

**„You've got mail“  
Wie E-Mail in die Mailbox  
kommt**

KNF Kongreß 2001

Henning P. Schmiedehausen

<hps@intermeta.de>

# Geschichtsstunde, Teil 1

- Die erste E-Mail wurde vor 30 Jahren über das ARPANet verschickt (Ray Tomlinson, BBN)
- seitdem wird das @-Zeichen als Trennzeichen zwischen lokalem Benutzer und Host verwendet
- Sie enthielt „etwas banales wie QWERTYUIOP“
- zwei Jahre später war E-Mail schon 2/3, fünf Jahre später 75% des ARPANET Traffic

# Geschichtsstunde, Teil 2

- 1982 löst TCP/IP die bisherigen ARPANET Protokolle ab.
- IP = INTERNET Protocol
- August 1982: Jon B. Postel veröffentlicht RFC821 „SIMPLE MAIL TRANSFER PROTOCOL“
- SMTP ist (mit Erweiterungen) bis heute das Standardprotokoll für E-Mail im Internet

# Geschichtsstunde, Teil 3

- 2000: 550 Millionen Menschen benutzen E-Mail
- Etwa 1,5 Milliarden gültige E-Mail Adressen
- mehr als 10.000.000.000 E-Mails pro Tag

(Quelle: Critical Path)

rund 15% davon sind SPAM

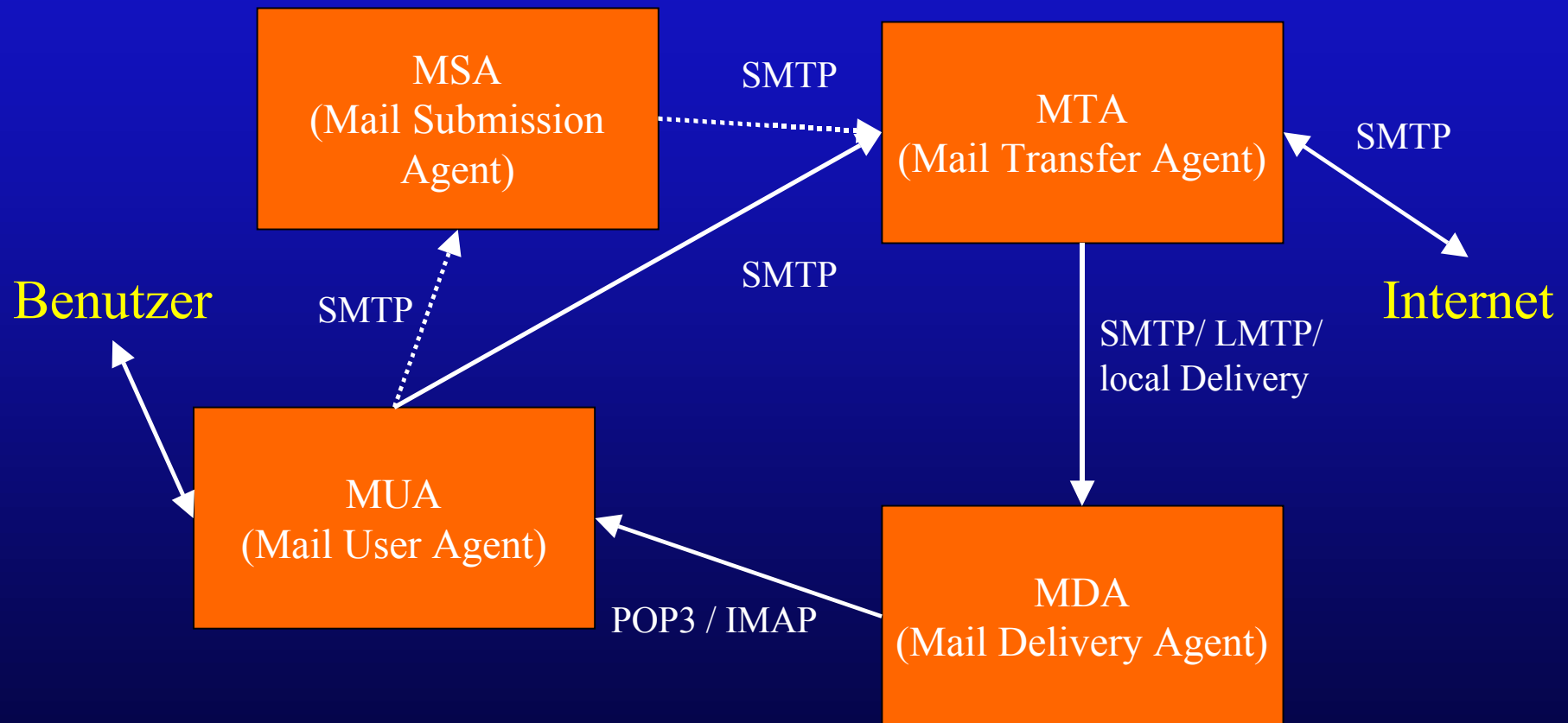
# E-Mail ist DIE Killerapplikation des Internets

