THE UNIVERSITY OF MELBOURNE

Tom Fifield
fifieldt@unimelb.edu.au

Belle II Data Handling Prototype

Recall

- The goal of the testbed is to test the concepts proposed by Kuhr
- A solution using gLite middleware, with Belle II-specific software based on ideas used at CDF
- A prototype demonstrating this was deemed worth creating (Computing Meeting)

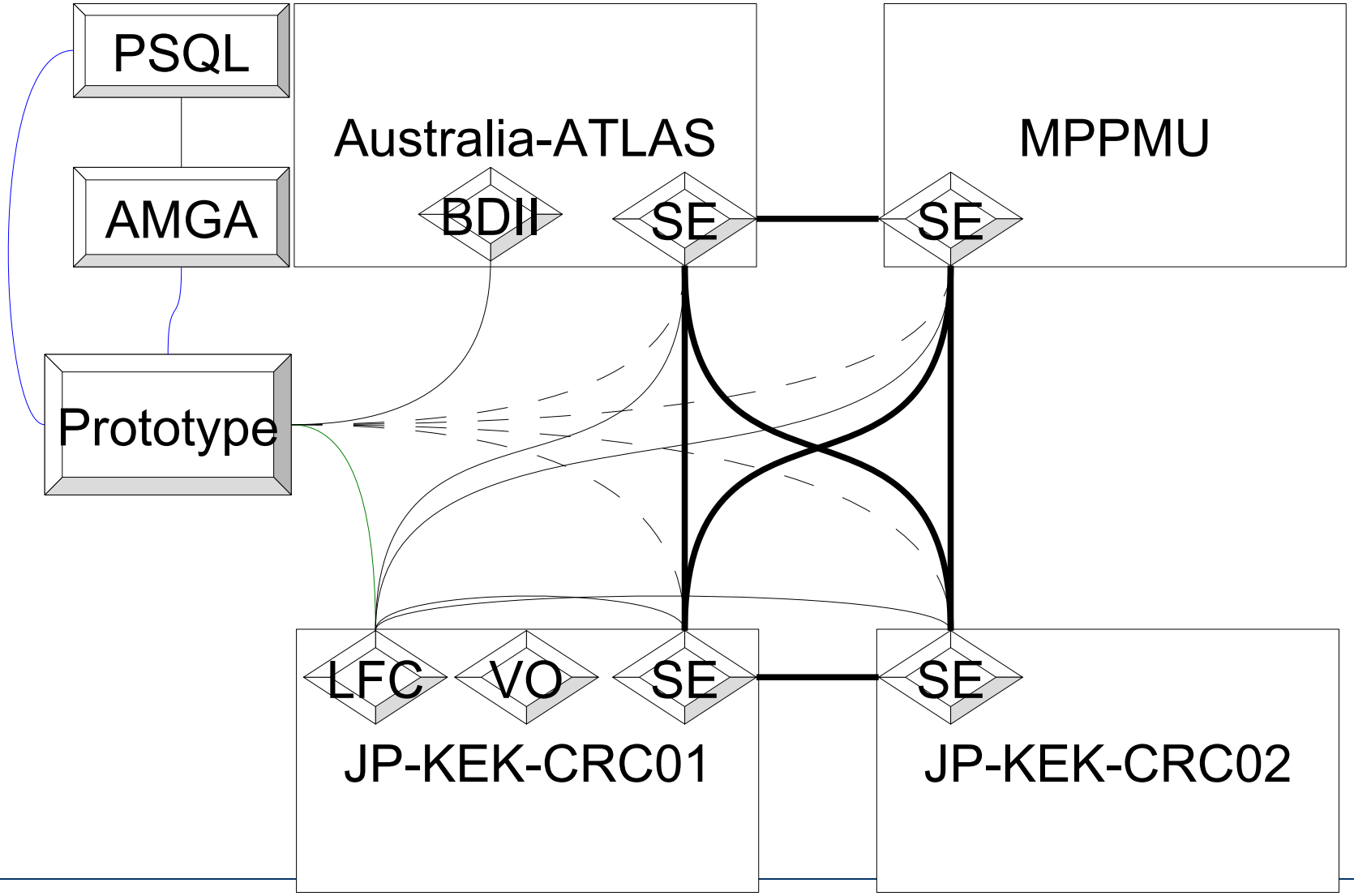
Aims

- Testing technical possibilities
  - Determining strengths, weaknesses and potential problems
- Making an application that's reusable for simulation
  - Can be used to add test data for “real” system
- Providing Polish developers with a basis

Implementation

- “Hacky” Python
  - Single thread, poor performance
- Python bindings for
  - LFC
  - AMGA
  - LCG-UTILS
- Uses existing Belle VO

