Daya Bay Offline Software: A Status Report

Daya Bay Reactor Neutrino Experiment

Xinchun Tian

Institute of High Energy Physics, CAS

April 28, 2008

Outline

1 Neutrino Physics Basics

Outline

1. Neutrino Physics Basics
2. Daya Bay Experiment Basics
   - Daya Bay Detector
   - Statistics & Systematics
   - Sensitivity

Outline

1. Neutrino Physics Basics
2. Daya Bay Experiment Basics
   - Daya Bay Detector
   - Statistics & Systematics
   - Sensitivity
3. Daya Bay Offline Software
   - Road-map
   - Framework Basics
   - TES/AES Implementation
   - Data Processing Stages
   - Software organization & Installation

Neutrino Physics Basics

Mass eigenstates are different with flavor eigenstates

- Pontecorvo-Maki-Nakagawa-Sakata (PMNS) Matrix

$$\begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix} = \begin{pmatrix} U_{e1} & U_{e2} & U_{e3} \\ U_{\mu 1} & U_{\mu 2} & U_{\mu 3} \\ U_{\tau 1} & U_{\tau 2} & U_{\tau 3} \end{pmatrix} \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}$$

- The PMNS matrix can be parameterized as

$$\begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix} = \begin{pmatrix} c_{12} & s_{12} & 0 \\ -s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_{23} & s_{23} \\ 0 & -s_{23} & c_{23} \end{pmatrix} \begin{pmatrix} c_{13} & 0 & s_{13}e^{i\delta} \\ 0 & 1 & 0 \\ -s_{13}e^{-i\delta} & 0 & c_{13} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & e^{i\alpha/2} & 0 \\ 0 & 0 & e^{i\alpha/2+i\beta} \end{pmatrix} \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}$$

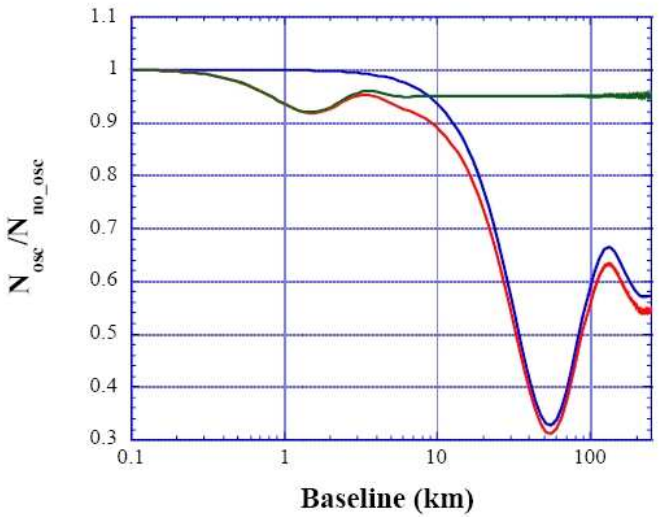
12 sector	23 sector	13 sector	mass sector
solar/reactor	atm./acce.	reactor/acce.	$0\nu\beta\beta$

θ_{13} is the gateway of CP violation in lepton sector!

Neutrino Physics Basics

Reactor antineutrino disappearance probability

$$P_{ee} \approx 1 - \sin^2 2\theta_{13} \sin^2 \left(\frac{\Delta m_{31} L}{4E_\nu} \right) - \cos^4 \theta_{13} \sin^2 2\theta_{12} \sin^2 \left(\frac{\Delta m_{21} L}{4E_\nu} \right)$$



$\bar{\nu}_e$ disappearance at short baseline (∼2km): unambiguous measurement of θ_{13}

