

Mitarbeiterprofil

Name:	Thomas Boor
Ausbildung:	Studium der Elektrotechnik Organisationsprogrammierer
Methoden:	Objektorientierte Analyse und Design mit UML und Coad & Yourdon, Strukturierte Analyse und Design, ISOTEC, Datenbankdesign (ER-Modellierung) und -Normalisierung
Hardware:	IBM DataBlades, Sun Fire, Sun Netra, SNI RM- & MX-Serie, IBM RS/6000, SUN SPARCstations NIXDORF TARGON /35, IBM /370, PCs
Betriebssysteme:	SuSE-Linux, AIX (bis 6.1), Sun-Solaris 10, Linux (Ubuntu., Arch-Linux, Mint), SINIX SVR4 (bis Rel. V.4.3), DG-UX 5.4, SCO-UNIX 3.2, SunOs 5.4, MS-DOS/WINDOWS, VM/SP, MVS/ESA
Programmiersprachen:	C++ (bis v14), C, Python (2.7 und 3.6)
TK / Netze:	TCP-UDP/IP, BSD-Sockets, ONC-RPCs, RMI/JNDI, CMX
Protokolle:	SIP, SDP, RTP, RTCP, Radius, SNMP, http, Soap
DB-Systeme:	REDIS, DB2V9-Vista, Informix Online bis 10.0, Oracle 7.3, Postgres, MySQL, ddb4, CICS, VSAM, MS-QLServer, MS- Access
Sonstiges:	PyCharm, Atom, Eclipse-SDK, Sun-Forte-IDE, Together, Rational Rose, UNIX-Shells & tools scons, git, gerrit, jenkins, redmine, confluence/draw-io omniORB, omniORBpy, SonarQube
Veröffentlichungen:	<u>PolarPlot - Plotten von Funktionen in Polarkoordinaten</u> mit Sharp PC-1500/CE-150 Würzburg, 1987 (Vogel-Verlag, NE: T.Eikenkötter) <u>vi-Referenzhandbuch -</u> <u>Das Lehr- und Nachschlagewerk zum</u> <u>UNIX-Standardeditor</u> München, 1996 (Verlag Prentice Hall, NE: Hutter/Pribas)

Executive Summary

(Kundenbezogener Einsatz)

Zeitraum	Auftraggeber	Alt. Kunde(n)	Tätigkeit
06.2021 – 06.2025	IBM Frankfurt	DTAG	Support bei Betriebsaufnahme
06.2020 – 06.2021	RM Electronics		SW-Entwicklung C++
06.2006 – 02.2020	IBM NGN CC	DTAG, TSI, terravoice, IBM	SW-Design u. -Entwicklung in C++, C, Python, Teamleitung
05.2004 – 09.2006	Ticketcorner GmbH		SW-Design u. Entwicklung in C++, Python
04.2002 – 03.2004	Qivive GmbH		SW-Design u. Entwicklung in C++
09.2000 – 03.2002	START-Informatik		SW-Design u. Entwicklung in C++
10.1998 – 08.2000	START-Ticket		SW-Design u. Entwicklung in C++ QS f. SW-Abteilung
01.1998 – 09.1998	IBM Frankfurt	otelo	Regressionstest
02.1992 - 12.1997	START-Ticket		SW-Design u. Entwicklung in C++

Kleinere, parallel ausgeführte Projekte sind hier nicht gelistet

Detaillierte Einzel-Projektliste

(Angabe in umgekehrt chronologischer Reihenfolge)

Support bei Betriebsaufnahme einer bundesweiten Plattform

Zeitraum	06/2021 - 06/2025
Branche	Telekommunikation
Rolle	Berater & Servicekraft
Kunde / Auftraggeber	DTAG / IBM
Tätigkeit	Support bei Einführung einer DevOps-orientierten Deployment-Toolchain, Schreiben von ansible-playbooks zur automatisierten Validierung des deploy-Ergebnisses. Python-Tools für bundesweite N:M-Verbindungsprüfungen. Schulungen zu fortgeschrittener Nutzung der git-Versionsverwaltung. Beratung zum Thema Canary-Deployment. Wiki-Dokumentation zu allen Tasks inkl. Abbildungen. AI (machine-learning und NaturalLanguageProcessing) basierte python scripts um wöchentliche reports mit tagesgenauen Daten bis zur akt. Woche zu validieren und die kommende Woche zu prognostizieren (inkl. Vertrauens-Schranken), präsentiert als Dreifach-Graph.
Methoden	scrum
Programmiersprache(n)	Python 2.7 u. 3.6, bash, ansible, Jinja2
Betriebssystem(e)	Suse-Linux
IDEs	MS VisualStudio, ansible-tower
Tools	git, gitk, gitlab, ansible-tower, artifactory, Vault, BIG-IP F5. Anritsu, jira, gerrit, confluence, guard, topedia, OpenOffice. CheckMK, Kibana, anritsu
Sonstiges	Webex, Slack

Softwareentwicklung im Bereich Fahr-Simulation

Zeitraum	06/2020 - 06/2021
Branche	Simulations-Software
Rolle	Entwickler
Kunde / Auftraggeber	RM Electronics
Tätigkeit	Erstellung von Software-Komponenten für grosse Simulationssysteme mit Bosch-Rexrodt-Bewegungssystem, Dell 180° Monitoren, force-feedback seats & wheels - Treiber f. Logitech-Wheel 920 - Gateway-Prozess zur Konvertierung von UDP-Sensordaten in SHMs - Generator für Codefragmenten gemäß von Device-Listen - Logik-Prozess zum Verwalten von Fahrzeug-Systemzuständen
Methoden	scrum
Programmiersprache(n)	C++ V11, python 3
Betriebssystem(e)	Suse-Linux, CentOs, Azure-Linux, Ubuntu
IDEs	Eclipse, PyCharme
Tools	SonarQube, gitlab, git, gitk scheme, tk, jira, gerrit, confluence, draw-io
Sonstiges	webex, mattermost

IoT-Überwachungs- und Steuerungssystem für autonome Energieversorgung (PV und Windkraft) Parallel-Projekt

