

Freiberufler - Profil

Ralf Schukey

Auf der Schleide 73
D-53225 Bonn

Telefon: +49 (0)228-46 50 32
Mobil: +49 (0)151-21 23 29 39
E-Mail: info@schukeysoft.de
Web: www.schukeysoft.de

Persönliches

- Nationalität: deutsch
- Sprachen:
 - Deutsch (Muttersprache)
 - Englisch (fließend)
 - Französisch (Grundkenntnisse)

Das Besondere an mir...

- ...ist ein hohes Maß an Zielstrebigkeit, Kreativität und methodischem Vorgehen
- ... verkörpert eine sehr gewissenhafte und gründliche Arbeitsweise
- ... ist ein hohes Maß an Verlässlichkeit und Loyalität



Auf einen Blick

- Informatikstudium an der Universität Bonn mit Vertiefungsgebiet Künstliche Intelligenz und Expertensysteme, Nebenfach BWL; Abschluss Diplom mit der Gesamtnote „gut“
- Fachlicher Schwerpunkt: Softwareentwicklung in Java, Anwendungs-, Kommunikations- und Backend- Entwicklung, Systemintegration (u.a. im EAI-, ERP- und B2B Umfeld)
- über 25 Jahre Berufserfahrung in unterschiedlichen IT-Rollen:
 - Sieben Jahre als freiberuflicher Java-Softwareentwickler, besonders im ITK - Umfeld
 - Sechs Jahre als Java- Softwareingenieur (EDI, EAI, ERP) und fünf Jahre im Qualitätsmanagement, in Festanstellung, bei Sterling Commerce GmbH, Düsseldorf (seit 2010 gehört Sterling Commerce zu IBM)
 - Acht Jahre in der Qualitätssicherung, Softwareentwicklung (EDI- Entwicklung in C) und Consulting, bei o.tel.o communications GmbH & Co., Köln, vorher bei LION Gesellschaft für Systementwicklung mbH, Köln
 - Während des Studiums: Vier Jahre Pascal Programmierer und Anwendungsberater am Hochschulrechenzentrum der Universität Bonn; ein Jahr TeX / LaTeX Buchlayouter bei der Fraunhofer Gesellschaft (ehemals GMD), Sankt Augustin
- Weiterbildung in den Bereichen IT und Management wie z.B. „Mastering J2EE and EJB“, „Software-Qualitätssicherung“, „Middleware im Vergleich“ oder „Project Management Essentials“
- Selbststudium von u. a. Java 8, Java EE 6, SOA und Microservices

Detailkenntnisse

- Branchen: Systemhaus, Telekommunikation (EDI), ITK (IT + TK), Logistik, Industrie, Handel, Banken, Versicherungen
- Programmiersprachen: Java (Schwerpunkt), C, C++, JavaScript, Pascal, Lisp
- Datenbanken / -sprachen: SQL, PL/SQL, MySql, Oracle, MS-SQL
- Software Design: OOP, OOA/OOD, Design Patterns, Modellierung, UML, Domain-Driven Design, Parallelprogrammierung
- Vorgehensmodelle: Wasserfall-Modell, Rapid Application Development, Agile Software-Entwicklung (XP, Feature-Driven Development, Scrum)
- Java - Technologien: Java SE, J2EE/JEE, EJB, Java Swing / JGoodies, JSP, Spring Batch, Spring Boot, JPA, DOM, JDOM, JDBC, SAX/JAXP, JAXB, JMS, JCA/JCE
- Middleware: EAI, Netty, RMI, SAP Jco (Java Connector), RFC (SAP)
- Protokolle & Standards: XML, HTML, DTD, XML Schema, SAP-IDOC, BPML, PKI, X.509, AirPlay (RAOP, RTSP/RTP, SDP), VDV-KA
- Kommunikation, EDI: UN/EDIFACT, OdetteFTP, IPv4, IPv6, X.400, X.25
- Produkte: Eclipse, Visual Studio Code, Xcode, SonarQube, Tomcat, Oracle Weblogic Server, Jboss, GlassFish, Jenkins, CruiseControl, HP Quality Center, UC4, ServiceNow, MS-/ Open Office, Sterling Integrator
- allgemeine Werkzeuge: CVS, SVN, Git, Ant, Maven, Unix Shells (bash, csh, ksh, sh), Unix Tools (awk, sed, u. a.), TeX / LaTeX
- Testtools /-methoden: JUnit, Mockito, Komponententests, Integrationstests, Regressionstests, Lasttests (Smoke Testing)
- Betriebssysteme: MS Windows (NT – Windows 10), MS-DOS, MacOS, Unix (Darwin, AIX, HP-UX, Sun/Oracle Solaris, Linux)

Soziale und persönliche Kompetenz

- Zielorientierte, systematische und kundenorientierte Arbeitsweise
- Konzeptionelle Fähigkeiten, selbstständige Arbeitsweise
- Analytisches und strukturiertes Denken
- Sorgfalt, Genauigkeit, Selbstdisziplin
- Teamfähigkeit, hohe kommunikative Kompetenz, Empathie

Was kann ich für Sie tun?

Ich hoffe, ich konnte Ihnen einen ersten guten Eindruck meines bisherigen beruflichen Werdeganges vermitteln, der noch mittels der von mir bereits durchgeführten Projekte auf den nachfolgenden Seiten ergänzt werden soll.

Ich freue mich, wieder von Ihnen zu hören. Besuchen Sie mich doch auch auf Xing:



Ralf Schukey

Projekte und Tätigkeiten in der Softwareentwicklung (die letzten 15 Jahre)

