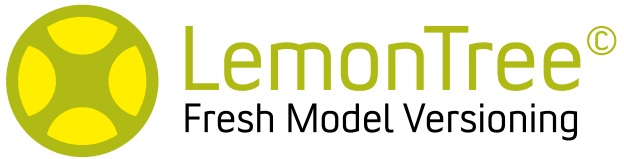


LEMONTREE USE CASES

CONFIGURATION MANAGEMENT FOR MODELS





Responsible for content:



LieberLieber Software GmbH
Handelskai 340, Top 5
1020 Vienna, Austria
+43 662 90600 2017
welcome@lieberlieber.com

CONFIGURATION MANAGEMENT FOR MODELS

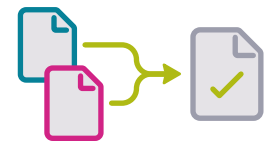


LEMONTREE USE CASES

Software configuration management makes changes in the development of complex systems controllable and traceable. The following methods have been developed over time:

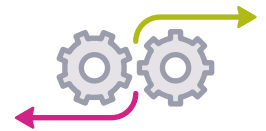
Use Case 1: Versions

6



Use Case 2: Builds

8



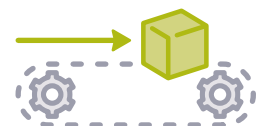
Use Case 3: Changes

10



Use Case 4: Releases

12



Good configuration management in a company has the following advantages:

▶ **Changes are controllable and traceable:**

Each change is carried out in isolation and transferred to the production model after a release process.

▶ **Product configuration is always under control:**

A released state of the model becomes a component of the product configuration.

▶ **Improved transparency:**

Different product configurations can be managed and maintained in a structured manner, the current development can be compared with the requirements.

▶ **Quality assurance:**

Only the model changes that have passed the specified quality checks (review, validation, etc.) are included in the product configuration. Any potential error can be easily identified, investigated, traced and corrected as needed

▶ **Productivity increase:**

Changes can be made in parallel and in isolation. Each model change can be transferred to the desired product configuration in a targeted and flexible manner. The use of tools established during software development makes team communication simpler.

Configuration management is also important for models

Configuration management evolved in the aerospace industry in the 1950s and 1960s to make the process of developing complex systems controllable and repeatable. The principles were later adopted by the software industry and developed further. Today, several configuration management methods are state-of-the-art in software development: change, version, build and release management, as well as continuous integration.

However, when developing complex cyber-physical systems the entire development cycle must be considered. Configuration management must therefore incorporate all development-relevant artifacts, from requirements and modelling, to documentation and source code. This is also demanded by common functional safety standards such as ISO 26262. Here models play a central role by enabling traceability between the individual artifacts of the development process. In the following use cases we outline our solution offering for each sub-area.

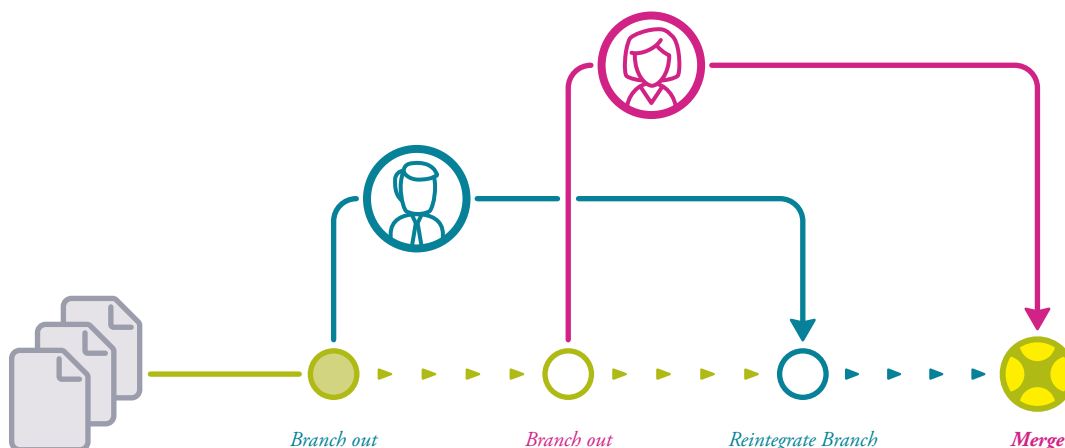
VERSION MANAGEMENT OF MODELS WITH STATE-OF-THE-ART VERSIONING SYSTEMS

A model-based development approach allows teams to work efficiently on parallel versions of a model. LieberLieber LemonTree for Enterprise Architect offers detailed model comparison and guarantees the precise versioning of models defined by modelling languages such as UML, SysML, BPMN, etc. (OMG open standards).

