

# THE IT-ARCHITECTURE PROFESSIONALS

## High-End Application for Billing on “Per Usage” Basis

### Industry

Transportation, National Austrian Railways

### Application

- J2EE application for determining charges on a “per usage” basis
- Integrated with an input application providing schedule data, various operative systems for handling schedule and train data, transfer of data to SAP system

### Business Benefits

- Immediate savings of 35% in the development process
- Optimized customer service with billing system that reflects the actual usage of the railway infrastructure
- Optimized internal workflows, process automation

### Technical Benefits

- Uniform and easy-to-maintain component infrastructure
- Implementation of a future-oriented corporate IT-architectural style
- Competent consulting and full service support by Interactive Objects

### Austrian Railways

With a staff of more than 49,000 and a railroad network of 10,000 kilometers, Austrian Railways (Österreichische Bundesbahnen ÖBB) is one of the largest transportation enterprises in Europe and a major building block in the trans-European railroad network.

The ÖBB Netz division with a staff of 12,000 is responsible for track management and planning. In addition, this division serves the function of a One-Stop-Shop for all users within the framework of the Freeway Projects.

### The New ÖBB Application

The LDZ “Leistungsdaten Zugfahrt” application was designed to implement a billing logic that reflects the actual usage of the railway infrastructure and provides customers with invoices showing clearly itemized travel records. The LDZ application, one of the largest J2EE/EJB systems ever built in Europe, is a comprehensive system based on a distributed four-tier J2EE infrastructure of components and Web applications also serving as a main application integration component for diverse distributed legacy systems. It is the first application of its class in Europe and the first system to fulfill the billing requirements for rail infrastructure usage as required by the European Union.



AWARDS 2002

Finalist

**“Immediate savings in the development process due to MDA-compliant approach and the usage of ArcStyler amounted to approx. 35% compared to the estimated figures for development without using this approach and IDE.”**

Österreichische Bundesbahnen

### The New ÖBB IT-Architectural Style

However, this project also served a more general and very far-reaching purpose: the implementation of a new cross-corporate architectural style for all future IT projects at ÖBB. In addition to allowing for easier application integration including vendor and customer system interfacing, the objectives of such a style comprised shorter time-to-deployment, increased system quality and future-safe development of easy-to-maintain component infrastructures.

### Headquarters

Interactive Objects Software GmbH  
Basler Str. 65 • D-79100 Freiburg  
Germany • Phone +49-(0)761-400 73-0

Interactive  
Objects

[www.io-software.com](http://www.io-software.com)

# Controlling the Train Costs with LDZ

## Scheduling Trains

The ÖBB has approximately 15,000 scheduled track segments available. These tracks are used by an average of 8,000 trains per day. 90 percent of these trains are scheduled, leaving 10 percent that are not, either because additional trains are used to meet ad hoc demands, or because trains must

the data and return the resulting actual status information such as weight, time, distance, etc. to the LDZ application.

## Generating Invoices

Approximately 50 records with actual data are recorded for each train, i.e. the system processes an average of 400,000 records per day with schedule and actual data. This data is used to create the records for the invoices. LDZ verifies the information in terms of plausibility and completeness

so they are modeled in a single representation that can be understood both by the railway and the IT experts. At an abstract level, this model is completely independent of any special technology platform. From here, the IT people take over and add technical detail at various levels of model refinement.” This ensures that the entire system logic is decoupled from technology life cycles. The models can be used to map this logic to specific technology platforms by means of automatic code generation mechanisms.

## ArcStyler Becomes New ÖBB Development Platform

The LDZ application was the first project to be realized according to this model-driven approach. “We chose ArcStyler from Interactive Objects as our development platform because it supports all stages of model driven development,” explains Erich Auer, ÖBB Project Manager. “In addition, Interactive Objects is not just a tool vendor, but a competent IT solutions provider. Their support ensured a fast and smooth start for our team, and they provided comprehensive consulting whenever we needed it,” adds Erich Auer.

## Modeling the Business Dimension

The ArcStyler’s C-BOM business object modeler module is the first significant step toward the automatic generation and deployment of a convergent system. This tool assists in requirements assessment, domain modeling and the creation of the initial business object and application model. C-BOM supports these critical steps using the substantiated techniques of Use Case-based Analysis By Design, Responsibility Driven Design and graphical Use Case scenarios, backed up by rule-based model verifiers. ÖBB used the module at this abstract modeling level in order to create some 15 to 20 basic components for the application.

## From Business Abstraction to UML Detail

The ArcStyler’s C-RAS module bridges the gap between business modeling and UML refinement. It graphically assists the designer in refining the responsibilities and collaborators of the business objects into UML attributes, operations and associations. The result of this refinement step is a convergent design, which preserves direct visibility of the original business model in the resulting UML component model.



be rescheduled due to delays and other events. This means that ‘plan’ data received from ROMAN-D, the ÖBB’s scheduling system, must be complemented with new data. This new schedule information, i.e. set values, must be introduced into the system and coordinated with the existing data.

