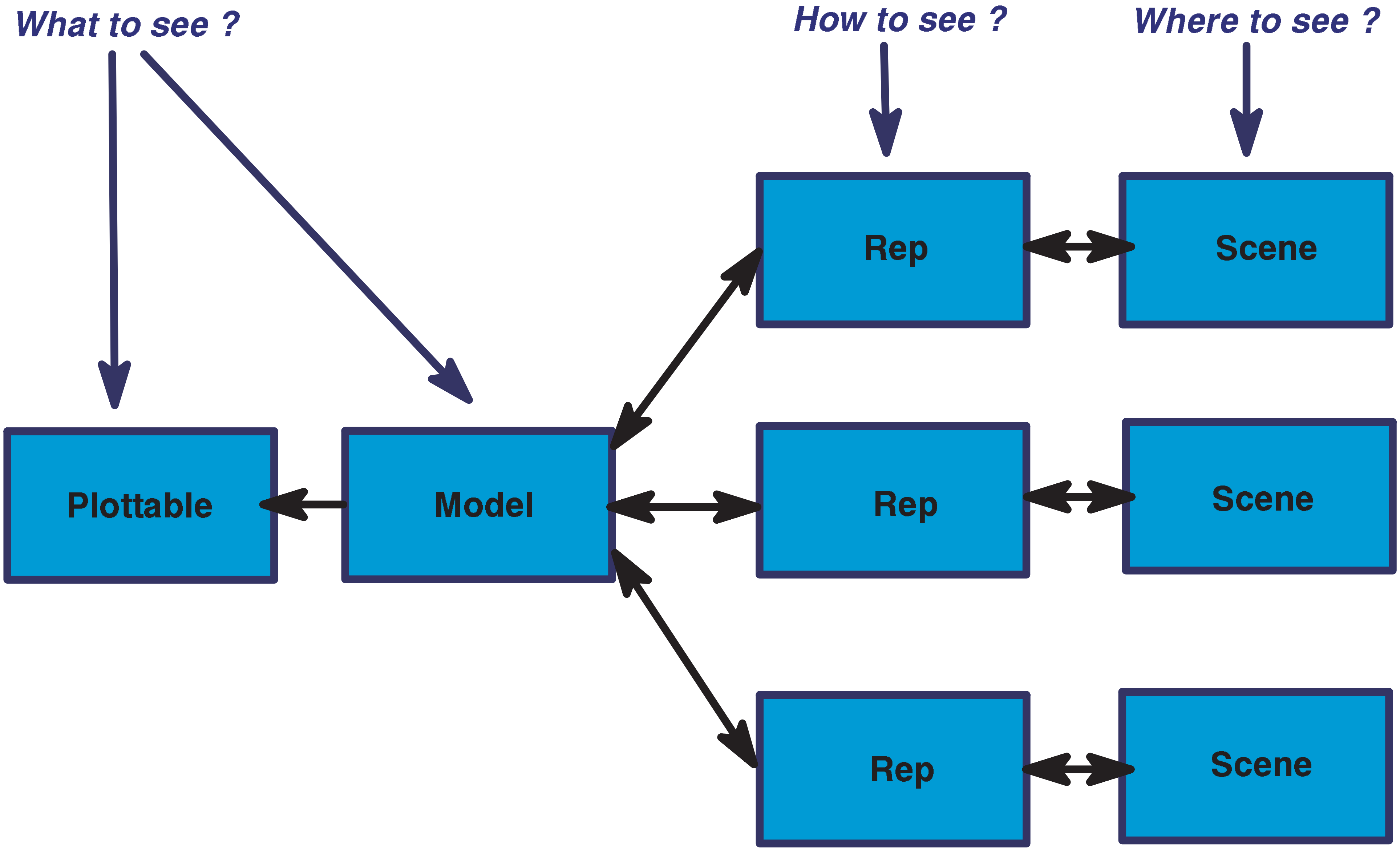
How To Visualize Objects ?

