



# Persistency Project (POOL) Work Plan Proposal for 2002

---

**Dirk Düllmann**

*dirk.duellmann@cern.ch*

SC2 Meeting, August 2<sup>nd</sup> 2002

# Persistency Project Timeline

---

- Started officially 19 April
  - Initially staffed with 1.6 FTE
  - Zhen Xie/CMS (100%) - MySQL scalability and reliability test
  - DD (60%) – Completed discussion about requirement list and deployment plans with experiments
- Persistency Workshop 5-6 June at CERN
  - More requirement and implementation discussions
  - Draft work package breakdown and release sequence proposed
  - Additional experiment resource commitments received
- Since beginning of July
  - Real design discussions in work package started
    - active participation now ~5FTE
    - Daily meetings with core component WPs
    - Meet twice a week for Catalog WP
    - Collection and Meta Data discussion just starting now
  - Project h/w and s/w infrastructure being defined and becoming available

# Experiment Deployment Plans

---

- Summary of first round of discussions with the experiments
- Numbers referring to minimal requirements for a first release (rather than to constraints imposed by the design/implementation)
- Timescale for first release – September (CMS)
  - Others are interested but more relaxed about date
- Volume – 10-50TB (CMS)
- Files – several 100k (ALICE)
- Distribution – O(10) sites (CMS)
- Number of population jobs – 10k (CMS)
- Use of REFs per Event (CMS/ATLAS/LHCb)
  - LHCb: O(100) refs per event; CMS (100-1000);
  - ALICE: not relying on Refs so far
- Recovery from jobs failures (CMS, ATLAS)
  - CMS: Existing setup allows to just re-issue same job
    - Append to existing files
  - LHCb: each file is written by exactly one job
  - ALICE: single event spans several files

# Experiment Focus of Interest

(prioritised by # of votes)

---

- RootIO integration with grid aware catalog (ALL)
  - Transparent Navigation (ATLAS/CMS/LHCb)
    - ALICE: maybe, but only in some places
  - EDG (ALL), Alien (ALICE), Magda (ATLAS)
- MySQL as RDBMS implementation until first release (ALL)
- Consistency between streaming data and meta-data (CMS/ATLAS)
  - At application defined checkpoints during a job
- Early separation of persistent and transient dictionary (ATLAS/LHCb)
- Initial release supports persistency for non-TObjects (ATLAS/LHCb)
  - without changes to user class definitions
- Support for shallow (catalog-only) data copies (CMS)
  - formerly known as cloned Federations
- Support for deep (extracted part-of-object hierarchy) copies (CMS)

# Practical Project Organisation

---

- Stay close to initial RTAG component model
  - Work Packages are (nearly) completely aligned
- Constraint:
  - at least two significant contributions (>25% FTE) per work package to insure independent work
  - have core component developers co-located at CERN
- Some interface/design discussion will need to take place
  - inside co-located work packages outside CERN
  - by email / phone meetings
  - small workshops at CERN

# Proposed Work Package Split

---

- Earlier Drafts have been shown on the Persistency Workshop and also later by Torre
  - Catalog and Grid Integration
    - **Zhen Xie** (50%), Maria Girone (50%), Mathias Steinecke
  - Storage Manager & Refs
    - **Markus Frank** (50%), Giacomo Govi (50%), Fons Rademakers(25%)
  - Dictionary & Conversion Services
    - **Craig Tull**, Stefan Roiser (50%), Victor Perevoztchikov (30%)
  - Collections and Meta Data
    - **David Malon**(30%), Chris Laine (50%), Julius Hrivnac (20%), Sasha Vaniachine (20%), RD Schaffer(30%)
  - Common services, integration and testing
    - **Giacomo Govi** (50%), Maria Girone (50%), Zhen Xie (50%)
- some commitment still under discussion

# Basic S/W Infrastructure Setup

---

- **Persistency is the first LCG project**
  - will need to operate before some of the basic LCG services are fully agreed and implemented
- **Need to participate during setup basic s/w infrastructure**
  - Repository directory structure
  - Need to build on a few basic services
    - Error Reporting and Exception hierarchy
    - UUID generation
    - Component loading and configuration
    - Configuration control, build and automated test system
  - For now start as pragmatic (and simple) as possible
    - Extract implementation for all of the above from existing projects
- **Expect to change as soon as wider scale agreement is achieved in LCG**
  - will require additional work for re-factoring eg after V0.0.1 release

# POOL Hardware Resources

---

- Discussed hardware requirements with other groups in IT
  - Proposal distributed by Bernd Panzer
- Now-December
  - 2 disk server machines to be setup as catalog servers
    - one with native MySQL catalog / meta data server
    - one installed as EDG testbed node acting as replica location server
  - 10 client nodes for concurrency test
    - installed as testbed nodes once EDG for RH7.2 is available
- September-December
  - stability and distribution test (decoupled from development setup)
    - 2 additional disk servers for decoupled stability and distribution tests
    - 3 disk servers for ROOT data
- November-December
  - more resources for experiment framework integration tests from experiment data challenge allocations as required by experiments

# Proposed Release Strategy

---

- AIM
  - complete first internal release cycle as soon as possible
- Start by integration of core components
  - StorageMgr, FileCatalog, Reflection, CacheMgr & Refs
  - Result: transparent navigational store operational
- Other (more external) components will start design and implementation already
  - but will only be integrated in later releases
  - in particular Collections and MetaData are essential for final product but less tightly coupled
- Only the first external release will have all components integrated
  - usable for expert users (eg no automatic installation)

# Proposed Release Sequence

---

- **September - R 0.0.1 - Basic Navigation**
  - all core components for navigation exist and interoperate
    - StorageMgr, Refs & CacheMgr, Dictionary (r/o), FileCatalog
  - some remaining simplifications
    - Assume TObject on read/write – simplified conversion
- **October - R 0.0.2 – Collections**
  - first collection implementation integrated
    - support implicit and explicit collections on either RDBMS or RootIO
  - persistency for foreign classes working
    - persistency for non-TObject classes without need for user code instrumentation
  - EDG/Globus FileCatalog integrated
- **November -R 0.0.3 – Meta Data & Query (external release)**
  - meta data annotation and query added
  - for event, event collection and file based meta data

# Summary

---

- Real design work has started
  - more than half of the people who signed up for the project are actively working towards a very first release
    - Consultancy and (in some areas) modifications from ROOT team
  - expect some inefficiencies because of vacations and project being a test case for new LCG s/w components and procedures
- Need at least one design cycle to obtain
  - realistic estimate of available and required resources
  - fully specified release descriptions for public releases
  - as input for a more detailed and longer term work plan as input to SC2 & PEB
- Current proposal up to V0.0.3 Release (November)
  - Assuming that committed resources become available