

# Introducción a ROOT

## Clase 2: Árboles

Fernando Quiñonez PhD.  
*Universidad Industrial de Santander*

2016 Junio 29

## Overview

## TNtuple, TTree y TChain

## TNtuple, el pasto



- ▶ Esta estructura de datos se trata de una entrada plana de valores **del mismo tipo**.
- ▶ En versiones modernas de ROOT fue reemplazada totalmente por el TTree que es más general, porque puede almacenar diferentes tipos de datos.
- ▶ Ejemplo TTreeFromDat1 el resultado final es un TTree llamado T que se encuentra almacenado en un archivo `electronWater.root`.

# TTree, el árbol



# Tipos de datos para un TLeaf

:<variable>/<dataType>:

The symbols used for the type are:

- C: a character string terminated by the 0 character
- B: an 8 bit signed integer
- b: an 8 bit unsigned integer
- S: a 16 bit signed integer
- s: a 16 bit unsigned integer
- I: a 32 bit signed integer
- i: a 32 bit unsigned integer
- L: a 64 bit signed integer
- l: a 64 bit unsigned integer
- F: a 32 bit floating point
- D: a 64 bit floating point
- O: [the letter 'o', not a zero] a boolean (Bool\_t)

# TChain, el bosque



## Creando una clase de análisis desde un árbol .root

```
root -l  
TChain* c = new TChain("T");  
c->Add("file1.root");  
c->Add("file2.root");  
c->Add("file3.root");  
c->Add("file4.root");  
c->Add("file5.root");  
c->MakeClass("AnalysisBase");  
.q
```

Esto crea dos archivos llamados AnalysisBase.C y AnalysisBase.h.