Zeitraum	2018, 1 Monat
Branche	Eigene Produktentwicklung, Social Media u.a.
Produktentwicklung <i>Genral-Parser</i>	<p>Weiterentwicklung des seit 2015 ruhenden eigenen Softwareproduktes <i>AddressEventRecognizer</i> (s. weiter unten), nun aber noch konsequenter domänenunabhängig gestaltet, daher auch mit neuem Namen versehen.</p> <p>Textanalyse in möglichst beliebigem Kontext bzw. Domäne durch natürlich sprachliche Texterkennung und semantische Zuordnung oder bei statischen Texten über Templates.</p> <p>Das Produkt befindet sich derzeit noch in der Entwicklung. Eine erste Version wird eine Erkennung von Terminen und Kontakten enthalten.</p>
Rolle	SW-Architekt, Designer, Softwareentwickler
Eingesetzte Techniken	<p>MacOS / Windows, Visual Studio Code, Java SE/EE, Datenmodellierung, Design Patterns, OOA/OOD, Expertensysteme (KI), UML, XML, Maven, Git, Commons – MultiMaps, JDOM, JAXP, JPA, JAXB</p> <p>Testing: JUnit</p>
neue Aktivitäten	<ul style="list-style-type: none"> • Refactoring des alten Codes, diverse Fehlerbehebungen • Festlegung der neuen Architektur und der Datenmodelle unter der Berücksichtigung, möglichst viele beliebige Domänen hiermit beschreiben und dann auch erkennen zu können. • Exemplarisch werden in der XML - Konfigurationsdatei, die in strukturierter Form die jeweils involvierten Domänen beschreibt, die beiden Domänen Event/Termin und Adresse den Vorgaben entsprechend angepasst, u.a. Auslagerung der bisher im Code hinterlegten Artefakte wie z. B. verwendete Artikel und Präpositionen. • Abgleich des JAXP Parsers, um obige XML – Konfigurationsdatei zu erkennen und die entsprechenden Regeln daraus abzuleiten. • Verbannung aller domänenabhängigen Klassen wie z.B. <i>DateTime</i> oder <i>Location</i>. Stattdessen müssen alle nach dem Parsen gefundenen und erzeugten Artefakte über ein „Mapping“ in eine domänenspezifische Java API abgebildet werden. Hierzu werden je Domäne über ein Interface vorgegebene Methoden in einer domänenspezifischen Klasse implementiert. Diese Klasse kann bei Bedarf um weitere notwendige Geschäftslogik ergänzt werden. • Für die Domäne <i>Event/Termin</i> wird konkret die freie iCal4j Java - API verwendet, die aus den erzeugten Objekten ein nach RFC 5545 konformen Kalendereintrag in dem Format iCALENDAR erzeugt. • Implementierung der domänenspezifischen Klasse für die Domäne <i>Event/Termin</i>
Erfolg	Modellierung in XML, Verwendung verschiedener Parser

Zeitraum	2018, 2 Monate
Branche	Intern
<i>Admin Projekt</i>	<p>Internes, administratives Projekt: Wechsel der bisher für die eigenen Projekte eingesetzten Sourcecodeverwaltung SVN nach Git.</p> <p>Evaluierung der neuen und sehr schnellen IDE Visual Studio Code (VS Code) von Microsoft.</p> <p>Weiterhin werden alle Projekte auf Java 8 umgestellt.</p>
Rolle	Administrator, Softwareentwickler
Eingesetzte Techniken	<p>MacOS / Windows, Java SE/EE, Visual Source Code, Ant, Maven, Git</p> <p>Testing: JUnit</p>
Aktivitäten	<ul style="list-style-type: none"> • Evaluierung von MS Visual Studio Code mit dem Java Extension Pack für die Programmiersprache Java; hierbei wurden mir wichtige Aspekte für das alltägliche Arbeiten als Entwickler im Vergleich zu Eclipse berücksichtigt, die bis auf zwei Punkte erfüllt werden konnten: <ul style="list-style-type: none"> ○ Einstellmöglichkeiten: weniger als in Eclipse, aber ausreichend; einziger Wermutstropfen: die von Eclipse her bekannten umfangreichen Compiler – Errors/Warnings – Einstellungen gibt es in VS Code nicht ○ Erscheinungsbild, Java SE/EE, Persistenzschichten wie z. B. JPA ○ Übernahmefähigkeit von Eclipse spezifischen Projekten, Code Completion, Hovering, Refactoring, Methodenextraktion ○ Einbindung und Handhabung von Build Werkzeugen wie Maven oder Ant (Ant: erfolgt in VS Code mittels sog. Tasks) ○ Sourcecodeverwaltung wie z.B. Git (favorisiert) oder SVN ○ Compilierung und Debugging; Einschränkung: ein „normales“ Run ohne Debugging eines Java Programmes innerhalb der IDE ist bisher nicht möglich, das Ausführen von JUnit-Tests hingegen ist jedoch möglich. ○ Einbindung von VS Code Extensions; hier ist die Auswahl natürlich noch nicht so groß wie bei den Eclipse Plugins • Der Wechsel der bisher eingesetzten Sourcecodeverwaltung SVN nach Git konnte nach einer relativ kurzen Einarbeitung in Git sehr einfach vollzogen werden. Sechs eigene und zwei Open Source Projekte waren davon betroffen. Leider konnte systembedingt bei dem Wechsel die Historie des alten SVN nicht in Git übernommen werden, was aber zu verschmerzen ist • Anhebung der eingesetzten Java Version in allen Projekten, wenn noch nicht geschehen, auf die Version 8 • Einige Mac spezifische Projekte erforderten noch den Austausch von nur unter JDK 6 laufenden Maven Plugins durch andere bzw. neuere Plugins, wie z. B. das appbundle-maven-plugin
Erfolg	<p>Die Umstellung auf Git und Java 8 erfolgte relativ schnell und problemlos, das Arbeiten mit VS Code macht nach einiger Einarbeitung und Umgewöhnung richtig Spaß und geht flott von der Hand. VS Code ist trotz einiger Einschränkungen die vor Eclipse favorisierte IDE.</p>

