



C++ szkriptvázak generálása részecskefizikai adatelemzéshez

Hajdu Ákos

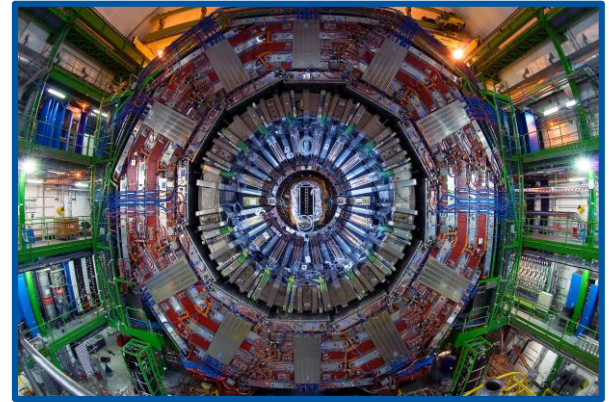
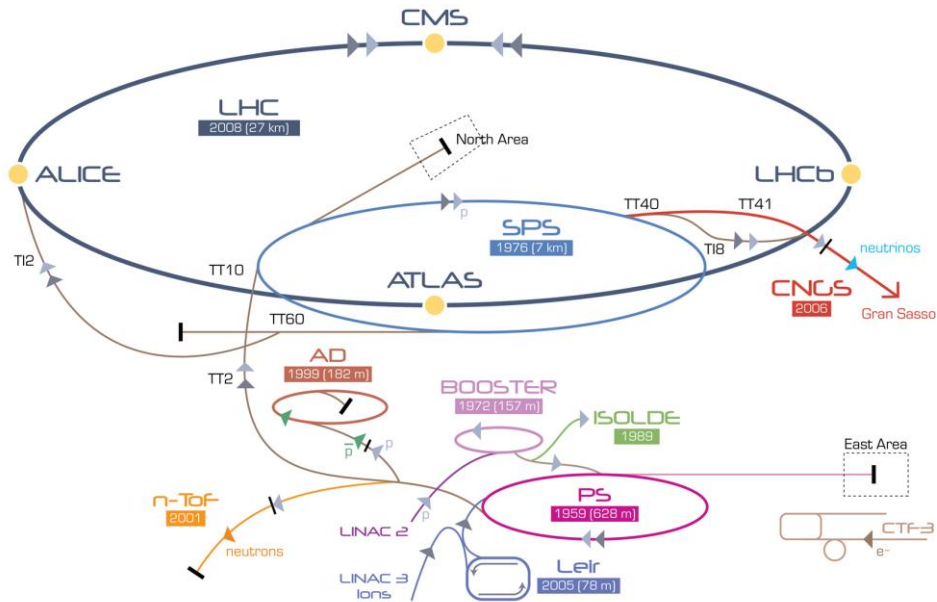
Nyári gyakornoki program 2015

Konzulensek: Bertrand Bellenot, Axel Naumann

CERN PH SFT

CERN

- Részecskefizikai kutatóintézet



<http://home.cern>

Detektortól a publikációig

$$\mathcal{L} = -\frac{1}{4} F_{\mu\nu} F^{\mu\nu} + i\bar{\psi}\not{D}\psi + h.c. + \sum_i y_i \bar{\psi}_i \phi \psi_i + h.c. + \frac{1}{2} \mu^2 \phi^2 - V(\phi)$$

Elmélet

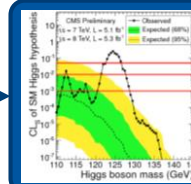
Hogyan kellene működni?



Gyakorlat

Hogyan működik valójában?

Szimulációs
adatok



Adatelemzés
Statisztikai összehasonlítás

Kísérleti
adatok



Eredmények

Elfogadás / cáfolás / ???

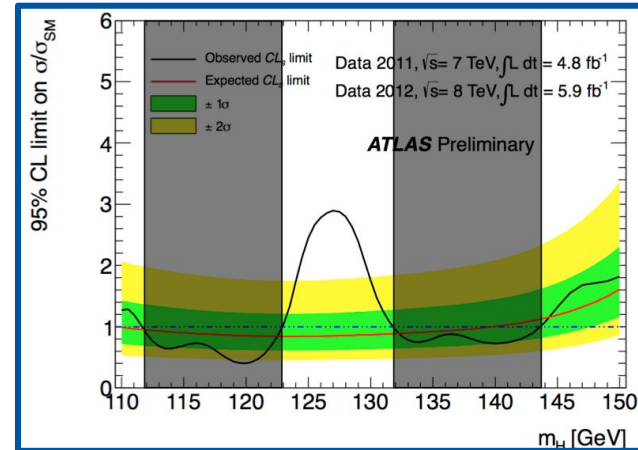
ROOT keretrendszer

- Adatelemző keretrendszer
 - C++ interpreter → szkriptek
 - Hisztogram, görbeillesztés, kiértékelés, minimalizálás, grafika, vizualizáció, ...
- CERN fejlesztés
 - 1995 óta, aktuális verzió 6.06/08
 - ~2005 óta alapkomponeus
 - 3,5M+ LOC, C++

```
auto f = TFile::Open("events.root");
TTreeReader myReader("events", f);

using fourVectors = vector<ROOT::Math::PxPyPzEVector>;
TTreeReaderValue<fourVectors> tracksRV(myReader, "tracks");

while(myReader.Next()){
    auto tracks = *tracksRV;
    for (auto&& track : tracks)
        cout << "Track Pt is " << track.Pt() << endl;
}
```

