection:*
**Add service
data to your
service**

2. *Service Naming:*
Share files using identifiers

Client

Virtual Organizations

- Distributed resources and people

Build a Grid Service Using GT3

Virtual Organizations

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

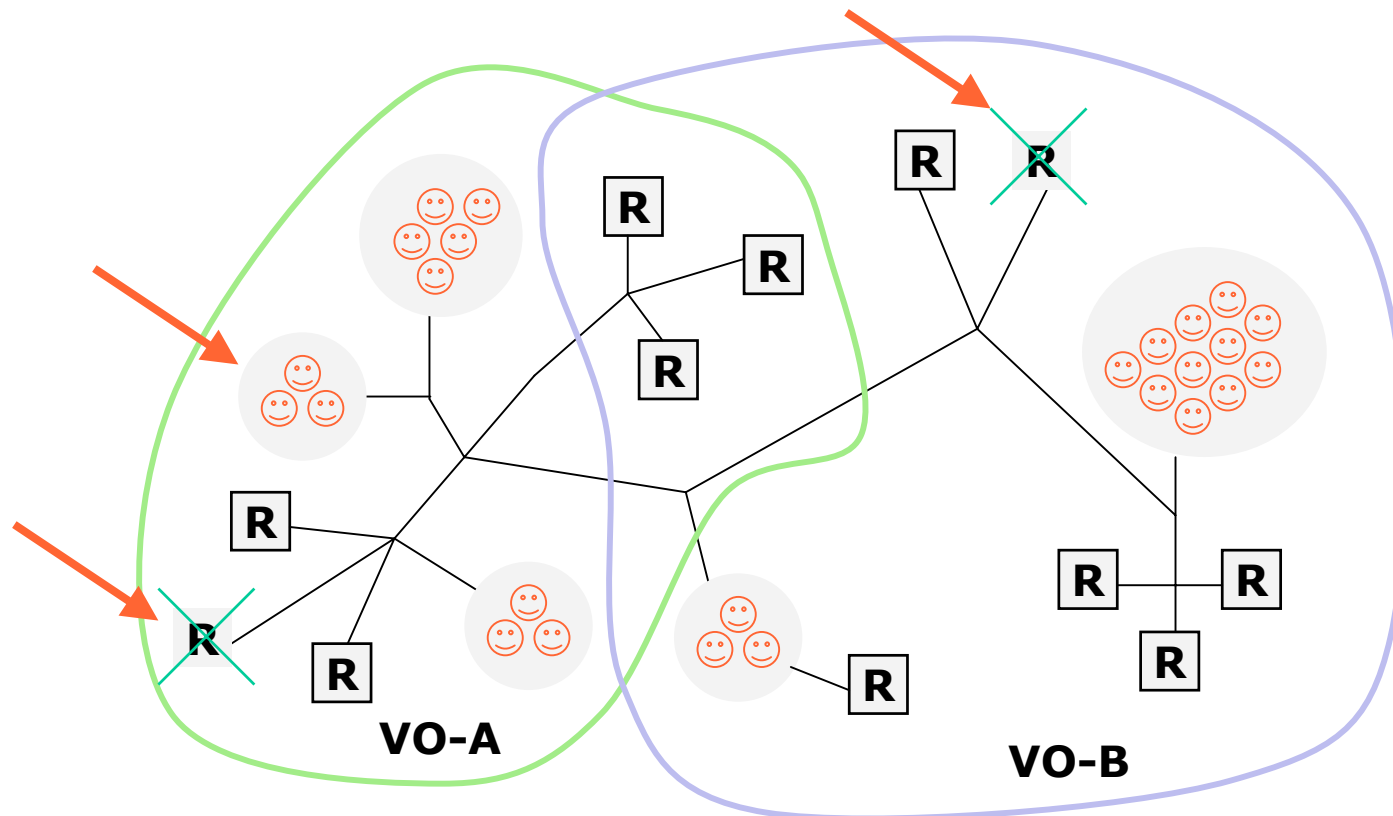# Virtual Organizations

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

Build a Grid Service Using GT3

Virtual Organizations

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

VO-A

VO-B

Build a Grid Service Using GT3
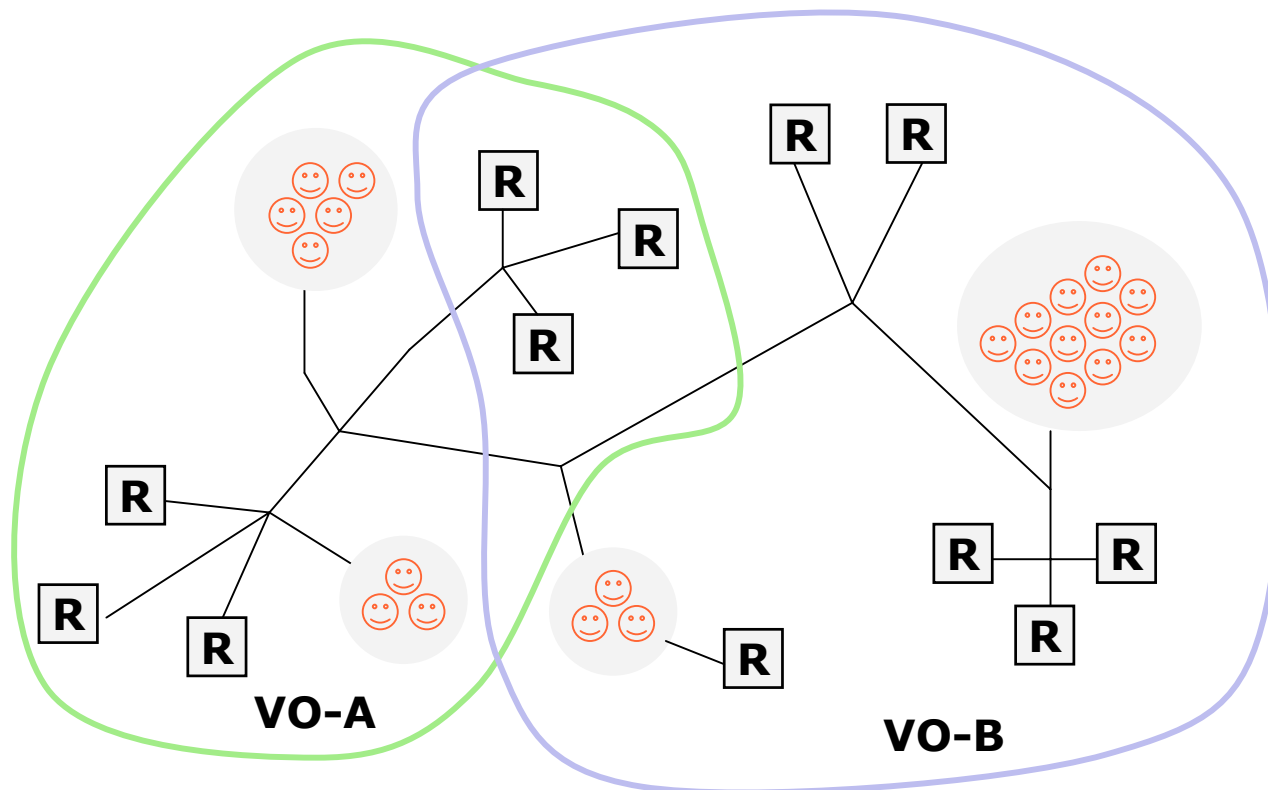
Virtual Organizations

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

Tutorial VO

File Shares

....

