A Brief ROOT Tutorial

"...the insane root
That takes the reason prisoner."
-Shakespeare

Michael Betancourt
Massachusetts Institute of Technology
NEPPSR 2009

What is ROOT?

What is ROOT?

- NOT a program

What is ROOT?

- NOT a program
- NOT a command line interpreter

What is ROOT?

- NOT a program
- NOT a command line interpreter
- A collection of C++ classes

What is CINT?

- CINT is a dynamic C++ interpreter, in other words it allows one to run C++ code without having to compile it

What is CINT?

- CINT is a dynamic C++ interpreter, in other words it allows one to run C++ code without having to compile it
- Advantages: Quick prototyping and testing

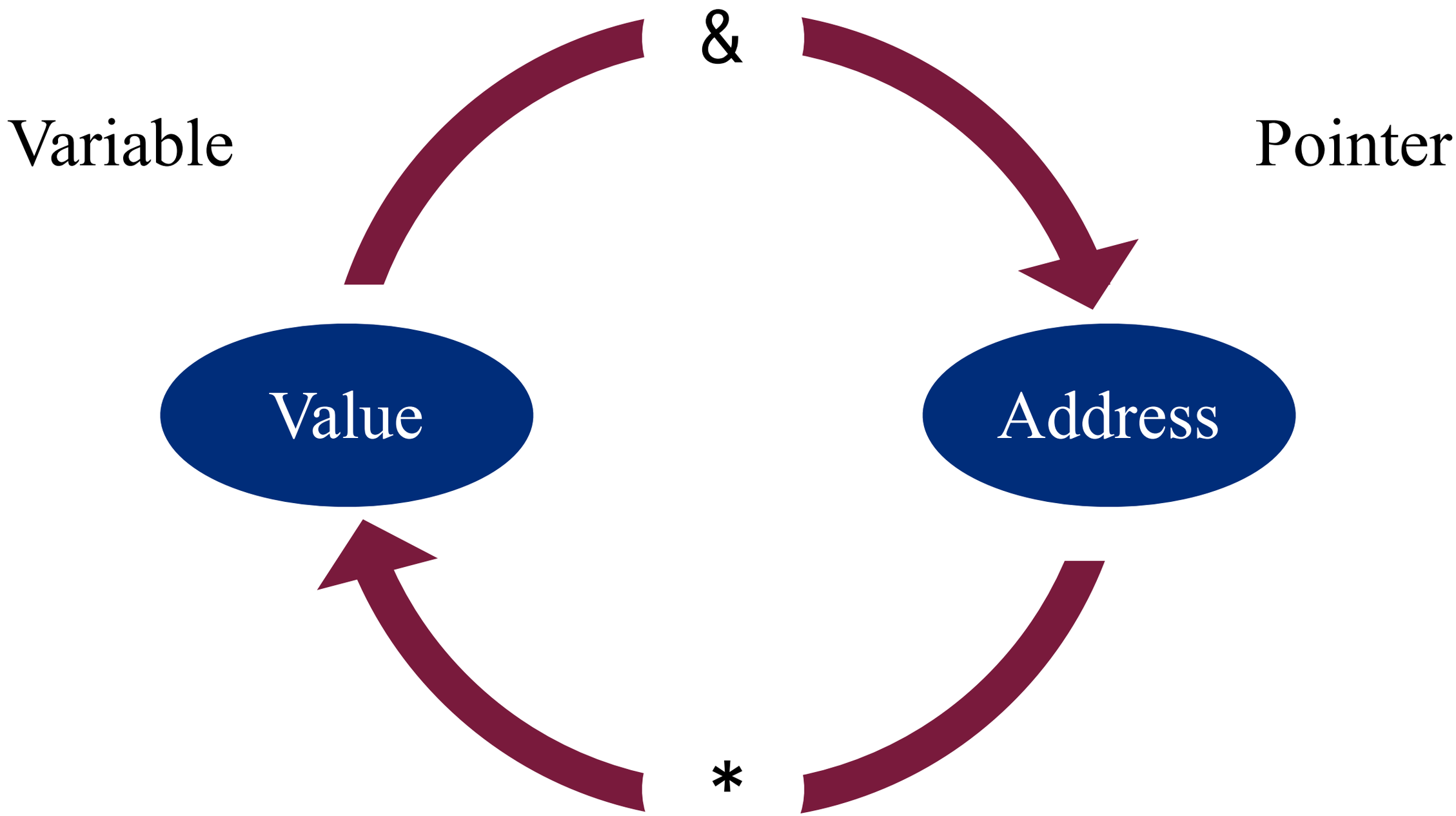
What is CINT?

- CINT is a dynamic C++ interpreter, in other words it allows one to run C++ code without having to compile it
- Advantages: Quick prototyping and testing
- Disadvantages: Slow, awkward properties, limited support outside of ROOT classes

Some Quick Notes on C++

- Common mistake: trying to learn “ROOT” without first learning basic C/C++

Pointers



Pointers

```
double variable = 10;
double *pointer = &variable;
```

```
variable = 10
pointer = &variable = 0x482a8e94
*pointer = variable = 10
```

Classes

```
class meal
{

  public:

    meal(unsigned int nServings);
    ~meal();

    addIngredient(ingredient newIngredient);
    prepareMeal();
    serveMeal();

  private:

    unsigned int mNumIngredients;
    vector<ingredients> mIngredients;

};
```

Classes In Practice

```
meal chickenPipian(2);

chickenPipian.addIngredient(tomatillos);
...
chickenPipian.addIngredient(salt);

chickenPipian.prepareMeal();
chickenPipian.serveMeal();
```

Classes In Practice

```
meal *chickenPipian = new meal(2);

(*chickenPipian).prepareMeal();

chickenPipian->prepareMeal();

delete chickenPipian;
```

References

- Bjarne Stroustrup, *The C++ Programming Language*
- Steve Oualline, *Practical C++ Programming*
- Scott Meyers, *Effective C++*
- *http://cplusplus.com*

Common ROOT Classes

```
TFile inputFile("input.root", "read");

inputFile.ls();

TFile outputFile("output.root", "recreate");

outputFile.Write();
outputFile.Close();
```

Common ROOT Classes

```
TTree t("treeName", "Tree Title");
t.Branch("varName", varAddress, "varName/D");
t.Fill();
t.Write();
```

```
TTree *t = (TTree*)f.Get("treeName");
t->Print();

t->SetBranchAddress("varName", varAddress);
t->GetEntry(n);
```

Common ROOT Classes

```
TH1F histogram(“histName”, “Hist Title”, 100, 0, 50);
histogram.Sumw2();

histogram.Fill(variable);

histogram.Draw();
```

References

- ROOT Class Browser
  - *http://root.cern.ch/root/html522/*
  - *http://root.cern.ch/root/html522/TTree.html*
- ROOT Users' Guide
  - *http://root.cern.ch/drupal/content/users-guide*

One Last Note

- PyRoot
  - *http://root.cern.ch/root/HowtoPyROOT.html*