

## Profil von Mirko Wawrowsky



### Kontakt

Sommestraße 28  
71384 Weinstadt  
0173-3914249  
mirko@wawrowsky.de  
<http://www.wawrowsky.de>

Jahrgang  
Familienstand  
Staatsbürgerschaft  
Einsatzort

1966  
verheiratet, 2 Kinder  
deutsch  
Raum Stuttgart, remote deutschlandweit

## Ausbildung

---

Diplom-Informatiker (FH)  
Informatik-Studium an der FH Würzburg mit Schwerpunkt Wirtschaft  
Fremdsprachen: Englisch (fließend), Französisch (Grundkenntnisse)  
Sun Certified Java Programmer  
Kurs „Maschinelles Lernen“ des DFKI (Deutsches Forschungszentrum für KI)

## Tätigkeitsschwerpunkt

---

Full Stack Entwicklung mit Java und Angular

## Kenntnisse

---

Backend	Java, Java EE, Spring Boot, Spring Web, REST, SOAP, JPA/Hibernate, SQL
Cloud	Docker, Kubernetes, Azure, Cloud Foundry
Frontend	Angular, TypeScript, HTML 5, CSS, Thymeleaf, JavaScript, JSF, JSP
Sonstiges	Maven, junit, Mockito, XML, XSL, XSLT, JAX-WS, JAX-RS, Jersey, Apache CXF, JAXP, JAXB, iText, FOP, JasperReports, POI, Solr/Lucene, Swing, Eclipse RCP/SWT, ANTLR
CI/CD	Git, Jenkins, SonarQube, GitHub, GitLab, SVN
AI	Python, MathML
Tools	JIRA, Confluence, Upsource, IntelliJ IDEA, Eclipse, VS Code, IBM RAD, NetBeans, Anaconda
Applikationsserver:	JBoss AS, IBM WebSphere
DB	SQL, Oracle, DB2, Informix, MySQL/MariaDB, MSSQL
Methoden, Tools	Scrum, UML
Sprachen	Java, SQL, HTML, CSS, JS, TypeScript, XML, Python

## Referenzen

---

Allianz Lebensversicherung AG  
Alcatel SEL AG  
Auswärtiges Amt  
Crealogix  
Daimler AG  
Deutsche Rentenversicherung Bund  
DKB  
Dr. Güldener GmbH/Optica Abrechnungszentrum  
dv/d systempartner GmbH (heute cellent AG)  
EnBW  
ETH Eidgenössisch Technische Hochschule Zürich  
GfA Gesellschaft für Anwendungssoftware mbH  
Hallesche PKV  
Landesbank Baden-Württemberg LB-BW  
mhplus Betriebskrankenkasse  
Porsche AG  
Siemens AG  
Targens GmbH  
T-Systems  
Techem AG  
Thales Group  
Vector Informatik GmbH  
Württembergische Versicherung AG

## Projekte

---

**Kunde/Branche:** Öffentlicher Sektor

**Zeitraum:** Jan 2021 – Sep 2021

**Projekt:** eID-Karte

**Sprachen/APIs:** Java 11, Spring Boot, Angular, Docker, REST, JSON, HTML, CSS, SQL

**Tools/Systeme:** IntelliJ IDEA, Postgres

**Tätigkeit:** Konzeption, Realisierung nach SCRUM

**Beschreibung:** Unionsbürger bekommen mithilfe der eID-Karte die Möglichkeit, sich in Deutschland elektronisch zu identifizieren. Es wurde die Applikation zur Beantragung, der Ausgabe sowie der nachgelagerten Bearbeitung (Sperrung etc.) entwickelt. Die Anwendung weist ein ansprechendes UI auf, das in Angular realisiert wurde. Das Backend ist mit Java realisiert und bedient sich einer Postgres SQL-Datenbank. Die Kommunikation findet über eine REST-API statt.

Automatische Unit- und Integrationstests sowie ein aufwändiger Reviewprozess hatten im Entwicklungsprozess einen überdurchschnittlichen Stellenwert.

**Kunde/Branche:** Energieversorger

**Zeitraum:** Okt 2020 – Dez 2020

**Projekt:** Customer Service

**Sprachen/APIs:** Java 11, Spring Boot, Azure, Docker, Gradle, REST, JSON, HTML, CSS, Thymeleaf

**Tools/Systeme:** IntelliJ IDEA, Postman, Azure DevOps

**Tätigkeit:** Konzeption, Realisierung nach SCRUM

**Beschreibung:** Optimierung des Customer Service durch Erhöhung der First Contact Resolution. Integration mehrerer SaaS Systeme. Erstellung von Microservices, welche ein Messaging-System zur Abwicklung der Kommunikation zwischen den SaaS-Systemen bilden.

## Evaluationsprojekte

**Zeitraum:** 2020

**Projekte:** Evaluation, Architekturentwicklung, POCs

**Sprachen/APIs:** Angular, TypeScript, Java, Spring Boot, Maven

**Tools/Systeme:** Visual Studio Code, IntelliJ IDEA, Postman, Cloud Foundry, Docker, Azure

**Tätigkeit:** Evaluationsprojekte

**Beschreibung:**

Entwicklung einer Standardarchitektur für Web-Applikationen. Erstellung von POCs mit aktuellen Frontend-Technologien und Cloud Services. Testprojekte zur Erstellung von Web-Applikationen mit Angular. Abgedeckt wurden u.a. die Bestandteile Components, Routing, Reactive Forms, Pipes, DI, NgRx, REST sowie Layouting mit Semantic UI. Als Cloud-Technologien Einsatz von Java, Spring Boot, Cloud Foundry, Microsoft Azure.