Detection of $\bar{\nu}_e$

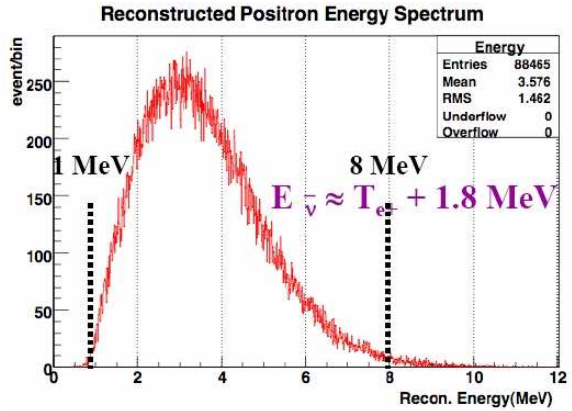
Inverse β Decay

$\bar{\nu}_e + p \rightarrow e^+ + n$

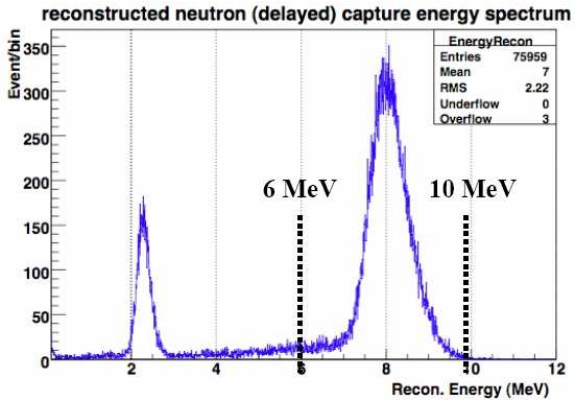
$| \rightarrow +p \rightarrow$ D+γ (2.2 MeV) $t \sim 180\mu s$ 0.3 b

$| \rightarrow +$Gd$\rightarrow$Gd* $\rightarrow$Gd+γs (8 MeV) $t \sim 30\mu s$ 50,000 b

Basic Distribution



(a) Prompt Energy Signal

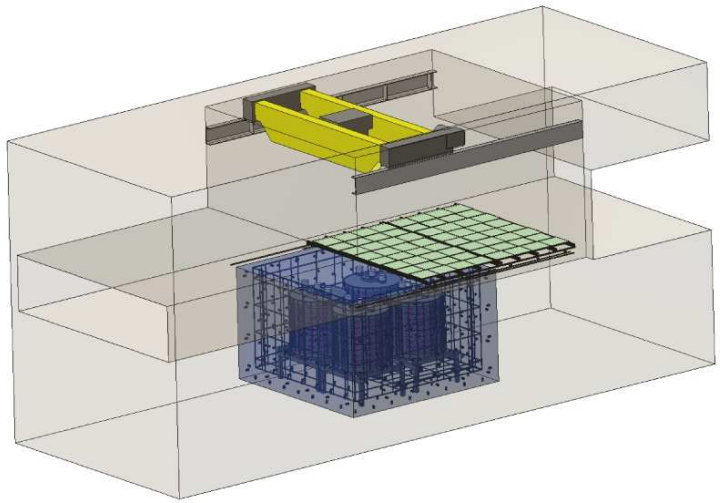


(b) Delayed Energy Signal

Time, space and energy -tagged signal $\Rightarrow$ suppress background events

Daya Bay Detector

- Anti-neutrino Detector
- Veto Muon System
  - RPC
  - Water Cerenkov



Statistics & Systematics

Goal of Daya Bay Experiment

$\sin^2 2\theta_{13} < 0.01$ @ 90% C.L.

