DELTA software technology

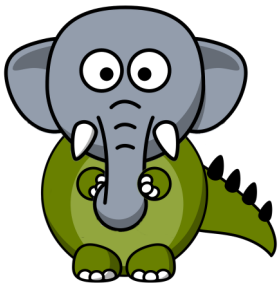
AMELIO
Modernization Platform

Hunting the Technology Zoo: Automated Technology Change

The longer an application lives, the greater the probability that it uses different technologies for one and the same task or a technology that endangers the usability of the application. In order to guarantee the maintainability, performance and future security in the long term, these technologies need to be consolidated or replaced. With a rule-based factory, such a technology change can be performed successfully during operation.

The Technology Zoo: Crocofants and Dinosaurs

Applications often live significantly longer than expected and they usually also live longer than the technologies they contain. Over time, new technologies come onto the market and they offer new features all the time. These new technologies are then used to develop new (parts of the) applications. By using old and new technologies in one application, a technology zoo with crocofants and dinosaurs emerges over time.



Crocofants: Sometimes the old technology is at least partially replaced by a new one. But only in the rarest of cases everything is completely replaced. The result is a co-existence of different technologies for one and the same task within an application landscape.

At some point in time such a mix of different technologies leads to problems:

- If the technologies have to interact with each another, the performance of the application suffers as a result.
- The application is becoming increasingly difficult to maintain.
- Experts must be available for all technologies used.

Dinosaurs: If an old technology remains in an application because it is not completely or not at all replaced by a new one, this can endanger the future security of the application:

- The old technologies do not have all features that the new technologies support. As a result, the applications cannot be adapted to new requirements or only with great effort.
- A platform change (whereby a platform can be a combination of hardware, compilers, DB and TP system) is made difficult or completely prevented if a technology is not available for the new target platform.

Hunting the Technology Zoo: Automated Technology Change

DELTA software technology

To solve these problems, it is necessary to completely replace an old technology or to consolidate several technologies. However, the technologies are usually firmly anchored in the applications. And these applications are large, complex, and business-critical. If you wanted to perform the replacement manually, this would be doomed to fail: effort and risk would be enormous, necessary tests could not be performed completely or would be very time-consuming. In addition, regular maintenance would be massively impaired and the application or parts of it would have to be frozen over a long period of time.

A flexible, automated solution, on the other hand, enables a successful replacement with an acceptable risk and test effort, parallel to ongoing operation.

Automated technology replacement with AMELIO-Factory

For a secure and efficient technology replacement we rely on the AMELIO Modernization Platform, a factory that is precisely built for the respective task, the project or the customer. Through targeted analyses, the factory creates different models, such as the Extended Abstract Syntax Tree or control and data flow models, for all artefacts of the application. All of the following steps are then performed on these models.

The technology replacement is performed rule-based in the three steps discovery, analysis and transformation on the previously determined models.

- **Discovery**: In this step, all points in the application that may need to be transformed are determined. These candidates are called Points of Interest (PoI).
- **Analysis:** In the analysis, all PoI determined in the discovery are analysed in more detail and their context is evaluated. It is determined whether a transformation is required and if so, which one.
- **Transformation**: The actual transformation is then performed depending on the result of the analysis. For this purpose, the models are modified using transformation rules and the new code is generated.

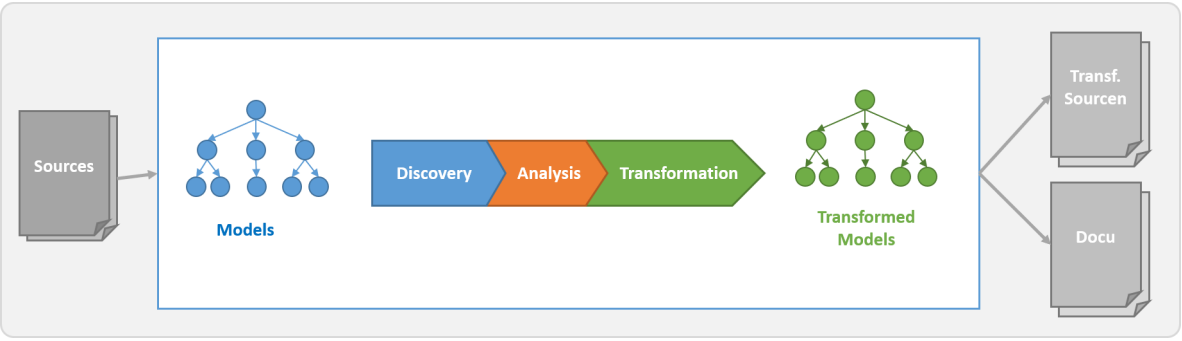
Due to the model and rule-based approach, the transformations are absolutely uniform and can be reproduced at any time. The reproducibility makes it possible to divide the application into packages that are processed one after the other - parallel to ongoing operation and considering the changes caused by regular maintenance.

A solution that fits 100%

Would a standard solution fix the problem? Certainly not. Differences between applications and their environment (sometimes even within a company) mean that no two projects are absolutely identical. Therefore, the factory must be configured precisely for the upcoming task.