Tutorial Index
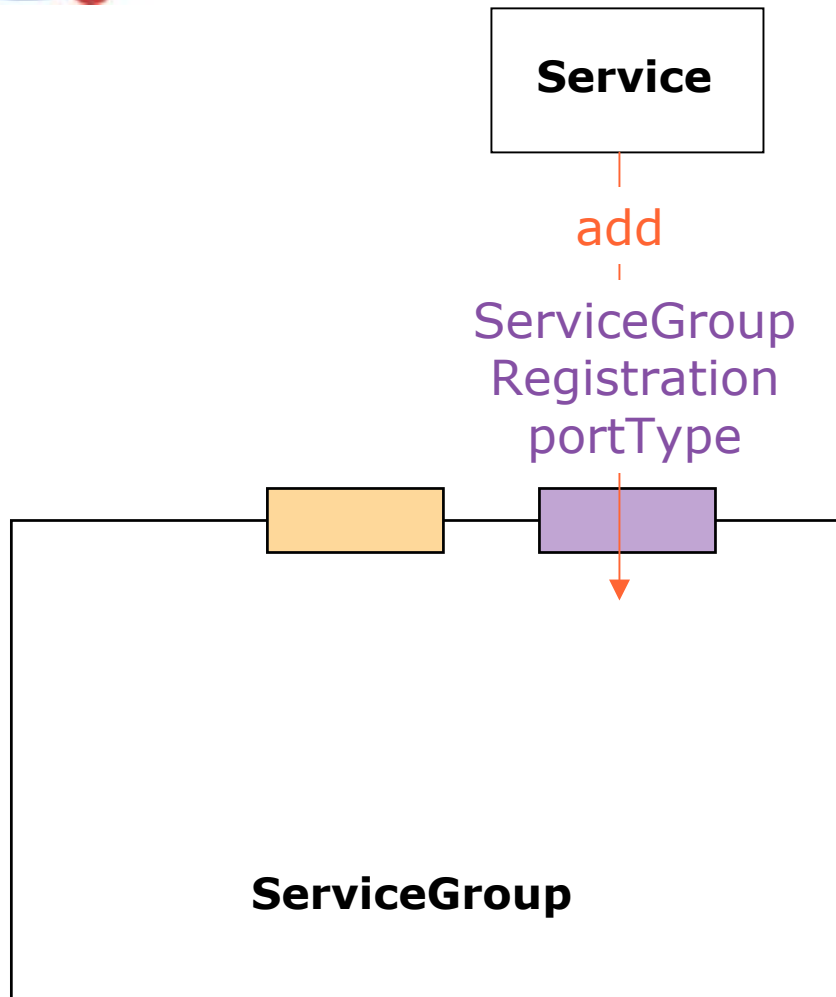
Build a Grid Service Using GT3
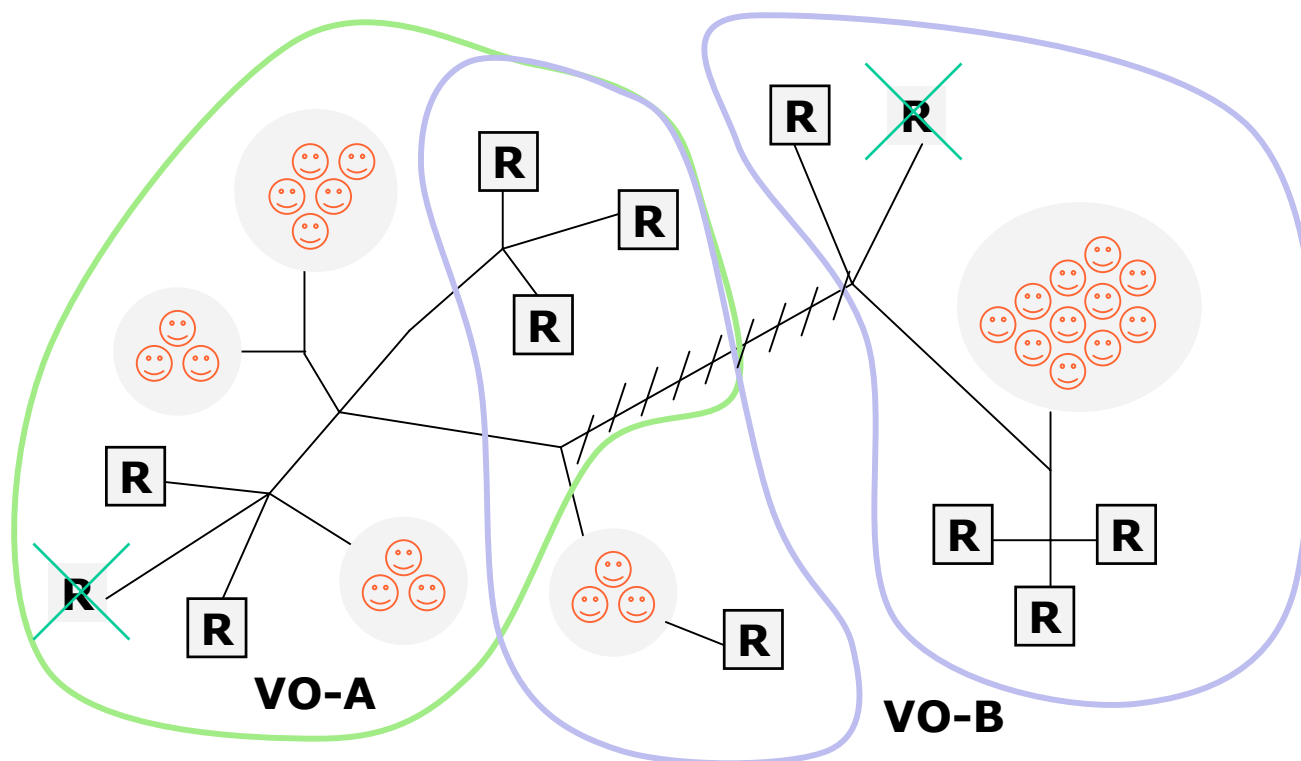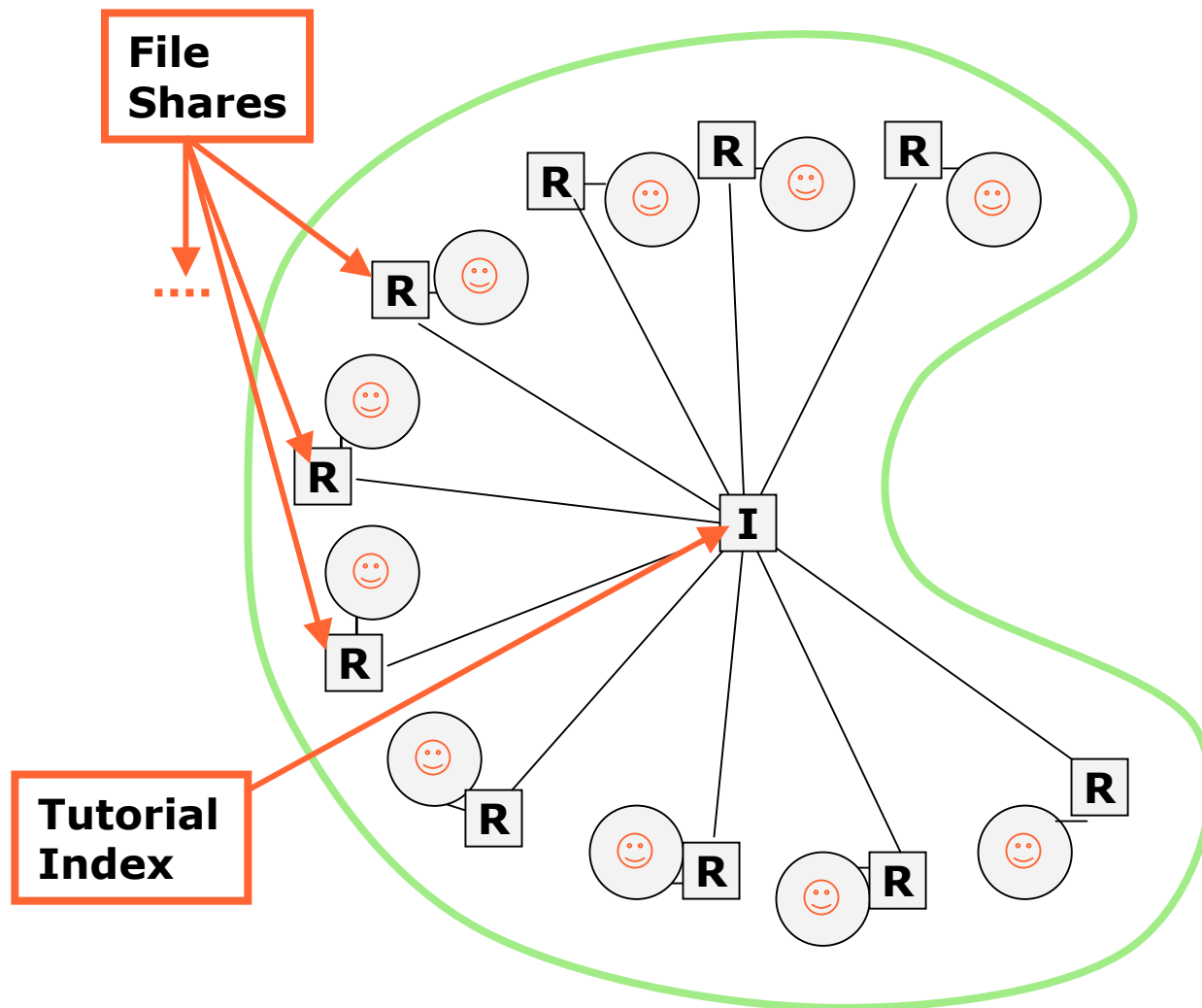
Service Group