**Kunde/Branche:** Finance

**Zeitraum:** Aug 2019 – Dez 2019

**Projekt:** Clustering mit Machine Learning (Künstliche Intelligenz)

**Sprachen/APIs:** Java 11, Spring Boot, Apache Commons Math ML, Python, XML, iText, JFreeChart, Maven

**Tools/Systeme:** IntelliJ IDEA, Git, Jira, SonarQube

**Tätigkeit:** Konzeption, Realisierung nach SCRUM

**Beschreibung:** Kundensegmentierung mit Verfahren des Machine Learnings. Durch Einsatz des Clustering-Algorithmus' K-Means wird die Bildung von Kundengruppen automatisiert. Das in Python erarbeitete Verfahren wurde mit Java als Spring Boot Kundenapplikation entwickelt. Als Machine-Learning Bibliotheken wurden u.a. H<sub>2</sub>O, Apache Spark und Apache Commons Math ML evaluiert. Letztere kam schließlich zum Einsatz. Die Ergebnisse des Clusterings und der Segmentierung werden in einem PDF-Dokument protokolliert und mit JFreeChart grafisch dargestellt. Das Vorgehensmodell war Scrum.

**Kunde/Branche:** Automotive  
**Zeitraum:** Apr 2019 – Juli 2019

**Projekt:** Mobility Services

**Sprachen/APIs:** Java, Spring Boot, JSON, XML, REST, OpenAPI/Swagger, Docker, Cloud Foundry

**Tools/Systeme:** IntelliJ IDEA, Visual Studio Code, Postman, SonarQube

**Tätigkeit:** Konzeption, Realisierung nach SCRUM

**Beschreibung:** Bereitstellung von innovativen Mobilitätslösungen als Digital Services. Erstellung und Erweiterung von Microservices mit Java, Spring Boot, Benutzeroberflächen in Angular und React. Integration von Fremdsystemen über REST APIs. Das Vorgehensmodell war Scrum.

### **Training, Self Studies**

**Zeitraum:** Jan 2019 – März 2019

**Projekte:** Künstliche Intelligenz, Machine Learning, Angular, React, Website-Erstellung

**Sprachen/APIs:** Python, TensorFlow, React, Javascript, Angular, TypeScript

**Tools/Systeme:** Anaconda, Visual Studio Code, WebStorm, WordPress

**Tätigkeit:** Kurse, Konzeption und Realisierung einer Website, Eigenprojekte

**Beschreibung:**

1. Zum eigenen professionellen Kompetenzaufbau wurden Kurse im Bereich des AI/Machine Learning, mit Fokus auf der Anwendung der Methoden, absolviert sowie einfache Anwendungen durchgeführt.
2. UI Erstellung mit Angular und React
3. Konzeption und Realisierung einer professionellen Website

**Kunde/Branche:** Bank

**Zeitraum:** Sept 2017 – Jan 2019

**Projekt:** Onlinebanking- Portal, Kreditkarten-Portal

**Sprachen/APIs:** Java, JavaEE, abaXX, JPA, Spring, Spring Boot, SQL, JUnit, JSP, HTML, CSS, XML, REST, JSON, Maven

**Tools/Systeme:** IDEA, SVN, Oracle, Jenkins, Confluence, Postman, SonarQube, Windows

**Tätigkeit:** Realisierung und Test nach SCRUM

**Beschreibung:** Diverse Entwicklungstätigkeiten im Umfeld eines Onlinebanking-Portals. Erweiterung des Webportals, Erstellung von REST-APIs. Erstellung eines neuen Kreditkarten-Portals. Das Vorgehensmodell war Scrum.

**Kunde/Branche:** Automotive

**Zeitraum:** Nov 2016 – Aug 2017

**Projekt:** WLTP

**Sprachen/APIs:** Java, JavaEE, EJB, JPA, PAI, Spring, Spring Boot, SQL, JUnit, Mockito, JSF, XML, REST, JSON, Maven

**Tools/Systeme:** Eclipse, SVN, DB2, Jenkins, Nexus, Sonar, WebSphere, Windows, Linux

**Tätigkeit:** Realisierung und Test nach SCRUM

**Beschreibung:** Erstellung einer JavaEE Applikation mit REST-Schnittstelle für die Bereitstellung von CO<sub>2</sub>- und Verbrauchsdaten von individuellen Fahrzeugen zur Einführung der neuen Zulassungsprozeduren nach WLTP.

**Kunde/Branche:** Gesundheitswesen

**Zeitraum:** Mai 2015 – Okt. 2016

**Projekt:** Kundenabrechnung

**Sprachen/APIs:** Java, Spring, Spring Boot, Swing, SQL, JUnit, FitNesse, Mockito, XML, REST, JSON, Maven

**Tools/Systeme:** Eclipse, IntelliJ IDEA, Git, CVS, Informix, Jenkins, Windows

**Tätigkeit:** Realisierung und Test nach SCRUM

**Beschreibung:** Weiterentwicklung einer Software zur Kundenabrechnung bei einem Abrechnungszentrum für medizinische Leistungen sowie Bereitstellung eines RESTful WebServices für die Archivierung. Der Schwerpunkt der Tätigkeit war neben der Entwicklertätigkeit die Sicherstellung der fachlichen Korrektheit der Berechnungen in Zusammenarbeit mit den Fachabteilungen.

