

Atelier d'aide à la programmation

Partage de compétences

Léo BAUDOUIN

`baudouin.leo@gmail.com`

03-04 juin 2019

Gestion de projet informatique



Gestion de projet informatique

Besoins

- Gestion du code source
- Wiki
- Rapport de bug
- Gantt
- Historique des modifications/versions
- Règles de codage (code style)

Gestion de projet informatique

Besoins

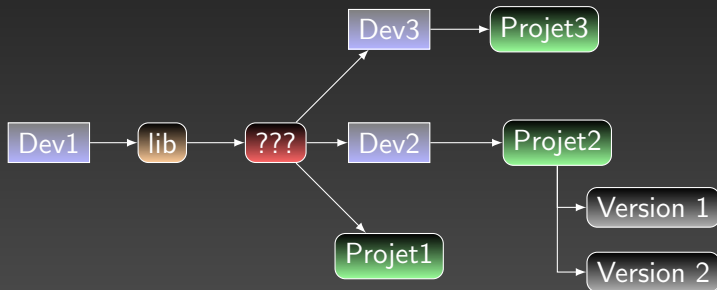
- Gestion du code source
- Wiki
- Rapport de bug
- Gantt
- Historique des modifications/versions
- Règles de codage (code style)

Solutions

- Redmine \Rightarrow `http://forge.clermont-universite.fr/`
- Jira, Github, GitLab, ...
- Bitbucket, Trac, Mantis, ...

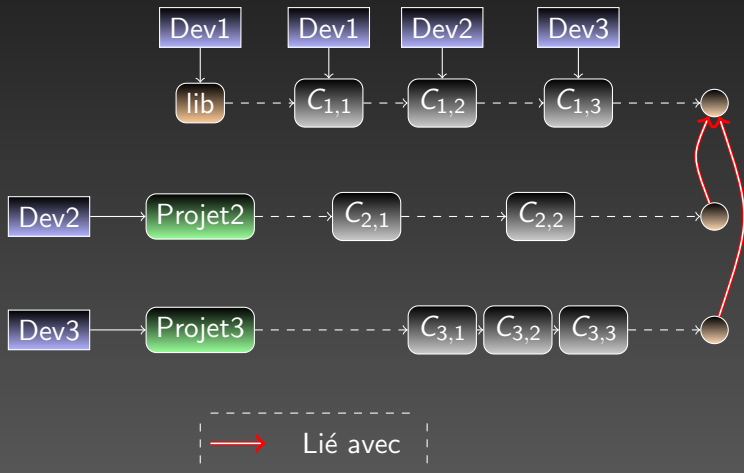
Partage de codes-sources

Exemple : Partage d'une bibliothèque



Partage de codes-sources

Exemple : Partage d'une bibliothèque



Les outils

Utilisation d'outils de base



L^AT_EX



Git



CMake

Aide à la compilation



- Projet multi-plateformes :
Linux / Mac / Windows
- Recherche automatique des dépendances
- Configuration des projets
- Pas de **makefile** à la main

Bonus

Projet vierge prêt à la compilation et au déploiement

Doxygen

Utilisation

Documentation automatique / création d'un site web

Exemple C++ :

```
/**  
Apply a rotation to a transformation.  
\f[ m \leftarrow m \times \exp \left( \begin{array}{ccc} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{array} \right) \f]  
*/  
  
LIBV_GEOMETRY_EXPORT void apply_rotation  
( Matrix3d &m ///< A transformation matrix.  
 , const Vector3d &d ///< A rotation in axis-angle representation.  
 )  
{  
    Matrix3d skrew;  
    skrew << 0, -d.z(), d.y(), d.z(), 0, -d.x(), -d.y(), d.x(), 0;  
    m *= skrew.exp();  
}
```

Doxygen

Rendu :

Member Function Documentation

```
LIBV_GEOMETRY_EXPORT void libv::apply_rotation ( Matrix3d & m,  
                                                  const Vector3d & d  
                                                  )
```

Apply a rotation to a transformation.

$$m \leftarrow m \times \exp \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix}$$

Parameters:

- m** A transformation matrix.
- d** A rotation in axis-angle representation.

Definition at line **31** of file **geometry.cpp**.