Next Generation Internet
2. Internet-Architektur
— Prinzipien und Entwicklung

INSTITUT FÜR TELEMATIK

KIT – Universität des Landes Baden-Württemberg und
nationales Forschungszentrum in der Helmholtz-Gemeinschaft

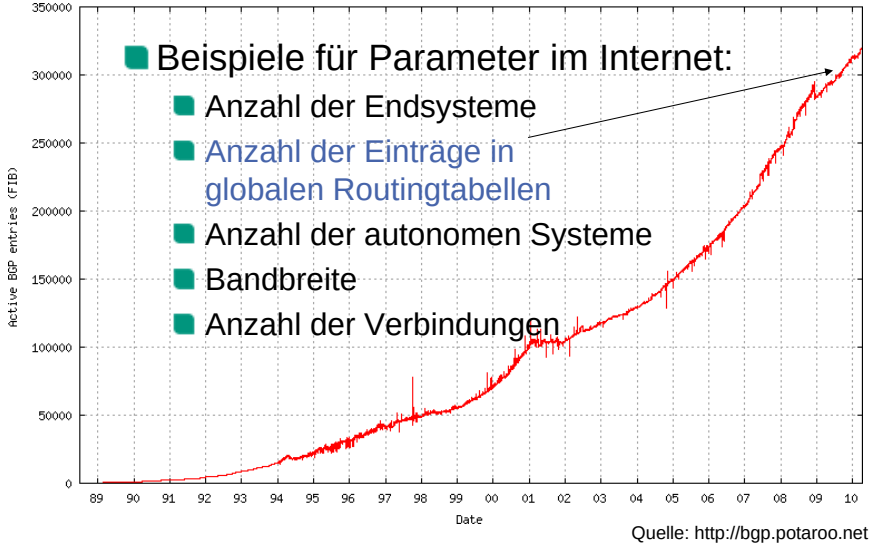
www.kit.edu

Kapitelübersicht

2.1 Wachstum und Skalierbarkeit

- Einzige Konstante im Internet: Ständige Veränderung
- Vor allem: Wachstum
- Technologischer Fortschritt bringt oftmals Steigerungen um Größenordnungen (z.B. Moore‘sches Gesetz, CPU, Bandbreite, Speicher)
- Generelle Frage: wie kommt das technische System damit klar?
  - Funktioniert es noch? Wie lange?
  - Nimmt die Leistung ab?

Beispiele: Wachstum im Internet



- Beispiele für Parameter im Internet:
  - Anzahl der Endsysteme
  - Anzahl der Einträge in globalen Routingtabellen
  - Anzahl der autonomen Systeme
  - Bandbreite
  - Anzahl der Verbindungen

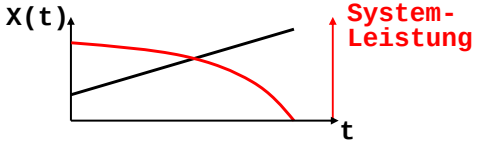
Quelle: http://bgp.potaroo.net

Die Graphik zeigt die Einträge in der Weiterleitungstabelle (FIB) eines Routers im Kernnetzbereich des Internets. Zu jedem Zielnetzpräfix gibt es dort genau einen Eintrag, der den nächsten Router entlang des „kürzesten“ Weges ausweist. Die richtigen Routingtabellen enthalten hingegen deutlich mehr alternative Routen (in der Regel zwei- bis dreimal so viele). Die Reduktion im Wachstum in der Zeit 2001-2002 ist auf strikteres Filtern zur Eliminierung unnützer Routinginformation zurückzuführen.

Bedeutung der Skalierbarkeit

- Das Internet hat das rasante Wachstum gut verkraftet
  → Es funktioniert noch!
  - Man sagt deshalb auch: es ist „skalierbar“?
  - Was bedeutet Skalierbarkeit?
- Begriff der Skalierbarkeit:

  Ein skalierbares System funktioniert auch bei starkem Wachstum (z.B. um mehrere Größenordnungen, d.h. über mehrere Skalen hinweg) bestimmter Parameterwerte des Systems
- Beispiel für keine bzw. schlechte Skalierbarkeit:



- Die Leistung eines nicht skalierbaren Systems nimmt bei Wachstum bestimmter Parameterwerte stark ab, ggf. bis zur Funktionsunfähigkeit
- Skalierbarkeit daher zentraler Aspekt beim Entwurf vieler Systeme
  → Erweiterung möglich, aber kein Neu-Design erforderlich

Skalierbarkeitsaspekte

- Skalierbarkeit bezieht sich auf bestimmte Systemparameter → welche berücksichtigen?
- Anwachsen um mehrere Größenordnungen betrachten, d.h. Steigerung um Faktor 10, 100, 1000, 10000, …
  → Funktioniert das System dann noch?
- Manchmal ist die Dynamik ein Problem
- Selbst lineares Wachstum bestimmter Parameter kann Problem sein!