**Kunde/Branche:** Automotive

**Zeitraum:** Juli 2014 – April 2015

**Projekt:** Vehicle Systems Engineering Environment

**Sprachen/APIs:** Java, Eclipse, JAX-RS, JAX-WS, Apache CXF, Spring, Hibernate, JUnit

**Tools/Systeme:** Eclipse, SVN, Jenkins, SoapUI, Oracle, Windows

**Tätigkeit:** Realisierung und Test nach SCRUM

**Beschreibung:** Neuentwicklung zweier WebServices (RESTful und SOAP) mit JAX-RS und JAX-WS zur Anbindung des Engineering Systems an Fremdsysteme. Beim Engineering Environment handelt es sich um ein Produkt zur Entwicklung der Elektrik im Kfz-Bereich, vor allem der Embedded Systems. Es unterstützt das Anforderungsmanagement, die Architektur, das Netzwerk-Design, die Entwicklung und den Test der Software. Technisch ist das System als n-Tier-Anwendung bestehend aus GUI-Client, Application Server und Datenbank für ca. 1000 Benutzer ausgelegt.

Neben der Entwicklung der WebServices beinhaltete die Tätigkeit die Erweiterung des Produktes um neue Funktionen und Unterstützung bei Test und Fehlerbeseitigung.

**Kunde/Branche:** Transportation

**Zeitraum:** Jan. 2013 – Juni 2014

**Projekt:** Train Routing Human-Machine-Interface

**Sprachen/APIs:** Java, Eclipse RCP, SVG, JUnit

**Tools/Systeme:** Eclipse, SVN, Windows, Linux

**Tätigkeit:** Realisierung und Test nach SCRUM

**Beschreibung:** Neuentwicklung eines Systems zur Zuglenkung und Zugbeeinflussung (nach ETCS Standard) sowie zur Visualisierung und Bedienung von Stellwerken.

Die Software visualisiert die Gleisbilder von Stellwerken und Bahnhöfen. Über das grafische Interface können Kommandos an das Stellwerk abgesetzt werden. Die Fahrdienstleiter können damit die Zuglenkung und Zugbeeinflussung vornehmen.

Auf Grund der hohen Kritikalität bzgl. der Sicherheit fällt dem Test eine herausragende Rolle zu und nimmt einen großen Teil des Entwicklungsaufwandes ein. Als Vorgehensmodell dient SCRUM.

**Kunde/Branche:** Private Krankenversicherung

**Zeitraum:** April 2012 – Dez. 2012

**Projekt:** Integriertes Leistungssystem

**Sprachen/APIs:** Java/JavaEE, Spring, JSF

**Tools/Systeme:** WID Websphere Integration Developer, WAS, Eclipse, SVN, Windows 7

**Tätigkeit:** Realisierung, Test

**Beschreibung:** Das hauseigene System zur Leistungsabrechnung wurde erweitert. Alle leistungsrelevanten Vorgänge werden teilweise automatisiert vorverarbeitet und den Sachbearbeitern zur Bearbeitung zur Verfügung gestellt. Die Vorgänge für alle verschiedenen Leistungserbringer wie Ärzte (GOÄ), Zahnärzte (GOZ), Krankenhäuser (DRG), Apotheken u.a. werden integriert über die gleiche Oberfläche bedient. Der gesamte Workflow vom Scanergebnis bis zur Zahlungsanweisung wird vom System abgedeckt.

**Kunde/Branche:** Versicherung

**Zeitraum:** Mai 2011 – April 2012

**Projekt:** Vertriebstool für den Kundenberater

**Sprachen/APIs:** Java/JavaEE, JPA/Hibernate, Spring, Maven, SQL, iText, GWT

**Tools/Systeme:** DB2, Tomcat, Eclipse, SVN, Windows 7

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Die Versicherungsberater werden in die Lage versetzt, alle Vorsorgeprodukte des Versicherungskonzerns mit einer Software zu beraten.

Das System ist hochmodular. Einzelne Komponenten können ohne Programmieraufwand ein- und ausgeklint werden. Die Anwendung kann sowohl offline auf dem Beraterarbeitsplatz als auch in einem Portal betrieben werden.

Die Anwendungsentwicklung wird durch einen Frameworkansatz stark vereinfacht. Der Entwickler hat für alle Elemente der Anwendung fertige Komponenten zur Verfügung, die Persistenzschicht ist komplett gekapselt.

Zu meinen Tätigkeiten gehörten die Entwicklung der Druck-API des Frameworks, das Coaching der Anwendungsentwickler, das Entgegennehmen von Anforderungen aus der Anwendungsentwicklung an die Frameworkentwicklung sowie die Performanceoptimierung.

**Kunde/Branche:** Krankenkasse

**Zeitraum:** Aug. 2010 – Apr. 2011 (mit Unterbrechungen)

**Projekt:** Monitoring, Geschäftstellenauskunft, Mitarbeiter-Auskunftssystem

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, XML, iText, JasperReports, Quartz

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Mehrere Systeme aus früheren Projekten wurden erweitert.

