

6.12 sprint status report

E. Guiraud

PPP meeting 06/07/2017





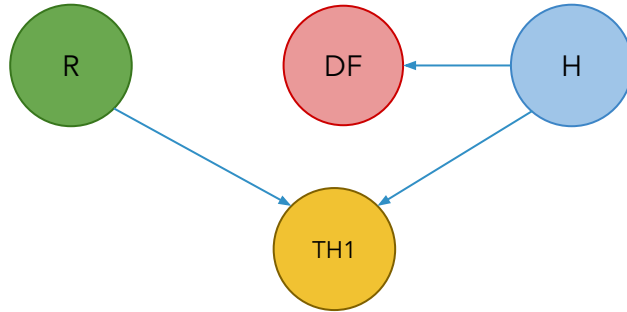
ROOT-8877: defer jitting operations

```
auto h = df.Histo1D();
```



ROOT-8877: defer jitting operations

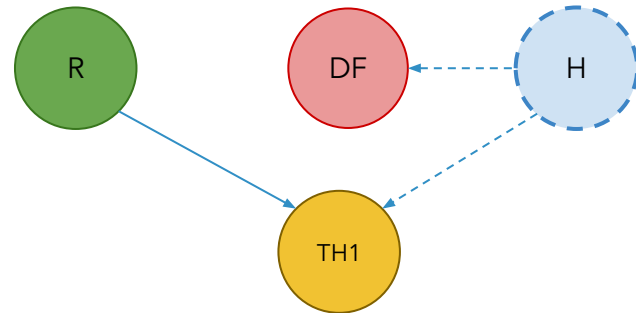
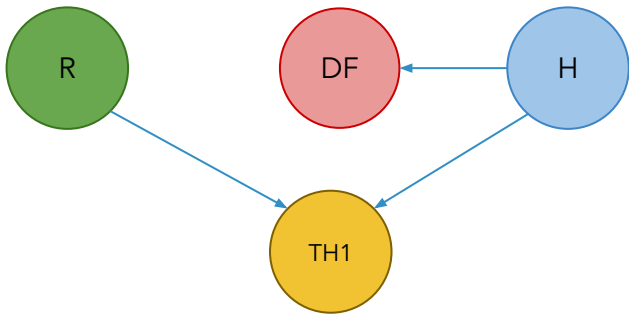
```
auto h = df.Histo1D();
```





ROOT-8877: defer jitting operations

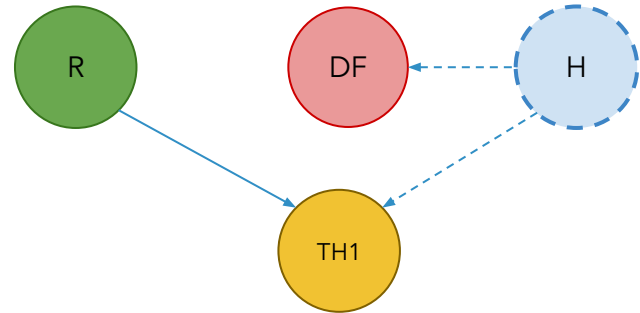
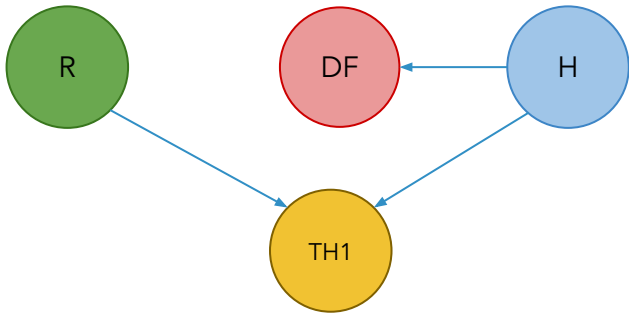
```
auto h = df.Histo1D();
```



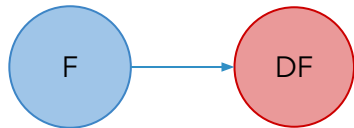


ROOT-8877: defer jitting operations

```
auto h = df.Histo1D();
```



```
auto f = df.Filter("b>0");
```





