dCache SRM

Timur Perelmutov
Fermilab

SRM Collaboration

Jefferson Lab

Bryan Hess
Andy Kowalski
Chip Watson

Fermilab

Don Petravick
Timur Perelmutov

LBNL

Arie Shoshani
Alex Sim
Junmin Gu

EU DataGrid WP2

Peter Kunszt
Heinz Stockinger
Kurt Stockinger
Erwin Laure

EU DataGrid WP5

Jean-Philippe Baud
Stefano Occhetti
Jens Jensen
Emil Knezo
Owen Synge

SRM Motivation (1)

- Grid Architecture promises Reservation and Scheduling of the Following Shared Resources
  - Computing Resources
  - Network Resources
  - Storage Resources (often neglected)
- SRM provides Reservation and Scheduling of the Storage Resources

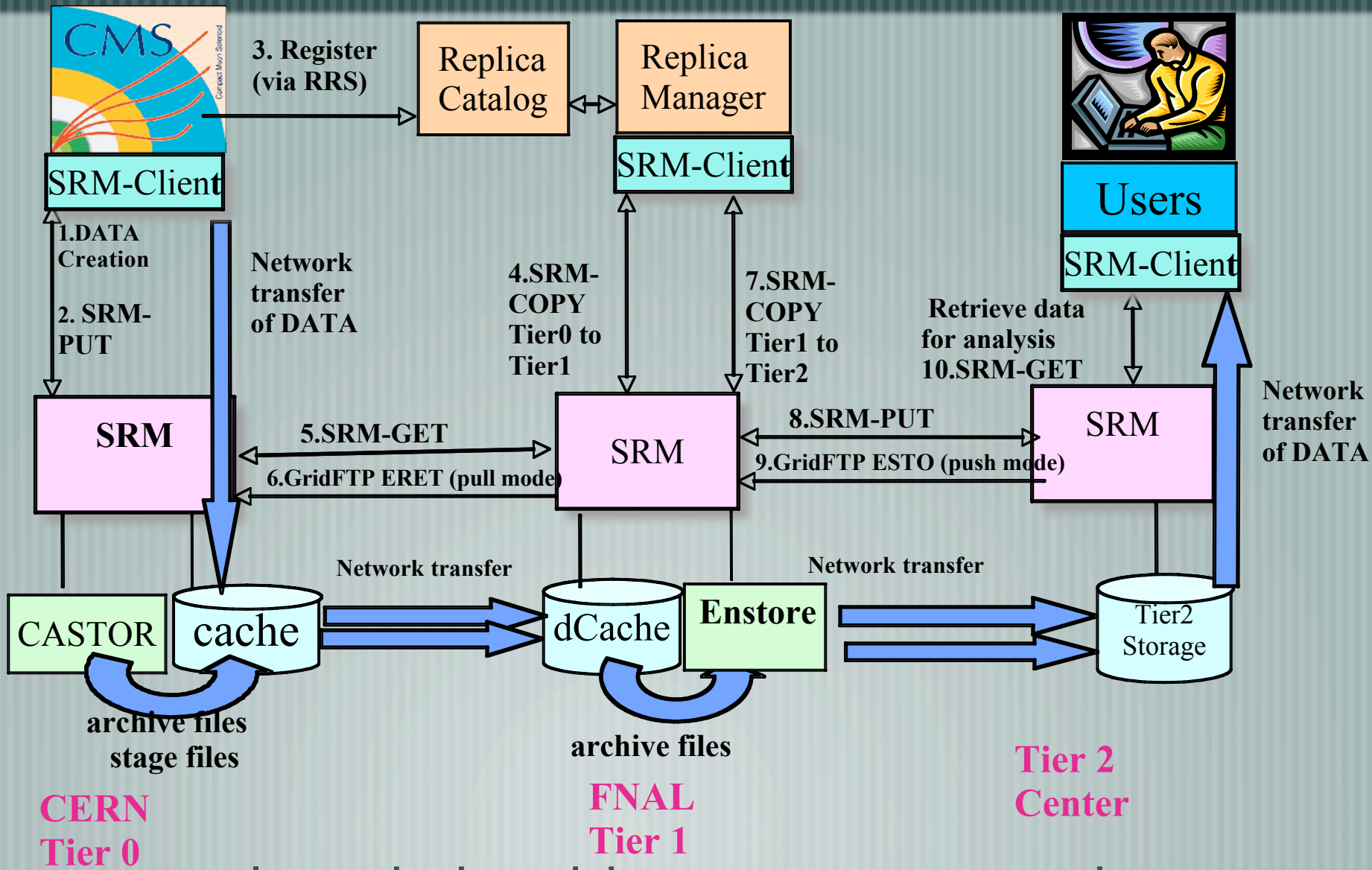
SRM Motivation (2)

- High Energy Physics Collaborations span multiple institutions where
  - A large variety of types of Storage Systems exist
    - Single Disk / Raid
    - Robotic Tape Storage System (Enstore, HPSS)
    - Distributed Disk Cache (dCache)
    - Hierarchical Storage System (dCache - Enstore)
  - Heterogeneous environments and proliferation of custom Mass Storage Systems (MSSs) exist
  - User applications often need to access data at multiple institutions on multiple MSSs
  - SRM provides Standardized Uniform Access to Heterogeneous Storage

Storage Resource Managers

- SRMs are middleware components that manage shared storage resources on the Grid and provide:
  - Uniform access to heterogeneous storage
  - File Transfer Protocol negotiation
  - Dynamic Transfer URL allocation
  - Access to permanent and temporary types of storage
  - Advanced space and file reservation

Data life cycle on the grid and a role of SRM on the GRID



Storage Resource Manager versions

- Two SRM Interface specifications
  - SRM v1.1provides
    - Data access/transfer
    - Implicit space reservation
  - SRM v2.1 adds
    - Explicit space reservation
    - Namespace discovery and manipulation
    - Access permissions manipulation
- Fermilab SRM implements SRM v1.1 specification

