

1 Persönliche Daten

Name Volker Schmidt

Geburtsdatum 30. November 1965

Nationalität Deutsch

Familienstand verheiratet

Kinder 1

Fremdsprachen Englisch: fließend

EDV-Erfahrung seit 1992

Position Administration, Support, Programmierung, Dokumentation,
DTP

Fachliche Schwerpunkte Perl, Unix, Administration

Programmiersprachen Perl, Shell (Bourne-, ...), C, T_EX, L^AT_EX, VRML,
Working Knowledge in Java, JavaScript, C++

Betriebssysteme Unix: Linux, Solaris (Sun Cluster), FreeBSD, NetBSD,
OS/9, IRIX, AIX, Tru64 (Compaq TruCluster), Anwenderkenntnisse
in Windows, Windows CE und MS-DOS

Datenbanken mSQL, mySQL: Perl DBD/DBI

Netzwerkkenntnisse, Datenkommunikation Ethernet, Fax, Internet,
Intranet, ISDN, SMTP, NNTP, HTTP, TCP/IP, UUCP, Windows
Netzwerk (Samba)

Hardware PC, Sun Sparc, Silicon-Graphics

Produkte, Standards, Erfahrungen Web Server Aufbau,
Administration: Internet, HTML, HTTP, TCP/IP, SMTP, News,
Web Entwicklung: CGI, Perl (Entwicklung im eCommerce Umfeld),
DBD/DBI, MySQL, XML, XSL, XSLT, Xalan, Samba, Sendmail,
Apache, YAML

Arbeitsgebiete, Branchen EDV-Dienstleistungen, Telekommunikation,
Multimedia, Internet Hosting

2 Erfahrungen

Oktober 2017 – Juni 2018 Internet-Dienstleister: Preisvergleich

Titel Ein Perl-Projekt wird durch eine Neu-Entwicklung abgelöst

Thema Software-Entwicklung

Merkmale –

- Analyse von Legacy Perl Code

Stichwörter –

- Perl
- Legacy Code

Situation Älterer historische gewachsener Perl-Code befindet sich in einem Zustand in dem die Weiterentwicklung aufwendig ist. Der Kunde ist dabei, den Code durch eine Neu-Entwicklung in Java zu ersetzen.

Dabei wird produktiver Perl-Code analysiert um Aussagen über dessen Verhalten zu gewinnen, die dann in die neue Entwicklung einfließen können.

Daneben soll der produktive Code um Features ergänzt werden, die auch in der Neu-Entwicklung implementiert werden.

Funktionalität Von Kunden gelieferte Daten werden analysiert, auf tolerante Weise geparkt, dann in einer Datenbank zur Verfügung gestellt.

Erfahrung Methoden zur Analyse von Code müssen gefunden werden. Grenzen der Machbarkeit werden entdeckt.

Veränderte Funktionalität Der bestehende Perl-Code erfährt Veränderungen, indem neue Merkmale implementiert werden. Die Entwicklung in Java an dem neuen Code wird durch die Analyse des bestehenden Codes unterstützt.

Verwendete Produkte –

- Perl,
- PPI

Zukunft Mit der Arbeit einerseits am bestehenden Code und andererseits bei der Entwicklung an neuem Code ist es möglich, den bestehenden Code zu ersetzen.

Januar 2017 Online-Händler in Hannover

Umfeld Betreiber von Shops im World-Wide Web

Thema Entwicklung in Perl

Verwendete Produkte –

- Perl.

2015 Eigenbedarf

Umfeld Archivierung, Multimedia-Daten, Konfiguration, Document Management

Thema Vereinnahmen und Archivieren von Daten, Konfiguration, Extraktion von Konfigurations-Daten

Zusammenfassung –

- Wir beabsichtigen, Veränderungen am Dateisystem zu entdecken um so Konfigurations-Daten im System zu finden.
- Wir finden redundante Dateien und Datei-Strukturen im Datei-System
- Wir finden Abweichungen von einer Datei-System-Hierarchie-Konvention oder
- stellen sicher, dass die Konvention beim Vereinnahmen von neuen Dateien eingehalten wird.
- Dateien werden zum schnellen Auffinden indiziert.