The schedule information received is processed by LDZ. Recurring tracks (i.e. trains that run at regular intervals) are automatically activated. LDZ also serves as the operative basis for planning, assembling and activating tracks for all additional trains not running according to the standard schedule.

The complete train and schedule data processed by LDZ is transferred to two operative systems controlling the actual train traffic: ARTIS (Austrian Railways Transportation Information System), and RZÜ (Rechnergestützte Zugüberwachung (Computer-Controlled Train Monitoring)), a field monitoring system. Both systems process

before generating XML files which are then used by an SAP system for billing. ÖBB uses the actual data with the detailed weight, distance and time information to determine the charges for each train. The customers can now receive monthly invoices showing detailed records of their usage of the ÖBB infrastructure.

## Controlling the Development Costs with ArcStyler

### Model Driven Architecture and Convergent Architecture

Says Werner Wüster, ÖBB Organization Management: “We opted for Convergent Architecture because the logic behind it is absolutely compelling. The business dimension and the IT dimension of an application are two sides of the same coin,

## Use Cases and the Accessor Framework

While a number of the basic Business Objects were modeled with the C-BOM module, the majority of user interaction objects were described as so-called Use Cases and then modeled with UML. These Use Cases were mapped to so-called Accessors, usually one or two Accessors per Use Case. Accessors represent the dynamic process of user interaction (e.g. assembling a track segment) with all state transitions. An Accessor has several Representers, which automatically provide a designed user interface for user interaction in the form of HTML pages. The ArcStyler's patented Accessor Framework enabled the fully automatic generation of the complete Web application infrastructure (100%) for all these components.

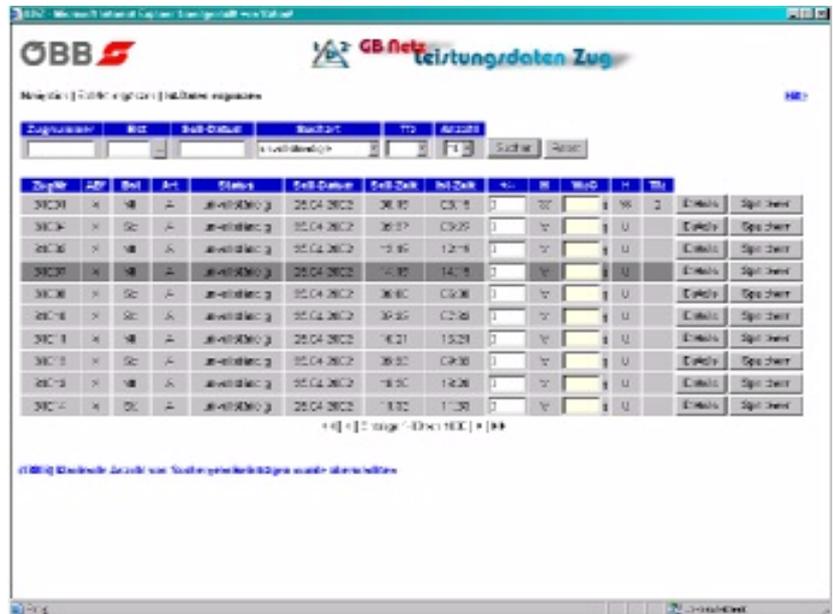
## J2EE/UML Refinement

ÖBB used the ArcStyler's C-REF UML refinement module to refine the UML models. The refinement activities included the modeling of attributes, operations, associations, dependencies, inheritance, object behavior, EJB properties, container-specific properties, database mapping, J2EE-compliant Accessors, deployable components representing the physical deployment structure of the component system as well as test cases and test classes.

This resulted in highly detailed UML models which serve as the basis for the automatic generation of instantly executable systems. ArcStyler's patented design for capability-sensitive UML-debugging with verifiers allowed the ÖBB developers to immediately check whether the current model adequately supported the configured J2EE/EJB infrastructure. Erich Auer summarizes the advantages of these features: "There was no need to build the system just to find out something in the blueprint did not work properly. This just-in-time feasibility check at UML model level improved both the quality and speed of development across the entire project."

## The Full Generative Power of the ArcStyler Approach

ArcStyler uses so-called MDA-Cartridges as the automation engine for code generation. As opposed to previous CASE approaches, these MDA-Cartridges are completely open and can be adapted and extended as required. This way, the MDA-Cartridges provide an extra layer of flexibility between the UML model and the low-level code. They enable



the automated production of architecture-driven code by enforcing patterns and coding guidelines across all modules. The MDA-Cartridges also ensure a unique location for all changes: modifying a single generator template changes all files generated by that template, no matter how many projects are involved. Finally, the MDA-Cartridges assure architectural integrity by providing a single place and a powerful vehicle for the IT architect to express structures and patterns that define the system structure at all levels. ArcStyler automatically generated the complete EJB infrastructure optimized for the Application Server, the Java classes, the Web application with Accessors and Representers (HTML pages and JSPs/Servlets) as well as the test classes and Build scripts for ANT. "The MDA-Cartridge already provided excellent results as it was. But as the ArcStyler follows an open approach, we were able to extend its functionality and add features to generate test classes and Build environments that were tailored exactly to ÖBB-specific requirements," says Werner Wüster.

