

A Direct Line to Insurance – Thanks to Java



Requirements of an uncomplicated, user-friendly software system

By Ralf Loose

A direct insurer like Direct Line Versicherung AG has a particular need to offer its customers and potential customers optimum advice on the telephone or online. On the one hand, this requires competent call-center agents. On the other hand, however, it also requires a software package which can transfer customer data into the inventory system for the customer agent on the telephone reliably and easily. This project report gives you an overview of the requirements and the resulting development of such a software system with modern Java technology.

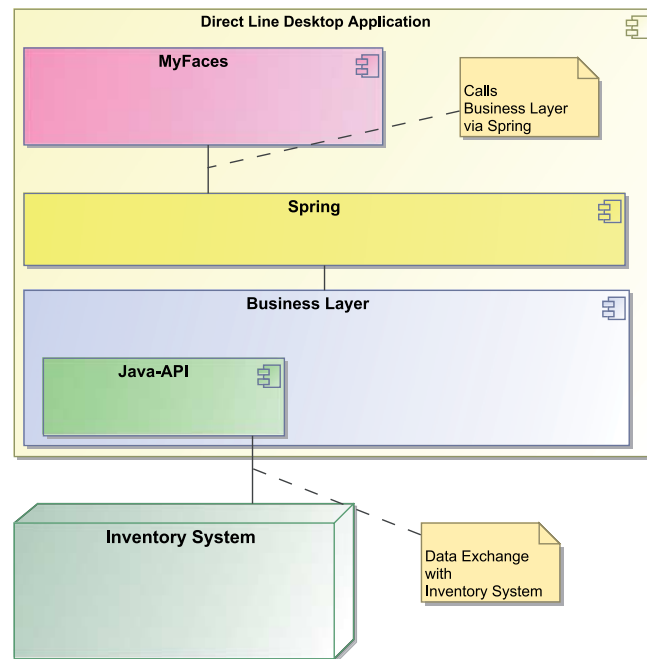
Requirements

The market for car insurance is competitive. Fast reaction times when responding to damage claims and uncomplicated further processing are significant competitive factors. A powerful software system is an essential prerequisite. For these reasons, Direct Line Versicherung decided on the modernization of their applications and contracted Consist for the new development. After a thorough analysis of the needs and technical conditions, these were the requirements determined for the new system:

- High degree of scalability
- High availability

- Very good usability
- Good user interface
- Easy extensibility
- Modularity

The high degree of scalability is needed because Direct Line Versicherung AG primarily offers automobile insurance. Since most of the business in that industry takes place about the end of the year, during that time the number of clients increases, along with the need for simple connection. The high availability is a result of the 24x7 business of the direct



Graphic: Consist Software Solutions

Architecture of the application

insurer, while user friendliness is of primary importance for staff on the telephone. Agents should only need to worry about carrying out their business tasks, not about learning a complicated software package. Thus the new system will walk the user through the process of concluding a policy in order to prevent data from being entered which is not allowed due to business rules. This corresponds to the desire for a good user interface. Simple extensibility is needed because Direct Line Versicherung AG is planning several new products which need to be integrated easily into the system. Modularity ensures that all products can use certain modules in common.

Another basic condition was that the system should be developed against a Java interface from a third-party provider, which handles the transfer of data into the inventory system via a CICS system (Customer Information Control System).

Technology

These requirements, in combination with company-internal specifications, led to the choice of implementing a web application in the Java Server Faces implementation of MyFaces. This ensures the high scalability of the application, which runs on Tomcat web containers. As a web application, the system is also easy to maintain centrally, and the connection of additional clients is no problem. This technology also allows the simple connection of cooperation partners

without any problems. Access must simply be secured using VPN or SSL.

Architecture

The application communicates with a Java interface which is also included in the business model, on which the client works directly. This “flat“ architecture, despite the need for several server calls for many client requests, ensures good performance, and as a side effect it speeds up development. The result is a modular service layer which encapsulates the calls to the Java interface and largely avoids business logic. This layer was implemented using the Spring framework, allowing the integration of additional services using dependency injection. Through this layer, other services besides the interface to the inventory system can also be addressed, for instance a print service which generates PDF documents, or a service which calls up data from the telephone system during incoming customer calls.

Furthermore, Spring-AOP is used to implement error handling in the business layer and the Java interface. Hard errors are caught here, information collected about them, and they are forwarded to an error page. At the same time, an e-mail with an exact description of the error is sent to the administrator. Input errors are indicated to the user so that they can be corrected.

