Objy Infrastructure and activities

- Schema management
  - Currently, there is SRT support for running ooddlx for any .ddl file
    - This is what defines the type numbers for Objy
  - Today we use ***named schemas*** to decouple the definition of the type numbers for different packages
    - the Database/DB_Schema package defines ***ALL*** named schemas
    - each package with persistent types has ***ONE*** ddl file: <package>.ddl
      - EventStructure, InDetEvent, LArEvent, MuonEvent, TileEvent, TruthEvent, eventually the DetDescr packages
    - all packages with ddl ***depend on*** (I.e. use) DB_Schema package
      - forces DB_Schema to be built first

Objy Infrastructure and activities (2)

- Today's build procedure
  - Check out "all" packages with ddl files
  - Create federation, e.g. on atlobj01, the developer server
    - see DB documentation
  - Set OO_FD_BOOT to point to boot file
  - configure with "--with-objectivity"
  - run gmake
    - This runs ooddlx and compiles .cxx files
- Near-term evolution
  - Can exploit tool to dump/load schema to ASCII files
    - can eliminate the running of ooddlx for existing schema
    - ooddlx used only for packages being developed

Objy Infrastructure and activities (3)

- HepODBMS
  - Today we use some elements of HepODBMS
    - Application manager, clustering, want to try out their conditions db
  - HepODBMS defines their own named schemas
    - These are included into DB_Schema
    - One can then use ooschemaupgrade to load schema into federation

Objy Infrastructure and activities (4)

- Status of storage in Objy
  - We are currently able to store the digits for all systems and Truth (vertices and tracks)
  - Uses visitor pattern and strategies of the current event model
  - Persistent storage for detector description information associated with the events is currently being implemented
    - transformation matrices, decoding parameters
  - This should then allow to load databases and work without Zebra
  - Next steps will be to evolve this to work within the framework once Transient Event Store and transient/persistent mapping has been worked out