Situation –

Veränderungen an einem Datei-System Die Menge an Konfigurations-Dateien, die von einem System, zum Beispiel einer Server-Maschine, verwendet wird, weicht ab vom Zustand unmittelbar nach der Installation. Hier Veränderungen zu entdecken hilft dabei, den Konfigurations-Status des Systems festzustellen. Die Konfigurations-Änderung wird als eine Menge von Dateien, die Veränderungen erfahren haben, festgestellt. Also, zunächst ist eine Analyse auf der Ebene von Dateien, nicht von Datei-Inhalten, beabsichtigt.

Datei-Redundanzen im Datei-System Datei-Hierarchien die zum Beispiel auf mobilen Geräten benutzt werden, die keine Netzwerk-Verbindung haben, können mutieren und damit von einander abweichen. Redundanzen und Abweichungen sollen entdeckt werden.

Eine Datei-System-Hierarchie-Konvention Abweichungen von einer Datei-System-Hierarchie-Konvention sollen entdeckt und bereinigt werden. Zum Beispiel kann nach einem einfachen Herunterladen von Bild-Daten von einem Kamera-Medium geprüft werden, ob die Daten der Konvention entsprechen. Gegebenenfalls können die Daten an den richtigen Platz gebracht werden.

Die gefundenen Ansätze sollen zum Beispiel genutzt werden um Dateien von Kamera-Medien oder von Dokumenten-Scans zu vereinnahmen.

Benutzte Funktionalität Betriebssysteme –

- Linux,
- NetBSD,
- etwas Windows

Veränderte Funktionalität Perl Scripts helfen, Paare von Verzeichnissen zu entdecken, die Datei-Redundanzen enthalten.

Effekt Techniken zur Versionierung, zur Archivierung, zur Verwaltung von Dokumenten sind unverzichtbar. Das entstehende Script-Werk kann später als Sicherheits-Netz verwendet werden, etwa um sicherzustellen, dass nicht redundanterweise Kopien von Dateien aufbewahrt werden.

Verwendete Produkte –

- Perl,
- PostgreSQL

August 2014 – Dezember 2014 Eigenbedarf

Titel Inhalt wird von PDF-Dateien extrahiert

Thema Software-Entwicklung

Merkmale –

- Textverarbeitung
- Buchhaltung

Stichwörter –

- Datei-Konversion
- PDF
- Buchhaltung

Situation Eine Quelle von Daten für die Buchhaltung ist die manuelle Eingabe von Daten aus Dokumenten. Einige der Dokumente liegen primär in Form von Dateien im PDF-Format vor.

Die Maschine soll den Vorgang der Dateneingabe aus PDF-Dokumenten übernehmen. Zunächst sollen Daten aus einigen bestimmten Dokumenten, die im PDF-Format vorliegen, gewonnen werden.

Die Motivation besteht hauptsächlich in konstruktiver Faulheit. Wir hoffen auf diese Seiteneffekte:

- Mehr Daten können in die Buchhaltung einfließen
- Daten können zeitnäher verwendet werden, indem die manuelle Eingabe durch einen mehr oder weniger automatisierten Prozess ersetzt wird.
- Die Extraktion von Daten aus Dokumenten ist weniger fehleranfällig als die manuelle Eingabe.

Es muss ein Weg gefunden werden, Daten aus PDF-Dokumenten zu übernehmen, die von Anderen zum alleinigen Zweck der Präsentation in zweidimensionalen Medien vorbereitet wurden. – Die Aktivität wird als Teil des Weges zu einem angenehm automatisierten Buchhaltungs-Prozess gesehen.

Die generierte Funktionalität soll konfigurierbar sein, so dass sie mit wenig Aufwand an sich ändernde Dokumenten-Formate oder andere Dokumente angepaßt werden kann.