- Service Groups represent a collection of services
  - The OGSI spec defines Service Groups as a generic "bag" of entries
  - Developers extend the semantics in ways that are meaningful in their problem space
- The Tutorial Index is written on top of Service Groups
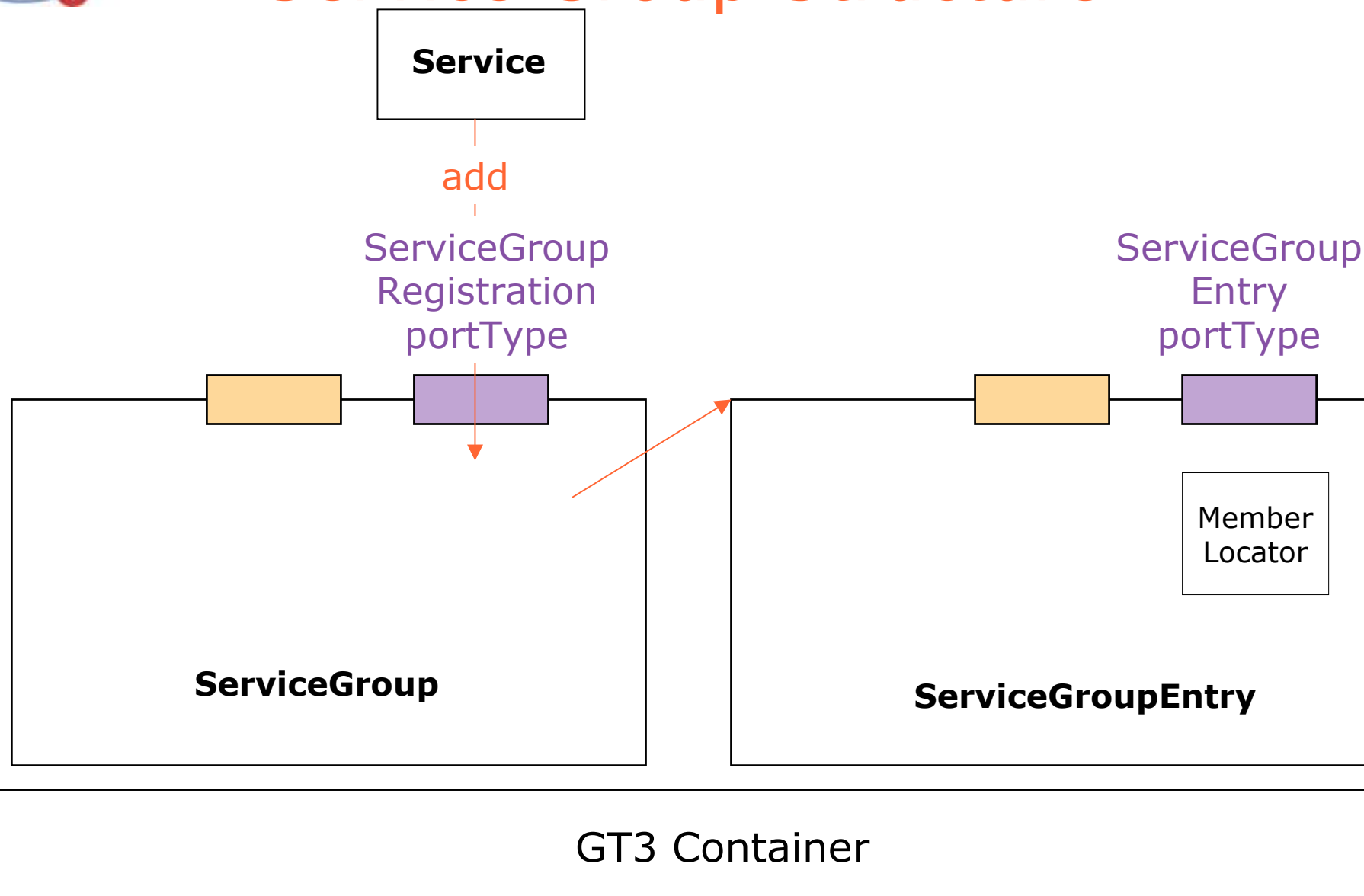
Service Group Structure
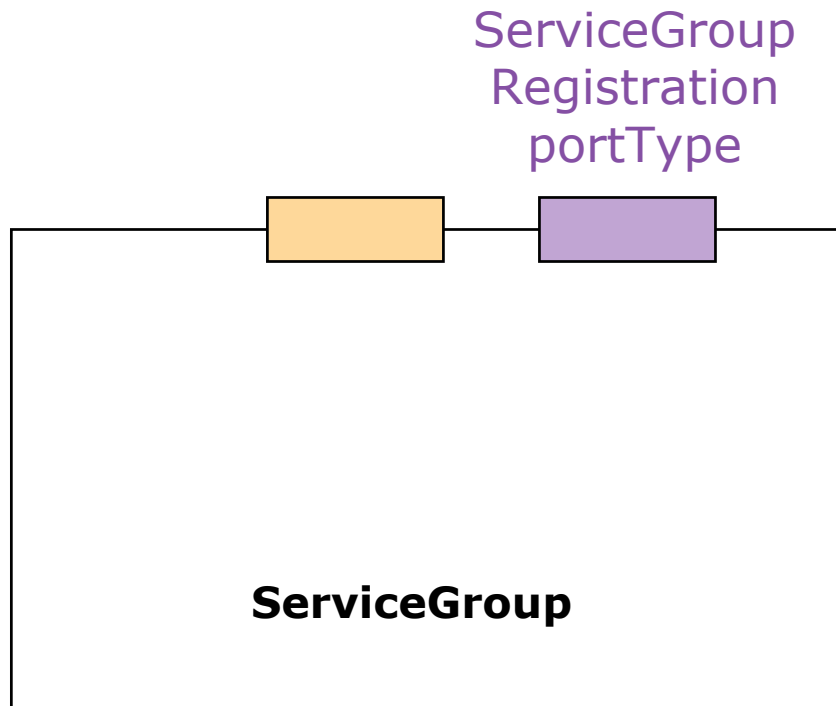
ServiceGroup
Registration
portType
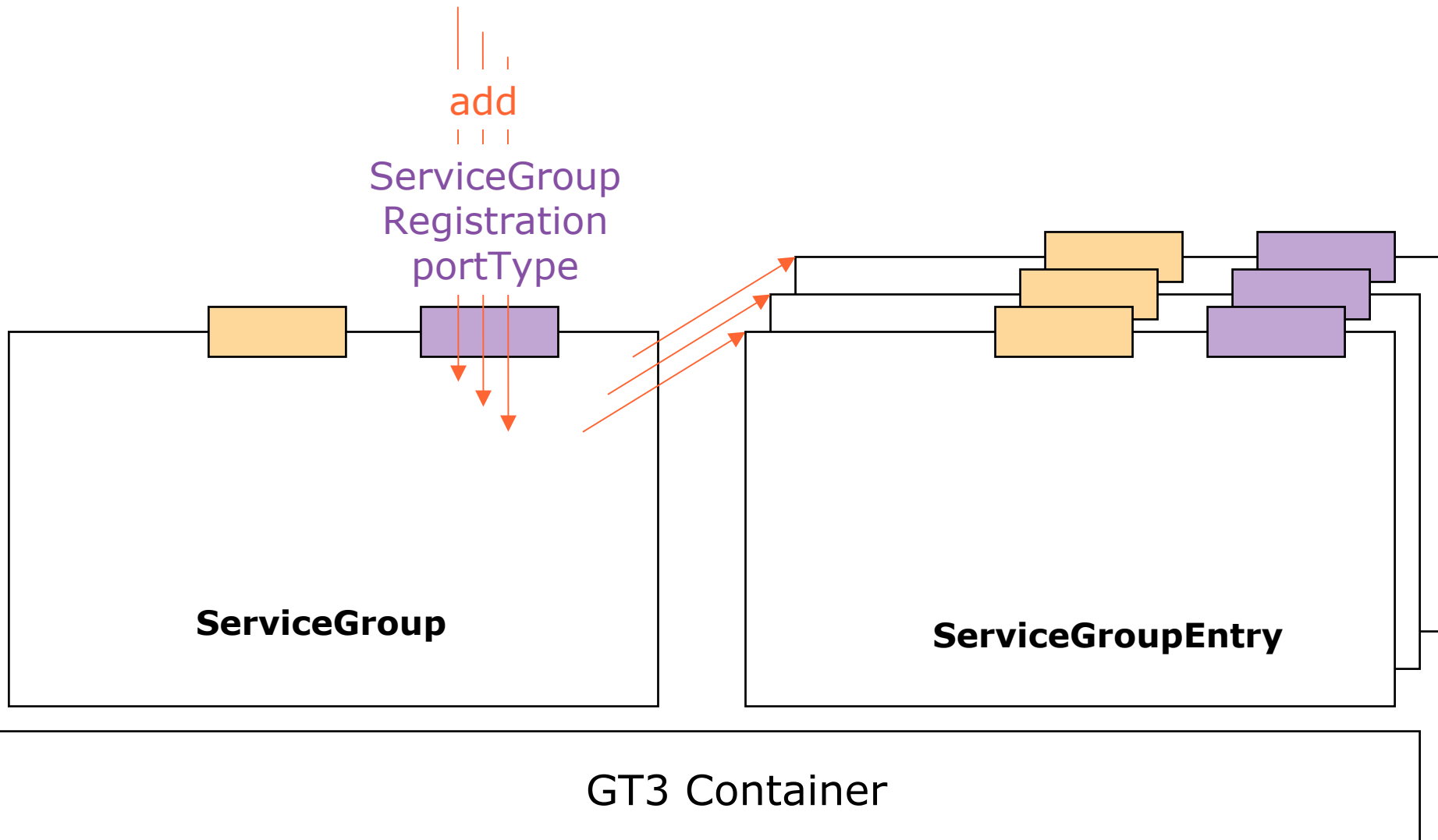
ServiceGroup

