

sPHENIX plans

Coordinated Analysis

Analysis Organization

- To first order more a sociological than a technical problem
 - “Easy solutions” will always win, users vote with their feet
 - Sticks don’t work, there has to be a tangible benefit for users to get on board

- PHENIX evolution

Internal data structure to access objects implemented from the beginning
2000/2001 Reconstruction and Analysis written in CINT macros

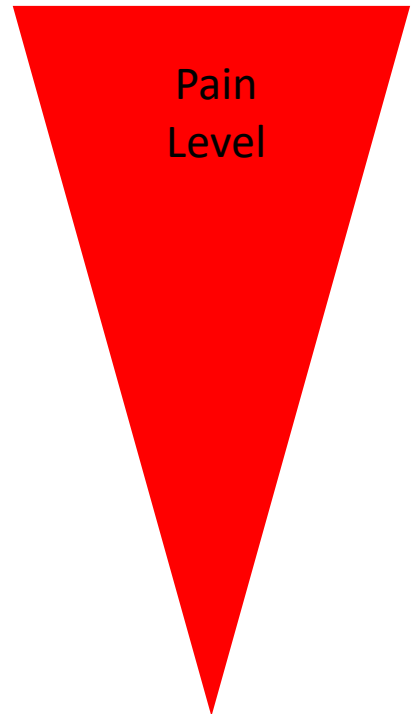
- “fast” development but impossible to debug
- 1000 line spaghetti

2002 Moved to compile only

- Finally could use a debugger
- Everything hardcoded – no flexibility
- 1000 line spaghetti

2003+ Fun4All

- Reco/Analysis Chain Configurable with CINT macros
- Baseclass to interface to “code which does something”



Keep it simple – The only class a user ever needs

What we tell users: You need to inherit from the SubsysReco Baseclass (offline/framework/fun4all/SubsysReco.h) which gives the methods which are called by Fun4All. If you don't implement all of them it's perfectly fine (the beauty of base classes)

- **Init(PHCompositeNode *topNode)** : called once when you register the module with the Fun4AllServer
- **InitRun(PHCompositeNode *topNode)** : called before the first event is analyzed and whenever data from a new run is encountered
- **process_event (PHCompositeNode *topNode)** : called for every event
- **ResetEvent(PHCompositeNode *topNode)** : called after each event is processed so you can clean up leftovers of this event in your code
- **EndRun(const int runnumber)** : called before the InitRun is called (caveat the Node tree already contains the data from the first event of the new run)
- **End(PHCompositeNode *topNode)** : Last call before we quit

I haven't seen an example yet where this is not sufficient

The Battle for the Analysis Chain

- PHENIX makes a persistent copy of the node tree
 - This DST is a snapshot of the state of the analysis/reconstruction chain
 - Node Tree structure reflected in horribly not CINT parsable TBranch names
 - It is not easily T.Draw()'able (remember “easy”?)
- Run2Tree (2002)
 - Dumps DST content into a simple TTree for “easy” analysis
 - Came with its own “analysis framework”

“Easy” doesn't cut it this time – let's go for tangible benefit

The tangible benefits

- Recalibrators
 - No persistent output is ever final, calibrations need to be applied
 - Added recalibration feature (with DB backend and automatic selection of needed calibrations)
 - Added by one line in the steering macro (remember “easy”)
- Reading files in parallel and synchronizing the content
 - We started splitting our output into calorimeter and tracking data and into central arm and muon arm data and separating triggers
 - Also small subsets of separate hits files for embedding studies
 - Fun Fact – that is the precursor of the sPHENIX offline event builder
 - Root TTrees from Run2Tree by themselves neatly mix your events without telling you