Funktionalität Die Buchhaltung wird mit Hilfe einer Menge von schnellgestrickten Scripts durchgeführt. Daten werden ausschließlich manuell mit Hilfe von einem Bildschirm, einer Tastatur und einem Bild des Quell-Dokumentes eingegeben.

Erfahrung Es wurde gezeigt, daß eine Lösung gefunden werden kann, die sicherstellt, dass keine relevanten Daten verloren gehen. Der Inhalt des Quell-Dokumentes wird in einer Form weitergereicht, wie er von folgenden Prozess-Schritten verarbeitet werden kann.

Veränderte Funktionalität Wir verwenden verschiedene Merkmale von pdfbox (<http://pdfbox.apache.org/>) um ein existierendes PDF-Dokument in Text zu wandeln. Ein Perl-Script extrahiert die relevanten Informationen von diesem Text. Die Extraktion wird durch eine Konfiguration gesteuert.

Ziele:

- Die Verlustlose Extraktion vom Inhalt einer PDF-Quell-Datei ist sichergestellt.
- Falls das Quell-Dokument unerwartet formatiert ist, werden entsprechende Warnungen ausgegeben. Es wird angemerkt, falls der Extraktions-Prozess in Bezug auf die Extraktion von relevanten Daten nicht komplett durchgeführt werden kann. Auf diese Weise wird vermieden, dass sich die unerwartete Formatierung des Quell-Dokuments oder auch Fehler in der Konfiguration unbemerkt auf die ausgegebenen Daten auswirken.
- Daten werden in einem benutzbaren Format weitergereicht, zum Beispiel formatiert als YAML.
- Die Anpassung der Konfiguration an eine geänderte Formatierung des Quell-Dokuments oder das Erstellen einer Konfiguration für ein neues Quell-Dokument-Format ist einfach.

Verwendete Produkte –

- Perl,
- pdfbox

Zukunft Die fast ausschließlich mit Mitteln von Perl generierte Funktionalität könnte in pdfbox integriert werden. Damit könnte die Benutzbarkeit und die Erweiterbarkeit erhöht werden.

Effekt Erfahrungen sollen zeigen, ob die Funktionalität zum Beispiel beim Lesen von Kontoauszügen oder Telefon-Rechnungen eingesetzt werden kann.

März 2014 – April 2014 Verlag in Düsseldorf

Umfeld Publishing, Cross-Media (Paper, Online)

Aufgaben Beratung, Wartung von Legacy Perl Code

Technischer Hintergrund Perl, Omnimark, XML, XSL

Eingesetzte Tools Perl

Fachlicher Hintergrund Perl, Software Entwicklung, Umgang mit Legacy Code

Eingesetzte Skills Perl, XML

Januar 2013 – Dezember 2013 Automotive-Hersteller in Esslingen

Umfeld Software-Entwicklung, CAD

Thema Einführung von SCM-Maßnahmen, Werkzeugen, Prozessen, Methoden für die Software-Entwicklung: Der Kunde hat die Absicht, in einem bestehenden Software-Entwicklungs-Team Agile Methoden einzuführen.

Analyse von Legacy-Perl-Code mit dem Ziel, Wartbarkeit und Stabilität zu verbessern.

Entwicklung in Perl Fachlicher Hintergrund: Entwicklung, Software Configuration Management Linux, Perl, Jazz RTC, Software, Development Cycle, Software Development, Agile, Perl, Jazz RTC, Entwicklung, Perl, Software Configuration Management, Agile Software Development

Merkmale –

- Entwicklung von einem Framework für Analyse und Refactoring von Perl-Code.
- Erarbeiten von Vorschlägen mit dem Ziel eine hohe Code-Qualität und Wartbarkeit von Perl-Code zu erreichen, i. e., die Qualität von Legacy-Code zu verbessern.
- Erarbeiten von Vorschlägen für einen Minimal-Standard für die Dokumentation innerhalb vom Quell-Code.
- Einführung eines Workflows für die Dokumentation.
- Einführung von Tools für die Dokumentation.

