



Data Management



The European DataGrid Project Team

<http://www.eu-datagrid.org>

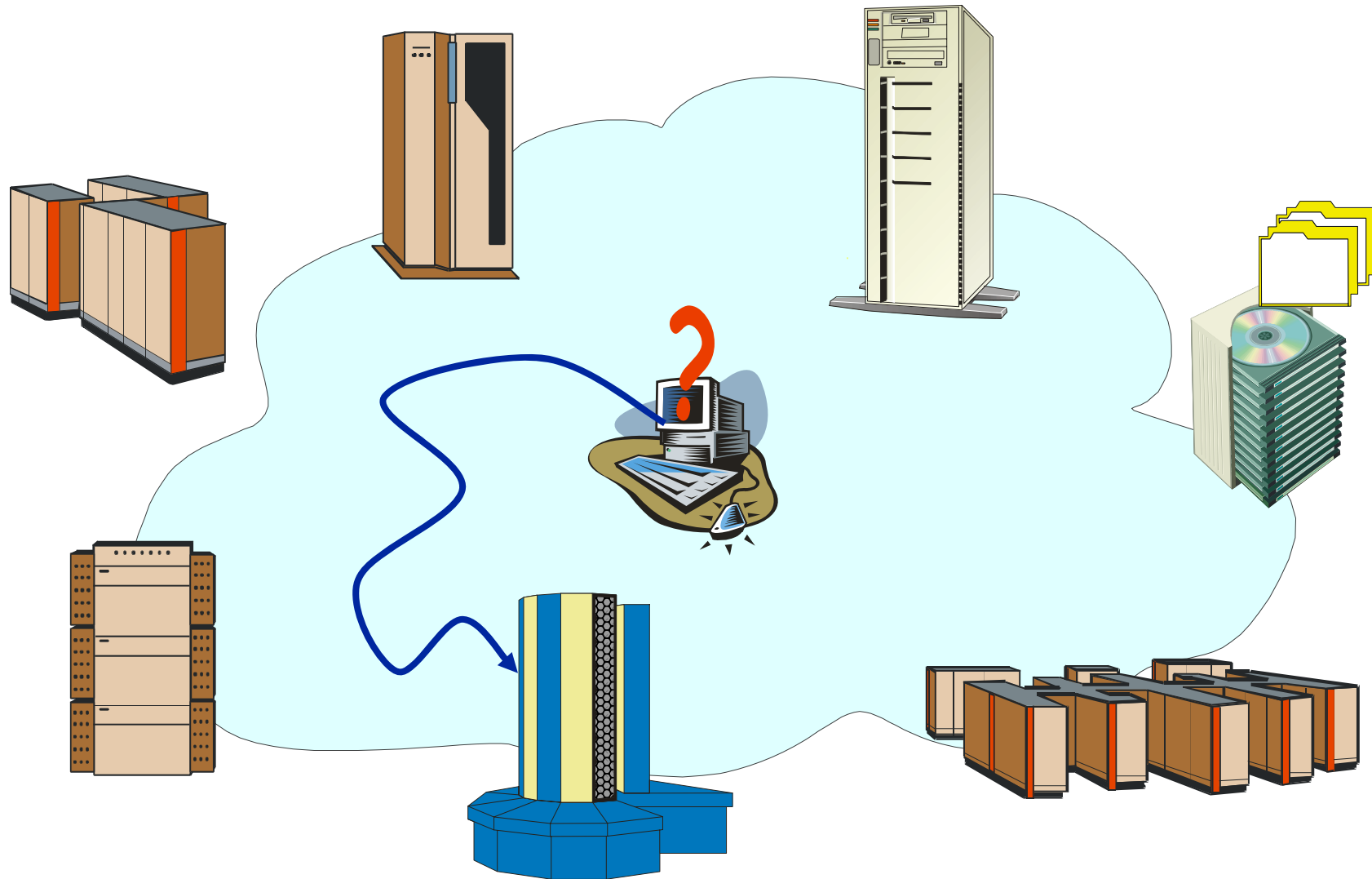


Overview

- Data Management Issues
- Main Components
 - EDG Replica Catalog
 - EDG Replica Manager
 - GDMP

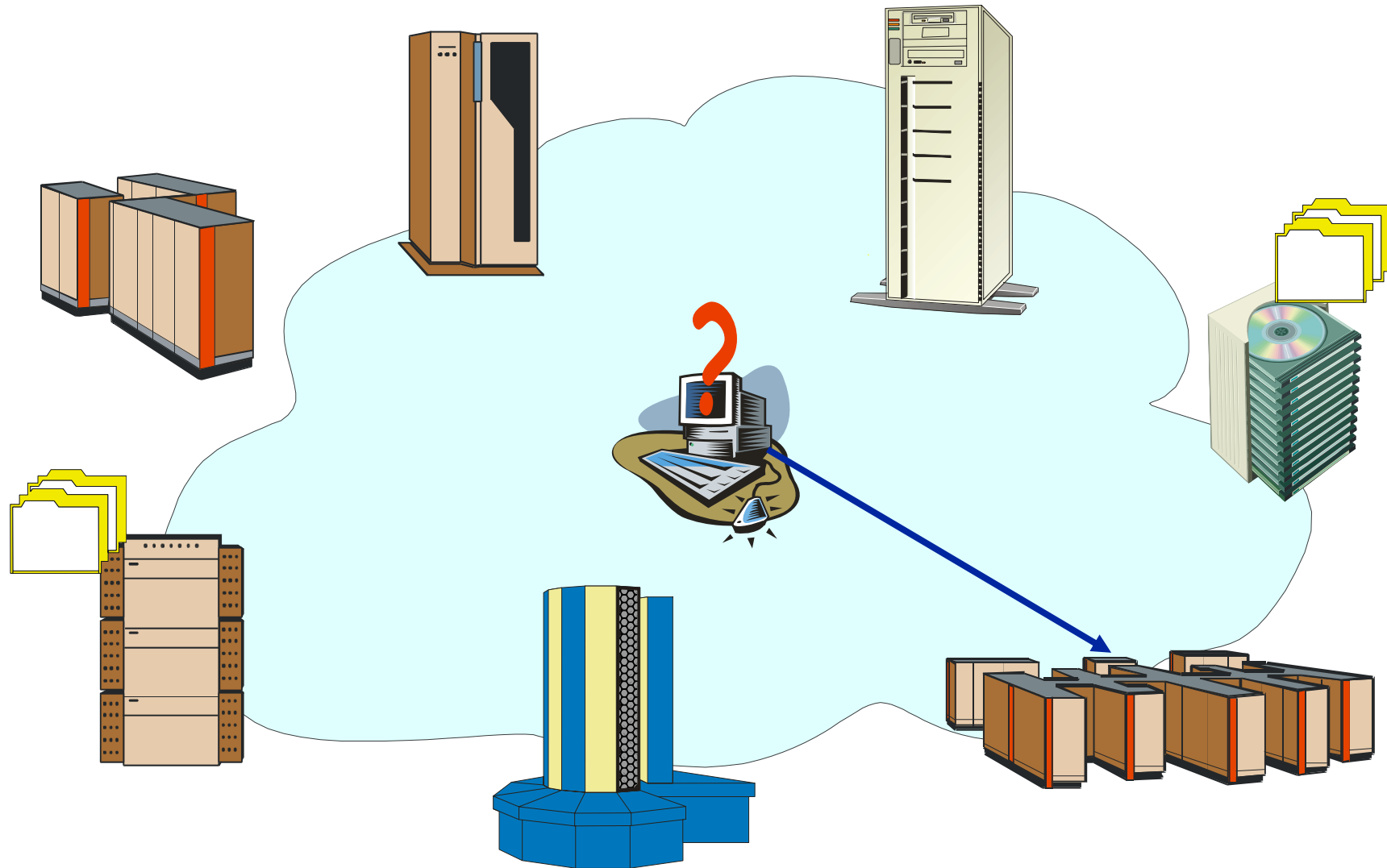


Data Management Issues





Data Management Issues





Data Management Tools


- Tools for
 - Locating data
 - Copying data
 - Managing and replicating data
 - Meta Data management


- On EDG Testbed you have
 - EDG Replica catalog
 - globus-url-copy (GridFTP)
 - EDG Replica Manager
 - Grid Data Mirroring Package (GDMP)
 - Spitfire



