**Snapshot**: Whenever a user saves a tracked file to disk from within the Snapstore Folder, a snapshot containing that file’s contents is created and stored in the snapshot graph for that file. The user can revert the file to a previous state by navigating this snapshot graph and finding the correct snapshot. When the user reverts the file to this snapshot, the file’s contents are modified, making a new snapshot.

**Snapstore Folder**: The user can see every file accessible to Snapstore by looking in the Snapstore folder. There, users can change the tracking of files as well as edit those files to create new snapshots.

**Tracked File**: By making a file tracked, the user causes snapshots to be made of that file whenever edits of that file are saved to disk. These snapshots are stored in the local and upstream repository, and collaborators on the branch in which those snapshots were created can see them.

**Untracked File**: The user can cause snapshots to stop being created for a file by making it an untracked file. Untracked files will not be saved in the local or upstream repository, and all collaborators cannot see changes made to untracked files.

**Group**: A user can place logically related snapshots in a group to increase the organization of their branch. The user can then return to this group by searching for its group name, and all collaborators on that branch can see the organizational structure supplied by the group.

**Tag**: When a user places a tag on a group, it becomes findable by that tag’s name as well as the group name. This tag is also shared with all collaborators of that user. When the user reverts this tagged group, every file in that group is reverted to their snapshot in that group.

**Upstream Repository**: Whenever a user makes a change on a branch that other users have access to, that change is propagated by the upstream to those other users. Upstreams synchronize the changes made by collaborators so that each client has the same data.

**Branch**: When the user switches branches, Snapstore hides the old branch's data, shows them the current branch's data, and allows them to start adding data to the current branch. Branches separate data on independent lines of development for the user.

**Conflict Snapshot**: When the merge of two snapshots results in a conflict, a conflict snapshot is created. This conflict snapshot shows the user where the conflict exists using conflict markers. Fixing the conflict creates a new snapshot.

**Local Repository**: All saved changes made by the user are first stored persistently in the local repository, allowing them to work offline. When a network connection is restored, the local repository will push any data created while offline to the connected upstream.