Zeitraum	2017, 4 Monate
Branche	Versicherungen
Projekt <i>Migration-KVK</i>	<p>Für den In- und Exkassobereich eines Versicherungskonzernes sollten im Rahmen eines konzernweiten Wechsels der IT-Systemlandschaften von BS2000 hin zu SAP bereits vorhandene Daten von den Altsystemen in das Neusystem migriert werden. An dieser Migration war je betroffener Versicherungstochter ein spezifisches Java Framework maßgeblich beteiligt, das als Software-Gateway, firmenintern <i>Proxy</i> genannt, zwischen dem Altsystem und dem SAP System fungierte.</p> <p>In diesem Kontext sollte ich einen dieser Java - Proxies, zuständig für den BS2000 basierten In- und Exkassobereich <i>KVK</i>, weiterentwickeln und parallel für das bereits produktive <i>Proxy Release</i> die Administration und das alltägliche Monitoring übernehmen. Zudem sollte ich einen neuen internen Mitarbeiter in die Migrationsthematik der insgesamt vier sich im Einsatz befindenden Proxies einarbeiten.</p> <p>Die Projektbeauftragung für dieses Projekt erfolgte durch ein u. a. auf Versicherungen spezialisiertes Beratungsunternehmen.</p>
Rolle	Berater, Softwareentwickler, Qualitätssicherung, Operating
Eingesetzte Techniken	Windows 7 Enterprise unter Citrix XenDesktop / Red Hat Linux, Java SE/EE, Eclipse, OOA/OOD, UML 2, Maven, SVN, SQL, Oracle 11g, XML, Spring Batch, UC4 / Atomic, TSMP / ServiceNow, Harpoon 4.0 Testing: JUnit 4.5, Mockito, HP Quality Center, Jenkins
Aktivitäten	<ul style="list-style-type: none"> • Softwaretechnische Einarbeitung in die bestehende Migrationsthematik, insbesondere den weiter zu entwickelnden Java - Proxy. Architektonisch gesehen waren dabei zwei Java Frameworks von größerer Bedeutung: <ul style="list-style-type: none"> ○ da die Proxies im Kontext einer systemweiten Jobsteuerung via UC4 agierten, diente <i>Spring Batch</i> als Bindeglied zur Außenwelt ○ das firmeneigene <i>Harpoon</i> Java-Framework, das u. a. Lese- und Schreibroutinen für EBCDIC Dateien zur Verfügung stellt • Operating: Einarbeitung in das Monitoring mit Atomic (Nachfolger von UC4) und das IT Servicemanagement mit ServiceNow, zuvor TSMP • Durchführung und Erweiterung umfangreicher manueller und automatisierter Tests auf unterschiedlichen Umgebungen zur Vorbereitung eines bevorstehenden größeren Releasewechsels in der Produktion. Manuelle Tests wurden meistens mit dem Fachbereich abgestimmt • „Wissenstransfer“ dieses gesamten Themenkomplexes an einen von mir einzuarbeitenden neuen, internen Mitarbeiter am Standort Köln • Implementierung neuer Release - Anforderungen in dem Proxy, so das verglichen mit dem Vorgängerrelease mehr Geschäftsfallarten in das neue SAP System transferiert werden konnten • Darauf aufbauend mussten auch die bestehenden Prüfsummenberechnungen jeweils für Eingangs- und Ausgangsseite erweitert und korrigiert werden
Erfolg	Alle durchgeführten Tests für die Proxies nach erfolgreich verlaufendem Releasewechsel in der Produktion wurden durch die IT-Abnahme verifiziert; auch die Migration insgesamt war erfolgreich verlaufen.

Zeitraum	2015 - 2016, 12 Monate
Branche	ITK (IT + TK)
Projekt <i>((eTicket</i>	<p>Die DB Systel, als führender Hersteller von ((eTicket Deutschland, auch VDV-Kernapplikation¹ genannt, hat im Rahmen dieses Projekts für ca. 20 der über 100 deutschlandweit bestehenden Verkehrsverbünde das Verfahren eTicket Datendrehscheibe (DDS) entwickelt. Die eTicket-Funktionalität kommt in Form des Moduls <i>eTicket-Handler</i> in den zu integrierenden Funktionseinheiten Ticketautomat, dem Kontrollgerät <i>MTx</i> und der DDS zur Anwendung.</p> <p>Die Aufgabe bestand sowohl in der Wartung und der Weiterentwicklung des eTicket-Handlers als auch in der Implementierung von Anforderungen seitens bereits beteiligter oder neu hinzugekommener Verkehrsverbünde. Alle neuen oder geänderten Funktionalitäten sollten durch entsprechende JUnit-Tests abgedeckt werden.</p>
Rolle	Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Windows 7 Enterprise, Android 4.2.2 Java SE/EE, Eclipse, Design-Patterns (insb. Visitor Pattern), OOA/OOD, UML 2, Maven, SVN, SQL, Oracle 11g, JAXB, PKI, XML, HTML, VDV-KA Spezifikation Vers. 1.1.09; Agile Software-Entwicklung, Wasserfall-Modell, Codereviews Testing: JUnit 4.5, Mockito, HP Quality Center, Jenkins
Aktivitäten	<ul style="list-style-type: none"> • Einarbeitung in die Thematik, VDV-KA Spezifikation und Architektur bzw. Funktionsweise des eTicket-Handlers und dessen Integration in die Funktionseinheiten Ticketautomat, MTx und DDS • Aufbau einer im Projekt eingesetzten Java-Entwicklungsumgebung, basierend auf der Eclipse IDE, einer Simulation für Ticketautomaten und einer speziellen Android Umgebung, die in dem Kontrollgerät <i>MTx</i> zum Einsatz kommt • Umfangreichere Fehlerbehebungen, meist verbunden mit lokalem Refactoring und verbesserten Log-Ausgaben des eTicket-Handlers. Für das Fehlertracking wurde HP Quality Center verwendet • Implementierung einer Anforderung für einen bestimmten Verkehrsverbund, der eine flexible Strategie zum Löschen von Berechtigungen auf der Karte benötigte • Anpassung der „räumlichen Prüfung“ im eTicket-Handler anhand neuerer Anforderungsbeschreibungen für zwei Verkehrsverbünde • Durchführung von Codereviews bei / nach umfangreicheren Arbeiten • Integration des eTicket-Handlers in die Funktionseinheiten MTx und Ticketautomat. Das Verhalten des Ticketautomaten selbst konnte mit einer Softwaresimulation nachgestellt werden • Erstellung oder Vervollständigung von JUnit-Tests für wichtige bestehende bzw. korrigierte Funktionalitäten im eTicket-Handler. Das Verhalten der Datenbank oder der Kartenleser wurden ggfs. durch Mockito-Konstrukte überbrückt. Automatische Builds und ein Monitoring erfolgte über Jenkins • Erstellung von Systemdokumentationen • Erstellung neuer bzw. Erweiterung bereits existierender Tools
Erfolg	Stabilisierung des eTicket-Handlers; alle entstandenen Kundenanforderungen und Problemfälle konnten erfolgreich abgeschlossen werden.

1 Die VDV-Kernapplikation (kurz: VDV-KA) ist ein offener Daten- und Schnittstellen- Standard für das elektronische Fahrgeldmanagement (Electronic Ticketing) im Öffentlichen Personenverkehr in Deutschland, der durch das Bundesministerium für Bildung und Forschung (BMBF) gefördert wird.