**Kunde/Branche:** Forschung/Hochschule

**Zeitraum:** März 2010 – März 2011 (mit Unterbrechungen)

**Projekt:** Dokumenteninformationssystem (Enterprise Search)

**Sprachen/APIs:** Java/JavaEE, JSF, Facelets, JBoss Seam, Richfaces, Lucene, Solr

**Tools/Systeme:** Tomcat, Eclipse, SVN, Windows 7

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Anwendung im Bereich Enterprise Search: In einer Volltext-Indexdatenbank werden beliebige textuelle Inhalte gespeichert und für eine Suche bereitgestellt. Zur Suche dient eine Web-Oberfläche. Die Software besteht aus drei Komponenten: Komponente 1 und Kern ist die Indexdatenbank auf Basis Apache Solr. Komponente 2 ist das Collector-Framework, das beliebige

Quellen wie Filesysteme, Übertragungsprotokolle wie HTTP, FTP etc. anzapfen kann, um Dokumente mit textuellen Inhalten zu gewinnen. Das können z.B. Office-Dokumente, XML/HTML-Dateien oder auch Multimedia-Dateien sein, deren Metadaten ausgelesen werden. Komponente 3 schließlich ist die Webapplikation, die leistungsfähige Suchmöglichkeiten zur Verfügung stellt. Die Dokumente können per Volltextsuche, nach Kategorien oder per Facettensuche aufgelistet werden.

**Kunde/Branche:** Rentenversicherung

**Zeitraum:** März 2010 - Juli 2010

**Projekt:** Enterprise Search/Dokumenteninformationssystem

**Sprachen/APIs:** Java/JavaEE, Struts, jQuery, Apache Lucene, PDFBox, POI, Lotus Notes API, SQL

**Tools/Systeme:** Oracle, Tomcat, IBM RAD 7, CVS, Windows XP

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Sämtliche elektronisch verfügbaren Dokumente werden in einer Index-Datenbank gespeichert und stehen für eine Volltextsuche zur Verfügung. Die Dokumente liegen sowohl in Notes- als auch in Oracle-Datenbanken. Zu den Dokumentkategorien zählen sowohl textuelle als auch tabellarische (statistische) Dokumente. Eine komfortable Webapplikation bietet umfangreiche Suchmethoden für verschiedene Dokumentkategorien und Anwendungsfälle an. Die Originaldokumente sind über die Webapplikation einsehbar.

**Kunde/Branche:** Forschung/Krankenhaus

**Zeitraum:** März Aug. 2009 - Sep. 2009

**Projekt:** E-Learning, Volltextdatenbank

**Sprachen/APIs:** Java/JavaEE, JBoss Seam, RichFaces, JSF, Apache Lucene, POI, iText

**Tools/Systeme:** Tomcat, Eclipse, CVS, Windows XP

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Eine Forschungsklinik in den USA bildet ihre Chirurgen regelmäßig fort. Zu diesem Zweck wurde ein Softwaresystem entwickelt, das die Beschreibungen von Operationstechniken verwaltet und eine Suche darin ermöglicht. Es bildet zugleich die Basis für die Durchführung von Prüfungen der auszubildenden Chirurgen. Die Texte werden in einer Datenbank gespeichert und sind im Volltext durchsuchbar. Die Texte zu den Suchergebnissen können in PDF-Dateien ausgegeben werden. In zwei verschiedenen Modi werden entweder die Lehrtexte oder die Prüfungstexte mit Fragen ausgegeben. Die Erfassung der Texte erfolgt mittels Excel-Dateien. Die Dateien werden automatisch mit POI in die Volltextdatenbank eingelesen und stehen unmittelbar nach dem Speichern für die Suche zur Verfügung. Die Benutzerschnittstelle ist als Webapplikation ausgelegt und ist mit mithilfe des JBoss Seam Frameworks realisiert, wobei vor allem RichFaces (JSF) genutzt wurde. Die Volltextdatenbank wurde mit Apache Lucene realisiert.

**Kunde/Branche:** Krankenkasse

**Zeitraum:** Sept. 2005 – 2011 (mit Unterbrechungen)

**Projekt:** Monitoring

**Sprachen/APIs:** Java/JavaEE

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Das CRM-System bezieht Informationen über Kundenkontakte von mehreren anderen Systemen. Der Import der entsprechenden Daten wird über XML-Dateien abgewickelt. Dieser Import birgt diverse Fehlerquellen. Daher wurde ein Dienst entwickelt, der den

Import überwacht und Fehlerinformationen bereitstellt.

**Projekt:** Eingangspostverarbeitung

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Die Eingangspost soll automatisch in den elektronischen Postkorb der Mitarbeiter eingestellt werden. Dazu wurde eine automatische Zuständigkeitsermittlung entwickelt, die über ein Administrationsfrontend verwaltet werden kann. Das System erfasst sämtliche Briefvorlagen der Krankenkasse und ordnet sie den Prozessen und Mitarbeitern zu. Damit werden das Formularwesen, der elektronische Postkorb und die Prozesssteuerung mit der Vorgangsbearbeitung integriert. Die Verarbeitung der Eingangspost und die Zuordnung zu den einzelnen Systemen geschieht automatisch. Für die manuelle Steuerung gibt es eine Bearbeitungsmaske in der WebApplikation. Mit ihr kann manuell in die Zuordnung der Eingangspost eingegriffen werden und zusätzliche Informationen erfasst werden.

