

POOL Status & Plans for 2004

Dirk Düllmann
IT-DB & LCG-POOL
LCG SC2 meeting
16th April 2004



POOL Status



- Main functionality has been implemented
 - RDBMS independence layer and meta data storage still to be achieved
 - Proof of concept completed successfully
 - Some remaining issues with Storage Element integration
 - SRM and remote file I/O (next generation Grid Middleware?)
 - Integration into experiment frameworks is proceeds well in ATLAS, CMS and LHCb
 - Close contacts between POOL and core s/w teams
 - First Data Challenges using POOL successful so far
 - Main goal for 2003: three successful data challenges using POOL
 - Performance can still be improved with this feedback
 - Main penalties are non-POOL issues
 - Conditions DB sub-project started
 - Workshop showed clear interest and agreement in IOV service
 - First pre-releases made



Main Goals for 2004



- Stabilise POOL s/w products
 - Focus on performance improvements rather than large functionality changes
 - In line with the experiments plans for the data challenges
- Help to simplify the integration into experiment frameworks
 - Tight coupling between POOL and experiment development and production teams
 - Automated schema loading, usability tools, documentation improvements
- Establish the new LCG ConditionsDB in production
 - After an initial consolidation round
- Achieve POOL independence of the RDBMS backend
 - And extend the set of supported RDBMS systems



2004 Work Plan



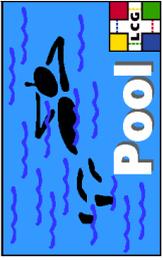
- Draft document has been discussed in POOL, Architect Forum and last weeks PEB
 - http://lcgapp.cern.ch/project/docs/POOL_POW-05-04-2004.pdf
- Based on work package plans and experiment priorities
 - Thanks to all experiments for their concrete and detailed input
 - Significant overlap between the different experiment requests



Follow-up on Internal Review



- All review recommendations covered by the POOL work plan
- Schema Evolution (Q2/Q3)
 - Will start from the ROOT support and try to confirm that POOL does not restrict the ROOT functionality significantly
 - Setup a test suite for all cases documented on the ROOT side and document the POOL support
 - Will need to address also schema evolution on the RDBMS layer
- Provide ROOT plug-ins for Ref<T> and POOL Collections
 - Allows to use POOL functionality inside ROOT as an interactive analysis environment (Q2)
- Restructure the POOL documentation (Q1)
 - All documentation formats derived from a shared set of DocBook text modules
 - Minimises the overlap and possible inconsistencies between design and user documents.



POOL integration & usability



- Received requests for tools which would simplify the daily development work of pool users
 - Create or recreate a POOL catalog from a set of interrelated POOL files (Q1)
 - Provide command line tools for consistent file manipulation of POOL files (Q2)
 - Eg copy, move and rename in the local file system together with the associated POOL catalog changes
- Provide eg in the POOL contrib area a central repository of scripts developed by an experiment
 - To share the experience in the deployment of POOL (Q1)
 - May later support and distribute some of these tools if there is sufficient agreement



Storage Manager



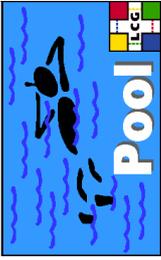
- Optimisation required in several areas
 - Client side resource usage - memory, CPU, file handles (Q1-Q4)
 - Mass storage handling (minimise costly requests) (Q3)
 - "Transparent" double to float mapping (Q3)
- Automated schema loading (Q2)
 - Based on upcoming SEAL service
 - In cooperation with ROOT team to allow late integration of data types for already open files
- Bug fixes - more complex cases
 - Eg std containers with user defined allocators which define local data - aka CLHEP Matrix (Q1) (done in 1.6.2)
- Move to ROOT V4.0 (Q3)
 - After careful investigation of backward compatibility and schema evolution issues
- RDBMS backend based on the RDBMS Abstraction Layer
 - Storage of simple data structures into a RDBMS via the same interface as for objects stored on the streaming layer
 - Two step plan:
 - First allow for objects which can trivially be mapped to SQL tables (Q3)
 - Possibly later an extension to more complex C++ objects (Q1 '05?)



File Catalog



- Significant development completed already (Q1)!
 - Support for LCG-2 (V1.5)
 - Support for Composite Catalogs (V1.6)
- File Catalog as model for handling and exchanging data could be a prototype for other (very similar) meta data catalogs (Q2/Q3)
 - Collection catalog and Collection entries
 - Condition Folder catalog and Condition Data
- Cataloguing, extraction based on meta data, publishing are all very similar
 - Even the component implementation could be factorised out and shared
 - Performance of XML as exchange format for larger data amounts needs to be evaluated
- Would like to start an activity to propose a common approach at least for the persistency framework projects
 - Closely coupled to a possible emerging LCG activity of deployment of heterogeneous databases



Collections & Analysis



- Proposed earlier a joined Work Package with ARDA
 - Not a priority for ARDA right now
 - Will continue work to address the outstanding issues on the POOL side
 - Event Collection Workshop likely mid-May
 - Have asked experiments for principal analysis contacts
- Provision of End-User Collection Interface (eg AIDA Tuple) (Q2)
 - Should start now - may move to PI or ARDA later
- Collection catalog(not POOL!), extraction and publishing tools
 - Can we achieve a single model for distributed meta data catalogs?
 - File Catalog, Collection Catalog, Conditions Folder Catalog
 - One basic mechanism of data exchange across RDBMS vendor boundaries based on the POOL relation abstraction layer?
- Separation of logical and physical collection identification (Q3)
 - Introduce a Collection (Fileset?) catalog
 - First implementation could simply be based on the existing File Catalog components, but as a separate service
- Integrate POOL collections with ARDA provided services
 - Migrate to ARDA provided catalog and meta data(?) services



