

Porting EventStore to GlueX

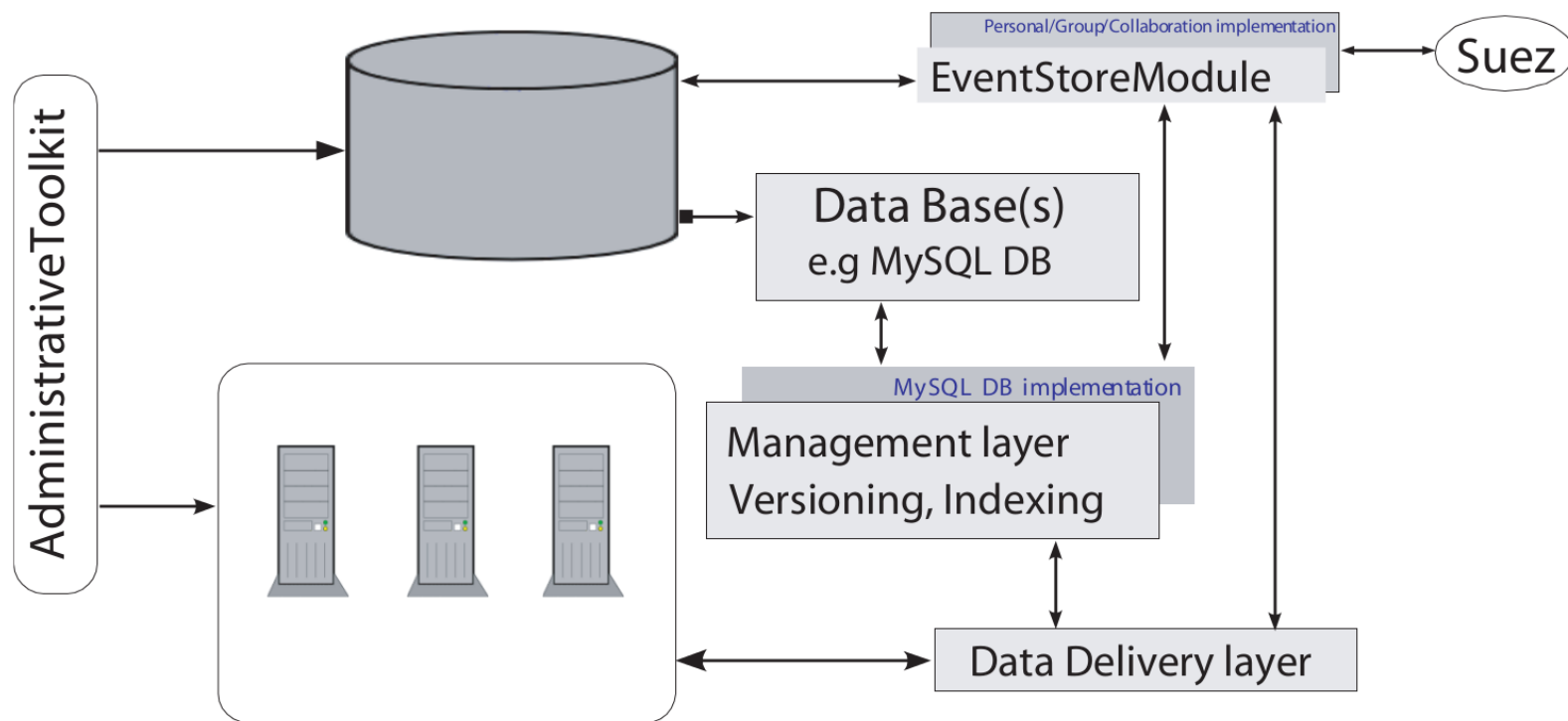
Sean Dobbs
Northwestern U.