**Projekt:** Dokumentenverarbeitung für PZU-Druck

**Sprachen/APIs:** Java/JavaEE, Hibernate, iText, Quartz

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Einzelne erzeugte PDF-Dateien mit Briefen werden für die Druckaufbereitung in ein PDF zusammengefasst. Dazu passend werden Postzustellungsurkunden als PDF erzeugt.

**Projekt:** Kursdatenbank

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, XML, iText, JasperReports, Quartz

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Die Versicherten haben Anspruch auf Kostenerstattung für Gesundheitskurse. Welche der Kurse zertifiziert sind, musste bislang in verschiedenen Systemen ermittelt werden. Das neue System integriert alle bisherigen Datenbestände in einer Datenbank und bietet darüber hinaus die Möglichkeit, weitere Kurse zu erfassen. Der Datenbestand des Bundesverbandes wird regelmäßig importiert/abgeglichen. Die Kursinformationen können anhand umfangreicher Kriterien einschließlich Umkreissuche durchsucht und das Suchergebnis ausgedruckt werden.

**Projekt:** Geschäftsstellenauskunft

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, XML, Axis

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Im Internet besteht für die Versicherten die Möglichkeit, sich über die Geschäftsstellen zu informieren. Um die Auskunft komfortabler zu gestalten, wurde eine Umkreissuche erstellt, die für eine Postleitzahl und einen Umkreis alle Geschäftsstellen liefert. Die Anbindung des Suchdienstes an das CMS erfolgt mittels Webservice. Die Geodaten wurden aus der freien Geodatenbank ‚geonames.org‘ übernommen.

**Projekt:** Vier-Augen-Prüfung

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, XML, SOAP, Axis, Betwixt, Quartz

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Das System zur Brieferstellung mit zentralem Druck wurde um eine Anwendung ergänzt, die eine Vier-Augen-Prüfung von Briefen ermöglicht. Die entsprechenden Briefe werden in die Vorgangsbearbeitung eingestellt und können dort mit ihren Bearbeitungszuständen verfolgt werden. Die Kommunikation mit dem Briefdrucksystem erfolgt über SOAP-WebServices. Für zeitgesteuerte Verarbeitungen wurde ein Scheduler entwickelt

**Projekt:** Workflowsystem

**Sprachen/APIs:** Java/JavaEE, Hibernate, SQL, jBPM, JSF

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung

**Beschreibung:** Die bisherige Vorgangsbearbeitung sollte durch ein leistungsfähigeres Workflowsystem abgelöst werden. Als proof-of-concept wurde eine WEB-Applikation mit jBPM erstellt, die die Machbarkeit eines eigenentwickelten Workflowsystems im Krankenkassenumfeld aufzeigen sollte.

**Projekt:** CRM-Portal

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, XML

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung

**Beschreibung:** In einem Portal sollten mehrere Anwendungen integriert werden. Dazu zählten im ersten Schritt Versichertenverwaltung, Mitarbeiter-Auskunftssystem und Versichertenhistorie. Das System ist offen angelegt und kann durch beliebige Teilsysteme erweitert werden.

**Projekt:** Listenverarbeitung mit Workflowsteuerung

**Sprachen/APIs:** Java/JavaEE, Struts, SQL, XML, Betwixt

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Mit Hilfe des neuen Systems können beliebige Excellisten bzw. CSV-Dateien als Vorgänge in ein Workflowsystem hochgeladen werden. Die Zuordnung zu Versicherten und Mitarbeitern geschieht dabei automatisch. Die Struktur der CSV-Daten wird dazu per XML beschrieben.

**Projekt:** Mitarbeiter-Auskunftssystem

**Sprachen/APIs:** Java/JavaEE, Hibernate, Struts, SQL, iText, JasperReports

**Tools/Systeme:** MSSQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Die Ermittlung eines zuständigen Sachbearbeiters für einen Versicherten und Vorgang wurde automatisiert. Verzeichnislisten wurden abgelöst durch ein Auskunftssystem, das abhängig vom Versicherten bzw. dessen Arbeitgeber und Vorgangsart den/die zuständigen Mitarbeiter auflistet. Eine reine Mitarbeitersuche (Telefonbuch) ist ebenfalls Teil des Systems. Zu einem Mitarbeiter können Hauspostmitteilungen ausgedruckt werden.

**Projekt:** Briefversand Mahnwesen

**Sprachen/APIs:** Java/JavaEE, Struts, SQL

**Tools/Systeme:** MS SQL Server, Tomcat, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Der Mahnbriefversand sollte optimiert werden. Die Prüfung der Versendung anhand von Excellisten wurde durch eine Workflowsteuerung ersetzt, und die Serienbrieferzeugung wurde automatisiert.

**Projekt:** DB-Transformationen

**Sprachen/APIs:** Java, SQL

**Tools/Systeme:** MSSQL Server, Eclipse, CVS, Windows XP, Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Der Import einer Datenbankschicht wurde bislang mit Hilfe von DTS-Paketen abgewickelt. Die Importverarbeitung wurde einem Redesign-Prozess unterzogen und in Java neu implementiert.

