LHC Computing Grid Project - LCG

File Access Proposal to the Applications Forum

21st May 2003

David Foster – LCG Chief Technology Officer
Information Technology Division
CERN – European Organization for Nuclear Research
Geneva, Switzerland

david.foster@cern.ch

Introduction

- Access to files is required by worker nodes to avoid much copying of potentially large files. (e.g CMS pileup)
- Identified in the GDB-WG1 report
- This report was mandated by the GDB Meeting of the 9th December.
- Written document has been widely reviewed and needs more work to have a coherent written description.
- The intent is to describe what we think is reasonable in terms of functionality and achievable in a short time frame (next 6 months)

Thanks ….

- Thanks to Michael Ernst and Don Petravick for their work on creating the initial working document and many intermediate revisions including the post-CHEP meeting..

- Many people have extensively reviewed the working document.
  - The STAG
  - The GAG
  - The LCG Deployment Team
  - EDG-WP5
  - EDG-WP2
  - The contributors from CHEP

This is a summary of the conclusions re-cast into a development plan for LCG-1

- First … the language ….

Terminology

- A service is a process that is running which responds to input from a user interface or via a protocol interaction with another process.
- An API is a programmatic interface that can be called from another program.
- A storage system (SS) is a combination of:
  - Local disk storage
  - Mass storage system
  - Various services

