Calibration

◆ Status Overview	Dave Brown
◆ IR2 Database	George Zioulas
◆ Config Download	Yury Kolomensky
◆ Datastream Calibration	John Bartelt
◆ Subsystem Surveys	Dave Brown
◆ What’s needed for May	All

Status Overview

- The cosmic run was a limited success
  - Basic toolkits in active use
    - Accumulation, fitting,...
  - Calibration ODF framework in general use
    - Data-driven configuration without code reload
    - Subsystem specific configuration data attached to CalCycleTC hierarchy
  - Datastream calibration infrastructure completed and excercised (John)
    - DCH time-to-distance calibration
- There were 3 ‘failures’ for calibration during the cosmic run
  - Large (>250K Bytes/crate) Datagrams cannot be transported by Dataflow
    - Lots of time and effort expended to trace this back
    - Represents a new fundamental constraint not considered in the existing design
  - Conditions DB was not used to store constants
    - No suitable IR2 federation available
    - No subsystem code to exploit the existing tools (including DIRC)
    - Exacerbated by above problem (broke Odf-> Oep connection)
  - Physics / Calibration/ Physics run change not excercised
    - Key-driven configuration not used (private keys for calib, keyvalue=1 for physics)
    - Reverse-dataflow, database download server not ready

What Are We Doing About These Problems?

- Database-driven configuration (Yury)
  - We cannot make an abrupt transition to this from the cosmic operation model
    - Lots of information in flat files
    - Little subsystem experience in the technology
  - The Necessary (IMHO) requirements for configuration in May are:
    - Unique, managed top-level configuration keys for physics + calibration running
      - defines reproducable, documented configurations
      - Associates DAQ, DC, and L3 configurations
    - User managed scratch configuration keys for fast diagnostics and testing
    - Access to current conditions data (calibration)
  - This system is available NOW
    - Online release 2.4.1
    - CalConfigAction and CalBeginAction interface change: **subsystem action REQUIRED**
    - Existing files must be converted or rewritten
- Online use of Objectivity Databases
  - IR2 Federation (George Zioulas)
  - User survey (Dave Brown)

Large Datagram Transmission

- Most subsystems are affected by this to some degree
  - Diagnostic mode
- Dch, Emc developed private workarounds for the cosmic run
  - No connection between Dataflow and Oep!
- Official Workarounds are under construction
  - Provide a 'normal' Oep connection
  - Data compression (gzip)
    - Reduces CalTC datagram size by ~factor of 3
    - Should allow ALL systems to transmit calibration results via dataflow
  - NFS file transport
    - Official version of Dch, Emc workaround
  - Serial transmission (1 ROM per endMxxx transition)
    - Accumulation and fitting still done in parallel
    - Works (automatically) in platform space
    - Transparent to Oep
  - Prototypes will be ready near the end of February
  - Interface changes will be necessary (Oep, CalCycleTC)

CalTCProxy

- Enables download of current calibration constants
  - Needed for interim (file-based) download
  - Used directly in database implementation of BdbTCServer
- Fully implemented templated class
  - Instantiate on the CalChan subclass
  - Construct with the CalType name
- Will only serve data in the conditions DB
- Must be instantiated in a subclass of BdbTCProxyLoader
  - example: DrcCond/DrcCalTCProxyLoader

User Survey of Online Database Usage

- Survey asked basic questions to subsystem calib experts
  - What DB functionality is needed to run your system in May?
  - What core code is needed to support that?
  - What subsystem code needs to be written to provide that?
  - Are you planning on storing cosmic run calib constants in the DB?
  - What are your concerns about using Objectivity online?
- Respondants: Emc, Drc, Ifr, Dch

Survey Results

- All respondants need some DB constants to configure
  - Configuration data
  - Conditions data (calibration, hot channels)
- Core functionality seems adequate
- All respondants need to develope code
  - Exploit existing tools to store and download constants
- Cosmic run constants are generally not needed in the DB
  - Important exception: Dch time-to-distance calib
  - Drc T0s would be nice (improve resolution)
- Concerns about Objectivity
  - Perceived as unnecessarily complicated and unreliable by some
  - Accepted as necessary by others

Getting Ready for May

- Core development must be completed
  - large datagram transmission workarounds
- Support subsystem exploitation of download + calib DB
  - Examples
  - Prompt response to problems and bugs
  - Tutorials, programming help, documentation, ...?
- Develope an operation plan for calibration running in April