Zeitraum	01/2019 - 03/2020
Branche	Energie
Rolle	Designer / Entwickler
Kunde / Auftraggeber	thovid.com
Tätigkeit	<p>konzeption und Realisierung eines Überwachungs- und Steuerungssystem für autonome Energiegewinnung mittels PV und/oder Windkraft.</p> <p>Auf einem raspberryPi mit Debian-Linux sammelt ein in Python geschriebener Collector Daten von den Energiequellen (Strom, Spannung, Drehzahl u.a.) und sendet diese per TCP als Json-Objekt zum DCA. Der DCA liest zusätzlich per WiFi Daten von einer Wetterstation (WS980WiFi) und persitiert diese Daten in einer REDIS-Datenbank.</p> <p>Generierung einer Webseite mit Google-Gauges für 12 Anzeigen pro Quelle sowie auswählbarer Historie (dann als Graphen), die die Überwachung von n Energiequellen ermöglicht.</p> <p>Regelung der envertec-Spannungswandler per IoT-DCA</p>
Programmiersprache(n)	python 3
Betriebssystem(e)	Debian-Linux, Ubuntu-Linux
IDEs	PyCharme, Atom
Tools	Git, gitk, git-gui, confluence, draw-io
Sonstiges	REDIS, google-developer-tools

CI/CD (Continuous Integration / Continuous Deployment)

Zeitraum	11/2019 - 02/2020
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM Frankfurt
Tätigkeit	Erstellung von Playbooks und Resourcefiles zum Deployen verschiedener Komponenten einer Internet-Access-Plattform.
Methoden	scrum
Programmiersprache(n)	Ansible, python 3
Betriebssystem(e)	Suse-Linux, CentOs, Azure-Linux, Ubuntu
IDEs	Eclipse, PyCharme, Atom
Tools	Gitlab, ansible, ansible-Tower, jfrog Artifactory, checkmk, Kubernetes, curl Slack, The Box, git, gitk, gitlab jira, gerrit, confluence, draw-io
Sonstiges	webex, slack

Neuentwicklung eines Accounting-Transfer-Service

Zeitraum	07/2019 - 10/2019
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Entwicklung eines Verteil-Systems für Accounting-Pakete, die per UDP oder via REDIS-stream angeliefert werden und regelgesteuert in verschiedenen Files/Directories landen oder per UDP/TCP an bestimmte Partner versendet werden müssen. Redis-Client- und -Server in C, epoll für asynchrone Kommunikation in C, IPC über hostlokale Prozessgrenzen via Shared-memory und REDIS.
Methoden	UML
Programmiersprache(n)	C, python
Betriebssystem(e)	Suse-Linux, ubuntu
Tools	umbrella, plantuml, redis-5.0.2
Sonstiges	

Systemtest eines auf cf-engine basierenden Deploy-Tools auf einer VM-Farm

Zeitraum	05/2019 - 07/2019
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Deployment verschiedener, im NGN CC entwickelter Anwendungen auf einer VM-Farm. Erzeugung von Leitfäden zur Installation mit confluence/draw-io, Defect-Tracking mit cq-web. Entwicklung von python-Scripten zur Automatisierung wiederkehrender Arbeiten
Methoden	
Programmiersprache(n)	Python 2.7
Betriebssystem(e)	Suse-Linux
Tools	confluence, cq-web
Sonstiges	cf-engine

Erweiterung eines Accounting-Gateways um tagesspezifische Statistiken in SHM und Datenbank

Zeitraum	10/2018 - 04/2019
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Erweiterung eines bestehenden Accounting-Gateways um tagesspezifische Statistiken mit einer Laufzeit von 1 Monat. Ablage in einem Shared-memory, bei Tageswechsel Persistierung in eine relationale Datenbank. Retrieval-Möglichkeiten per externer Programme wahlweise aus beiden Quellen. Dokumentation mit zahlreichen Grafiken (drawIO) in confluence. Testsuite in python zum beschicken der Anwendung mit qualitativen Einzel-Records und Masendaten.
Methoden	UML
Programmiersprache(n)	C++ Version 11, python 2.7
Betriebssystem(e)	Suse-Linux
Tools	umbrella, plantuml, confluence, cq-web
Sonstiges	Scrum

Entwurf und Realisierung eines generischen SNMP-Abfrage- und Aufbereitungstools

Zeitraum	06/2018 - 09/2018
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	<p>Tool, das die MIB-Bäume eines servers per snmpwalk ausliest und eine Abstraktionsebene auf Tabellen, Zeilen, Spalten und Skalare bietet.</p> <p>Der Anwender kann die API des Tools in eigenen templates mit literalem Text mischen und so Programmausgaben nach eigenem gusto kreieren.</p> <p>Unterstützung von Mengenoperationen, Filtern, lambda-Funktionen und arithmetischen Tabellen-Operationen.</p> <p>Programmdoku mittels pydoc, Benutzerdoku in confluence mit draw-io-Grafiken.</p>
Methoden	UML
Programmiersprache(n)	Python 2.7
Betriebssystem(e)	Suse-Linux
Tools	umbrella, plantuml, confluence, cq-web
Sonstiges	Scrum

Erweiterung eines CallLimitingServers um Call-Attempt Limit

Zeitraum	03/2018 - 06/2018
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Erweiterung eines bestehenden Call-Limiting-Servers um Callraten-Begrenzung. Damit kann die Zahl der Rufaufbauten pro Zeiteinheit abhängig vom gewählten Paket des Kunden limitiert werden. Design-Dokumentation gem UML-V2, Interface-Entwurf mittels AsciiDoc dokumentiert und beschrieben, Code-Umsetzung gem. C++ V14. Erstellung von Testscripts zum Testen des Verhaltens sowie die Überwachung von Zuverlässigkeit und Speicherfreigabe bei massiv parallelen Tests.
Methoden	UML
Programmiersprache(n)	C++
Betriebssystem(e)	Suse-Linux, ubuntu
Tools	umbrella, plantuml
Sonstiges	

Accounting-Distributor für Telekom-Provider, Parallelprojekt bei gleichem Kunde

Zeitraum	03/2018 - 09/2018
Branche	Telekommunikation
Rolle	Entwickler (LeadDeveloper)
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Erweiterung eines Accounting-Distributing Systems um Multicasts. Per Konfiguration sowie vom Inhalt der Radius-Pakete gesteuert, leitet diese Lösung accounting-records einer Telefonie-Plattform an verschiedene Backend-Systeme über verschiedene Kommunikationsmechanismen. Diese Lösung soll über einen asynchronen Multicast-Sender erweitert werden, heisst: pro Paket und Routing-Ergebnis kann es mehrere Empfänger geben, die über UDP angesprochen werden und deren responses asynchron (select, epoll) verarbeitet werden.
Methoden	UML
Programmiersprache(n)	C, C++, python, perl,
Betriebssystem(e)	Suse-Linux, ubuntu, omvs
Tools	umbrella, plantuml, mq-series, db2
Sonstiges	