**Kunde/Branche:** Bank

**Zeitraum:** Apr. 2005 - Jul. 2005

**Projekt:** Schnittstellenversorgung für Meldewesen Basel II

**Sprachen/APIs:** Java (J2EE), Swing, SQL, XML

**Tools/Systeme:** DB2 UDB, ANTLR, Eclipse, CVS, Windows XP

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Eine Standardsoftware für das Meldewesen für Basel II muss aus dem Operativen System mit Daten versorgt werden. Die Datenquellen werden mittels einer eigens entwickelten Beschreibungssprache generiert. Der Compiler für die Sprache ist mit ANTLR entwickelt. Aus den Beschreibungen werden mithilfe des Compilers die Zugriffsklassen auf die relationale Datenbank generiert. Das Zielsystem wird mit Daten im XML-Format versorgt. Die Klassen für die Erzeugung der Zieldaten werden ebenfalls generiert. Dazu wird Apache JaxMe eingesetzt, das die Schema-Datei ausliest und die Zielklassen generiert. Für die Verarbeitung der Daten wurde ebenfalls eine Regelbeschreibungssprache entwickelt und mit ANTLR prototypisch ein Compiler entwickelt. Für das Produktivsystem wird die Verarbeitung allerdings klassisch implementiert, da die Wartung des Systems durch einen Java-Entwickler ohne zusätzliche Kenntnisse möglich sein soll. Zum Starten der Applikation wurde mit Swing eine grafische Benutzeroberfläche erstellt, die den Aufruf über die Kommandozeile überflüssig macht.

**Kunde/Branche:** Automobilkonzern

**Zeitraum:** März 2002 - März 2005

**Projekt:** Autorensystem für Werkstattinformationssystem

**Sprachen/APIs:** Java (J2EE, Eclipse Platform, SWT)

**Tools/Systeme:** IBM WebSphere 5.0, JBoss, Eclipse, SWT, Hibernate, DB2, CVS, Windows 2000, Solaris

**Tätigkeit:** Realisierung, Test

**Beschreibung:** Ziel des Projekts ist die Verfügbarkeit von durchgängig einheitlichen Informationen mit einheitlicher Systematik für die Werkstätten, wie Ersatzteile, Arbeitsanweisungen und Reparaturzeitenbestimmung. Dazu werden diverse Fremdsysteme über Schnittstellen eingebunden. Die Informationen sollen weltweit mit aktuellem Stand verfügbar sein. Als Lösung wurde eine J2EE-Applikation mit Rich-Client entwickelt. Als Basis für den Client dient die Eclipse-Plattform mit SWT. Die Verarbeitung ist in EJB-Komponenten gekapselt.

**Projekt:** Steuerung von Host-Prozessen per Web-Applikation

**Sprachen/APIs:** Java (J2EE), Struts, SQL, XML

**Tools/Systeme:** IBM WebSphere 5.0, WSAD (Eclipse), DB2 UDB, CVS, Windows XP

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Zur komfortablen Steuerung von Host-Prozessen wurde eine Web-Applikation erstellt. Diese war um eine Funktion zu erweitern, um Daten im Browser in einer Tabelle mit dem Komfort wie von MS-Excel gewohnt bearbeiten zu können. Dazu wurde die EBA:Grid-Komponente eingebunden und Datenzugriffsmethoden für die Datenbankzugriffe erstellt.

**Projekt:** Auskunftssystem Fahrzeugdaten

**Sprachen/APIs:** Java (J2EE, JSP, Servlets), SOAP

**Tools/Systeme:** Eclipse, Apache Tomcat 4.1, Apache Axis Framework, MQ-Series, CVS, Windows 2000,

Linux

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Für die Unterstützung der Ermittlungsarbeiten der Kriminalämter wurde ein Auskunftssystem entwickelt, das zu einem Fahrzeug Daten zur Ausstattung liefert. Die Implementierung erfolgte als Webservice nach dem SOAP-Standard.

**Projekt:** Serviceabwicklung

**Sprachen/APIs:** Java (J2EE, JSP, Servlets, EJB), SQL

**Tools/Systeme:** IBM WebSphere, Apache Tomcat, Apache Web Server, Apache Struts, IBM DB2 7.2 UDB, Oracle 8i, MySQL, IBM VisualAge for Java 3.5, NetBeans, CVS, Windows 2000, Solaris

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Neuentwicklung einer Web-Applikation für die werkstattübergreifende Servicedokumentation und -abwicklung von PKW mit weltweitem Zugriff. Das System ist integriert mit einem CRM-System (ebenfalls eine Web-Applikation), das die Kundenbetreuer im Pre- und After-Sales Marketing unterstützt. Das System weist eine J2EE-basierte Mehrschichtarchitektur auf (Java Servlets, JSP, Struts, EJB). Als Datenbankserver kommen DB2 UDB (Fahrzeugdaten) und Oracle 8i (Benutzerdaten) zum Einsatz. Im Produktiveinsatz läuft die Applikation auf dem Application Server IBM WebSphere mit DB2/Oracle, in der Entwicklungsphase wurden MySQL und Apache Tomcat eingesetzt.

**Projekt:** Verwaltungs- und Auskunftssystem Ausstattungsschlüssel

**Sprachen/APIs:** Java (J2EE, JSP, Servlets)

**Tools/Systeme:** IBM WebSphere 3.5, Eclipse, Apache Tomcat 3.3/4.1, Apache Struts Framework, MQ-

Series, CVS, Windows 2000

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Eine Hostapplikation sollte durch eine browsergestützte Intranetapplikation ergänzt werden. Die Applikation dient zur Auskunft von Ausstattungsmerkmalen für beauftragte Fahrzeuge im Nutzfahrzeugbereich in kundenverständlicher Form. Dazu wurde die bewährte Architektur aus dem Vorprojekt verwendet, erweitert um komfortable Elemente für die Benutzerschnittstelle.