- Skalierbarkeit bezieht sich immer auf bestimmte Systemparameter
  → Entscheidung jeweils im konkreten Fall erforderlich, welche berücksichtigt werden sollten.
- Selbst lineares Wachstum bestimmter Parameter kann u.U. bereits zu einer nicht mehr handhabbaren Gesamtkomplexität führen!

2.2 Internet-Architektur: Entwurfsziele

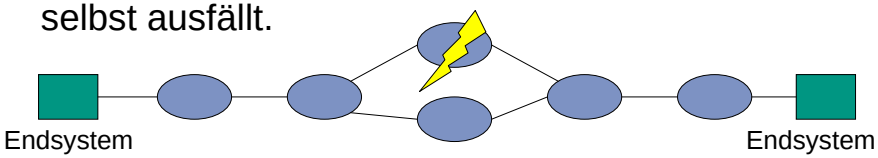
Paper von D. Clark *"The Design Philosophy of the DARPA Internet Protocols"* nennt: [Clark88]

- Hauptziel: Internetworking, d.h. Verbinden existierender Netzwerke
- Weitere Ziele (in Reihenfolge der Wichtigkeit):
  - Robustheit
  - Unterstützung mehrerer Arten von Kommunikationsdiensten
  - Heterogenität: Berücksichtigung einer Vielfalt von Netzwerken
  - Verteiltes Management der Ressourcen
  - Kosteneffektivität
  - Anschluss von Endsystemen mit wenig Aufwand
  - Ressourcennutzung muss abgerechnet werden können

- „Robustheit" in diesem Kontext: Internet-Kommunikation aufrecht erhalten trotz Ausfall von Netzen und Routern

Robustheit gegen Ausfall

- „Fate-Sharing": es ist akzeptabel Zustandsinformation, die mit einer Instanz assoziiert wird, zu verlieren, wenn die Instanz selbst ausfällt.

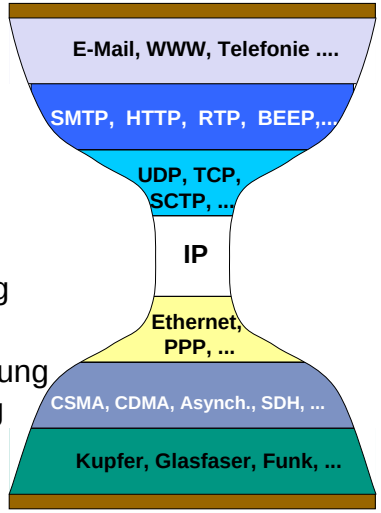


- Keinen Zustand im Netzwerk halten → stattdessen in den Endsystemen
- Datagramm-Konzept als Folge

- Warum heißt es „Fate Sharing"?: Schicksal der Verbindung ist an das Schicksal des Endsystems gebunden
- Was müsste sonst bei Zustandshaltung durchgeführt werden, wenn eine Instanz ausgefallen ist?
- Welche Zustände werden auch im Innern des Internets gehalten?
- Datagramm-Konzept: alle notwendigen Daten sind im Paket enthalten, kein Kontext in den Zwischensystemen erforderlich

Weitere Folgen

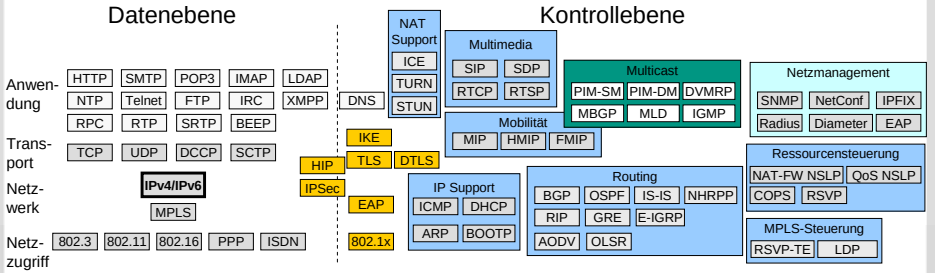
- Sanduhr-Modell
  - Einziges Protokoll
  - Globale Adressierung
  - Schlankes Protokoll
- Paketbasierte Kommunikation
  - flexibler als Leitungsvermittlung (weshalb?)
  - effizienter? → bessere Auslastung durch statistisches Multiplexing
  - schlechter abzurechnen



Sanduhr-Modell ist Folge der Ziele Interoperabilität und Heterogenität

- Einziges Protokoll
  - Maximiert Interoperabilität
  - Minimiert die Anzahl der Dienstschnittstellen
- Globale Adressierung
  - Konnektivität
- Schlankes Protokoll
  - Anspruchslos
  - maximiert Anzahl der einsetzbaren unterliegenden Netzwerke
- Leitungsvermittlung kann oberhalb von Paketvermittlung realisiert werden → virtuelle Leitungen

Schlanker Mittelteil?