EDG Replica Catalog

- Based upon the Globus LDAP Replica Catalog
- Stores LFN/PFN mappings and additional information (e.g. filesize):
 - Physical File Name (PFN): host + full path & and file name
 - Logical File Name (LFN): logical name that may be resolved to PFNs
 - LFN : PFN = 1 : n
- Only files on storage elements may be registered
- Each VO has a specific *storage dir* on an SE
- Example PFN:
`lxshare0222.cern.ch/flatfiles/SE1/iteam/file1.dat`


host


storage dir
- LFN must be full path of file starting from storage dir
LFN of above PFN: `file1.dat`



EDG Replica Catalog

➤ API and command line tools

- `addLogicalFileName`
- `getLogicalFileName`
- `deleteLogicalFileName`
- `getPhysicalFileName`
- `addPhysicalFileName`
- `deletePhysicalFileName`
- `addLogicalFileAttribute`
- `getLogicalFileAttribute`
- `deleteLogicalFileAttribute`

<http://cmsdoc.cern.ch/cms/grid/userguide/gdmp-3-0/node85.html>



globus-url-copy

- Low level tool for secure copying

```
globus-url-copy <protocol>://<source file> \  
                <protocol>://<destination file>
```

- Main Protocols:

- gsiftp - for secure transfer, only available on SE and CE
- file - for accessing files stored on the local file system on e.g. UI, WN

```
globus-url-copy file://`pwd`/file1.dat \  
                gsiftp://lxshare0222.cern.ch/ \  
                flatfiles/SE1/EDGTutorial/file1.dat
```




The EDG Replica Manager

- Extends the Globus replica manager
- Only client side tool
- Allows replication (copy) and registering of files in RC
- Keeps RC consistent with stored data.



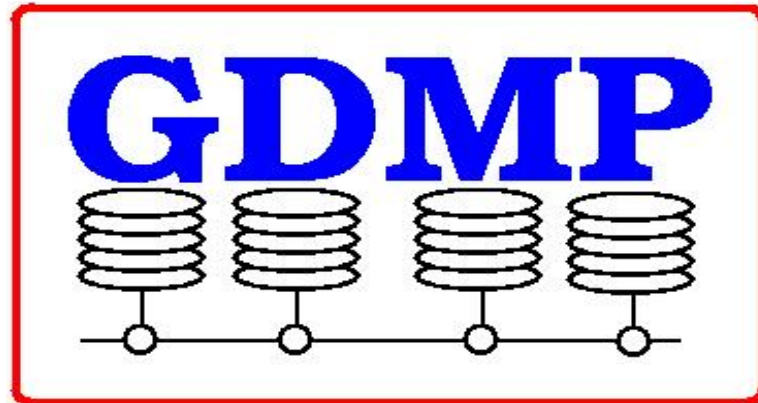
The Replica Manager APIs

- *(un)registerEntry(LogicalFileName **lfn**,
FileName **source**)*
 - Replica Catalogue operations only - no file transfer
- *copyFile(FileName **source**,
FileName **destination**,
String **protocol**)*
 - allows for third-party transfer
 - transfer between:
 - two StorageElements or
 - ComputingElement and Storage Element
 - Space management policies under development
 - all tools support **parallel streams** for file transfers



The Replica Manager APIs

- *copyAndRegisterFile(LogicalFileName **lfn**,
FileName **source**,
FileName **destination**,
String **protocol**)*
 - third-party transfer but :
 - files can only be registered in Replica Catalogue if **destination** PFN contains a valid **SE** (i.e. needs to be registered in the RC)!
- *replicateFile(LogicalFileName **lfn**,
FileName **source**,
FileName **destination**,
String **protocol**)*
- *deleteFile(LogicalFileName **lfn**,
FileName **source**)*

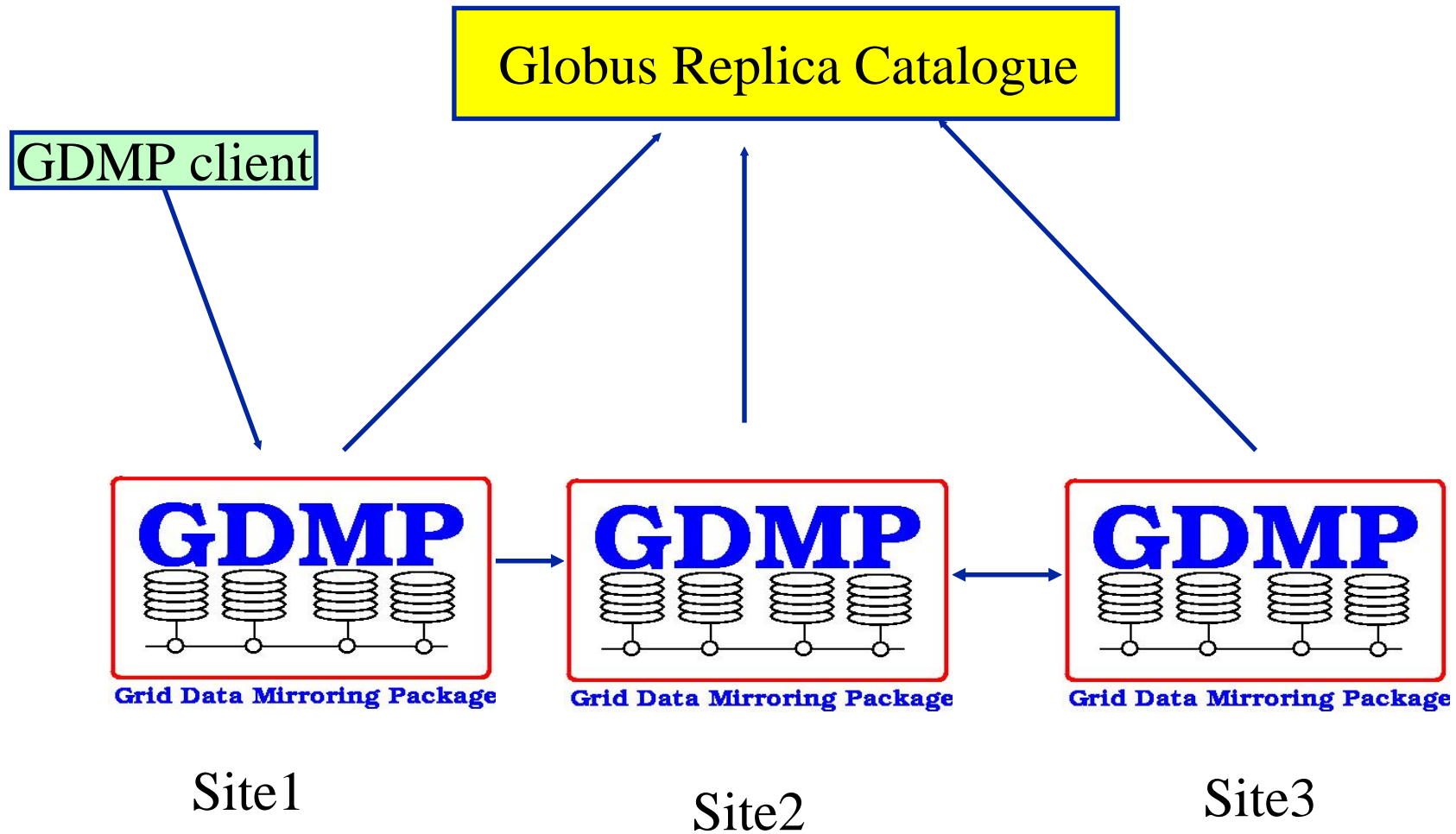


Grid Data Mirroring Package

- based on CMS requirements for replicating Objectivity files for High Level Trigger studies
- **production** prototype project for evaluating Grid technologies (especially Globus)
- experience will directly be used in DataGrid
 - input also for PPDG and GriPhyN
- <http://cern.ch/GDMP>