Access-Plattform für Telekom-Provider

Zeitraum	04/2017 - 03/2018
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	<p>Refaktorisierung einer Online-Provisionierungs-Lösung für eine Access-Plattform mit IP- und Bandbreitenvergabe.</p> <p>Überarbeitung und Modularisierung der bestehenden Lösung, basierend auf IBM-DB2 sowie IBM-MQ-Series in C++, Version 11 auf diversen UNIX-Systemen sowie auf Open-MVS.</p> <p>Etablierung von Micro-Services zum kundenspezifischen tailorn der Anwendung.</p> <p>In einem weiteren Schritt Unterstützung von XML/SPML zur Ansteuerung eines LDAP-Datenbank (Nokia C-NTDB).</p> <p>Versionsverwaltung mit GIT.</p>
Methoden	UML
Programmiersprache(n)	C++,
Betriebssystem(e)	Suse-Linux, ubuntu, omvs
Tools	umbrella, gerrit, jenkins, mq-series, db2
Sonstiges	scrum

Geschäftskunden-Telefonie-Plattform für SIP-Trunks und PBXe

Zeitraum	10/2016 - 03/2017
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Verteiltes Build-, Distributions- und Testsystem python tool für die Etablierung eines verteilten deploy- und Test-System auf linux-Rechnern inkl. verschiedenen, virtuellen Systemen (docker, lxc, virtual-box, vm-ware). Integration von scons, git, gerrit und jenkins in den workflow. Konfiguration der load-sets per json files. Fortschrittskontrolle mittels generierter Webseiten, flask und ajax. Komponententest mit selbstentwickelter CORBA-Umgebung, um Partnersysteme auf fremden Hosts remote aus der Testkontrolle anzusprechen.
	Re-engineering eines account-spoolers Überarbeitung eines bestehenden account-spoolers, der RADIUS-Paekete verschiedener Clienten in einem spool-Verzeichnisbaum erwartet, per UDP an eine Accounting-Instanz sendet und quittierte Pakete aus der spoolarea entfernt. Integration eines snmp-Interfaces zur Remote-Überwachung. Anpassung an C++-Version 11
Methoden	UML
Programmiersprache(n)	C++, python
Betriebssystem(e)	Suse-Linux, ubuntu
Tools	umbrella, gerrit, jenkins, confluence/draw-io, redmine
Sonstiges	

IOT (Internet of things)

Zeitraum	10/2015 – 09/2016
Branche	Umwelt-Organisation
Rolle	Entwickler
Kunde / Auftraggeber	Kontip GmbH
Tätigkeit	Erweiterung einer IOT-Bridge um Handling von SIP Einbau eines SIP-Stacks in die Bridge, Transformierung von SIP in IOTF-Json und umgekehrt. Entwicklung eines ApplicationServer-Proxies, der Requests an die IOTF ausleitet und das erhaltene Ergebnis weiter prozediert sowie STATUS-Messages der B-Seite ebenfalls an die IOTF ausleitet und sie dem Original-Sender zustellt. Damit können Sensoren bzw. Aktoren, die über Sip-Gateways angebunden werden (z.B. einer AVM-Frtzbox) supported werden. Entwurf und Implementierung einer Bridge zwischen einem LORA-Network-Server und der IBM-IOTF (BlueMix) Software-Design im UML, Entwicklung in C++ unter Linux (ubuntu, SuSe) eines Komponenten-basierten Servers, der als bridge zwischen einem Network-Server – der JSON über Websockets kommuniziert – mit der IBM-IOT-Foundation (BlueMix), die die Protokolle "REST" (für die Provisionierung) und "MQTT" (für den Datenaustausch) spricht. Persistierung der Proxy-Devices in einem Shared-Memory. Server ausgelegt für mehrere 100000 Devices (Sensoren und Aktoren) und mehrere Mandanten in Richtung BlueMix. Design und Entwicklung eines Application Servers für IOT (Internet of things) Software-Design im UML, Entwicklung in C++ unter Linux (ubuntu, SuSe, arch-linux) eines Application-Servers für das "Internet der Dinge" (Industrie 4.0), an den bis zu 100.000 Sensoren und Aktoren per gateways (fritzbox, raspberry, LORA-Funknetz) angeschlossen sind, die für beliebig viele Abnehmer zu Monitoring-Zewcken asynchron abfragbar sind (SUBSCRIBE-Mechanismus) oder imperativ per SIP-MESSAGE gesteuert werden können. Kommunikation mit den Sesonoren/Aktoren per JSON, Datenlieferung zu Monitoring-Zwecken als XML-Body in SIP-NOTIFYs.
Methoden	UML
Programmiersprache(n)	C++
Betriebssystem(e)	Suse-Linux, ubuntu
Tools	umbrella, gerrit, jenkins, confluence/draw-io
Sonstiges	

Geschäftskunden-Telefonie-Plattform für SIP-Trunks und PBXe

Zeitraum 08/2014 - 09/2015
Branche **Telekommunikation**
Rolle **Entwickler**
Kunde / Auftraggeber **IBM NGN CC f. TSI GmbH**
Tätigkeit **Design und Entwicklung eines Status-Monitors für PBXe an einem SIP-Trunk**

Software-Design im UML, Entwicklung in C++ unter Linux. Application-Server, der DIAMETER-Requests vom Type LIR entgegennimmt und bei einem Registry-Server den Connection- und Registrierungs-Status angeschlossener PBXe ermittelt und daraus die DIAMETER-Antwort vom Type LIA erzeugt. Multithreaded-Lösung.

Application Function für das feature Seiteneinstiegsschutz (CUG)

Interne Application-Funktion zur Realisierung der "Closed User Group"-Funktionalität in einem SIP-Telefonie-System. Parsen von Multi-Bodies im XML-Format, Bewertung des Inhaltes gegenüber konfigurierten Einträgen. Darauf basierend Ablehnung oder Akzeptanz von Anrufen.

