Core, Externals, structural changes

A. Tsaregorodtsev,
CPPM-IN2P3-CNRS, Marseille

- ▶ v6r15 current production release

- ▶ Supported release branches
  - ▶ rel-v6r13, current latest patch v6r13p21
  - ▶ rel-v6r14, current latest patch v6r14p31

- ▶ Previous release branches are not supported any more
  - ▶ No more patches accepted

- Support for RFC proxies (A.Casajus)
  - Should be handled with care: some services require RFC proxies, some do not support them
  - Non-RFC proxies by default
  - `dirac-proxy-init –r`

- The VOMS extension can be decoded now without globus dependent commands
  - Can understand VOMS extension on the Mac

- Enhanced client-server communication protocol with multiple service instances (Z.Mathe)
  - Switch from one URL to another in random order
  - Try several times the same URL
  - With gradually increasing time delay, the algorithm attempts to minimize the service response even if some instances are down
    - E.g. for the service restart

- CRLs are taken into account in the DISET client-server handshaking

- Freed from "convenient" importing of multiple modules and some code execution
  - E.g. evaluating the DIRAC platform
  - Unnecessary slow down of command lines
    - Significant increase in responsiveness for the COMDIRAC commands
  - Still importing gLogger, gConfig, S_OK, S_ERROR

- Also necessary for code documentation generation
  - Involves module import and object instantiation

▸ S_ERROR enhanced to make errors uniform and ensure proper error propagation (C.Haen)

▸ A dictionary that contains
  ▸ integer Error number
  ▸ Message
  ▸ traceback information

▸ Traceback is printed out in the DEBUG level by the logger if S_ERROR dictionary is passed to it

▸ S_ERROR as a dictionary advantage
  ▸ No special object encoding is necessary
  ▸ Can use json-based encoding for the DISET protocol
    ▸ Factor 3-10 faster encoding/decoding (Z.Mathe)
    ▸ Can return S_ERROR structures in the service call

- DErrno module contains
  - Standard numerical error codes
  - Associated short error strings
  - Extensions can add there own Error numbers

- Calling sequence
  - `S_ERROR( Errno, "Message" )`
  - The old calling sequence is interpreted as
    `S_ERROR( 0, "Message")`
    to retain backward compatibility

- Comparing errors utility
  - cmpError(Error, DErrno)
    - Error:
      - Int error code
      - String message
      - S_ERROR dict

▸ DIRACDocs and TestDIRAC projects are going to be suppressed, replaced by *docs* and *tests* top level directories in the DIRAC project

    ▸ This is following the standard structure for a Python project

▸ Eventually the top level directory repository structure will become just

      DIRAC

      docs

      tests

▸ The release building tools are already adapted

    ▸ E.g. VMDIRAC 2.0 code is included this way while keeping 1.0 code in the old structure

▸ Once we will move to this structure developers will have to synchronize their local repos

    ▸ Will be announced before then

- Each PR is merged only if there is a Review OK stamp from at least one reviewer
  - In general this is a very good practice, helps to spot problems and suggest better solutions

- Can take certain time for not so obvious PRs slowing done merging
  - E.g. massive code update to follow (pylint/pep8) standards

- We should probably dedicated one release to just following the coding conventions
  - Seemingly harmless but numerous changes can create a nightmare with resolving conflicts
  - Do not do code beautifications in the patches, please !

Github auxilliary services

▸ Travis CI, Coveralls, Landscape

✓ **All checks have passed**
   2 successful checks

Show all checks

✓ **This branch has no conflicts with the base branch**
   ~~Merging can be performed automatically~~

landscape-bot commented 4 days ago   +😊 ✏️ ✕

`health` `69%`

Repository health increased by 0.01% when pulling **8632ca1** on **JanEbbing:rel-v6r15** into **a1e86c3** on **DIRACGrid:rel-v6r15**.

- 4 new problems were found (including 0 errors and 1 code smell).
- 9 problems were fixed (including 0 errors and 1 code smell).

▸ It is now mandatory to accept a PR

   ▸ Coveralls can be an exception

▸ 10

- ▶ **ReadTheDocs**
  - ▶ Code documentation tools updated to be used with the ReadTheDocs service (André Sailer)
  - ▶ http://dirac.readthedocs.io/en/rel-v6r15/

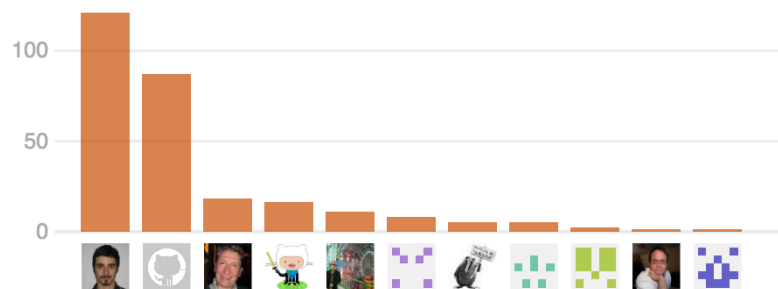 | Documentation sources | |
  |---|---|
  | **_User Guide_** | **_Developer Guide_** |
  | Everything users need to know | Adding new functionality to DIRAC |
  | **_Administrator Guide_** | **_Code Documentation (v6r15)_** |
  | Administration of the DIRAC service | Code reference |

  This documentation in also available in PDF version

  - ▶ diracgrid.org documentation pointer will be redirected
    - ▶ rel-v6r15 or latest

DIRAC
THE INTERWARE

▶ Commits

  ▶ From 15 distinct authors in the last year

  ▶ From 11 distinct authors in the last month

Excluding merges, **11 authors** have pushed **261 commits** to integration and **275 commits** to all branches. On integration, **788 files** have changed and there have been **41,895** additions and **18,661 deletions**.

▶ 84 open issues

  ▶ Have to update their status

- Externals are binary libraries and third party python packages needed to run DIRAC clients and services
- The current version being shipped with DIRAC v6r15 is v6r4p1
  - Openssl 1.0.2f
    - Free from recently reported security holes
  - Recent versions of readline, ncurses, sqlite, bzip2, zlib libraries
  - Imaging libraries (libpng, freetype, libart_lgpl) – for servers
  - MySQL 5.6.23, MySQL-python 1.2.5
- Python modules
  - Sqlalchemy, pexpect, requests
- Python interpreter 2.7.8
  - Version with Python 2.6 is not compiled any more

Platforms

- Server set of v6r4p1 Externals
  - SL6 (glibc 2.12)
  - CC7 (glibc 2.17)
  - Ubuntu 14.04 (glibc 2.18)
  - Support for SL5 is dropped
- Client set of v6r4p1 Externals
  - SLC5
  - SLC6
  - CC7
  - Ubuntu 14.04
  - Darwin 10.10

- dirac-install makes use of the CVMFS if available
  - dirac.egi.eu CVMFS directory
  - Hosted at RAL
    - Updated as part of the release procedure
  - Contains release tarballs
    - Not using pre-installed DIRAC software like LHCb

- dirac-install-client script to guide through the usual client installation procedure
  - Still not obvious how to use for an average non-expert user

- Becoming really difficult to support each laptop requirements
  - No Darwin VM, for example

- Using Docker on a user machine can be suggested
  - Installing Docker is not more complicated than installing a DIRAC client

- Running client with Docker:
  docker run -it -v /Users/atsareg/.globus:/root/.globus -v /Users/atsareg:/home/atsareg **diracgrid/client:gridfr** /bin/bash

- One gets without a previous DIRAC installation a fully configured client
  - Need to setup the user certificate