I will not talk about a storage element

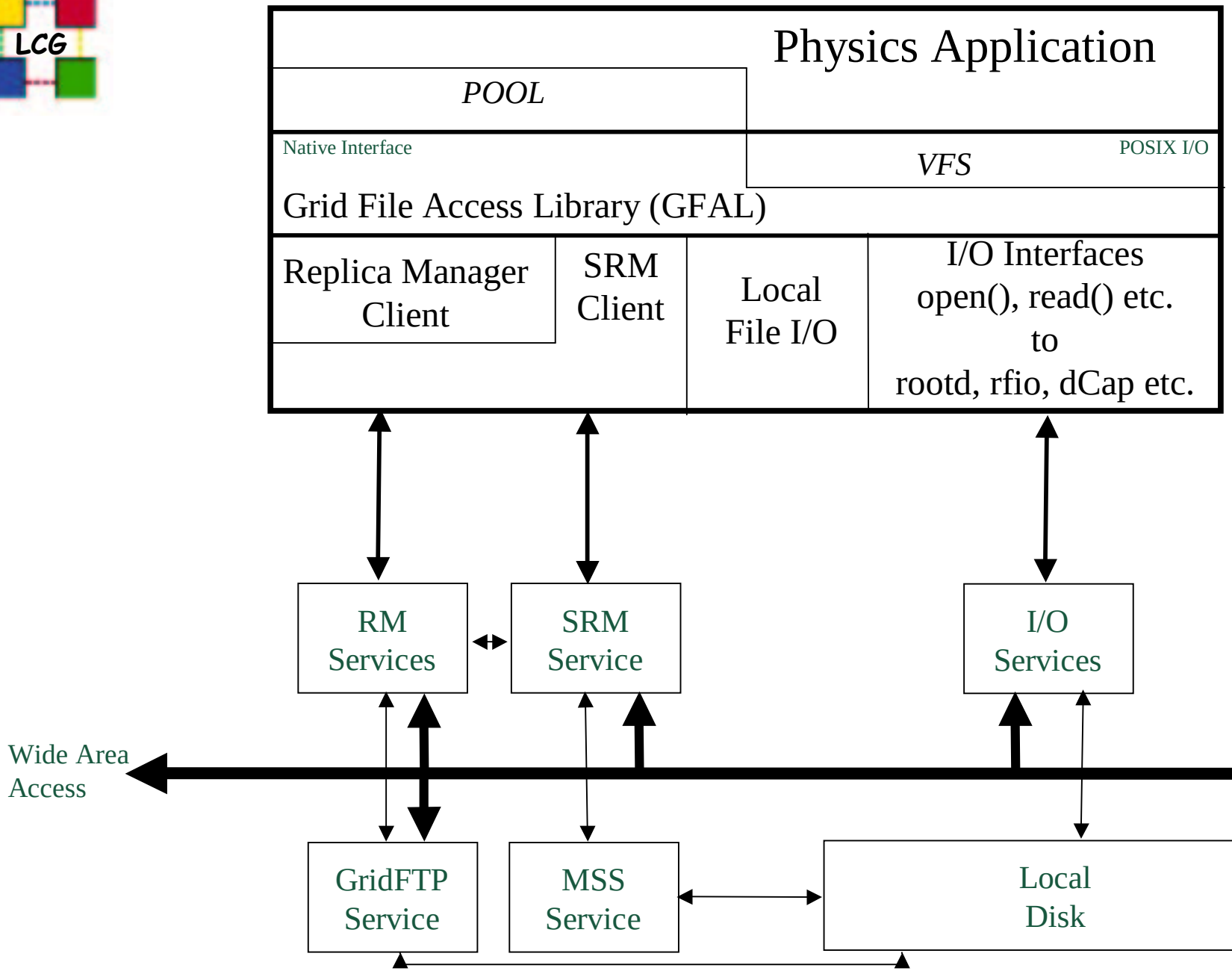
Terminology - 2

- A GUID is a globally unique identifier of a file (a bunch of numbers and letters).
- A SURL is a specification of a file that contains an access point specification (host and port) and a file path.
  - Given a GUID the RLS will return an SURL
  - The access point identifies the SRM service to be contacted.
- A SFN is the file path part of the SURL so is easily computable from the SURL
- A TURL is a specification of a file that contains the protocol to be used, the host and port to be accessed and the file path.
  - Given a SFN and a protocol the SRM will return a TURL

Terminology - 3

- Storage Resource Manager (SRM) is a service that provides
  - A command set for manipulating files on an MSS
- A Local Replica Catalog (LRC) is
  - A service that provides GUID->SURL mapping
- The Replica Location Service (RLS) is a collection of services (LRC, RLI etc). The Replica Manager is a Client API which uses these.
- A Replica Manager (RM) is an API
  - But has a service component in development.
  - Permits wide-area location/management of files.
- GridFTP is an API and a service that provides
  - File copying across a wide area
- A File Access Protocol (FAP) is
  - A protocol for accessing files via a client-server mechanism