Statistics

Near Site	Far Site
0.05%	$\leq 0.24\%$

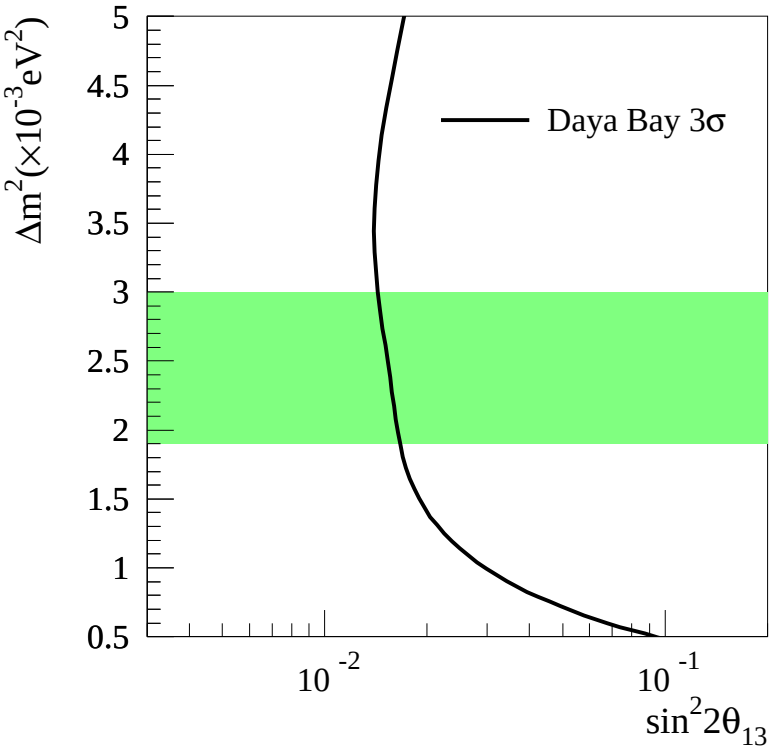
Systematics

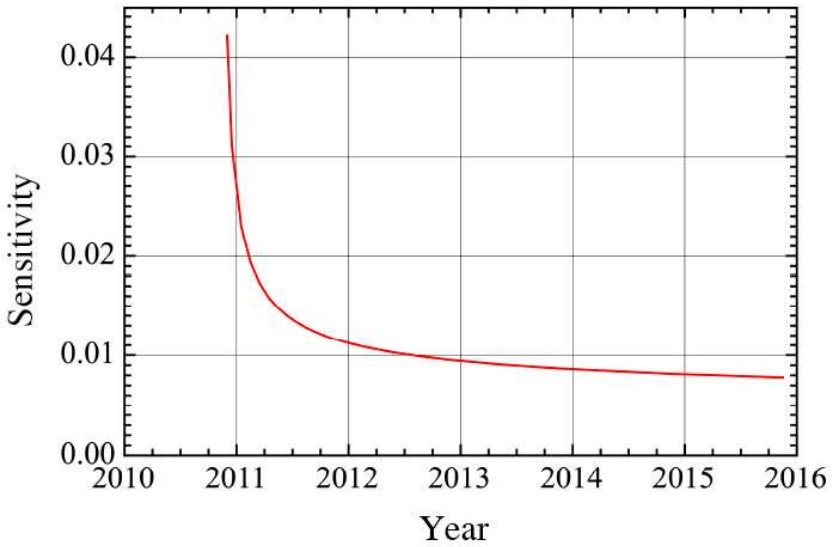
Source of Uncertainty	Near Site	Far Site
Reactor Related	$\leq$0.2%	
Detector Related	$\leq$0.38%/module	
Background Related	0.3% per site	0.2%