Zeitraum	2014, 5 Monate
Branche	Banken
Projekt <i>Trier</i>	<p>Als Teil des Deutsche Bank - IT-Projektes <i>Magellan</i> wurden <i>in</i> einer ehemals akquirierten Privatkundenbank unter dem Projektnamen <i>Trier</i> verschiedene Ratingsysteme in eine unternehmensweite Lösung integriert. Die Projektbeauftragung erfolgte durch ein u. a. auf Banken spezialisiertes Beratungsunternehmen.</p> <p>In diesem Kontext sollte eine bereits bestehende Three - Tier Rating-Anwendung in einem neuen Release um die vollständige Erfassung und die zugehörige Backend-Verarbeitung von Kundendaten erweitert werden, um eine Ratingberechnung durchführen zu können. Hierbei sollten zur Vermeidung von Redundanzen bereits existierende Daten aus anderen Systemen (z. B. aus externen Datenbanktabellen oder über ein Inhouse SAP System) in die Business Logik mit eingebunden werden. Ebenso sollte für die Berechnung auf schon existierende Modelle und Regelmechanismen zurückgegriffen werden.</p>
Rolle	Berater, Designer, Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	<p>Windows XP Professional / Solaris, Eclipse IDE/RCP, Java SE/EE, EJB 3.0, MVC-, DAO Pattern, OOA/OOD, Java Swing / JGoodies, JPA, JDOM, Ant, CVS, XML, Oracle DB, Weblogic Server 10.3, CRS - API (WebService zur DB)</p> <p>Testing: JUnit, Vorlagen des Fachfeinkonzepts (manuell, Dummy Client)</p>
Aktivitäten	<ul style="list-style-type: none"> • Einarbeitung in die bestehende Rating-Anwendung (fachlich und software-technisch); der Java Swing Client (eine unter Eclipse RCP entwickelte RCP Anwendung) und die Businesslogik (EJB Module) • GUI - Erstellung mit Hilfe von Java Swing und JGoodies, einer Swing Erweiterung; das Layout war größtenteils sehr komplex und umfangreich; Layout bzw. nachträgliche Änderungen am Fachfeinkonzept, das als Ausgangsbasis diente, erfolgten in enger Abstimmung mit dem Fachbereich der Privatkundenbank und den Business Analysten im Projektteam • Abgekoppelt von Datenbank und Oracle AS erfolgte eine Bereitstellung von Testdaten je Eingabemaske in einem Dummy Client • Architektonisch erfolgte eine strikte Einhaltung des MVC – Design Patterns; serverseitige Persistierung der Daten via JPA in einer Oracle Datenbank, gekapselt durch Data Access Objects (DAO); Implementierung der Business Logik in Session Beans (EJB Module) • Hauptansprechpartner für das QS-Team und die Business Analysten im Projekt während einer länger andauernden Urlaubsvertretung für den hauptverantwortlichen Softwareentwickler im Entwicklerteam. So sollte ein Vorgängerrelease der Anwendung getestet bzw. dokumentiert werden. Insbesondere war ich für Fragen, entstandene Fehlersituationen im Datenfluss über die neu entwickelte CRS - Schnittstelle zur Deutschen Bank und ihrer Behebung verantwortlich • Review entwickelter Programmteile im neuen Release, die die CRS - Schnittstelle thematisch behandelten, z. B. Aufbau eines JDOM Baumes, der später via XML in den Webservice überführt wird
Erfolg	Rechtzeitig vor Start des kundeninternen Releasetests konnten alle Meilensteine der Beauftragung zu voller Zufriedenheit erfüllt werden

Zeitraum	2013 - 2015, mit Unterbrechungen, bisher 11 Monate
Branche	Eigene Produktentwicklung, Social Media u.a.
Produktentwicklung <i>Address-Event-Recognizer</i>	<p>Textanalyse durch teilweise natürlich sprachliche Texterkennung und semantische Zuordnung oder bei statischen Texten über Templates.</p> <p>Ein in einer beliebigen Anwendung (Textverarbeitung, Internetbrowser, Mailprogramm, ...) markierter oder importierter Text soll bzgl. seines logisch zusammenhängenden Kontextes wie z. B. Kontakt, Termin bzw. Event oder auch einem komplexeren Kontext wie eine Rechnung erkannt werden. Die so ermittelten Objekte werden in einem zweiten Schritt über ein Mapping als kontextbezogene strukturierte Daten für weiterverarbeitende Programme (wie Adressbuch, Terminkalender oder Buchhaltung) zur Verfügung gestellt.</p> <p>Das Produkt befindet sich noch in der Entwicklung. Eine erste Version wird eine Erkennung von Terminen und Kontakten enthalten.</p>
Rolle	SW-Architekt, Designer, Softwareentwickler
Eingesetzte Techniken	<p>Mac OS X / Windows / iOS, Java SE/EE, Datenmodellierung, Design Patterns, OOA/OOD, Expertensysteme (KI), UML, XML, Eclipse, Maven, SVN, Commons – MultiMaps, JDOM, SAX/JAXP, JPA</p> <p>Testing: JUnit</p>
Aktivitäten	<ul style="list-style-type: none"> • Festlegung der Architektur und der Datenmodelle unter Berücksichtigung einer leichten Wartbarkeit und späteren Erweiterbarkeit auf andere Kontexte; möglichst Verwendung etablierter Standards • Definition einer XML basierten Konfigurationsdatei zur Spezifikation der später zu erkennenden Artefakte und Kontexte • Aufbau eines dynamischen Regelbaumes und mehrerer Schlüsselwortbäume, die als zentrale Elemente zur Textanalyse hinzugezogen werden (Verw. von Apache Commons – MultiMaps) • Mehrsprachlichkeit (Sprachen nach ISO8859-1, zuerst „de“, „en“) • Ermittlung länderspezifischer und anderer sprachlich relevanter Parameter wie z. B. Zeitzone oder Datums- und Uhrzeitformat • Interpretation des Textes je nach Modus mit unterschiedlichen Parsern: natürlich sprachliche Texterkennung bei fortlaufendem Text oder Verwendung von Templates bei Textblöcke • Erzeugung der daraus abgeleiteten und vollständigen Objekte • TODO: Zugriff auf externe Datenquellen über z. B. Webservices, um bestimmte Felder wie z. B. Ortsangaben oder ein bestimmtes Format einer Rechnungsnummer zu verifizieren • Persistente Speicherung der Suchbäume (s.o.) und anderer häufig benutzter Daten mittels JPA in einer Datenbank • TODO: Datelexport von Kontakt / Adresse, Termin oder Rechnung in ein abhängiges Standardformat (zuerst nur Adresse und Termin) • Erstellung von JUnit Tests • TODO: Auswahl einer geeigneten Client-Server Architektur, clientseitig verwendbar auf Desktops, Smartphones oder Tablets: <ul style="list-style-type: none"> • clientseitiger Ansatz: JavaFX - RoboVM (iOS) / Dalvik (Android) • serverseitiger Ansatz: JSF 2.2, PrimeFaces oder ZK (Zkoss)
Erfolg	Verwendung verschiedener Parser Frameworks oder den Commons – MultiMaps, Anwendung natürlich sprachlicher Texterkennung (KI)