## Optimizing the Application

In terms of performance, there may be faster approaches than the mapped EJBs. Explains Erich Auer: "Maximum performance is only one aspect in the design of a software application. Other important factors include development time and maintainability. And in view of this big picture, we have the optimum balance of all factors. The first version of the system was up and running in a very short period of time. And after a few optimization efforts, we have very good performance."

These optimizations included the creation of special Data Access Objects containing the isolated code for the database access. In concert with some further optimizations (verification prior to creation of beans, transaction modes and number of beans in pool or cache), this ensures that the system performs well and excels with superior maintainability in spite of the complexity. The system comprises approx. 400,000 EJBs. These are not all concurrently active, but a large portion of them may be, and the system has been tested for adequate performance with large numbers concurrently active.

## Optimizing Teamwork

One of the main problems in managing large-scale development projects with distributed teams and external partners is the coordination of the people and organizations as well as the results they produce - usually a major cost and risk factor. The ArcStyler's MDA approach tackles this problem at the source by raising the level of code and model management. It allows for the propagation and automatic enforcement of clean architectural and modeling styles.

"This modeling style ensured that everybody involved in the project spoke the same language. UML provides a sort of common alphabet. The ArcStyler's approach uses this alphabet to add a common language on top," explains Erich Auer. "Distributed development, automatic enforcement of modeling guidelines, automatic consolidation and integration of the results - the ArcStyler really boosted the productivity of our teams and streamlined the workflows."

# The Bottom Line - Convincing Benefits

## Clean Architecture

Erich Auer summarizes the experience gained in the project: "ArcStyler allowed us to implement a corporate architectural style and develop an application at the same time." The model-driven approach with the automatic generation of the components results in an enormous increase in productivity and quality. Continues Auer: "It's simple - whatever you don't have to type can't go wrong. The models, automatic verifiers and the generator allow us to propagate architectural guidelines directly into the code, i.e. no error-prone, paper-based directives."

## Efficient Application of Manpower

In addition, the model-driven approach allows the development team to focus on the actual assets of the application - the system logic. Explains Werner Wüster: "All developers could fully concentrate on the ÖBB-specific details of the application.

Little time needed to be dedicated to the technical aspects of the target platform as this was taken care of by the ArcStyler."

## High Degree of Automation

The model-driven automatic generation of the system turned out to be another time-saving factor. This also includes the automatic creation of the Build and Test infrastructures, as the effort that has to go into the Build and Test environments for a complex system is underestimated in many projects.

## High Savings

According to ÖBB, immediate savings in the development process due to the MDA-compliant approach and the usage of ArcStyler amounted to approx. 35% compared to the estimated figures for development without using this approach and IDE. The savings result from faster development, more efficient application of manpower, a high degree of automation in code and infrastructure generation and automated quality management. The MDA/UML model will result in additional long-term savings due to easier maintainability of the LDZ application. While it is difficult to quantify

these savings since there is no "non-MDA" application to serve as a basis for comparison, they will be substantial. Concludes Erich Auer: "With ArcStyler, we were up and running very quickly. The bottom line is that ArcStyler offers unparalleled and totally flexible support of an incremental development, deployment and test cycle for a great number of runtime environments while at the same time preserving future-safe model independence from the particular runtime infrastructure. ArcStyler will be our platform for the upcoming projects."

**"With ArcStyler, we were up and running very quickly. The bottom line is that ArcStyler offers unparalleled and totally flexible support of an incremental development, deployment and test cycle for a great number of runtime environments while at the same time preserving future-safe model independence from the particular runtime infrastructure. ArcStyler will be our platform for the upcoming projects."**

Erich Auer, ÖBB Project Manager

## The Development and Technical Perspectives

|   |   |
|---|---|
| <b>Number of Use Cases:</b>                       | 20  |
| <b>Number of components</b>                       |   |
| All types:  | 540   |
| Accessors   | 26  |
| Representers:                                     | 72  |
| Entity EJBs:                                      | 67  |
| Session EJBs:                                     | 49  |
| <b>Test classes:</b>                              | 60  |
| <b>ValueObjects (transp. Objects):</b>            | 100   |
| <b>Data Access Objects:</b>                       | 34  |
| <b>Number of external application interfaces:</b> | 5   |
| <b>Generated components</b>                       |   |
| EJBs  | 100% of Home and Remote interfaces                  |
| Java classes                                      | All operations and attributes with Accessor methods |
| Web applications:                                 | Accessors and Representers                          |
| Test and Build environment:                       | Test classes and ANT build support (100%)           |



[www.ArcStyler.com](http://www.ArcStyler.com)