- Analyse von existierendem Programm-Code, besonders im Hinblick auf die Dokumentation und das Entdecken von Verbesserungs-Potential.
- Erarbeiten von Vorschlägen von Maßnahmen, die Qualität von vorhandenem Code zu verbessern.
- Erarbeiten eines Vorschlags für die Vorgehensweise zu der Verbesserung der Qualität und der Wartbarkeit von vorhandenem Code, also eines Prozesses für die Analyse und Veränderung von Programm-Code.

Situation Der Kunde unterhält eine Eigen-Entwicklung, die benutzt wird um Konstrukteuren CAD-Daten zur Verfügung zu stellen. Das geschieht in einem Prozess, der dem Software-Build-Prozess ähnelt.

- Bei der Entwicklung wurde nicht einheitlich einem Coding Standard gefolgt.
- Die Entwicklung erfolgte verteilt auf mehrere Standorte, wobei unterschiedliche Standorte an jeweils unterschiedlichen Teilen der Software arbeiteten und unterschiedliche Programmier-Stile hatten.
- Entwicklung folgte nicht einem definierten Prozess.
- Wir sahen das Ziel darin, Prozesse so zu definieren und Code so zu verändern, dass ein gelegentlicher Entwickler in der Lage wäre, Veränderungen am Code durchzuführen, e. g. bei der Lösung eines Problems als auch, reguläre Entwickler bei ihrer Arbeit zu unterstützen.

Benutzte Funktionalität Perl-Scripts, die mit Job-Control in einer Cluster-Umgebung umgehen.

- Perl-Scripts, die mit CAD-Daten umgehen.

Erfahrung Analyse von Perl-Code mit bloßen Augen und mit Hilfe von Funktionalität, die für diesen Zweck entwickelt wurde.

- Kontakt mit Team-Mitgliedern in einem interkontinental verteilten Team mit dem Ziel, ihre Arbeitsgewohnheiten kennenzulernen.
- Entwicklung in Perl.

Wirkung Der Kunde bekam von uns

- Perl-Funktionalität, die bei Analyse und Refactoring von Perl-Code helfen kann.
- Vorschläge, wie Misfeatures in existierendem Code entdeckt und ausgemerzt werden können, um so eine vernünftige Wartbarkeit zu erreichen.
- Vorschläge, wie der Prozess, Perl-Code zu verändern um Misfeatures zu beseitigen teilweise automatisiert werden kann.

Veränderte Funktionalität –

- Wir haben existierenden Perl-Code mit den gewöhnlichen Mitteln der Text-Verarbeitung (also awk, sed, . . . , Perl) analysiert um Merkmale des Codes zu visualisieren („Software Visualization“).
- Wir haben auf der Basis von PPI ein Framework entwickelt, das für die Analyse und die Veränderung von Perl Code verwendet werden kann, e. g., indem der Code auf mehrere Dateien aufgeteilt wird, während Symbole so verändert werden, daß Referenzen erhalten bleiben.
- Wir haben Werkzeuge und Konventionen für die Dokumentation im Code eingeführt.
- Wir haben Coding Conventions vorgeschlagen.
- Wir haben bei der Einführung von Jazz RTC assistiert.
- Wir haben beim Aufsetzen von Cacti assistiert.

Netto Effekt Es war eine interessante Erfahrung mit der Grammatik von Perl und mit PPI bei der Analyse von Perl-Code.

Verwendete Produkte –

- Programmiersprache: Perl.
- Libraries: PPI.
- SCM-Werkzeuge: Doxygen, Jazz RTC.
- Werkzeuge zur Administration: Cacti.

Oktober 2012 – Dezember 2012 Einzelhandel in Neckarsulm

Umfeld Perl Entwicklung, Administration

Aufgaben Beobachten, Verwalten, Visualisieren von Prozessen. Das Ziel war, bestehende Prozesse pflegeleichter zu machen.

