Fermilab

Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

Using Inspire and discussing the db schema

Hans Wenzel

Geant 4 Validation repository weekly meeting

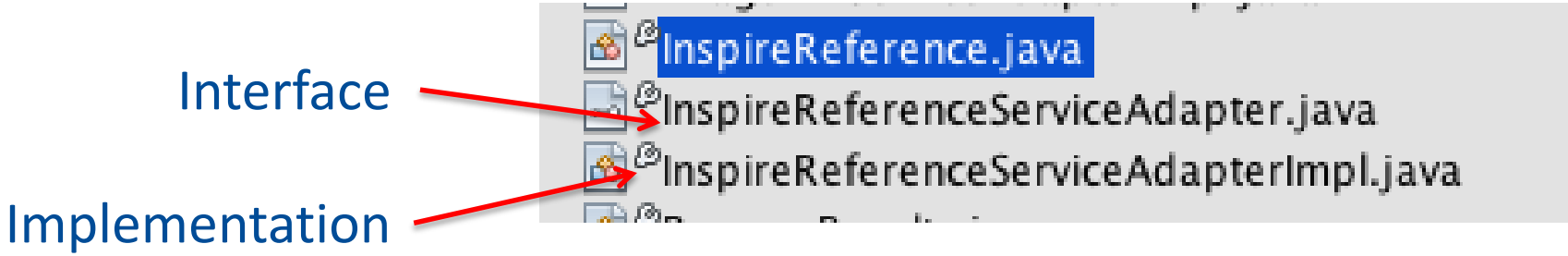
25th November 2015

Plan for utilizing Inspire

- Use to replace reference tables→ next slides
- For now don't use the experimental data available in Inspire instead we maintain experimental data used by geant 4 in our db.
  - Meta data doesn't allow to access the data in the way we prefer
  - Not complete
  - ASCII table
- But if we can identify a dataset → store reference in our data might be useful later on.
- Discuss our requirements with the Inspire team.

Integrate Inspire

- Database table stores inspire record id (assume it is permanent.)
- Extend the API:
  - Class: InspireReference((ServiceAdapter)Implementation)
    - constructor takes the record id as input to query inspire web api,
    - the response is an input stream which is encoded in xml,
    - Input stream is then parsed for title, authors etc. → data member of class
  - Caveat
    - not fast → populate at start up of WebApp
    - So far concentrated on extracting from db
  - Changes have been committed to svn (trunk)
- Use in Web application



Sql statement to create and populate the table

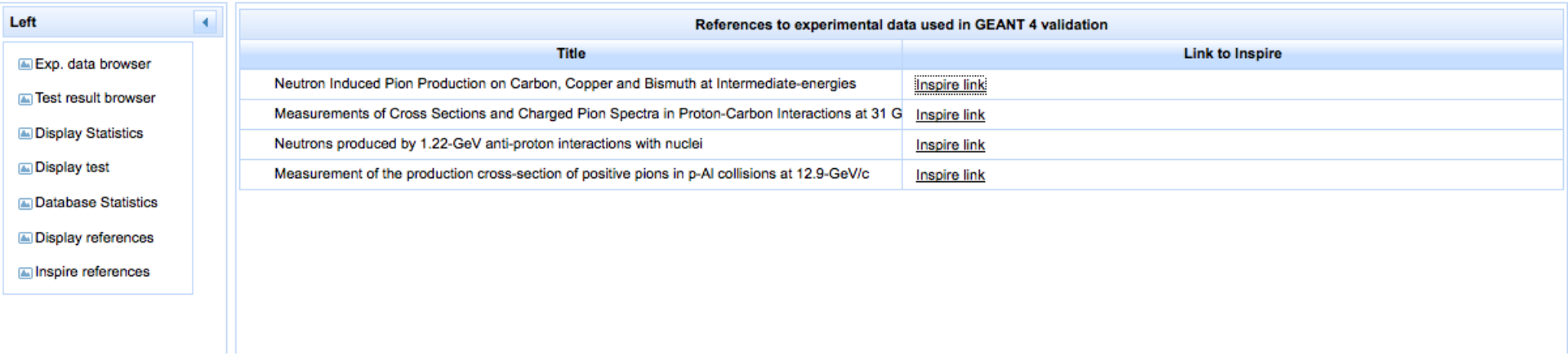
```
CREATE TABLE inspirereference(
   inspireid  Integer not null PRIMARY KEY
);
GRANT SELECT, UPDATE, INSERT, DELETE ON  inspirereference TO g4valwriter;
GRANT SELECT ON inspirereference  TO g4valreader;
INSERT INTO public.inspirereference (inspireid) values (287791);
INSERT INTO public.inspirereference (inspireid) values (886780);
INSERT INTO public.inspirereference (inspireid) values (532743);
INSERT INTO public.inspirereference (inspireid) values (695147);
```