The Functional View

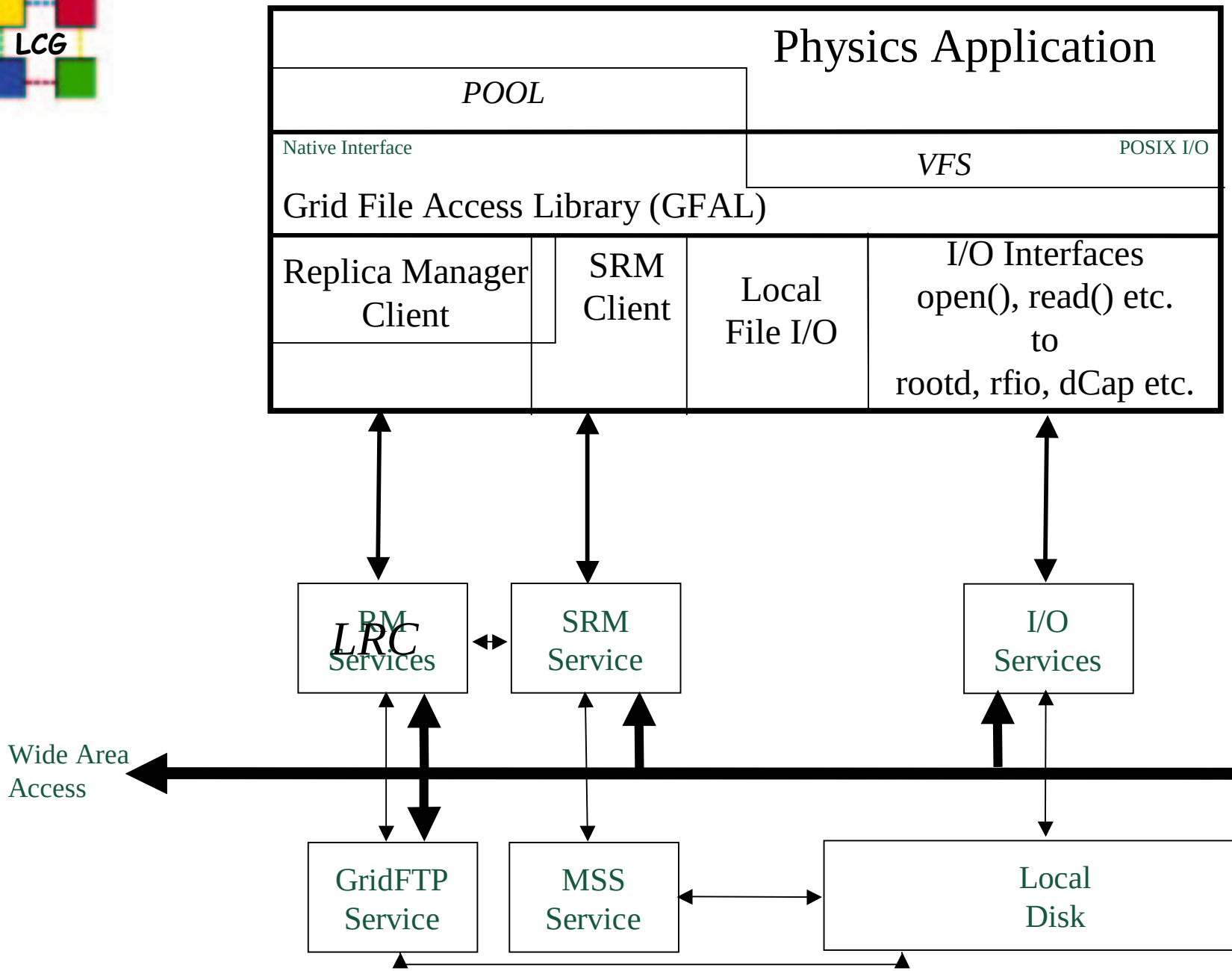


What the Services Do

- Replica Manager
  - Selects the best replica available using RLS, Replica Optimisation Services (ROS) as appropriate.
  - Can arrange to copy to the local storage system if not local but how wide area file access be managed from a policy point of view is to be discussed. May use GridFTP for this or SRMCopy
- SRM
  - Stages files to/from mass storage.
  - Checks file space availability (write)
- dCap, rfio etc
  - Transfers files to/from disk on a storage system