- Kontrollprotokolle machen inzwischen einen Großteil der Gesamtkomplexität aus
- TCP/UDP + HTTP als neuer Mittelteil?

2.3 Entwurfsprinzipien: Ende-zu-Ende-Argument

- Welche Funktionalität wird benötigt?
- Wo sollen bestimmte Funktionen platziert werden?
  - In den Endsystemen/ Anwendungen?
  - Im Netzwerk?



[SaRC81]

- Wichtiges Entwurfsprinzip (erst 1981 explizit formuliert von Saltzer, Reed und Clark): Das Ende-zu-Ende-Argument (E2E argument)
  - Wissen und Hilfe der Anwendung notwendig → Funktionalität ins Endsystem
  - Keine anwendungsspezifische Funktionalität im Netz bereitstellen

Ende-zu-Ende Argument:

„Die fragliche Funktion kann nur vollständig und korrekt mit Wissen und Hilfe der Anwendung implementiert werden, welche sich an den Kommunikationsendpunkten befindet. Deswegen kann die fragliche Funktion nicht als Merkmal des Kommunikationssystems selbst bereitgestellt werden (manchmal kann aber eine durch das Kommunikationssystem bereitgestellte unvollständige Version der Funktion der Leistungssteigerung dienen).“

Diskussion Ende-zu-Ende-Argument

- Das heißt also insbesondere: spezifische Funktionen auf Anwendungsebene können üblicherweise und sollten vorzugsweise nicht im Netzwerk selbst platziert werden
- Minimalitätsprinzip: Vermeide, mehr als nur die wesentliche und notwendige Funktionalität ins Netzwerk zu integrieren. Halte „unnötige“ Funktionalität aus dem Netzwerk fern → Keep it simple
- Kein striktes Gesetz, eher Richtlinie

Beispiele für das Ende-zu-Ende-Argument

- Zuverlässige Übertragung einer Datei
  → mögliche Fehlerquellen:
  - Lesefehler im Endsystem
  - Softwarefehler während Kopieren oder Puffern von Daten durch Dateisystem, Dateitransferprogramm
  - Hardwarefehler während dieser Vorgänge in CPU, Speicher, Bus, etc.
  - Verlust, Bitfehler oder Duplikate im Kommunikationssystem
  - Endsystemcrash/-ausfall (Sender oder Empfänger) während oder nach der Übertragung
  - Zuverlässigkeit im Kommunikationssystem behebt nicht alle Fehler
- Aufteilung von TCP/IP in TCP und IP Ende der 70er Jahre
- Ende-zu-Ende-Sicherheit
- Unterdrückung von Duplikaten (z.B. durch Anwendung selbst erzeugt)

E2E Argument – weitere Ziele/Folgen

- Innovationsschutz
  - Einfaches Hinzufügen neuer Dienste
  - Infrastruktur nur schwer zu ändern (vgl. Einführung Multicast, IPv6, ECN, usw.)
- Zuverlässigkeit und Robustheit
  - Gegen Ausfall und Fehlfunktion der Endsysteme und Netzwerkkomponenten
  - Wenn Netzkomponenten Zustand halten müssen, wächst Wahrscheinlichkeit für Verbindungsausfall mit zunehmender Netzgröße

Internet-Architektur: Prinzipien

- RFC 1958: „Architectural Principles of the Internet“
- Unabhängigkeit von Medium und Hardware-Adressierung [RFC1958]
- Zustände (z.B. Routen, QoS-Garantien, Header Compression, ...) sollten „selbst-heilend“ sein
  - Adaptive Prozeduren und Protokolle zum Verwalten und Herleiten der Zustände
  - „Soft-State“-Konzept
  - Reduktion der Zustandsinformation auf Minimum (insbes. manuell konfigurierte Zustände)

Wenn Zustände gehalten werden müssen (z.B. Routen, QoS-Garantien, Header Compression, ...) sollten diese „selbst-heilend“ sein. Eine Möglichkeit dazu bietet das nachfolgend beschriebene Soft-State-Konzept.

