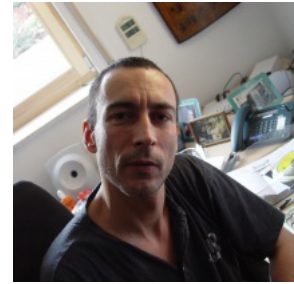


Profil
Ralph Gerstmann



Zur Person	Geburtsjahr:	1965
	Einsatzorte:	Weltweit
Ausbildung	<p>1971-79: Diverse Schulen in Bonn, London, Bukarest. 1980-84: Abitur in Helsinki. 1984-85: Studium Informatik, abgebrochen. 1986-88: Lehre DV-Kaufmann, abgeschlossen. 1989-92: Angestellter bei zwei Systemhäusern. In dieser Zeit 12 Wochen UNIX und Internet-Kurse. Seitdem: Learning by doing.</p>	
Praktische Kenntnisse und Erfahrungen		
Aufgabenbereiche	<p>Erstellung von Designs und Spezifikationen. Begleitung bei deren Umsetzung mit Rat und Tat. Software-Entwicklung und Programmierung, Fehleranalyse, Beratung, Projektmanagement, Administration und Support.</p>	
Fachlicher Schwerpunkt	<p>UNIX, Linux, Solaris, TCP/IP, OpenSource, System.- u. Netzwerkmanagement, Troubleshooting und Debugging. Client/Server Programmierung, Internet Service Provider. Hochverfügbare Systeme und Netzwerke. (Failover/Loadbalancing/Replikationen/Virtualisierung)</p>	
Netzwerkkenntnisse (OSI1-6)	<p>Internet, Intranet, ISO/OSI, Router, Switches, dynamisches und asynchrones Routing, TCP/IP, ICMP, ARP, RPC (Sun-RPC, ONC), Windows Netzwerk, Traffic Analysis. (Ethernet, STP). Diverse Firewalls.</p>	
Netzwerkprotokolle (OSI7)	<p>HTTP, HTTPS, SMTP, SNMP, DHCP, DNS, VPN, FTP, NFS (+ mehr)</p>	

Datenbanken	MySQL, SQL, Oracle, DB2. (Sybase, MaxDB, Postgres, Informix.)
Betriebssysteme	Unix, Linux, SUN OS, Solaris, (HPUX), Windows Server. Diverse Linux-Derivate, Diverse Unix-Derivate, BIG-IP LTM.
Programmiersprachen APIs und Tools	C, C++, Perl, PHP, Shell, awk, MinGW, cygwin, COBOL, SQL, ESQ/C, libc, Sun-RPC, ONC, sockets, gdb, make, makepp, git, CVS, svn. C++: boost, RCF, mysql++, poco
Virtualisierung/High Availability/Load Balancing	Kvm, xen, vmware, libvirt, pacemaker, heartbeat, openais, corosync, drdb, csync2, rsync. Application Traffic Management: F5 BIG-IP Local Traffic Manager. Load Balancing: DNS, HTTP, SMTP, MySQL-Master-Master, Linux, Masquerading, NAT, OpenVPN.
Anwendungen und Tools	Open Source Software, Linux, Apache, Asterisk, BIND, DNS, Dovecot, IMAP, Cyrus, Cacti, Nagios, RRDTOOL, tcpdump, wireshark, strace, truss, MRTG, Big Brother. Request Tracker, Postfix, Sendmail, dspam, amavis, clamav. Samba, Squid, cygwin. Iptables, shorewall, csf, Firewall Builder. (Tomcat.)
Branchen	Hosting, Internet Service Provider Rechenzentrum, Systemhersteller Bank, Finanzdienstleister, Telekommunikation, Industrie, Handel
Sprachen	Deutsch: Muttersprache Englisch: Verhandlungssicher in Wort und Schrift Spanisch: Grundkenntnisse Französisch: Grundkenntnisse