Einbau DS-Field-Support (QoS) in diverse Kommunikations-Libraries

DSCP/TOS als Einmalaktion auf Netzwerk-Sockets setzen oder als sog. Ancillary-Data für Pro-Paket-Qos, sowohl für IPv4 als auch IPv6. Für ancillary-Data Abfragemöglichkeit auf Empfängerseite (IPV6_RECVTCLASS, IP_RECVTOS) durch Parsen des cmsghdr. Erweiterung bestehender Libraries für die Kommunikation via TCP, UDP, Radius, Diameter, http, MGCP

Neu-Entwicklung eines Nummern-Portierungs-Server für bis zu 200 Mio. Telefonnummern

Neu-Entwicklung eines bislang auf SQL/DB2 basierenden Portierungs-Servers. Design und Entwicklung von Softwarekomponenten für die Ablage / Abfrage der Portierungsdaten (i.W. Provider-Kennungen) für bis zu 200 Millionen Einträge. Aufgrund harter Performance-Anforderungen Wahl einer Big Data-Lösung mit non-SQL-InMemory-Storage und Dezimal-Bäumen. Kaskadierte Server-Programme, die für die Aufrufer als cloud wirken. Kommunikationsprotokolle basierend auf Google-protobuf Funktions- und Massentest (C++, teilw. Mit boost-library)

Methoden UML
Programmiersprache(n) C++
Betriebssystem(e) Linux
Protokolle Diameter, SIP, RTP, SDP, protobuf, TCP/UDP
Tools Wireshark, SIPp, Voiper, Confluence, phabricator, ClearQuest,

umbrella, gerrir, jenkins

Sonstiges

Entwurf, Realisierung und Weiterentwicklung einer MRF (Media-Ressource-Funktion)

Zeitraum 10/2009 – 03/2011 und 03/2012 – 07/2014

Branche **Telekommunikation**

Rolle **Entwickler**

Kunde / Auftraggeber **IBM NGN CC / terravoice.eu**

Tätigkeit

Design und Realisierung einer IMS-Media-Ressource-Funktion

IMS-konforme MRF zum Streamen von Audio-/Video-Assets, Erkennen von DTMF-Events, Auswertung von RTCP-Resports und Proxiiien von Sprach-Rtp-Daten an ein Spracherkennungssystem (IVR).

Neben dem Streamen an einzelne Endgeräte können Streams auch synchronisiert an n-Endgeräte (z.B. Videowand) gesendet werden.

Zu streamende Assets können komplett oder bei Bedarf partiell eingelesen werden

und werden von einem Hilfstool in alle Instanzen verteilt.

Multiplexen von Live-Streams durch Ringbuffer als Asset-Quelle.

Aufteilung gemäß IMS-Modell in Controller und Prozessor-

Schicht, deren Instanzen in n:m-Beziehungen auftreten können.

Auftragsannahme per SIP-INVITE oder per library-Funktion.

Entwurf und Realisierung als Multithreading Lösung.

Rework einer bestehenden MRF

mit deutlich erhöhten Anforderungen

- IPv4 und IPv6 Streams aus einer Anwendung

- Dynamische Speicherverwaltung für lokal vorgehaltene Assets

- Proxiiien von Unicast-Live-Streams

- IGMP-Multicast-joins um T-Home-Entertain-Medien proxiiien zu können

- Schreiben von RADIUS-Accounting Paketen

- Verarbeitung von Pinhole-Requests, um Firewall-Paarungen (lokale IP/Port zu Remote IP/Port) zu öffnen

- Html-Testseite mit webtrc, ajax, javascript u. Java-applet zum Absetzen eines Pinhole-Reuests

- DSCP (Differentiated services code point) Settings per Konfiguration für gewünschte Endpunkte, um QoS-Anforderungen zu erfüllen.

Design und Entwicklung eines dynamischen Memory-Managements

Für einen bestehenden Media-Resource-Server (MRS) war ein dynamisches Memory-Management gefordert, dass den

partiellen Zugriff auf assets bei Multi-User-Betrieb bei minimalem Ressourcen-Verbrauch ermöglicht. Slice-Orientierte read-aheads, Avisierung demnächst erforderlicher Speicherbereiche durch posix_fadvise, um die Daten asynchron in den Platten-Cache lesen zu lassen.

Methoden	UML
Programmiersprache(n)	C++, python
Betriebssystem(e)	AIX, Linux
Tools	scons, OpenOffice, asciidoc, doxygen, ClearQuest/Case, gstreamer, vlc
Protokolle	SIP, RTP, SDP, TCP/UDP
Sonstiges	

Aufbau eines Onlineshops (CMS) für einen Fahrrad-Händler

Zeitraum	08/2013 – 09/2013, Parallel-Projekt
Branche	Einzelhandel
Rolle	Designer und Entwickler
Kunde / Auftraggeber	Radhaus Bürgstadt
Tätigkeit	Untersuchung verschiedener CMS auf ihre Eignung hin. Nach Auswahl des Produkts <i>Shoppingcart</i> (opensolution.org) Übersetzungen, Anpassung und Versionierung der angepassten Software. Zur Anpassung gehörten Grafik-Themes, Zahlungswege, Admin-Tool, Web-CI. Erstellung von Diashows für den youtube-Kanal des Anbieters.
Methoden	
Programmiersprache(n)	php
Betriebssystem(e)	Ubuntu-Linux, MS-Windows-7
Tools	shoppingcart, gimp, Typo-3
Sonstiges	Google+, youtube und facebook-Auftritt für den Händler

Internet-Access-Plattform

Zeitraum 01/2013 - 09/2013
Branche **Telekommunikation**
Rolle **Entwickler**
Kunde / Auftraggeber **IBM NGN CC / IBM Deutschland GmbH**
Tätigkeit **Untersuchung geschachtelte Virtualisierung und Plattform-**

Aufbau

Evaluierung von vmware-Hypervisors (ESXi-5.0 und 5.1) bezüglich der Schachtelungsmöglichkeit mit dem Ziel, eine komplexe Plattform (n-Standorte mit m-Hosts) auf einem einzigen physikalischen Rechner simulieren zu können. Nach erfolgreicher Untersuchung Aufbau eines Virtualisierungs-Baums mit Strukturverläufen wie ESXi -> ESXi -> VM mit SuSe sles11, Konzeption des deployments, Testinstallationen und Testbetrieb.

Erweiterung einer Anwendungsüberwachung für eine verteilte Internet-Access-Plattform

Entwurf und Entwicklung eine Überwachung für neue Komponenten und Hosts mit zentralen und dezentralen Applikationen. Zentrale Komponenten unter IBM-z/OS in C und Rexx, die die dezentralen Kommunikationspartner anfragen, das Ergebnis parsen und damit TIVOLI-Netview-Attribute setzen, um den Zustand der gesamten Plattform grafisch darzustellen. Dezentrale Applikationen in perl unter z/OS, zOS-UNIX, zLinux, AIX und SuSe-Linux, die entweder 'passiv' per snmpget auf MIBs zugreifen oder 'aktiv' Requests absetzen und damit Verfügbarkeit, Laufzeit und Fehlerraten ermitteln, die schliesslich in einer Änderung der TIVOLI-NETVIEW-Anzeige münden. View- und Gruppen-Definition in der grafischen TIVOLI-Überwachung.