RDBMS use in the Grid



- POOL will implement the application side infrastructure to use Relational Databases for Catalogs and Meta Data
- Several deployment issues still open in this area
 - Eg Grid wide access to RDBMS data
 - Eg Transaction and Distribution models
- Expect either ARDA/EGEE or a specific distributed database activity to give guidance to POOL
 - We provide a simple model based on exchange of XML meta data fragments as a candidate



POOL Infrastructure



- SEAL Component Model (Q2/Q3?)
 - Once picked up by the experiments
- Parallel build and test machinery (Q2)
 - Including effective build on windows
- Automated data format regression testing (Q1)
 - Incorporation of experiment defined test suites into the POOL release test procedure (Q2)
- Evaluate Appworks and possibly migrate to it (Q1)
 - Depending on the future of SCRAM support
- Complete move to QmTest (Q1)
- Next Ports
 - ICC 8, ECC, MacOSX
 - Following schedule to be defined by the Architect Forum



Conditions Database



Still being discussed with the developers involved

- Interface review and implementation for Interval-of-Validity (IOV) service (Q2)
 - Need commitment from experiments (ATLAS) to support consolidation in LCG
- Demonstrate connection of IOV to POOL Data (Q2)
 - Work has started already, but some pending configuration issues (compilers / platforms)
- Review of extensions to the basic interface (Q3)
- Condition Folder Cataloging review (Q3)



2004 Level 2 Milestones



- *POOL RDBMS abstraction layer completed (31 May)*
- *RDBMS independency achieved for POOL relational components (30 June)*
- *Common interface for Conditions DB defined (30 June)*
- *First release of the POOL Relational Storage Manager (31 August)*
- *Condition DB production release (31 July) assuming manpower commitment from experiments*
- *POOL meets scalability requirements (31 October)*
- *POOL integrates ROOT 4 (31 October)*



POOL Development vs. Support vs Maintenance



- See significant shift of focus
 - Many developers participate now significantly
 - into experiment integration
 - into the setup of back end services (RLS, RDBMS)
 - Significant load for support, performance studies in the data challenges
 - In my opinion this is required to develop deployable software!
 - Don't expect a change in the first half of the year
- Available manpower for new development is decreasing
 - Need to be conservative about new developments
 - ~5 FTE out of 10.6 FTE listed for POOL seems ok
 - Need to plan for longer term maintenance of POOL
 - Need to identify people who can maintain the code in the longer term and may need redistribute responsibilities



POOL Tasks & Manpower



Work Item	FTE/week	FTE
Automated (re)creation of POOL file catalogs	22	0.55
Consistent move and delete of files in a local file system	2	
Use Relational Abstraction Layer for production FC/Col	4	
Integrate with SEAL Component Model	10	
Additional POOL port	4	
	74	1.85
Fix for CLHEP matrix problem	2	
Remote LCG SE access	6	
Performance and Storage optimisation	12	
POOL/ROOT interactivity (Ref<T> plugin)	4	
Schema Evolution	10	
Support for automated translation of doubles to floats	2	
Follow the proposed evolution of SEAL dictionary	8	
Object storage on the RDBMS layer (Q2/Q3)	8	
On demand Dictionary loading	10	
Storage Manager scripting interface in Python	2	
ROOT v4.0 integration	10	
	32	0.8
Support the LCG-2 RLS	4	
Composite Catalog support	4	
Support for pluggable replica optimisation	4	
POOL and LCG replica manager fully interoperate	8	
Integrate POOL with ARDA provided file catalog	8	
Participate in ARDA/EGEE requirement gathering	2	
Support GLOBUS developers in their catalog integration	2	
	37	0.925
Support for Tokens in AttributeLists	1	
Provide POOL collection plug-in for ROOT	8	
Separation of logical and physical collection identification	8	
Collection Extraction and publishing tools	6	
Integrate POOL collections with ARDA provided services	8	
Collection end user interface (eg AIDATuple)	6	
	38	0.95
Review documentation structure	4	
Complete automated data format regression test	4	
Incorporation of experiment defined test suites	10	
Evaluate/Move to AppWorks	8	
Complete move to QmTest	4	
Parallel build and test machinery	6	
Extend POOL online tutorial	2	
	203	5.1

Storage Manager	M. Frank, G. Govi, I. Papadopoulos
File Catalog	C. Cioffi, M. Girone, Z. Xie
Collections & Meta Data	J. Hrivnac, K. Karr, D. Malon, K. Moscicki, H. Schmuecker, S. Vaniachine
Infrastructure & Testing	T. Barrass, R. Chytracek, M. Girone, G. Govi, I. Papadopoulos
Conditions DB	A. Valassi ??

Available people ~ 10 FTE

Development effort ~ 5 FTE

Leaves 5 FTE for support, integration, deployment preparations

- problem to involve POOL people with low (<50% FTE) contribution in support
- Need definitive commitments for ConditionsDB sub-project



Summary



- **Main Development Goals for 2004**
 - Consolidation and Optimisation
 - RDBMS vendor independence
 - Common model for distributed, heterogeneous meta data catalogs (incl. data exchange across vendor boundaries)
 - ConditionsDB production release and integration with POOL
- **Non-development Goal**
 - Support three successful data challenges based on POOL
 - Essential that developers are involved in support and service definition
- **POOL work plan 2004**
 - Shift of developer focus affects available development effort
 - Planned developments seem to matches available effort
- **Need to define a longer term maintenance model soon**