- „Soft-State“-Konzept:
  - Zustand wird periodisch mit Hilfe erneuert („aufgefrischt“)
  - Zustand wird automatisch nach Ablauf einer bestimmten Zeitspanne (Timeout) gelöscht, falls nicht bis dahin durch explizite Nachricht erneuert; Zeitgeber wird bei Eintreffen einer Auffrischungsnachricht zurückgesetzt
  - Folge: Zustand wird automatisch gelöscht, falls Auffrischen ausbleibt, Zustand wird automatisch (wieder) etabliert
  - Vereinfacht Zustandsverwaltung im Netz, falls Endsysteme für Auffrischen des Zustands sorgen müssen
- Manuell konfigurierte Zustände sollten auf ein absolutes Minimum reduziert werden → Verwaltbarkeit sinkt sonst

RFC 1958 – Generelle Designpunkte

- Heterogenität
- Wiederverwendung bewährter Lösungen
- Skalierbarkeit
- Leistung und Kosten
- Einfach halten (keep it simple)
- Modularität
- Vermeide Optionen und Parameter wo möglich
  - verringern Usability und Interoperabilität
  - ansonsten: automatische Konfiguration und Aushandlung solcher Parameter
- Postel-Prinzip (s. ff. Folien)
- Zirkuläre Abhängigkeiten vermeiden

Erläuterung bzw. weitere Aspekte:

- Modularität ist gut, verwende Separation wo möglich
- Besser unvollständige Lösung übernehmen als darauf warten, die perfekte Lösung zu finden
- „Strenge beim Senden, Toleranz beim Empfangen“ → Postel-Prinzip
- Sparsamer Einsatz von Multicasts und Broadcasts
- Objekte sollten weitgehend selbstbeschreibend sein
- Gleiche Terminologie und Notation sowie Bit-/Byte-reihenfolge verwenden
- Existenz mehrerer, interoperabler, funktionierender Implementierungen
  spricht für Klarheit und Eindeutigkeit der Spezifikation

RFC 3439 – Richtlinien für Internet Backbone Provider

- "Some Internet Architectural Guidelines and Philosophy" (Ergänzung zu RFC 1958) [RFC3439]
- Erfahrung: Große Netze sind anders zu konstruieren als kleine und mittlere Netze
  - Nicht-Linearitäten bei Architektur, Entwurf
    - Ursache: Verstärkungseffekt, denn kleine Ereignisse können große Wirkung zeigen, bis zur Instabilität (z.B. Resonanzverstärkung)
    - Beispiele
      - Tacoma Narrows Brücke

      - erhöhte Interkonnektivität im Internet führt zu komplexerer und langsamerer Konvergenz (z.B. Verteilung v. Routinginformation)
  - Gegenmaßnahme
    → lokale Änderungen rufen nur lokale Effekte hervor

Einfachheits-Prinzip (Simplicity Principle)

- Komplexität wird oft durch
  - Robustheitsanforderungen in unsicheren Umgebungen
  - sowie zusätzliche Komponenten, die über die normale Funktionalität hinaus erforderlich sind,

  erzeugt
- Komplexität verhindert effizientes Skalieren: erhöht monetäre und operationale Kosten
- Komplexe Systeme weisen oft schwer kontrollierbare Abhängigkeiten zwischen Komponenten auf → Kopplungsaspekte
- Daher Komplexität kontrollieren
  → einfache Lösungen suchen

Kopplungs-Prinzip

- Kopplungs-Prinzip
  - Wachsendes System
    → erhöhte Abhängigkeit zwischen Komponenten
  - Mehr Ereignisse gleichzeitig
    → Interaktion von Ereignissen wird wahrscheinlicher
    → Unvorhergesehene Wechselwirkung
- Beispiele
  - Routing Update Synchronisation
  - TCP Slow Start Synchronisation
  - Congestion Collapse
  - Ariane-5 Crash
  - AT&T SS7 Failure

[FlJa94]
[Jaco88]
[ESA96]
[Neum90]

Ariane-5

→ Synchronisation verhindern: permanent Zufall hinzufügen

Je mehr Ereignisse sich gleichzeitig ereignen, desto größer die Wahrscheinlichkeit, dass mindestens zwei interagieren, dies führt zu unvorhergesehenen Wechselwirkungen

Beispiel Routing Update Syncronisation:

- Routingprozess: alle 30s eine Update-Nachricht (allg. Soft-State: Refresh)
- Obwohl alle Router unterschiedliche Startzeitpunkte wählen, wird der Routingverkehr irgendwann synchron → stark korrelierter, burst-artiger Verkehr von Routingdaten ist ungünstig
- Übergang zur Synchronisation erfolgt abrupt
- schwache Kopplung: Eintreffende Updates erzeugen neue
- Abhilfe: Zufall hinzufügen
  - erforderliche Menge der Zufälligkeit relativ groß
  - Wähle Sendezeit aus Intervall $[0.5T_p, 1.5T_p]$, T_p: Periode
  - Zufall bricht Synchronisation auf, d.h. wenn sich das System einmal zufällig in einen synchronisierten Zustand begibt, wird dieser durchbrochen

- AT&T SS7 Fehler: Der Ausfall wurde durch einen Software-Fehler in einer Wiederanfahrprozedur einer Vermittlungsstelle nach einem Ausfall verursacht. Eine Vermittlungsstelle fiel aufgrund eines Hardware-Fehlers aus und rief einen Ausfall der benachbarten Vermittlungsstellen beim Wiederanfahren aus. Diese erzeugten wiederum Ausfälle bei anderen Vermittlungsstellen beim Wiederanfahren usw.