GT3 Container

Service Group Structure

Service

add

ServiceGroup
Registration
portType

ServiceGroup

GT3 Container

Service Group Structure

Service

add

ServiceGroup
Registration
portType

ServiceGroup
Entry
portType

ServiceGroup

ServiceGroupEntry

Member
Locator

GT3 Container

Service Group Structure

add

ServiceGroup
Registration
portType

ServiceGroup

**ServiceGroupEntry**

GT3 Container

The Tutorial Index

- The Index will be running on one of the instructor's machines

- The Index represents a collection of services

  - A way for each FileShare to advertise its existence to the VO

  - A copy of each FileShare's service data to be cached in a single place
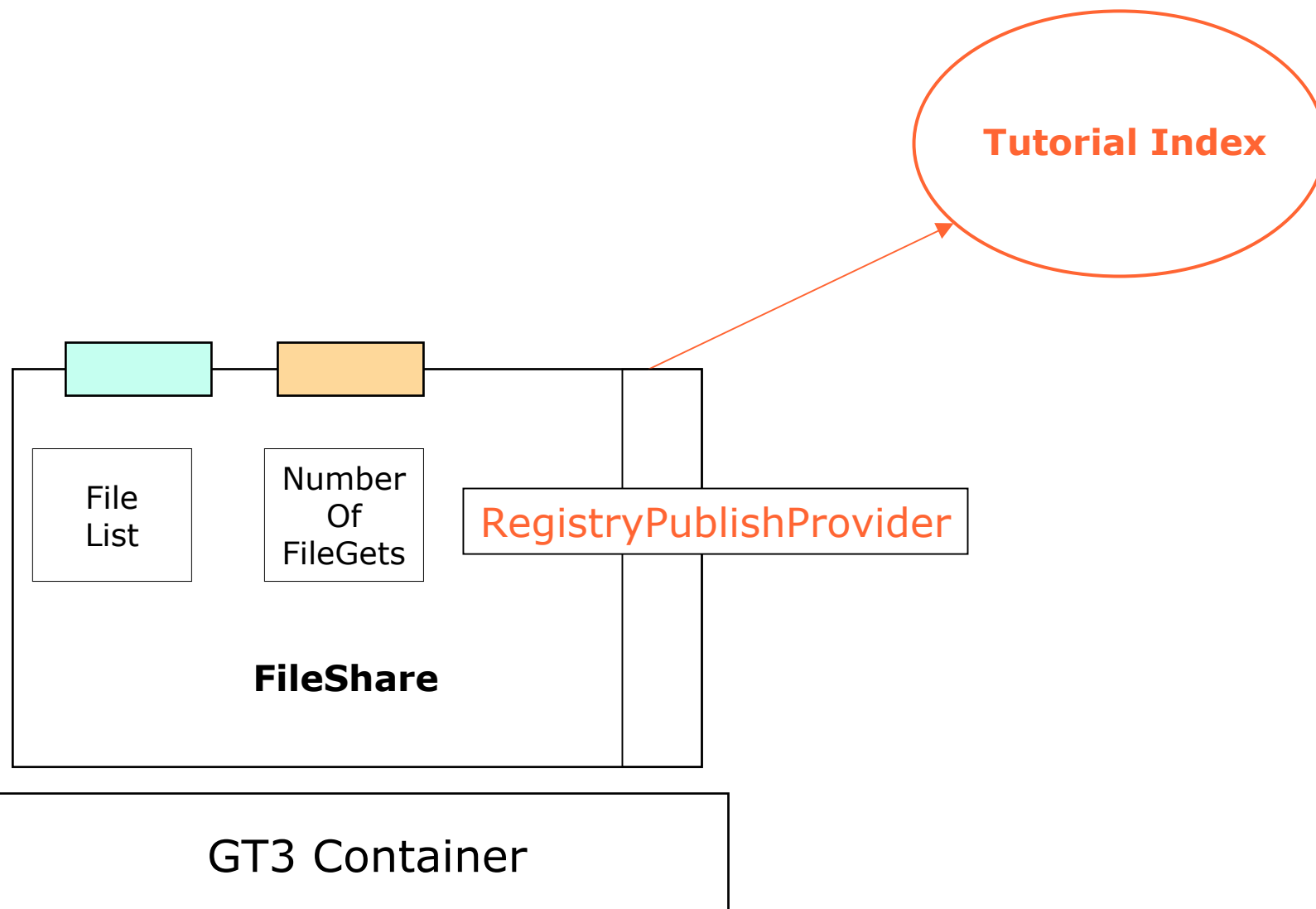
FileShare-Index Interaction

Tutorial Index

FileShare

File List

Number Of FileGets
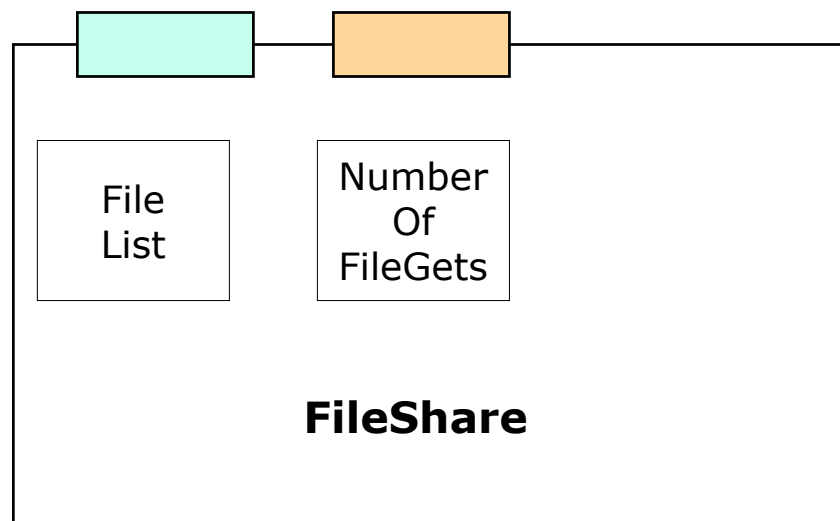
GT3 Container

In order to add our FileShare as an entry in the index we will add an operation provider called RegistryPublishProvider to the definition of the FileShare service

FileShare-Index Interaction

the globus toolkit®
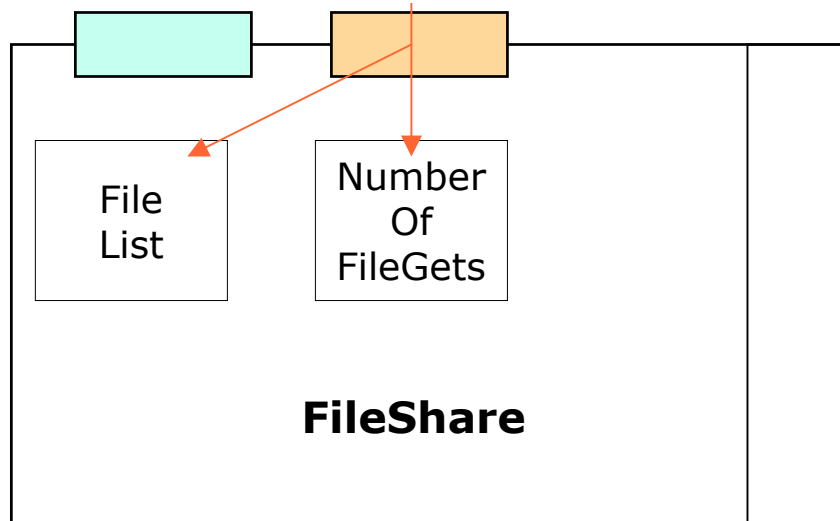www.globustoolkit.org

Tutorial Index

File
List

Number
Of
FileGets

RegistryPublishProvider

FileShare

GT3 Container

FileShare-Index Interaction

findServiceData ——————— **Tutorial Index**

GridService
portType

File
List

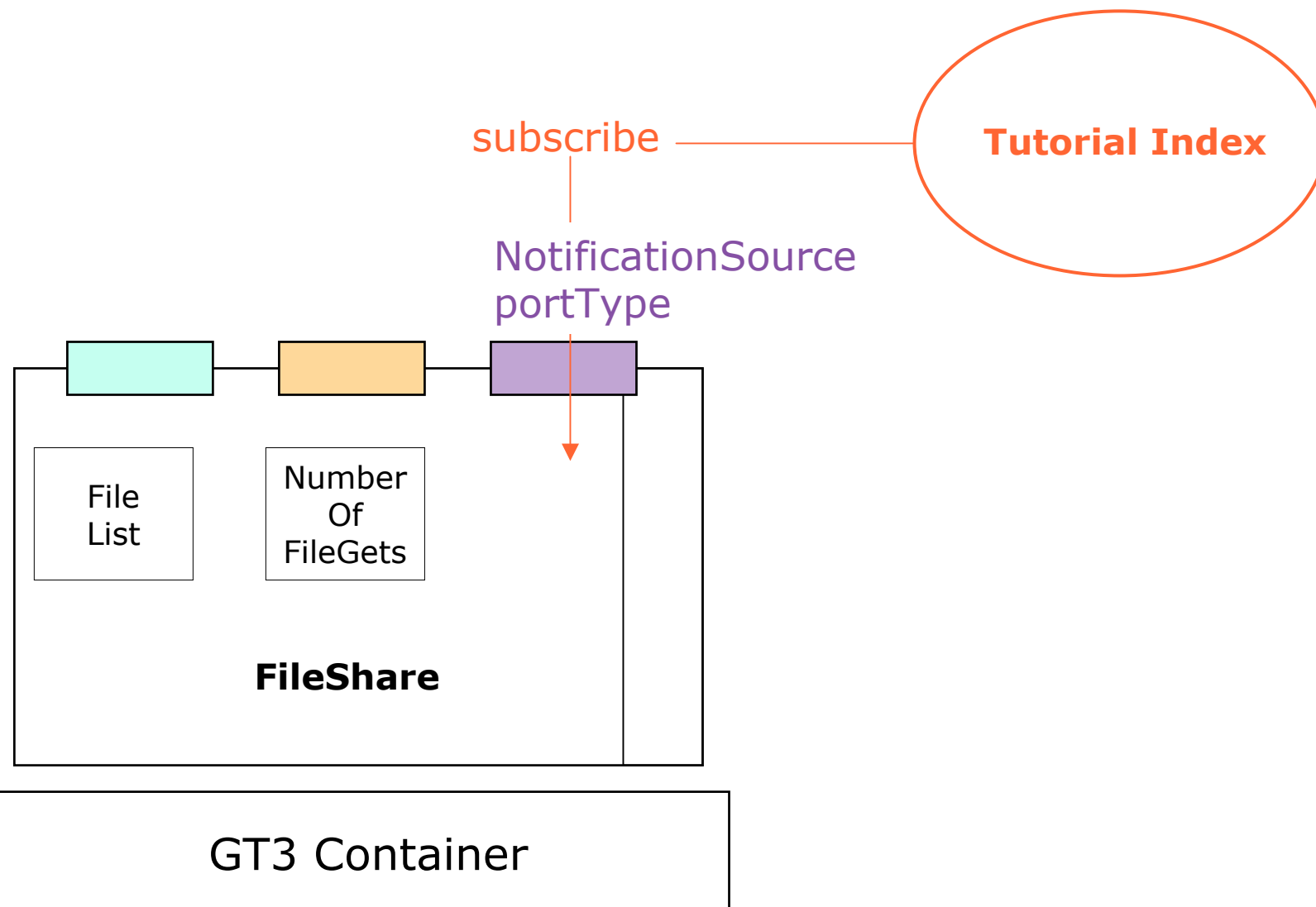Number
Of
FileGets

FileShare

GT3 Container

Once a FileShare service is a member of the Tutorial Index, the index will retrieve the FileShare's service data in order to cache a local copy