Zeitraum	2013, 2,5 Monate
Branche	Handel
Projekt (Produktentwicklung) <i>Directory-Watcher</i>	Refactoring und Weiterentwicklung des universellen Batchverarbeitungs- und Queueing Programmes Directory-Watcher (siehe 1. Version in 2010 / 2011 und auch eine genauere Beschreibung). Intensive Tests der parallel laufenden „Arbeits-Threads“ und Sicherstellung, das auch im Fehlerfalle jeder dieser Threads sich beendet bzw. wohl definiert beendet wird.
Rolle	Architekt, Designer, Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Mac OS X / Windows, Batchprogrammierung, Parallele Programmierung (Multithreading), Design Patterns, OOA/OOD, Eclipse, Xcode, Ant, SVN, Java SE, Java Swing, jpathwatch; Testing: JUnit, SonarQube
Aktivitäten	<ul style="list-style-type: none"> • Kompletter build Prozess jetzt auch unter Windows möglich, unter OS X wird eine Application (.app) erzeugt (enthält .jar Dateien) • Stabilisierung des parallel laufenden Prozessgerüsts; Vermeidung nicht terminierter Threads, die z. B. durch einen Fehlerfall einer Operation nicht ordnungsgemäß beendet werden konnten • Einbindung eines SonarQube Ant PlugIns zur automatisierten Erstellung einer Sonar-Analyse • Fehlerbehebung und Refactoring (auch laut Sonar-Analyse): <ul style="list-style-type: none"> • Verfeinerung der Bestimmungsalgorithmen von UTF-8 Dateien • Zulassung eines flexibleren Parsen der Dateilisten • Testen unter Mac OS X und Windows, manuell und mit JUnit
Erfolg	Das Programm wird wegen seiner großen Flexibilität weiterhin mit Erfolg beim Kunden und auch intern bei mir eingesetzt.

Zeitraum	2012, 3 Monate
Branche	Open Source Projekt
Projekt <i>AirReceiver</i>	Weiterentwicklung und Stabilisierung des Open Source Projektes <i>AirReceiver</i> , mit dessen Hilfe Audiodaten von Apple iOS Geräten oder vom Programm iTunes via Apple's eigenem <i>AirPlay</i> Protokoll an einen Desktop PC versendet werden können. Neben dem Kennenlernen der verwendeten Middleware Netty und des AirPlay Protokollstacks inklusive der ausserdem verwendeten Protokolle standen auch Maven und insbesondere die Einbindung des SonarQube Maven PlugIns im Mittelpunkt des Interesses.
Rolle	Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Mac OS X / Windows, Eclipse, Java SE, Fat Client Entwicklung, Maven, SVN, Parallele Programmierung (Multithreading), JCA/JCE, SonarQube, Netty (asynchrones Event getriebenes Netzwerk Application Framework); AirPlay: RAOP, RTSP/RTP, NTP, Multicast DNS (jmDNS), SDP
Aktivitäten	<ul style="list-style-type: none"> • Einarbeitung in die bestehende Software und den dabei verwendeten Protokollstack (RAOP, RTSP/RTP, SDP) und Libraries wie jmDNS (Multicast DNS) oder der Netty Middleware (s. o.) • Stabilisierung und Optimierung des RTP- Retransmit Mechanismus bei nicht oder fehlerhaft übertragenen RTP Paketen • Einbindung eines SonarQube Maven PlugIns zur automatisierten Erstellung einer Sonar-Analyse • Refactoring an diversen Stellen (inkl. Fehler laut Sonar-Analyse) • Umbau des Maven build Prozesses durch Verwendung von Profilen, um plattformabhängig unterschiedliche Endprodukte zu erlangen (unter OS X: .app, unter Windows ein einziges ausführbares .jar)
Erfolg	Open Source Entwicklung, Kennenlernen von AirPlay Protokollstack, Middleware Netty, Maven Profile, SonarQube