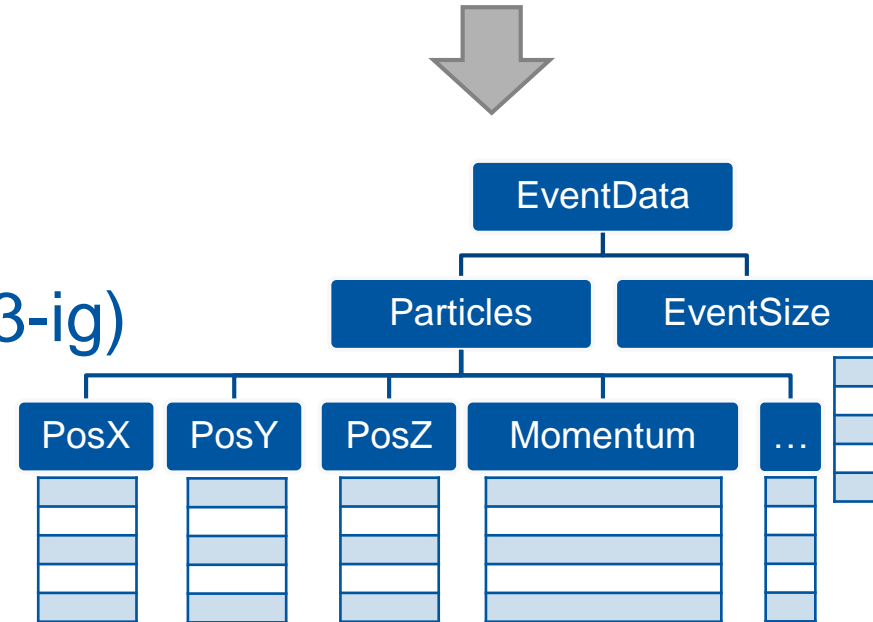


<http://root.cern>

TTree adatszerkezet

PosX	PosY	PosZ	Momentum	EventSize	...

- Oszloponkénti tárolás
- Fa struktúra
- Nagymennyiségű adat
 - **177 PB** LHC adat (2013-ig)
- Hatékony I/O



Adatelemzés

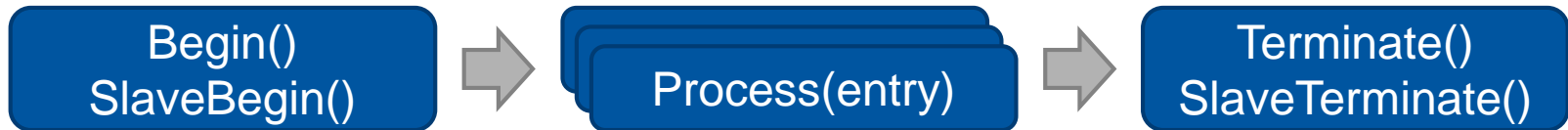
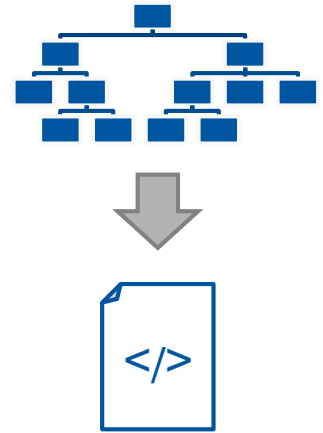
- Általában az összes elem bejárása



- Ismétlődő kódrészletek ☹️
 - Szkriptvázak generálhatóak 😊

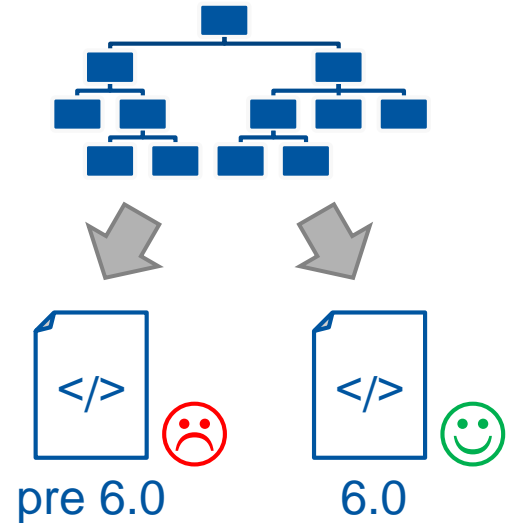
Szkriptvázak

- TTree példányból generálható
- Tartalma
 - Változódeklarációk
 - Kitöltendő metódusok