For our purposes, the Tutorial Index will need to keep the copies of FileShare service data current
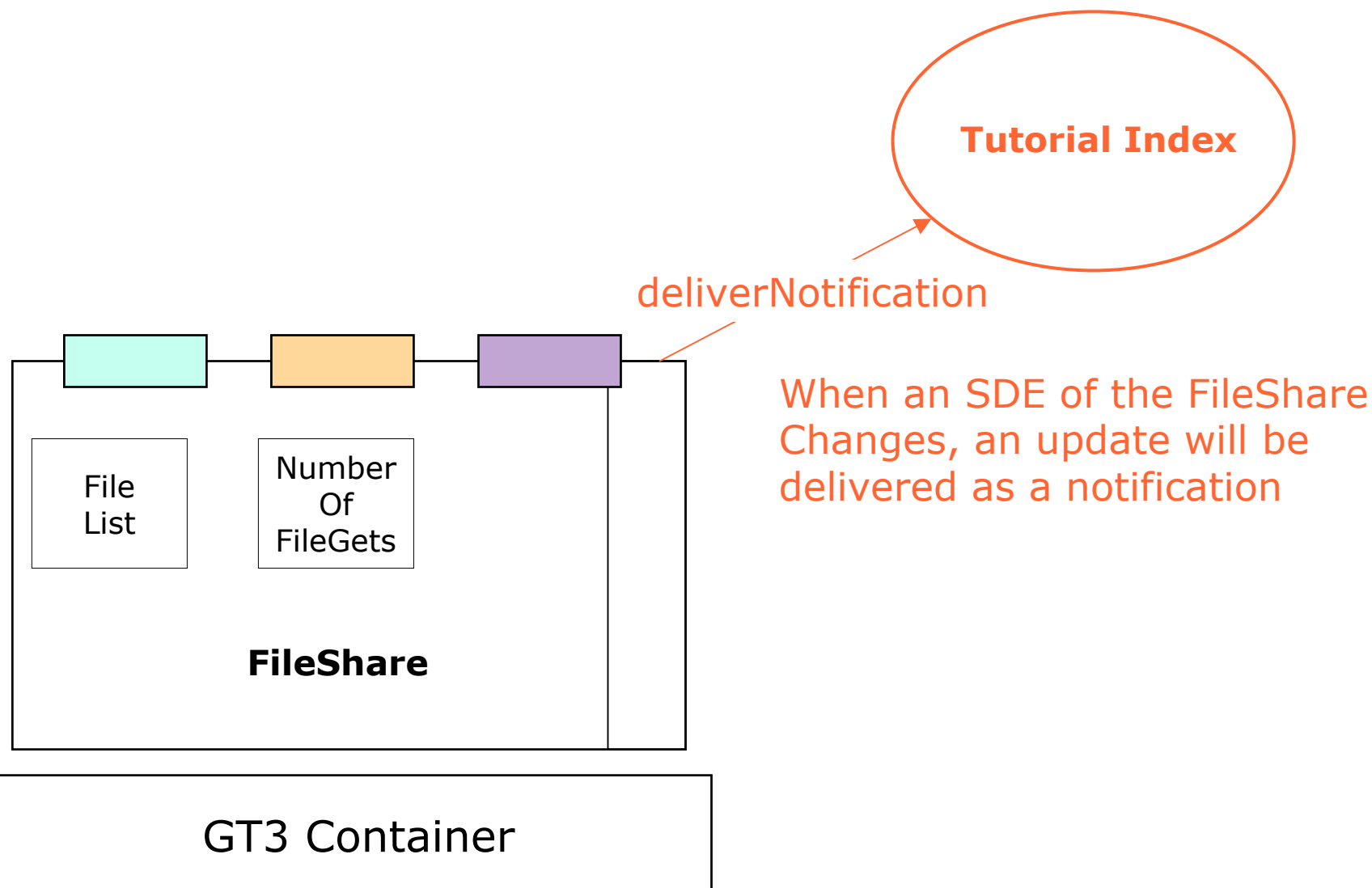
Subscriptions

- NotificationSource portType

  - an optional OGSI interface

- Used by our index to subscribe to the service data of FileShare services

- As part of the OGSI notification pattern, the Tutorial Index implements the NotificationSink portType

FileShare-Index Interaction

the globus toolkit®
www.globustoolkit.org

Tutorial Index

subscribe

NotificationSource
portType

File
List

Number
Of
FileGets

FileShare

GT3 Container

FileShare-Index Interaction

Tutorial Index

deliverNotification

File
List

Number
Of
FileGets

FileShare

When an SDE of the FileShare
Changes, an update will be
delivered as a notification

GT3 Container

Concepts in Exercise

- operationProviders is an entry in the WSDD file

- Operation providers add functionality

- If the additional functionality includes the addition of a public interface, you need to add an entry to the .gwsdl

- If the additional functionality requires runtime parameters, you need to add them to the .wsdd

RegistryPublishProvider

- RegistryPublishProvider requires parameters in wsdd file:
  - **registry** GSH of the Tutorial Index service group (mandatory)
  - **registry-keepalive** set to 'true' to keep registration alive (optional)
  - **registry-lifetime** time parameter to specify how long the entry should live (optional)
  - **registry-remove** 'true' if the entry should be removed at shutdown (optional)
- Defines no operations, does not need anything to be changed in GWSDL

NotificationSourceProvider

- NotificationSourceProvider makes our service data available for subscription
  - Defines an operation that will be used by the Index service we're registering to
  - Therefore, needs to appear in our GWSDL
- In the end, our GWSDL will extend both GridService and NotificationSource
- Does not require any WSDD parameters

What Attendees Should Do

- Uncomment operationProviders and parameters in WSDD

- Add NotificationSource portType to GWSDL

- Deploy, restart container

- Don't need to modify any service code!

Build a Grid Service Using GT3

What Attendees Should See

A representation of your FileShare inside a representation of the tutorial index

Your IP

Number of files shared

Exercise 4 Review

- operationProviders allow you to add functionality without writing code

  – We saw this with RegistryPublish and Notification Source

- Service Groups contain ServiceGroupEntries representing members of the group

- The Tutorial Index uses subscriptions to receive updates

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - **5. Lifetime Management: Maintain service registration**
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
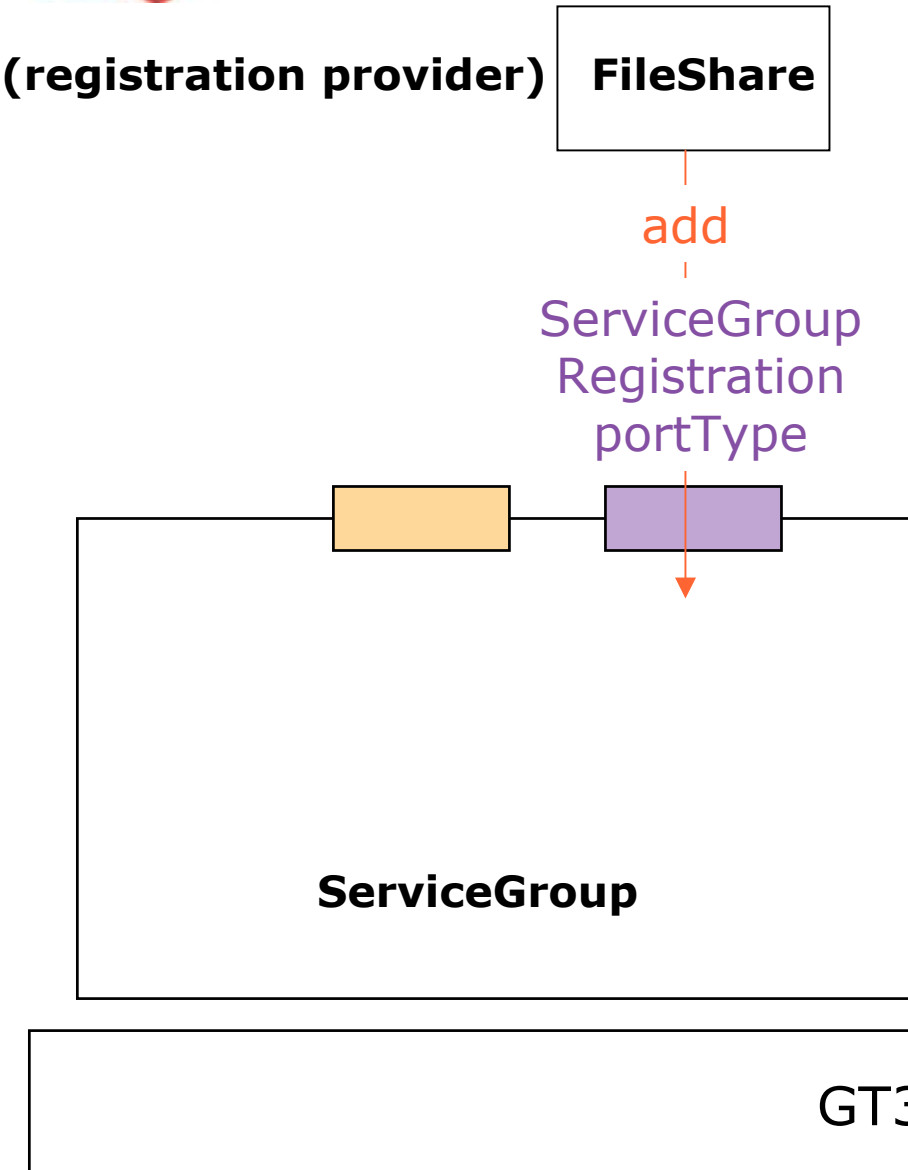- Publish your Grid Service: The GTR
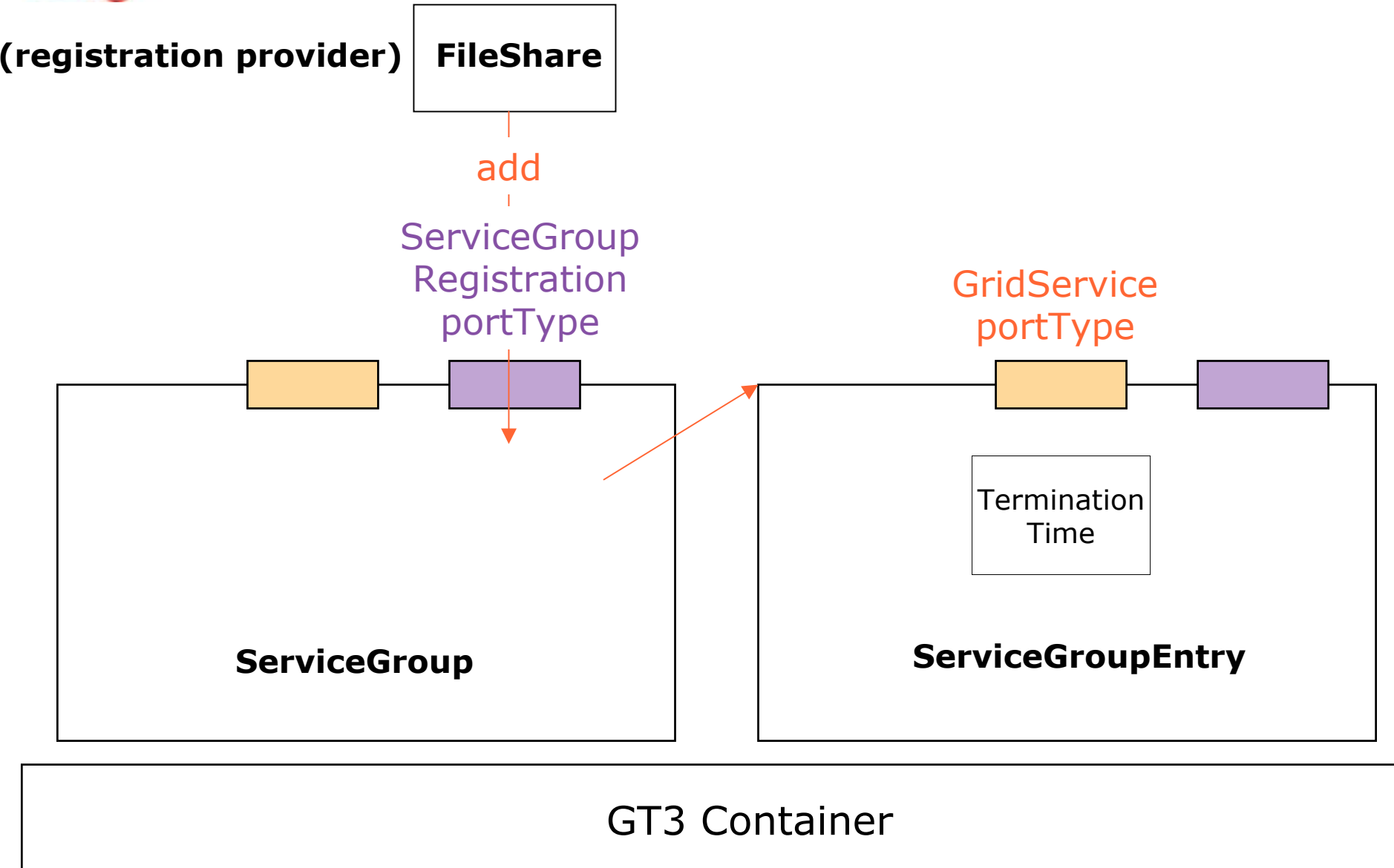
Overview of Tutorial Content