Java Class InspireReference

```java
import java.net.URL;
import nu.xom.*;

/**
 *
 * @author wenzel
 */
public class InspireReference implements Serializable {

    private final String USER_AGENT = "Mozilla/5.0";
    private Integer recid;
    private ArrayList<String> authors;
    private String title;
    private String sectitle;
    private String ern;//electronic-resource-num
    private String pages;
    private String volume;
    private String number;
    private String year;
    private String abstr; //abstract
    private final String linkurl; //link to corresponding inspire record

    public InspireReference(Integer recordnr) throws Exception {
        recid = recordnr;
        String url = "https://inspirehep.net/record/" + recordnr + "/export/xe"; // connection to spires API.
        linkurl = "https://inspirehep.net/record/" + recordnr;
        URL obj = new URL(url);
        authors = new ArrayList();
        HttpURLConnection con = (HttpURLConnection) obj.openConnection();
        con.setRequestMethod("GET");
        //add request header
        con.setRequestProperty("User-Agent", USER_AGENT);
        int responseCode = con.getResponseCode();
        Document doc = new Builder().build(con.getInputStream());
        Element root = doc.getRootElement();
        Elements elements = doc.getRootElement().getChildElements();
        for (int i = 0; i < elements.size(); i++) {
            Elements records = elements.get(i).getChildElements();
            for (int j = 0; j < records.size(); j++) {
                Elements contributors = records.get(j).getChildElements("contributors");
```

Use in Web Application

Deployed on development server
http://g4devel.fnal.gov:8080/G4WebAppNG/



Will add: Journal year etc., tests where the reference is used, allow to display all data without leaving our site.

Schema tables we need (for now)

- "Primitive" objects:
  - 1D, 2D Histograms
  - XY point sets with systematic and statistical errors
  - Matrices (e.g. for covariance matrix)
  - Vector like (e.g. for parameter sets values and uncertainties)

- Objects describing test, test result, experiment:
  - Experiment/Test description
  - Testresult defining meta data and link to histogram, xy data set (ditto for experimental result)

Sql statement to create Histogram2D table

```
CREATE TABLE Histogram2D (
    ID SERIAL PRIMARY KEY,
    name      text,
    xdes           VARCHAR(50),
    ydes           VARCHAR(50),
    zdes           VARCHAR(50),
    nbinsx         Integer,
    minx           real,
    maxx           real,
    bincenterx     real[],
    binwidthx      real[],
    nbinsy         Integer,
    miny           real,
    maxy           real,
    bincentery     real[],
    binwidthy      real[],
    binvalues      real[][],
    errorup        real[][],
    errorlow       real[][]
);
GRANT SELECT, UPDATE, INSERT, DELETE ON  Histogram2D TO g4valwriter;
GRANT SELECT ON  Histogram2D TO g4valreader;
```

Change name to title

Sql statement to populate Histogram2D table

```
INSERT INTO public.Histogram2D
(       name,
        xdes,
        ydes,
        zdes,
        nbinsx,
        minx,
        maxx,
        bincenterx,
        binwidthx,
        nbinsy,
        miny,
        maxy,
        bincentery,
        binwidthy,
        binvalues,
        errorup,
        errorlow)

VALUES ('test',
        'x-axis',
        'y-axis',
        'sigTot[mb]',
        3,
        0,
        3,
        '{0.5,1.5,2.5}',
        '{1.,1.,1.}',
        3,
        0,
        3,
       '{0.5,1.5,2.5}',
        '{1.,1.,1.}',
        '{{341,314,290},{342,312,292},{343,313,293}}',
        '{{441,414,390},{442,412,392},{443,413,393}}',
        '{{241,214,190},{242,212,192},{243,213,193}}'
);
```

Sql statement creating xydataset

```
--
-- table representing set of x y points with statistial
-- and systematic uncertainties
--
CREATE TABLE xydataset (
    ID SERIAL PRIMARY KEY,
    title text,                         -- title of data set
    xdes character varying(50),         -- x-axis title
    ydes character varying(50),         -- y axis title
    x real[],                           -- x values of xy points
    y real[],                           -- y values of xy points
    staterrx real[],                    -- statistical error in x
    staterry real[],                    -- statistical error in y
    syserrx real[],                     -- systematic error in x
    syserry real[]                      -- systematic error in y
);
ALTER TABLE xydataset OWNER TO g4valwriter;
GRANT SELECT, UPDATE, INSERT, DELETE ON xydataset  TO g4valwriter;
GRANT SELECT ON xydataset TO g4valreader;
```

Working group dictionary sql statement to create and fill

```
--
-- table representing dictionary of geant 4 working group
--
CREATE TABLE wgroups (
    wgid serial  primary key,              -- working group id
    wgname      character varying(50)      -- name of the working group
);
ALTER TABLE wgroups OWNER TO g4valwriter;
GRANT SELECT, UPDATE, INSERT, DELETE ON wgroups  TO g4valwriter;
GRANT SELECT ON wgroups TO g4valreader;

insert into public.wgroups (wgname) values ('electromagnetic');
insert into public.wgroups (wgname) values ('hadronic');
insert into public.wgroups (wgname) values ('medical');
```

Could be expanded to convener, contact info etc.

