How to store multidimensional data

Dmitri Konstantinov

Geant Validation DB Meeting

November 18

2015

PostgreSQL

For this study I have created a sandbox:

virtual machine with postgresql 9.2

PyGreSQL - python API – to read/write data into database

- Using PostgreSQL we can keep
  a) JSON (any data can be serialized into JSON, even with axis names etc)
  b) arrays – REAL 1D_ARRAY[]
     - many dimensional data - REAL 2D_ARRAY[][]
       but only for same dimensions, i.e. m x m

1D histograms

- 1D histogram
  - xlow[m]
  - xfoci[m]
  - xhigh[m]
  - yval[m]
  - yerr_stat[m]
  - yerr_sys_plus[m]
  - yerr_sys_minus[m]

And this can be easily described by very same postgres table.

2D histograms

- 2D histogram
  - x1_low[m]
  - x1_foci[m]
  - x1_high[m]
  - x2_low[n]
  - x2_foci[n]
  - x2_high[n]
  - yval[m x n]
  - yerr_stat[m x n]
  - yerr_sys_plus[m x n]
  - yerr_sys_minus[m x n]

- 2D histogram
  - dimensions[] = { m, n}
  - x_low[m + n]
  - x_foci[m + n]
  - x_high[m + n]
  - yval[m x n]
  - yerr_stat[m x n]
  - yerr_sys_plus[m x n]
  - yerr_sys_minus[m x n]

Therefore we can easily to keep any dimensional data using Postgres arrays.

- The similar can be done using JSON data type
  - In python it can be easily read/written using standard “json” module
  - In C++ API it can be done using Boost library

Plan

- draw a DB schema which will satisfy our requirements
- fill DB prototype with all possible data “types” we have up to now for validation to see possible shortcomings.