Methoden
Programmiersprache(n) C, Rexx, perl
Betriebssystem(e) vmware ESXi, SuSe-sles11, AIX, z/OS-UNIX,
Tools ClearCase, git, IBM IMM
Sonstiges

Entwicklung einer PBX-Provider-Lösung

Zeitraum	09/2006 – 12/2012
Branche	Telekommunikation
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Erweiterung eines SIP-Stacks für Geodaten-basierten Notruf Einführung neuer SIP-Header wie z.B. Geolocation, Geolocation-Routing und User-to-User in einem IMS-orientierten, SIP-basierten VoIP-System gemäß http://tools.ietf.org/html/rfc6442 und http://tools.ietf.org/html/draft-ietf-cuss-sip-uui-reqs-09 . Auflösung der gewählten Notrufnummer gemäss dem Aufenthaltsort des Anrufers.

Testsuite zum Bewerten von RTP-Verbindungen über mehrere Media-Gateways

Toolset, das die Verfügbarkeit der RTP-Verbindung über mind. 2 Media-Gateways (D/A – und A/D-Wandlung) prüft und dabei Messungen der Laufzeiten, Latenzen sowie der Jitter-Buffer-Füllung vornimmt.

Rewrite eines LDAP-Clients mit verbesserter Autorisierung

Neu-Implementierung eines LDAP-Clients mit Autorisierungsmodul. Test gegen einen openldap-Server mit privatem und öffentlichem Telefonbuch

Erstellung einer Funktionssammlung zum Reduzieren bzw. Expandieren von SIP-History-Info zum Sicherstellen des Topology-Hiding

Reduzierung der History-Info auf Weiterleitungsspezifische Informationen des Zielteilnehmers mit Ausschluss von SimRing und forking-Informationen.
Konvertierung (SIP-) Diversion in (SIP-) History-Info.
Einarbeitung der weiter geschriebenen History-Info bei Responses in die ursprüngliche History-Info.

Design und Implementierung eines SIP-SUBSCRIBE-Servers in python zur zyklischen Auslieferung von MIB-Countern via SIP-NOTIFYS

Subscribe-Server in python mit den Paketen Zope, Twisted und SipSimple. Auslesen der MIB-Counter mittels snmpbulkwalk. Definition zweier Event-Packages zur Umsetzung von 'MIB over SIP' in Notifies.

Erweiterung eines bestehenden AS um Gruppen-Dialog-Subscriptions

Kunde: IBM NGN CC

Branche: Telekommunikation

Ein Application-Server, der DIALOG-Subscriptions nach RFC-

4235 realisiert hat,
 wurde zum Einsatz in einer PBX um Subscriptions auf eine Gruppe von Principals erweitert. Konfiguration der Gruppenzugehörigkeit durch eine neue DB-Tabellenstruktur. Ignorierung der Principal-Profiles bei Gruppensubscriptions und Definition und Auswertung von Gruppen-Profiles.

Testsystem mit bereitgestellten Tools (SIPp) sowie eigenen Python-Lösungen für den Integrationstest. (Partieller Unittest in C++)

Methoden	UML
Programmiersprache(n)	C++, perl, Python
Betriebssystem(e)	SuSe-Linux, Ubuntu, AIX
Tools	Wireshark, SIPp, Voiper,
Sonstiges	

Entwicklung Grafik-Bearbeitung für Internet-Domain <http://www.wheesl.com>

Zeitraum	12/2010 – 12/2010 (nebenläufiges Projekt)
Branche	Internet-Präsenz
Rolle	Entwickler
Kunde / Auftraggeber	www.wheesl.com
Tätigkeit	Entwicklung eines Programms, das Standbilder (jpg) in unterschiedliche Formate konvertiert, mit (sichtbaren) Wasserzeichen versieht und (nicht sichtbare) steganografische Authentifizierungskennungen einbringt.
Methoden	
Programmiersprache(n)	Python, C
Betriebssystem(e)	Linux, MS-Windows
Tools	libjpeg
Sonstiges	

Entwicklung einer Anwendung zum Lesen von mifare-class-RFIDs

Zeitraum	06/2010
Branche	Internet-Präsenz
Rolle	Entwickler
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	Erstellung eines Programms zum Lesen von mifare-classic-RFID-Kartendaten mit einem Pegoda-Reader. Das Programm nutzt die librfid zum Zugriff auf das usb-device. Bei Änderung des Lesestatus (keine RFID-Karte bzw. 1 oder n Karten) erfolgt die Aktualisierung einer html-Statusseite sowie das Absenden eines SOAP-Requests an einen WebApplicationServer. Reader zum embedded-Betrieb auf einer FritzBox mit MIPS-crosscompiler entwickelt.
Methoden	
Programmiersprache(n)	C++
Betriebssystem(e)	
Tools	Git, make
Sonstiges	Linux-desktop, fritzbox mit MIPS-CPU

IMS Application Server für eine Telefonie-Plattform

Zeitraum	08/2006 – 05/2010
Branche	Telekommunikation
Rolle	AD (Architect of development) mit 12 Entwicklern
Kunde / Auftraggeber	IBM NGN CC
Tätigkeit	<p>Design und Realisierung eines verteilten IMS-Application-Server für CallCompletion (Rückruf) im VoIP-Umfeld mit non-VoIP-Endgeräten</p> <p>IMS-konformer AS für netzzentrisches Feature CallCompletion (CCBS, CCNR, CCNL) im Umfeld nicht SIP-fähiger Engeräte gemäss http://tools.ietf.org/html/draft-ietf-bliss-call-completion-04 mit Aufteilung der Serverfarm in Agent- und Monitor-Seite. Nutzung weiterer Applicationserver zur tracken von Teilnehmerzuständen mittels Dialog-Subscriptions gem. RFC-4235 und Register-Subscriptions (gem. RFC-3680). Eigene Entwicklungsarbeit: Client- und Serverseite zum Subscribieren auf den event "call-completion" in einem hoch performanten mulithreaded Umfeld</p> <p>Design und Realisierung eines verteilten IMS-Application-Server</p> <p>IMS-konformer AS für teilnehmerbezogene Features bei flexibler Service-Orchestrierung. Dialog-Status-Tracking gem RFC-4235. Erste Features: Dialog-Subscriptions für Rückrufszenarien, Schreiben von Call-Listen zum Ermöglichen des Features 'Spit-Blocking'. Hohe Skalierbarkeit mit dynamischer Lastverteilung. Aktuelle Design-Patterns: ComponentRepository und ComponentMediator.</p> <p>ETSI-Application Server für Dialog-Subscriptions nach RFC-4235</p> <p>Design und Implementierung eines Application Server für Dialog-Subscriptions zur Abbildung von Funktionen wie z.B. Call-Completion (Rückruf), Group-Calls/Call-Pickup sowie zu Info-Notifies für ChefSec-Funktionen. HTML-Doku mittels asccidoc, das wie der Sourcecode im git-Repository versioniert wird.</p> <p>Testsystem mit SIPp-Skripten und Python-Eigenentwicklung für den Performance- und Regression-Test</p>
Methoden	UML
Programmiersprache(n)	C++, Python
Betriebssystem(e)	AIX, Suse-Sles11, ubuntu-Linux

