



Vladislav Nazarenko

Systems Engineer



SOCKET

Key Skills

Analyse, Entwicklung und Integration der IT Systeme
Umsetzung komplexer Projekte mit Verwendung der neusten Technologien
Ergebnisorientierter Team Player

Qualifikationen

Cloud
Unix/Linux Systeme
Kenntnisse verschiedener Programmiersprachen
Configuration Management
Netzwerkkennntnisse
Datenbanken: SQL und NO SQL

Wichtige Informationen

Schwerpunkt	Cloud Infrastructure, Network, Security
Wohnsitz	Frankfurt
Erfahrung	16
Stundensatz	Auf Anfrage

Persönliche Daten	
Name	Vladislav Nazarenko
Ausbildung	Technische Universität Taganrog, Russland
Grad	Ingenieur im Fach „Computergestützter Entwurf(CAD)“
Alter	40
Sprachen	Deutsch, Englisch, Russisch
Tätig in	Rhein-Main-Gebiet
E-Mail	vnazarenko@socket.de
IT Kenntnisse	
Cloud	Kubernetes, OpenStack, OpenShift, Azure, Google Cloud
Betriebssysteme	RedHat, Debian, Ubuntu, SuSE
Configuration Management	Puppet, Ansible
Virtualisierung	KVM, VMWare
Networks	Cisco, Citrix
Load Balancing	HA Proxy, NetScaler, Apache, Nginx
Datenbanken	MySQL, PostgreSQL, MongoDB, couchbase, memcache
Monitoring	Prometheus, Grafana, Zabbix
Programmiersprachen	Python, Ruby, C/C++, Shell/Bash, PHP, PowerShell
Verteilte Systeme	HAProxy, Pacemaker, Keepalive, Squid
Web Server	Apache, Nginx
Zertifizierungen	
2021	Certified OpenStack Administrator
2019	Certified Kubernetes Administrator
2018	Red Hat Certified Specialist in Ansible Automation
2017	Cisco Certified Entry Networking Technician
2014	Puppet Certified Professional
2013	SuSE Certified Linux Administrator
2012	ITIL 2011 Foundation
2011	Red Hat Certified System Administrator
2011	MySQL Professional Database Administrator
Persönliche Merkmale	
Mehrere Jahre Erfahrung als	Systems Engineer Software Engineer Datenbankadministrator
Persönliche Eigenschaften	Analytische Denkweise Team-orientiert Flexible Arbeitsethik

Zeit	Projekt
<p>10/2021 -</p> <p>PFALZKOM</p> <p>Systems Engineer</p> <p>OpenStack, Docker, Networks, Ceph, Kolla-Ansible, Artifactory</p>	<p>Migration der OpenStack Cluster</p> <p>Der Kunde hat schon seit mehreren Jahren eine eigene Privat Cloud auf der Basis von OpenStack erfolgreich betrieben. Das Ziel des Projekts war die Verbesserung des Betriebs der OpenStack Cluster. Das Management hat sich einen flexibleren Betrieb, einen schnelleren Releasewechsel sowie eine bessere und schnellere Skalierbarkeit gewünscht.</p> <p>Dafür wurden die OpenStack Umgebungen von dem klassischen Betrieb auf Kolla-Ansible (Containerisierte Dienste) umgestellt. Nach Wunsch des Kunden wurde Migration ohne Service-Unterbrechung durchgeführt, sodass die End User den Wechseln gar nicht mitbekommen haben.</p> <p>Für eine bessere Testbarkeit beim Releasewechsel wurde ein Verfahren eingeführt, das automatisiert die Funktionalität der Cluster überprüft und somit vor einer Regression des Service nach dem Releasewechsel oder anderen Änderungen warnen kann.</p> <p>Anforderungen</p> <ul style="list-style-type: none"> • Verbesserung des Betriebs • Schnellere Releasewechsel • Bessere Skalierbarkeit • Keine Service-Unterbrechung • Bessere Testbarkeit <p>Aufgaben</p> <ul style="list-style-type: none"> • Erstellung der Dokumentation • Neuaufbau und Migration der Cluster • regelmäßige Updates der bestehenden Cluster • Anbindung der Storage Systeme • User Support und Bug Fixing
<p>12/2020 – 12/2021</p> <p>Behörde</p> <p>DevOps Engineer</p> <p>OpenShift, S3 Storage, ElasticSearch, Ansible, Nvidia GPU,</p>	<p>Administration OpenShift</p> <p>Im Rahmen des Projekts sollte eine moderne Infrastruktur aufgebaut werden, die einerseits agile Verfahren unterstützt und andererseits klassischen Ansätzen nicht widerspricht, sodass nach dem Aufbau nach und nach die Migration der vorhandenen Applikationen stattfinden kann.</p> <p>Dafür wurden mehrere OpenShift 4.X Cluster aufgebaut, deren Anbindung an die schon vorhandene Infrastruktur (Loadbalancer, Storage, LDAP, Image Registry, Ansible Tower) die Hauptaufgabe war.</p> <p>Parallel dazu wurde an eine AI-Applikation gearbeitet, deren Betrieb später auf der OpenShift Basis erfolgen sollte. Dazu wurde ein</p>

