First Public Pool Release
V0.3

Dirk Duellmann

Release Platform

- POOL V0.3 has been released on December 18th (today!)
  - the cvs release tag is "POOL_0_3_0"
  - Public release targeting expert developers expecting to use POOL directly (eg experiment framework developers)
- Upgraded to new supported platform (still only one so far)
  - RedHat 7.3 using gcc 3.2
- Require several external packages hosted in /afs/cern.ch/sw/lcg/external
  - MySQL (4.0.3-beta)
  - MySQL++ (1.7.9)
  - ROOT (3.04/01 – last week's release)
    - plan to upgrade asap to the 3.04/02 Root released today
  - Xerces-C (2.1.0)
- Support the LCG Apps build system
  - SCRAM – Many thanks to Torre!
    - now only final adjustments (eg #include policy) required

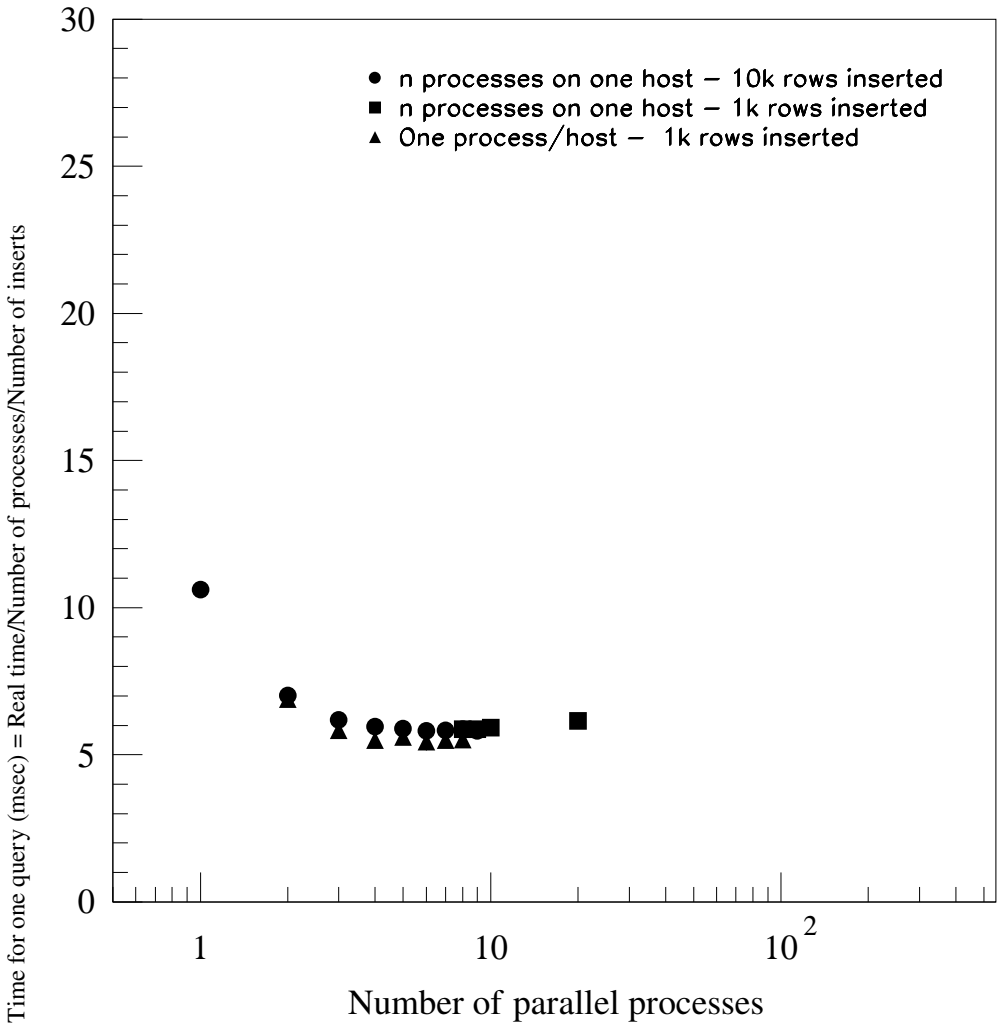
Release Content

- Collections
  - Implicit and explicit collections
    (based on containment in root files or list of object references)
  - Provides coherent iterator access
  - Component implementation in C++ based on MySQL
- Optional Indicum package provided in pool contrib
  - Alternative Meta Data implementation by Julius based on JDO
- Merge between two (very similar) approaches for the Meta Data interface still in progress
  - need to converge soon
- Type safe end user interface on top of the low level component interface needed

