

# Project partitioning

In life, all large things are built from smaller parts. This also applies to large projects. For large models with several weakly dependent parts, it is advisable to split them into several separate project files (or project resources in a collaborative environment), also called used projects.

Project partitioning opens up additional possibilities, such as:

- Reusing the same model part (e.g. library) in several projects.
- Separate administration of different project parts. You can define different access rights in the collaborative environment for each used project. Separate developers or teams of developers can be assigned for each resource, clearly establishing responsibility boundaries and preventing inadvertent modifications when these projects are used in the read-only mode.
- Separate versioning. Each used project can have its own version history with its own tagging and branching, life cycle of development, feature-freezes, and stable releases.
- Increased performance on very large projects. Whenever used projects are marked as loaded manually, a user can selectively load only the necessary parts of the project. He/she can also open the isolated used project separately.

A model can be decomposed into separate projects. A used project is a part that can be separated from the main project and can be used for a specific purpose.

MagicDraw supports the following used project accessibility modes:

- Read-only
- Read-write

The decision to use a project as read-only or read-write depends on the maturity of a project and the organization ownership or responsibility rules for projects using other projects. If the library in the used project is complete (changes to it are not expected / likely / possible) it should be used in read-only mode. If a project is developed by a separate team and this team is responsible for this project and the project is reused in another project, it is recommended to use this project as read-only. This prevents inadvertent changes in the project by another team.

If a project is actively developed and evolves together with the projects that are using it, this project can be used in read-write accessibility mode. In this case you must be careful and remember that your changes to the used project will be reflected in other projects. Using Teamwork Cloud might be advisable in this case. And, of course, there can be mixed usage situations - when a project is used as read-only in some projects and as read-write in others.

Instructions on how to partition your project can be found in the following pages:

- [Decomposing model](#)
- [Composing model](#)
- [Managing used projects](#)
- [Automated project usages](#)
- [Maintaining decomposed model integrity](#)
- [Validating project integrity](#)