**Projekt:** Customer Order Fulfillment

**Sprachen/APIs:** Java (J2EE, JSP, Servlets)

**Tools/Systeme:** IBM WebSphere 3.5, Eclipse, Apache Tomcat 3.3/4.1, Apache Struts Framework, MQ-

Series, POI, CVS, Windows 2000

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Die Hostapplikation für die Kundenbestellabwicklung wurde durch eine browsergestützte Intranetapplikation ergänzt. Die Applikation dient zur Unterstützung des Bestellwesens im Nutzfahrzeugsbereich. Die bewährte Architektur aus den Vorprojekten wurde wieder übernommen und weiter ausgebaut, um erhöhten Anforderungen an die Benutzerschnittstelle gerecht zu werden.

**Projekt:** Auftragsinformation, Fahrzeugreservierung

**Sprachen/APIs:** Java (J2EE, JSP, Servlets)

**Tools/Systeme:** IBM WebSphere 3.5, Eclipse, Apache Tomcat 3.3/4.1, Apache Struts Framework, MQ-

Series, CVS, Windows 2000

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Zwei fachlich und technisch verwandte Projekte zur Auftragsinformation und Fahrzeugreservierung wurden erweitert um Schnittstellen zu Fremdsystemen. Die Anwendungen können nun von einem Fremdsystem aus geöffnet werden. Die Daten werden per XML ausgetauscht.

**Kunde/Branche:** Haustechnik

**Zeitraum:** Nov. 2001 – Feb. 2002

**Projekt:** Test- und Simulationssoftware

**Sprachen/APIs:** Java, Swing, JavaComm

**Tools/Systeme:** NetBeans 3.2.1, Windows 2000/XP, techem assisto

**Tätigkeit:** Realisierung, Test

**Beschreibung:** Das Home Automation System des Elektronunternehmens verfügt über eine inspielbare Systemsoftware. Um die Qualitätssicherung während der Entwicklung der Systemsoftware gewährleisten zu können, wurde eine Software entwickelt, die alle Vorgänge in der Anlage protokollieren und wieder abspielen kann. Dazu werden http-Requests und Meldungen an der seriellen Schnittstelle abgefangen, gespeichert und zu einem beliebigen Zeitpunkt reproduziert. Die Software stellt eine Swing-Benutzerschnittstelle zur Verfügung und verwendet Sockets und die JavaComm API.

**Kunde/Branche:** Lebensversicherungskonzern

**Zeitraum:** Nov. 2000 - Sep. 2001

**Projekt:** B2B Online Rentenversicherung

**Sprachen/APIs:** Java, XML, XSL

**Tools/Systeme:** IBM WebSphere, Windows NT, VisualAge for Java, Apache FOP, Xalan, Xerces

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Realisierung von XSL-Stylesheets für die automatisierte Erzeugung von Antragsformularen zum Abschluss einer Rentenversicherung beim Versicherungskonzern. Erstellung der Applikationskomponenten in Java zum Erzeugen der XML-Daten mit den Antragstellerdaten.

**Projekt:** B2B Online Kontoeröffnung

**Sprachen/APIs:** Java, J2EE, Servlets, JSP, XML, XSL, UML, SQL

**Tools/Systeme:** IBM WebSphere, Windows NT, VisualAge for Java, Apache FOP, Xalan, Xerces

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Der Projektauftrag umfasste drei Teilaufgaben:

1. Realisierung des Rechenkerns für die finanzmathematische Berechnung einer Altersvorsorge nach dem neuen Altersvorsorgegesetz
2. Erweiterung der Online-Kontoeröffnung um weitere Formulare
3. Realisierung des XSL-Stylesheets für die automatisierte Erzeugung eines Formbriefs für die Benutzerverwaltung des Maklerportals

**Projekt:** B2B Online Kontoeröffnung

**Sprachen/APIs:** Java, J2EE, Servlets, JSP, XML, XSL, UML, SQL

**Tools/Systeme:** IBM WebSphere, Windows NT, Oracle, VisualAge for Java, Apache FOP, Xalan, Xerces,

Rational Rose

**Tätigkeit:** Realisierung, Test

**Beschreibung:** Ziel des Projektes war die Umstellung der Konto- und Depotöffnung von manueller Formularerfassung auf eine Web-gestützte Intranetapplikation. Die Antragsdaten werden nun online durch die Vertreter erfasst und in einer relationalen Datenbank gespeichert, bis sie an das Backend (IBM Mainframe) übergeben werden. Die Formulare können online in Form eines generierten PDF-Dokuments ausgedruckt und dem Kunden zur Unterschrift vorgelegt werden. Eine Teilaufgabe war die Konzeptionierung der Anwendung auf Grundlage eines J2EE-basierenden Frameworks.

Eine weitere Teilaufgabe war der eigenständige Entwurf und die Realisierung der Datenbankzugriffsschicht (OO to Relational Mapping). Die dritte Teilaufgabe war, den Druck der Antragsformulare mithilfe XML/XSL und dem Tool FOP der Apache Group zu entwerfen und zu realisieren.