Tools	Wireshark, SIPp, scon, OpenOffice, asciidoc, doxygen, ClearQuest/Case
Sonstiges	DB2v9

Vertriebsplattform für eine Internet-Präsenz

Zeitraum	05/2004 - 09/2006
Branche	Touristik
Rolle	Entwickler
Kunde / Auftraggeber	Ticketcorner GmbH
Tätigkeit	<p>Entwurf und Realisierung der WebServices für einen Internet-Anbieter</p> <p>Planung der Webservice-Schnittstelle bezogen auf die vorhandenen Funktionen eines Internet-Distributionssystems. Aufbau der technischen Infrastruktur auf einem Solaris-Server (tomcat-cluster, axis).</p> <p>Realisierung der WebServices, teilweise als EJB-Client für einen ATG-Webcontainer. Kommunikationsprotokoll SOAP. Anhand der generierten WSDL Testtreiber in java und Python. Abbildung der Business-Rules durch mehrstufige Konfiguration. Erstellung Executive-Summary und ausführliche Schnittstellen-Doku für externe Kunden.</p>

Entwicklung eines Intranet-Tools zur Zahlungsautorisierung via Web

Design der Benutzeroberfläche als Teil eines DV-Konzepts. Standalone-Tool in python zur Kommunikation via http und https für die Funktionen Authorisierung, Commit und Retrieval. Definition der Datenbank-Tabellen für das Retrieval. Realisierung, Test und Dokumentation.

Entwicklung eines Gateways zur Übertragung von RFID-Kartendaten an ein Einlass-System

Entwurf der Schnittstelle, UML-Design dokumente, Realisierung eines asynchronen Gateways zur Übertragung der RFID-Kartendaten im Whitelist-Verfahren.

Entwicklung in einem Internet-Projekt, Teilbereiche Suche und Bezahlen

Weiterentwicklung eines Internet-Verkaufssystems um eine schnelle Suche mit wählbaren Sortierkriterien und fragmentierter Lieferung des Ergebnisses. Erstellung aller Komponenten von der JSP-Oberfläche bis zum DB-Zugriff. Hohe Performance wg. EJB-Freiheit der Lösung.

Erweiterung der Zahlfunktionen um Anforderungen für den schweizer Markt (i.w. Postdienst-bezogene Zahlarten).

Konzeption und Realisierung eines Doku-Generators

Entwurf einer Meta-Sprache, deren Elemente in Programmcode (beliebige Sprachen) eingelagert werden können und so die automatische Generierung von Programmdokumentation erlaubt.

Gegenüber javadoc erheblich erweiterter Funktionsumfang mit Einbinden von externem Prosatext sowie von Grafiken. Erzeugung der Ausgabe als PDF-Dokument.

Methoden

Programmiersprache(n) Python,

Betriebssystem(e) debian-Linux, Windows XP, Sun-Solaris 10

Tools dle, cvs, mySQL 5.0, Postgres 10, eclipse

Sonstiges

Entwicklung eines Videograbbers mit Motion-Detection, Parallelprojekt

Zeitraum 11/2003 - 04/2004

Branche **Sicherheitstechnik**

Rolle **Entwickler**

Kunde / Auftraggeber

Tätigkeit Realisierung eines Videograbbers (zyklische Standbilder zur Ablage im Filesystem) sowie softwaretechnische MotionDetection (Bewegungserkennung) für ein Security-Unternehmen. Videograber mit einer jpeg-Bibliothek. Bewegungserkennung durch Realisierung verschiedener Filter zur Kantenerkennung (u.a. Roberts-Cross-Operator, Prewitt-Funktion, Binomial-Laplace-Filter, Erosion u. Dilatation), die per Konfiguration zusammengestellt werden können und per Differenz zweier Bilder die Deltaerkennung ermöglichen. Umfangreiche Konfiguration mittels Web-Browser. Parallelprojekt, Abwicklung in den eigenen Räumen.

Methoden Bildmuster-Bearbeitung mit Matrizen

Programmiersprache(n) C (Kundenwunsch), python-cgi-Scripts zur Konfiguration

Betriebssystem(e) SuSe-Linux

Tools

Sonstiges

Anbindung Internet-Vertriebssystem an ein Legacy-Inventory-System

Zeitraum 04/2002 - 03/2004
Branche **Touristik**
Rolle **Entwickler**
Kunde / Auftraggeber **Qivive GmbH**
Tätigkeit **Portierung und Erweiterung eines Gateways zw. einem Inventory- und einem Distributionssystem**
Funktionale Erweiterung des bestehenden Gateways, technische Anpassung aufgrund höherer Performance-Anforderungen. Einbau eines schnellen Caches mit IPC- und LargeFile-Komponenten.

Design und Implementierung eines schnellen Caches unter Solaris 8

Entwurf eines Caches unter Nutzung von Ipc-Mechanismen (InterProzessKommunikation) mit geschlossenem Hashing als Verwaltungsfunktion. Programmiersprachenunabhängiger Netzwerkzugang via TCP. Realisierung des Caches in C++, Testprogramme in C++ und Python.
HTML-Prosa-Doku und Betriebshandbuch.

Methoden

Programmiersprache(n) C++, Python

Betriebssystem(e) Sun-Solaris 8

Tools

Sonstiges

Gateway-Programmierung

Zeitraum 09/2000 - 03/2002

Branche **Touristik**

Rolle **Entwickler**

Kunde / Auftraggeber **START Informatik**

Tätigkeit **Realisierung eines performanten Gateways zum Zugriff auf ein UNIX-Inventorysystem aus dem Internet**

Entwurf des Prozessmodells für einen parallelen Nachrichtenkonverter unter Verwendung bestehender Anwendungssoftware; programmiersprachen-unabhängige Schnittstelle zu den Frontends. Realisierung in C++ (inkl. STL)

Entwicklung eines Multiplexers zur Anbindung von > 8000 Windows-Endgeräten an ein UNIX-System

UML-konformes Design, Aufstellung und Bewerten von Mengengerüsten.