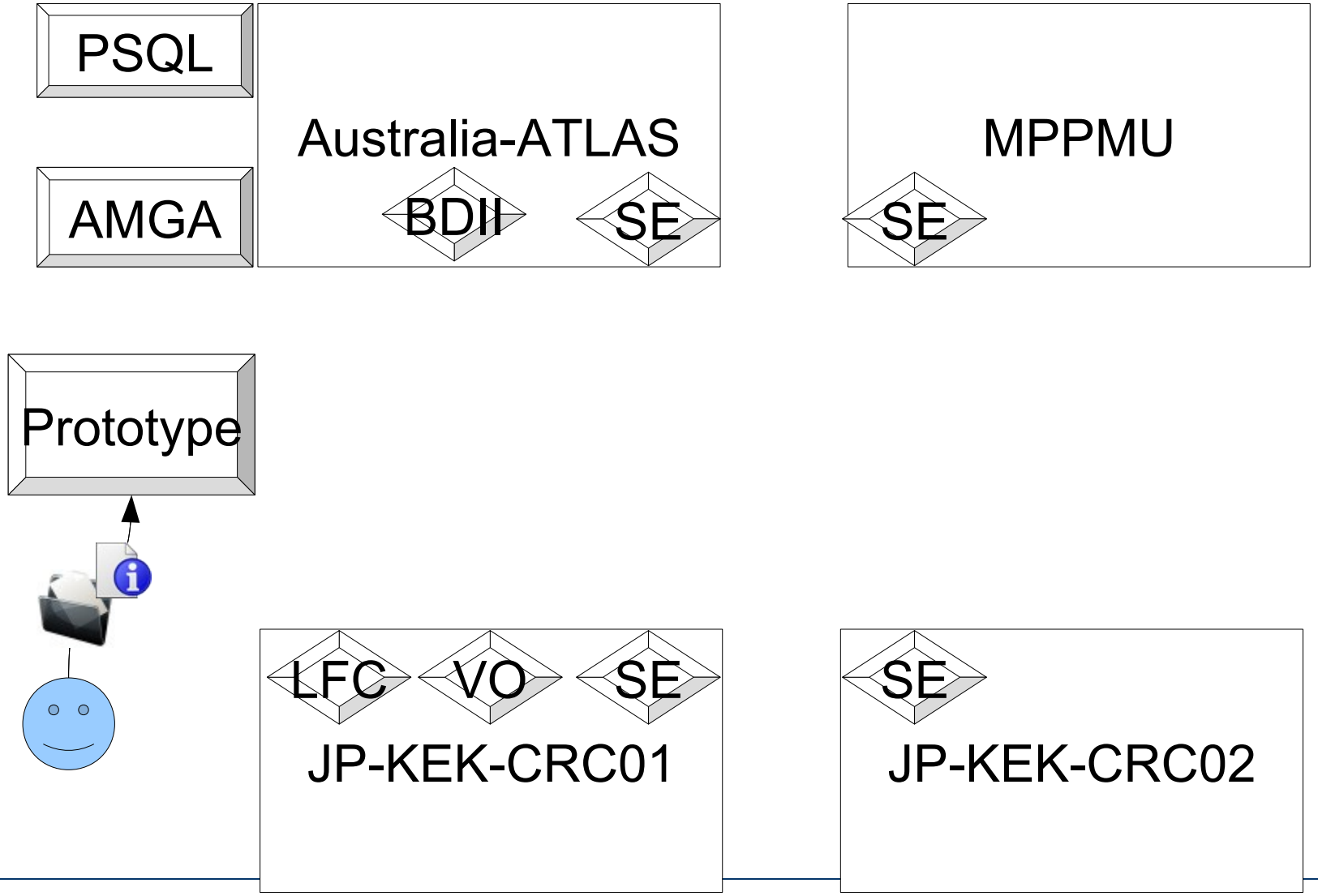
Block Diagram



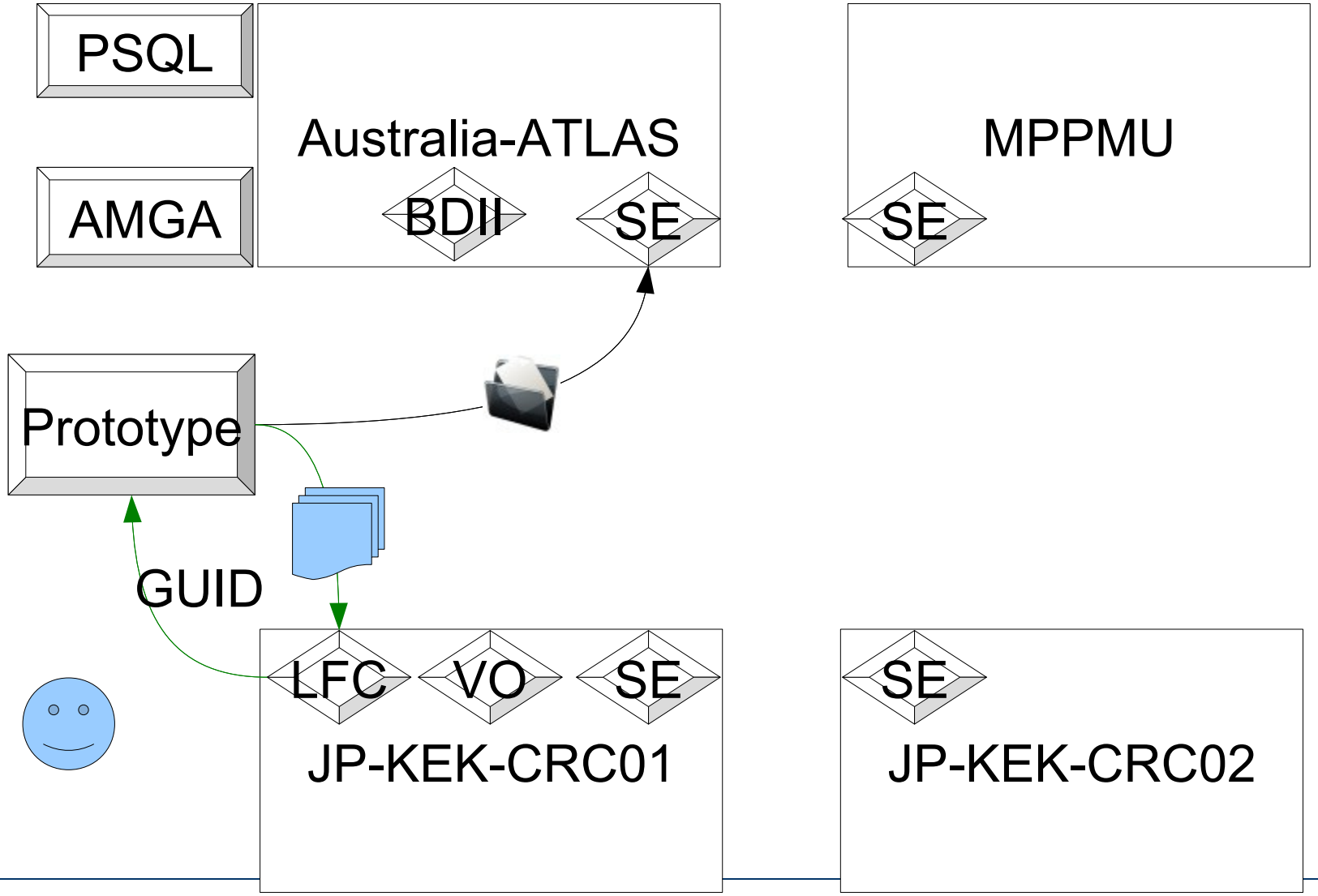
Datasets

- Create:
  - protoype -c /local/path/to/dataset [-s SE]
  - Inside local directory are files, filename.meta.data, and meta.data
    - Only like this for testing
  - Copies files to designated SE
  - Updates LFC with information
  - Registers metadata and GUID in AMGA