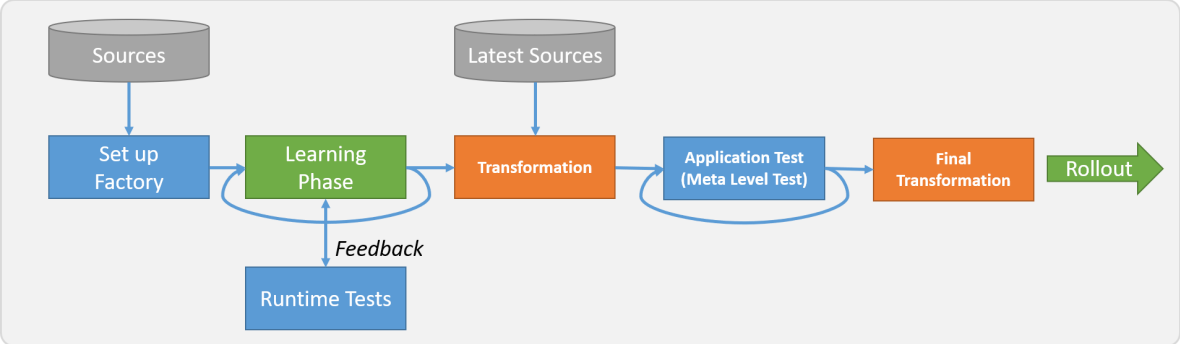
The factory is built with the AMELIO Modernization Platform from prefabricated components and subsequent step-by-step adaptation to the task and other project specifications. For this purpose, a basic factory is built in a first step, which can read the sources and generate the corresponding models. The next step is to gradually configure the project-specific rules for discovery, analysis and transformation. With the transformation rules, it is important to observe company policies for the new generated or changed code. If there are different options for a transformation, a decision is made during project which is the best for the present task.

In order to get the first results as quickly as possible, the factory is trained using examples, starting with examples where only a few different transformations are necessary. After that, the rules in the factory will be gradually extended. As soon as these examples have been transformed, the tests are performed. If the



results of these tests do not meet the requirements, technically or with regard to performance, the corresponding transformation rule is adjusted. Parts that have already been transformed can, but do not have to, be transformed again with this new knowledge.

After a successful test, the current source status is imported into the factory again and the transformation will be performed. An extremely short freeze, e.g. a weekend, is only necessary at this point. Only these transformed sources are then transferred to production.



In packages, steps or big-bang?

Applications in which a technology should be replaced, are usually very large. Converting, testing and setting such an application into production in a big bang, is hardly possible in many cases. Most of the time it is advisable to divide the application or task into smaller and more manageable parts. You can make use the fact that the results of the factory can be reproduced at any time. Both functional and technical dependencies can be used to divide applications into smaller parts.

- **Package-wise:** Based on the dependencies, the application is divided into packages that can be converted independently of one another. When teaching the factory, the packages that require as few different changes as possible are started and then gradually extended. However, all necessary changes for the respective package are implemented directly before the package first goes into testing and then into production.
- **Step-by-step:** For some changes it is necessary to implement them directly in the entire application. In this case, it makes sense to divide the task into several steps, but in every step a change is implemented in the factory and then applied and set into production for the entire application. Further changes are then implemented in subsequent steps. This procedure means that the software may have to be set into production several times, but the number of changes implemented can be limited.
- **Hybrid**: A mixed approach, in which some changes are implemented in packages and others step-by-step, can also be implemented with a flexible factory.

Meta level test: Test rules instead of changes

Technologies are usually firmly anchored in an application. A technology replacement can therefore result in many thousands to millions of changes. In case of a manual procedure or a classic test procedure, each of these changes would have to be tested.

However, the test effort can be drastically reduced without losing security by taking advantage of the fact that the AMELIO factory works rule-based and thus delivers uniform and reproducible results. Because: Once a rule is correct, it is correct even if it is applied several times. It is therefore not necessary to test all changed programs and all implemented changes, only all rules defined in the factory. By selecting a suitable test set that covers all the rules but only represents a comparatively small section of the application, the test effort can be significantly reduced.

Flexibility to the end

But the AMELIO factory scores not only in terms of security, but also offers great flexibility

- **“Point of no Return”**: Compared to a manual procedure, there is no or only a very late “point of no return”. If it turns out in the course of the

AMELIO
Modernization Platform

DELTA
software technology

project that a decision made at the beginning was not optimal, the corresponding rules are adapted and the sources are simply transformed again.

- **Freeze**: While the factory is built and trained, the regular maintenance and further development of the application can continue unhindered. A short freeze is only required when the factory training (for the current package) has been completed. Then the current source status is imported into the factory and transformed. With maximum parallelization of the factory processes, even very large applications can be transformed within a few hours or over a weekend. After the transfer to production, the regular maintenance can be resumed.

Technology change: automated and during operation

A technology zoo with crocofants and dinosaurs doesn't have to be. A technology change is also possible for large and complex applications during running operation if an automated approach is chosen. With AMELIO's rule-based approach, a factory can be built that is precisely tailored to the task. All transformations performed can be reproduced at any time. This makes it possible to transform the application step-by-step or in packages. Above all, however, regular development can continue unhindered. Changed sources are simply imported into the factory and transformed again. The automated removal of the crocofants and dinosaurs creates a clean technology landscape that enables efficient development and maintenance of the application and also ensures that the application is future-proof.

Delta Software Technology

Delta Software Technology is a specialist for generative development tools that automate the modernisation, integration, development and maintenance of individual IT applications. Our solutions help you to quickly and safely adapt your applications to new business requirements, architectures, technologies and technical infrastructures.

Delta has a more than 45-year track record of successfully delivering advanced software technology to Europe's leading organisations, including AMB Generali, ArcelorMittal, Deutsche Telekom, Hüttenwerke Krupp Mannesmann, Gothaer Versicherungen, La Poste, RDW, Suva and UBS.

Get in touch with us

Delta Software Technology GmbH
Eichenweg 16
57392 Schmallenberg
Germany

Phone +49 2972 9719-0
E-mail info@delta-software.com

delta-software.com

Order number: MT 21'095.01 – January 2022