Eingesetzte Tools Perl

Fachlicher Hintergrund Unix, Perl

Eingesetzte Skills Perl Entwicklung

Dezember 2011 – März 2012 Halbleiter-Hersteller in München

Umfeld Software-Entwicklung im Bereich Halbleiter

Aufgaben Software-Entwicklung, Support für eine Toolchain, die von 2000 Entwicklern benutzt wird.

Technischer Hintergrund Linux, Perl, Clearcase, Software Development Cycle, Problem Management, Continuous Integration, Test

Eingesetzte Tools Perl, Clearcase

Fachlicher Hintergrund Entwicklung, Software Configuration Management

Eingesetzte Skills Entwicklung, Perl, Software Configuration Management

Juli 2011 – September 2011 Telekommunikations-Unternehmen in Bonn

Umfeld Software-Test

Kunde Telekommunikations-Anbieter in Bonn

Thema Software Test Automatisierung

Konzeptionelle Arbeit auf dem Weg zur Nutzung eines Frameworks für die Automatisierung von Software-Tests. Ziel ist die Entwicklung dieses Frameworks.

Zusammenfassung Die Analyse der bestehenden Situation führte zu Vorschlägen, die die Struktur des zu entwickelnden Frameworks und den Arbeiten auf dem Weg dorthin betreffen.

Merkmale –

- Analyse der bestehenden Arbeitsweise

- Vorschlag eines Modells zur Vorgehensweise bei der Entwicklung von Software

Situation Der Software-Test wurde manuell oder mit Hilfe von kleinen von einzelnen Kollegen für den eigenen Arbeitsplatz erstellten Scripts durchgeführt.

Der Kunde zeigt Interess daran, existierende Software-Test-Prozesse zu automatisieren und dafür ein System zu entwickeln. Es wird die Möglichkeit gesehen, das entstehende System auch an anderen Stellen im Haus einzusetzen.

Benutzte Funktionalität:

- Verschiedene Scripts,
- Bleistift und Papier :-).

Wirkung Von der bestehenden Situation ausgehend leiten wir Merkmale ab, die ein neues System haben sollte. Dabei nehmen wir an, dass der Kunde neue Funktionalität früh übernehmen wird. Flexibilität und Kontrolle sollen erhalten bleiben. Zum Beispiel soll das System Möglichkeiten bieten, einen Test-Vorgang zu beobachten.

Wir leiten Struktur-Merkmale für die Funktionalität des neuen Systems ab.

Wir definieren eine Menge von Aufgaben und deren Beziehungen zu einander ab, um den Entwicklungs-Prozess zu beschreiben.

Netto Effekt Kommunikation im Team ist wichtig: Das gesprochene Wort, Papier und die Arbeit an Programm-Code.

Juni 2006 – Mai 2010 Telekommunikations-Unternehmen in Bonn

Umfeld Software-Entwicklung bei einem Telekommunikationsanbieter: Im Umfeld des Arbeitsplatzes wurde in Teams von insgesamt etwa 12 Entwicklern für die Abrechnung von Gesprächsdaten in den Sprachen C++ und Java entwickelt.

Aufgaben Software Configuration Management, Build Management, Test, Entwicklung, Weiterentwicklung von Script für SCM, Build Management, Trouble Shooting

- Compilieren der durch Entwicklung gelieferten Software und Lieferung der compilierten Artefakte an eine zentrale Integration, Troubleshooting und Support von Entwicklung, Kommunikation mit Release Management, Test.
- Durchführung von System-Tests
- Vorbereiten von Dokumentation für die Publikation: Der Kunde war im Begriff, Dita einzuführen. Dafür wurden Dokumente aus dem Word-Format konvertiert.