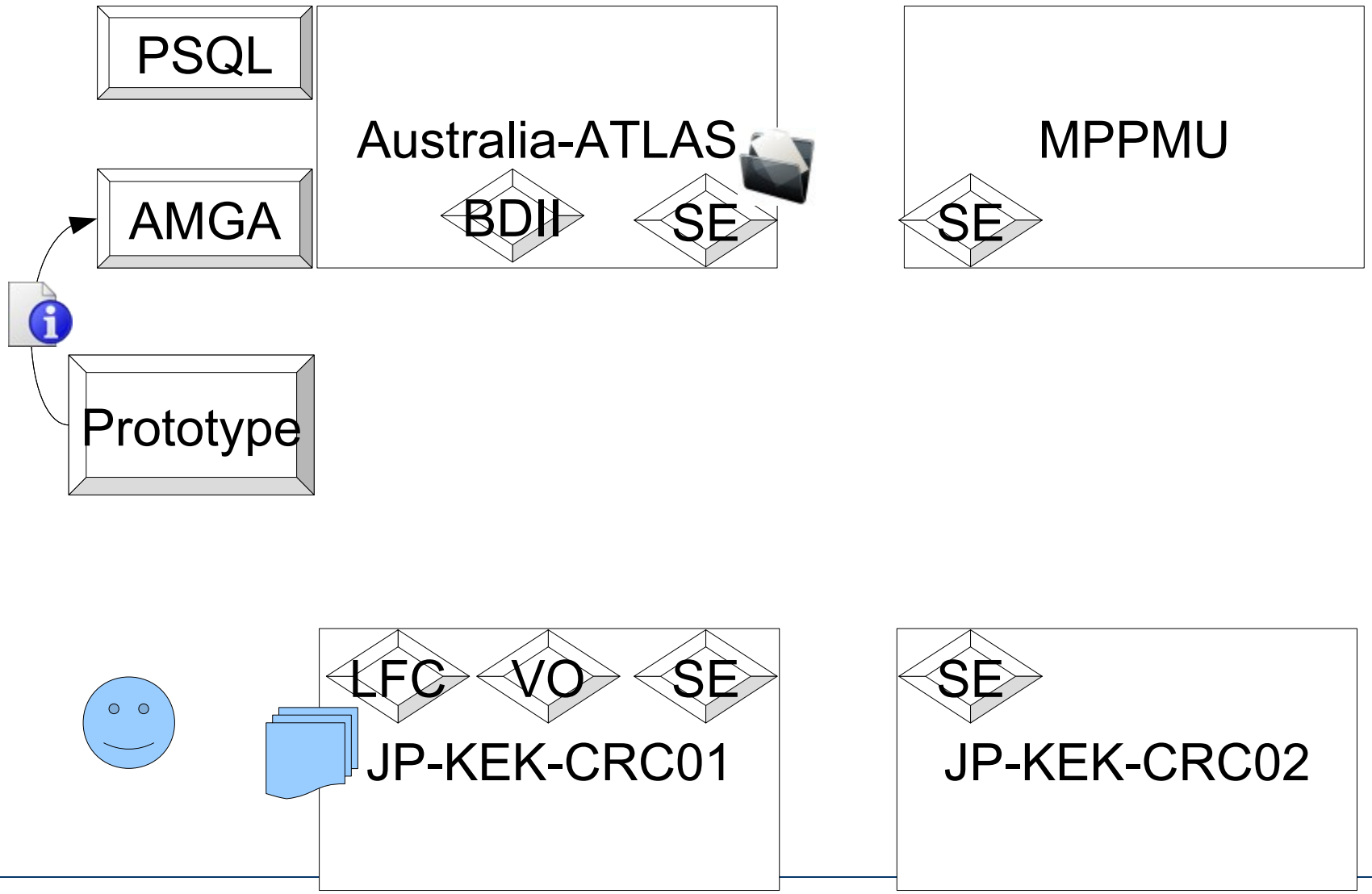
Dataset creation



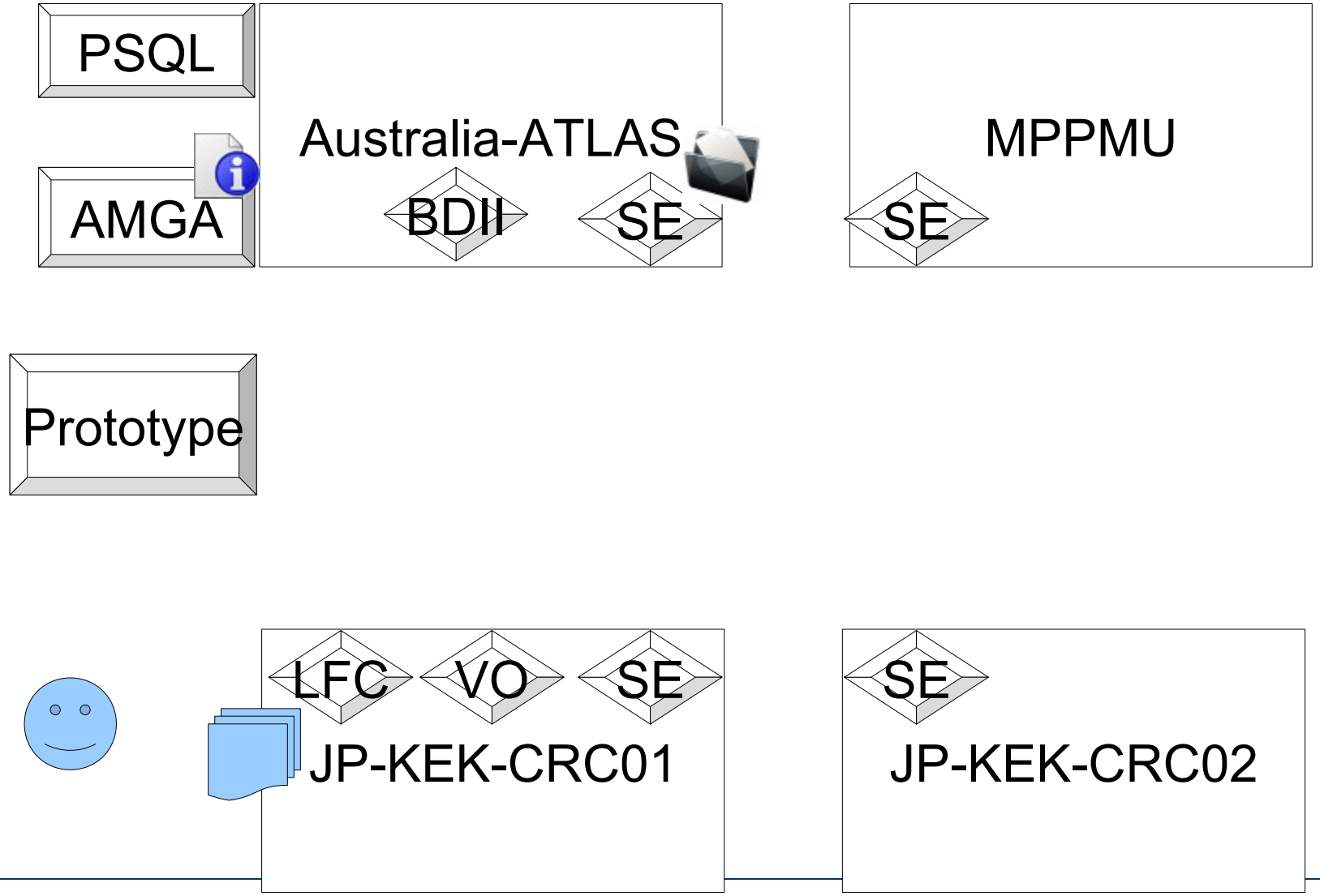
Dataset Creation



Dataset Creation



Dataset Creation



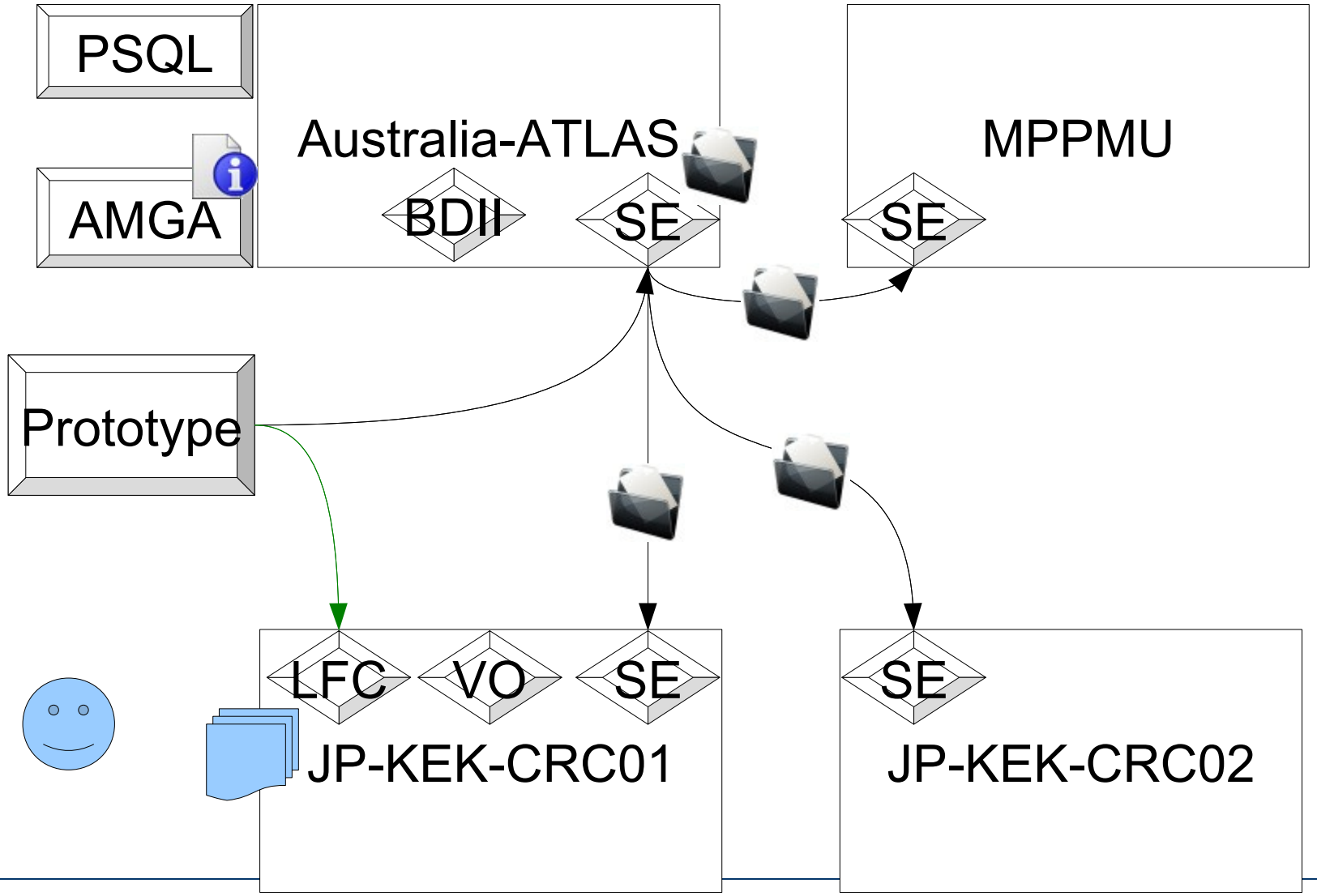
Examples

```
[fifieldt@ui ~]$ ./prototype.py -c ./belle2/data/63/HadronB
Doing SSL handshake using builtin SSL
...
Sending >  addentry /belle/data/63/HadronB complete '0' date '2009-06-24 09:31:39' name 'HadronB' description '"HadronB
      dataset for experiment 63"' data_type 'real' data_level 'mdst' acc_mode 'continuum' user_id 'fifieldt' soft_id 'b2009' <
Session ID 0
Using grid catalog type: lfc
Using grid catalog : dg16.cc.kek.jp
SE type: SRMv2
Destination SURL :
      srm://agh3.atlas.unimelb.edu.au/dpm/atlas.unimelb.edu.au/home/belle/fifieldt//belle/data/63/HadronB/file5
Source SRM Request Token: fede2715-0774-4e46-8ab3-ba753b6e6e8f
Source URL: file:/home/fifieldt/./belle2/data/63/HadronB/file5
File size: 14
VO name: belle
Destination specified: agh3.atlas.unimelb.edu.au
Destination URL for copy: gsiftp://ags1.atlas.unimelb.edu.au/ags1.atlas.unimelb.edu.au:/gridstorage/pool1/data/belle/2009-
      06-24/file5.280499.0
# streams: 1
         14 bytes      0.04 KB/sec avg      0.04 KB/sec inst
Transfer took 1010 ms
Using LFN: lfn:/grid/belle/fifieldt//belle/data/63/HadronB/file5
Using GUID: guid:c36462f6-b1bd-4e08-a8b3-9e7c7dc6e7db
Registering LFN: /grid/belle/fifieldt//belle/data/63/HadronB/file5 (c36462f6-b1bd-4e08-a8b3-9e7c7dc6e7db)
Registering SURL: srm://agh3.atlas.unimelb.edu.au/dpm/atlas.unimelb.edu.au/home/belle/fifieldt//belle/data/63/HadronB/file5
      (c36462f6-b1bd-4e08-a8b3-9e7c7dc6e7db)
...
```