SRM Protocols and Groups of Functions

- SRM interface consists of the following groups of functions:
  - Space Management Functions – v2.1
  - Data Transfer Functions – v1.1 and v2.1
  - Directory Functions – v2.1
  - Permission Functions – v2.1
  - Status Functions – v1.1 and v2.1

SRM V1 Interface Details

Data Transfer Functions:

Get

Put

Copy

getRequestStatus

getFileMetaData

setFileStatus

Remaining Functions:

Pin

Unpin

getProtocols

getEstGetTime

getEstPutTime

AdvisoryDelete

SRM V2 Interface Details

Space Management Functions

SrmReserveSpace
SrmReleaseSpace
srmUpdateSpace
srmCompactSpace
srmGetSpaceMetaData
srmChangeFileStorageType
srmGetSpaceToken

Directory Functions

SrmMkdir
srmRmdir
srmRm
srmLs
srmMv

Data transfer functions

srmPrepareToGet
SrmPrepareToPut
srmCopy
SrmRemoveFiles
srmReleaseFiles
srmPutDone
srmAbortRequest
srmAbortFiles
srmSuspendRequest
srmResumeRequest

Status Functions

srmStatusOfGetRequest
srmStatusOfPutRequest
srmStatusOfCopyRequest
srmGetRequestSummary
srmExtendFileLifeTime
SrmGetRequestID

Permission

srmSetPermission
srmReassignToUser
srmCheckPermission

Srmcp Client Server Negotiation(1)

- Srmcp reads/writes a file(s)

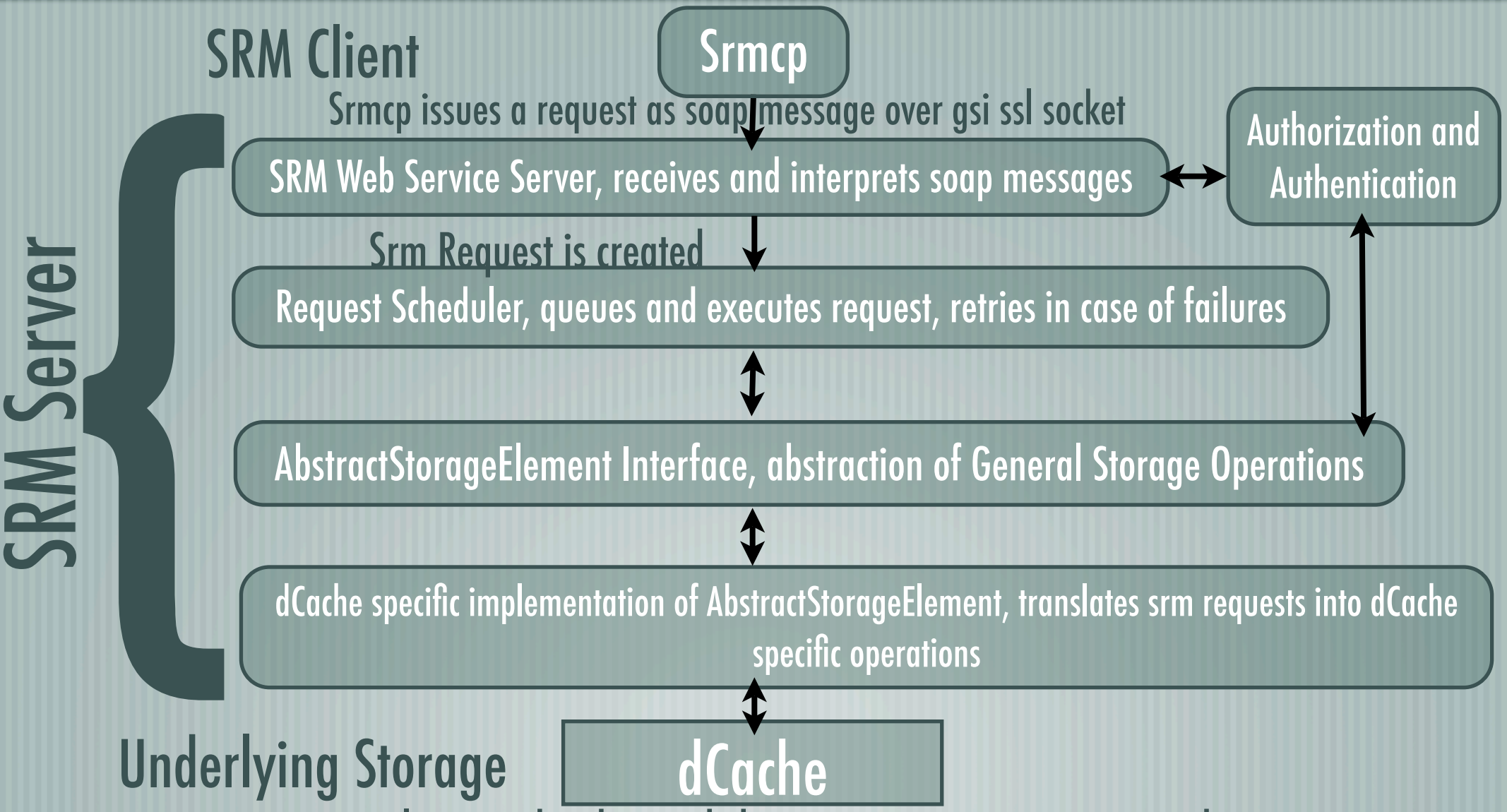
1. srmcp issues get/put, gets request id back
2. while request status is “Pending”, update request status
3. once status is ready and TURL(s) is available perform transfer from/to TURL(s)
4. once transfer completes, set file status to “Done”

Srmcp Client Server Negotiation(2)