The key feature of LemonTree is the ability to compare and merge different model versions, allowing users to easily trace changes to the respective model versions. LemonTree offers the four key components of a modern model insurance system:

- ▶ Each change can be made in isolation
- ▶ The status before/after a change is archived. This allows each change (changeset) to be explicitly referenced and documented
- ▶ Review and release of each change is represented by the changeset
- ▶ The validity of the models is always guaranteed

LemonTree is seamlessly integrated into existing versioning systems. Our customers have already successfully used LemonTree with SVN, Git and PTC, and benefit from the features they offer. The model versions are managed centrally and automatically, enabling parallel processing of the models in a team, as has been the case for many years in source code development.



*Example of
application in
practice*

HIMA Paul Hildebrandt GmbH

The HIMA Group is a provider of smart security solutions for industry. Around 60 developers use Enterprise Architect to create models and generate software code for embedded and PC applications. As adherence to the functional safety specifications during development is required, exact traceability is an essential criterion for model versioning. Stefan Müller, Team Leader of Development Software at HIMA:

„In general, standards such as IEC 61508 require the existence of a configuration management system. This applies to all elements, including the UML models. For us, LemonTree from LieberLieber is the key to determining what was changed in which revision.“

One of the major challenges at HIMA was to integrate Subversion (SVN), a software for central version management of files and directories. As the developers create five to ten new versions a day, the integration process must be fast. Oliver Mummmenthey, software developer at HIMA, explains:

„The LieberLieber team was able to hugely increase the performance of LemonTree for our models of up to 6 GB in size. As a result of our close cooperation, we achieved excellent results in the project, strengthening our trust in LieberLieber immensely.“

*Special
highlights*

LemonTree offers decisive advantages for model versioning:

- ▶ **Revision security according to ISO standards**
- ▶ **Traceable changes**
- ▶ **Parallel, independent teamwork**
- ▶ **Version management of the models using tools such as SVN and Git**
- ▶ **Model Branching**
- ▶ **Change history available**

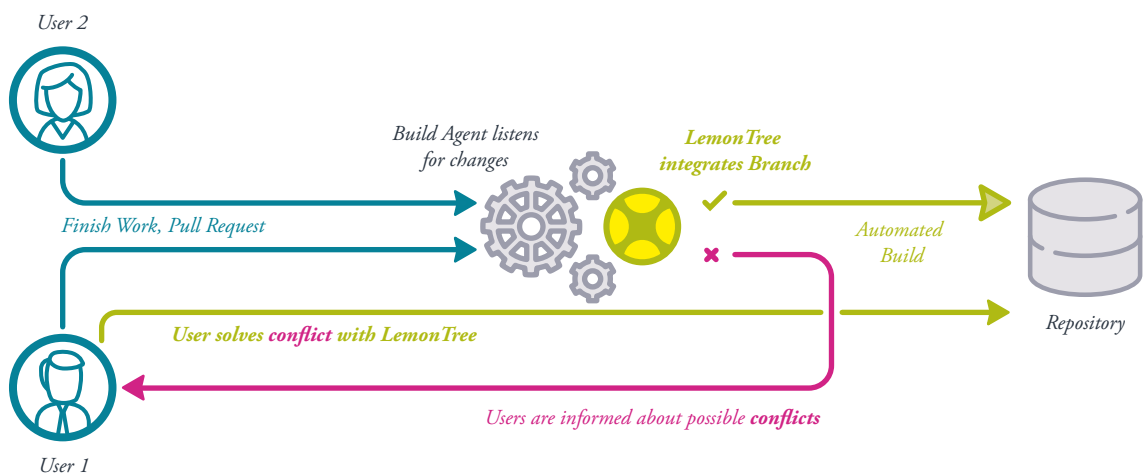
*Benefits
for users*

System engineers and modellers enjoy a high degree of security when using LemonTree to develop complex mechatronic systems. Proven concepts are applied by outsourcing version management to tools such as Git and SVN.

