REST-for-Physics Framework

1.2 ROOT I/O

24.01.2023 - Javier Galan - javier.galan@unizar.es

Our main objective in this session is to provide enough information to understand the basics and to break the walls that may impede others to use ROOT in a first instance.

There are many ROOT components, concepts and features that we are not covering in our lectures.

Those concepts are already properly described inside the ROOT courses, where you will also find working tutorials.

We hope that these lectures will help you as an introduction to ROOT, and as contextualization of the REST-for-Physics framework lectures.

A ROOT file is accessible through a **TFile** instance. A **TFile** is organized as a filesystem itself. It contains keys, **TKey**, that can be distributed in directories, TDirectory.

A TFile itself is a directory, it inherits from **TDirectory**.

When we open a new **TFile**, the active directory where we are writing data is the latest file open.

It is critical to understand that when invoking **TObject::Write()** we will write our objects to the <u>active directory</u>. In case we are working with several files we will need to specify the file where we want our objects to be written.

Thus, you need to do `fileN->cd()` just before writing your object `objectN->Write()`.

This applies to other objects in ROOT, such as when you have several canvas objects where you can draw, you then need to specify which is the active canvas, `canvasN->cd()`.

One of the most powerful features of ROOT it is the ability to serialize the data members of a class. This is translated into the ability to write/read C++ objects to/from disk.

This serialization is done through the TObject class. Any class inheriting from TObject will inherit methods to write and access the data of the inherited object.

The second most powerful feature of ROOT is the ability to extend it with new classes or upgrade the existing ones with additional features that define a custom behaviour we are interested in.

ROOT is a development framework, as such REST-for-Physics is an extension of ROOT that can be at the sametime customized, or upgraded by others.

> **In this session we will show how you can create custom classes fully integrated with ROOT!**
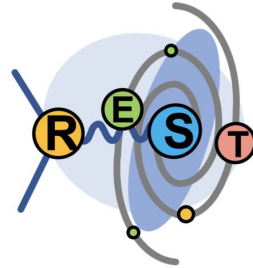
REST-for-Physics Framework

Time for exercises!

24.01.2023 - Javier Galan - javier.galan@unizar.es