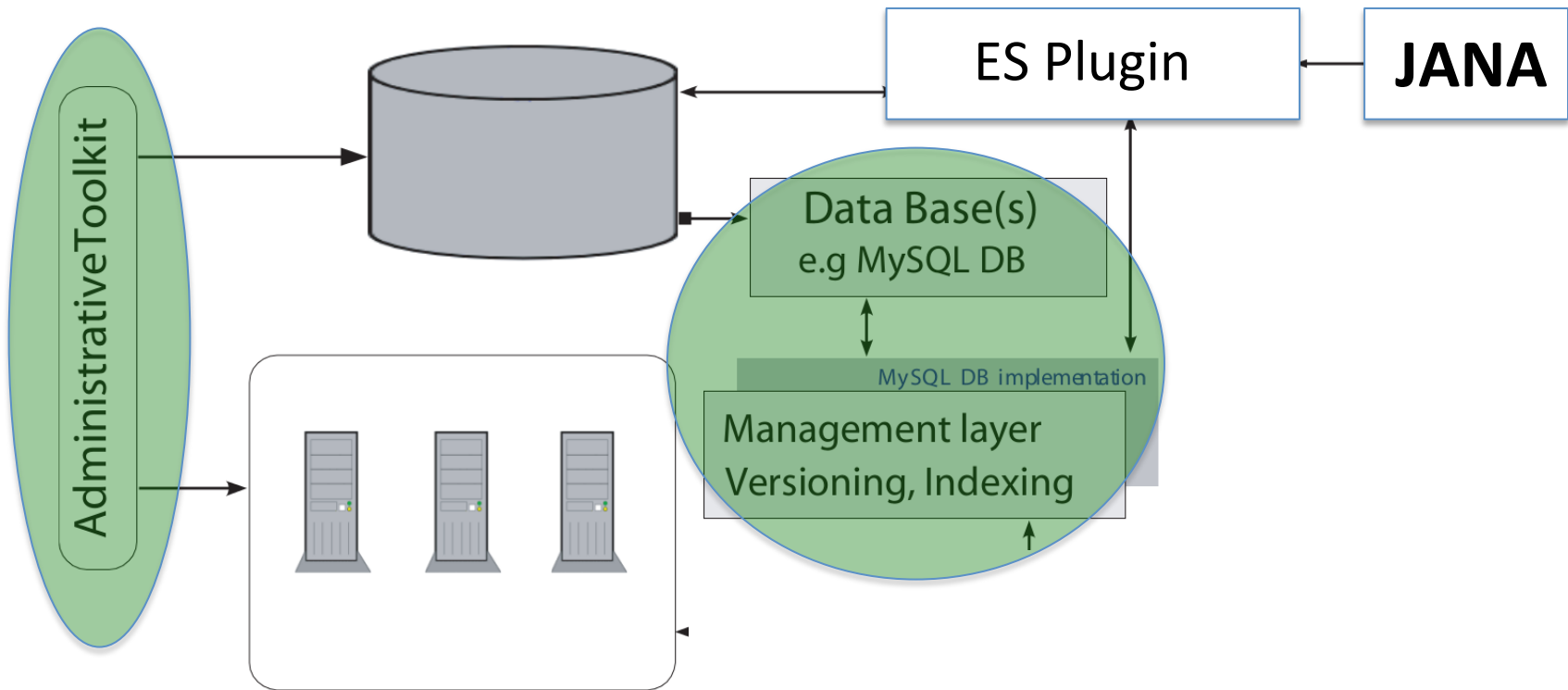
EventStore

- EventStore provides a framework to
 - Manage multiple versions of data
 - Provide physics-oriented interface to access data
 - Access various subsets or skims of data
- Is implemented with standard technologies
 - C++/python, uses SQLite/MySQL DBs
 - Metadata DB is accessed as web service
 - Modular design to allow support for accessing different file types and delivering data to different systems (i.e. JANA)
 - Don't delete anything, retain history

Example Invocation

- eventstore in 20061202 physics runs 22034 22156
 - “20061202”: timestamp
 - “physics”: grade of data (daq, pass2, dskim, etc...)
 - Various metadata
 - “runs X Y”
 - “dataset dataXX”
 - “energy psi(2S)”
 - etc...





CLEO Data Model & EventStore

- Reconstructed data is saved in several files per run, corresponding to several physics categories
- Data format used is PDS (Packed Data Storage)
 - Unindexed, sequentially accessed collection of records
 - Support for several “streams” corresponding to various types of event and calibration data
 - Data records are identified by run, event, and stream, timestamps used for synchronization
 - Information is kept on which proxies are needed to read the data and construct objects
- EventStore keeps track of
 - Which events are stored in which files
 - Different “views” (skims) of events
 - Allows for personal and Collaboration-wide DBs

Data Life Cycle

- HDDM files generated
- Indexes generated
 - Relates run/event numbers to index into file
 - Separate index for each skim
- Generate ES DB for a particular run, provide timestamp and version information
- Merge standalone ES DBs into main ES DB
- Reprocessed data? It should be in a different location, follow the same procedure.
- New skim? Generate a new set of indexes and update DB.
- For more info on DB schema see, e.g. <https://wiki.classe.cornell.edu/HEP/SWIG/EventStore>

