Handling cross-protocol redirections in ROOT

Łukasz Janyst

XRootD

ROOT I/O Workshop
Geneva, 25.06.2014

What? #1

- The new client in XRootD 4 has a capability to process **out-of protocol** redirections

- When you open a file with XRootD, the server may tell that it’s more efficient to try something else:
  - read a local file
  - try reading over another remote access protocol

```
]==> root –b blah.C
[cut]
Error in <TNetXNGFile::Open>:
[Error] Unhandled redirect: file://…/atlasFlushed.root
```

What? #2

- We'd like to have the following scenario handled in ROOT:

  1. Get a URL (ie. **(x)root://**)
  2. Load the appropriate plugin (ie. **TNetXNGFile**)
  3. The plug-in contacts the server
  4. The server says you should actually go elsewhere
  5. Restart at **1)** with the new URL

How?

- Add a new virtual to TFile
  - virtual TString **GetNewUrl**() const;

- Then use it in TFile::Open
  - the first (naïve) implementation is fairly straightforward:

```
       if (f && f->IsZombie()) {
+         TString newUrl = f->GetNewUrl();
          delete f;
-         f = 0;
+         if( newUrl.Length() )
+            f = TFile::Open( newUrl, option, ftitle, compress );
+         else
+            f = 0;
       }
```

Why? #1

- Some sites wish to deploy their XRootD clusters on **batch worker nodes** to optimise their storage
- When XRootD notices that the client is **co-located** with the server (both are at the same box)
- The server may tell the client to read the file directly using **file://** protocol avoiding having to go via network interface

Why? #2

- Useful with **Federated Storage** use cases:
  - if a file is unavailable via one protocol, the client may be told to use another

Status

- The trivial implementation works
- Need to make sure all the **corner-cases** are handled and **performance test**
- Need to make ROOT work with XRootD 4
  - Gerri says he has the PROOF patches ready and tested

Thanks!

Thanks for your attention!

Questions? Comments? Concerns?