Layering Considered Harmful

- Zahlreiche Mechanismen werden in verschiedenen Schichten wiederholt eingesetzt
  - Adressierung, Verbindungsaufbau, Fehlerkontrolle, Flusskontrolle, Fragmentierung
- Kapselung der Funktionen erschwert Optimierung
  - Es kann nur jede Schicht für sich optimiert werden
  - aber schichtenübergreifende Optimierung erhöht Kopplung zwischen den Schichten → verletzt Einfachheitsprinzip
- Zunehmende Schichtung und Abhängigkeiten zwischen den Schichten verletzten Einfachheitsprinzip
  - Reduktion der Komplexität am Beispiel IP Transport IP/ATM/SONET → IP/SONET/WDM → IP/WDM

Weitere Architektur-Richtlinien

Ebenfalls als gefährlich erachtet:

- Optimierung
  - erhöht meistens Komplexität, erzeugt engere Kopplung
- Überfrachtung mit Funktionen („Feature Richness“)
- Konvergenz-Schichten (z.B. IP over ATM)
- Universelles Interworking

Weitere Entwurfsaspekte

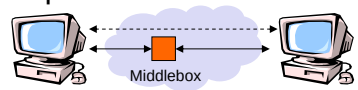
- RFC 3426 „General Architectural and Policy Considerations“ [RFC3426]
  - grundsätzliche Fragestellungen zum Protokoll-/System-Entwurf
  - keine Richtlinien, keine Checkliste
  - Diskussion und Erläuterung anhand zahlreicher Fallbeispiele (z.B. ECN)
- RFC 1122 „Requirements for Internet Hosts – Communication Layers“ [RFC1122]
  - gute Dokumentation und Diskussion der Entwurfsentscheidungen

Postel-Prinzip

- Robustheitsprinzip (Jon Postel, siehe auch http://www.postel.org):
  "Be liberal in what you accept, and conservative in what you send" [RFC1122]
  - Software sollte auf jeden – auch wenn noch so unwahrscheinlichen – Fehler angemessen reagieren können
  - Eingehendes Paket kann beliebige Kombination von Attributen sowie Fehler enthalten
  - Annahme mutwilliger/böswilliger Erzeugung solcher Pakete

2.4 Neuere Entwicklungen (1)

- Viele Aspekte haben sich seit Anbeginn des Internets geändert
- „Bedrohungen" für das Ende-zu-Ende-Argument?
  - Vertrauensverlust zwischen Endsystemen [RFC3724]
    → Einführung von Sicherheitstechniken
  - Middleboxes (Proxies/NATs/Firewalls/Caches/...)
    → Bruch des Ende-zu-Ende-Prinzips (insbesondere bzgl. Sicherheitsmechanismen)



  - Neue Dienstmodelle: Dienstgüte wird Bestandteil des Dienstes (Streaming A/V) → Server werden verteilt und näher zum Nutzer platziert (z.B. Akamai, Realnetworks...)

Neuere Entwicklungen (2)

- Beispiel: negative Effekte durch Sicherheitstechniken
  - Rigoroses Filtern von ICMP-Paketen: kein PATH-MTU-Discovery
  - Filtern von Paketen mit gesetzten ToS-Bits: verhindert Explicit Congestion Notification
  - Private Adressierung in „Intranets“: Einschränkung der Erreichbarkeit und verfügbarer Dienste
- Mögliches Vorgehen für zukünftige Mechanismen, die scheinbar gegen das Ende-zu-Ende-Prinzip verstoßen: Zerlege E2E-Argument in Bestandteile
  - Innovationsschutz
    - Einführen neuer Mechanismen in Endsystemen einfacher
  - Zuverlässigkeit/Robustheit und Vertrauen
    - Sicherheit hinzufügen, wo nötig

Beispiel: Open Pluggable Edge Services (OPES)

Idee: Dienste im Netz können nutzerspezifisch gesteuert werden, z.B. Nutzerabhängige Transformation von Inhalten, Zusammenstellen benutzerspezifischer Web-Seiten (z.B. regionalisiert), Virenschutz, Sprachübersetzungen

Diskussion [RFC 3238]

- Steht dem Ende-zu-Ende-Prinzip teilweise entgegen
  - Wirft Fragen bezüglich Integrität, Privatsphäre und Sicherheit auf
  - Mögliches Fehlverhalten eines Dienstes (z.B. falsche Positive eines Virenscanners)
  - Kompromittierung eines OPES-Zwischensystems
  - Überwiegen die Vorteile die architekturellen Nachteile?

