

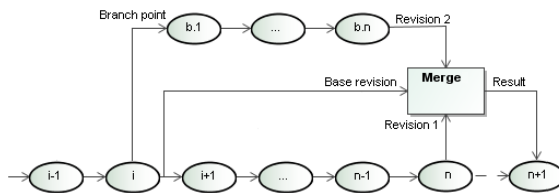
# Understanding merge types

There are two types of model merge: 3-way merge and 2-way merge. Choose the way of merge according to a particular case.

## 3-way merge

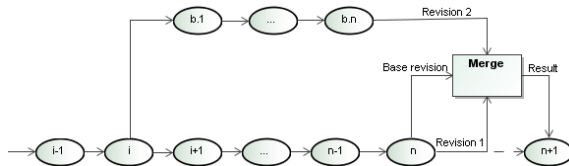
Conceptually, 3-way merge is a reconciliation of two difference sets, if we assume that a *difference set* contains changes between a contributor and the ancestor. For example, a difference set v1-v2 represents changes between project v2 and ancestor v1 as well as v1-v3 does for project v3 and the same ancestor.

Use the 3-way merge, if you need to compare and merge changes from two projects into one and consider the ancestor of both projects. For example, you may need to merge two branches of the same server project that have been simultaneously developed by two different users in a collaborative environment, as shown in the following figure. As you can see, branch b.1 has been created from project version i. Then both versions have been developed in parallel: i+1, ..., n-1, n and b.1, ..., b.n. The merge operation copies changes that have been made in the branch version to the trunk version.



## 2-way merge

2-way merge is a specific case of the 3-way merge. Use this type of merge for joining together two versions of the same local or server project. The logic of this type of merge is the same as of 3-way merge, only the target project is taken as the ancestor.



## Related pages

- [Model Merge](#)
  - [Understanding change concept](#)
  - [Preparing for merge](#)
  - [Starting Model Merge](#)
  - [Analyzing and managing merge results](#)
  - [Finishing merge](#)