These tools show which user has changed model content, and when. They also allow a team to make parallel changes to models without affecting the work of other team members. Users can reference specific revisions in order to ensure an orderly change and reporting process, also for models.

BUILD MANAGEMENT - CONTINUOUS INTEGRATION OF MODELS

Continuous integration is the most advanced configuration management methodology in software development. Each released change in the source code is continuously merged in a repository and then automatically checked, formed, tested and packaged on a build server. This ensures that the developed software is always tested and executable. LieberLieber LemonTree allows the continuous integration method to be used in the modelling world for the first time. Similar to traditional software development, team members can work on different areas of a model and make their changes in a distributed and time-shifted manner. LemonTree Automation then merges the individual versions on the build server (e.g. Jenkins) into an integrated model version without the need for any user interaction. Only in complete conflicts, i.e. when contradictory changes to the same model element occur, are the affected users notified and required to resolve these conflicts manually using LemonTree.



Example of application in practice

A leading German automotive supplier

Improvements in software and system development are increasingly becoming the focus of strategic considerations, particularly in the automotive and automotive supply industries. A department of one leading German automotive supplier has taken the courageous and forward-looking step of converting its customary document-based software development into a completely model-based process.

„Our goal is to make the model the core of our specification. We also want to work with the models as comfortably as with code in agile software development. In addition, in a project with the Technical University of Munich we have developed our own method which will be applied in the models“, explained the project manager, outlining the stringent specifications.

Example of application in practice

In this project functioning configuration management for the models has been identified as a decisive factor in the overall success.

The goal was the automated and continuous integration of models, or „continuous integration“. The decision was taken to transfer the established methods used in software development to the modelling world. Changes to the model are continuously integrated into a baseline, but, unlike in the past, with precise control according to the specified process.

Each change is ordered via a ticket system and goes through several steps:

1. Specification
2. Implementation in a versioning system
3. Review
4. Release
5. Automatic integration into the productive model by a build server

A particularly challenging aspect of the collaboration with the automotive supplier was to automatically transfer the modified model versions to the central model with every check-in by an architect (Build Management - Continuous Integration of Models). The company uses a build server that automatically transfers the changes ordered, executed and released via ticket into the productive model on a regular basis. Previously this required the manual merging of the changes in LemonTree, but this is now also integrated into the automatic process.

Special highlights

The use of LemonTree in the build management of models offers decisive advantages:

- ▶ Automatic merging of changes
- ▶ 3-way comparison/merge
- ▶ Conflict resolution by power user
- ▶ Change history available

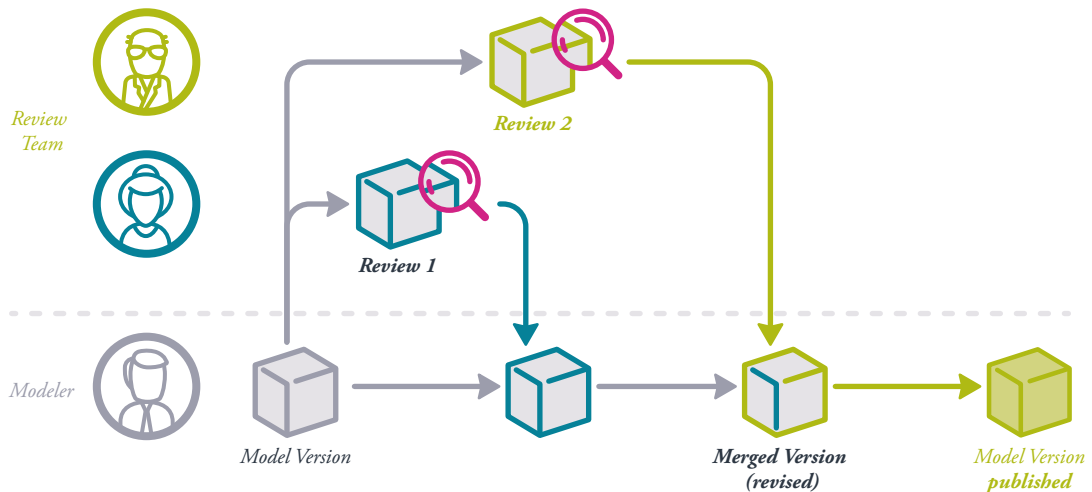
Benefits for users

Continuous integration for models using LemonTree ensures that the modellers can work independently but concurrently on several changes, and always with a consistent model level. The LemonTree merge functionality allows them to merge different model versions quickly, securely, with built-in support and without conflict.