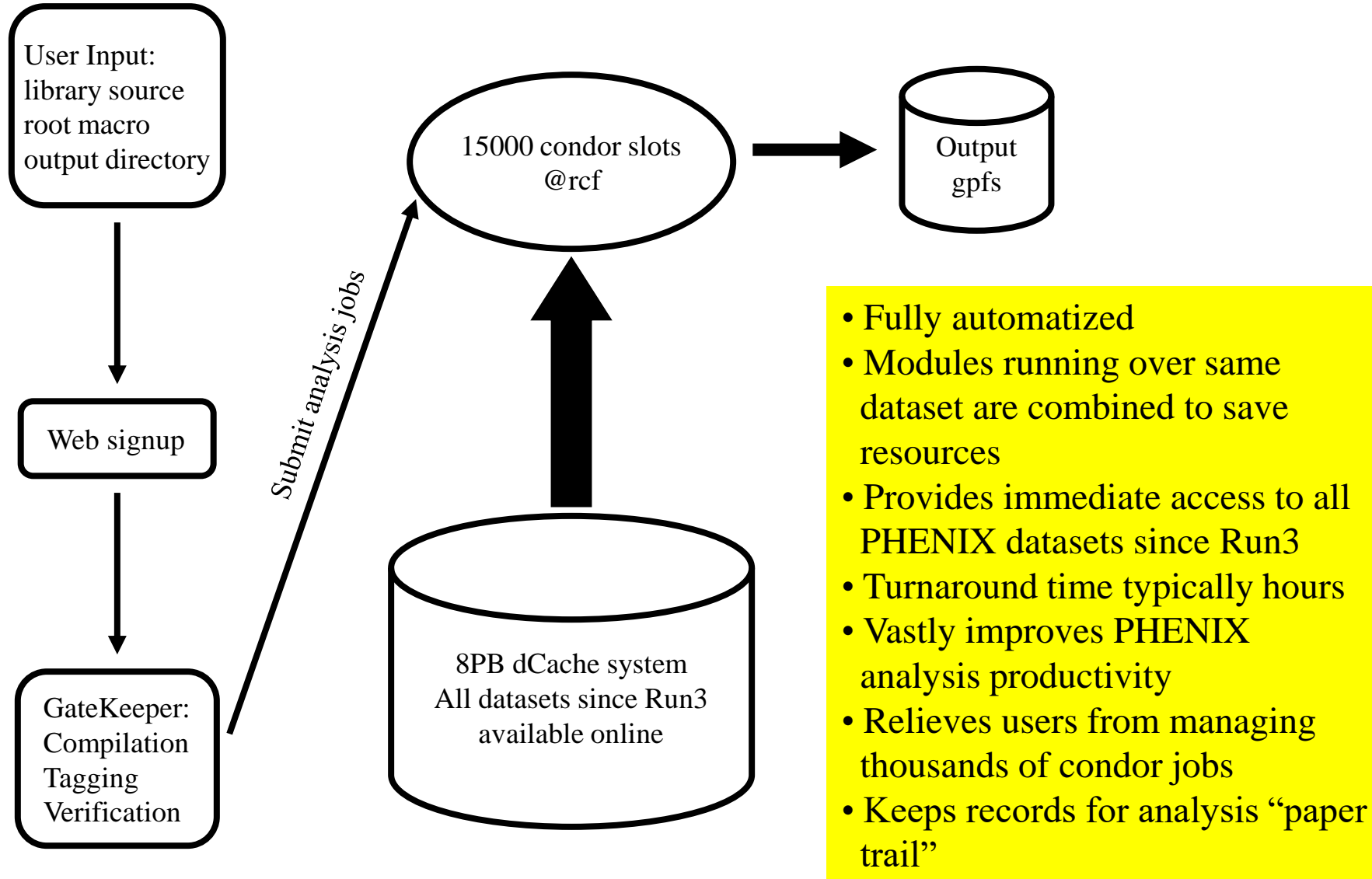
Now all PHENIX analysis code derives from the common base class
That code typically writes Ttrees, ntuples, histograms for easy analysis

Brief History of the Analysis Train/Taxi

- 2004: First incarnation - DST's did not fit on central disk storage
 - Tried subset on disk and replace occasionally, not much space left for users
 - Organized fetching of files from hps and process locally (freeing up central disk space), use common interface to call user code and call this analysis train
- Late 2004: dCache – giving access to previously unused disk space on compute nodes with enough space to keep DSTs
 - Network now bottle neck (fast ethernet – 100Mb/s)
 - Weekly runs for as many modules as were signed up
- 2012: making use of 1Gb/s connectivity
 - Switched to on demand submission for single users → Analysis Taxi (“easy”)
 - Simultaneous requests for the same dataset still get combined
 - Local disk i/o now limiting factor – redesign DSTs to trade off space for cpu
- 2023: sPHENIX Probably starting with tape again

No user interface was changed, all datasets since 2003 are available for Analysis

Analysi taxi Schematics



Signup (easy for them) and Monitoring (easy for us)

Registration form for Anatraining modules. Fields include Name, E-mail, Macro name, Analysis Code Directories, PWG Disk (radio buttons for hhj, plhf, spin, mpcecx), and User Name.