Zeit	Projekt
Artifactory	<p>Deployment-Verfahren erstellt und Elasticsearch Cluster aufgebaut, sodass die Migration auf OpenShift möglichst schnell und mit wenig Aufwand durchgeführt werden konnte.</p> <p>Anforderungen</p> <ul style="list-style-type: none"> • Stabilität im Betrieb • Unterstützung der autonomen Arbeitsweise der Teams • Betrieb einer AI-Applikation (Auf Baremetal und im OpenShift) <p>Aufgaben</p> <ul style="list-style-type: none"> • Administration der OpenShift Cluster • Aufbau eines Elasticsearch Clusters • Aufbau der Deployment Pipelines • Anbindung eines S3 Storage Systems • Support der Entwickler
06/2020 – 10/2020	POC: Einführung OpenShift
Hornbach Baumarkt	<p>In den letzten Jahren wurden in eigenen Rechenzentren des Kunden mehrere große Kubernetes Cluster aufgebaut. Damit wurde der Betrieb der „Containerized Applications“ ermöglicht.</p>
DevOps Engineer	<p>Im Bereich Betrieb der Kubernetes Cluster sind aber auch weitere, neue Aufgaben entstanden und haben zu einer zusätzlichen Belastung des Betriebsteams geführt. Ebenfalls hat man im Bereich Applikationsentwicklung den Wunsch gemerkt, agiler beim Auf- und Abbau der Kubernetes Cluster zu sein.</p>
OpenShift, OKD, OpenStack, Kubernetes, CoreOS	<p>Im Laufe des Projekts sollte überprüft werden, inwieweit OpenShift Cluster die bestehenden Kubernetes Cluster ablösen können, sodass auch gleichzeitig der Betrieb besser unterstützt wird und dadurch das Betriebsteam entlastet werden kann.</p>
	<p>Anforderungen:</p> <ul style="list-style-type: none"> • automatisierte Installation und Updates der Cluster • Unterstützung von „Scale“ und „Auto Scale“ Funktionen • Homogenität der Cluster • Erweiterbarkeit (Anbindung an operatorhub.io) • Unterstützung von Abläufen
	<p>Aufgaben:</p> <ul style="list-style-type: none"> • Evaluieren der OpenShift 4.X und OKD 4.5 Versionen