Forderungen

- Erhalten der Robustheit des Ende-zu-Ende-Prinzips
- Schutz der Ende-zu-Ende-Datenintegrität: Detektion von und Reaktion auf Fehlverhalten der OPES-Zwischensysteme durch Endsysteme
- OPES-Dienst muss mindestens einseitig autorisiert sein

Verlust der Internet-Transparenz

- Internet-Transparenz: [RFC2775]
  - ursprüngliches Konzept eines einzigen universellen, logischen Adressierungsschemas
  - Mechanismen, durch die Pakete im Wesentlichen unverändert von Quelle zu Ziel fließen
- Verlust der Transparenz durch:
  - Intranets („Sicherheit“, Einschränkung der Anwendungen und Adresstransparenz, Netzadministrator hat Kontrolle)
  - Private Adressen (nicht eindeutig, Einschränkung der Erreichbarkeit und globalen Kommunikation)
  - Freiwillige Isolation (z.B. WAP-Proxies) und Partner-Netzwerke
  - Middleboxes:
    - Firewalls (Einschränkung Dienste und Erreichbarkeit)
    - Network Address Translators (NATs)
    - Application Level Gateways, Proxies, Caches
  - Dynamische Adressen (SLIP/PPP, DHCP)
  - Split-DNS
  - Tricks zum Lastausgleich

Split-DNS: DNS antwortet nach intern anders als nach extern.
Beispiel: private Adressen im internen DNS funktionieren zwar, aber die Secondaries, die außerhalb liegen, können diese Adresse nicht kennen (private Adresszone kann nicht transferiert werden, da nicht eindeutig). Daher kommen dann Effekte zustande, dass der Name einmal auflöst und ein andernmal nicht, nämlich dann, wenn bei den externen Secondaries angefragt wird.

Tussle in Cyberspace [CWSR02]

- Früher: gemeinsames Ziel der Internetgemeinde
- Heute: Kampf um Interessen verschiedener Parteien
  - Musiktauschbörsen ↔ Musikindustrie/Rechteinhaber
  - Private Konversation ↔ Abhörmöglichkeiten für Regierungen
  - ISP Interconnection ↔ ISP-Konkurrenz
  - Nutzer in Regierungs- und Firmennetzen durch Firewalls abgeschottet → Nutzer suchen Auswege (Tunneln, externe Zugänge, andere Routen, ...)
  - ISPs weisen nur eine öffentliche IP-Adresse zu → Nutzer schließen ganze Netzwerke darüber an

Beispiele für verschiedene Parteien: Nutzer, kommerzielle ISPs, Private Netzbetreiber, Regierungen, Patentinhaber, Inhalteanbieter und Dienstbetreiber,...

Tussle in Cyberspace: Neue Prinzipien

- Systemdesign sollte Spielraum bzgl. Nutzungs-möglichkeiten erlauben
  - unterschiedliche Möglichkeiten je nach Interessen
  - Konflikte sollten innerhalb des Systemdesigns ausgetragen werden und dieses nicht verdrehen oder verletzen
- Neue Prinzipien
  - Modularität entlang von Konfliktgrenzen
    - → Konflikte können keine Auswirkungen auf andere Bereiche haben
  - Berücksichtigung von Wahlmöglichkeiten innerhalb des Designs
    - → Je nach Bedürfnis der Parteien
    - Beispiel: Wahlmöglichkeit für die Interaktionspartner (z.B. Konfiguration der bevorzugten SMTP-, POP-, News-Server), aber: schlecht für naive Nutzer

Tussle in Cyberspace: Aspekte der Konflikte

- Lösung erfordert offene Schnittstellen (ermöglicht auch Wettbewerb)
- Sichtbarkeit der Wahlentscheidung (direkt oder indirekt über die Auswirkungen)
- Gegensätzliche/sich ergänzende Interessenskonflikte: Synergie: Vergütung von Diensten (Geld o. andere Werte)
- Konflikte dauern an, entwickeln und verändern sich
- Prinzip:
  Der Entwurf von Erweiterungen sollte eine Analyse beinhalten, welche Interessenskonflikte auftreten können und wie mit diesen umgegangen werden kann. Oftmals kann Wettbewerb richtungsweisend sein.