Datasets (cont...)

- Replicate
  - protoype -r /path/to/dataset -s “['site', 'site',...,'site']”
  - Copies files to each SE in list
  - Uses closest SE to destination SE
  - Updates LFC
  - No AMGA update needed – AMGA tracks GUIDs which do not change
- Delete
  - protoype -d /amga/path/to/dataset
  - Deletes the dataset and metadata

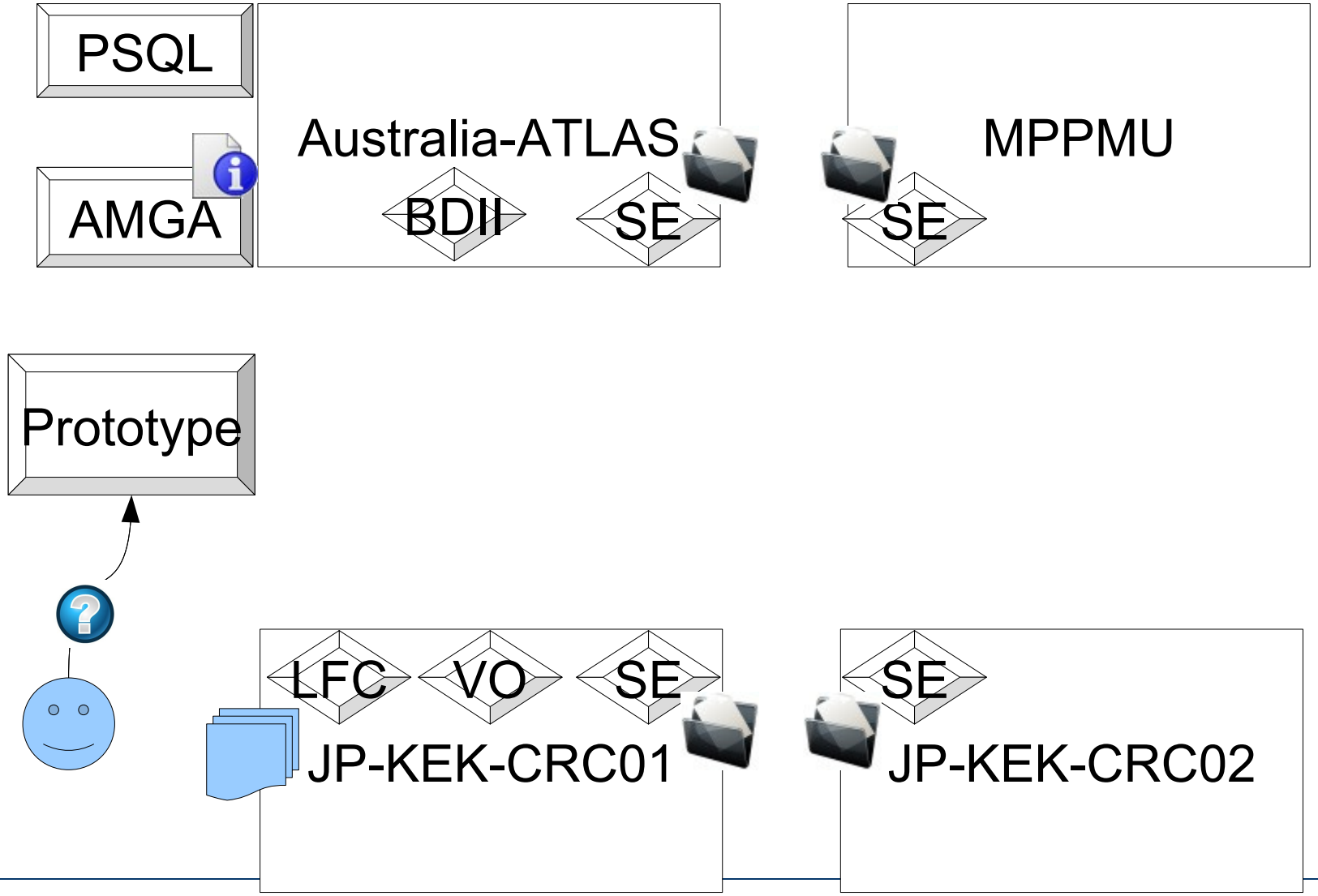
Dataset Replication