## Nicht WWW

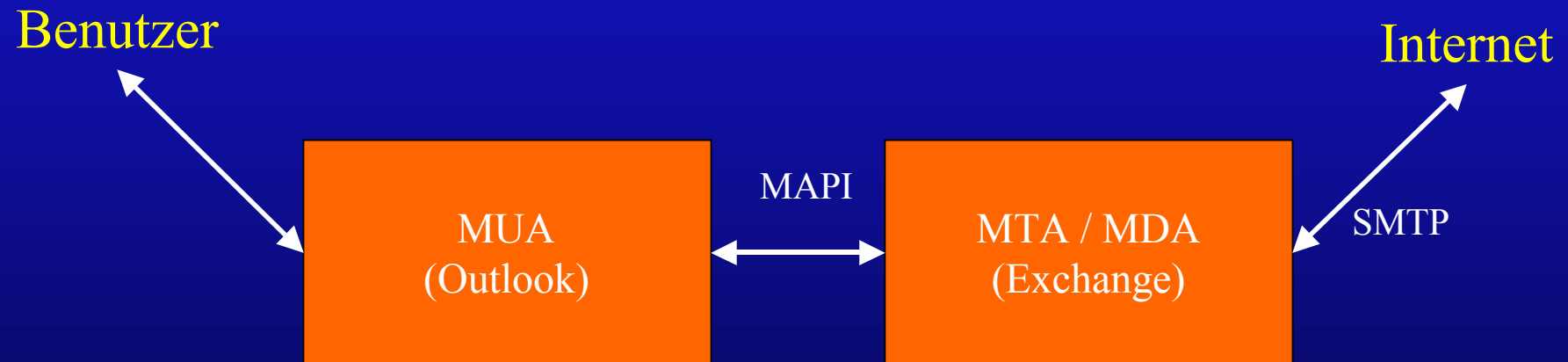
# „three letter acronyms“

- MUA - Mail User Agent
- MTA - Mail Transfer Agent
- MDA - Mail Delivery Agent
- MSA - Mail Submission Agent

# Offene Protokolle



# Un-offene Protokolle



- MAPI ist ein proprietäres Protokoll
- Outlook XP benutzt WebDAV (offenes Protokoll)