- Konflikte werden oftmals über Schnittstellen ausgetragen (z.B. BGP für ISPs zum regeln der Verkehrs- und damit auch der Geldflüsse)

Net Neutrality – Netzneutralität

- Net Neutrality beschreibt Interessenskonflikt

[Crow07]

- ISPs/Netzbetreiber transportieren Verkehr, mit dem andere sehr viel Geld verdienen
- ISPs wollen auch daran partizipieren
  - Idee: Google-, E-Bay-, Amazon-, YouTube-Inhalte nur noch über ISP erreichbar, wenn dieser von den Anbietern entsprechend bezahlt wird.

Net Neutrality – Folgen

- Grundsatzfrage:
  Dürfen ISPs/Netzbetreiber Verkehr, den sie transportieren, beeinflussen?
  - Beispielsweise mittels Priorisierung, Filterung von Verkehr (z.B. kein VoIP-Verkehr via UMTS, kein P2P-Verkehr mehr)?
  - Probleme:
    - Bestimmte Inhalte dann nur noch über bestimmte Netzbetreiber (gut) zugänglich (YouTube-Zugang wird mit DSL-Anschluss von Betreiber X gekoppelt)
    - Neue Anwendungen und Dienste werden gehindert
    - Kleine Inhalteanbieter haben das Nachsehen
- Ziel der „Net Neutrality“-Bewegung
  - Erhaltung der Netztransparenz: Alle Pakete, Inhalte und Dienste müssen gleichberechtigt im Internet transportiert werden. Keine Diskriminierung!
  - Kein Blockieren
    - von legalen Inhalten und Anwendungen
    - rein aus Netzbetreiberinteressen

Net Neutrality – Lösungen

- Regulierung durch Gesetzgeber?
- Eingriffe seitens des Netzbetreibers
  - sinnvoll wenn es z.B. um Sicherheit geht
  - wenn es um intelligentes Ressourcenmanagement geht → Dienstgüteunterstützung
    - aber wenn möglich: nicht-diskriminierend gegenüber bestimmten oder einzelnen Anwendungen
  - sollten offengelegt und nicht willkürlich sein
- Kontrolle sollte auf Nutzerseite liegen
  - Vorgabe, wie bestimmte Daten behandelt werden sollen, z.B. Priorisierung oder explizite Dienstgütesignalisierung

Routing und Adressierung

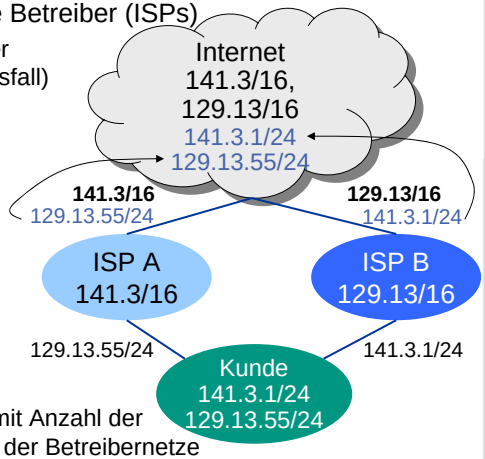
- Routingtabellen wachsen zu stark
  - Site-Multi-Homing [RFC4984]
  - Traffic-Engineering
  - Provider-unabhängige Adressen (nicht aggregierbar)
- Architektur nicht auf Bedürfnisse ausgerichtet
  - Endkunden wollen
    - Multi-Homing
    - Kein Umnummerieren von IP-Adressen bei Providerwechsel
    - Provider-unabhängige Adressen
  - Provider wollen
    - Kontrolle über Datenverkehrsfluss
    - Kleine Routingtabellen

- Site-Multi-Homing stört die Aggregierung von IP-Addressen und vergrößert so die Routing-Tabellen
- Traffic-Engineering wird aus den folgenden Gründen betrieben
  - Verwendung von speziellen Routen zum Load-balancing (bessere Ressourcennutzung durch gleichmäßigere Auslastung)
  - Kostenreduzierung durch billigere Provider
  - politische Entscheidungen, dass Verkehr eines Landes nicht durch ein anderes Land laufen darf

Site Multi-Homing

- Abhängigkeit vom Internet wird zunehmend größer
  → zuverlässige Konnektivität wichtig
- Internet-Anbindung über mehrere Betreiber (ISPs)
  - Verbesserte Toleranz gegenüber Ausfällen (einschließlich ISP-Ausfall)
  - Lastverteilung
- Situation heute:
  - Kundennetz hat zwei oder mehr Netzadressen

  oder
  - Kunde hat Betreiber-unabhängige Netzadresse
- Problem:
  - Routen-Aggregation unwirksam
  - Globale Routingtabelle skaliert mit Anzahl der Kundennetze anstatt mit Anzahl der Betreibernetze