Sensitivity

Sensitivity @ 90% C.L. with "baseline" detector uncertainties

The sensitivity will be based on the rate and spectral shape analysis





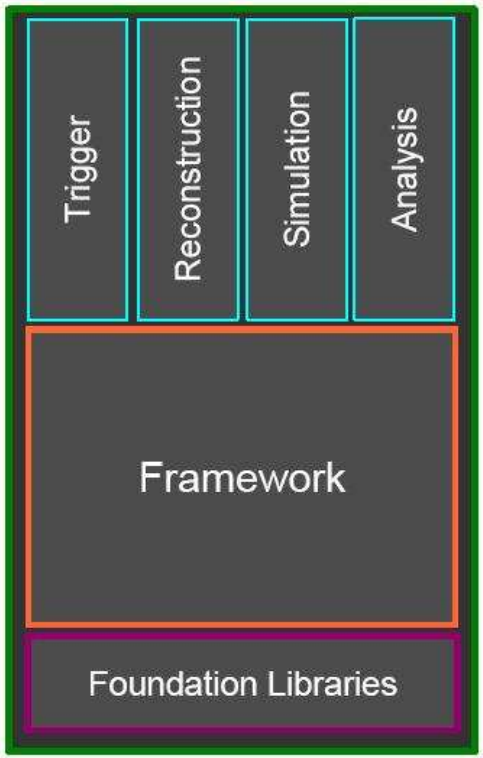
Road-map

- Gaudi as framework (developed by LHCb)
- Other LHCb projects
  - lcgcmt, lhcb
- CMT for software Management
- Subversion as software repository
- Geant4 for Simulation
- dybinst as the auto-installer
- Object-Oriented Programming
- External packages, all the 3^{rd} party support softwares
  - AIDA, CMake, HepPDT, Python, Boost, Geant4, MySQL, ROOT, CLHEP, GSL, OpenMotif, XercesC, CMake, GCCXML, HepMC, OpenScientist

Overview of our frameworks's structure

Software Organization

- Applications using framework components (Algorithms, Services, *etc.*)
- Provides basic services, common interfaces, data exchange and persistency mechanisms, interactivity
- Basic libraries (STL, ROOT, GSL, *etc.*)



Brief outline of the various features of the Gaudi framework

- Algorithm
  - user written module, execute() called once per event
  - execution sequence defined at runtime
- AlgorithmTool
  - called on demand, shared between multiple Algorithms
  - managed by ToolSvc
- Service
  - software component provided by framework, available globally
- DataObject
  - atomic data unit
- Transient Data Store
  - repository for DataObjects, used by Algs. and Tools to exchange data
  - framework manages insertion and retrieval, life cycle, load on demand
  - multiple instances: Event, Detector, Histogram
- Converter
  - conversion between transient and persistent formats
- Property
  - runtime modifiable parameters controlling behavior of Algorithms and Services

TES/AES Implementatio (LBNL)

Daya Bay TES & AES

- In Gaudi the TES is cleared before each processing cycle
  - HEP = Beam Crossing, Daya Bay = Readout
- Daya Bay needs access to previous Readout and their derived data
  - e.g., anti-neutrino is a neutron combined with a positron from previous Readout
- Daya Bay's TES has an extra dimension
  - The Archive Event Store (AES)
- Objects placed in the TES are automatically copied into the AES
- For an object in the TES, the AES has a correlated collection of objects
  - "front" object in AES collection is the "most recent"
  - TES collections are collections of collections in AES
  - Paths in the AES are the same as the TES

Data Processing Stages

- Generation & kinematics stage
- Detector simulation & hits stage (SimuAlg/DetSim)
- Electronics simulation & digits stage
- Trigger/Readout stage
- Reconstruction stage
- Physics Analysis stage