J.Hrivnac/Atlas Graphics Group

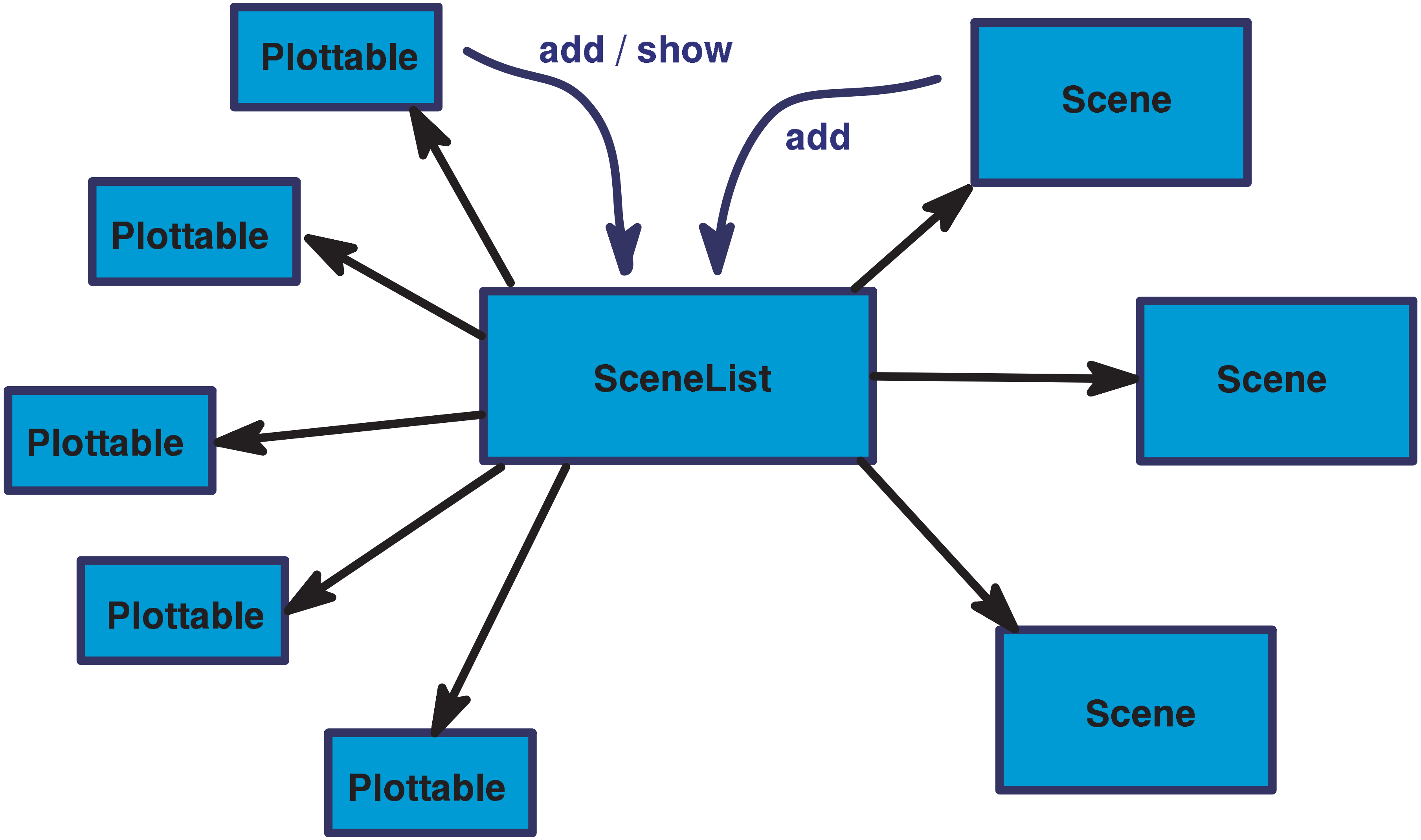
- What should be available to visualize object ?
- How is object visualized ?
- Who does all the coding ?
- How to create Rep ?
- How to use it ?
- What is already available ?
- What will exist in future ?

http://atlasinfo.cern.ch/Atlas/GROUPS/GRAPHICS/

What should be available to visualize object ?



How is object visualized ?



Who does all the coding ?

- <u>**Plottable**: little trivial change needed</u> - it should inherit from common base class
- <u>**Model**: default one created automaticaly</u>
- <u>**Rep**: default one created semiautomaticaly</u> - the required appearance should be defined by human

How to create Rep ?

```
void HBTupleTruthTrack::fill(HBookFile* tFile,
                             const int level)
  {
  HepTuple* tuple = tFile->ntuple("TruthTrack", 2);
  TruthTrack* pTrack = ((TruthTrack*)(_pModel->plottable()));

  if(level > 0)
    {
    tuple->column("pdg", pTrack->pdg_code(), 0);
    tuple->column("px",  pTrack->p_x(), 0);
    tuple->column("py",  pTrack->p_y(), 0);
    tuple->column("pz",  pTrack->p_z(), 0);
    tuple->dumpData();
    }
}
```

<u>already exist:</u> SiDigit, SiDetector, TRTDigit, TRTDetector, TruthTrack
<u>soon:</u> MagneticField, Muons
<u>then:</u> Calo, ...