CHANGE MANAGEMENT AND REVIEWS OF MODELS

Functional safety standards such as ISO 26262 require configuration and change management for all development-related artifacts. Models are now an integral part of the development process, especially for safety-relevant mechatronic systems, and must therefore be subject to appropriate change management. Thus special attention must be paid to changes to the models which describe the artifacts being developed („first-class entities“).

With LieberLieber LemonTree, these changes are prepared in a comprehensible way and can be queried at any time. It is not only the current state of the model which is important, but also the changes to the model and differences between individual versions of the model. Indicating which user has changed what in the model, and when, not only helps in teamwork, but is also indispensable for functional safety. LemonTree also makes it possible to review these changes. The ingenious display of the changes and the filter options in LemonTree facilitate this work enormously. Furthermore, LemonTree offers sophisticated features for saving a review configuration, including the model versions requiring review, and the current status of the review. This makes it possible to pause the review process at any time and subsequently continue without any loss of information. Last but not least, review comments can be clearly added to individual branches in order to optimise the review process.



Example of application in practice

A leading German automotive supplier

In the automotive industry in particular, verifiable compliance with functional safety standards such as ISO 26262 is essential. Consequently, manufacturers and suppliers are also obliged to perform configuration and change management in models.

Example of application in practice

„For us, safety standards, including Automotive SPICE, and compliance with the Automotive Safety Integrity Level ASIL, are decisive factors in software development. The Enterprise Architect modelling platform has been successfully used in the Software Architecture department for some time now, and we are now convinced of the benefits of model-based software development. But it is only through contact with LieberLieber and the use of LemonTree that our daily work with the models has been considerably facilitated and accelerated“, says an employee of a leading German automotive supplier who works in the functional software department in creating safety-critical software.

Reviews are particularly important for the working group, demonstrating compliance with all

legal requirements even when changes are made to the software. The software development work results are checked manually to ensure the required quality.

„At our company every change in the model - also known as a change request - must be subjected to a review. LemonTree offers an excellent means of doing so, which greatly accelerates our work“, the employee says. Whereas previously only small changes between two model versions could be reviewed, LemonTree enables reviews of more complex models and supports the work with many useful features.

Special highlights

The use of LemonTree for reviewing models offers decisive advantages:

- ▶ **Revision security according to ISO standards**
- ▶ **Traceable changes**
- ▶ **Merge review comments**
- ▶ **Additional checkmarks for filtering already reviewed changes**
- ▶ **Filter options in the change list**
- ▶ **Export the changes as CSV**
- ▶ **XML export for custom change visualisations**
- ▶ **Change history available**