1. *Deployment:* Stand up a FileShare service on your laptop

5. *Lifetime Mgmt:* Maintain service registration

Tutorial Index

4. *Virtual Organization:* Register your service with a community index service

FileShare

3. *Inspection:* Add service data to your service

2. *Service Naming:* Share files using identifiers

Client
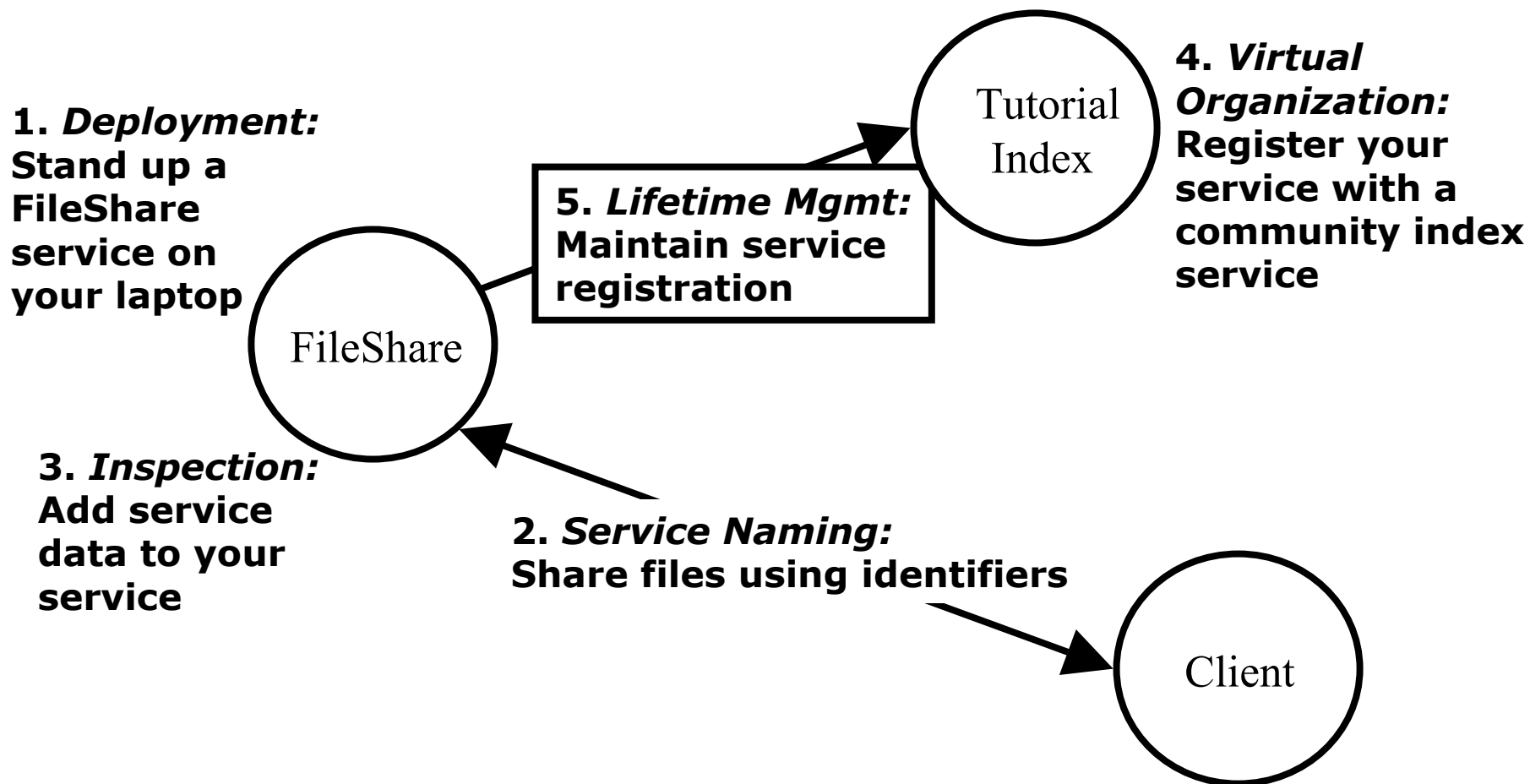
Soft state

- We need a mechanism to clean up old/unwanted state

- A service group has an remove operation, but this is not enough. For example, what happens if a service dies without removing itself?

- All of our state has a limited lifetime. If a service is still alive and wants to remain registered, it must keep indicating its interest in that state.
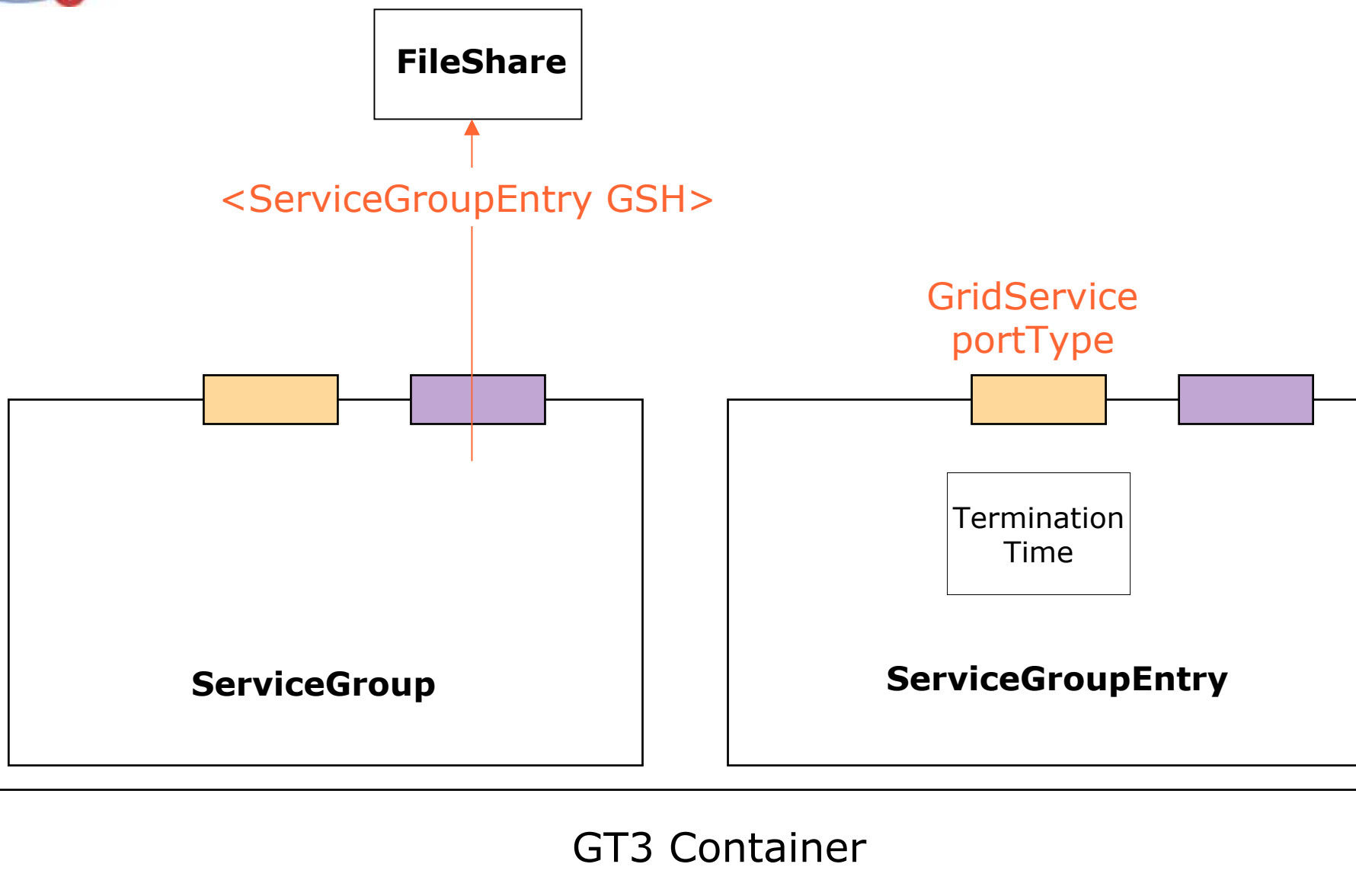
(registration provider) | **FileShare**