**Kunde/Branche:** Elektrokonzern

**Zeitraum:** Juli 2000 - Okt. 2000

**Projekt:** Zentrales Forderungsmanagement-System

**Sprachen/APIs:** Java, J2EE, Servlets, JSP, JavaBeans, HTML, C++, UML

**Tools/Systeme:** Brokat Twister 4.0, Windows NT, Solaris 2.7, Oracle, SAP, RSA ACE/Server SecurID, Together

**Tätigkeit:** Technische Projektleitung, Konzeption, Realisierung

**Beschreibung:** Mit dem Projekt vereinfacht und zentralisiert der Elektrokonzern sein Forderungsmanagement.

Über eine Intranet-Anwendung können die Konzern-Mitarbeiter Kundeninformationen einsehen, orderungen auflisten und Zahlungsankündigungen (Avisé) erfassen. Über eine Internet-Applikation önnen die Kunden direkt ihre offenen Rechnungen einsehen und Avisé eingeben. Die Realisierung rfolgte mittels JavaServer Pages (JSP) und Servlets. Als Entwicklungssystem diente das SUN JDK .2.2 mit dem Texteditor UltraEdit und für das Design in UML wurde Together/J benutzt. Meine Aufgaben waren die technische Projektleitung des Web-Frontends in der ersten Realisierungsstufe, das Design und die Implementierung eines Frameworks für die Servlets und JSP-Seiten nach Model 2 mit Anbindung des Applikationsservers Twister und die Anbindung des SAP-Systems für das zentrale Forderungsmanagement. Des weiteren wurde eine Twister-Erweiterung (Accessor) in C++ für die Anbindung des Produktes ACE/Server (SecurID) von RSA realisiert.

**Kunde/Branche:** Telco

**Zeitraum:** Jan. 1999 - Jun. 2000

**Projekt:** Entwicklung im Bereich CustomerCare and Billing

**Programmiersprache:** ColdFusion, Informix, HTML, JavaScript, C++

**Tools/Systeme:** Windows NT

**Tätigkeit:** Konzeption, Realisierung, Test

**Beschreibung:** Der Kunde vertreibt eine Standardsoftware für die Abrechnung bei Stadtnetzbetreibern. Dabei handelt es sich um eine herkömmliche Windowsapplikation. Diese Applikation galt es nun zu erweitern und intranetfähig zu machen. Dazu wurde eine Intranetapplikation entwickelt. Sie Intranetapplikation ermöglicht dem Stadtnetzbetreiber die Abwicklung seines erngeschäfts. Dazu zählen die Abrechnung der Kundengespräche (Billing), die Anwerbung von Neukunden (Marketingkampagnen), die Ablauforganisation (Workflow) sowie die Bearbeitung von Fehlermeldungen.

Zu meinen Aufgaben gehörte die Konzeption und Programmierung von System- und Anwendungskomponenten, die Konzeption der Datenbank sowie die Erweiterung der Entwicklungssprache ColdFusion um CustomTags in C++.

Ein größeres Teilprojekt war die Entwicklung eines Web-Frontends, das es den Kunden der Stadtnetzbetreiber ermöglicht, über das Internet ihre Rechnung einzusehen und umfangreiche Statistiken über die geführten Gespräche zu erstellen, die grafisch im Webbrowser dargestellt werden.

**Kunde/Branche:** Bank

**Zeitraum:** Apr. 1998 - Dez. 1998

**Projekt:** Entwicklung im Bereich Client/Server

**Programmiersprache:** Enfin Smalltalk

**Tools/Systeme:** OS/2, DB2

**Tätigkeit:** Realisierung, Test

**Beschreibung:** Der Beraterarbeitsplatz der Bank war um einige neue Funktionen zu erweitern. Die neuen Funktionen waren in das bestehende Framework einzufügen. Beim Beraterarbeitsplatz handelt es sich um die SW, die den Kundenberatern in den Filialen die Bearbeitung aller Vorgänge bei der Kundenbetreuung ermöglicht.

**Projekt:** Berechtigungs- und Journalisierungsservices

**Programmiersprache:** Visual Age C++, Java

**Tools/Systeme:** OS/2, Hostanbindung

**Tätigkeit:** Konzeption, Realisierung

**Beschreibung:** Entwicklung einer Berechtigungssteuerung und Journalisierung als zentraler Dienst in einer heterogenen Systemlandschaft (Mainframe, PCs) mithilfe einer Middleware auf der Basis von Java und C++. Für diese Middleware wurde ein Framework entwickelt. Eine größere Teilaufgabe war die Entwicklung der Kommunikationsschicht dieses Frameworks. Dazu waren Komponenten für die Kommunikation in TCP/IP-Netzwerken zu erstellen.

## **Tätigkeiten in Festanstellung**

**Zeitraum:** Dez. 1994 - März 1998

**Branche:** Fertigungsindustrie, Handel

**Projekt:** Eigenständige Entwicklung einer Data Warehouse Software zur dynamischen Erstellung multidimensionaler Berichte (OLAP-Tool).

**Programmiersprache:** Borland Delphi, SQL

**Tools/Systeme:** Windows NT, MS-SQL Server, Paradox lokale DB

**Branche:** Fertigungsindustrie, Handel

**Projekt:** Vertriebsabwicklungs- und Produktkonfigurationssoftware

**Programmiersprache:** Objective-C++

**Tools/Systeme:** NEXTSTEP, Informix

**Branche:** Fertigungsindustrie, Handel

**Projekt:** Außendienstlösung für die mittelständische Fertigungsindustrie

**Programmiersprache:** Objective-C++

**Tools/Systeme:** NEXTSTEP, Informix