

# Distributing BaBar Data using the Storage Resource Broker (SRB)

W. Kröger (SLAC), L. Martin (Univ. Paris VI et VII), D.  
Boutigny (LAPP - CNRS/IN2P3), A. Hanushevsky  
(SLAC), A. Hasan (SLAC)

For the BaBar Computing Group

CHEP, 23 March 2003

# BaBar Data Distribution

## Problem:

- Large amount of data files.
- Computing sites are geographically distributed which requires transfer and replication of data.

## Would like to have a system that:

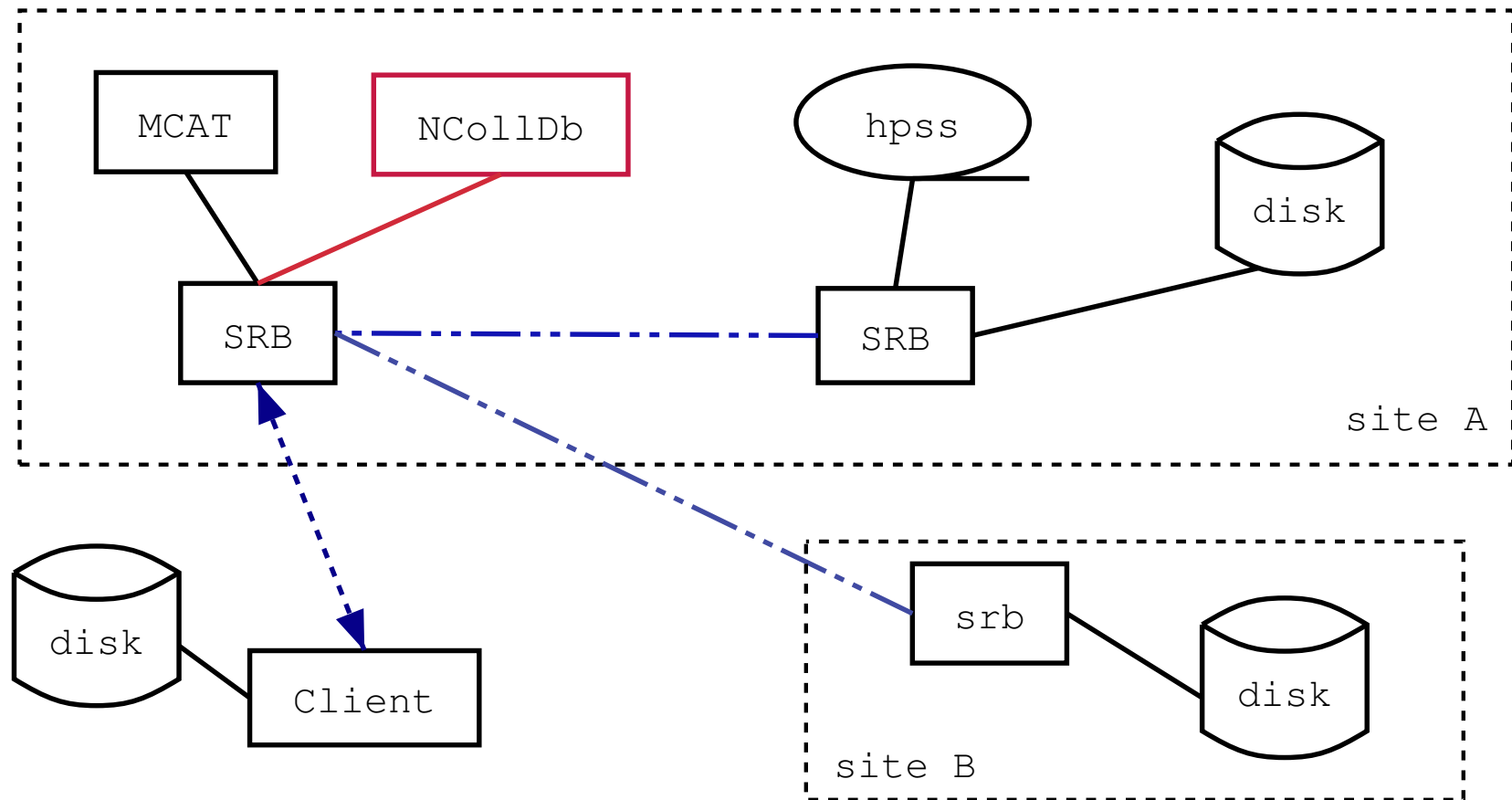
- Allows transparent access to files.
  - Replicate many files between Tier-A computing centers.
  - Download data to local sites in order to run analysis jobs.
  - BdbServer++: Make a deep copy of events at a Tier-A site, and copy the data to a local site (see poster).

- Allows access to data files and collections by attributes.
  - Find all files that were produced during a certain time period.
  - Find files that belong to a specific Objectivity federation.
  - Find collections from a particular run period.
  - Find number of events in a collection.
  - Find databases that belong to a collection.