Projekte		
Jan. 2012 – Juni 2014	SAP	
<p>Virtual Appliance Factory (VAF):</p> <p>Entwicklung von Tools zum automatisierten abtrennen (und anbinden) von SAP Systemlandschaften von den (an die) jeweiligen Betriebssysteme(n).</p> <p>Datenbankserver, Debugging, SAP Hostagent, sapinst, Virtualisierung.</p> <p>Programmiersprachen: Active Perl, Shell-Scripte.</p> <p>Programmertools: Eclipse, svn.</p> <p>Betriebssystem: ESX (Virtualisierung), Windows Server 2008 und 2003, SLES 11.</p> <p>Datenbanken: Oracle, DB2, Sybase, MaxDB.</p>		
Juni 2010 – Dez. 2011	NetworkSys	
<p>System.- u. Netzwerkmanagement + Client/Server Programmierung.</p> <p>Beratung, Design, Implementierung und Wartung von komplexen Serverlösungen.</p> <p>Webserver, Datenbankserver, Mailserver, Monitoring, Alarming, Troubleshooting, Debugging, Virtualisierung, redundante Auslegung, Fehlerprognostizierung, Ausfallverhinderung, Hardwareauswahl, RAID, Security, SPAM.- und VirenFilter, Firewalls, Upgrades, Neuinstalltionen, Logfileauswertungen, Trouble-Ticketing.</p> <p>Design und Implementierung einer verteilten c++ basierten Client/Server-Lösung um Windows Client-Entwicklern ein transparentes API zur Webanwendung (MySQL, Asterisk) zu geben.</p> <p>Programmiersprachen: Shell-Scripts, Perl, PHP, c++, CGI, HTML.</p> <p>Programmertools und APIs: g++, MinGW, cygwin, gdb, make, makepp, valgrind, git. (C++: boost, RCF (remote call framework), mysql++, poco)</p> <p>Betriebssystem: KVM (Virtualisierung), Linux (Gentoo, RedHat), Windows.</p> <p>Hochverfügbarkeit: libvirt, pacemaker, heartbeat, openais, corosync, drdb, csync2, (ocfs2)</p> <p>Anwendersoftware: MySQL-Master-Master, Apache, openVPN, Boonex Dolphin, Asterisk, Dovecot-IMAP, Postfix, Squirrelmail, Roundcube, DSPAM, amavis, PHPmyAdmin, RRDTOOL, Nagios, Cacti, awstats, iptables, shorewall, csf. Request Tracker.</p>		

2008 – Mai 2010	Telefónica o2 Germany	
<p>Hosting Development Design</p> <ul style="list-style-type: none">• Design und Implementierung von komplexen Serverlösungen.• Entwicklung eines Installtions-Servers für redundant ausgelegte F5 BIG-IP Local Traffic Manager (LTM) Loadbalancer-Pärchen, welcher die Konfigurationsdaten weitgehend aus einer hauseigenen Hardware.- und Konfigurationsdatenbank berechnet.• Design, Referenzimplementierungen, Dokumentation, Freigabetests, betriebliche Bereitstellung von F5 Loadbalancerkonfigurationen, die gleichzeitig sowohl<ul style="list-style-type: none">○ NAT basiertes Application Delivery Networking (L7) beherrschen,○ als auch hocheffizientes Direct Server Response (asynchrones L4) (nPath).• Design eines Open Source basierten Mailserverclusters, um Lotus Notes abzulösen.• Mitarbeit im Projekt Hosting Capacity Planning. (Diverse Scripte).• Script zur vollautomatischen Paketerstellung für Redhat und Solaris.• MySQL Plugin zur flexibel konfigurierbaren Schwellwertüberwachung. <p>Programmiersprachen: Shell-Scripts, Perl, AWK, C.</p> <p>Betriebssystem: F5 BIG-IP LTM (9.3.1, 9.4.x, 10.01, 10.1), Linux (RedHat), Sun-Solaris.</p> <p>Anwendersoftware: MySQL, Postfix, Dovecot, OpenLDAP, Confluence, T-Wiki.</p>		
2005-2007 (verteilt ca. 18 Monate)	Handel	
<p>System.- u. Netzwerkmanagement für KARE:</p> <p>Beratung, Mailserver, Faxserver, Webserver, LDAP, DNS, Monitoring, Alarming, Troubleshooting, Debugging, Virtualisierung, redundante Auslegung, Ausfallsicherheit, Hardwareauswahl, RAID, Security, SPAM.- und VirenFilter, Firewalls, Upgrades, Neuinstalltionen, Logfileauswertungen, Trouble-Ticketing, Installationsserver, CMS.</p> <p>Programmiersprachen: PHP, Perl, Shell-Scripts, CGI, HTML.</p> <p>Betriebssystem: XEN, Linux (Gentoo), Windows.</p> <p>Anwendersoftware: Cyrus-IMAP, Postfix, Squirrelmail, DSPAM, amavis, Apache, OpenLdap, phpLDAPadmin, Hylafax, mySQL, PHPmyAdmin, RRDTOOL, Nagios, Cacti, awstats, webalizer, bind, iptables, fwbuilder, Joomla.</p>		