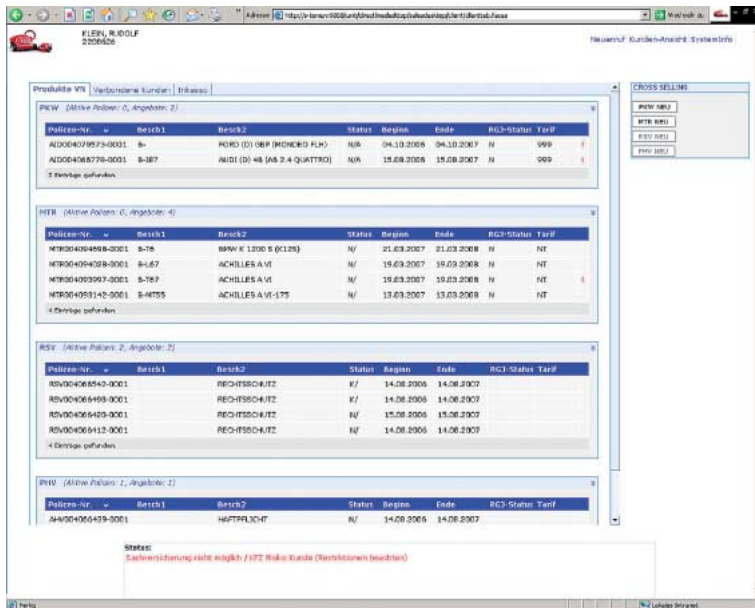
Spring-AOP was also used during the test phase of the application to monitor performance of calls to

About Direct Line

Direct Line was founded in Great Britain in 1985. The company is part of RBS Insurance, with about 18,000 employees and 26 million policies around the world, making it England's largest car insurer. Besides England, RBS Insurance is also represented in Spain, Italy, and Germany under the Direct Line brand. With their attention-getting logo, a red telephone, Direct Line Versicherung AG, headquartered in Teltow near Berlin, has been active on the German market since the start of March, 2002, and their roughly 300 employees insure about 280,000 vehicles, with a premium volume of 115 million Euros yearly. Besides automobile and motorcycle insurance, the company also offers private liability and private legal protection insurance. Direct Line belongs to the Royal Bank of Scotland Group (RBS), one of the world's largest banking group.

business methods, in order to identify critical points.

The view layer works with MyFaces, and the use of Tiles guarantees simple page generation using templates. MyFaces was extended with custom components, for instance to lead the user through the



Screenshot of a user mask

different input forms for a product by means of a tab. A special list box is also used for entry of values depending on the product and the time of the offer (e.g. for specific fee generation). During the course of development, moreover, some tools were developed to simplify testing. Besides the Spring-AOP

Experiences

In retrospect, the technologies described above proved to be a good choice, since it turned out that the use of Spring contributed a great deal to the modularity of the application. Similarly, thanks to the use of MyFaces, the user interface could

Acceptance by users has high priority, and contributes to the success of the project

performance measurements for the business layer already mentioned, there were also filters developed for the web container: First, a Jamon filter for measurement of performance of individual page calls, and secondly a filter which generates a list of allocations of labels to IDs of input fields in the application. Using these filters, the development of Selenium tests for functional testing was nearly 100% automatic.

be developed rapidly, at which the application of Tiles was especially found to be favorable. The two together simplified the integration of new products into the application tremendously. For the user interface, Alias components were used in numerous places, so that the application is easy to maintain, since changes need only be made in one place. However, the structure of these components must be consid-

ered carefully in order to preserve clarity. Parallel changes to the Java interface were necessary in order to be able to add new products, which simultaneously led to optimization of the design.

The technologies provided allowed a variety of functionality and load tests. Smaller problems in later production operation, for instance the fact that performance in one external call center was insufficient due to a weak connection, could be improved by additional optimization of the application.

The acceptance among the users is quite positive, because the application is clearer and stabler than its predecessor. A few particular goodies were integrated just for the staff: For instance, it is possible to use different stylesheets to change the look and feel of the application to match individual taste.

Due to this positive experience, plans are now underway to integrate additional applications of the enterprise into this architecture.

For further information:

Dr. Ingo Büll
Phone: +49(0)431/3993-554
E-Mail: buell@consist.de