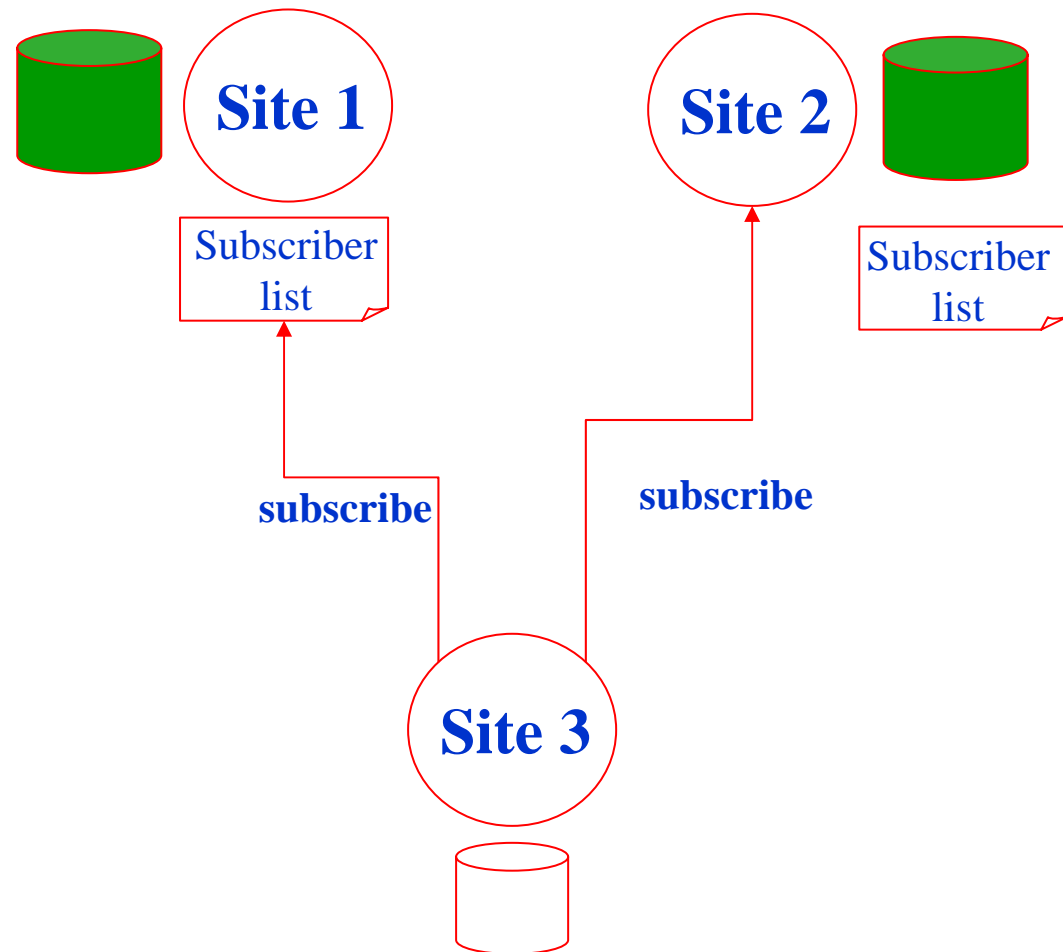


Overview of Components



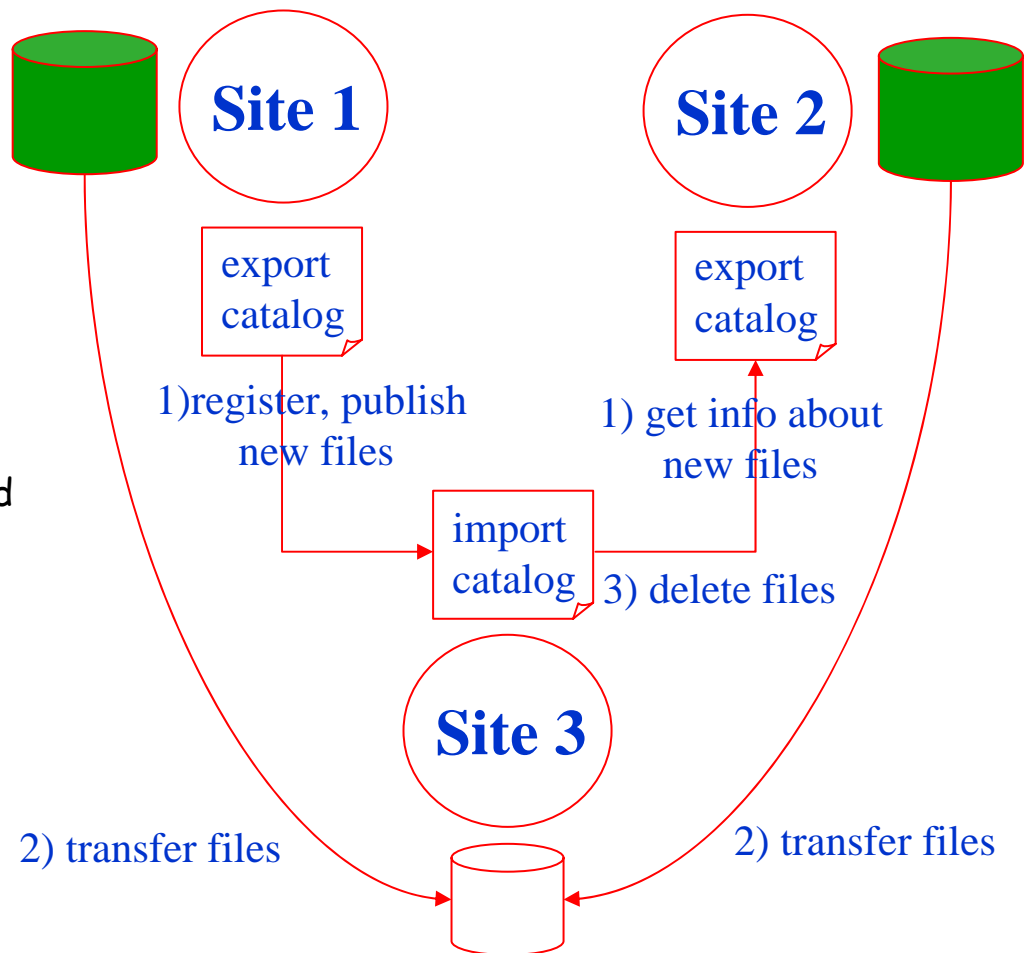
Subscription Model

- All the sites that **subscribe** to a particular site get **notified** whenever there is an update in its catalog.



Export / Import Catalogue

- Export Catalog
 - information about the new files produced .
 - is published
- Import Catalog
 - information about the files which have been published by other sites but not yet transferred locally
 - As soon as the file is transferred locally, it is removed from the import catalogue.
- Possible to pull the information about new files into your import catalogue.





Usage

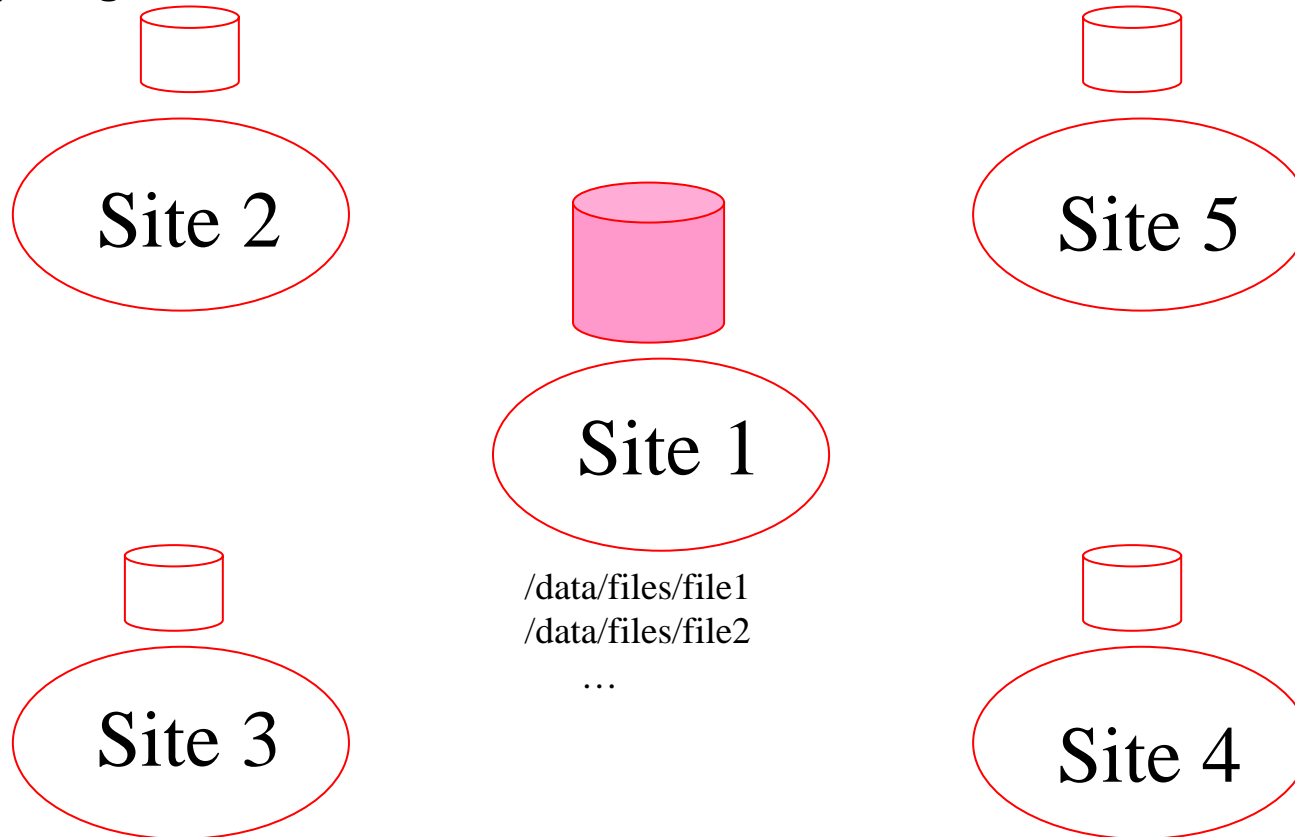
- **gdmp_ping**
 - Ping a GDMP server and get its status
- **gdmp_host_subscribe**
 - first thing to be done by a site
- **gdmp_register_local_file**
 - Registers a file in local file catalogue but NOT in Replica Catalogue (RC)
- **gdmp_publish_catalogue**
 - send **information** of newly created files to subscribed hosts (**no** real data transfer) - update RC
- **gdmp_replicate_get** - **gdmp_replicate_put**
 - get/put all the files from the import catalogue - update RC
- **gdmp_remove_local_file**
 - Delete a local file and update RC
- **gdmp_get_catalogue**
 - Get remote catalogue contents - for error recovery