- Srmcp copies a file from one SRM server to another

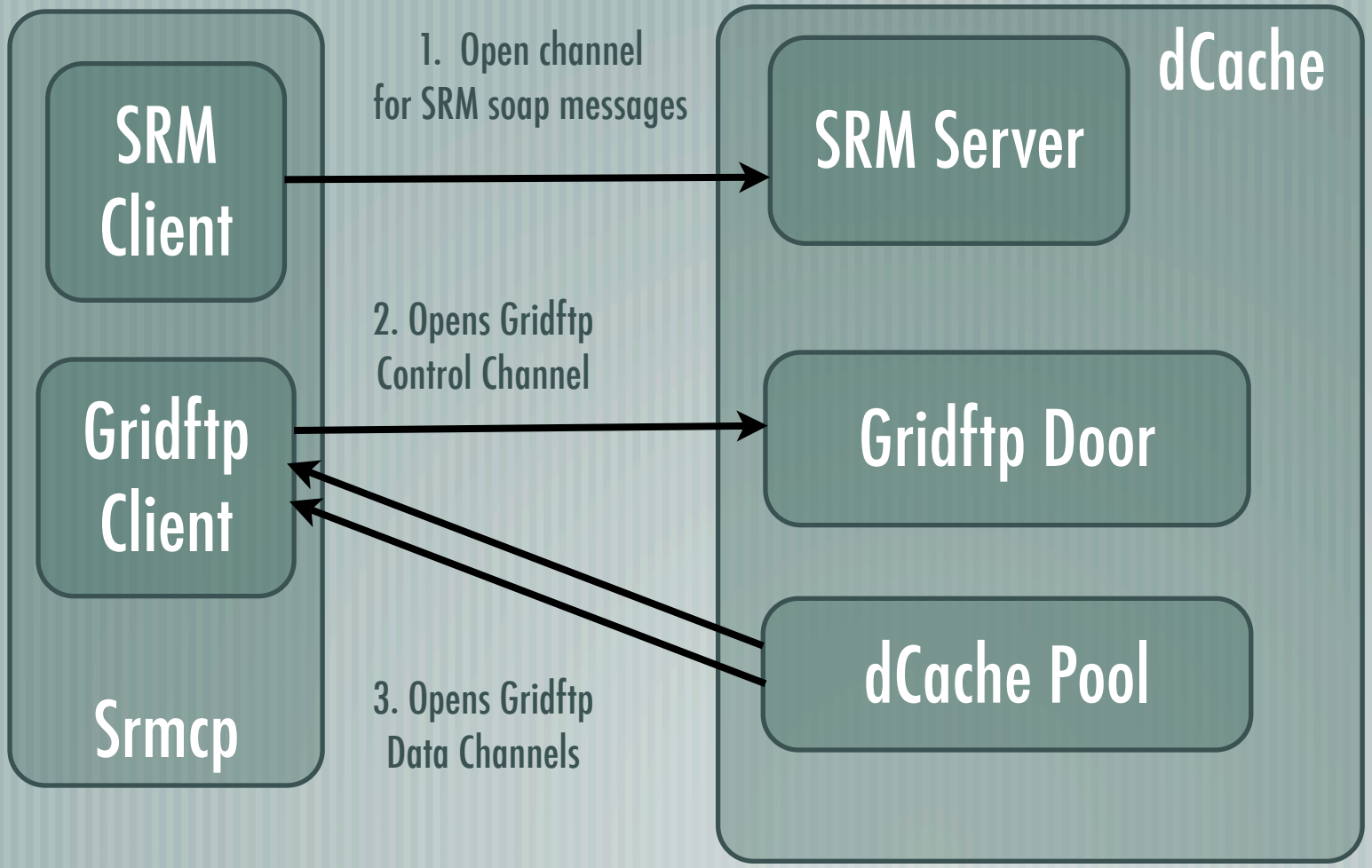
1. srmcp issues copy, gets request id back
2. while request status is “Pending”, update request status
3. once status is “Done” transfer has completed, report result and exit.