## Storage Resource Broker (SRB)

- Storage Resource Broker has been developed by SDSC (<http://www.npaci.edu/DICE/SRB/>) .
- Client-server middleware.
- Uniform access to heterogeneous data resources distributed across a network.
- Metadata catalog (MCAT) allows access to files and resources based on attributes.
- Tools to put files into SRB, transfer and replicate files (S-commands).
- C-API to build tools.

## Storage Resource Broker, cont...



## Example of how to use SRB (S-commands)

# initialize connection to SRB

```
>> Sinit
```

# Register a file in SRB

```
>> Sregister -S 486817792 -R hpss-slac \  
    /objy/evs_g_aod00FA59.bdb    fdb/501/aod/evs_g_aod00FA59
```

# change directory (collection) in SRB

```
>> Scd fdb/501/evt
```

# Replicate a file to disk at IN2P3

```
>> Sreplicate -S in2p3Export evs_g_SP_aod00FA59 \  
    /objydatabases/tmp/test.bdb
```

**# list directory content**

```
>> Sls -l
```

```
0 hpss-slac      486817792  2002-11-02-23.08  evs_g_aod00FA59
1 ufs-slac       486817792  2002-11-13-17.39  evs_g_aod00FA59
2 in2p3Export    486817792  2002-11-21-12.34  evs_g_aod00FA59
```

**# remove a particular replica**

```
>> Srm -n 2 evs_g_SP_aod00FA59
```

**# list all file that were added to SRB after Nov 1, 2002**

```
>> Sls -A 'DATE>2002:11:01'
     evs_g_SP_aod00FA59
```

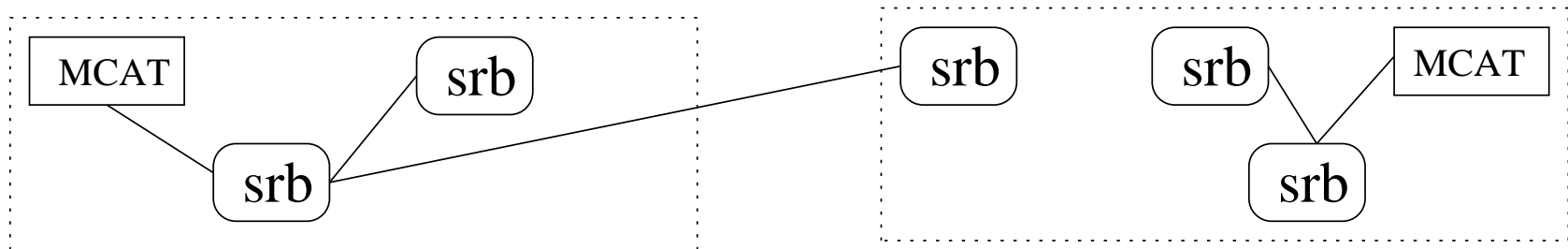
## Experiment Specific Catalog: NCollDb

- NCollDb: Experiment specific catalog. Contains attributes that are specific to BaBar and the Objectivity based event store.
  - Objectivity databases: name, ID, FDB ...
  - collections: name, run, status, databases, Objy-fdb ...
- Information stored in different catalog (not SRB MCAT)
- Catalog is queried through an interface SRB provides. Write SQL queries or use predefined queries.
- NCollDb is linked to the MCAT.



## Distribution of Objectivity Dbs

- Use SRB to transfer Objectivity data between two Tier-A sites, SLAC and IN2P3.
- Each site has its own SRB MCAT. Manages only its local data.
- Objectivity databases also have to be imported into an Objectivity federation.
- The location of the databases in HPSS is registered.
- Not used to manage disk resident copies for analysis.



## Distribution of Objectivity Dbs

Typical scenario to transfer data:

- Find Data files in SLAC MCAT, e.g: all files registered after certain date, or all files belong to a federation.
- Transfer the files from SLAC to IN2P3 (Sget, Sreplicate).
- Get Objectivity related metadata from NCollDb tables (using SRB).
- Register file with correct path in IN2P3-MCAT.
- Delete the replica from the SLAC MCAT.
- Run import Tools using the metadata to import database into Objectivity federation.

## Current Status

- SRB is installed at SLAC and IN2P3.
- MCAT servers run at both sites.
- The SRB version is 1.1.8 but the new 2.0 release was installed three weeks ago.
- NCollDb exists, but not completely stable yet.
- System still in test phase.
- Transferred files successfully, but in 1.1.8 the file transfer is too slow. New version much faster using multiple streams.
- Scripts are being developed to automate the transfer for many files.

## Problems/Issue

- SRB was designed to have only one MCAT, does not provide functionality to federate MCATs.
- Now, with multiple MCATs one has to connect to each of them in order to find out if a file is available at a site.
- SRB assumes that it has control over files. Might not match with existing infrastructure at a site. If files get deleted outside of SRB one has to propagate this information to SRB.
- The transfer rate was slow, but we see large improvements in new version.

## Conclusion

- SRB allows easy access to data.
- SRB has been used to transfer Objectivity files from SLAC to IN2P3.
- New version allows faster transfer.

### Future:

- Finish automation to transfer many files.
- Extend the usage of SRB beyond Objectivity databases.
- Setup a production system to allow users/application to use the SRB.
- Understand if SRB could play a role for data distribution in the new BaBar computing model.