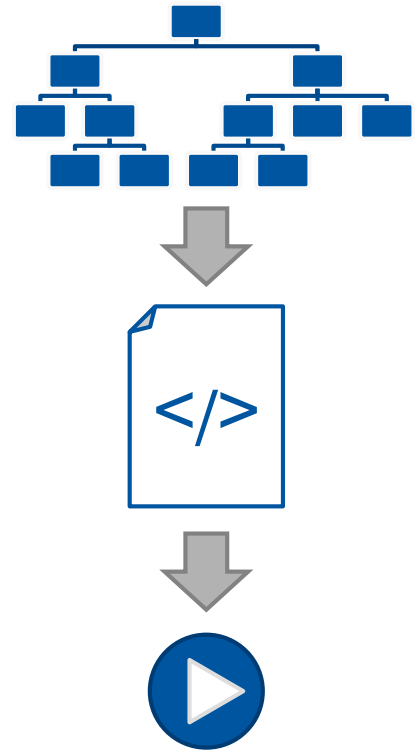
Szkriptvázak

- Létező kódgenerátorok
 - Hiányosságok, hibák
 - ROOT 6.0 előtti kód
 - Pointer mágia, void*, ...
- Cél: ROOT 6.0 kódgenerátor



Megvalósítás

- Kódgenerálás
 - Fa bejárása és elemzése
 - Kód fájlba írása
- Tesztelés
 - Teszt fák készítése
 - Szkriptváz kitöltése és futtatása



Kihívások

- Létező kódgenerátorok
 - Hiányosságok, hibák
 - Bonyolult kód (~4000 sor)
- Sok speciális eset
 - Beágyazott osztályok, változatos tárolók, ...
 - Kompatibilitási kényszerek
- ROOT 6.0 hibák



Példa (.h)

```
class EventTree : public TSelector {
```

```
public :
```

```
    TTreeReader fReader;
```

Hivatkozás a fára

C++11 😊

```
    TTreeReaderArray<double> fParticles_fMomentum = {fReader, "fParticles.fMomentum"};
```

```
    TTreeReaderArray<double> fParticles_fPosX = {fReader, "fParticles.fPosX"};
```

```
    TTreeReaderArray<double> fParticles_fPosY = {fReader, "fParticles.fPosY"};
```

```
    TTreeReaderArray<double> fParticles_fPosZ = {fReader, "fParticles.fPosZ"};
```

```
    TTreeReaderValue<int> fEventSize = {fReader, "fEventSize"};
```

Olvasók

```
    virtual void Begin(TTree *tree);
```

```
    virtual void SlaveBegin(TTree *tree);
```

```
    virtual bool Process(Long64_t entry);
```

```
    virtual void SlaveTerminate();
```

```
    virtual void Terminate();
```

Kitöltendő

```
};
```

Példa (.c)

```
bool EventTree::Process(Long64_t entry) {  
    fReader.SetEntry(entry);  
    cout << "EventSize: " << *fEventSize << endl;  
    for (int i = 0; i < fParticles_fPosX.GetSize(); ++i)  
        cout << fParticles_fPosX[i] << endl;  
    return true;  
}
```

Aktuális elem

Olvasás

Olvasás

Diákként a CERN-ben

- Summer student
 - 8 – 13 hét
- Technical student
 - 4 – 12 hónap
- Doctoral student
 - 6 – 36 hónap



jobs.web.cern.ch/join-us/students

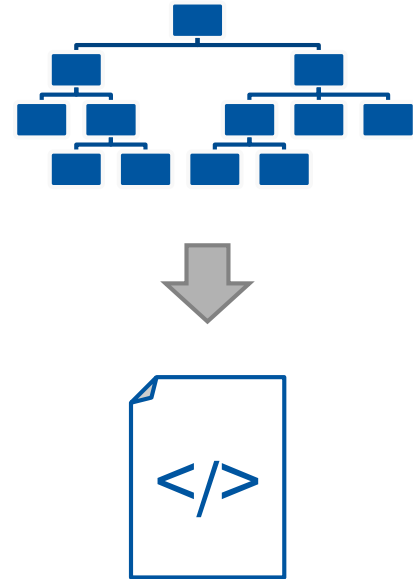
Összefoglalás

- CERN nyári gyakornoki program
- Adatelemző szkriptvázak generálása a ROOT keretrendszerben

hajdua@mit.bme.hu

<http://root.cern>

<http://github.com/root-mirror>





www.cern.ch