Entwurf eines Prozessmodells, das die geforderte Leistung mit möglichst wenigen Prozessen erbringt (< 50).

Realisierung in C++ (ohne Verwendung der STL),

Testtreiber-Erstellung, Visualisierung des Systemzustands mit div. Java-Applets

und -Servlets. Python-Tool zur Doku-Generierung.

Design und Entwicklung einer System-Bibliothek zur einfachen Erstellung von Netzwerkdiensten

Entwurf der Bibliothek mit den Paketen

- **Event** (für ereignisorientierte Prozessmodelle)
- **InterProzessKommunikation** (OO-Wrapper um die Betriebssystem-Mechanismen)
- **Netzwerk** (TCP, UDP und Stream-Sockets für C++)
- **Kommunikation** (Universelle Kommunikationsklassen)
- **Ablaufsteuerung** (Framework für Prozessrahmen)
- **Kontextdatenbank** (Netzwerk-Datenbank, mit obigen Paketen realisiert)

UML-konformes Design, Realisierung in C++ unter Linux sowie ReliantUnix.

Kontextdatenbank sowie div. Testprogramme zur Demonstration des Einsatzes.

HTML-Doku mit Together aus Sourcen generiert; HTML-Prosa-Doku des Designs

Methoden UML

Programmiersprache(n) C++, java, Python

Betriebssystem(e) Sinix V.4, ReliantUnix 5.4

Tools Together, Eclipse

Sonstiges

Erstellung der Web-Präsenz für ein Pferde-Gestüt

Zeitraum 09/1999 – 10/1999 (Neben-Projekt)

Branche **Gestüt**

Rolle **Entwickler**

Kunde / Auftraggeber **Jomm Ranches, Grosswallstadt**

Tätigkeit HTML-Seiten-Erstellung, Aufbereitung von div. Fotomaterial, Erstellung von RealVideo-Clips aus VHS-Bandmaterial. Abwicklung der Aktivitäten zur Einrichtung der Domain. Einrichtung der Infrastruktur zur eigenen Erstellung von Web-Seiten, Benutzer-Schulung.

Methoden

Programmiersprache(n)

Betriebssystem(e) MS-Windows

Tools PhotoImpact, RealVideoGen,

Sonstiges

Systemprogrammierung für einen Ticketing-Vertriebssystem-Entwickler

Zeitraum 10/1998 – 08/2000

Branche **Touristik**

Rolle **Entwickler**

Kunde / Auftraggeber **START-Ticket GmbH**

Tätigkeit **Realisierung einer Unix-Host-Kopplung für den Kartenvertrieb der EXPO-2000**
Entwurf eines Umsetzers von Host-Anfragen für ein UNIX-Ticketingsystem,
Kommunikation zum Host via TCP/IP; Prozessmodell mit einem Master und multiplen Co-Prozessen, die sich auf beliebige Rechner auslagern lassen (Kommunikation auch hier TCP/IP).
Erfolgreiche Erfüllung besonders hoher Performance-Anforderungen.

Library-Entwicklung

Design und Realisierung mehrerer Bibliotheken zur Nutzung von UNIX-Mechanismen in C++.
Erstellung von Wrapperklassen zum Einsatz von Record-Locking, Streams,
Sockets u.a. in einer POSIX-konformen Umgebung.
Entwicklung eines Mechanismus zur Ablauf-Analyse und Performance-Untersuchung in C- und C++-Legacy-Systemen.
Erstellung von HTML-Prosadokumentation.

Designerstellung für ein POS-Kopplungsprogramm

Design eines Gateways zur Ankopplung einer bestehenden POS-Lösung an ein Kartenverkaufssystem. Definition der Schnittstellen.
Erstellung div. UML Diagramme für die zu realisierende Lösung.

Realisierung verschiedener Kommunikations-Gateways

Design und Implementierung einer Framework-Klassenbibliothek zur Kommunikation unter UNIX mit einem BS-2000 Großrechner gem. RFC1006 (ISO over TCP).
Realisierung von verschiedenen, darauf basierenden Gateway-Programmen.
Design nach UML, Realisierung mit C++ (inkl. STL), Intranet-Programmdokumentation (generierte und manuell erstellte Dokumentation).
Leitung des Entwicklerteams (4 MA).

Methoden

Programmiersprache(n) C++

Betriebssystem(e)	Linux, Sinix, DG-UX
Tools	Paradigm-Plus, Visio
Sonstiges	

Systemintegration und Regressionstest in einem Telekommunikationsprojekt

Zeitraum	01/1998 – 09/1998
Branche	IT
Rolle	Regression-Tester
Kunde / Auftraggeber	IBM Deutschland GmbH
Tätigkeit	Definition und Erstellung automatisierter Regressionstests sowie Generierung von EDIFACT-Testdaten jeweils mit Standard-Shell-Mitteln. Problemverwaltung mit IBMs TeamConnection. Bewertung von Architektur-und Realisierungs-Mängeln von Fremdleistern.
Methoden	
Programmiersprache(n)	
Betriebssystem(e)	AIX 4.1, Windows-NT 4.0
Tools	LotusNotes 4.6, TeamConnection V2.0, MS-Office 95, Oracle 7
Sonstiges	

