Transition to JIRA and Git

- Alina Gabriela Grigoras
- Alina.Grigoras@cern.ch

From Savannah to JIRA

- Why switch from Savannah to JIRA?
  - Further development and support for Savannah tool has stopped
  - End of 2013 will mark the complete shut down of CERN Savannah instance
    - All projects that were not marked to be migrated will be lost!
    - All migrated project will be still accessible using the old Savannah links, no documented reference to Savannah tickets will be lost
- Why JIRA?
  - Following up an extensive evaluation CERN IT department chose JIRA as tracking and planning tool
    - Features:
      - Record tasks, requests and bugs
      - Track their progress
      - Organize, assign and prioritize
      - Search, filter and report
    - Developed and supported by a private company
   - Synchronized with CERN Central Service for authentication and authorization

From Savannah to JIRA

- Current status
  - SFT implemented a plugin to facilitate the migration of all projects and issues from Savannah to JIRA
  - The plugin was able to migrate the ALICE Offline projects, including custom fields, from Savannah to JIRA 5
  - The plugin is not behaving properly when migrating custom fields from Savannah to JIRA 6
    - No news from SFT!
    - Final shutdown deadline still on!
  - If the plugin problem will be fixed than the migration will be smooth
    - Only 30 minutes of shutdown
      - The Savannah repository will be closed
   - All issues will keep their previous Savannah links and will be redirected to JIRA repository

Why move from SVN?
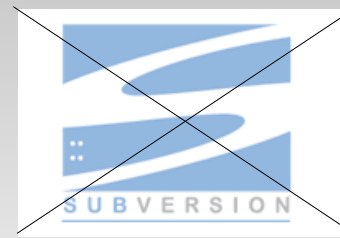
- Why move from SVN?
  - Present experience showed that SVN, and especially CERN http(s) instance has problems when dealing with
    - big commit - large files or many files
    - big merge
    - binary files
  - If the repository integrity gets broken it takes a lot of time to fix it:
    - During this time write access is closed
    - Implies dumping and loading the full repository ~ 2days
    - Synchronizing with both CERN IT Git team and AFS team
  - Git offers some advantages that can improve the quality of developing process
  - CERN IT department offers and supports a Central Git Service
   - Uses CERN Single Sign On
    - Hosted on AFS

SVN vs Git
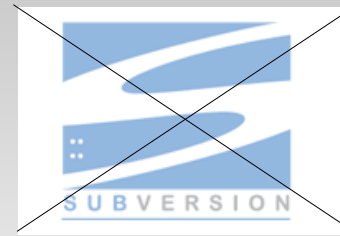
- **Advantages over SVN**
  - Distributed version control system – every user has a clone of the repository, if the central instance gets broken it can be fixed immediately
  - Strong support for nonlinear developing
    - easy branching, fast and efficient (it doesn't create a whole copy of your current working state, compared to SVN where branching is a work around directories)
    - easy merging
  - Fast operations due to local repository (commit, checkout, diff etc)
- **Disadvantages over SVN**
  - Impossibility to partially clone the repository, for example to checkout only a branch or a tag
  - Notifications per project, users with commit rights will receive emails for each commit, including in directories they are not allowed to commit
  - Although similar to SVN, users will need to do some effort to get used to it

Migration procedure
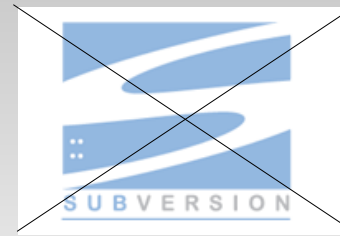
- Migration procedure
  - Fast migration, just a few minutes
    - Close write access to SVN repository
    - Final synchronization between SVN repository and Git repository
    - Start using AliRoot Git repository
  - SVN repository will remain accessible in read only mode
  - Git repository will be accessible using the same credentials as the SVN repository
  - SVN repository will not be synchronized with the Git repository, commits in Git will not be submitted back to SVN
  - An email will be sent before the actual migration containing the exact timing and a quick "How to"
- Update your scripts to use the Git repository instead of SVN repository!

Start using AliRoot repository

- Please take time to read and try
  - SVN basic commands to Git commands
    - http://aliceinfo.cern.ch/Offline/node/2912
  - Test AliRoot Git repository
    - http://git.cern.ch/pubweb/AliRoot.git
      - Clone only
      - Synchronized with SVN repository every 3 hours
  - Send feedback!

Have fun using Git!