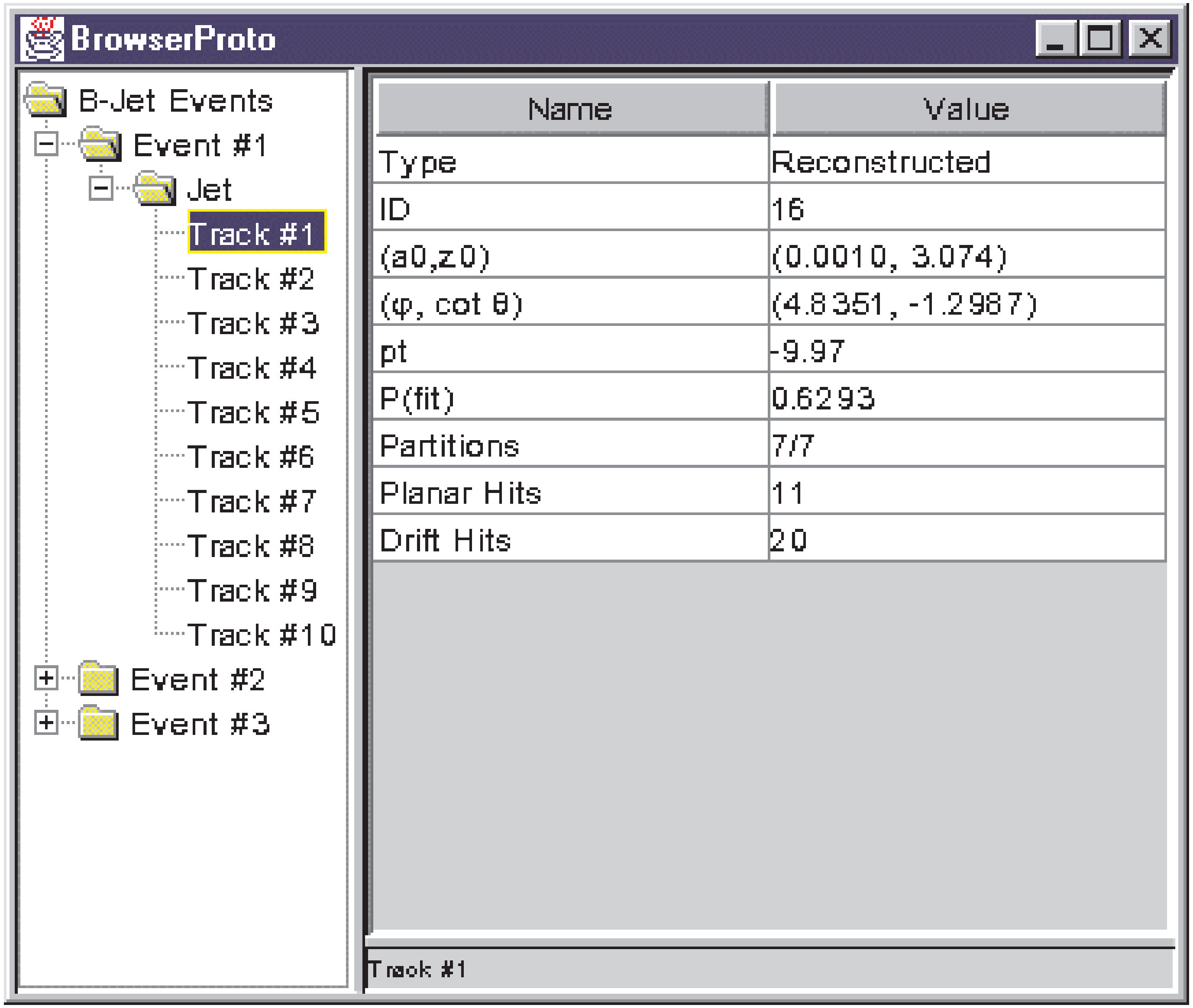
How to use it ?

```
// Create SceneLists
SceneList list1;
SceneList list2;

// Create Scenes
XMLScene xml("MyEvent.xml");
AVRML vrml("MyEvent.wrl");
HBookTuple tuple("MyEvent.hbook");

// Get Plottables
Track* track1 = ...;
Track* track2 = ...;

// Add Scenes into SceneLists
list1.add(&xml);
list1.add(&vrml);
list1.add(&tuple);
list2.add(&tuple);

// Show Plottables
list1.show(track1);
list1.show(track2);
list2.show(track2);
```

What is already available ?

- **Event Display:**
  - **AVRML** - file describing 3D Virtual Reality, browsers are available - full
  - **Atlantis** - sofisticated physics display (special views: fish-eye, clock-view, V-plot,...) - full till Xmas
  - **Wired** - Java based - full till Xmas
  - **Aravis** - Arve graphics - full till Feb'99
- **Histogram:**
  - **HBookTuple** - Paw ntuple, histogram could be added - full
  - **AHisto** - LHC++ HistOOgram - works, but requires Objy + LHC++
  - **ARoot** - in principle works, ugly common objects required - proof of concept
  - **...**
- **Misc:**
  - **AsciiText** - log file - full
  - **XMLScene** - standard textual interchange format, used by Atlantis and Wired - usable, will be extended

What will exist in future ?



- **Also:**
  - **HEPVis**
  - **JAS**
  - **G(UI)**
  - **...**