Zeitraum	2012, 7 Monate
Branche	ITK (IT + TK)
Projekt <i>ISTP</i>	<p>Erstellung eines neuen Releases des hochverfügbaren Monitoring-, Scheduling- und Dispositionssystems ISTP (Informationssystem Transportleitung Personenverkehr), welches der Überwachung des Soll-Fahrplans im gesamtdeutschen Schienennetz für die Transport- und Verkehrsleitungen der Deutschen Bahn AG dient. Das aus mehreren Fat Clients bestehende System wird mit seinen über 20 Schnittstellen u. a. auch vorrangig als Datenquelle für andere Systeme (wie z. B. RIS) der Deutschen Bahn AG herangezogen.</p> <p>Das neue Release bestand im wesentlichen in der Umsetzung folgender neuer Anforderungen der Konzernmutter Deutsche Bahn AG: neben der bisherigen Disposition von Zügen sollte nun auch die Disposition der eingesetzten Fahrzeuge und des Zugpersonals realisiert werden. Weiterhin sollten die Anschlüsse von Zügen aus Sicht der Abbringerzüge besser disponierbar und kommunizierbar sein.</p>
Rolle	Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	<p>Windows Vista Enterprise, Java SE/EE, Fat Client Entwicklung, Eclipse, Java Swing, Design-Patterns, OOA/OOD, Ant, SVN, SQL, PL/SQL, Oracle 11g, Enterprise Architect 7.x, UML, XML, Tomcat 6.0, Entronal und andere eigene vom Kunden entwickelte Werkzeuge;</p> <p>Agile Software-Entwicklung / Stand-Up Meetings</p> <p>Testing: JUnit 4.5, Fixtures, HP Quality Center, Jenkins</p>
Aktivitäten	<ul style="list-style-type: none"> • Einarbeitung in die Thematik, Konzeption und Architektur, von ISTP, insbesondere mit Fokus auf die neuen Anforderungen (s. o.) • Aufbau einer im ISTP Kontext eingesetzten Java-Entwicklungsumgebung, basierend auf der Eclipse IDE und dem Apache Tomcat Server • Bugfixing „leichterer“ Fehler zwecks erstem Kennenlernens des Javacodes und der etablierten Vorgehensweisen im ISTP Projekt. Exemplarisch wurden Fehler in dem ISTP- Administrationstool behoben. Für das Fehlertracking wurde das Softwarewerkzeug HP Quality Center verwendet • Neu-Architektur und Re-Implementierung einer DAO Schicht, um u.a. drei wesentliche Entitäten nach etwa gleichem Schema in der verwendeten Oracle Datenbank persistent abzubilden. Historisch bedingt wird in ISTP eine eigene proprietäre Persistenz-API verwendet und darüberhinaus werden alle Datenbankzugriffe über PL/SQL Prozeduren gekapselt. Für die in der DAO Schicht verwendeten Objekte mussten ebenfalls entsprechende PL/SQL Prozeduren geschrieben werden • Implementierung weiterer Objekte, wie z. B. Fahrzeug- oder Personalübergänge, die auf obigen DAO Objekten aufsetzten • Erstellung umfangreicher und vollständiger JUnit-Tests für alle wichtigen Zugriffsmethoden der neuen DAO Objekte. Verwendung von Fixtures zum Bereitstellen von Testdaten • Refactoring des ISTP-Administrationstools hinsichtlich seiner Datenbankzugriffe
Erfolg	Die neuen DAO's bildeten u. a. den Kern des neuen Releases, das auch später erfolgreich intern beim Kunden abgenommen wurde.

Zeitraum	2011, 5 Monate
Branche	ITK (IT + TK)
Projekt <i>Mobiserv</i>	<p>Für eine beim Kunden sich schon im Einsatz befindliche Java basierte Client / Server Anwendung (Thin Client auf Clientseite) zur Disposition von Bahnmitarbeitern bei Ihrer Ausübung von Hilfeleistungen für Fahrgäste mit eingeschränkter Mobilität stand eine neue Programmversion mit erweiterter Funktionalität für die Abnahme bereit.</p> <p>Aufgabe war, für diese neue Version einerseits die neue Funktionalität und andererseits auch die bisherige Funktionalität an entscheidenden und kritischen Stellen im Programmcode per JUnit Tests zu verifizieren und gegebenenfalls Korrekturen gefundener Fehler vorzunehmen.</p>
Rolle	Softwareentwickler, Qualitätssicherung, Operating, Administration
Eingesetzte Techniken	<p>Windows Vista Enterprise / RedHat Linux, Java SE/EE, OOA/OOD, Eclipse, JPA, Ant, SVN, Thin Client, Parallele Programmierung (Multi-threading), XML, JSF 1.2, Tomcat 6.0;</p> <p>Agile Software-Entwicklung / Pair Programming, Codereviews</p> <p>Testing: JUnit 4.5, HP Quality Center, Jenkins / Hudson</p>
Aktivitäten	<ul style="list-style-type: none"> • Einarbeitung und Aufbau einer üblicherweise beim Kunden angepassten Java-Entwicklungsumgebung (SEU), basierend auf der Eclipse IDE • Erstellung von JUnit Tests für verschiedene Module, aber besonders für die Dispositionsmodule, um eine der neuen Funktionalitäten, die parallele Disposition von mehreren Clients ausgehend zu überprüfen • verstärktes Bugfixing und lokal eingesetztes Refactoring im Gesamtcode: diese Notwendigkeit ergab sich durch einige negative Ergebnisse der JUnit Tests und durch die Aktivitäten eines parallel arbeitenden Testteams, das vermehrt Fehler in der Altanwendung festgestellt hatte • Reviews und „Pair - Programming“ ausgesuchter Programmteile • Continuous Integration: regelmäßige Erstellung einer neuen Version der Anwendung; später konnte für Teilaufgaben auch Jenkins (unter Windows) bzw. Hudson (unter Linux) verwendet werden • Monitoring und Fehleranalyse am Gesamtsystem (Linux, Tomcat) • Durchführung systemadministrativer Tätigkeiten
Erfolg	Mittels Bugfixing und lokalem Refactoring konnte die anfangs stark ansteigende Fehlerrate wieder gesenkt werden. Durch den Einsatz eines Continuous Integration Werkzeugs wie Jenkins und Hudson waren auch schnelle Reaktionen auf veränderte Versionsstände ermöglicht worden.