Using GDMP

Register all files in a directory at site 1

• `gdmp_register_local_file -d /data/files`

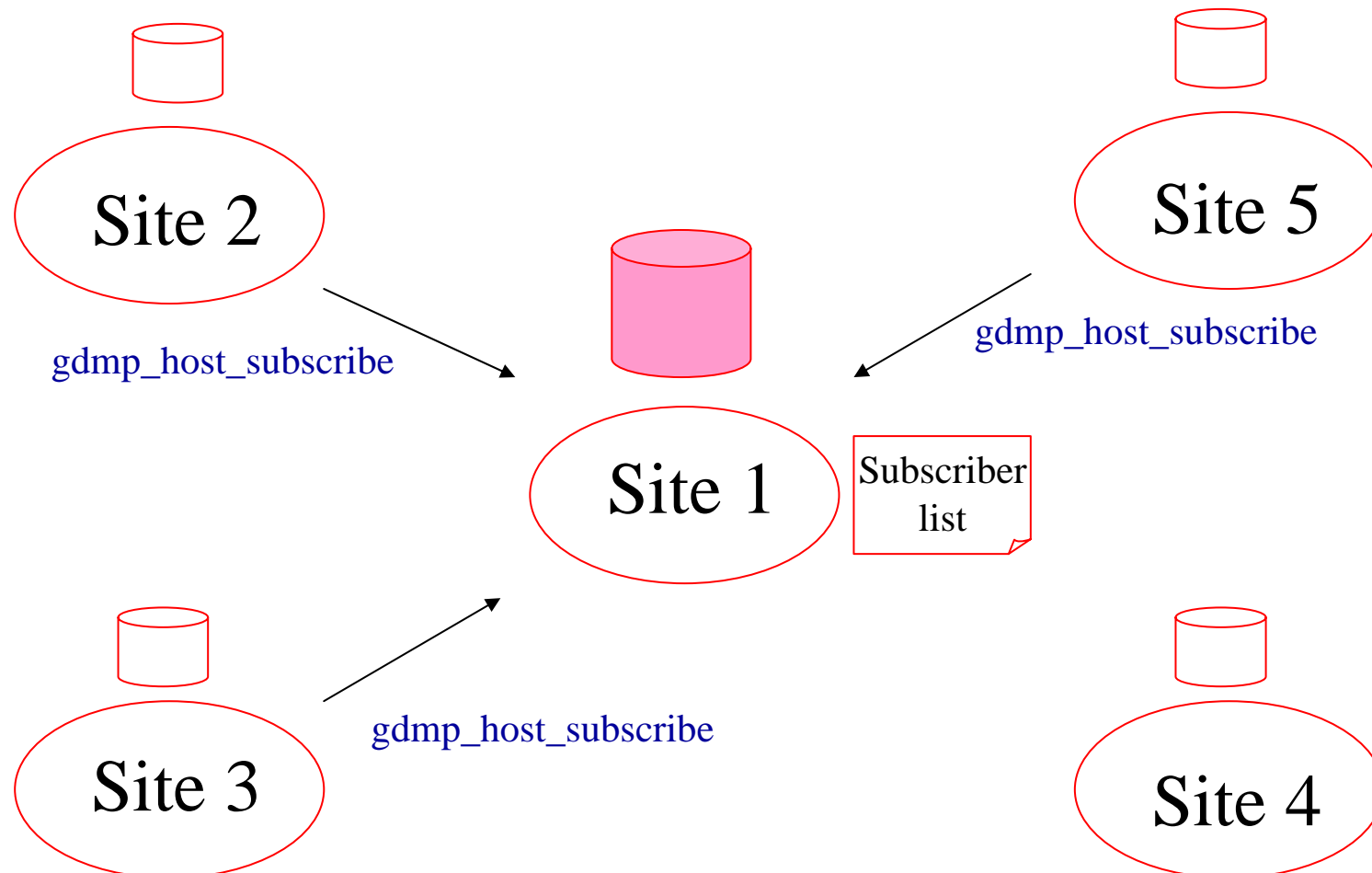


Data produced at site 1 to be replicated to other sites



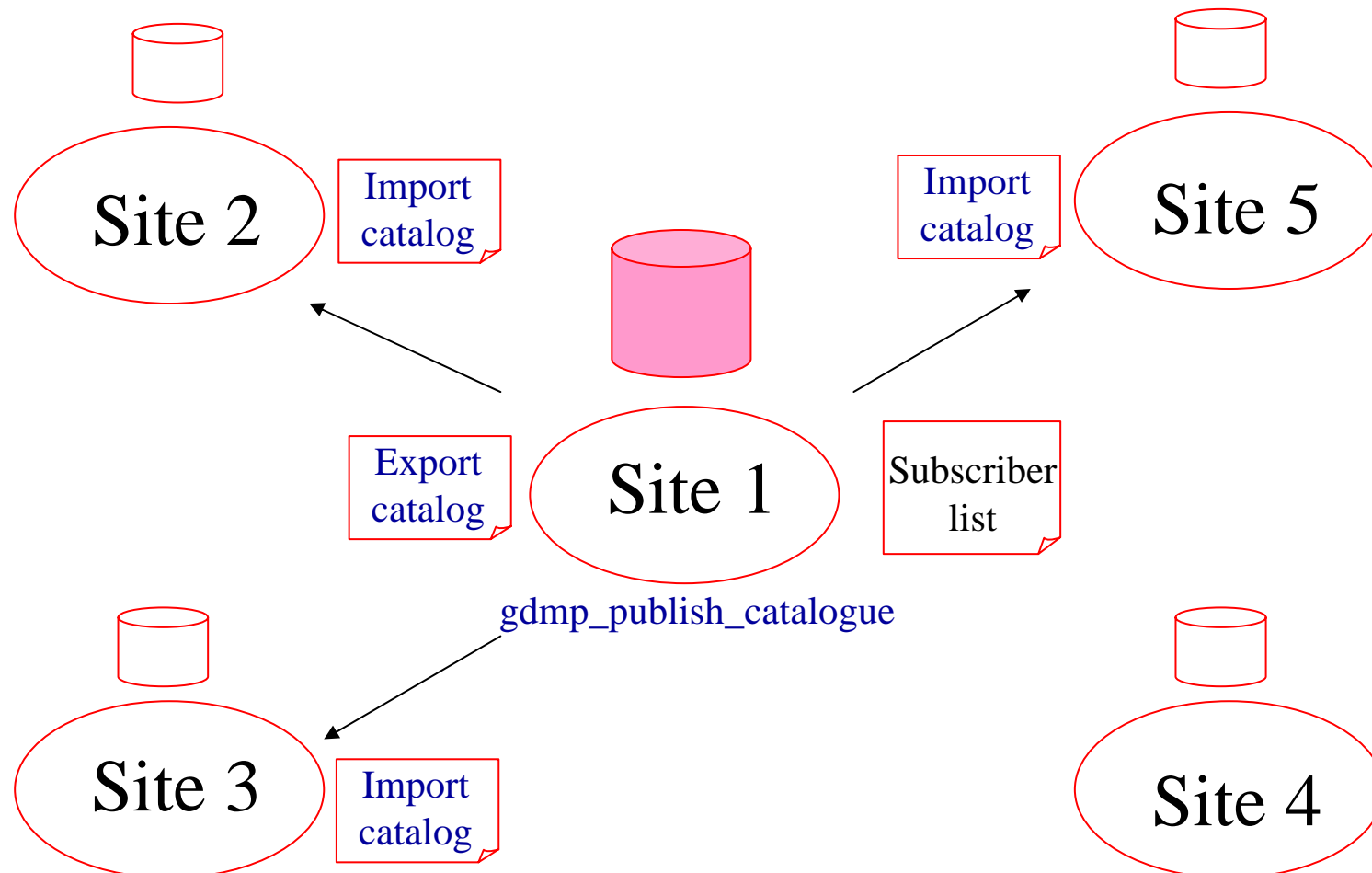
Using GDMP 2

- Start with subscription
 - `gdmp_host_subscribe -r <HOST> -p <PORT>`



Using GDMP 3

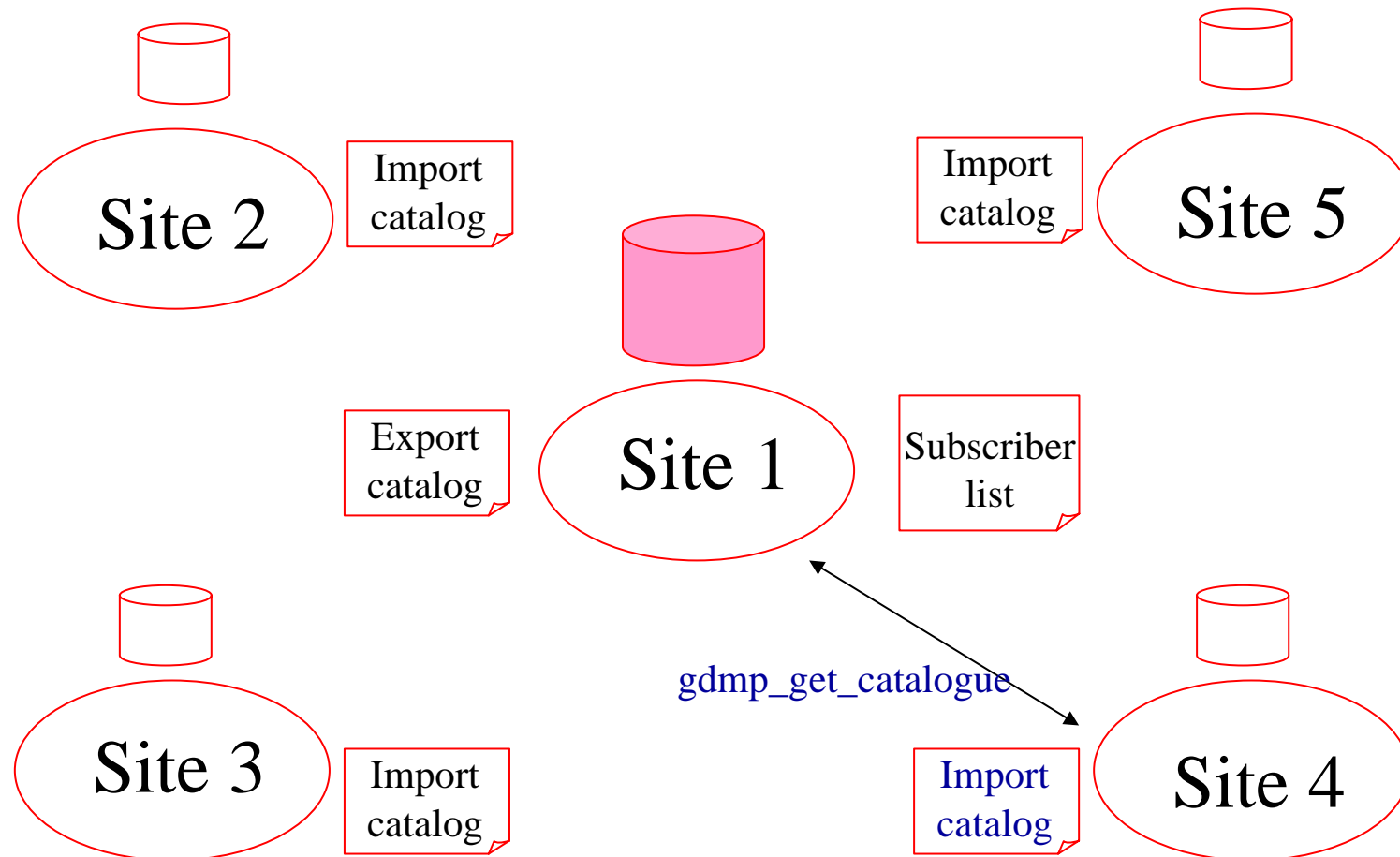
- Publish new files...can combine with filtering