Zeit	Projekt
	<ul style="list-style-type: none"> • Vorbereitung zur Ablösung der vorhandenen K8S Clustern • Vorbereitung zur Ablösung der RedHat/CentOS Cluster Knoten durch CoreOS/FedoraCoreOS • Bessere Integration mit der bestehenden Infrastruktur
06/2016 – 10/2020	Design, Aufbau und Betrieb einer PaaS Lösung
Hornbach Baumarkt	<p>Im Rahmen des Projekts wurde nach einer Plattform as a Service(PaaS) Lösung gesucht, die den agilen Entwicklungsteams ermöglichen würde, entsprechend schon aufgesetzten Scrum-Prozessen und völlig autonom von anderen Teams (Netzwerk, Infrastruktur) zu arbeiten.</p>
DevOps Engineer	<p>Die folgenden Anforderungen wurden gestellt:</p>
Docker, Kubernetes, Ansible, RedHat, Python, Java-Umfeld	<ul style="list-style-type: none"> • Unterstützung der Service orientierten Architektur • Skalierbarkeit • simple und standardisierte Recovery-Prozesse • Anbindung an die bestehende Legacy Environment mit gemischter Infrastruktur (Windows/Linux, Physical/Virtual) • Unterstützung der autonomen Arbeitsweise der Teams • Möglichkeit weitere Services von der alten Infrastruktur abzukoppeln und auf die PaaS Lösung zu übertragen.
	<p>Nach der Evaluierungsphase wurde die Entscheidung getroffen, die PaaS Lösung auf der Basis von Docker und Kubernetes aufzubauen. Der Aufbau erfolgte in Zusammenarbeit mit folgenden Teams des Kunden:</p>
	<ul style="list-style-type: none"> • Netzwerk • Infrastruktur • Entwicklung
	<p>Nach dem Aufbau erfolgte die Übergabe des Betriebes an das Infrastruktur-Team, das zuvor entsprechend geschult wurde.</p>
	<p>Aufgaben:</p>
	<ul style="list-style-type: none"> • Evaluierung der verfügbaren Open Source Tools • Design und Aufbau der Plattform • Configuration Management • Erstellung der Dokumentation • Unterstützung der Entwickler-Teams • Schulung und Übergabe des Betriebs an das Infrastructure-Team
07/2019 – 12/2019	Einheitliches und umfassendes Berechtigungskonzept
Hornbach Baumarkt	<p>Im Laufe der Jahre haben sich mehrere Probleme im Bereich Verwaltung der Berechtigungen gezeichnet. Es gab kein einheitliches Berechtigungskonzept, Systembetreuer haben jeweils einen Abschnitt im Active Directory angefordert, um dann allein und unabhängig von den</p>

Zeit	Projekt
DevOps Engineer Active Directory, Security Assertion Markup Language (SAML), LDAP, JSON Web Token (JWT), Ansible, PowerShell, Kubernetes, GitLab, Jenkins, ELK Stack, Nexus Repository, Microsoft Azure, Google Cloud, Docker Registry	<p>anderen Ihre Systeme verwalten zu können, infolgedessen zum einen das Berechtigen eines Benutzers recht aufwändig geworden ist, da kein Automatismus eingesetzt werden konnte. Zum anderen war ebenfalls jede Reorganisierung/Optimierung der Organisationsstruktur mit großem Aufwand verbunden. Der Grund dafür lag an organisationspezifischen Berechtigungen, die dabei alle angepasst werden mussten.</p> <p>Die Analyse des IST- und SOLL- Zustands ergab mehrere wichtige Punkte. Nach mehreren Diskussionsrunden und in Zusammenarbeit mit</p> <ul style="list-style-type: none"> • Systembetreuern • Teamleitern • Administratoren von Active Directory <p>wurden Anforderungen für das Konzept festgelegt und anschließend notwendige Tools dafür entwickelt und in Betrieb genommen.</p> <p>Anforderungen:</p> <ul style="list-style-type: none"> • Alle vorhandene Systeme + • Schnittstellen LDAP, SAML + • Projektbezogene Berechtigungen • Teamleiter können Struktur des Teams festlegen (Team Roles) • Teamleiter können Ihre Mitarbeiter für Ihre Projekte berechtigen • Unterstützung der Systeme verschiedenster Arten (System Roles) • Idempotenz • Erweiterbarkeit <p>Aufgaben:</p> <ul style="list-style-type: none"> • Analyse des IST- und SOLL- Zustands • Forschung nach möglichen Lösungen und Ansätzen • Das Berechtigungskonzept ausarbeiten und validieren • Entwicklung des Tools zur Umsetzung des Konzepts • Inbetriebnahme des Tools • Dokumentation und Übergabe
09/2014 – 06/2016 Plus.line Systems Engineer Ansible, Cloudstack, KVM, Networking, RedHat	<p>Design und Aufbau einer Cloud Plattform</p> <p>Eine Cloud Infrastruktur kann dabei helfen, flexiblere aber auch vordefinierte(gut erprobte) Lösungen einzusetzen, mehr Unabhängigkeit von Hardware-Bestand zu erreichen und deswegen auch schneller und qualitativ besser Kundenbeauftragungen umzusetzen.</p> <p>Für die Umsetzung dieses Projekts wurden zuerst oVirt, CloudStack und OpenStack ausgewählt, aufgesetzt und evaluiert. Die Ergebnisse aus der Evaluierungsphase wurden in eine Entscheidungsvorlage eingearbeitet und dem Kunden vorgelegt. Cloudstack wurde für den Einsatz gewählt, weil das Tool am besten Anforderungen passt, flexibel und gleichzeitig</p>