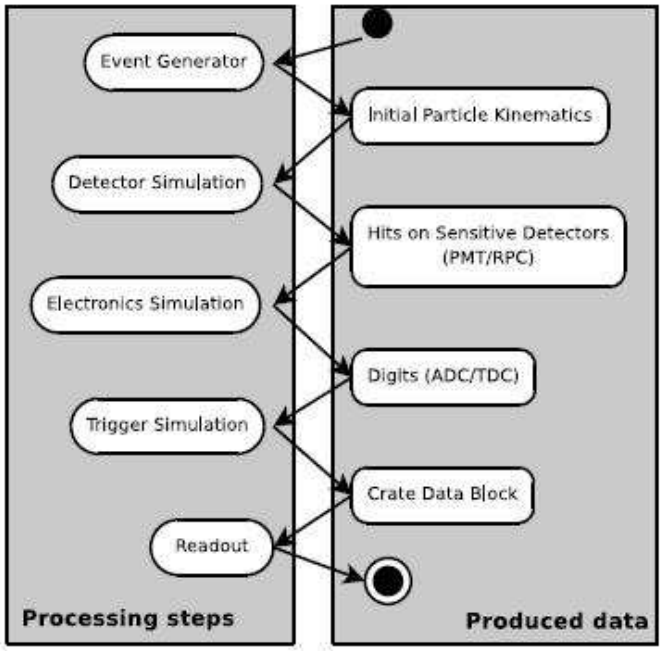


Figure: General processing steps on the left and their input and output data on the right

Generation & kinematics stage (BNL, IHEP, IIT)

GenTools

- Provides a full featured kinematic generator package
- Supports writing to Gaudi TES, Gaudi configuration mechanism

- Generation of Initial Kinematics
  - kinematics = spatial/temporal vertex, particle types and 4-momenta
  - Model all initial interactions that we don't want to leave for Geant4
- Kinematics Data
  - HepMC::GenEvent: vertex, an event number, process ID
    HepMC::GenVertex: 4-vertex, type, incoming/outgoing particles
    HepMC::GenParticle: static and dynamic particle quantities
  - Default location in TES: /Event/Gen/HepMCEvents, a simple collection of HepMC::GenEvents

Generation & kinematics stage – continue

Suite of Existing Tools

- GtHepEvtGenTool
  - Converts sources of HEPEvt formatted data
- GtGunGenTool
  - Parameterized particle generation
- GtPositionerTool
  - Generate a vertex (3-vertex)
- GtTimeratorTool
  - Set event time based on an average rate

GenTools Algorithms

- GtGenerator
  - Driving algorithm for a simple processing model (single event type push)
- GtHepMCDumper
  - Dump generator information to terminal

Detector simulation & hits stage (SDU, IHEP, BNL)

Detector simulation

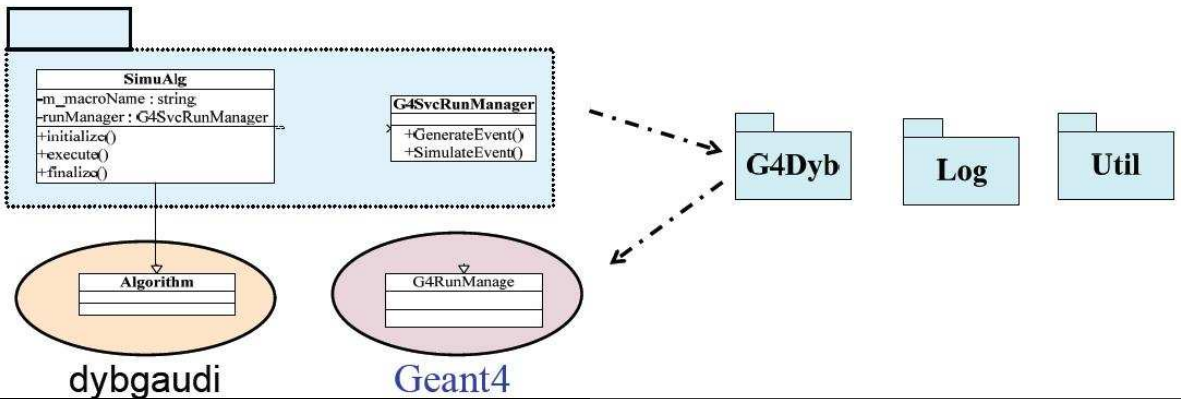
- Geant4-based G4dyb (XML & C++ based Geometry)
- Gaudi-based G4dyb (XML & C++ based Geometry)
- Gaudi-based G4dyb (GDML based Geometry)
- Gaudi-based DetSim (DetDesc based Geometry)

Hits storage

- Event Data Model
- Persistency
- Algorithm
- Converter

Gaudi-based G4dyb (XML & C++ based Geometry)