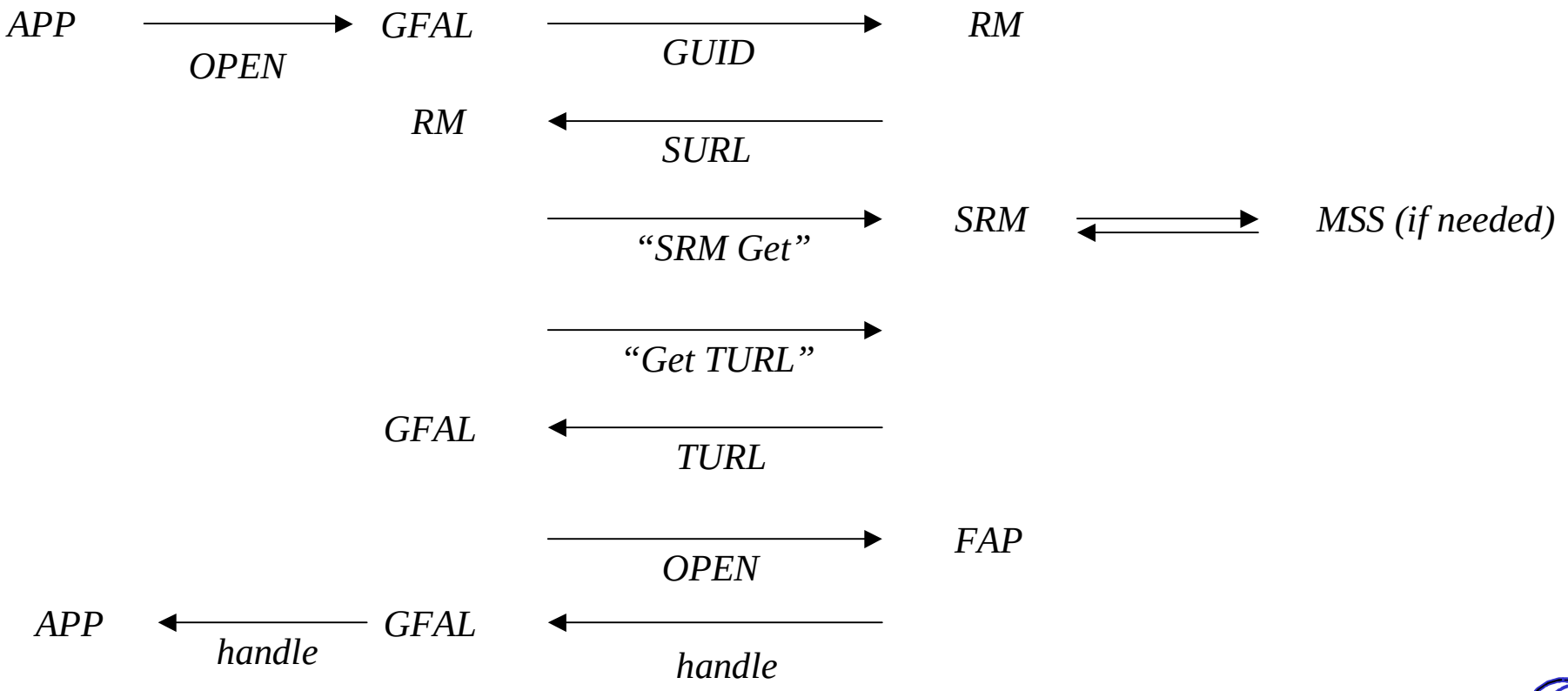
- The Grid File Access library orchestrates the interactions with these services transparently to the application but will need to be developed.
- The services can all act as 3rd party proxies for wide area interactions but this will require additional development in some cases.

The Functional View



The Simple Read Case

- Example Reading a file from the storage system (starting with a GUID)