 - `gdmp_publish_catalogue` (might use filter option)





Using GDMP 4

- Poll for change in catalog (pull model)...can combine with filtering...also used for error recovery.
 - `gdmp_get_catalogue -host <HOST>`

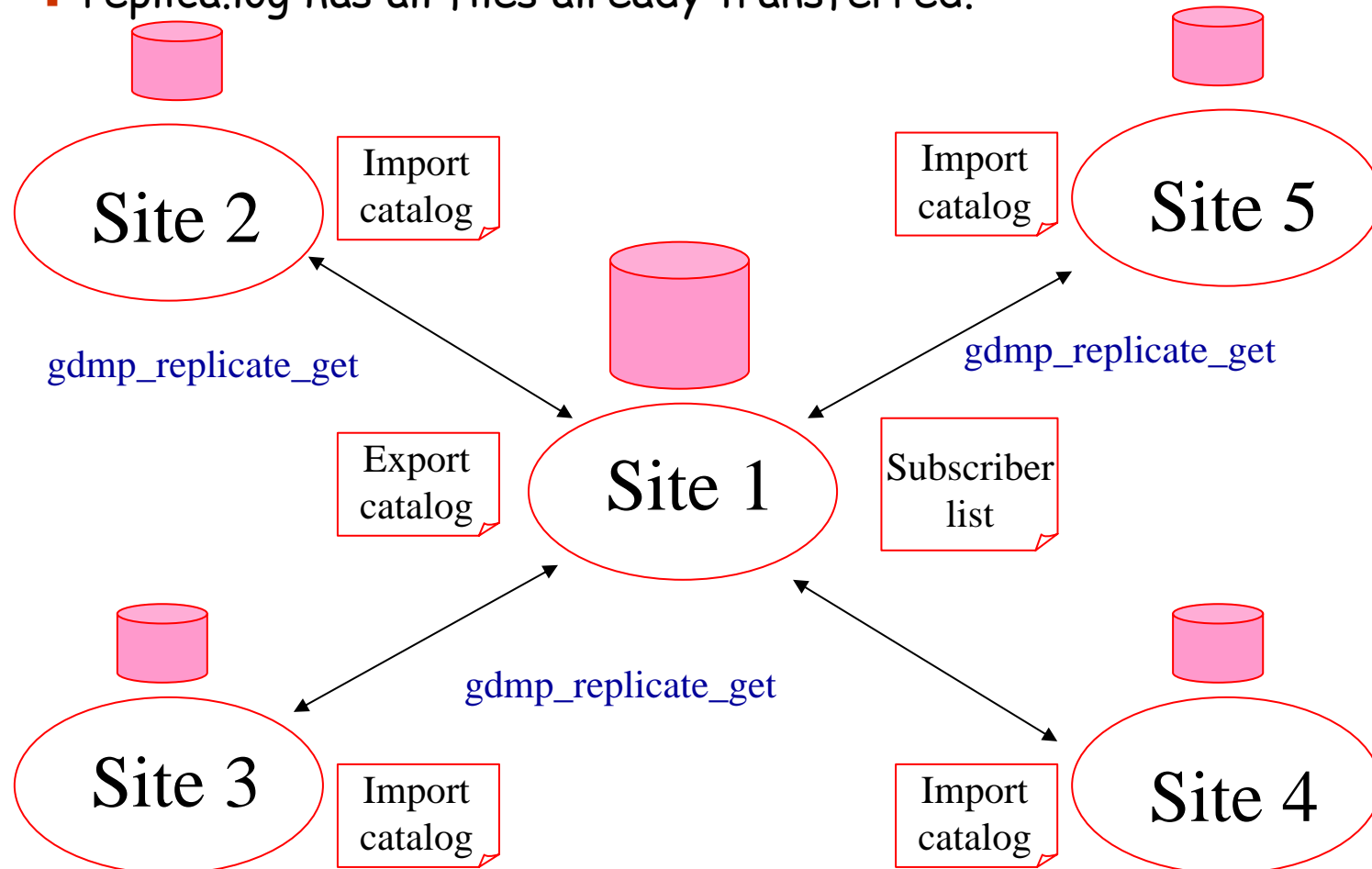




Using GDMP 5

➤ Transfer files...can use the progress meter

- `gdmp_replicate_get`
- `get_progress_meter`...produces a `progress.log`.
- `replica.log` has all files already transferred.