SRM to dCache communication



SRM Get Network Flows

srmcp srm://dCache:8443/dir1/file1 file:////tmp/file1

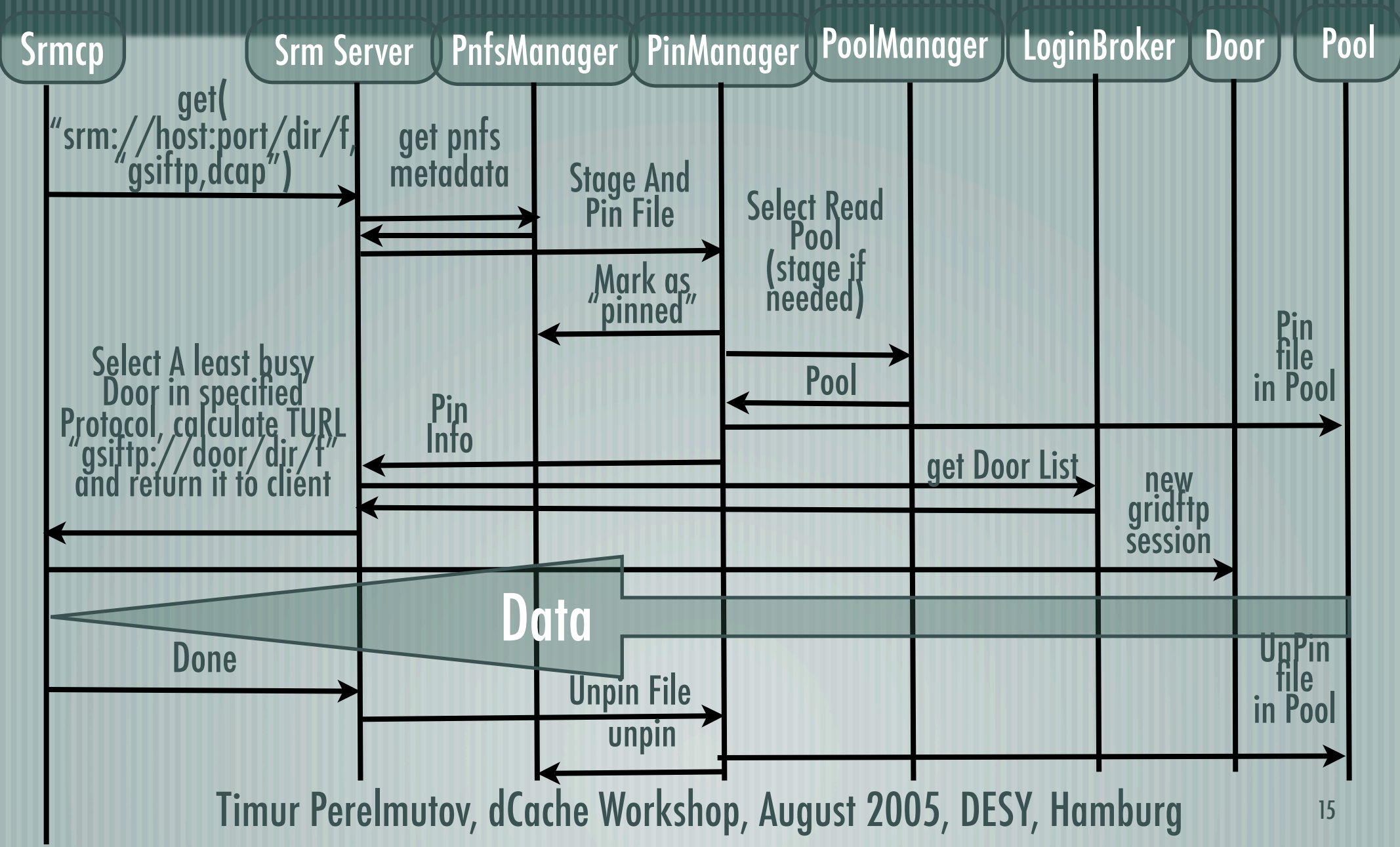


Firewalls:

- Srmcp Client port range needs to be open and configured when behind a Firewall
- Pools can be behind firewall
- Srm server port must be open on srm node
- Gridftp server port must open on Gridftp door

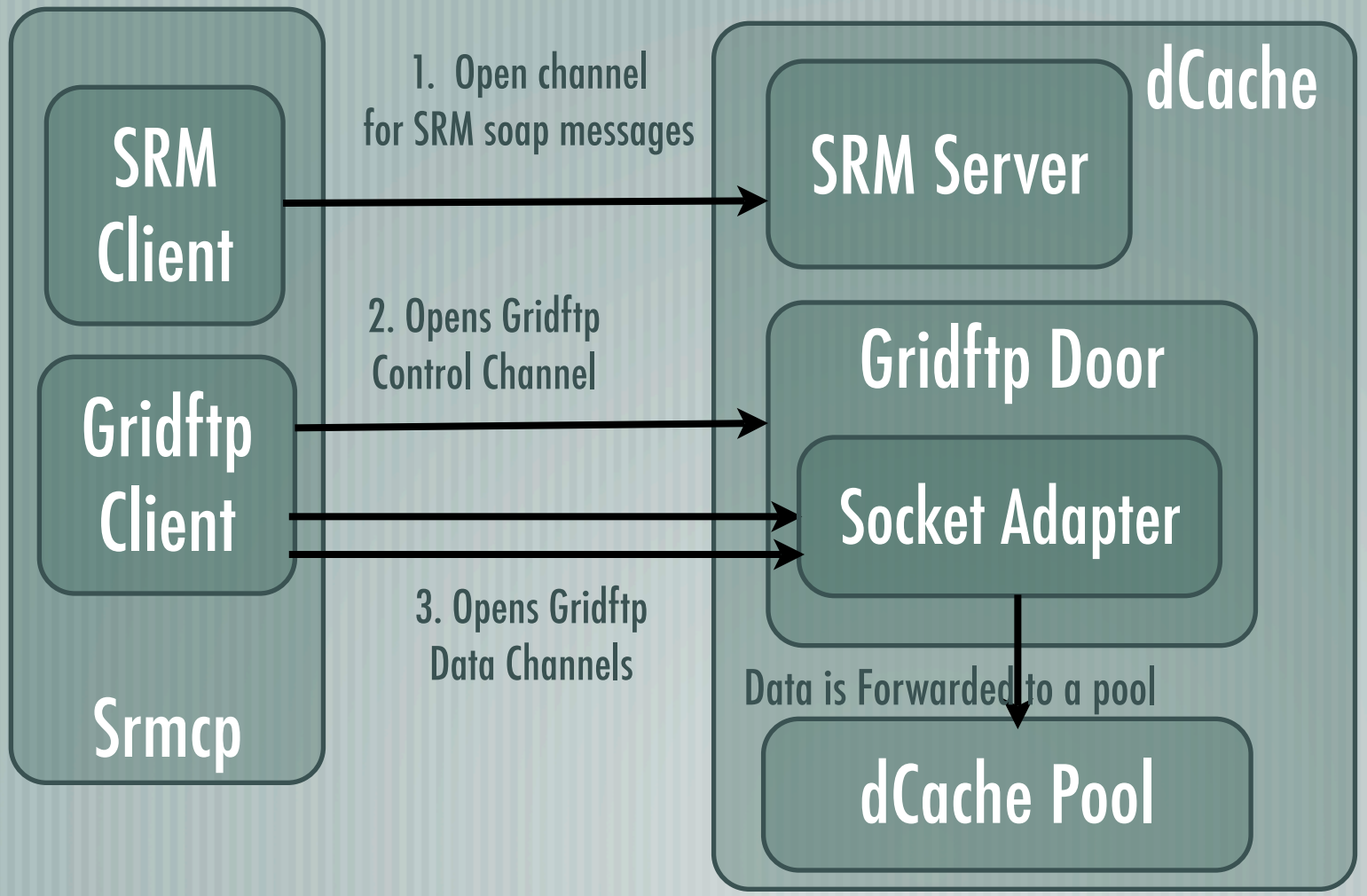
Configuration details at http://www.dcache.org/manuals/experts_docs/firewall.html