add

ServiceGroup
Registration
portType

ServiceGroup

GT3 Container

Lifetime Management

(registration provider) **FileShare**

add

ServiceGroup
Registration
portType

GridService
portType

Termination
Time

ServiceGroup

ServiceGroupEntry

GT3 Container

Build a Grid Service Using GT3

Lifetime Management

Lifetime Management

the globus toolkit®
www.globustoolkit.org

(registration provider) | **FileShare**

requestTerminationAfter

GridService
portType

Termination
Time

ServiceGroup

ServiceGroupEntry

GT3 Container

Lifetime Management

(registration provider)   **FileShare**

Example of soft-state lifetime
management:
  The ServiceGroupEntry goes
  away if the FileShare service
  disappears

requestTerminationAfter

GridService
portType

Termination
Time

ServiceGroup

ServiceGroupEntry

GT3 Container

SDE Attributes

- SDEs can contain OGSI-defined attributes describing quality/lifetime of the data they contain

  - goodFrom: start of validity
  - goodUntil: end of validity
  - availUntil: Will be purged after this time

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - **6. Discovery: Find a File**
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
- Publish your Grid Service: The GTR

Exercise 6: Discovery

1. *Deployment:*
Stand up a
FileShare
service on
your laptop

FileShare

Tutorial
Index

4. *Virtual*
Organization:
Register your
service with a
community index
service

5. *Lifetime Mgmt:*
Maintain service
registration

6. *Discovery:*
Find services that
publish the file
you wish to retrieve

3. *Inspection:*
Add service
data to your
service

2. *Service Naming:*
Share files using identifiers

Client

the globus toolkit®
www.globustoolkit.org

What is Discovery?

- We want to find a service that has some property

- In our case, a person wants to find a service that is sharing a particular filename ('haggis.txt')

- The Tutorial Index knows about of all the services and which files are available

- We can search the Tutorial Index for the service that is publishing 'haggis.txt'

- The identification of the service(s) meeting our criteria is called Discovery

Implementation Details

- The Tutorial Index publishes all of the information it has as service data (an SDE called ogsi:entry)

- We can use standard service data querying methods for accessing the Tutorial Index's data

- However, our ogsi:entry will contain a large amount of data; it will be necessary to search inside of the SDE to find the data we need

Searching Inside an SDE

- To search the SDE data we can use a GT3-specific query mechanism that allows us to assemble an XPath query

    An XPath overview in two bullets:

    > XPath is a convenient query language for searching XML documents

    > XPath queries are formed by identifying a route to the desired data

We shall provide you with an XPath query to search the SDE of the Tutorial Index…

Searching the SDE of the Tutorial Index

- We can find the service that is sharing 'haggis.txt' by delivering the following XPath query to the Tutorial Index:

/ogsi:entry[ogsi:content/ns1:serviceDataValues/ fs:FileList/item/fs:filename= 'haggis.txt']/ogsi:memberServiceLocator

A human translation of this syntax:

"Select all the services that have a filename of 'haggis.txt' in their FileList SDE, and then return the locators to those services"

What Attendees Should Do

- Create a uniquely-named file in $GLOBUS_LOCATION

- Use the GetFileIndex client to perform an XPath query against the tutorial Index to retrieve the newly created file

- Use GetFileIndex to retrieve a file from your neighbor

Exercise 6 Review

- The Indexing Service Group provides a way to discover services based on SDEs

- The IndexingServiceGroup aggregates the SDEs of services registering to it

- Because SDEs are XML, GT3 allows XPath queries for searching and retrieving SDEs

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - **7. GT3 Security: Share Files Securely**

  Time permitting:
  - 8. Transience: Create and Destroy FileShares
- Publish your Grid Service: The GTR

Exercise 7: Security

1. *Deployment:* Stand up a FileShare service on your laptop

5. *Lifetime Mgmt:* Maintain service registration

4. *Virtual Organization:* Register your service with a community index service

Tutorial Index

FileShare

7. *GT3 Security:* Secure your service

6. *Discovery:* Find services that publish the file you wish to retrieve

3. *Inspection:* Add service data to your service

2. *Service Naming:* Share files using identifiers

Client

GT3 Security Details

- Built on top of PKI
  - Each entity has two keys: public and private
  - Data encrypted with one key can only be decrypted with other
  - The private key is known only to the entity
- The public key is given to the world encapsulated in a X.509 certificate

Certificates

- A X.509 certificate binds a public key to a name

- It includes a name and a public key bundled together and signed by a trusted party (Certificate Authority)

- An example of a Distinguished Name (DN):

  "/O=Tutorial/OU=User/CN=Charles Bacon"

Certificate Authorities

- A Certificate Authority (CA) signs certificate requests

- To verify a certificate signature, you must have a copy of the CA certificate

- By default, stored in /etc/grid-security/certificates

- For our tutorial, stored in $TUTORIAL_LOCATION/certificates

Proxy Certificates

- Proxy certificates contain a new keypair, and are signed by the original certificate
  - Also has shorter lifetime
  - Stored in /tmp/x509up_u$UID
  - Protected by filesystem permissions
- Create a proxy using org.globus.tools.ProxyInit
  - Full GT3 install includes C command line clients as well

Service-side Authentication

- XML configuration files designed to set security parameters for a service
- Allows per-method authentication settings
- auth-method
  - none: no authentication
  - pkey: GSI Secure Message
  - gsi: GSI Secure Conversation
- run-as
  - caller: Execute method with caller's credential
  - system: Execute method with container credential
  - service: Execute method with service credential
- Need to mention the XML configuration file in the wsdd as securityConfig parameter

Client-side Authentication

- Can set authentication properties programmatically

- For example, our GetFile client will set GSI Secure Conversation authentication

  - ((Stub)portType)._setProperty(Constants.GSI_SEC_CONV, Constants.ENCRYPTION);

Authorization

- **GT3 allows for different authorization methods**
  - Client
    - > None: no authorization will be performed
    - > Self: service will be authorized if it has the same identity as the client
    - > Host: service will be authorized if the host returns an identity containing the hostname
  - Server
    - > None: no authorization will be performed
    - > Self: client will be authorized if it has the same identity as the service
    - > Gridmap: User will be authorized as identity listed in gridmap

Gridmap Files

- A mapping from certificate subject names to local resource identities
    - "/O=Tutorial/OU=User/CN=Charles Bacon" bacon
- Used in the gridmap authorization methods
- Each service may have its own gridmap, specified by the gridmap parameter
- Allows per-site authorization
    - Decentralized control required for VOs

What Attendees Should Do

- Uncomment the securityConfig parameter in WSDD

- Uncomment the security code in GetFile.java

- Try to GetFile from your service

- Try querying the SDEs of your service

- Create a proxy, then try both again

What Attendees Should See

- Without a proxy, you cannot interact with your service

- With a proxy, both operations are successful

Exercise 7 Review

- Service security is configured through parameters in the WSDD file, and in the securityConfig XML file

- Client security is configured by setting properties in the Java code

- Service-side authentication may be specified on a per-operation basis

How to Build a Grid Service Using GT3

- Overview of Grid Services and GT3
- Build a Grid Service
  - Overview
  - 1. Deployment: Stand Up a FileShare Service
  - 2. Naming: Share Files using Identifiers
  - 3. Inspection: Add Service Data
  - 4. Virtual Organization: Register with a Community Index
  - 5. Lifetime Management: Maintain service registration
  - 6. Discovery: Find a File
  - 7. GT3 Security: Share Files Securely

  Time permitting:
  - **8. Transience: Create and Destroy FileShares**
- Publish your Grid Service: The GTR

Exercise 8: Transience

the globus toolkit®
www.globustoolkit.org

8. *Transience:* Create and destroy services

Factory

1. *Deployment:* Stand up a FileShare service on your laptop

Tutorial Index

4. *Virtual Organization:* Register your service with a community index service

5. *Lifetime Mgmt:* Maintain service registration

FileShare

7. *GT3 Security:* Secure your service

6. *Discovery:* Find services that publish the file you wish to retrieve

3. *Inspection:* Add service data to your service

2. *Service Naming:* Share files using identifiers

Client

Transience

- FileShare is persistent, with one instance running whenever the container is running

