BaBar Databases

Jacek Becla

Stanford Linear Accelerator Center

For the BaBar Collaboration Meeting, London, Sep'02

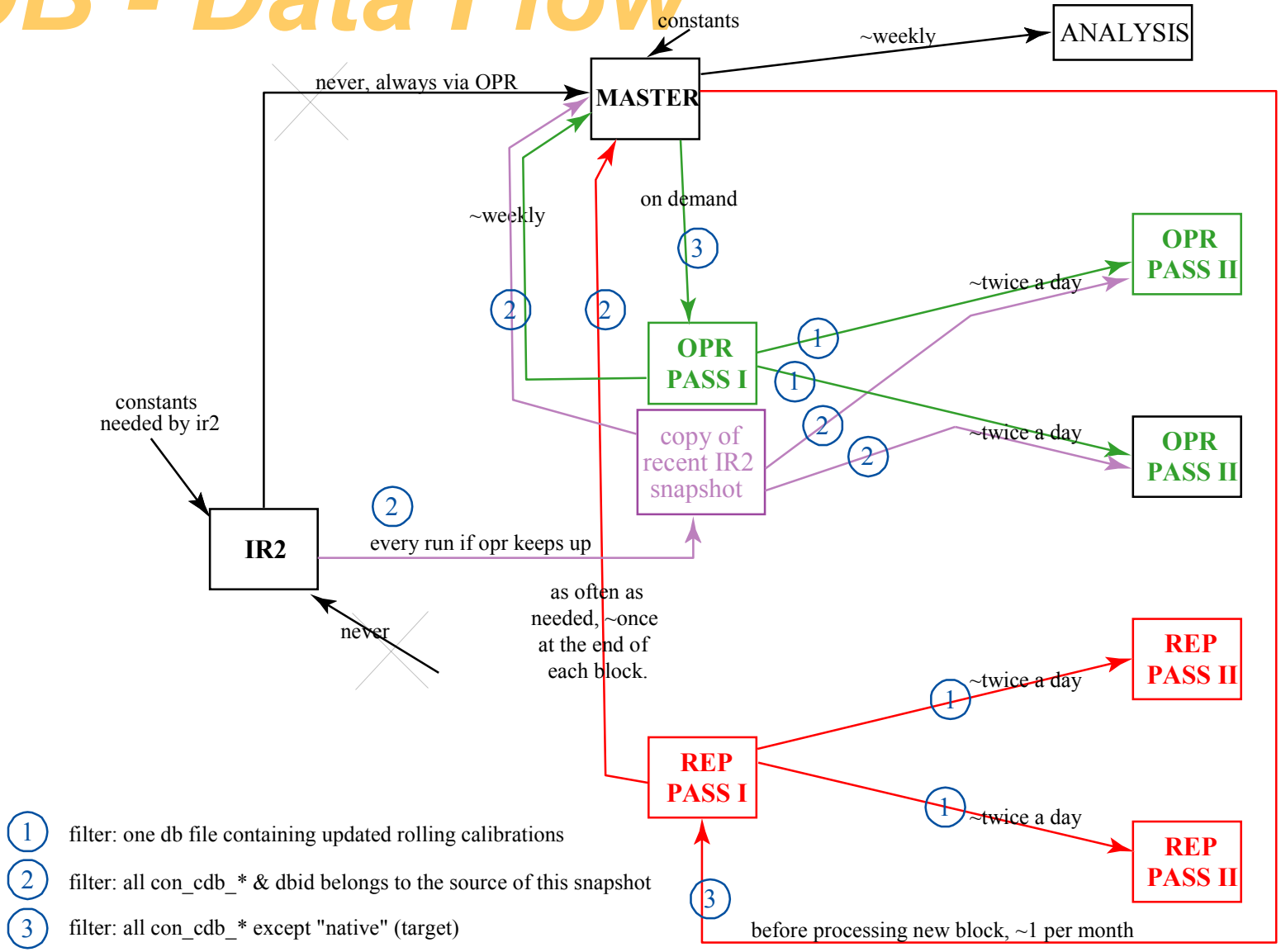
Outline

- Recent activities
- Current development
- New ideas
- Summary

Condition DB Redesign

- Implementation finished
  - First round of tests done
- Deploying now
  - Full production cycle exercised now
  - Working on CDB data distribution tools
  - Conds in new CDB format available ~ mid Sep'02
  - Switching to CDB
    - 12.3.1 and up needed to access new CDB
    - Backwards compatible
      - Old code will access cond produced before conversion to new CDB
    - transparent to applications

CDB - Data Flow



CDB - cont

- What next?
  - It's a whole new culture, need to learn how to use its full potential
  - Few more months of development needed to further strengthen CDB
    - Incremental data sweeps
    - Extracting subset (deep copy)
    - GUI
    - Authorization
- Likely to be a very good product ☺