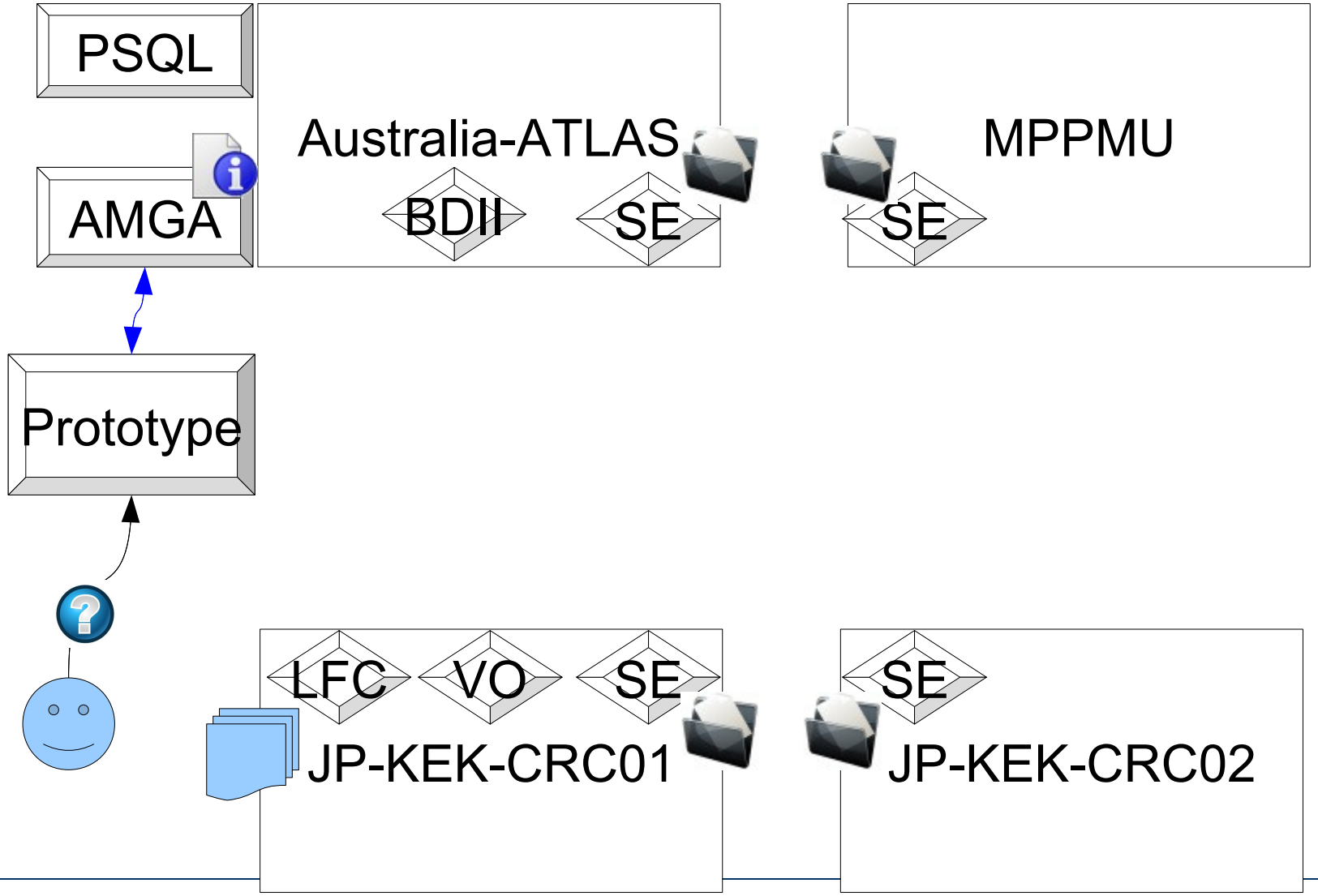
User Queries

- prototype -q <data/MC> -e exp_no -t skim_type -s 'attr=val'
- prototype -q <data/MC> -e [exp_no, exp_no2, ...] -t '[“skim_type”,...] -s "attr='val' and attr2='val' and ... and attrN='val'"
  - Aiming for flexibility
  - Search string (“-s”) is a free list of any metadata element – you can even use boolean logic
  - Can accept multiple experiments, skims