Technischer Hintergrund Linux, Perl, Ruby, Python, Shells, ClearCase

Eingesetzte Tools Perl, Ruby, Python, Shells, Cleartool, Clearmake, QualityCenter, Oracle (toad), CVS, Apache, Tomcat

Fachlicher Hintergrund Administration, Troubleshooting, Entwicklung

Eingesetzte Skills Entwicklung, Perl, . . . , SCM, Linux-Kenntnisse, XML, XSLT, Dita

November 2005 – April 2006 Internet-Dienstleister in Karlsruhe

Umfeld Internet-Services

Aufgaben Installation und Wartung einer großen Anzahl Server-Maschinen

Technischer Hintergrund Linux, Perl, Shells, Cfengine

Eingesetzte Tools Perl, Shells Cfengine

Fachlicher Hintergrund Entwicklung, Administration

Eingesetzte Skills Entwicklung, Administration, Perl, Linux-Kenntnisse, Cfengine

Februar 2005 – September 2005 Finanzdienstleistungs-Unternehmen in Münster

Umfeld Automatische Installation von Linux-Servern und Windows-Clients in Filialen

Aufgaben Installation, Support, Troubleshooting, Wartung und Entwicklung von Perl- und Shell-Scripts

Technischer Hintergrund Linux, Perl, Shells

Eingesetzte Tools Perl, Shells, . . .

Fachlicher Hintergrund Entwicklung, Administration

Eingesetzte Skills Entwicklung, Administration, Perl,
Linux-Kenntnisse

Juni 2003 – April 2004 Telekommunikations-Unternehmen in Ulm

Aufgaben Installation, Applikationssupport, Troubleshooting.
Wartung und Entwicklung von Perl- und Shell-Scripts. Analyse
von ASCII-Log-Daten des Mail-Systems eines großen ISPs mit
dem Ziel, Anhaltspunkte für Mißbrauch und Spam zu entdecken.

Technischer Hintergrund Unix, Linux, Perl

Eingesetzte Tools Perl, Shells, . . .

Fachlicher Hintergrund Entwicklung in Perl,
Systemadministration

Eingesetzte Skills Entwicklung, Administration, Perl, XML, UNIX,
MySQL, PostgreSQL

Februar 2003 – April 2003 Telekommunikations-Unternehmen in
Eschborn

Aufgaben Software Installation scripten, Testen

Technischer Hintergrund verschiedene UNIX Derivate, darunter
Linux

Eingesetzte Tools Perl

Eingesetzte Skills Perl, Shell Script

Juli 2002 – August 2002 Telekommunikations-Unternehmen in Münster

Aufgaben Systemadministraton, Applikationssupport,
Troubleshooting

Technischer Hintergrund Sparc (E6500, E10K, E15K), Sun
Cluster, Compaq TruCluster

Eingesetzte Tools UNIX Diagnose Tools, Tivoli

Funktion im Projekt Systemadministrator

Fachlicher Hintergrund Systemadministration für insgesamt etwa 400 UNIX-Systeme (Solaris und Tru64); Process Quality (OS), Troubleshooting; 2nd Level Support und Koordination von Änderungsmaßnahmen zwischen Betrieb, Systemtechnik und Lieferanten (SUN, Compaq); Performance-Tuning; Applikation von Patches; Softwareinstallationen und -konfigurationen

Eingesetzte Skills Solaris und Tru64, Script (Shell)

März 2002 Telekommunikations-Unternehmen in Düsseldorf

Umfeld Einführung eines Systems, das Benutzer und Arbeitsplatz-Ressourcen verwaltet

Aufgaben Analyse und Überarbeitung von Benutzerdaten

Technischer Hintergrund Linux, Perl, Java, MySQL

Eingesetzte Tools NetBeans, Perl

Funktion im Projekt Entwickler

Fachlicher Hintergrund Überarbeitung von Java Code, Entwicklung von Perl Scripts mit Zugriff auf eine MySQL Datenbank

Eingesetzte Skills Linux, Perl, Java

März 2001 – Dezember 2001 Versicherungs-Unternehmen in Köln

Umfeld Betrieb von Internet Lösungen

Aufgaben Administration, Operating

Technischer Hintergrund Linux, Windows, Perl

Funktion im Projekt Entwickler, Operator

Fachlicher Hintergrund Entwicklung und Wartung von Perl Scripts im Rahmen der Administration, operative Tätigkeiten, Einrichten und Warten einer Linux-Firewall, Konzeption und Einrichten von High-Availability Services mit Heartbeat