2003 – 2007 (verteilt ca. 15 Monate)	Handel	
<p>osCommerce Webshops für KARE</p> <p>Die Firma KARE Reiter + Schönhofen OHG wünschte sich einen Webshop und wollte diesen von Grund auf programmieren lassen. Ich konnte die Geschäftsleitung davon überzeugen, dass osCommerce die kostengünstigste und professionelle Basis darstellt. So konnte ich mit relativ geringem Aufwand, zwei Shops zaubern, inclusive Anbindung an die hauseigene Datenbanken. Letzten Endes gingen rund 20 weitere Franchise Shops aus dem Projekt hervor. Ich habe die Shop-Software so modifiziert, dass der Franchisenehmer lediglich eine Übersetzung der Texte liefern braucht und die Installation auf meiner Seite dann nur noch 10 Minuten in Anspruch nimmt. Das letzte nennenswerte Unterprojekt (2007) war die Programmierung einer neuen Datenbankschnittstelle basierend auf XML.</p> <p>Programmiersprachen: PHP, Perl, Shell-Scripts.</p> <p>Betriebssystem: Linux (RedHat und Gentoo).</p> <p>Anwendersoftware: osCommerce, Apache, MySQL, mysqlcc, mysqladmin, phpMyAdmin, RequestTracker, OpenVPN, FwBuilder, iptables.</p>		

2002 – 2004	Bank	
<p>System.- u. Netzwerkmanagement für RBG/Fiducia: Weiterentwicklung, Troubleshooting und Support des hauseigenen System.- u. Netzwerkmanagement Programms INES. Fertigstellung, Debugging und Support eines multithreaded Bulk-SNMP-basierten Accountingsystems auf Basis von SOAP, zur Rechnungsschreibung aller (16.000?) bayrischen Raiffeisenbanken. Entwicklung eines Installations-Servers für Debian-Linux basierend auf FAI. Entwicklung eines Installations-Servers für RedHat-Linux basierend auf anaconda. Ausfallprognose, Fehleranalyse, Schwellwertüberwachung, Monitoring, Alarming von Linux-Hosts, Sun-Solaris-Hosts, Cisco-Routern, IBM-Routern, Bridges, Netzwerk.- Laufzeiten und Durchsatz, CPU, Festplatten, IP-Traffic/Fehler, Ethernet-Traffic/Fehler, TCP-Traffic/Fehler, UDP-Traffic/Fehler, Datenbankverfügbarkeit, Prozesse u.v.a.m. Accounting von Partneranbindungen via RRDTOOL, Mysql, Apache, mit Übergabe der Accountigdaten an DB/2. Benutzerauthentifizierung an Routern via Radius, TACACS und LDAP. Performance.- und Laufzeitanalysen des aus rund 100 Java basierten Sun Solaris E10K und E15K Systemen bestehenden Online-Banking Clusters. Sämtliche dieser Systeme sind komplett ausfallsicher (redundant) ausgelegt, weil beim Abschalten oder Updaten eines Systems der Betrieb ohne Ausfall auf Anwenderseite, Datenverlust oder Eingriff von Admins gewährleistet sein muß. Programmiersprachen: PHP, Perl, Shell-Scripts, CGI, HTML, AWK, C. Betriebssystem: Sun-Solaris, Linux (SuSE, RedHat, Debian), Cisco IOS. Anwendersoftware: Apache, Nagios, RRDTOOL, MRTG, DB/2, MySQL, SNMP, OpenLDAP, SOAP.</p>		