File Catalog

- Three alternative implementation to cover different use cases
  - XML (local file)
  - Native MySQL
  - EDG Replica Location Service (RLS) – new in this release
- IFileCatalog – abstract interface to all three
  - Basic registration & lookup
  - Catalog iteration (eg for browsing applications)
  - First GUI prototype based on python in the pool/contrib area
- Repeated performance tests against EDG catalog (Maria)
  - several implementation options for application server hosting the EDG RLS
  - conclusion: acceptable performance (~5 ms/catalog update)

EDG Catalog Performance Tests



Storage Manager & Cache

- Completed integration between smart pointer class Ref<T> and storage manager backend.
- Refs in persistent objects now work transparently without additional client code.
- Required coupling between Collections component and basic object persistency is minimal.
- New example code is much simpler and easier to understand than in earlier releases.

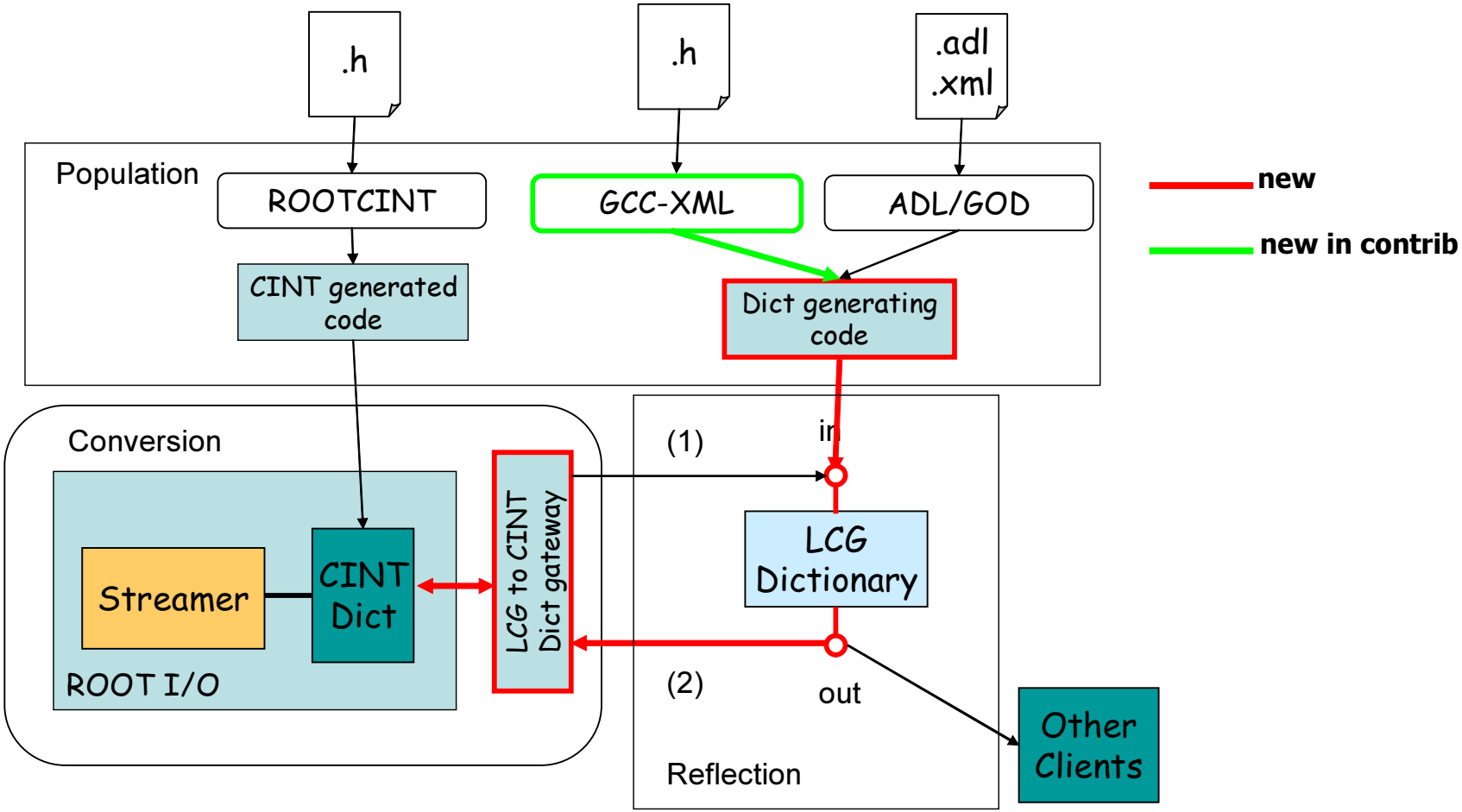
Reflection & Conversion

- Reflection package (aka LCG Dictionary) basically unchanged
  - will move to SEAL project after this release
- Dictionary Gateway to Root Dictionary
  - Proposal of Markus, Stefan & Rene defined and first implemented during this short release cycle!
  - Gateway is uni-directional in this release
    - LCG Dict -> Root Dict
  - Still quite a few limitations
    - only supports conversion of classes with elementary types as attributes, no arrays, no vectors, no strings, no namespaces
    - just populating the LCG dictionary is not the end of the story!
  - But enabling direct object persistency without any base class or need to create a technology dependent (aka Root) dictionary
- This is significant progress in very short time!

GCC-XML based Generator for LCG Dictionaries

- Proof of concept prototype in the pool contrib area developed by Zhen Xie
  - Takes xml output from a modified gcc front-end and generates the necessary stub code and calls to the LCG dictionary
  - Allows to parse the well known CMS class PSimHit
    - several templates, non-trivial inheritance hierarchy
- More work required in conversion work package to make sure that a generated dictionary actually works
  - Dictionary generation and dictionary use for conversion are now in two LCG projects
  - Need to keep good communication between the developers involved
- Need to define further how a valid LCG Dictionary should look like: remove ambiguities and limitations throughout the complete chain
  - extraction, generated code, pool gateway, root back-end