- However, OGSI services can be transient
  - Transience allows for the dynamic creation and destruction of services

- OGSI includes a Factory pattern in order to support service transience

Factory

- The OGSI Factory portType supports a createService operation

  - As a result a new service is created

- FactoryProvider is the GT3 implementation of the OGSI portType; it defines a createServiceObject method for service instance creation

Implementing the Factory Pattern

Create a FileShareFactory that includes an createServiceObject method which knows how to instantiate a FileShare

Factory portType

FileShareFactoryCallback:: createServiceObject

FileShareFactory

FileShare

The Factory Pattern Runtime

Factory
portType

**FileShareFactoryCallback::
createServiceObject**

FileShareFactory

GT3 Container

The Factory Pattern Runtime

Client

createService

Factory
portType

**FileShareFactoryCallback::
createServiceObject**

FileShareFactory

GT3 Container

The Factory Pattern Runtime

Client

createService

Factory
portType

**FileShareFactoryCallback::
createServiceObject**

FileShareFactory

FileShare

GT3 Container

The Factory Pattern Runtime

Client

<FileShare GSH>

Factory portType

FileShareFactoryCallback::createServiceObject

FileShareFactory

FileShare

GT3 Container

The Factory Pattern Runtime

Client

getFile

FileShare
portType

**FileShareFactoryCallback::
createServiceObject**

FileShareFactory

FileShare

GT3 Container

The Factory Pattern Runtime

createService

Factory
portType

**FileShareFactoryCallback::
createServiceObject**

FileShareFactory

FileShare

GT3 Container

Factory Operation Provider

- The FactoryProvider operation provider accepts a single parameter: "factoryCallback"

- factoryCallback should be a java class supporting an "createServiceObject" method
  - responsible for creating the service

GT3 Notification Factory Implementation

- We won't need to write the code for the factory

  – org.gridforum.ogsi.NotificationFactory

- We also don't have to write the WSDL

  – schema/ogsi/ogsi_notification_factory_service.wsdl

- The code we need to write is the class used by the factory callback

Additional WSDD Parameters for Factories

- Factories get two sets of parameters
  - One for them for the factory itself
  - One for the services it creates

- Parameters for created service are prefixed with "instance-"
  - \
  - \

Creating Services From a Factory

- Now we have an extra step before running our client
  - The "createService" method on the File Share Service Factory creates a new FileShare, returning the GSH of the FileShare service
- With the GSH, we can invoke the rest of our operations as usual
- Our service will have a termination time set by the factory
  - The default lifetime for the GT3 Factory implementation is infinity; this default may be overridden
  - If the service has an expiration time, the lifetime must be updated to avoid it being recycled

What Attendees Should Do

- Modify the WSDD to create a factory

- Use the CreateFileShare client to construct a FileShare service for a particular directory

- Retrieve a file from the newly-created service

- Attempt to connect to a neighbor's factory

- Retrieve from a neighbor's created service via the Index client

What Attendees Should See
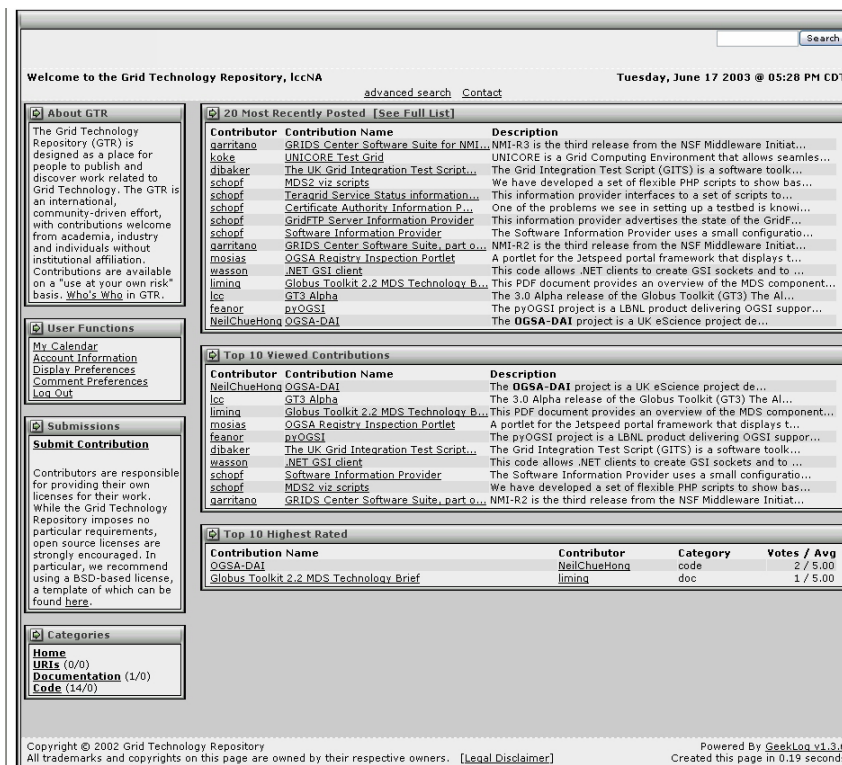
- Services created by the factory behave like ordinary FileShare services

- Can have separate security settings for factories and the services they create
  - Our factories have "self" authorization
  - The FileShare services they create have "none" authorization

The OGSI Grid Service
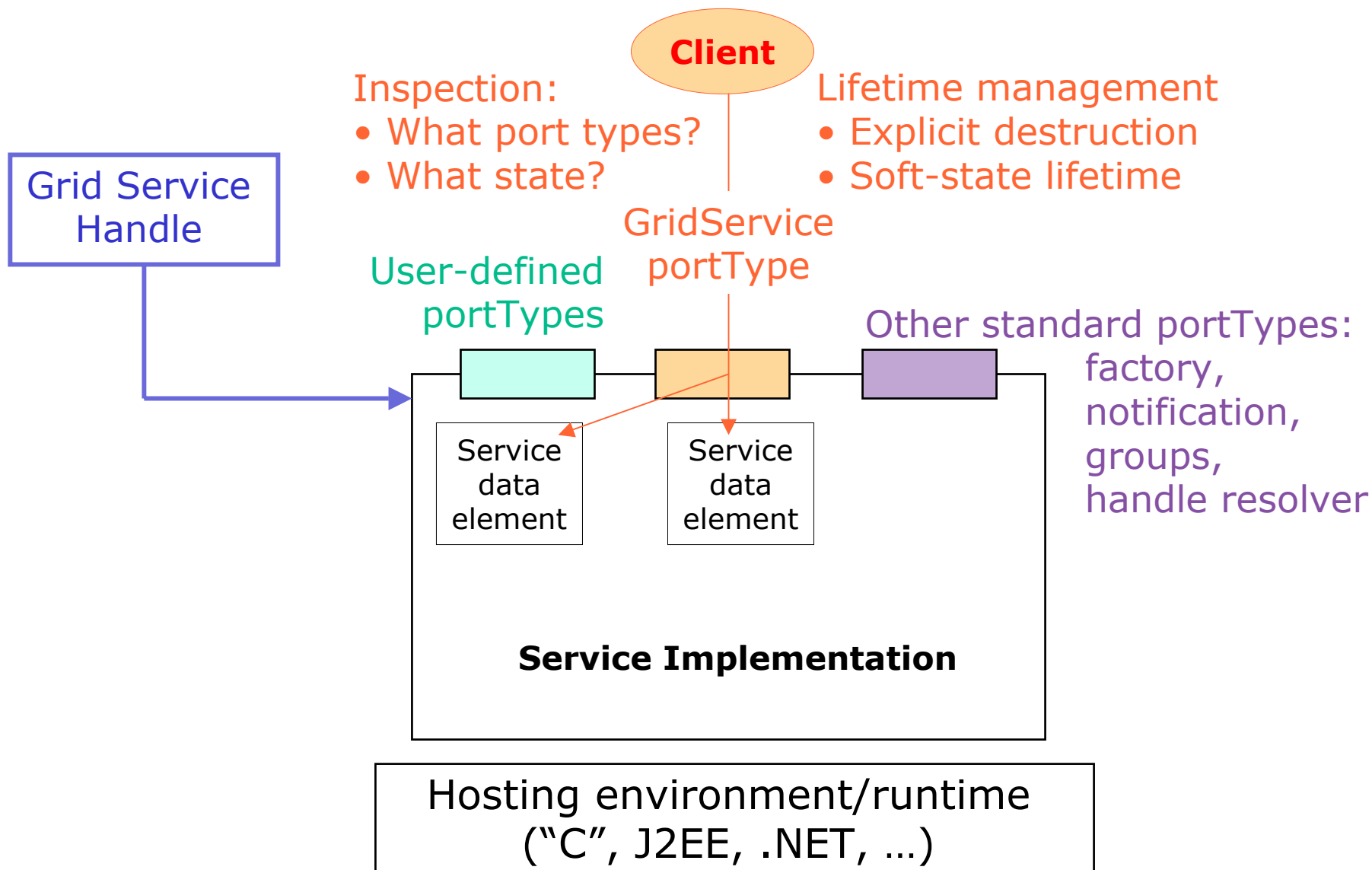
The Grid Technology Repository

- Designed to facilitate code reuse

- Appropriate for publishing OperationProviders and GridServices

- Low barrier to publish

- Licensing term set by contributors

http://gtr.globus.org   http://gtr.nesc.ac.uk

Bibliography

A Web-based GT3 "Build a Grid Service" tutorial

http://www.casa-sotomayor.net/gt3-tutorial/

GT3 documentation page

http://www-unix.globus.org/toolkit/documentation.html

Globus Alliance publications page

http://www.globus.org/research/papers.html

A definition: What is the Grid? *I.Foster, July 2002*

http://www-fp.mcs.anl.gov/~foster/Articles/WhatIsTheGrid.pdf