Zeit	Projekt
	<p>überschaubar komplex ist.</p> <p>In Zusammenarbeit mit Netzwerkspezialisten wurde dann eine Plattform auf CloudStack Basis für den produktiven Einsatz entwickelt. Entsprechend den typischen Anforderungen der Endkunden wurden Testszenarien ausprobiert und evaluiert, sowie auch Disaster/Recovery Tests ausgearbeitet und durchgeführt, um mögliche Probleme im Voraus zu erkennen und Lösungen dafür vorzubereiten.</p> <p>Anschließend wurde für den sicheren Betrieb der Plattform die Dokumentation erstellt und das Personal geschult.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Evaluierung der verfügbaren Open Source Tools • Design und Aufbau der Plattform • Configuration Management • Definition und Durchführung der Disaster/Recovery Tests • Erstellung der Dokumentation
<p>11/2015 – 02/2016</p> <p>Deutsche Vermögensberatung</p> <p>Systems Engineer</p> <p>Puppet, Ruby, Shell, RedHat, Projektmanagement</p>	<p>Migration der Linux Systeme</p> <p>Im Rahmen des Projekts mussten über 400 Linux Systeme auf den neusten Stand des Configuration Managements migriert (neu aufgebaut) werden.</p> <p>Es war sehr herausfordernd Systeme nicht nur abzubilden (Puppet), sondern vor allem die Abnahme durch die fachlichen Verantwortlichen und technischen Tester erfolgen zu lassen.</p> <p>Damit der Neuaufbau möglichst flexibel und in Nebenzeiten mit unterdurchschnittlicher Auslastung (nachts und am Wochenende) erfolgen konnten, wurden Mitarbeiter des Betriebes(24/7) zusätzlich geschult.</p> <p>Auf Grund der guten DevOps Erfahrungen wurden komplexe Systeme in der Produktion von erfahrenen Software und Systems Engineers und in der Zusammenarbeit erneuert, damit mögliche Fehler (Software oder Systemkonfiguration) auf einem sehr kurzem und schnellem Wege gefixt werden konnten.</p> <p>Nach der eigentlichen Migration wurde noch eine Phase eingeplant und umgesetzt, um mögliche neue Betriebsprobleme abzufangen und diese schnell zu lösen.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Abbildung der Systeme mit Puppet • Schulung der Betriebsmitarbeiter • Planung und Koordination der Arbeiten