SRM Get Details



SRM Put Network Flows

srmcp file:////tmp/file1 srm://dCache:8443/dir1/file1

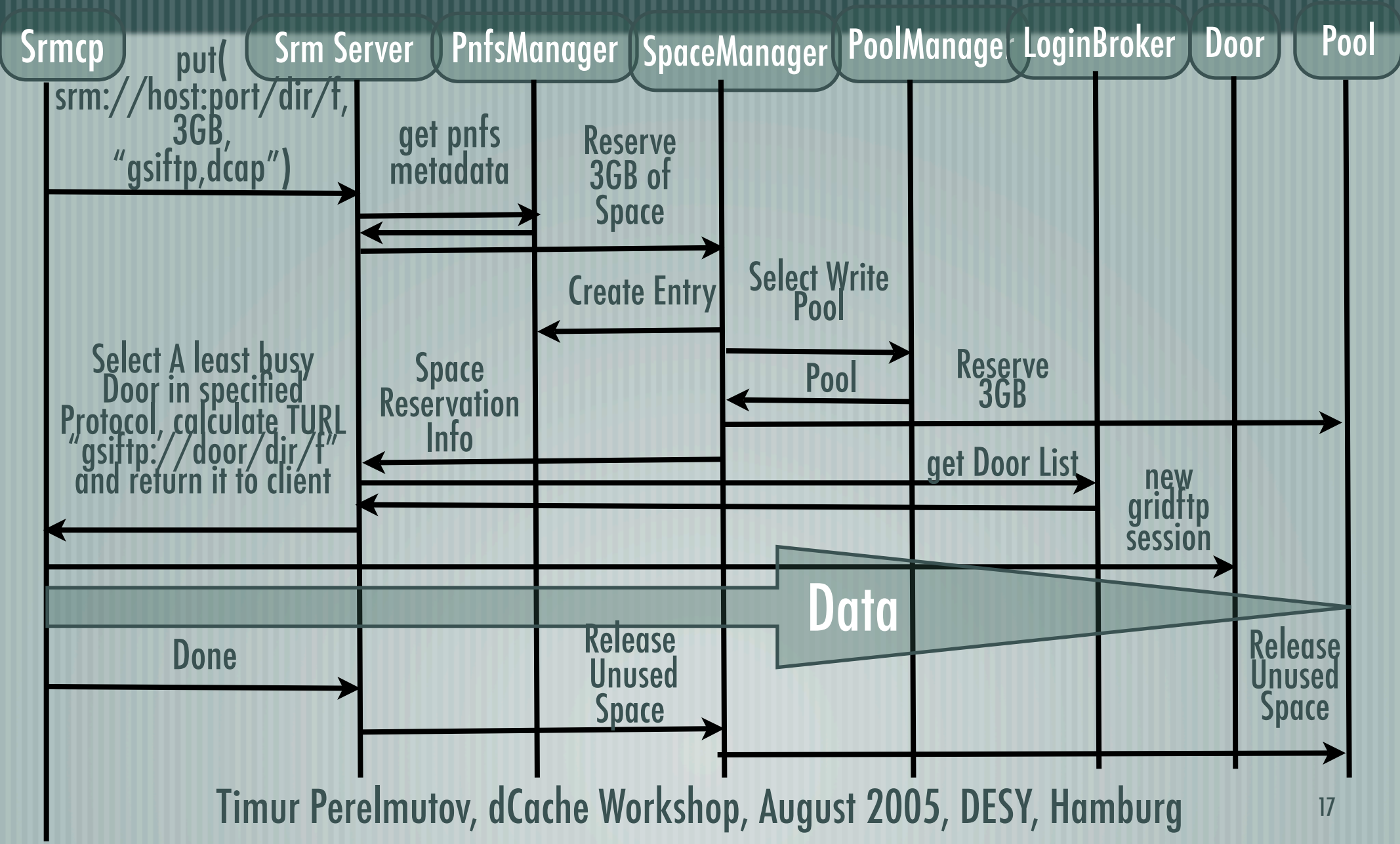


Firewalls:

- Srmcp Client can be behind a Firewall
- Pools can be behind firewall
- Srm server port must be open on srm node
- Gridftp port and port range for data must be configured on Gridftp door

