

Running the BGV Gaudi SW from the BGV AFS project space

Plamen Hopchev

CERN BE-BI-BL

30 June 2015

BGV project space (folder) on AFS

- **BGV project space on afs:** `/afs/cern.ch/project/lhcbgv`
- Replaces the previously used folder
`/afs/cern.ch/work/p/phopchev/public/LHCbSW`
- Info about the folder administration:
`https://twiki.cern.ch/twiki/bin/view/BGV/AFSProjFolder`
- Proposed R/W rights
 - By default – *read* for all users in CERN AFS, *write* for the “BGV AFS admins”
 - “Public user folder” writable by the respective owner

Proposed structure of the folder

- Folder **sw**

- Store standard scripts

- Currently we have **BGVLogin** and **SetBGVEnv**

- Optional: Install LHCb Gaudi projects (Gauss, Boole, ...)

- Install the BGV Gaudi packages (**SciFiEvent**, **SciFiDigi**, **PatRec**, ...)

- Store reference versions of LHCb+BGV code (“tags”)

- Keep correspondence to BGV **svn** tags

- Store also a fixed version of the geometry (XmlDDDB) and “small” data files produced with these versions

- Propose to have only read rights, except a few “release managers”

- Folder **user**

- One directory per user

- Write rights only for the respective user

- Folder **L0Trig**

- The switch to the BGV AFS project space is a good moment to define
 - [Standard ways](#) to setup and run the BGV SW
 - [Referencing/tagging mechanism](#) for the LHCb+BGV code
- The proposed approach is the following:
 - [Use the LHCb Gaudi projects as released by LHCb on AFS](#)
 - A “private” BGV installation is possible before the archival of the relevant project versions
 - [Get the BGV packages from BGV svn tags](#)
 - Compile in dedicated folders in the BGV afs tree, or privately, in `~/cmtuser`
 - [Define the commands needed to setup the environment for compiling and running the SW](#)

- The purpose of the BGV scripts is to provide
 - A standard way to setup the environment
 - A translation between BGV versions and LHCb project versions
 - E.g. `BGV v1r0` → `Gauss vArB`, `Boole vCrD`, ...

1. `BGVLogin`

- Set the env variable `BGVPROJDIR` to `/afs/cern.ch/project/lhcbgv`
- Call the standard LHCb script `LbLogin` (versioned!)

2. `SetBGVEnv`

- The only purpose of this script is to define a few BGV environment variables
 - It does not call any LHCb scripts
- Contains a dictionary for translating BGV versions to LHCb project versions
- Example usage given on the next slide
- Accepts up to 1 argument: the version of the BGV release (e.g. `v1r0`)

- Example calls and resulting script output:

- **SetBGVEnv v1r0**

Setting BGV SW version v1r0.

Defining environment variables:

\$BGVSim : Gauss v48r2

\$BGVDigi : Boole v29r6

\$BGVRec : Panoramix v23r0

\$BGVRELEASE : --user-area /afs/cern.ch/project/lhcbgv/sw/releases/bgv_v1r0

All OK.

- **SetBGVEnv v1r1**

Setting BGV SW version v1r1.

Defining environment variables:

ERROR: key *Sim_v1r1* not found in the VersionMap dictionary

See available versions below

BGV VersionMap dictionary:

*** Sim_v1r0 : Gauss v48r2

*** Digi_v1r0 : Boole v29r6

*** Rec_v1r0 : Panoramix v23r0

- If you want *only to list* the available versions, call **SetBGVEnv -l**

Workflow

Step 0: Login to LXPLUS

Step 1: Prepare the **BGV** environment

- `source /afs/cern.ch/project/lhcbgv/sw/scripts/BGVLogin`
- `SetBGVEnv v1r0`

Step 2: Prepare the **LHCb** environment

- Call `SetupProject` or `setenvProject` like shown in the table below
 - Using as example Boole v29r6 (LHCb column) and \$BGVDigi (BGV column)

Usecase	LHCb command	BGV command
Run a job from ~/cmtuser	<code>SetupProject Boole v29r6</code>	<code>SetupProject \$BGVDigi</code>
Run a job from a release	<code>SetupProject Boole v29r6 --no-user-area</code>	<code>SetupProject \$BGVDigi \$BGVRELEASE</code>
Develop code in ~/cmtuser	<code>setenvProject Boole v29r6</code>	<code>setenvProject \$BGVDigi</code>
Develop code in a release area	–	<code>setenvProject \$BGVDigi \$BGVRELEASE</code>

Summary

- The BGV AFS folder is ready for use
- The scripts `BGVLogin` and `SetBGVEnv` should be used to setup the BGV SW environment
 - The latter translates between a BGV version and LHCb project version, and sets a few BGV environment variables
- Pass these env. variables to `SetupProject` or `setnevProject` in order to complete the setup of the SW environment
- Running jobs is identical to LHCb