Eingesetzte Skills Perl, XML, Linux, Windows

September 2000 – Februar 2001 Web Shop Hosting Service

Umfeld Betrieb von WWW Shops für große Kunden

Technischer Hintergrund Linux, Solaris, BroadVision, Perl

Funktion im Projekt Entwickler

Fachlicher Hintergrund Anbindung der Shop Lösung an die beim Betreiber bestehende Infrastruktur: Import von Warenbeständen, Im-, Export von Order-Daten, Generieren von Endkunden-Rechnungen, Lieferscheinen, ...

Eingesetzte Skills Perl, Solaris, XML, XSL, XSLT und Xalan

Februar 2000 – August 2000 Consulting Unternehmen in Hüttlingen

Umfeld Microsoft Consulting Unternehmen

Aufgaben Entwicklung in C++, Perl, Visual Basic

Technischer Hintergrund Windows 2000, Perl, Visual Basic, Visual C++, PDA, Handheld, Windows CE

Funktion im Projekt Entwickler (als Angestellter)

Fachlicher Hintergrund Entwickeln in Perl, Visual Basic, C++ unter Windows 2000

Eingesetzte Skills Perl, Visual Basic, C++

1995 – 1999 Multimedia-Agentur in Aalen

Umfeld Computer-Animation, Multimedia-Präsentationen, World-Wide Web Projekte

Aufgaben Überarbeiten einer Promotion CD mit Macromedia Director, Aufbau und Administration eines Netzwerkes, in dem World-Wide Web Auftritte realisiert und gehostet werden, Entwickeln von Software fürs World-Wide Web, e. g. einer Anwendung, mit der Theater-Karten an Endkunden oder über Vorverkäufer verkauft, reserviert und ausgedruckt wurden

Technischer Hintergrund Perl, Linux, Networking, Wintel, IRIX

Eingesetzte Tools Macromedia Director, Apache, Sendmail, Perl, CGI, ...

Funktion im Projekt Administrator, Entwickler (als Angestellter)

Eingesetzte Skills Software Entwicklung, Administration

1999 Schulungs-Unternehmen in Ulm

Aufgaben Schulungen zur Einführung in UNIX und Administration

Technischer Hintergrund Linux, UNIX

Funktion im Projekt Dozent

Fachlicher Hintergrund UNIX Einführung, Linux Administration

Eingesetzte Skills Linux

1999 Hersteller von Friseur Einrichtungen in Schwäbisch Gmünd

Umfeld Sicherstellen des Betriebs eines AIX Servers

Aufgaben Bestehende Daten-Bestände mit Hilfe von
Datenbank-Scripts nutzbar machen

Technischer Hintergrund AIX, Ingres

Eingesetzte Tools AIX Administrations Tools (e. g. smitty), Ingres

Funktion im Projekt Entwickler, Administrator

Fachlicher Hintergrund Analyse und Überarbeitung von
bestehenden Datenbank-Scripts, Administrativa

Eingesetzte Skills UNIX, Datenbank Knowhow

1999 Telekommunikations-Unternehmen in Erfurt

Umfeld Sicherheits-Check

Technischer Hintergrund Windows, UNIX, Datenbank-Software

Funktion im Projekt Berater

Eingesetzte Skills UNIX

1997 Dokumentations-Unternehmen in Neuler

Umfeld Herstellung von Literatur

Aufgaben Verfassen der Schnellübersicht Java 1.1 (Markt + Technik, 1997)

Technischer Hintergrund Java 1.1 Syntax, AWT

Eingesetzte Tools Java Workshop

Funktion im Projekt Autor

Fachlicher Hintergrund Dokumentation, Java

Eingesetzte Skills Java

Köln, 24. Juni 2018