Sql statement to create test description table and fill it

```
--
-- table representing a geant 4 test
--
CREATE TABLE test (
    testid      integer not null,                       -- number matches geant 4 test number if applicable
    testname    character varying(50),                  -- name of the test
    description text,                                   -- text describing the purpose of the test
    responsible character varying(100)[],               -- names of persons responsible for the test
    wg          integer REFERENCES wgroups(wgid),       -- geant 4 working group that the test is associated with
    keywords    VARCHAR(50)[],                          -- keywords associated with the test
    inspirerefs integer[]                               -- array elements reference inspireid in table inspirereference
);
ALTER TABLE test OWNER TO g4valwriter;
GRANT SELECT, UPDATE, INSERT, DELETE ON test  TO g4valwriter;
GRANT SELECT ON test TO g4valreader;

insert into "public".test ( testid, testname,description,responsible,wg,keywords,inspirerefs) values
(19,'test19',
'High energy test, provides comparison with NA61 (31 GeV/c proton beam) and NA49 (158 GeV/c proton beam) data sets.',
'{"Julia Yarba (Fermilab)"}',
2,
'{"Inclusive production","high energy","thick Target"}',
'{886780,694016}'
);


insert into "public".test ( testid, testname,description,responsible,wg,keywords,inspirerefs) values
(10000,'Franz',
'Neutron-induced production of protons, deuterons and tritons by neutrons between 300-580 MeV',
'{"Hans Wenzel (Fermilab)"}',
2,
'{"particle production","differential cross section","Thin Target"}',
'{287791}'
);
```

Sql statement defining the testresult table

```
--
-- table representing a geant 4 test result
-- it defines the meta data and links to a histogram or xy data set
--
CREATE TABLE TESTRESULT (
   TRID        SERIAL  PRIMARY KEY,
   TID         INTEGER REFERENCES test(testid),          -- REFERNCES THE TEST THAT TEST RESULT IS ASSOCIATED WITH
   G4VERSION   INTEGER REFERENCES G4VERSIONTABLE(G4ID),  -- REFERENCES TAG OF GEANT 4 VERSION USED
   G4MODEL     INTEGER REFERENCES G4MODELTABLE(MODID),   -- REFERENCES TABLE WITH THE GEANT 4 MODEL OR PHYSICSLIST USED
   BEAM        INTEGER REFERENCES PARTICLETABLE(PDGID),  -- REFERENCES PARTICLE TABLE (PDGID,NAME, ...)
   KINE        REAL,                                     -- KINETIC ENERGY IN MEV
   TARGET      INTEGER REFERENCES MATERIALTABLE(MATID),  -- REFERENCES TABLE defining materials
   OBSERVABLE INTEGER REFERENCES OBSTABLE(MATID),        -- REFERENCES TABLE defininG OBSERVABLES (CROSS SECTION, MOMENTUM OF OUTGOING PARTICLES..
   SECONDARY   INTEGER REFERENCES PARTICLETABLE(PDGID),  -- REFERENCES PARTICLE TABLE (PDGID,NAME, ...)
   REACTION    INTEGER REFERENCES REACTIONTABLE(RID),    -- REFERENCES REACTION TABLE (SCATTERING, PARTICLE PRODUCTION, CAPTURE, DECAY....)
   DTYPE       INTEGER REFERENCES DTYPETABLE(DTID),      -- REFERENCES DATATYPE TABLE (1. 1D HISTOGRAM, 2. 2D HISTOGRAM.....)
   HIST01D     INTEGER REFERENCES HISTOGRAM1D(ID),       -- REFERENCES 1d HISTOGRAM
   HIST02D     INTEGER REFERENCES HISTOGRAM2D(ID),       -- REFERENCES 2d HISTOGRAM
   XYDSET      INTEGER REFERENCES XYDATASET(ID),         -- REFERENCES XYDATASET
   KEYWORDS    VARCHAR(50)[],                            -- keywords associated with the testresult e.g. angle of outgoing particle
);
ALTER TABLE TESTRESULT OWNER TO g4valwriter;
GRANT SELECT, UPDATE, INSERT, DELETE ON TESTRESULT  TO g4valwriter;
GRANT SELECT ON TESTRESULT TO g4valreader;
```

In conclusion

- Think if it makes sense and provide feed back→ would like to get agreement in december and start document.
- Happy thanksgiving to everyone!!
- I will be in Germany starting Thursday and all next week, will read email only occasionally.