Dictionary: Reflection / Population / Conversion



Release Target

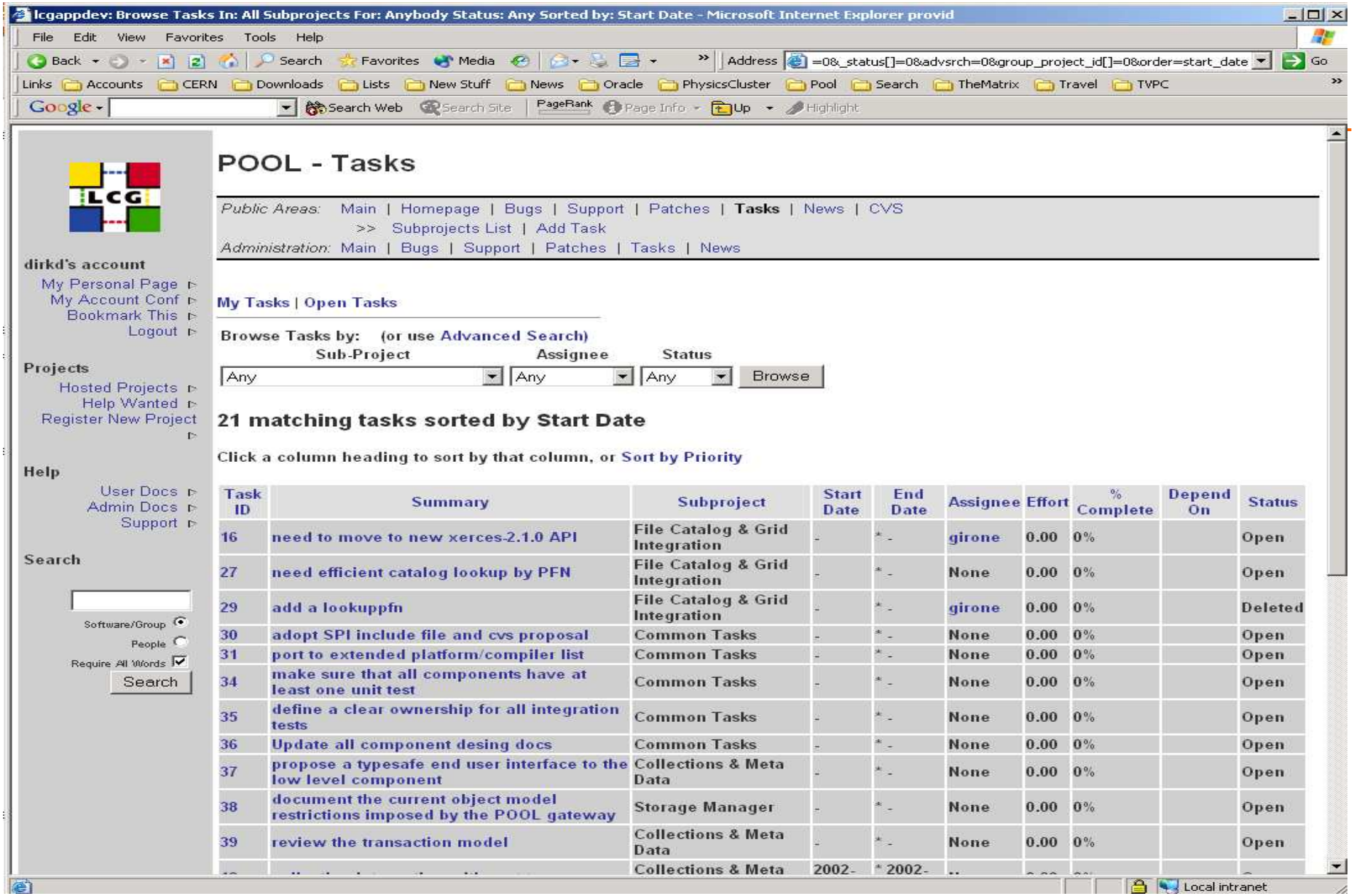
- Main Release Target:
  - Add meta data (key/value pairs for now) to already existing collections
  - Connect LCG Dictionary with the Root Dictionary
- POOL functionality as of V0.3
  - Provide persistency for classes not deriving from TObject (but also for Root classes), navigation between persistent objects, provide collections for them and allow to attach meta data to these collections
- Example Programs – eg /pool/examples/Collection
  - register a new file with a file catalog
  - write a few interrelated objects into root files
  - keep track of object references in a collection
  - attach simple meta data to each collection entry
  - read objects back based on collection iterators
  - allow for selection based on collection meta data

Other developments

- Much better defined release procedure as the pool developers got familiar with CVS
  - Ioannis defined the procedure and acted as Release Coordinator during this release cycle
  - Tagged component release are now happening
  - Testing before a component release is still poor (also because of still incomplete adaptation to new build system)
- Oval and CppUnits based testing has been introduced in some work packages
  - but are not yet widely used
- Savannah portal is used – very positive feedback so far!
  - Propose to use at least bug request, support request and task list for this POOL release

Next release

- Aiming for a V0.4 release at the end of February
- Need a longer cycle to get the component documentation up-to date and checked during the code review
  - this step was skipped because of lack of time during the last cycle
- Converge on the CVS structure and include policy which is currently being defined.
- Many more tasks listed on the POOL Savannah page waiting to be prioritised and scheduled for one of the next releases.