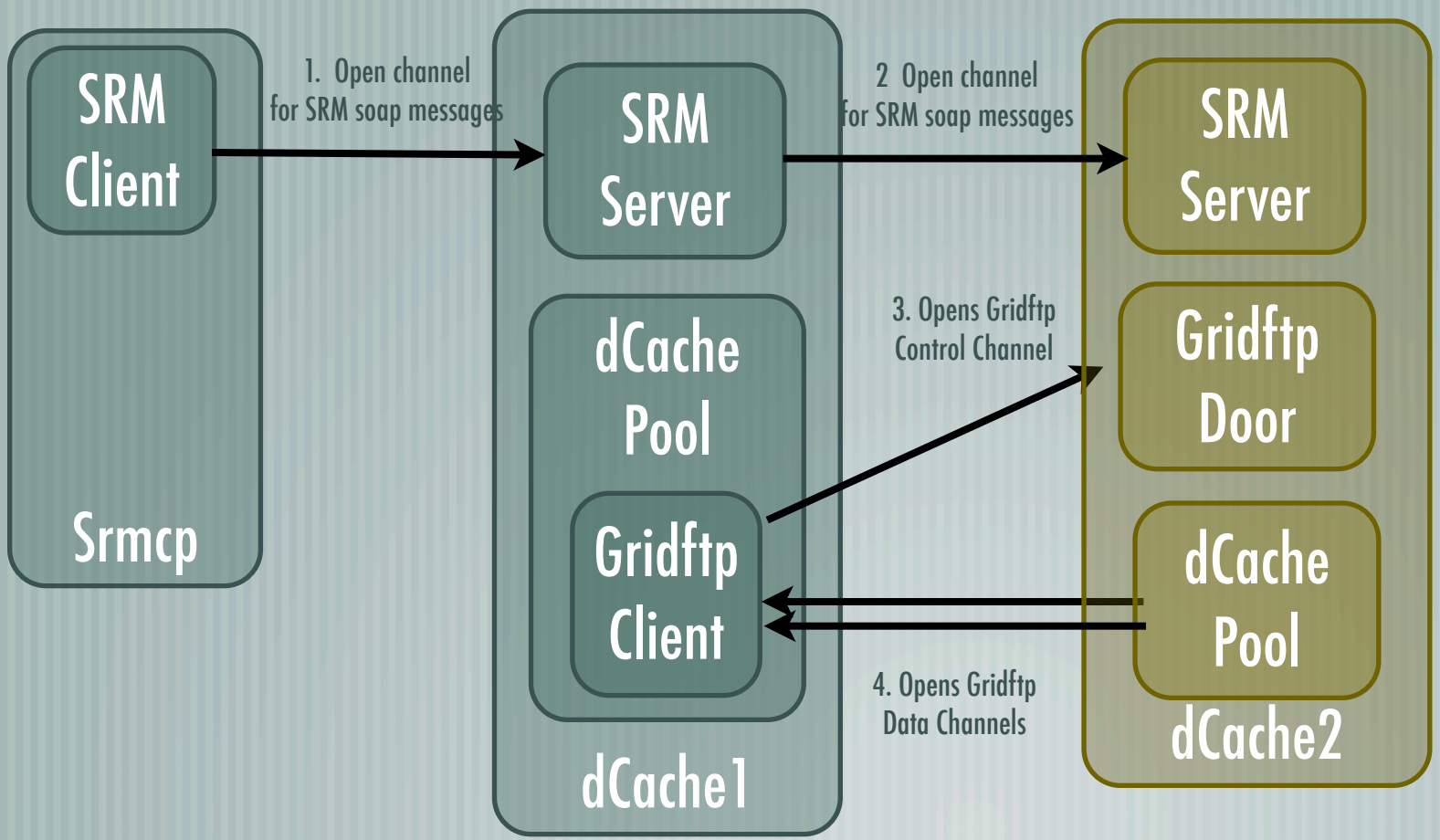
Configuration details at http://www.dcache.org/manuals/experts_docs/firewall.html