Zeit	Projekt
	<ul style="list-style-type: none"> • Support der Systeme während und nach der Migration
10/2013 – 05/2016	Administration und Betrieb der DVAG Plattform
Deutsche Vermögensberatung	Die DVAG Plattform stellt den Vermögensberatern notwendige Tools zu Verfügung. Außer der Produktion gehören zu der Plattform auch Umgebungen, die zum Testen, der Integration und Entwicklung benötigt werden. Eine sehr breite Palette der Technologien wird verwendet. Zu der Gesamtplattform gehören etwa 2000 verschiedene Server.
Systems Engineer	
Puppet, Ruby, Shell, RedHat, KVM, Tomcat, Serverspec	<p>Zu den wichtigsten Aufgaben gehört die Minderung des Administrationsaufwands, dafür wird „Puppet“ Configuration Management Tool eingesetzt. Die Ziele dabei sind: Einsatz der aktuellen Puppet Version, Qualität und Wartbarkeit des Puppet Codes.</p> <p>Der Puppet Code wird getestet, indem in einer separaten Umgebung „Nightly Builds“ gemacht und ausgewertet werden, Testaufgaben wurden anhand Skripte automatisiert. Serverkonfiguration wird durch das „Serverspec“ Tool geprüft.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Serveradministration mit Puppet • Betrieb der Plattform • Testautomatisierung • Verkürzung der Deployment Downtime
07/2011 – 10/2013	Betrieb der GM World Platform
General Motors	Die GM World Platform wurde konzipiert, um alle Märkte (Webauftritte) der GM Brands (Chevrolet, Opel, Buick, Cadillac ...) zu konsolidieren.
Linux Systems Engineer	Die gesamte Plattform besteht aus mehreren Umgebungen (DEV/ITL, PRE, PROD) und umfasst über 400 verschiedene Server.
RedHat, Xen, MySQL, Puppet, Shell, Perl	<p>Zu dem Betrieb gehören unter anderem die folgenden Aktivitäten:</p> <ul style="list-style-type: none"> • Durchführung der Changes • Behebung der Fehler • Störungsbeseitigung <p>Die Aktivitäten werden nach ITIL v3 Standard durchgeführt. Dem Kunden steht auch ein Monitoring Sytem und 24/7 Hotline zur Verfügung, sodass Probleme rechtzeitig erkannt und rund um die Uhr beseitigt werden können.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Betrieb • Technischer Support und Beratung der Kunden(Frankfurt, Detroit) • Performanzanalyse • Unterstützung des Testteams (Fehleranalyse) • Monitoring

Zeit	Projekt
02/2013 – 10/2013	Konzeption und Umsetzung der automatisierten Deployments
General Motors	Software Deployments auf GM World Platform können wegen der hohen Komplexität bis zu mehreren Stunden andauern, nehmen in Anspruch mindestens einen Administrator und verlangen von ihm eine sehr hohe Konzentration
Linux Systems Engineer	Die Platform besteht unter anderem aus mehreren Umgebungen, die den Life Cycle der Software unterstützen: Entwicklung, Testing und Betrieb. Somit wird ein Release mehrmals und in verschiedenen Umgebungen (DEV/ITL, PRE, PROD) installiert (bis alle Fehler behoben sind und das Release online genommen werden kann).
Shell, Perl, Python, Ant	Ziel des Projekts ist die Automatisierung solcher komplexen Deployments, damit der Prozess ohne Administrator Interaktionen ablaufen kann. Somit ist die Anwesenheit eines Administrators nur im Falle eines Fehlers erforderlich, sodass der Administrator mehrere Deployments gleichzeitig begleiten kann. Die Dauer der Deployments wird zusätzlich dadurch reduziert, dass die Deployments gleichzeitig auf mehreren Komponenten durchgeführt werden.
	Aufgaben: <ul style="list-style-type: none"> • Verkürzung der Deploymentzeit • Eliminierung der menschlichen Fehler • Gleichzeitige Deployments • Vorbereitung zur wachsenden Anzahl der Markets/Deployments
01/2010 – 06/2011	Wartung und Weiterentwicklung des zentralen Systems zur Verkehrssteuerung
Lufthansa Passage	Das System zur Verkehrssteuerung — STORM (System Tool for Operation Rotation and Maintenance Requirements Traffic) dient der Planung, Steuerung und Überwachung von Flügen, sowie der Bodenereignisse der eingesetzten Flugzeuge der Lufthansa Passage.
Projektleiter, Software Engineer	Als Realtime-System stellt es die Ist-Situation gegenüber dem geplanten Flugplan dar und erlaubt es dem Nutzer, korrigierende oder präventive Maßnahmen zur Sicherung von Pünktlichkeit, Regelmäßigkeit, Wirtschaftlichkeit und Sicherheit vorzunehmen.
C/C++, Python, Perl, Oracle, SQL, Linux, Shell Scripting, XML	STORM besitzt eine Vielzahl von Funktionen, die über die letzten Jahre speziell auf die Anforderungen der Lufthansa Passage zugeschnitten wurden. Ebenso existieren über 20 Schnittstellen zu anderen Systemen, um die Benutzer mit allen nötigen Informationen zu versorgen.
	Das System gehört somit zu den wichtigsten Applikationen zur Aufrechterhaltung des Flugbetriebs, weswegen eine höchst mögliche Verfügbarkeit gewährleistet werden muss.
	Aufgaben:

Zeit	Projekt
	<ul style="list-style-type: none"> • Projektmanagement • Release Planung • Ressourcenplanung • Wartung und Weiterentwicklung der Frontendapplikation • Erstellung der Dokumentation, Spezifikationen, Angebote • Erstellung, Durchführung und Auswertung der Testfälle • Forschung
11/2007 – 06/2011 Lufthansa Passage DB-Administrator Oracle 9i - 10g, SQL, PL/SQL, Linux, HP/UX, Shell Scripting, Perl	Administration der Crew Management Systeme(CMS) Datenbanken <p>Die Crew Management Systeme beinhalten insgesamt über 15 Oracle Datenbanken (für Test- und Produktionsumgebungen), die von über 22.000 Mitarbeitern benutzt werden., Die Datenbanken enthalten eine Vielzahl personenbezogener Daten, haben Schnittstellen zu circa 30 anderen Systemen und werden von 60 Mitarbeitern weiterentwickelt.</p> <p>Die wichtigsten Datenbanken laufen in Hochverfügbarkeitsclustern, um eine höchste Verfügbarkeit, mit einer maximalen Umschaltzeit von weniger als 5 Minuten in den Notbetrieb gewährleisten zu können.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Datenbankadministration • Erstellung und Pflege der Dokumentation • Konzeption Backup, Restore und Reorganisation • Regelmäßige Statusberichte zu den einzelnen Datenbanken • Sicherstellung und Überwachung der Hochverfügbarkeit und Datensicherungen • Konfiguration und Optimierung • Einführung und Verbesserung von Betriebskonzepten • Analyse der Datenbank auf Fehler und Speicherbedarfs-Trend • Tuning-Empfehlungen zu den einzelnen Anwendungen • Versionsplanung und Tests
09/2009 – 05/2010 Verschiedene Fluggesellschaften Web-Entwickler Python, MySQL, SQL, Javascript, Linux, Shell Scripting, Apache, Zope	Entwicklung eines Reporting Web Portals für europäischen Emissionshandel <p>Der EU-Emissionshandel (European Union Emission Trading System, EU ETS) ist ein marktwirtschaftliches Instrument der EU-Klimapolitik mit dem Ziel, die Treibhausgasemissionen unter minimalen volkswirtschaftlichen Kosten zu senken.</p> <p>Fluggesellschaften sind durch dieses Instrument aufgefordert, alle Flüge innerhalb eines Jahres, bei denen ein europäischer Flughafen angeflogen wird, an eine zuständige europäische Behörde zu berichten. Abhängig von der Größe einer Fluggesellschaft und aufgrund der komplexen Anforderungen der Europäischen Union, stellt die Erzeugung der benötigten Berichte einen enormen Aufwand verbunden mit einem komplizierten Prozess dar.</p>

Das Hauptziel der Web-Applikation liegt darin, den Reporting Prozess zu erleichtern sowie die aktuelle Situation abzubilden und Prognosen zu ermöglichen. Die Prognose ist hierbei von enormer Relevanz, da hierdurch bereits frühzeitig der Erwerb von benötigten Emissionszertifikaten initiiert werden kann und somit Einsparungen erzielt werden können. Weiterhin können mit dem System standardisierte Reports erstellt werden, die von den Fluggesellschaft jährlich an nationale Behörden weitergereicht werden müssen.