Zeitraum	2010 / 2011, 5 Monate
Branche	Handel
Projekt (Produktentwicklung) <i>Directory-Watcher</i>	<p>Universeller, Ereignis gesteuerter Batchverarbeitungs- und Queueing Mechanismus zur Verarbeitung beliebig vieler konfigurierbarer Batchjobs, die parallel oder sequentiell ausgeführt werden.</p> <p>Das System wird in Form eines allgemein benutzbaren Verzeichnisüberwachungsdienstes für Dateilisten realisiert: für beliebig aufgebaute Dateilisten², die von einer beliebigen Anwendung als Batchjobs in einem dediziertem Verzeichnis abgelegt werden, wird für jede Dateireferenz aus einer derartigen Dateiliste eine vom Benutzer frei konfigurierbare Operation (Step) ausgeführt.</p> <p>Dieses Projekt ist gleichzeitig auch eine eigene Produktentwicklung.</p>
Rolle	Architekt, Designer, Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Mac OS X / Windows, Batchprogrammierung, Parallele Programmierung (Multithreading), Design Patterns, OOA/OOD, Eclipse, Xcode, Java SE, Ant, SVN, Java Swing, jpathwatch; Testing: JUnit
Aktivitäten	<ul style="list-style-type: none"> • Gesamtkonzeption eines Ereignis gesteuerten Verzeichnisüberwachungsdienstes, der aus einem GUI und einem im Hintergrund laufenden passiv agierenden Pollingprozess besteht, der seinerseits für jede eintreffende Dateiliste (Ereignis) den entsprechenden Batchjob erzeugt und in eine Warteschlange (Queue) einstellt. • Die Spezifikation der verschiedenen zu erwartenden Arten von Dateilisten erfolgt über Property-Dateien (pro Art eine Property); Implementierung zum Lesen und Schreiben dieser Properties • Architektur und Implementierung des benötigten Prozessgerüsts (Multi-Threading) zum Abarbeiten der Queue: je Property wird einmalig ein neuer parallel ablaufender „Arbeits-Thread“ gestartet, Jobs der gleichen Property werden jedoch sequentiell ausgeführt • (Java - Thread-) Synchronisation und Eingrenzung maximal parallel laufender Arbeitsthreads erfolgt durch einen Semaphor • Implementierung der möglichen Operationen (Steps) je Property: Basisoperationen, wie <i>Copy</i>, <i>Move</i> und <i>Delete</i> oder einem beliebig konfigurierbaren externen Programmaufruf • Graphische Fortschrittsanzeige aller gleichzeitig aktiven Arbeitsthreads in separaten Java Swing Fenstern • Separate Logausgabe des Hauptprozesses und der gerade aktiven Arbeitsthreads, dabei sind jeweils vierstufige Loglevel konfigurierbar • Implementierung des GUI mit Java Swing: <ul style="list-style-type: none"> ○ Start / Stop der Überwachung (drei verschiedene Stop - Modi) ○ Konfiguration: Hinzufügen, Editieren und Löschen der zu verwendenden Operationen pro Dateilisten Spezifikation; Konfiguration des Hauptprozesses und des Semaphors ○ unter Mac OS X: Verwendung OS X - spezifischer Menüs • Testen unter Mac OS X, Windows und Linux, manuell und mit JUnit
Erfolg	Das Programm wird wegen seiner großen Flexibilität mit Erfolg beim Kunden und auch intern bei mir eingesetzt

2: *Dateiliste*: eine Textdatei, die beliebig viele Dateinamen (mit absoluter oder relativer Pfadangabe) enthält

HINWEIS!!!

Alle nachfolgenden Projekte sind aus der kontinuierlichen Produktweiterentwicklung des branchenübergreifenden und J2EE- basierten EAI- / B2B- Integrationswerkzeuges und Workflow- Managementsystems Sterling Integrator der Firma Sterling Commerce (seit Mai 2010 von IBM akquiriert) hervorgegangen.

Zeitraum	2006 - 2009, 3 Jahre
Branchen	europäische Automobilindustrie, Transportwesen, Finanzwesen, Chemie
Projekt (Produktentwicklung) <i>OFTP</i>	Planung, Architektur, Entwicklung und Implementierung eines in den oben genannten Branchen verwendeten Datenübertragungsprotokolls OdetteFTP (bzw. OFTP) zur direkten Übertragung von Dateien (Punkt-zu-Punkt Kommunikation) in Form eines Adapters für das Integrationswerkzeug Sterling Integrator
Rolle	Architekt, Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	<p>Unix / Windows, Eclipse IDE, Oracle SQL Developer, Java SE, J2EE, Java Script, JSP, OOA/OOD, Design Patterns, CVS, Ant, JCA, PKI, XML, JDBC, SQL, JDOM, SAX;</p> <p>Protokolle: OdetteFTP (Version 1.1 – 2.0), IPv4, IPv6;</p> <p>eingesetzte AS: JBoss, Weblogic, Websphere, Noapp (Eigenentwicklung)</p> <p>Feature-Driven und testgetriebene Software-Entwicklung</p> <p>Testing: JUnit, Regressionstests, Integrationstests, CruiseControl, HP Quality Center</p>
Aktivitäten	<ul style="list-style-type: none"> • Architektonische Gesamtkonzeption (OOA/OOD) anhand der OFTP Spezifikationen (RFCs 2204 und 5024); Herausarbeitung der erforderlichen Teilkomponenten • Konzeption und Implementierung einer Partnerprofilverwaltung für OdetteFTP-Partner; die Datenhaltung wurde zunächst in XML mittels SAX-API realisiert, in einer nachfolgenden Version datenbankgestützt via JDBC und SQL • Entwicklung eines synchronisierenden Import- und Exporttools, um die XML- basierten und die in der Datenbank gehaltenen Partnerdaten abgleichen und bearbeiten zu können (z. B. Einzel- oder Massenimport bzw. Datenbankexport) • Implementierung eines Schedulers, der basierend auf einmaligen oder mehrfach periodisch auftretenden Kalendereinträgen spezifische OFTP Operationen oder andere systemweite Services ausführt • Verschiedene Teilimplementierungen, wie OdetteFTP Zustandstransformationen oder eine auf PKI (Public-Key-Infrastruktur) basierte Zertifikateverwaltung • IPv4- und IPv6 Fähigkeit bei TCP/IP Kommunikation: besonderes Augenmerk galt hier der syntaktischen Analyse von IPv4- und IPv6- Adressen • Qualitätssicherung: Testfallerstellung, JUnit Testerstellung • Dokumentation für das Benutzerhandbuch (in Englisch)
Erfolg	Sterling Commerce hat ca. 50 Sterling Integrator Installationen mit diesem Adapter an Neukunden verkauft (Stand 09/2009)

Zeitraum	2005 / 2006, 12 Monate
Branchen	Alle Branchen
Projekt (Produktentwicklung) <i>MESA</i>	Entwicklung eines neuen Produktes „MESA Developer Studio“ für Sterling Integrator, architektonisch konzipiert als eine Eclipse Rich Client Platform, die u. a. den Kunden eine eigene Modulentwicklung, Monitoring und ein individuell anpassbares Web Design für Sterling Integrator ermöglichen soll.
Rolle	Softwareentwickler in nationalem und internationalem Team
Eingesetzte Techniken	Unix / Windows, Java SE, J2EE, OOA/OOD, Eclipse, Eclipse RCP, Plugin Entwicklung, CVS, Ant, WebDAV, Java RMI (Middleware), SQL, XML; Testing: JUnit, Regressionstests, Integrationstests, CruiseControl, Mercury TestDirector
Aktivitäten	<ul style="list-style-type: none"> • Erstellung eines Export Wizards als Eclipse Plug-in, der alle oder selektierte benutzerdefinierten Objekte innerhalb des MESA Developer Studios als .war – Datei exportiert • Ermöglichung eines vereinfachten Deployments in Sterling Integrator und dem verwendeten Appl.Servers, um mit MESA Developer Studio erstellte Module in Sterling Integrator leichter verfügbar zu machen • Erstellung und Durchführung von Tests • Dokumentation für Benutzerhandbuch (in Englisch)
Erfolg	Neues Produkt „MESA Developer Studio“. Es ist mir aber nicht bekannt, wieviele Lizenzen Sterling Commerce hiervon verkauft hat.