SRM Put Details



SRM Copy in Pull Mode Network Flows

srmcp srm://dCache2:8443/dir1/file1
srm://dCache1:8443/dir1/file1.copy

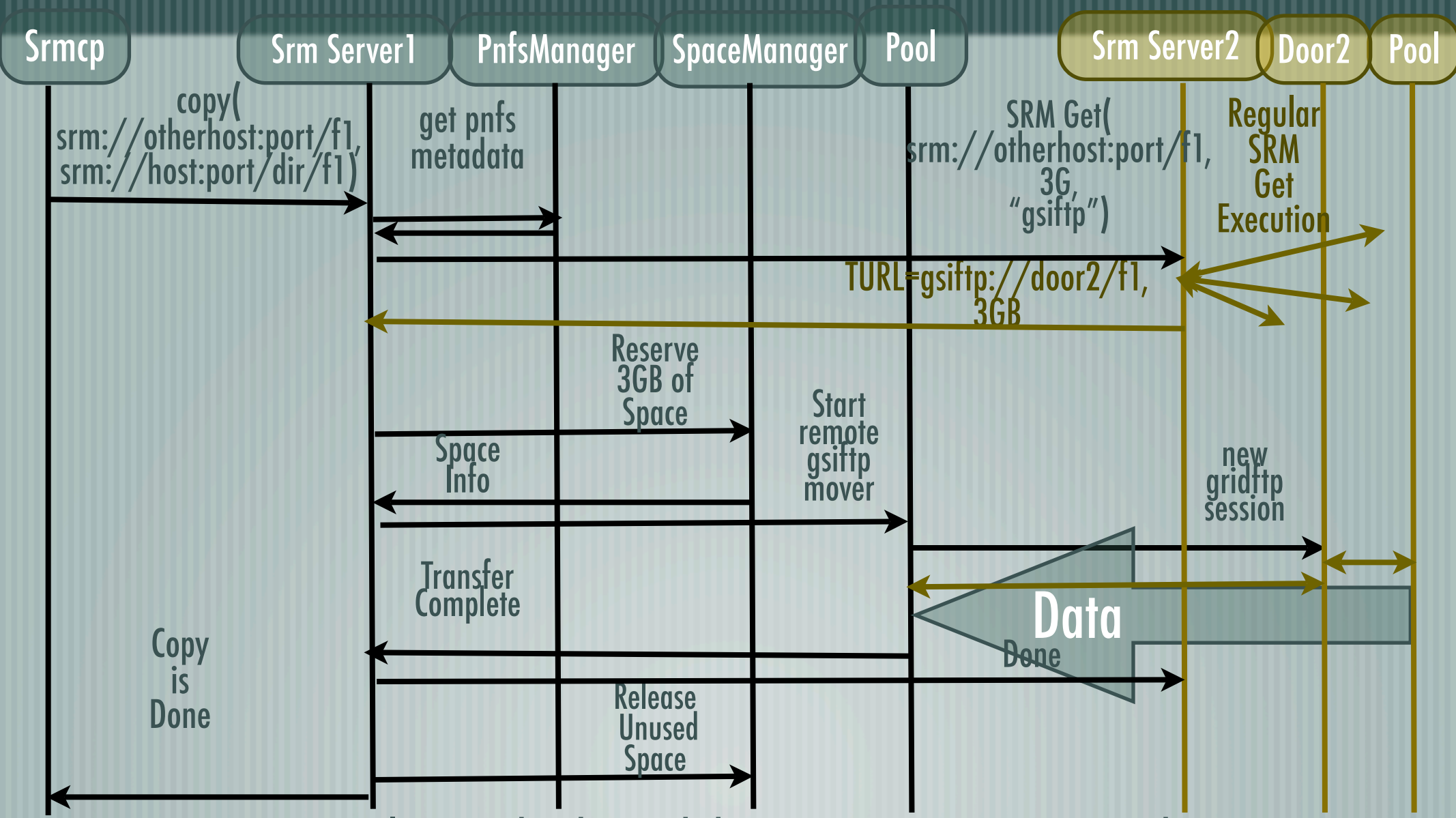


Firewalls:

- Srmcp Client can be behind a Firewall
- dCache1 Pools need a port range set up and configured
- Srm servers ports must be open on srm nodes
- dCache2 Gridftp server port must be open on Gridftp door node

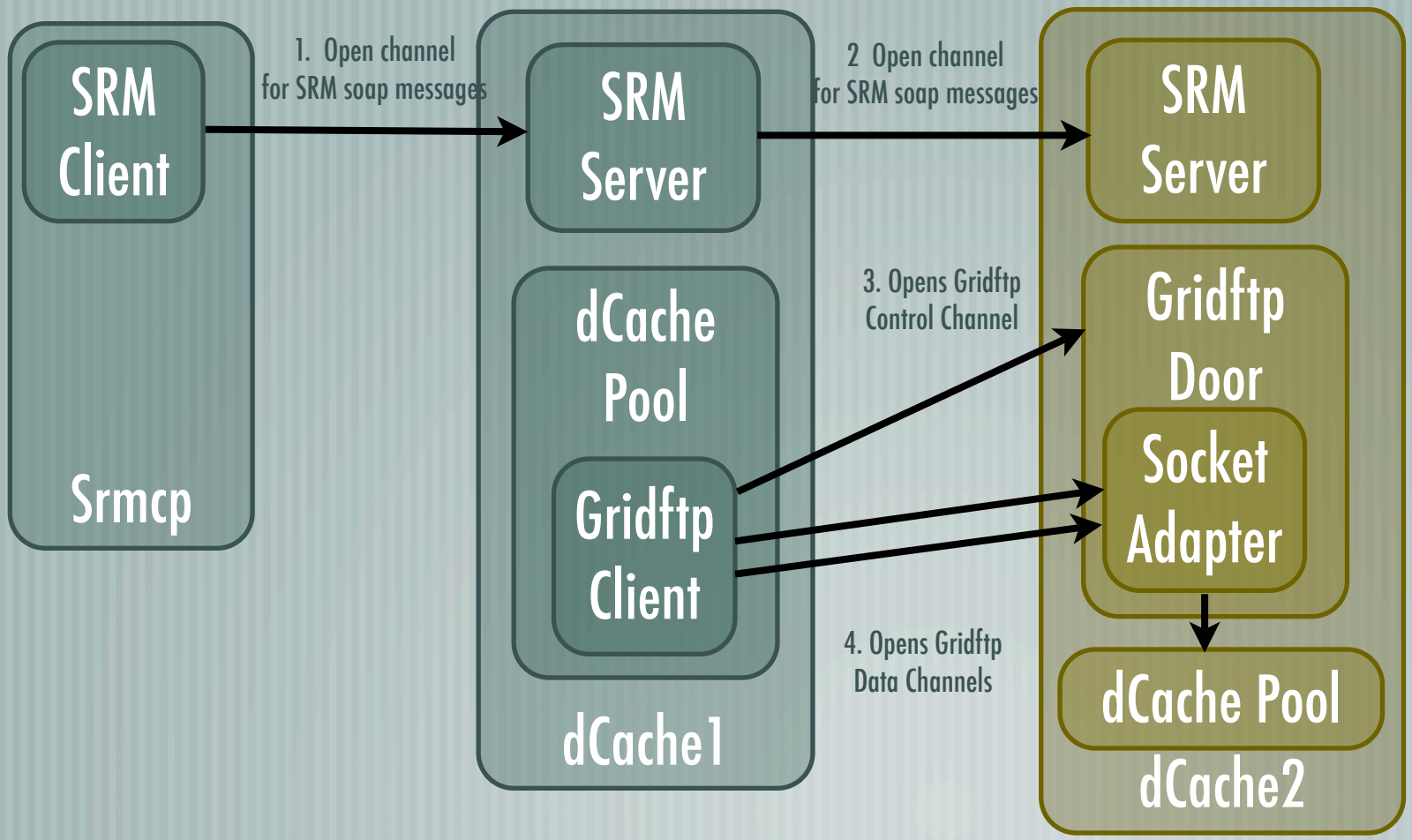
Configuration details at http://www.dcache.org/manuals/experts_docs/firewall.html