lcgappdev: Browse Tasks In: All Subprojects For: Anybody Status: Any Sorted by: Start Date - Microsoft Internet Explorer provid
File Edit View Favorites Tools Help
Back Search Favorites Media Address =0&_status[]=0&advsrch=0&group_project_id[]=0&order=start_date Go
Links Accounts CERN Downloads Lists New Stuff News Oracle PhysicsCluster Pool Search TheMatrix Travel TVPC
Google Search Web Search Site PageRank Page Info Up Highlight
LCG
dirkd's account
My Personal Page
My Account Conf
Bookmark This
Logout
Projects
Hosted Projects
Help Wanted
Register New Project
Help
User Docs
Admin Docs
Support
Search
Software/Group
People
Require All Words
Search
POOL - Tasks
Public Areas: Main | Homepage | Bugs | Support | Patches | Tasks | News | CVS
>> Subprojects List | Add Task
Administration: Main | Bugs | Support | Patches | Tasks | News
My Tasks | Open Tasks
Browse Tasks by: (or use Advanced Search)
Sub-Project Assignee Status
Any Any Any Browse
21 matching tasks sorted by Start Date
Click a column heading to sort by that column, or Sort by Priority
Task ID | Summary | Subproject | Start Date | End Date | Assignee | Effort | % Complete | Depend On | Status
16 | need to move to new xerces-2.1.0 API | File Catalog & Grid Integration | - | * - | girone | 0.00 | 0% | | Open
27 | need efficient catalog lookup by PFN | File Catalog & Grid Integration | - | * - | None | 0.00 | 0% | | Open
29 | add a lookuppfn | File Catalog & Grid Integration | - | * - | girone | 0.00 | 0% | | Deleted
30 | adopt SPI include file and cvs proposal | Common Tasks | - | * - | None | 0.00 | 0% | | Open
31 | port to extended platform/compiler list | Common Tasks | - | * - | None | 0.00 | 0% | | Open
34 | make sure that all components have at least one unit test | Common Tasks | - | * - | None | 0.00 | 0% | | Open
35 | define a clear ownership for all integration tests | Common Tasks | - | * - | None | 0.00 | 0% | | Open
36 | Update all component desing docs | Common Tasks | - | * - | None | 0.00 | 0% | | Open
37 | propose a typesafe end user interface to the low level component | Collections & Meta Data | - | * - | None | 0.00 | 0% | | Open
38 | document the current object model restrictions imposed by the POOL gateway | Storage Manager | - | * - | None | 0.00 | 0% | | Open
39 | review the transaction model | Collections & Meta Data | - | * - | None | 0.00 | 0% | | Open
Collections & Meta 2002- * 2002-
Local intranet

SCRAM release area created in AFS for POOL (and later other Projects)

- No need to recompile POOL component libraries as in the developer releases so far

```
$ scram project POOL POOL_0_3_0
$ cd POOL_0_3_0/src
$ cvs co -d examples -r examples-3-0-1 pool/examples
$ cd examples
$ scram b
$ cd Simple
$ eval `scram runtime -sh`
$ PopulateSimple
$ ReadSimple
```

- Detailed example description will soon appear on the POOL web

Documentation

- Juerg is working on a work book style document which will serve as "Getting Started with POOL"
  - Collection of several individual HowTo docs produced by the work packages
  - How to setup POOL against the shared SCRAM release area
  - How to run the example code
  - How to setup MySQL backend databases
  - Assemble FAQ listing common pitfalls
- First release expected before Christmas

Summary

- This was our shortest (and most painful) release cycle so far
  - because of Christmas/CHEP'03 constraints
  - Currently we can't sustain short (~1 month) cycles without impacting the development of larger features and the quality of design document documentation, testing and code
- The first public POOL release took place with 3 days delay
  - Planned functionality is provided
    - Still many limitations and probably instabilities throughout the all components
    - After only 3 releases this is unavoidable
  - Still, sufficient to ask for feedback from potential users
- Please use our savannah page http://lcgappdev.cern.ch/savannah/projects/pool/
  - For support requests and bug reports
    - make sure your question/report is new
  - Please check our task list to see where and when your requests could fit in
- POOL project has met it's 2003 work plan with only minor delay
  - Many thanks to all POOL developers …
    …and SPI infrastructure supporters and the ROOT team …!
  - We are looking forward to profit from SEAL developments in our next release.

Next Steps