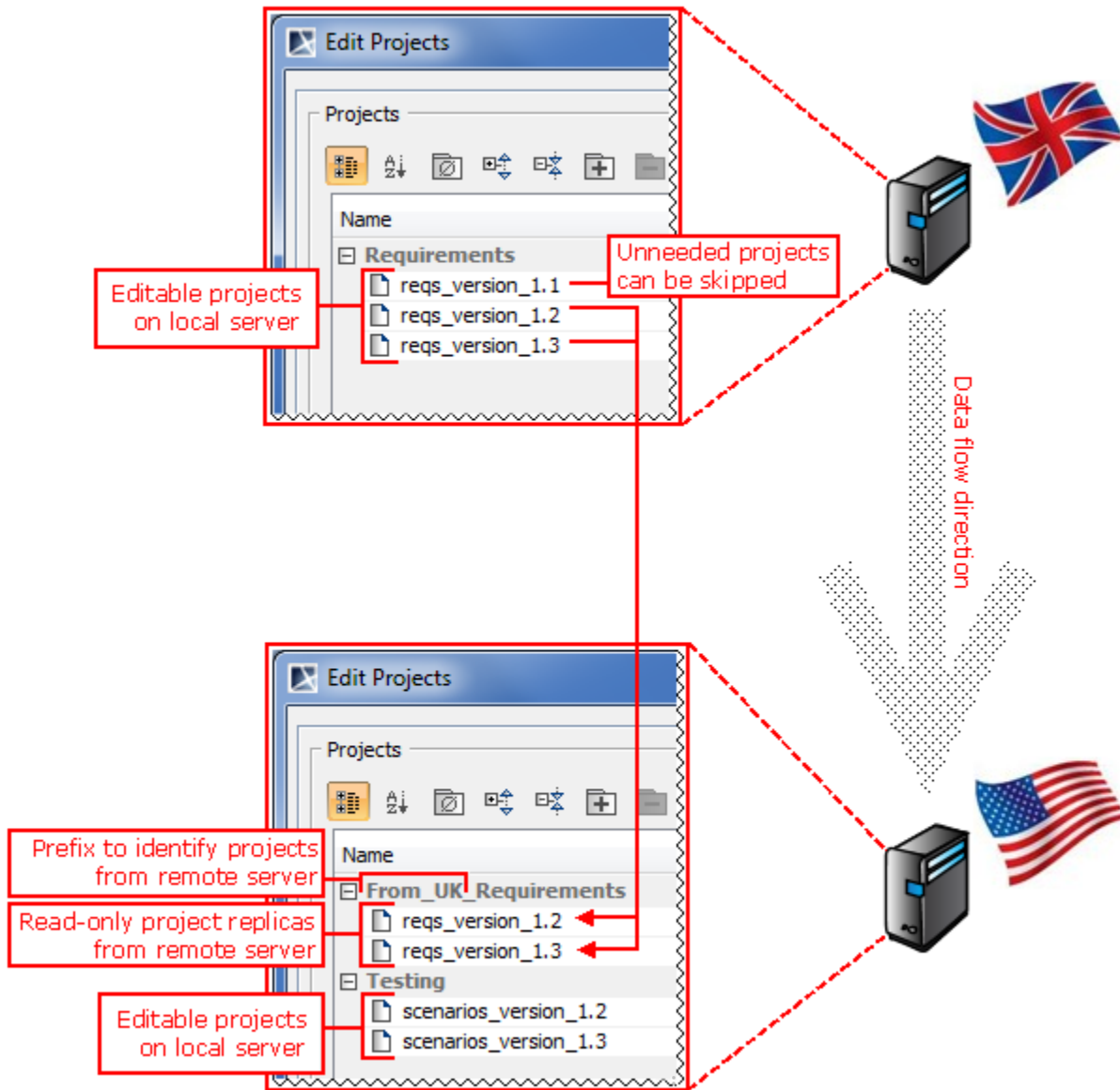


# Using Synchronization

As described earlier, the synchronization process is launched between two given servers: source (providing the information); and target (receiving the information). The target server is the driving server; it actively connects to the source server and pulls the necessary data. Synchronization can be bi-directional, but for each direction, a separate synchronization must be launched. Both directions are independent of each other and can be present or absent as necessary. Several synchronization processes can run in parallel on the same target server, pulling data from several different source servers.



The synchronization process takes the project history data (including the project and category names, all the commits, comments, and branches) from the source server and uploads it to the target server. This process is incremental, so only the changes from the last synchronization are transferred. Target server users can then access this data in the same manner as local projects of the target. These projects are listed among other projects, can be opened, or attached as used projects/libraries to other projects. The only limitation is that they cannot be edited and no further versions can be committed. Projects can only be edited on the home server. Project permissions of the source server are not synchronized - the target server administrator specifies the different access rules for the remote projects. Note that specifying write permissions on a remote project has no effect.

A remote project can be removed from the target server; however, please note that it will reappear after the next synchronization unless it is also excluded from the synchronized project set of the source server. The rename operations are handled in the same way; it is possible to rename the remote project on the server, but the name will be re-synchronized to the name in the source during the next synchronization.

Synchronization can be a long-running process, but it is not intrusive. The end users of the source and target servers do not feel its effects (no need to stop or restart servers during or after synchronization) except possibly some slight performance degradation.

The results of the pending synchronization appear on the target server gradually during synchronization. It is possible to see a particular remote project having 100 versions, then 110 after some time and 117 after synchronization is complete. However, this should not cause problems for clients on the target; partial version data will never be offered. For example, when there are 116 versions of a remote project declared, but the 116th version is only half-way downloaded, the 116th version will not be exposed.

**Related Pages:**



Unknown macro: 'list-children'