2002 (2 Monate)	Internet Service Provider. Telekommunikation	
<p>R-TuX: Es ist mein eigenes, single-floppy-based TDSL-Router-Firewall-Linux-System, welches sich automatisch aus dem Internet initialisiert (zentrale Software-Pflege). Mit nur einer Diskette kann aus jedem 486er, der 2 Netzwerkkarten und min. 24 MB Speicher hat, in wenigen Minuten, ein vollwertiger T-DSL-Router gemacht werden. Beherrscht sowohl das Booten über das IntraNet ohne Floppy, als auch automatisches ISDN-Fallback.</p> <p>Programmiersprachen: Shell-Scripts.</p> <p>Betriebssystem: Linux, busybox, trinux.</p>		
2001 (2 Monate)	Internet Service Provider	
<p>SetUp: Ist eine Tool-Sammlung, welche aus einer minimal installierten, sicheren Solaris Maschine, einen richtig brauchbaren GPL-Server (GNU) macht. Optimalerweise ist es in Kombination mit JumpStart einzusetzen. SetUp installiert aber auch Linux-Derivate.</p> <p>Programmiersprachen: Shell-Scripts, Perl CGI.</p> <p>Betriebssystem: Solaris/Linux.</p> <p>Anwendersoftware: CVS.</p>		
2000-2001	Internet Service Provider, Bank	
<p>WebMail/IntraNet Portal</p> <p>Ziel war, den bestmöglichen Mailserver zu bauen. Er bestand aus den edelsten Komponenten, Limits wurde ausschliesslich durch die Hardware gesetzt. Bei entsprechender Hardware sollten bis zu 0,5 Mio Benutzer möglich sein. Inclusive Projektverwaltung, Zeitverwaltung, TroubleTicketing, Chat und mehr.</p> <p>MRTG und Big Brother von HTML nach PHP portiert.</p> <p>Programmiersprachen: C, PHP, Perl, CGI, HTML.</p> <p>Betriebssystem: Linux/Solaris.</p> <p>Anwendersoftware: Apache, MRTG, Big Brother, Intas.</p>		

1999-2002	Internet Service Provider, Telekommunikation, Rechenzentrum, Dienstleistung	
<p>Projekt technischer Aufbau einer ISP Struktur</p> <p>Für einen Münchener Internet Service Provider habe ich die komplette Accounting.- und Abrechnungssoftware programmiert, diverse weitere Systeme konzipiert.</p> <p>Kundenbetreuung, Serverinstallationen und Wartung.</p> <p>Security, diverse Router.</p> <p>Betriebssystem: Solaris, Linux, Cisco IOS, Bay Networks Nautica, Concorde.</p> <p>Programmiersprachen: C, Scripts, PHP, Perl, CGI, HTML.</p> <p>Datenbanken: Oracle, MySQL, dbm.</p>		
1998 (6 Monate)	Finanzdienstleister	
<p>ZVT-700</p> <p>Ist das bekannte Zahlungsverkehrsterminal der Giesecke & Devrient. Spitzname: "Der Winkel", zu bewundern an fast jeder Tankstelle. Ich darf hier nicht viel mehr sagen, als ein wenig. Der Auftrag war, den in Assembler gewachsenen Code, durch eine strukturiertere Version in C zu ersetzen.</p> <p>Programmiersprachen: Keil-C, Assembler.</p>		
1997	Bank	
<p>CA Unicenter TNG</p> <p>Ich war verantwortlich für die Evaluierung dieses Beta-Systems, sowie aller zugehörigen administrativen Aufgaben, die Entwicklung eines SNMP-Agenten, welcher die 3 Windows/NT-Logfiles überwacht, filtert und entsprechende SNMP-Traps verschickt. Zum deutschlandweiten zentralisierten Sammeln von gefilterten Eventlogeinträgen und deren Integration in das NMS.</p> <p>Programmiersprachen: C, C++.</p> <p>Betriebssystem: Windows/NT.</p> <p>Programmiertools: Visual C++, ESQ-L-C (Oracle), awfactory.</p>		