SRM Copy in pull mode details



SRM Copy in Push Mode Network Flows

srmcp -pushmode srm://dCache1:8443/dir1/file1
srm://dCache2:8443/dir1/file1.copy

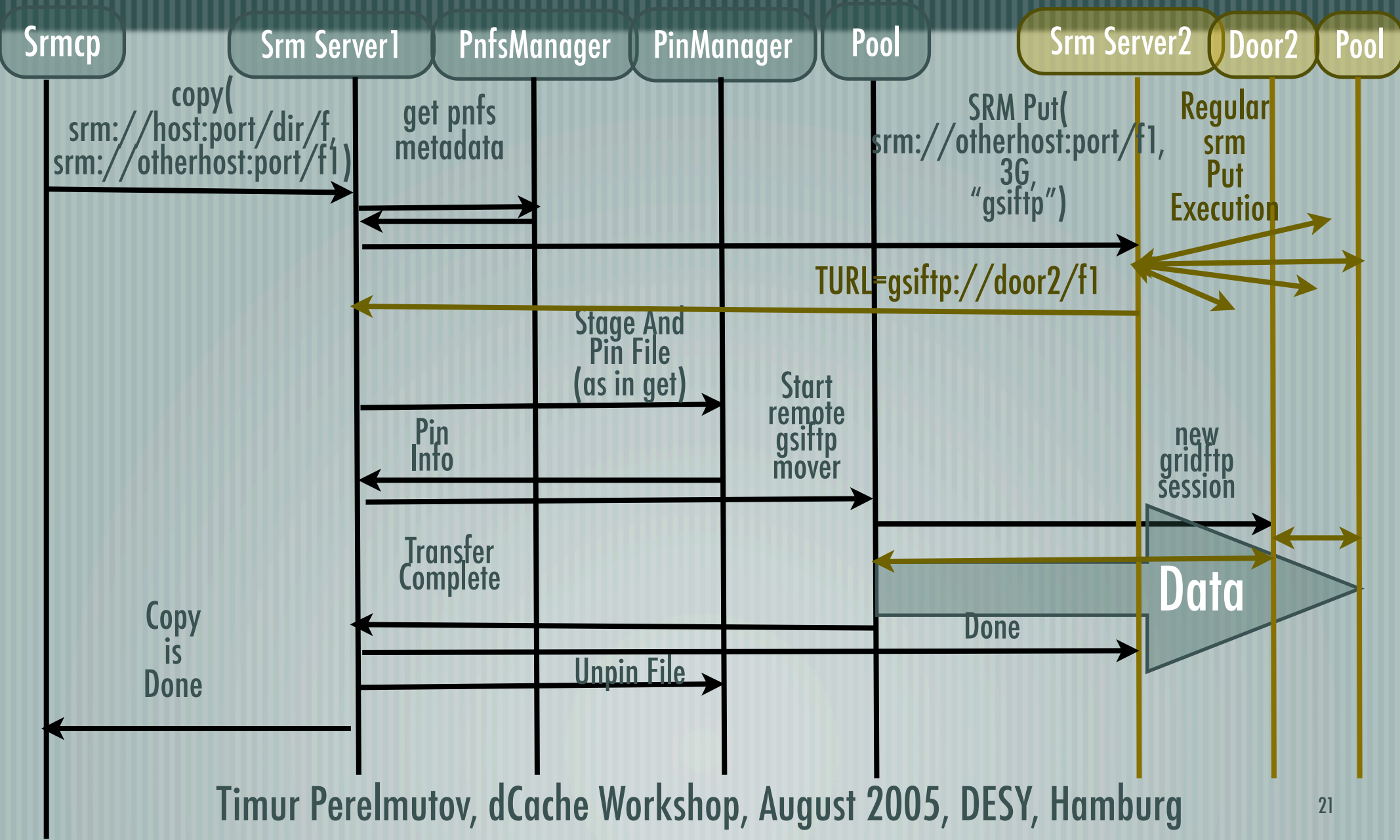


Firewalls:

- Srmcp Client can be behind a Firewall
- dCache1 Pools and dCache2 Pools can be behind firewals
- Srm servers ports must be open on srm nodes
- dCache2 Gridftp server must have a port range opne and configured port must be open on Gridftp door node

Configuration details at http://www.dcache.org/manuals/experts_docs/firewall.html

SRM Copy in push mode detailes



Status of Fermilab SRM Implementation

- SRM Interface to dCache Storage System
  - Data Transfer Functions (get, put and copy)
  - Load balancing, throttling, fairness
  - Scalable replication mechanism via gridftp
  - Automatic directory creation
- Fault tolerance and reliability achieved by providing persistent storage for transfer requests and retries on failures
- SRM interface as a standalone product, adaptable to work on top of another storage system through a SRM-Storage interface
- A reference implementation of the SRM-Storage interface to a Unix File System
- Implicit Space Management

Fermilab SRM Implementation Plans

- Full implementation of SRM Version 2.1 interface
  - Explicit Space Management
  - Support for at least Volatile and Permanent space types
  - Directory and Permission functions
- Research utilization of Lambda Station Interface by a Storage System. Lambda Station gives optical path allocation and per flow routing
- Monitoring, Administration and Accounting interfaces

Resources

- The Storage Resource Manager Collaboration, http://sdm.lbl.gov/srm-wg/
- Fermilab SRM Project , http://www-isd.fnal.gov/srm
- Patrick Fuhrmann, dCache, Grid Storage Element and enhanced use cases,http://indico.cern.ch/contributionDisplay.py?contribId=233&sessionId=10&confId=0
- DCache, Disk Cache Mass Storage System, http://www.dcache.org/
- US-CMS, http://www.uscms.org/
- Don Petravick, Lambda Station Proposal, http://hppc.fnal.gov/wawg/omnibus-text.pdf
- Arie Shoshany, Replica Registration Service,http://www.ppdg.net/mtgs/28jun04-wb/slides/PPDG-AH-0406-RMS-RRS.ppt
- Michael Ernst, US-CMS Grid File Access Proposal,http://www.uscms.org/sandc/reviews/doe-nsf/2003-07/docs/GFA-Proposal-Short-v1.0.pdf
- Michael Ernst, Managed Data Storage and Data Access Services for Data Grids,http://indico.cern.ch/contributionDisplay.py?contribId=190&sessionId=7&confId=0
- Philip DeMar, LambdaStation: A forwarding and admission control service to interface production network facilities with advanced research network paths http://indico.cern.ch/contributionDisplay.py?contribId=359&sessionId=11&confId=