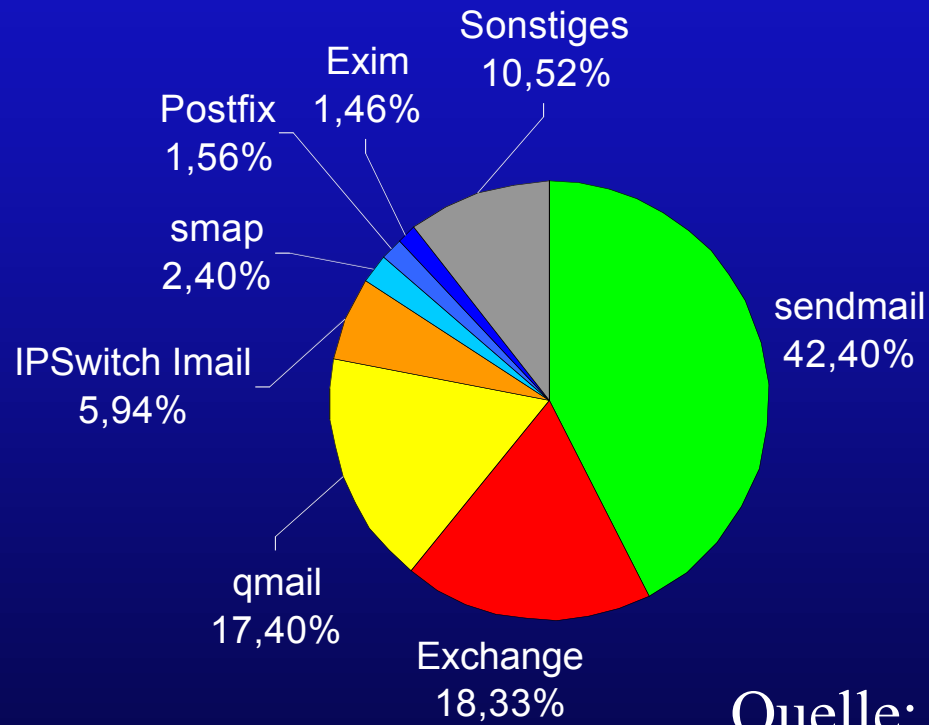


# Von diesem Punkt an geht es um MTAs und offene Protokolle

# MTAs - eine Übersicht

- MTAs (Mail Transfer Agents) dienen zum Transport von E-Mail zwischen Rechnern
- MTAs sind meist Hintergrundprogramme (Daemons (Unix), Services (Windows))
- ein MTA ist Bestandteil der meisten Unix-Varianten (/usr/lib/sendmail)
- MTAs sind keine „group ware“

# MTAs weltweit



Quelle: Dan J. Bernstein  
Mailer Survey 10/2001

E-Mail Transferprogramme  
sind das Rückgrat des Internets

Deswegen ist es wichtig, daß  
man das richtige Programm  
benutzt

# sendmail

# oder ein vergleichbares Programm

# Open Source MTAs

- sendmail ([www.sendmail.org](http://www.sendmail.org))
- qmail ([www.qmail.org](http://www.qmail.org))
- Postfix ([www.postfix.org](http://www.postfix.org))
- Exim ([www.exim.org](http://www.exim.org))
- ZMailer ([www.zmailer.org](http://www.zmailer.org))
- smail (<ftp://ftp.planix.com/pub/Smail/>)

# Open Source MTAs

- Jedes Programm hat Stärken und Schwächen
- Individuelle Schwerpunkte sind unterschiedlich
- Support und Weiterentwicklung wird unterschiedlich intensiv betrieben
- Verschiedene Open Source Lizenzen
- Diskussion „Welches ist der beste MTA?“ ist stark ideologiegetränkt
- Ein neutraler Standpunkt ist schwierig



# smail

- ca. 1987, aktuell ist 3.2.0\_114 (07/2001)
- ca. 68.000 Zeilen C-Code
- monolithisches Design (ein Programm)
- einfache Kommando-Syntax, mehrere Konfigdateien
- verschiedene Protokolle (SMTP, BSMTP, UUCP)
- kein TLS/SSL, LDAP o.ä.
- primitive SPAM-Schutzmaßnahmen

# ZMailer

- ca. 1988, aktuell 2.99.55 (02/2001)
- ca. 161.000 Zeilen C-Code
- Zwei-Daemon Design:  
Router und Queue Manager (Scheduler))
- Konfiguration durch Shellscript-ähnliche Syntax  
in mehreren Konfigurationsdateien
- Einfache Konfigurationen mit Hilfe weniger  
Variablen möglich

# ZMailer

- Primär SMTP orientiert
- Hilfsprogramme für andere Protokolle (UUCP)  
TLS Support für sichere E-Mail
- LDAP Support für Router und Aliases
- SPAM Schutz über DNS Abfragen und lokale Dateien
- IPv6 Support

# exim

- erste Version ca. 1995, aktuell 3.33 (08/2001)
- ca. 88.000 Zeilen C-Code
- als Ersatz für smail entworfen
- monolithisches Design (ein Programm)
- Kommandosyntax ähnlich smail, Aufteilung in Router, Transport und Directors
- Module werden entweder extern aufgerufen oder müssen einkompiliert werden