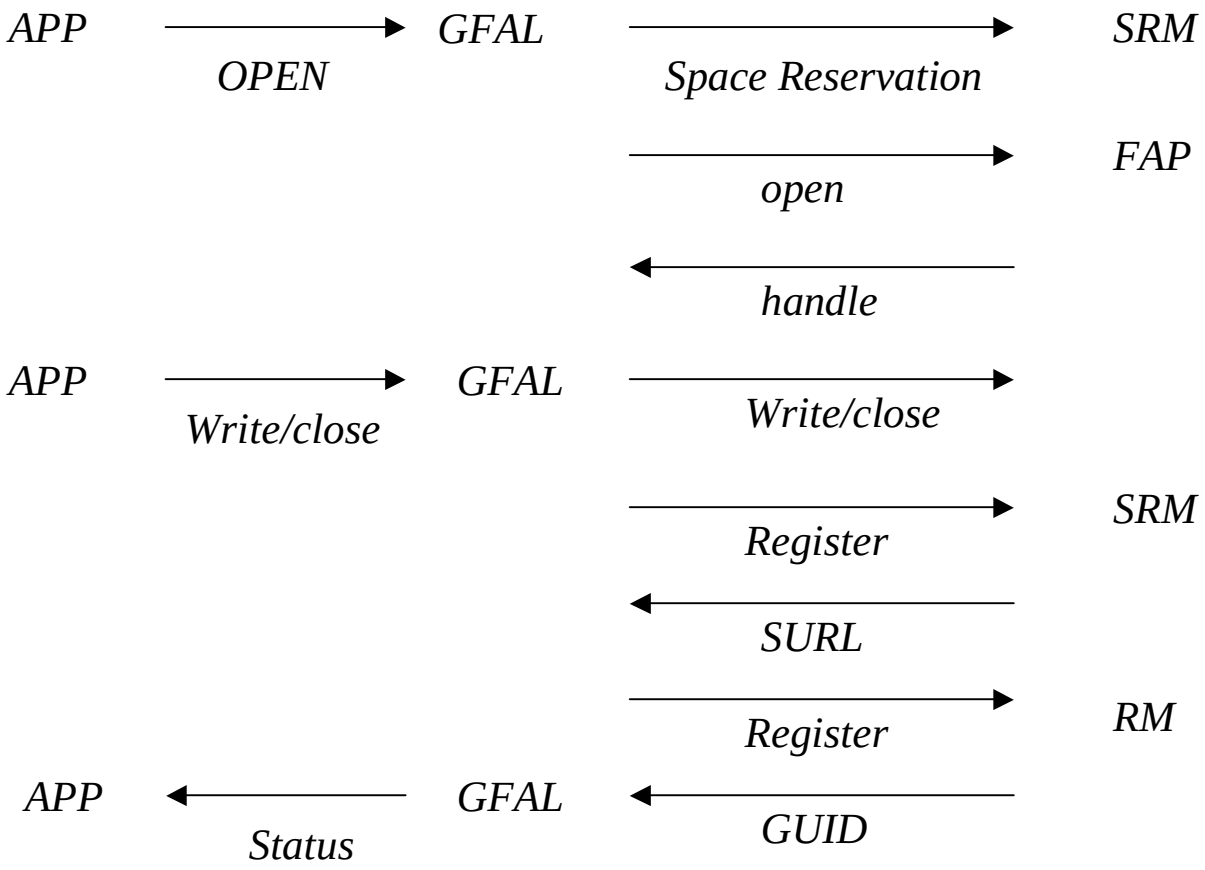
The extended cases

- What if the file is not local but a replica exists elsewhere?
  - RM services could copy the file locally (to the storage system or the worker node)
- What if the file is remote but on a remote MSS?
  - RM services could interact with the remote SRM to stage in the file before copying.
- Suppose I do not want to copy the file but have direct access?
  - Direct access to the remote file may also be technically possible through the file access protocols. But this a policy decision.
- What about writing to the wide area?
  - A policy decision.
- How do we deal with, interpret LFN's GUID's Collections etc.
  - Need to work on these issues now

But initially the assumption will be that the files needed are already present and registered on the local storage system

The simple write case

- Example Writing a file to the storage system



Some notes

- GridFTP service is available if needed by the application.
- The model of copying files to the worker node for read or writing to the worker node and then copying away is still possible.
- The model of copying and registering the files (before job execution) is still possible using replica manager services.
- The replica manager implementation will block until the file is recalled from the MSS. A more complex asynchronous mechanism can be envisaged but we have to understand how the higher levels will then handle this case.

What is needed?

- The Grid File Access Library
  - Configurable to support a number of underlying access protocols.
  - Single library deployable across all sites.
- The Storage Services
  - SRM interfaces to MSS and Disk Pools.
    - Enstore, Castor, HPSS, Atlas Data Store(RAL) all exist
  - File Access Protocol
    - Either rfio or dCap (or both .. Others ?)
  - GridFTP
  - Replica Manager Services under development (september) but API version is available.

To be done

- Identify software development resource and complete a design and implementation for the GFAL.
  - Verify Capabilities of SRM implementations on Tier-1's
  - Verify interface to Replica Manager
  - Verify the interface to POOL
  - Verify the interface to ROOT
  - Document all flows for file interactions with all components
- Target complete implementation 1.0 should be September. Early version in July to demonstrate basic read/write capability.
  - Simple local storage system access (needed files are pre-copied).
  - All interfaces to other packages design completed (Root, Pool)
- Understand the deployment issues.
  - How is this packaged/configured