Team Development Workflow

Team Sharing Workflow

A typical Servoy development project with multiple developers would have a workflow as follows:

- The development team will decide on a team provider technology and a team provider server or service. A repository needs to be created on this server. Every team member will properly connect with this team provider repository with the corresponding client tools.
- One of the developers (or the lead developer) will create a project for the application's main solution, projects for any applicable modules (this may include projects from another SVN, like ServoyForge), and the resources project, all in Servoy Developer.
- The designated (or lead) developer will share the main project, any module projects, and the resources project with the repository. This developer will also commit any code/files that may exist in these projects.
- The other developers will check out all the projects into their own workspaces. These developers should also create any local databases if necessary.

> ⚠ Note: These developers should start with a clean workspace for the entire application.

- Development begins. During development, developers will use the team provider client tools to interact with the repository (and by extension, each other). A typical scenario would be:
    - At the start of the day/development session, the developer will perform an update to get the latest version of the code.
    - At the end of the day/development session, the developer will perform a commit send their latest code to the repository
    - It may be prudent to commit or update code more frequently in order to reduce the chance of conflicts.
    - It may also prudent not to commit certain changes right away if those changes are incomplete and/or could have adverse effects that could affect other developers.
    - If there is an extreme variation in the code, a developer may choose to branch the code
- Ready to deploy: when it becomes necessary to deploy the application (for test/beta/final/patches), a single developer will update to get all the changes and create an export file for the application server to import. The developer may choose to create a tag at this time as well.