GDMP vs. EDG Replica Manager

➤ GDMP

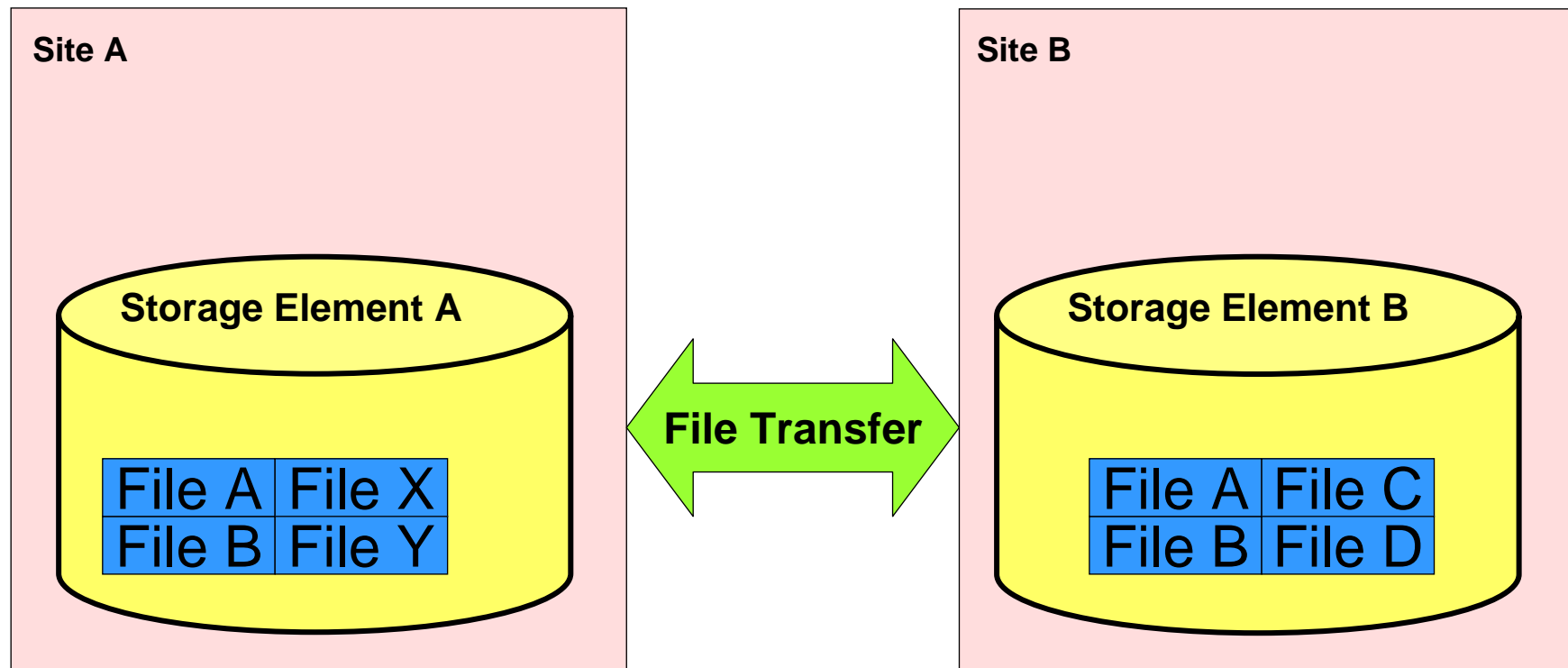
- Replicates sets of files
- Replication between SEs
- Mass storage interface
- File size as logical attribute
- Subscription model
- Event notification
- CRC file size check
- Support for Objectivity

➤ Replica Manager

- Replicates single files
- Replication between SEs, CEs to SE.