- Just for testing of Gaudi framework to give users some feeling how it works
- Keep most of the characteristics of the legacy G4dyb
  - Geometry: XML + C++
  - The interfaces to generator via pipe or reading from files unchanged
- Wrapping G4dyb as one algorithm (SimuAlg) in order for launching the simulation job in Gaudi
- Storing hits into TES and write them out into ROOT Ntuple files via THistSvc



Gaudi-based DetSim (DetDesc based Geometry)

Geometry - DetDesc

- What is DetDesc
  - C++ classes/objects for ini-memory representation
  - An XML schema for in-file representation
- Detector Description forms
  - XML Files: The source of (ideal) description
    TDS Objects: The full description as objects from Gaudi TDS
    Geant4 Geometry: TDS objects are convertible to G4 geometry objects for detector simulation
  - When necessary, an alignment DB can be built to supply offsets to TDS objects and thus to G4
- Detector Description Sections
  - Materials: The makeup of all materials
    Geometry: The full hierarchy of logical/physical volume containment
    Structure: The parallel, subset hierarchy of important Detector Elements
  - The **Surface** and **Tabproperty** ("tabulated properties") sections for defining properties

Gaudi-based DetSim (DetDesc based Geometry) – continue

Detector Simulation - DetSim

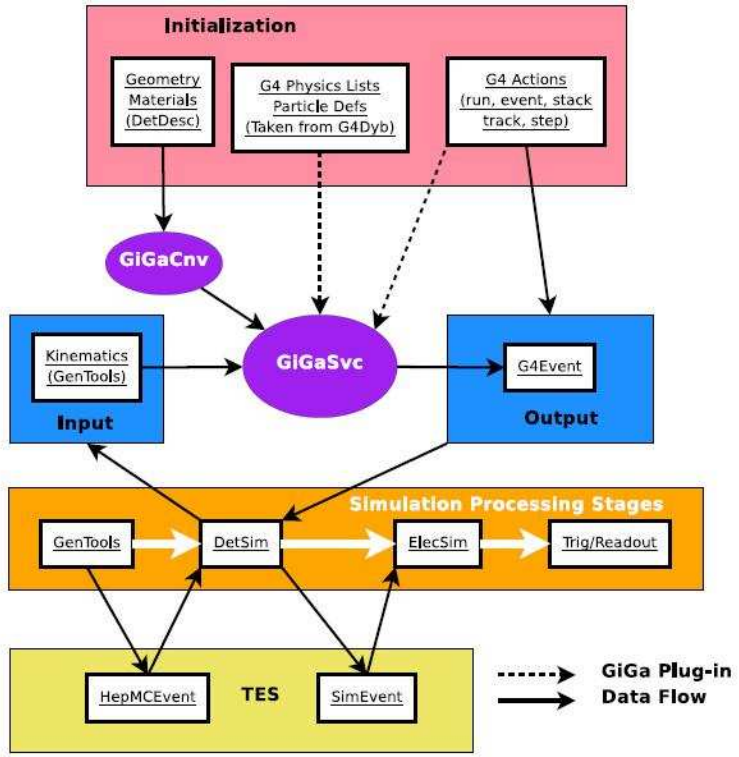
- MC integration method using G4 to track individual particles
- Runs in the Gaudi framework
- Uses the GiGa[1] package to organize Geant4 user code
- Uses the GiGaCnv package to convert detector description to G4 geometry objects
- Initial kinematics generated by the GenTools package
- Produces SimEvent objects
- Supports multiple processing models

[1] G4 Interface for Gaudi Applications or Gaudi Interface for G4 Applications

Gaudi-based DetSim (DetDesc based Geometry) – continue

Interface to Geant4

- DetDesc → G4 geometry
- PhysList classes from G4dyb
- Action classes for unobservable statistics & trajectory recording
- Kinematics in, G4 data out
- DetSim algs interface between Kinematics & GiGa/G4 and TES
- Simple linear processing model shown as example