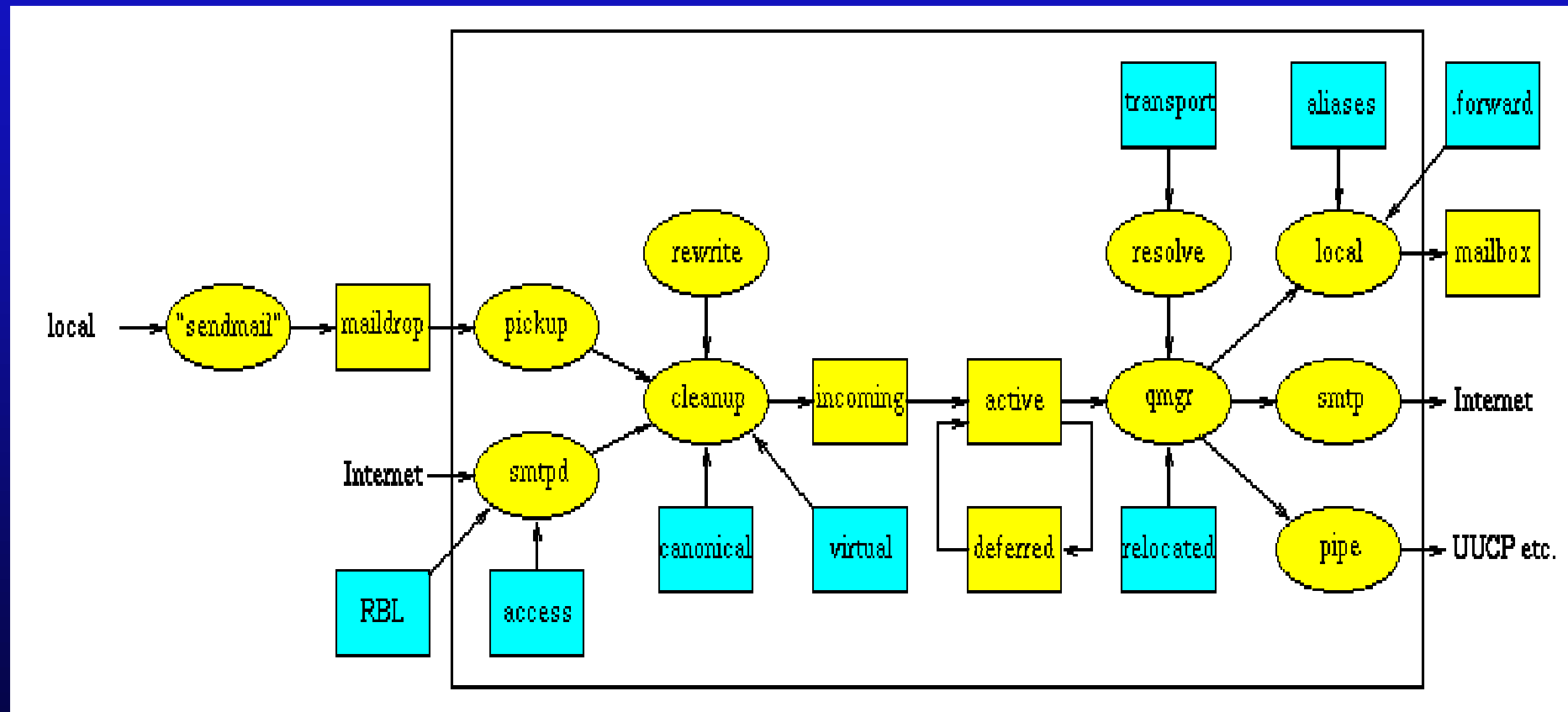
# exim

- SMTP orientiert, andere Protokolle (UUCP usw.) über Hilfsprogramme
- TLS Support für sichere E-Mail, IPv6 Support
- LDAP und SQL (MySQL, PostgreSQL) Support für Router und Aliases
- Alle o.g. Abfragemöglichkeiten (Dateien, DNS) können zur Authentifizierung (SPAM Schutz) verwendet werden

# Postfix

- seit 1998, aktuell 20010228p18 (11/2001)
- Entwicklung wird von IBM unterstützt
- Autor (Wietse Venema) betrachtet Postfix als „direct competitor to the qmail software [...]“
- Von Anfang an mit den Zielen „safety, robustness, security“ (unter anderem) geplant und entwickelt
- ca. 85.000 Zeilen C-Code
- Multi-Daemon Design (viele kleine Programme)