Benefits for users

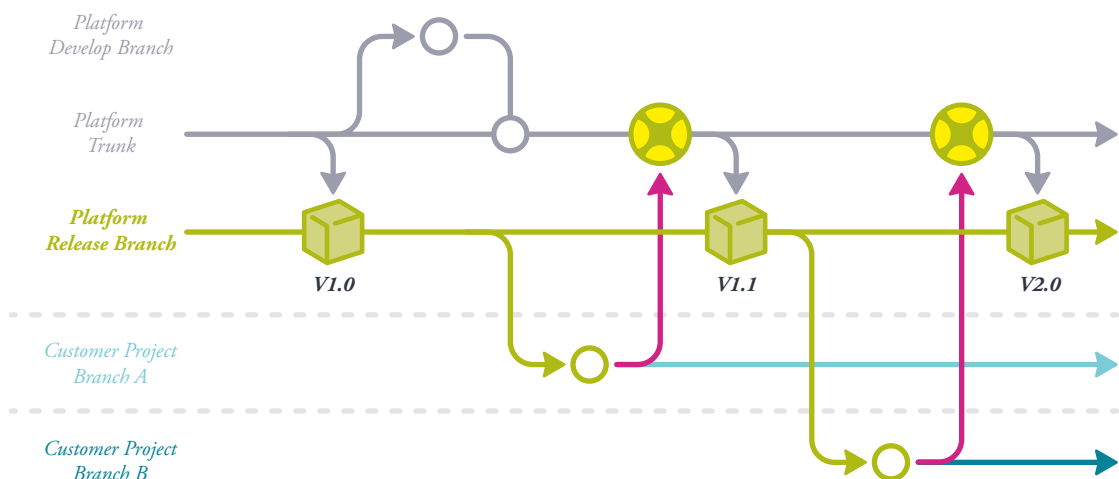
LemonTree gives modellers and reviewers the assurance that all changes to the UML or SysML models are reliably merged, with each change being precisely documented. Complete information about which user has changed exactly what, and when, is extremely important when developing security-relevant artifacts and in other areas. The model changes are presented in a comprehensible manner and can be subjected to a controlled review at any time. Supplementary review comments are maintained in a separate model version, guaranteeing the user that the productive model will not be destroyed. If several reviewers maintain such comments, these can also be easily merged with LemonTree. Various export options (e.g. XML) serve to document changes over time.

RELEASE MANAGEMENT FOR MODELS IN PRODUCT LINE DEVELOPMENT

Cost and innovation pressure are key drivers in the switch to model-based product line development. Close cooperation between platform development and the various customer projects is necessary to shorten development cycles. Completely new challenges arise at the model level for complex products with a large number of components, and when supporting specialist groups which formerly worked on the basis of a common model. It is difficult to establish the so-called „basic software“, which forms the basis of the individual product lines, as an independent platform and to adapt and expand it in individual customer projects. It is important to be able to use established software development methods in order to consider the different lifecycles of the projects and the platform.

With LemonTree, LieberLieber has developed a solution for modelling with Enterprise Architect, specifically for the introduction of model-based product line development with a package management system (MPMS). This allows distributed teams to work on different model versions while maintaining the „basic software platform“ in a consistent state. The central idea of this concept is to apply existing approaches already established in source code configuration management.

Parts of the model that have been cut out of the platform development can be used as components in project development without losing references, and at the same time changes can be merged in both directions.



*Example of
application in
practice*

Example: Preh Car Connect

Product line development supports companies by reusing or varying software that has already been created within the framework of a platform. The resulting variants differ in their functional scope, but the basic software components are used again and again. Tim Michaelis, software architect at Preh Car Connect:

„Since we are growing strongly within the company and implementing many series projects, components of our own software can be used again and again. So far, however, this has only been possible with time-consuming manual copying, especially of the models. That’s why we are working on a project together with LieberLieber to automate such processes. LemonTree is an important building block which will allow us to implement these development steps faster and more easily.“

Preh Car Connect has around 1,200 employees at eight locations worldwide. The company's headquarters and central development site are located in Dresden. For more than 20 years the company has been developing and producing high-quality infotainment systems for renowned OEMs, including the Volkswagen group.

*Special
highlights*

- ▶ **Versioning at package level**
- ▶ **Package management based on Git, etc.**
- ▶ **Update of a single package using LemonTree-Merge**
- ▶ **Dependency analysis between packages**
- ▶ **Release of certain package versions**

*Benefits
for users*

In this scenario, users can be certain they are using specific model artifacts in a defined version, and referencing them exactly. They can import individual versions of a specific package into a model without destroying their previous developments as LemonTree intelligently merges the parts. This allows a team to reuse individual parts of the development in different projects.

ABOUT **LIEBERLIEBER** SOFTWARE

We are a software engineering company. Our employees' expertise lies in model-based software and system design using tools such as Enterprise Architect from Sparx Systems.

Our customers are companies prioritising the quality of their software and systems development. They want a constant overview of their complex development scenarios while ensuring that security-relevant requirements are clearly represented in models.

We achieve this using our own special tools, including LemonTree and Embedded Engineer. We also offer a range of tool integration services to make our customers' development processes more productive.

More information:

www.lieberlieber.com