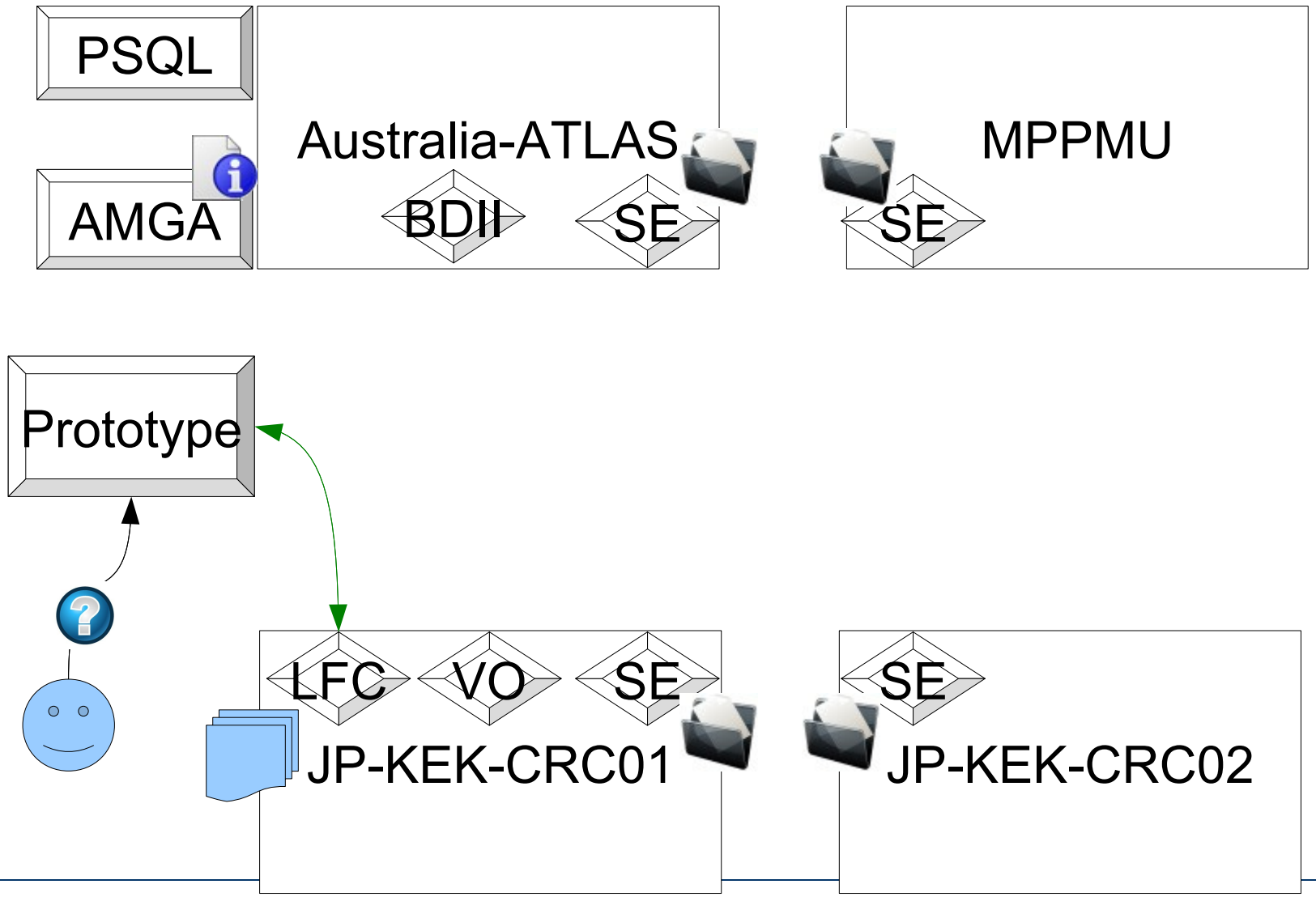
User Query


PSQL
AMGA
Australia-ATLAS
BDII
SE
MPPMU
SE
Prototype
LFC
VO
SE
JP-KEK-CRC01
SE
JP-KEK-CRC02