Aufgaben:

- Entwicklung und Integration des Frontends in ein Web Portal
- Koordination und Synchronisation der Entwicklung mit dem Backend
- Aufbau der Entwicklungsumgebung
- Erstellung und Durchführung der Testfälle
- Forschung(Verbesserung der Performanz)

02/2007 – 12/2009**Lufthansa Passage****Software Engineer****C/C++, Python, Perl,
Oracle, SQL, XML,
Linux, HP/UX, Shell
Scripting****Wartung und Weiterentwicklung des zentralen Systems zur Verkehrssteuerung**

Das System zur Verkehrssteuerung — STORM (System Tool for Operation Rotation and Maintenance Requirements Traffic) dient der Planung, Steuerung und Überwachung von Flügen, sowie der Bodenereignisse der eingesetzten Flugzeuge der Lufthansa Passage.

Als Realtime-System stellt es die Ist-Situation gegenüber dem geplanten Flugplan dar und erlaubt es dem Nutzer korrigierende oder präventive Maßnahmen zur Sicherung von Pünktlichkeit, Regelmäßigkeit, Wirtschaftlichkeit und Sicherheit vorzunehmen.

STORM besitzt eine Vielzahl von Funktionen, die über die letzten Jahre speziell auf die Anforderungen der Lufthansa Passage zugeschnitten wurden. Ebenso existieren über 20 Schnittstellen zu anderen Systemen, um die Benutzer mit allen nötigen Informationen zu versorgen.

Das System gehört somit zu den wichtigsten Applikationen zur Aufrechterhaltung des Flugbetriebs, weswegen eine höchst mögliche Verfügbarkeit gewährleistet werden muss.

Aufgaben:

- Wartung und Weiterentwicklung der Frontendapplikation
- Wartung und Weiterentwicklung der Decision Support Komponenten
- Administration der Workstations
- Erstellung der Spezifikationen
- Unterstützung bei Durchführung und Auswertung der Testfälle
- Forschung

Zeit	Projekt
11/2007 – 11/2008 Lufthansa Passage Software Engineer C/C++, Oracle, SQL, HP/UX, Linux	<p>Portierung des zentralen Systems zur Verkehrssteuerung von HP/UX nach Linux</p> <p>Zur Identifikation der für eine Performance Optimierung notwendigen und geeigneten Bereiche im Programmcode wurde vorbereitend eine Profiling-Analyse durchgeführt. Dabei wurden auf Basis definierter Standard-Geschäftsfälle die intern im Frontend benötigten Zeiten für Einzelfunktionen gemessen und ausgewertet. Auf Basis dieser Analyse wurden in Abstimmung mit dem Kunden Optimierungsmaßnahmen definiert. Eine dieser Maßnahmen sah Migration von HP/UX auf Linux vor.</p> <p>Die Migration des Verkehrssteuerungssystems auf Linux war notwendig, da der Hersteller der bisherigen Workstations keine aktuellen Modelle mit vier Grafikausgängen mehr auslieferte und insbesondere für eine solche Konfiguration keinen Support mehr leistete. Für das Betriebssystem Linux konnte hingegen eine Versorgung mit Hardware auf dem aktuellen Leistungsstand sichergestellt werden.</p> <p>Die Migration sollte die Performance der Applikation insbesondere in Bezug auf die Geschwindigkeit der GUI-Komponenten verbessern.</p> <p>Aufgaben:</p> <ul style="list-style-type: none"> • Anpassung des Software Codes an Linux • Konfiguration der Linux Workstations analog zu den alten HP/UX Workstations • Erstellung der Dokumentation und des Handbuchs zum Aufsetzen einer Linux Workstation • • Durchführung und Auswertung der Testfälle • Forschung