ROOT-8883: axis range deduction

```
autoh = df.Histo1D("x")
```

```
manualh = df.Histo1D(TH1D("x",100,0.,1.),"x")
```



ROOT-8883: axis range deduction

```
autoh = df.Histo1D("x")
```

```
manualh = df.Histo1D(TH1D("x",100,0.,1.),"x")
```

- problem: threads fill different histograms, must guarantee they can be merged
→ all histograms must have consistent ranges/binning
- current solution is only practical for 1D histograms, scales up to $O(1e8)$ entries*n_histograms
- proposed solution relies on providing a default model with an arbitrary initial range

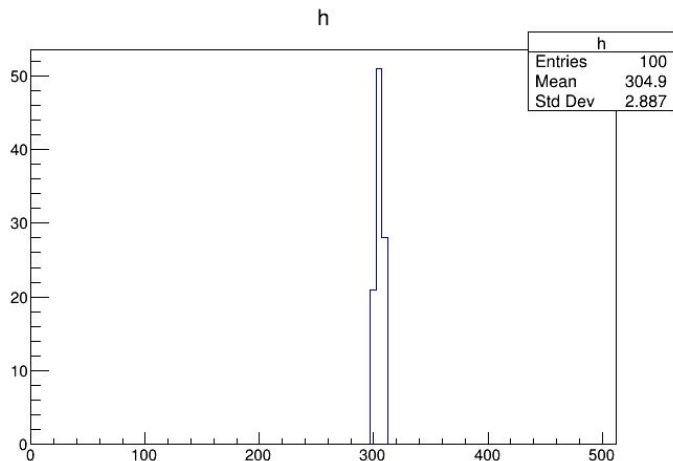


ROOT-8883: axis range deduction

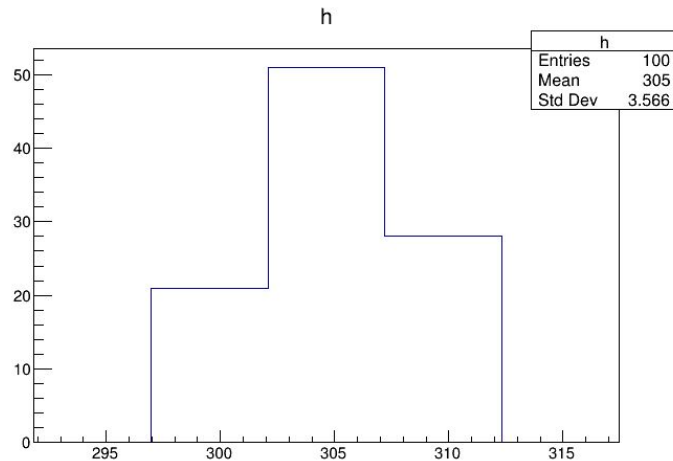
```
autoh = df.Histo1D("x")
```

```
manualh = df.Histo1D(TH1D("x",100,0.,1.),"x")
```

- problem: threads fill different histograms, must guarantee they can be merged
→ all histograms must have consistent ranges/binning
- current solution is only practical for 1D histograms, scales up to $O(1e8)$ entries*n_histograms
- proposed solution relies on providing a default model with an arbitrary initial range



which is better?





Early detection of branch name errors

- ROOT-8879: Consistent, unified selection of column names in TDF
- ROOT-8873: Reinforce the mechanism to detect non existing branches
- ROOT-8880: Leave TDF unmodified (and valid) if booking of transformation/action throws

The plan: call a validation function before anything else in all transformations and actions



Early detection of branch name errors

- ROOT-8879: Consistent, unified selection of column names in TDF
- ROOT-8873: Reinforce the mechanism to detect non existing branches
- ROOT-8880: Leave TDF unmodified (and valid) if booking of transformation/action throws

The plan: call a validation function before anything else in all transformations and actions

```
def SelectColumnNames(columnList, defaultList, nNeeded):  
    if columnList.empty:  
        # use default column list  
        if defaultList.size < nNeeded:  
            throw  
        return (take nNeeded from defaultList)  
    else:  
        # use user-provided column list  
        if columnList.size != nNeeded:  
            throw  
        if !CheckIfColumnsExist(columnList):  
            throw?  
  
    return columnList
```

- throw or return error?
- where to insert the TDataSource logic?