Tango Community Collaboration

- ***Reminder on current (good ?) practices***
- ***Requests for ideas to enhance the current situation***

Alain BUTEAU
SOLEIL Synchrotron

Introduction

- Why enforcing collaboration practices ?
- Project Management
- Issues and requests management
- Source Code Management

Why enforcing collaboration ?

- The first motivations mentioned in the 2002 M.O.U firmed between SOLEIL and ESRF were :
  - *Lower technical risks*
  - *Minimize developments costs*
- These statements are always true for all the Tango community
  - *Nevertheless during Tango community's history many times "Re-Development" seemed easier than "Collaboration"*
- Extending the collaboration to newcomers like industrial partners or new facilities enforces the need to have :
  - *Transparent collaboration practices*
  - *Make real the initial objectives (i.e lower risks and costs minimization)*

Project Management

Project Management

Current practices on tango-cs

- The Tango project is composed of various software packages which are in charge of institutes (and not persons !)
  - *This is clearly defined in the Tango Consortium Document*
- **Then it is the responsibility of each institute to present to the Tango Meeting :**
  - *the new features*
  - *and the foreseen roadmap of developments on the workpackages it has in charge*

Project Management : What can be enhanced

Proposals for enhancements on tango-cs

- **Enforce the responsibilities of each institute in charge of a package :**
  - By providing SYSTEMATICALLY a ReleaseNotes for each new version of the package
  - To help newcomers (or potential contributors) each package should be delivered with :
    - *Developers documentation : architecture and software design*
    - *End user documentation*

Project Management : What can be enhanced

Proposals for enhancements on tango-cs

- Tango Consortium document defines a process called "Enhancement Proposals" to manage this case
- When someone (person or institute) wants to influence the development roadmap of a workpackage
- The request must be written (a ***TEP***)
- It must be send before the "Tango meeting" to profit of these "physical" meeting to :
  - *have direct discussions on this specific request*
  - *and the possibility to insert it in the institute in charge roadmap*
- In case development priorities are diverging the Requester must "inject" resources to influence the development roadmap

Project Management

Current practices on tango-ds

- ➔ Nothing !
- ➔ Development are done by each institute without coordination with others

Proposals for enhancements on tango-ds

- ➔ Each institute Tango coordinator could write before the Tango meeting a document giving the list of the commercial instruments (or detectors) he will have to interface to Tango

Requests Management

Project Management

Current practices on tango-cs

- *The Sourceforge tracker is partially used (it depends on the software package !!)*
- *Nevertheless it is not really up-to -ate (some requests are opened since more than 5 years)*
  - *Which could sometimes give to the newcomers the feeling Tango is a dead project !*

Requests Management on tango-cs

Proposals for enhancements on tango-cs

- **The SourceForge tracker should be used for ALL features and bug requests**
  - *To give the whole community the visibility on other institutes requests*
  - *This also helps package responsible to write ReleaseNotes based on these requests*
  - **Potential limits :**
    - *Depending on the projects requests may be described in internal institute's tracker*
- **Before the Tango meeting , the institute in charge of a workpackage must clean up Tracker Request**

Requests Management on tango-ds

Current practices on tango-ds

- Before starting a new development , use the “green site” search engine to find if someone already solved the problem
- Each project (*i.e a particular DeviceServer*) comes with a file (generated by POGO) which contains the name of the developer of the project
- Before forking, contact him (or the institute ‘s Tango contact)
- Try to find an agreement to avoid forking !

Potential Issues

- The Device interface requirements may be different between the 2 institutes
- The solution is then to put as much development as possible in a C/C++ library rather than in the Device methods (exemple of LIMA)

Source Code Management

Source Code Management

Current practices on tango-cs

- Developers of each software package are well defined
- If someone needs to commit a patch he must contact the developer in charge

Current practices on tango-ds

- Developers of each software packages is defined in POGO generated file
- An SVN tag policy has been defined to protect “production version” and “development versions”
  - Development are done in the trunk
  - Production version are tagged release_x_y_z in separate folder
  - Doing branches is to be avoided

Source Code Management

Proposals for enhancements on tango-ds

- ➔ All institutes should put SYSTEMATICALLY on tango-ds DeviceServers which interface commercial instrumentation
- ➔ Put the development code in C/C++ libraries and not in the DeviceServers code to allow another institute have a different Tango interface

Conclusion

Far from the optimum but in progress ..

- My (personal) feeling is that current situation is (very) far from the optimum to minimize risks/resources on software developments done within the community
- Some processes (*like TEP*) are now defined by the Tango Consortium Document
- The responsibility of institutes members of the Tango collaboration must be enforced
- The new Tango Consortium organization is an opportunity to do so

Your suggestions/proposals are very welcome !!