

Conditions Database Internals

J. Ohnemus
LBNL

Overview

- Conditions data read every 30 minutes.
- Begin and end times encapsulated in an interval object.
- Interval objects form a time series (doubly linked list).

Overview

- interval objects are small
 - 1 interval db per subsystem
 - 1 container for each physical conditions device
- conditions objects are large
 - span multiple db

Versioning

- Versioning capability implemented
 - Uses Objectivity versioning mechanisms
 - interval object is versioned
 - original interval object information retained
 - Versioned intervals form a genealogy
 - tree structure: next and previous associations
 - genealogy object acts as multiplexer
 - genealogy members can be named
 - original interval is default version
 - doubly linked interval list follows latest version

Versioning

- Versioning is automatically implemented when needed
- Interface

```
BdbDatabase theDatabase( "emc" );  
BdbCondClusteringHint theHint( "emc" );  
BdbHandle(EmcPulserCal) theObject =  
    new( theHint.updatedHint() ) EmcPulserCal;  
theDatabase.store( theObject, "EmcPulserCal", theBeginTime );
```

Versioning

- new()
 - creates conditions object in the appropriate db and container
- store()
 - creates interval object in the appropriate db and container
 - uses class name for container name
 - implements versioning when needed

Versioning

- Basic algorithm

```
if ( theBeginTime > lastInterval._beginTime )
    { append new interval to end of list }
else
    { version an existing interval }
```

Versioning

- store() overloaded

```
BdbDatabase::store( BdbObject theObject, char* theContainerName,
                   BdbTime theBeginTime )
BdbDatabase::store( BdbObject theObject, char* theContainerName,
                   BdbTime theBeginTime, BdbTime theEndTime )
BdbDatabase::store( BdbObject theObject, char* theContainerName,
                   BdbIntervalBase theInterval )
```

New Interface

- Uses event store style clustering hints
- Multiple processes can read/write with Objectivity 5.0

```
BdbDatabase theDatabase( "emc" );  
BdbCondClusteringHint theHint( "emc" );  
BdbHandle(EmcPulserCal) theObject =  
    new( theHint.updatedHint() ) EmcPulserCal;  
theDatabase.store( theObject, "EmcPulserCal", theBeginTime );
```

Summary

- Conditions db code is working
 - versioning implemented
 - new user interface uses clustering hints
- Future additions
 - mechanisms for changing default path through interval chain
 - swap in code for multiple processes (Objectivity 5.0)
 - performance issues