Electronics simulation & digits stage (Caltech, LBNL, IIT)

Motivation

- Simulation can help the Electronics Design
- Simulation can help the small effects understanding
  - Pulse shaping (Effect of reflections?)
  - Pileup ($\sim$ of IBD events? Low-energy pileup?)
  - Dark hits ($\sim$ of channels?)
  - Many other effects

Status & Plan

- Electron Simulation package (ElecSim) is almost done!
- Gaudi Integration - Provide Gaudi interface to Electronics Simulation
  - Wrap each tool in a corresponding GaudiAlg
  - Allow ESObjects to be put in TES/TDS
  - Add "Pull" capability to GESSimHitReader & GESReadoutStreamer

Trigger/Readout stage (Caltech, LBNL, IIT)

Status

- Trigger Simulation
  - TrigSim figures out when the trigger conditions are met, and issues a trigger. This trigger then cross triggers all other appropriate detectors. It is also possible to explicitly issue a trigger at a user defined clock cycle (External Trigger).
- Readout Simulation
  - ReadoutSim uses the trigger information which consists of a clock cycle and trigger type to calculate a readout window which is the same for all channels. The FPGA processing then figures out which values are readout for each channel. The values can be different for each channel.

Plan

- Migrate all the code into Gaudi and make it work
- Factor out functionality of the code into a Gaudi Tool so that it can be used both in this stand-alone configuration and also in the "pull" model

AD Reconstruction (IHEP)

Status

- Two methods implemented
  - Charge Maximum Likelihood Fitting
  - Time Maximum Likelihood Fitting

Plan

- Integration with Gaudi
- Geometry from DetDesc
- Reconstruction Event Data Model

Muon Reconstruction (IHEP, BNL)

Status

- Two kinds of events (Initial value has effect on Rec.)
  - RPC & water events: initial value given by RPC
    $\sigma \sim 0.26$ m
  - water events: initial value given by PMT
    $\sigma \sim 0.48$ m
- $\sigma \sim 0.34$ m is achievable

Plan

- Trying to use better reconstruction method for type-2 muon events

Software organization

Software organization

- Package
  - Basic unit of CMT work
- Project
  - A logical unit of releasable software
- Product
  - A deliverable set of one or more projects
- Release
  - An immutable snapshot of a project or product

CMT

Configuration management Tool

- CMT is a very flexible system that organizes: source code, build process, runtime environment.

SVN (Server @IHEP, Mirror @NTU)

Subversion

SVN is a Software Configuration Management tool

- Next generation source code management system
- Acts as a shared, journaled file system

NuWa (女娲) Installations with dybinst

NuWa is the name of our offline software

- In Chinese mythology, ...
- In some lazy American English pronunciations which you will often hear around here in Long Island, could be taken to mean "newer" which is somehow fitting for our ever changing software.
- In Japanese, the phrase "Nu Wa" might mean "I'm talking about the neutrino". As in "nu wa daisuki" meaning "this is neutrino, i like".

dybinst will install

- CMT the basis of our build and runtime setup
- external packages all the 3^{rd} party support software
- LHC projects software taken from LHC effort (lcgcmt, gaudi, lhcb)
- dybgaudi the project holding our Daya Bay specific offline software

dybinst usage

- dybinst help

Major Milestones

Mile. No.	Milestone Description	WBS	Date
0	US CD-0 Approve Mission Need		Nov-05
	Chinese Funding Secured		March-07
6	CD-3a Approval & Long Lead Contract Award		March-08
27	PMT Dry Run & Offline Software Integration Challenge	1.5	18-Dec-08
45	Begin Overall System Testing - DB Near Hall		1-Sep-09
46	DB Near Hall Physics Ready		6-Oct-09
66	Begin Overall System Testing - Far Hall		8-Sep-10
67	Far Hall Physics Ready		13-Oct-10
68	US CD-4a Approval Request		Feb-11

The End

Thank you!