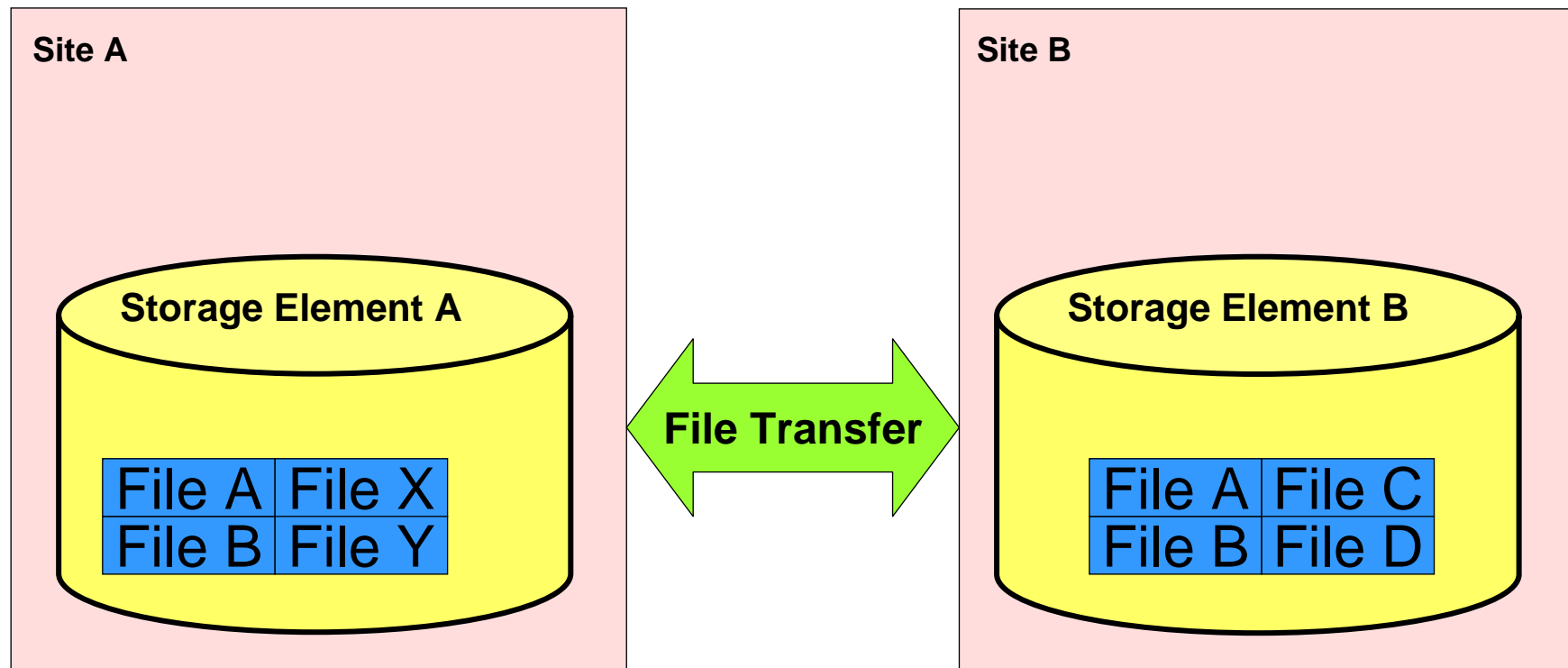
File Management Summary





File Management Summary

Replica Catalog:
Map Logical to Site files

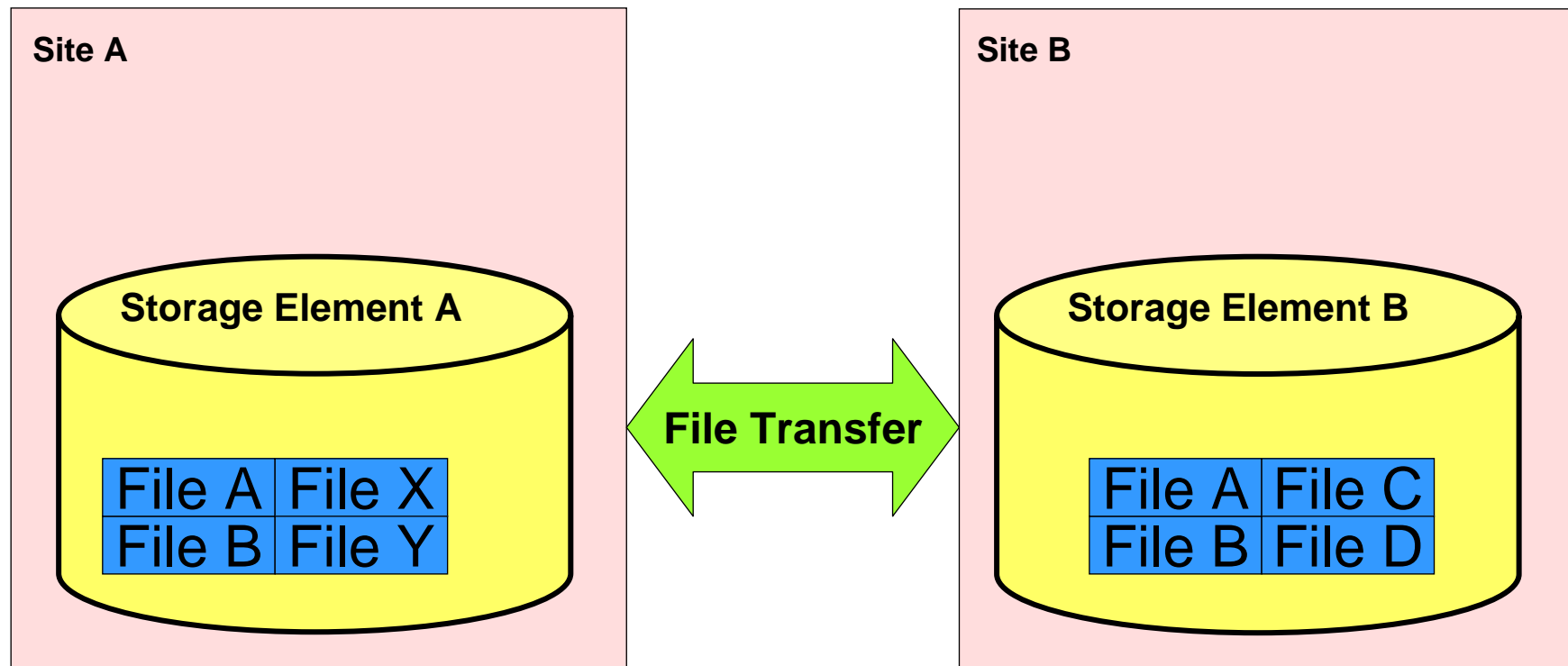




File Management Summary

Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file



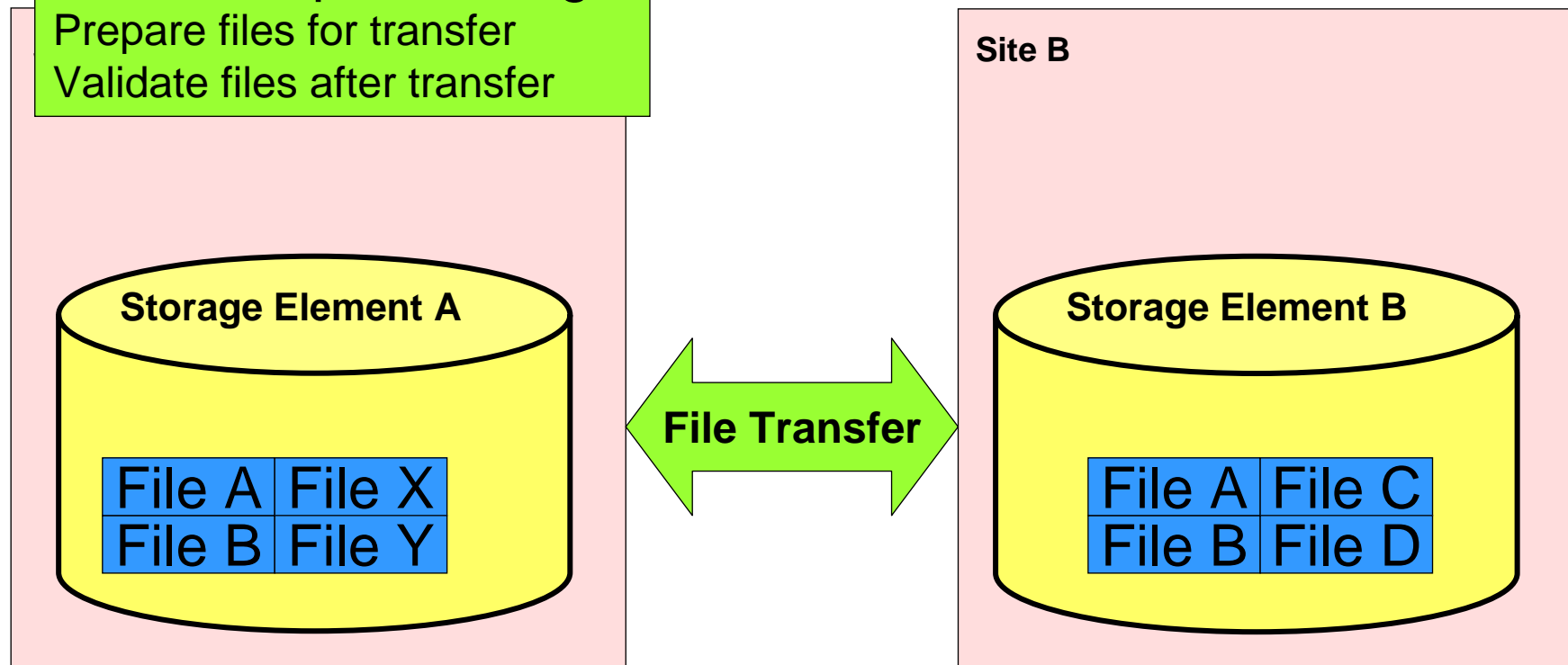


File Management Summary

Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer





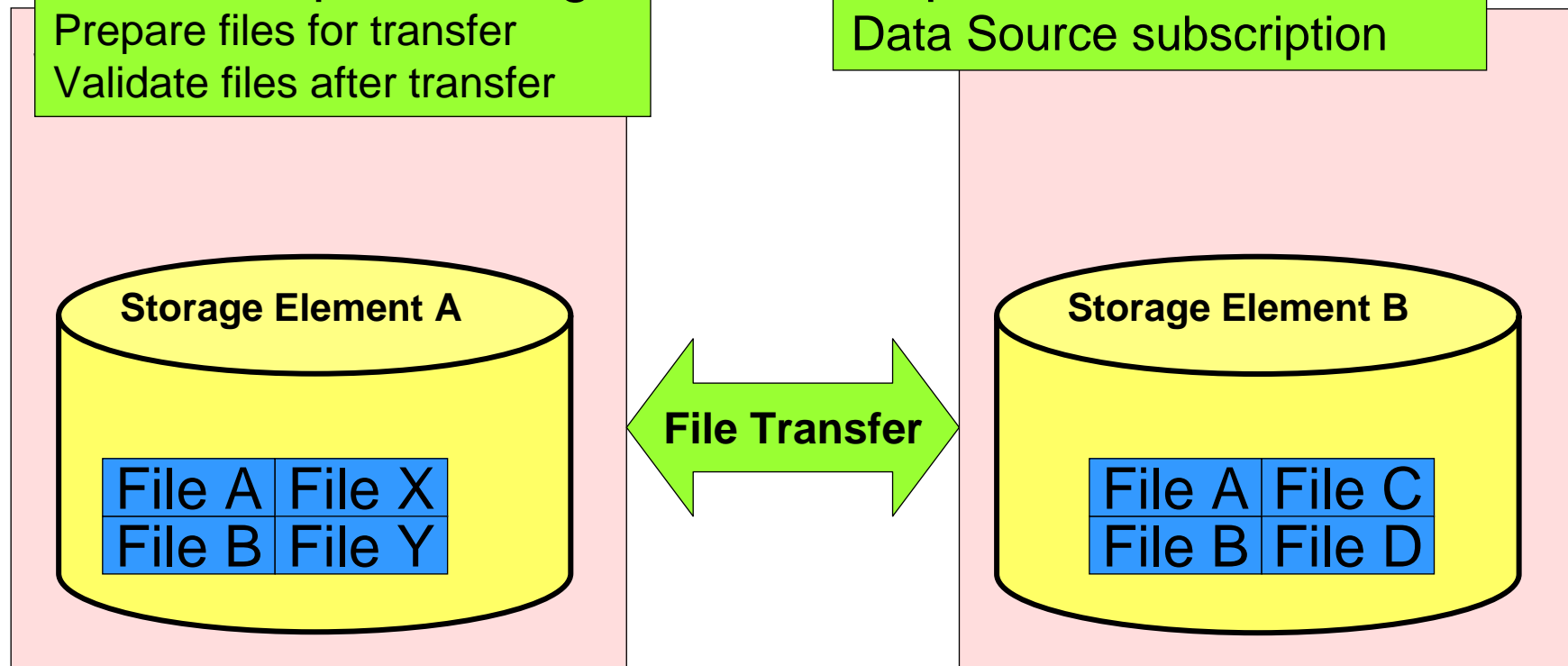
File Management Summary

Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer

Replication Automation:
Data Source subscription





File Management Summary

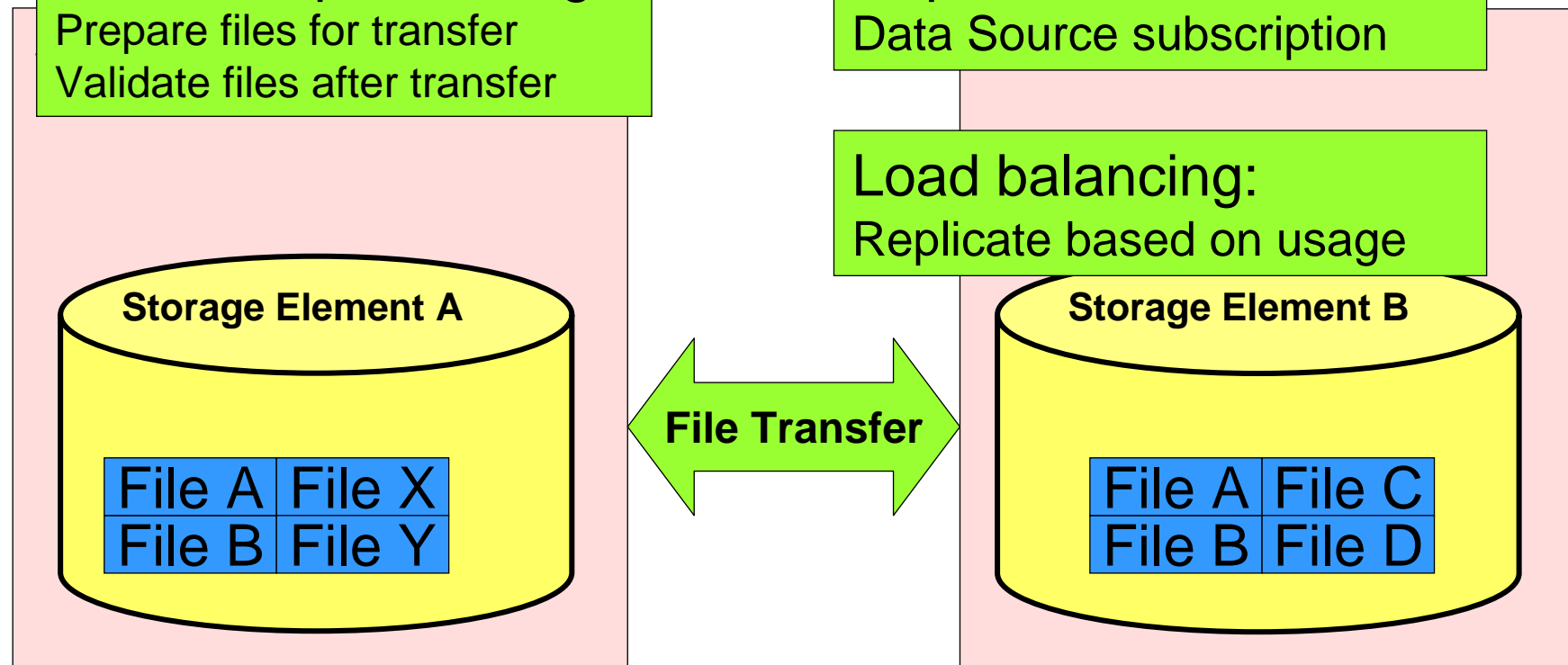
Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer

Replication Automation:
Data Source subscription

Load balancing:
Replicate based on usage





Replica Manager:
'atomic' replication operation
single client interface
orchestrator

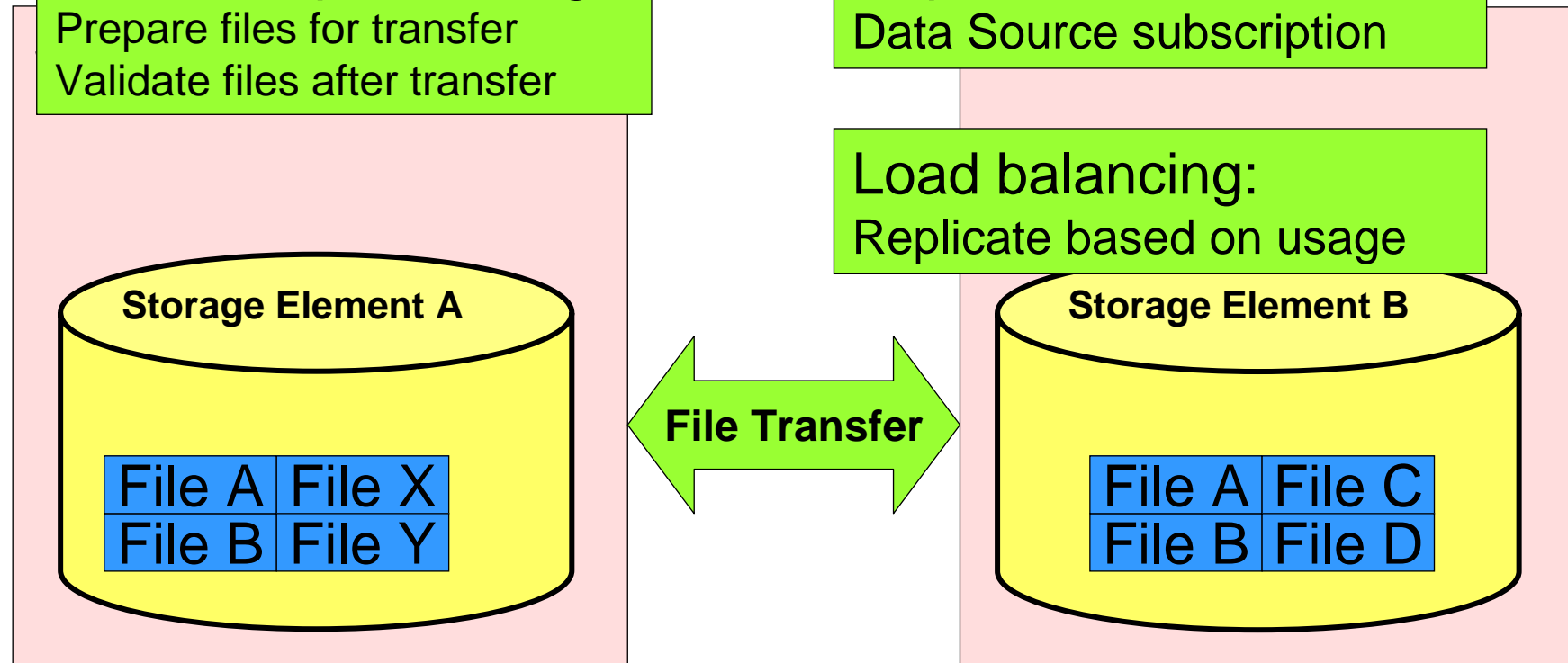
Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer

Replication Automation:
Data Source subscription

Load balancing:
Replicate based on usage





Replica Manager:
'atomic' replication operation
single client interface
orchestrator

Replica Catalog:
Map Logical to Site files

Replica Selection:
Get 'best' file

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer

Replication Automation:
Data Source subscription

Metadata:
LFN metadata
Transaction information
Access patterns

Load balancing:
Replicate based on usage

File A	File X
File B	File Y

File Transfer

Storage Element B

File A	File C
File B	File D



Replica Manager:
'atomic' replication operation
single client interface
orchestrator

Replica Catalog:
Map Logical to Site files

Replica Selection:
Get best replica

Pre- Post-processing:
Prepare files for transfer
Validate files after transfer

Replication Automation:
Data Source subscription

Load balancing:
Replicate based on usage

Metadata:
LFN metadata
Transaction information
Access patterns

File A	File X
File B	File Y

File Transfer

Storage Element B

File A	File C
File B	File D