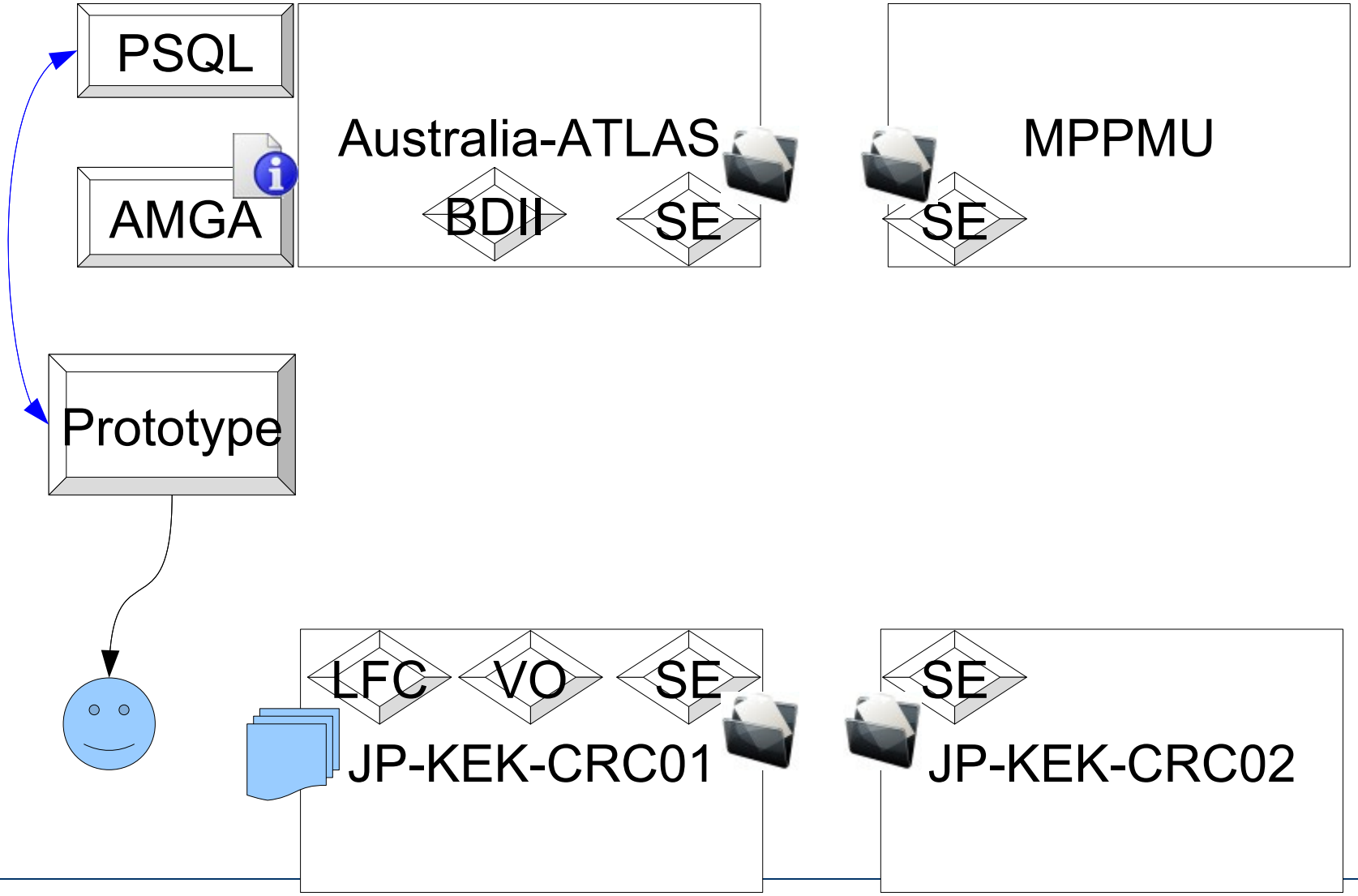
User Query


PSQL
AMGA
Prototype
Australia-ATLAS
BDII
SE
MPPMU
SE
LFC
VO
SE
JP-KEK-CRC01
SE
JP-KEK-CRC02

User Query


PSQL
AMGA
Australia-ATLAS
BDII
SE
MPPMU
SE
Prototype
LFC
VO
SE
JP-KEK-CRC01
SE
JP-KEK-CRC02

User Query


PSQL
AMGA
Australia-ATLAS
BDII
SE
MPPMU
SE
Prototype
LFC
VO
SE
JP-KEK-CRC01
SE
JP-KEK-CRC02

Examples

```
[fifieldt@ui ~]$ ./prototype.py -q data -e 61 -t HadronB -s 'user_id=”fifieldt”'
Doing SSL handshake using builtin SSL
Sending >  selectattr  /belle/data/61/HadronB/file2:guid 'user_id="fifieldt"' <
Session ID 0
Sending >  selectattr  /belle/data/61/HadronB/file3:guid 'user_id="fifieldt"' <
Session ID 0
Sending >  selectattr  /belle/data/61/HadronB/file4:guid 'user_id="fifieldt"' <
Session ID 0
Sending >  selectattr  /belle/data/61/HadronB/file5:guid 'user_id="fifieldt"' <
Session ID 0
Sending >  selectattr  /belle/data/61/HadronB/file6:guid 'user_id="fifieldt"' <
Session ID 0
Using grid catalog type: lfc
Using grid catalog : dg16.cc.kek.jp
Project ID#42
Can be run at:
JP-KEK-CRC-01
JP-KEK-CRC-02
MPPMU
Australia-ATLAS
```

Examples

```
[fifieldt@ui ~]$ ./prototype.py -q data -e '[61, 63]' -t HadronB -s 'user_id="fifieldt"'
/belle/data/61/HadronB available at  Set(['JP-KEK-CRC-01', 'JP-KEK-CRC-02',
    'MPPMU', 'Australia-ATLAS'])
/belle/data/63/HadronB available at  Set(['Australia-ATLAS'])
Set(['Australia-ATLAS'])
```

Project ID#68

Can be run at:

Australia-ATLAS

Examples

```
[fifieldt@ui ~]$ ./prototype.py -q data -e '[61, 63]' -t '["HadronB", "TauPair"]' -s
    'user_id="fifieldt"'
/belle/data/61/TauPair doesn't exist (ignoring)
/belle/data/63/TauPair available at Set(['JP-KEK-CRC-01', 'Australia-ATLAS'])
/belle/data/61/HadronB available at Set(['JP-KEK-CRC-01', 'JP-KEK-CRC-02',
    'MPPMU', 'Australia-ATLAS'])
/belle/data/63/HadronB available at Set(['Australia-ATLAS'])
Project ID#80
Can be run at:
Australia-ATLAS
```

- **You can see from the type of queries I run, I'm not a physicist (maybe HH and FullRecon would be better?)**
- **We need your help to make sure the metadata is useful to <u>you</u> – see the next presentation :)**

Examples

```
[fifieldt@ui ~]$ ./prototype.py -q data -e '[61, 63]' -t '["HadronB", "TauPair"]' -s
    'user_id="fifieldt" and no_evt<200'
/belle/data/61/TauPair doesn't exist (ignoring)
/belle/data/63/TauPair available at Set(['JP-KEK-CRC-01', 'Australia-ATLAS'])
/belle/data/61/HadronB available at Set(['JP-KEK-CRC-01', 'JP-KEK-CRC-02',
    'MPPMU', 'Australia-ATLAS'])
/belle/data/63/HadronB available at Set(['Australia-ATLAS'])
Project ID#84
Can be run at:
Australia-ATLAS
```

Future Work

- This handles the steps which
  - Get data where it needs to be
  - Allow everyone to find out what data is where
- Still to come
  - Making that data accessible to the framework on computing resources
  - Submit a job to a site using the project ID
  - Cream CE

Scaling

- AMGA supports replication
  - eg 'Master' at KEK, slaves at regional centers (i.e. “Tier 1s”)
- AMGA can also use oracle backend
- LFC also supports “central” and “local” instances
- All of these storage technologies are proven and in use on the WLCG
- One project server per region

THE UNIVERSITY OF

MELBOURNE