Load Balancing

- AMS automatically balances load on data servers
  - Fully transparent to users
    - Introduction in production
    - Adding servers / taking offline
- Will improve
  - Server response time
  - Robustness of analysis system

Load Balancing - Status

- Finished, tested
- Integrating with data distribution tools now
- About to start deployment
  - @SLAC and IN2P3
  - Fully transparent to users
  - Deadline: 3rd Q'02

Client-side Decompression

- Available
  - Comes with objy 7.1
  - Tried on Solaris, need to port to Linux
- Will put in production by Dec'02
  - Files compressed when brought to disk
    - Uncompressed format stored in HPSS at SLAC
      - For safety (potential bugs in compression libraries)
      - Hardware compression
  - Transparent
    - 12.3.1 and up will be able to decompress on client side
    - All others - AMS will do decompression
  - Currently requires AMS
    - Direct access will be supported shortly

Effects of Compression

- Pros
  - Average client elapsed time (simple tests made)
    - 1 client: ~20% slower
    - 25-75 clients: ~2-4% faster
  - Disk usage, disk & network I/O
    - Micro: ~60% smaller
    - Mini: ~35% smaller
  - AMS CPU usage
    - 6-20% lower, depending on load
- Con
  - Need CPU cycles to compress
    - I/O bound, ~5-11 MB/sec
    - 10 days --> 3 TB --> ~80 hours

Event Store Redesign

- Work in progress
  - Discussed with other BaBar experts
  - About to finish the new design
    - Expected reduction in size for navigational components (evt, evshdr, col) 70-80%
    - Detailed doc available on Bdb webpage
  - Implementation
    - Have enough manpower to finish quickly
    - Target: core features by end of ‘02 in release 13
- Work on AOD (micro) not started
  - Still waiting for the visitor from Novosibirsk

Miscellaneous

- Padova effort 
  - Continues, <30% problems Bdb related
- Received Objy 7.1
  - Testing now. If ok, will integrate with 12.x.y.
  - Next major release 1stQ‘03

Analysis @ Tier "D"
… or Bdb/Lite

- How easy/hard is to run Objy-based analysis on a laptop?
  - How about if no network access?
- Proof-of-concept done at SLAC
  - In < 2 weeks in between other tasks
  - Both micro and mini

Complexity – Setting Up

Installing release

- most difficult and time consuming

Installing objy

- literally trivial
  - 5 min to transfer file
  - 4 min to install
- No servers (AMS, lockserver) required

Importing conditions

- 30 min to transfer
- Few min to attach
- Easy, but needs work due to size

Importing data

- Easy thanks to existing components (bridge fd phase III, BdbServer)
- Will greatly simplify soon (BdbServer++)
- Extraction speed 5-10 Hz
  - work in progress on speeding it up
- If already extracted, matter of minutes

All done

- ~30 K events imported
- tested

Complexity - Running

- Objy – maintenance free
  - No servers
  - No hassle with locks collisions
  - No hassle with cleaning up locks
  - “In process lock server” feature
    - Restrictions
      - Only one writer at a time
      - Multiple simultaneous readers OK
    - Need to test
- Importing new collections
  - One simple command when BdbServer++ available
- Shipping data back to central fd
  - Evaluating technical aspects

Space Requirement

1) Release
   - Full: 2GB (Geant, cernlib, Root, RW, ACE/TAO, CVS,…)
   - <1 GB if stripped down
2) Objy
   - 0.2 GB
3) Conditions
   - Full snaphot - 31 GB
     - < 6 GB if compressed
   - Extracting small subset
     - A few months of work (new project)
4) Data
   - 4.2 kB/event micro
     - < 3 if compressed
   - ~12 kB/event mini
     - ~8 if compressed
   - Significant improvements in navigational components on the way

What Needs to Be Done?

	Scheduled?	FTE-month
Tool for extraction conditions	No	4
Direct (not via AMS) decompression	No	1
Finish BdbServer++	Yes	6
Test In Process Lock Server	No	1
Speed up data extraction	Yes	2
Improve release installation procedure	No	1?
Provide docs and productize all	No	2

Analysis on Laptop - Conclusions

- Complexity
  - It is not difficult for a non-expert
    - A few guinea-pigs will be trying next week
- Space
  - With 20 GB hard drive one can analyse close to 10^6 events (mini) now
- Enthusiastic reaction from all we talked to
- Some work (not much!) needed to make it simple for an average user
  - Need to allocate resources and productize it

Computing Review (Apr'02)

- Conditions DB redesign (milestone: June'02)
- Deployment of Objy analysis at Tier-C sites (milestone: Dec'02)
- Bridge fd phase II deployment and development
- Load balancing in analysis and OPR federations
- Reducing size of the AOD in Objectivity (milestone: Dec'02)

Summary

- ☺ Deploying CDB and load balancing
- ☺ Client side decompression available, soon in production
- ☺ Good progress with event store redesign
- ☺ Running Objy analysis on laptop might be much easier than all of us think