Entwicklung bei einem Ticketing-Vertriebssystem-Entwickler

Zeitraum	01/1995 – 12/1997
Branche	Touristik
Rolle	Entwickler, Teamleiter
Kunde / Auftraggeber	START-Ticket GmbH
Tätigkeit	<p>Aufbau Qualitätsmanagement für ein Entwicklungsprojekt Definition eines projektspezifischen Phasenmodells, Erstellung der Dokumentationsvorlagen für die verschiedenen Phasen (UML-basiert). Aufbau eines QS-Plans, Vorgaben für Testpläne und Testszenarien. Erstellung Programmier-Richtlinien und -Empfehlungen für C++ und ESQ/C. Konzeption eines PR- (ProblemReport) Tools. Koordination der Aktivitäten der QS-Gruppe (6 MA), Durchführung und Moderation von Workshops und Reviews.</p> <p>Entwurf, Design und Implementierung eines verteilten Message-Passing Systems (START-UNIX-Middleware & Framework) Aufbau eines verteilten OLTP-Systems für heterogene Systemlandschaften, Konzeption und Realisierung eines sicheren Nachrichtenverteilers unter Nutzung von UNIX-IPC und Rechnerkommunikation. Realisierung eines universellen APIs. Analyse & Design nach UML, Integration der Sourcen in SNIFF+ / rcs. Erstellung eines Toolsets zur Generierung von Quellcode-Dokumentation (Intranet und UNIX-manpages); HTML-Prosadokumentation</p> <p>Konzeption eines multimandantenfähigen Abrechnungssystem für ein Kartenverkaufs-System Analyse in Zusammenarbeit mit dem Produktmanagement, Entwurf eines Paradigmas zur Erstellung beliebiger Abrechnungsprogramme unter Nutzung von ESQ/C in einer C++-Umgebung. Realisierung einiger Abrechnungen bzw. Statistiken. Erstellung von HTML-Prosadokumentation.</p> <p>Integration von PointOfSales-Terminals in ein Ticketing-System DV-Design der angestrebten Lösung nach Coad/Yourdon; Ansteuerung der Kassenschnittstelle von POS-Terminals über die serielle Schnittstelle, Kommunikation mit dem Netzbetreiber und Routing zwischen Netzbetreiber, POS-Terminals und der Verkaufsanwendung.</p>

Methoden	Coad-Yourdan, Booch
Programmiersprache(n)	C++
Betriebssystem(e)	SINIX SVR4.2/4.3, SCO 3.2
Tools	Informix SQL
Sonstiges	

Erstellung einer branchenspezifischen Adressverwaltung für eine kommunale Tourismus-Organisation

Zeitraum	09/1994 – 12/1994
Branche	Touristik
Rolle	Designer / Entwickler
Kunde / Auftraggeber	TZH Hamburg
Tätigkeit	Realisierung einer branchenspezifischen Adressverwaltung unter MS-Access mit zentraler Datenhaltung auf einem UNIX-System und Kommunikation über ein PC-TCP/NFS-System; Beratung und Teststellung verschiedener PC-TCP/NFS-Produkte, Design und Realisierung des Benutzer-Frontends.
Methoden	
Programmiersprache(n)	MS-ACCESS-BASIC
Betriebssystem(e)	SINIX SVR4, MS-Windows
Tools	MS-ACCESS 2.0
Sonstiges	

Entwicklung eines Ticketing-Systems

Zeitraum	02/1992 – 01/1994
Branche	Touristik
Rolle	Designer / Entwickler
Kunde / Auftraggeber	START Ticket GmbH
Tätigkeit	<p>Konzeption Gebührenmodell für ein Ticketing-System Konzeption eines Gebührenmodells für Verkaufs- und Stornovorgängen in einem verteilten Ticketing-System unter UNIX. Objektorientierte Analyse und Design nach Coad & Yourdon, Klassen- und Schnittstellen-Definition sowie Implementierung.</p> <p>Shellscript-Programmierung und UNIX-pkg-Erstellung für ein bestehendes Software-System Erstellung von Shellscripts und div. pkg-Modellen zur Installation und Verteilung des Software-Systems.</p> <p>OLTP-System für verschiedene Anwendungen eines Reisemittlers Aufbau eines verteilten, objektorientierten OLTP-Systems; Konzeption und Realisierung eines ORB; Realisierung der Kommunikationssoftware Host <-> PC (TCP/IP-Sockets und RPCs).</p>
Methoden	
Programmiersprache(n)	C++
Betriebssystem(e)	SINIX SVR4, DG-UX SVR4, SCO-Unix
Tools	
Sonstiges	

Logistik-System für Transportwesen

Zeitraum	12/1991 – 01/1992
Branche	IT
Rolle	Entwickler
Kunde / Auftraggeber	SIEMENS AG
Tätigkeit	Design der Benutzerschnittstelle, Integrationstest
Methoden	
Programmiersprache(n)	Rosi-SQL, C
Betriebssystem(e)	SINIX V5
Tools	Informix-SE
Sonstiges	

Software-Entwicklung für eine Fluggesellschaft

Zeitraum	01/1991 – 11/1991
Branche	Dienstleistung
Rolle	Entwickler
Kunde / Auftraggeber	CONDOR
Tätigkeit	Ticketing-Clearingsystem für eine Fluggesellschaft Konzeption und Realisierung eines Abrechnungsprogramms zur Berechnung von Zahlungsverpflichtungen für externe Zubringerdienste.

Marketing-Informationssystem für eine Fluggesellschaft

Konzeption und Realisierung eines Kunden-Informationssystems mit integrierter Adressverwaltung und Mailing-System im UNIX-Rechnerverbund; Anwenderschulung, Benutzerdokumentation.

Methoden	
Programmiersprache(n)	C
Betriebssystem(e)	AIX, UNIX SVR3
Tools	Ddb4, Quantum-Catcher, UniScreen, DR-GEM, UniScreen
Sonstiges	

Software-Entwicklung für ein Telekommunikationsunternehmen

Zeitraum	05/1989 – 12/1990
Branche	Dienstleistung
Rolle	Entwickler
Kunde / Auftraggeber	Deutsche Telekom AG
Tätigkeit	Verkehrsgebühren-Nachverarbeitung für Telekom-Dienste Erstellung und Ausführung von Systemtest-Szenarien; Entwicklung von JCL, CLIST und Edit-Macros.
	Verwaltung Gerätetausch- und Wartung im Telekom-Bereich Realisierung, Integrationstest und Dokumentation
	Rechnungsdaten-Nachverarbeitung für DATEX-P-Dienst Realisierung, Integrationstest
Methoden	
Programmiersprache(n)	Delta-Cobol
Betriebssystem(e)	MVS/ESA, VM/SP
Tools	CMS, ISPF/PDF, SCRIPT/VS, DCF, CICS, VSAM
Sonstiges	

Verwaltungssystem für Bauvorhaben im Transportwesen

Zeitraum	12/1988 – 04/1989
Branche	Transport
Rolle	Entwickler
Kunde / Auftraggeber	Deutsche Bahn AG
Tätigkeit	Verwaltungssystem für Bauvorhaben im Transportwesen DV-Konzept, Realisierung, Integrationstest
Methoden	
Programmiersprache(n)	CLIPPER, dBase III+
Betriebssystem(e)	MS/PC-DOS
Tools	dBase III+, SPF/PC, Saywhat!?, R&R-Reportgenerator
Sonstiges	