1996 (6 Monate)	Bank	
<p>Aufbau Second Level Support</p> <p>Support- und Integrationsunterstützung für Investment Banking Systeme mit dem Schwerpunkt Solaris-Systeme.</p> <p>Integration in die bestehende Hostumgebung im Bereich Risikomanagement. Einarbeitung der Administratoren mittels Workshops.</p> <p>Evaluierung und Auswahl von einzusetzenden Softwarepaketen.</p> <p>Leitung der Entwicklung einer Software, die paralleles Batch-Processing realisiert. Administration von Solaris Systemen.</p> <p>Aushandeln und fixieren von Service Level Agreements.</p> <p>Programmiersprachen: Shell-Scripts, C.</p> <p>Betriebssystem: Solaris, Windows/NT.</p> <p>Tools: make.</p>		
1995 (3 Monate)	Industrie	
<p>Belegleser</p> <p>Der Teilbereich "Portierung von OS/2 nach Solaris" war zu realisieren. Da extrem Wert auf Performanz gelegt wurde, mußte u. A. mit Posix-Threads gearbeitet werden.</p> <p>Programmiersprache: Multithreaded C (Posix).</p> <p>Betriebssystem: Solaris/Intel.</p>		
1993-1995	EDV, Systemhersteller	
<p>Lance+ ist ein ziemlich komplexes NMS (Netzwerk Management System).</p> <p>Ich war verantwortlich für alle systemnahen Aufgaben, Portierungen auf weitere UNIX basierende Plattformen, Design und Implementierung von diversen Erweiterungen.</p> <p>Programmiersprachen: C, C++, Tcl/TK, Shell-Scripts.</p> <p>Betriebssystem: SunOS, HP-UX, Solaris.</p> <p>Programmiertools: ESQL-C (Ingres), Motif, X11.</p> <p>Tools: GNU (RCS, CVS, Emacs, gdb).</p>		

1993-1995 (20 Wochen)	EDV, Systemhersteller	
<p>Kurse zu folgenden Themen habe ich geleitet:</p> <ul style="list-style-type: none">• UNIX-Systemprogrammierung.• Netzwerkprogrammierung unter UNIX.• TCP/IP und NFS in der Praxis. <p>Programmiersprachen: C.</p> <p>Betriebssystem: SunOS, Solaris.</p> <p>Programmertools: SparcWorks.</p>		
1991-1992	EDV, Systemhersteller	
<p>WORM-Server</p> <p>Design und Realisierung eines verteilten WORM-Servers basierend auf NFS, mit Dateisystem in Informix und Oracle Datenbank.</p> <p>Programmiersprachen: C, Shellscripts.</p> <p>Betriebssysteme: SINIX, HP-UX, SunOS.</p> <p>Programmertools: RPC, ESQL-C (INFORMIX und Oracle), Motif, X11.</p>		
1990-1991	EDV, Systemhersteller	
<p>Consulting Netzwerke</p> <p>Als Projektleiter bei einem Systemhersteller, in der Abteilung Consulting Netzwerke tätig, Beratung von Großkunden im Bereich Internet.</p>		
1989	EDV, Systemhersteller	
<p>Ttytool ist ein OpenWindows-Programm zum Testen und Konfigurieren von seriellen Schnittstellen an Sun-Workstations. Mitarbeiter der Hotline wurden in diesem Rahmen von mir geschult. Ttytool hat, nicht zuletzt aufgrund der 100 Seiten umfaßenden Dokumentation, den Supportaufwand in diesem Bereich nachhaltig enorm reduziert.</p> <p>Programmiersprachen: C, Shell-Scripts.</p> <p>Betriebssystem: SunOS.</p> <p>Programmertools: DevGuide, OpenWindows, X11</p>		

1988	EDV, Systemhersteller	
<p>Second Level Support Unix/TCP-IP Deutschlandweiter Last-Level Support Solaris/SunOS, beim Hersteller.</p>		
1987	EDV, Handel	
<p>Hauseigene SINIX Distribution Im Rahmen dieses Projektes wurden alle hauseigenen SINIX spezifischen Systemanpassungen, die ich entwickelt habe, in einer eigenen hauseigenen SINIX-Installation zusammengefasst. Dazu gehörten Druckerbackends (C), Portierungen (Scripts & Schnittstellen), Kunden-Datenbank (Informix). Angebotsschreibung (HIT), Installationssoftware (SINIX-MUS), Administration und Konfiguration (eigene Scripts). Programmiersprachen: C, Shell-Scripts, und COBOL (MicroFocus). Betriebssystem: SINIX. Programmiertools: ESQL-C, MES. Anwendersoftware: INFORMIX, HIT.</p>		
1986-1987	EDV, Handel	
<p>Münchner Warenwirtschaft MüWa ist ein komplexes Programm zur Kunden-, Artikel- und Auftragsverwaltung. Mitarbeit als Projektleiter und Programmierer. Diverse individuelle Anpassungen wurden durchgeführt. Programmiersprachen: COBOL (RM), Clipper und dBASE III. Betriebssysteme: CP-M 3.0, RSX-11M, MS-DOS 2.11. Programmiertools: Hauseigener Cobol-Source-Generator.</p>		