Zeitraum	2004 / 2005, 6 Monate
Branchen	Alle Branchen
Projekt (Produktentwicklung)	Im Rahmen einer Anbindung von Sterling Integrator an das ERP System SAP R/3 sollten bereits auf der Basis von Unix Shell Script existierende Tests (Batch Testing, Smoke Tests) in ein Java-basiertes und Betriebssystem unabhängiges Regressionstestsystem (unter JUnit) transformiert werden.
Rolle	Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Unix / Windows, bash, ksh, Java, Eclipse, CVS, Ant, XML; Testing: Batch Testing, JUnit, Regressionstests, Smoke Testing, Integrationstests, CruiseControl, Mercury TestDirector
Aktivitäten	<ul style="list-style-type: none"> • Schrittweise Umstellung dreier verschiedenen Testtypen (Integrations-, Komponenten- und „stand-alone“ Tests) in Regressionstests unter Verwendung von JUnit • Erstellung zusätzlicher Komponententests um eine größere Abdeckung der zu testenden Funktionalität zu erreichen • Aufstockung aller Tests um Soll-Ist-Vergleichs Methoden • Durchführung umfangreicher Tests in der neuen Umgebung
Erfolg	Schaffung einer plattformunabhängigen Testumgebung, insbesondere die Windows Umgebung wurde damit neu einbezogen => Senkung der Fehlerrate, Vereinfachung des Hinzufügens neuer Tests

Zeitraum	2003 / 2004, 21 Monate
Branchen	Alle Branchen, die SAP einsetzen
Projekt (Produktentwicklung) <i>SAP Adapter</i>	Entwicklung einer Java Programmierschnittstelle (API) zur Weiterverarbeitung von Metadaten (SAP IDOC), welche in einem ERP System (SAP R/3) hinterlegt sind. Dies erfolgte im Rahmen einer Anbindung von Sterling Integrator an ein SAP R/3-System (SAP Adapter), zunächst in Form eines Kommandozeilen-Tools, später integriert in die grafische Benutzeroberfläche von Sterling Integrator
Rolle	Softwareentwickler, Qualitätssicherung
Eingesetzte Techniken	Unix / Windows, Java SE, Java Script, JSP, OOA/OOD, JDBC, SQL, Eclipse, CVS, Ant, SAPJco, JMS, SAP RFC, SAP GUI, RMI, XML; eingesetzte Appl.Server: JBoss, weblogic oder Websphere Testing: Regressionstests, Integrationstests, Smoke Testing, TestDirector
Aktivitäten	<ul style="list-style-type: none"> • Erster Teil des Projekts: Recherche und Evaluierung von RFC Aufrufen, wie mit der zu benutzenden Middleware (SAPJco) Strukturen aus einem SAP System im gewünschten Maß gelesen werden können • Erstellung der API unter Verwendung der gefundenen RFC Aufrufe, die die Weiterverarbeitung der Metadaten ermöglicht • Implementierung eines Kommandozeilentools unter Verwendung der erstellten API • Erstellung und Durchführung von umfangreichen Tests • Dokumentation des Kommandozeilentools im Benutzerhandbuch (in Englisch) • Zweiter Teil des Projekts: Integration der API in die graphische Benutzeroberfläche von Sterling Integrator; Entwurf und Realisierung entsprechender Masken unter Verwendung von JSP und Java Script • Erstellung und Durchführung von umfangreichen Tests • Dokumentation des GUI für das Benutzerhandbuch (in Englisch)
Erfolg	Sterling Commerce hat über 250 Sterling Integrator Installationen mit diesem Adapter an Alt- und Neukunden verkauft (Stand 09/2009)

Projekte und Tätigkeiten in der Qualitätssicherung

Die nachfolgenden Projekte beziehen sich auf Tätigkeiten in der Qualitätssicherung bei der Firma Sterling Commerce und davor bei o.tel.o communications GmbH & Co.

1998 - 2003 **Senior Quality Assurance Analyst (Sterling Commerce GmbH)**

für den Bereich Qualitätssicherung und Software-Entwicklung Regional Lab

- Testfallerstellung und Automatisierung manueller Testabläufe bzgl. der SAP R/3-Kopplung zwischen einem SAP R/3 System und dem EAI-/B2B Integrationswerkzeug Sterling Integrator. Aufsetzend auf einer bereits existierenden Shell Script-basierten Testumgebung (Batch Testing) wurde diese Umgebung für die o.g. SAP R/3 Kopplung erweitert; inkl. anschließendem Monitoring; Zum Einsatz kamen hierbei Testmethoden wie Regressionstests, Smoke Testing und Integrationstests
- Eigenverantwortliche Durchführung von Qualitätsmanagement- und Wartungsaufgaben sowie Realisierung kundenspezifischer Anpassungen (z. B. Jahr 2000-Umstellung) eines von o.tel.o communications GmbH & Co. übernommenen EDI-Systems

1994 - 1998 **Qualitätssicherungsbeauftragter (o.tel.o communications GmbH & Co)**

Qualitätsmanagement und Durchführung von Qualitätssicherungsabnahmen für ein EDI-System in selbstständiger Arbeit und eigener Verantwortung:

- Definition von Testfällen, Erstellung von Testdaten sowie manuelle Durchführung und Dokumentation der Tests
- Entwurf und schrittweise Implementierung automatisierter Testprozeduren
- Aufbau, Administration und Dokumentation einer Testumgebung
- Aufbau und Etablierung eines Fehlermanagement- und Anforderungssystems
- Kontinuierliches Fehlermanagement und -tracking
- Schnittstellenfunktion im Qualitätssicherungsprozess zwischen Software-Entwicklung, Support und Consulting