# Postfix Architektur



# Postfix

- Konfiguration in nur einer Datei, mit sehr wenigen Parametern einsatzbereit (sinnvolle Defaults)
- SMTP orientiert
- LDAP und MySQL Support
- SPAM-Schutz: „Postfix implements the usual suspects“, also DNS und Dateiabfragen usw.
- Kein TLS, kein IPv6



# qmail

- Der ideologisch am meisten belastete MTA
- „qmail is a modern SMTP server which makes sendmail obsolete“
- Der Autor (Dan J. Bernstein) führt eine Kampagne gegen viele Formen von populärer Software (BIND, sendmail, inetd, httpd usw.) die er für „minderwertig“ oder „fehlerhaft“ hält

# qmail (2. Versuch)

- erste Version 1998, aktuelle Version 1.03
- ca. 16.000 Zeilen C-Code
- Laut Autor „secure package“ weil niemand einen \$1000 Preis für einen qmail-Bug eingefordert hat.
- besteht ähnlich wie Postfix aus mehreren kleinen Programmen und Daemons
- Konfiguration erfolgt mit Hilfe von vielen kleinen Dateien in einem Konfigurationsverzeichnis

# qmail

- Reiner SMTP Mailer, andere Protokolle nur durch externe Programme unterstützt
- Bringt seinen eigenen MDA mit (und seinen eigenen POP Server)
- Kein TLS, kein LDAP, keine Datenbanken
- kein IPv6
- nur rudimentärer SPAM-Schutz (Liste erlaubter Domains)

# qmail

- qmail kann wenig Dinge
- qmail hat wenige Features
- qmail hat einen Autor und eine „Benutzer-gemeinde“, die neutrale User eher abschrecken

Warum also der Hype?

# qmail

- qmail ist schnell
- qmail ist sicher
- qmail hat wenig Code
- qmail hat einen relativ geringen Bedarf an Speicher und CPU
- qmail ist für Mail Switches eine sinnvolle Wahl

# sendmail

- „This is a monstrosity by almost any reasonable definition.“ Yet, it has been very successful. -- Eric Allman
- sendmail war das erste Programm, daß Mails zwischen verschiedenen Netzen transferieren konnte (ARPANet und BerkNet, ca. 1977)
- Aktuelle Version sendmail 8 hat mit dem Ur-sendmail nur noch den Namen gemeinsam (und sendmail.cf)
- sendmail ist kein MTA:  
„it implements a general purpose internetwork mail routing facility under the UNIX<sup>®</sup> operating system.“

# sendmail

- erste Version 1977, sendmail v8 ca. 1992
- aktuelle Version 8.12.1 (11/2001)
- 118.000 Zeilen C-Code
- monolithische Architektur (/usr/lib/sendmail)
- eine Konfigurationsdatei (sendmail.cf)
- Umfangreiches Macrosystem zur Konfiguration (m4 Macros), Konfiguration erfolgt durch sendmail.mc

# sendmail

- Unterstützt verschiedenste Protokolle (SMTP, UUCP, BITNET) bis hin zu direkter Fax-Auslieferung
- TLS, LDAP Support, keine SQL Datenbanken
- IPv6 Support, MSA Support
- Schnittstelle für externe Mailfilter (libmilter)
- Umfangreiche SPAM-Schutzmechanismen
- Kommerzieller Support (Sendmail, Inc.)



# sendmail

- sendmail hat viele (zu viele?) Features
- sendmail kann fast alles
- sendmail ist schon „lange am Markt“ und man bekommt kommerziellen Support

Warum also die Abscheu vor sendmail?

# sendmail

- Die Konfigurationsdatei (sendmail.cf) sieht für Normalsterbliche wie eine Binärdatei aus
- sendmail ist groß und unübersichtlich
- Code ist relativ alt und nicht leicht zu verstehen
- sendmail ist langsam, Skalierung erfolgt oft über Anzahl der Server
- sendmail ist die richtige Wahl, wenn ein MTA mit vielen Features gesucht wird

# Fazit

- Es gibt verschiedene Open Source MTAs
- Alle erfüllen die grundlegenden Aufgaben zufriedenstellend und haben spezifische Stärken und Schwächen
- Für den jeweilige Anwendungsfall sollte ein Mailer gezielt ausgesucht werden

Fragen?

Vielen Dank für Ihre  
Aufmerksamkeit