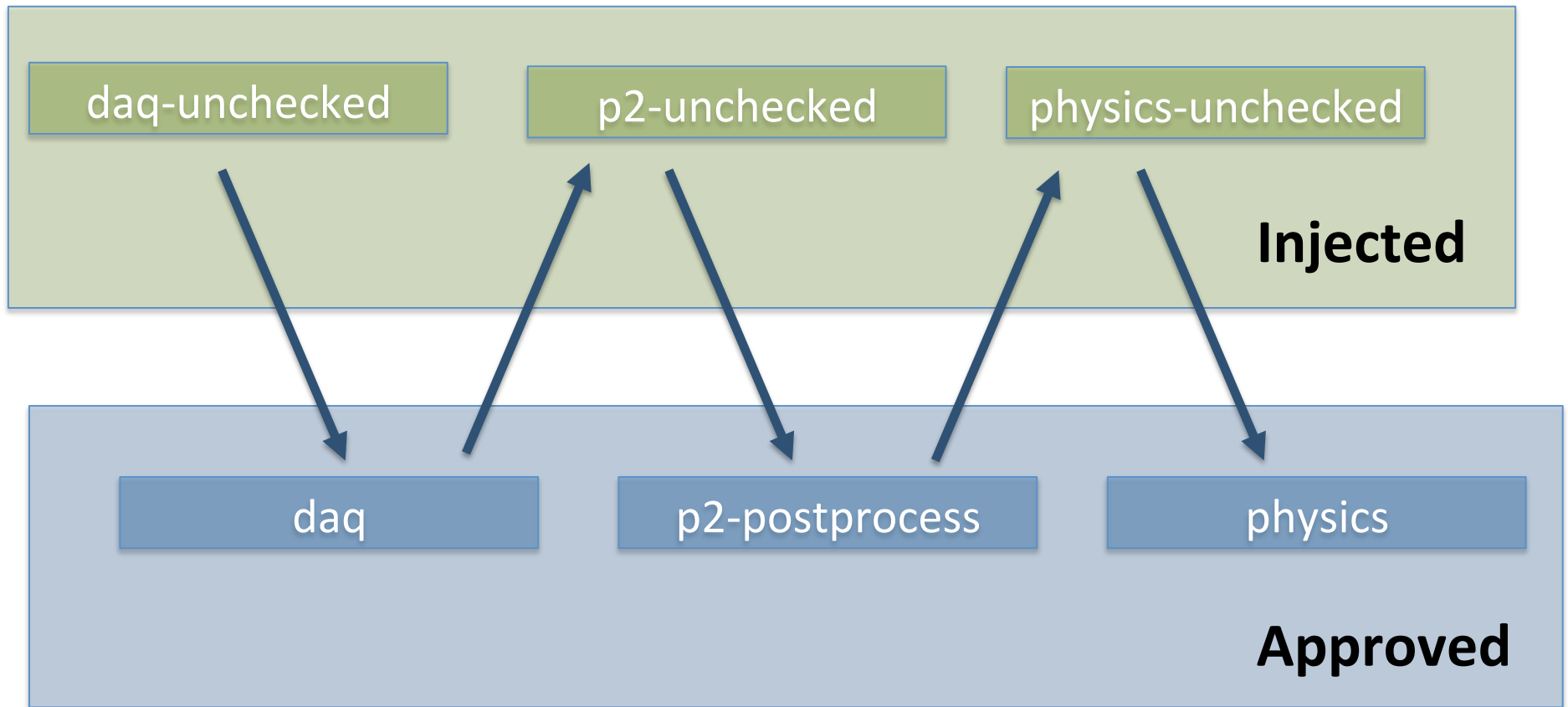
Metadata

- Run-based metadata is used to select ranges of runs based on certain criteria (i.e. beam energy)
- This data is stored in a separate DB and accessed via a SOAP-based web application.
- The separate metadata DB allows personal EventStores to access this data as well.
- How would such a DB relate to other run condition DBs?

Roadmap

- Review management layer
 - Write HDDM indexing code, python wrapper
- Write JANA plugin
 - How to specify ES query?
 - Make sure skims are handled in a consistent way
- Recommission metadata web service/DB – done!
 - What metadata do we want?
- Add some user friendly-ness
- What files do we want to manage?
- Consistent software version naming scheme would be useful.

CLEO Data Lifecycle



Other things

- Data version tags
 - dataX_sim-recon-XXXX_hddm-XX_ccdb-XXXX
- Metadata
 - Data or run period number (“dataX”, “runX”)
 - Electron beam energy (“12GeV”, “6GeV”)
 - Coherent peak energy (“9GeV”, “6GeV”, “none”)

Can it work with the grid?

- Yes, if we store file locations as URIs
- Need to make sure that transport layer properly serves up the files
- Need to add grid-compatible authentication mechanisms
- How about interacting with other job management systems?