Run 3
dAu 200 GeV

- Run-3 200 GeV d+Au pro48 (Electron) 109,261,656+? Events DSTs: CNT, PWG
- Run-3 200 GeV d+Au pro48 (MinBias) 104,479,320+? Events DSTs: CNT, PWG
- Run-3 200 GeV d+Au pro48 (Muon) 111,526,772+? Events DSTs: CNT, PWG
- Run-3 200 GeV d+Au pro48 (Photon) 60,819,272+? Events DSTs: CNT, PWG

Run 4

408 datasets

STANDARD RESULTS

- Run-16 200 GeV d+Au pro107 (ERT) 19,973,518 Events DSTs: MWG, CNT, DST_EVE
- Run-16 200 GeV d+Au pro107 (FVTX) 3,513,743 Events DSTs: MWG, CNT, DST_EVE
- Run-16 200 GeV d+Au pro107 (MinBias) 1,465,752,780 Events DSTs: MWG, CNT, DST_EVE
- Run-16 200 GeV d+Au pro107 (MPC) 239,150,672 Events DSTs: MWG, CNT, DST_EVE
- Run-16 200 GeV d+Au pro107 (Muons) 34,101,363 Events DSTs: MWG, CNT, DST_EVE
- Run-16 200 GeV d+Au pro107 (Others) 31,383,188 Events DSTs: MWG, CNT, DST_EVE

DST Types:

- CNT
- CWG
- DST_EVE
- DST_FCAL
- DST_FVTX
- DST_HBD
- DST_MPC
- DST_MPCEX
- DST_RXNP
- DST_SVX
- DST_TEC
- EWG
- LOCALPOL
- MuonDST
- MWG
- PWG

Aggregation? Yes No

Event Mixing (Embedding)? Yes No

Note that not all data sets after run9 are available for embedding. The datasets for which embedding works have "Embedding works" note alongside them. If you have questions, send an e-mail to Sasha Lebedev (lebedev@iastate.edu).

Runnumber List (enter runnumbers separated by commas). Leave blank to use all runs in dataset.

Estimated Disk Space: _____

Additional Requests: _____

Submit Form Clear Form

Anatraining module status

Show finished jobs

Show entries Search:

Job#	Username	Module	Macro	Dataset	Status
17498	cwong14	HadronCorrelation_GSU	Run_HadronCorrelation_GSU_pi_run10_C0.C	Run10AuAu200MinBias	Running
17497	cwong14	HadronCorrelation_GSU	Run_HadronCorrelation_GSU_pi_run11_C1.C	Run11AuAu200MinBiasPro101	Running
17496	cwong14	HadronCorrelation_GSU	Run_HadronCorrelation_GSU_pi_run11_C0.C	Run11AuAu200MinBiasPro101	Running
17494	mrpind	mrpind_mpcecxAna	Run_MpcecxAna_mrpind.C	Run16dAu200MPCexMBP110	Waiting
17491	pinkenbu	theta/src	Run_ChpTestTaxi.C	Run16AuAu200CAMP111	Running
17487	gelato	Run16AuAuPhoton	Run_photonCorr.C	Run16AuAu200CANVXMBP108	Running
17478	jfrantz	combinesimpleAbi	Run_combinedsimpleAbi_MB_run14AuAu_85_90.C	Run14AuAu200CAMP109	Waiting
17472	jfrantz	combinesimpleAbi	Run_combinedsimpleAbi_MB_run14AuAu_50_55.C	Run14AuAu200CAMP109	Waiting
17410	mrpind	mrpind_mpcecxAna	Run_MpcecxAna_mrpind.C	Run16dAu200MPCexMBP110	Waiting
17352	ajeeta	picodST_object	Run_picodstobject_run15pp_fvtxtracklet.C	Run15pp200CAMUP108	Waiting

Showing 1 to 10 of 11 entries Previous Next

Most Visited Getting Started

Jobnum	Running	Not Running	Comp and Good	Comp and Bad
17487	2	0	8419	0
17491	1	0	68	0
17496	5082	0	1054	0
17497	1805	0	4332	0
17498	3	0	0	0

17498 analysis modules run since inception in 2004

The tangible benefits

- Easy
 - Users do not have to deal with computing infra structure
 - Latest calibrations are applied automatically
 - Code testing (somewhat hated)
 - Fun fact – our smallest dataset is 4 files, users found this more convenient than running their own jobs
- Easy for us
 - Running centrally allows us to run things you would not hand to users
 - By now this runs completely on auto pilot

All users have access to the same data – no private data formats

sPHENIX

- Small number of datasets
 - AuAu MB datasets
 - Probably specialized sets with e.g. photon jets, upsilons
 - Trigger separated pp and pAu datasets
- Much larger datasets
 - Multi PB sized (90% reduction of 200PB raw data still 20PB)
 - PHENIX → 400TB (processed in 24 hours)

We will use the taxi idea – not the code. It's going to be an interesting challenge