Site: Unternehmen, das eigenes IP-Netz verwaltet

Site Multi-Homing bezeichnet die Anbindung des eigenen Netzes über mehrere Provider (im Gegensatz zu Host Multi-Homing)

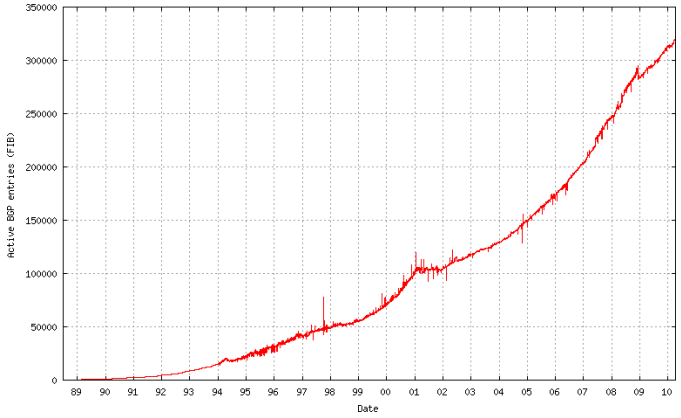
Für die heutigen Ansätze gilt: Routen können nicht mehr aggregiert werden → Routingtabellen wachsen zu stark…

Mehrere Routen notwendig, um Ausfall zu kompensieren

Eine Route je Multi-homed Site skaliert nicht

Routingtabellenwachstum

- Folge: starkes Wachstum der Routingtabellen [RFC4984]
- Prognose: Leistungssteigerung durch Hardware reicht nicht zur Kompensation



Unter der Annahme, dass IPv6 ähnlich viele Einträge aufweist und Multi-Homing und Traffic-Engineering genauso gelöst werden wie unter IPV4, ergeben sich Skalierbarkeitsprobleme.

Die Granularität des Inter-Domain Routingsystems wird feiner und die Anzahl der Verbindungen zwischen den Domänen wird größer, d.h. die Vermaschung nimmt zu. Das Internet wächst also nicht „am Rand“ sondern neue Netze kommen im Transitbereich hinzu, welche die Anzahl der Peerings und Verbindungen erhöhen. Quelle: http://www.potaroo.net/ispcol/2009-03/bgp2008.html

Identifikator/Lokator-Problematik

- Im Internet beinhaltet die IP-Adresse derzeit zweierlei:
  - Identifikation des Endsystems (Identifikator) als auch die
    - globaler Adressraum
  - Lokation des Endsystems (Lokator)
    - Adresse beinhaltet topologische Information für das Routing
- Diese Eigenschaft erschwert u.a.
  - Mobilität des Knotens
  - Multi-Homing
- Globale Adressierung überhaupt notwendig?
  - oftmals haben Netze nur beschränkten Geltungsbereich
- Ansonsten Abbildung ID → Lokator erforderlich
  - Dynamik und Komplexität im Mapping-System?
  - Aktuelle Arbeiten in der IRTF Routing Research Group

Ansatz LISP [FFLM09]

- Möglichst wenig Veränderungen, keine Änderungen in den Endsystemen
- Trennung von Endpoint Identifier (EID) und Routing Locator (RLOC)
  - EID auch als Präfix organisiert, entspricht praktisch PI-Präfix
  - RLOC: heutige PA-Präfixe, aggregierbar
  - LISP: Locator/ID Separation Protocol
- Abbildungsfunktion EID-Präfix → RLOC erforderlich
- Endsystem schickt Datenpaket an Ziel-EID
  - Ingress Tunnel Router fügt Tunnel-Header mit RLOCs hinzu (wenn Abbildung bekannt)
  - Egress Tunnel Router entfernt Tunnel-Header wieder
- Probleme
  - spiegelt die Abbildung die aktuelle Erreichbarkeit wider?
  - Problem in das Mapping-System verschoben?

PI: Provider Independent
PA: Provider Aggregatable
EIDs können innerhalb der zuständigen Domäne geroutet werden
Im ITR (Ingress Tunnel Router): Wenn Abbildung EID → RLOC unbekannt, muss ein Lookup den RLOC zum EID-Präfix besorgen, z.B. wird das über eine Dynamische Hash Tabelle (s. Kap. 10) ermittelt.
Anschließend kann das Ergebnis in einem Cache gehalten und für nachfolgende Pakete verwendet werden.
Für Multi-Homed Systeme gibt es dann mehrere RLOCs, die hinterlegt sind.

Content Centric Networking

- Paradigmenwechsel [JSTP+09]
  - bisher Kommunikation = Konversation zwischen zwei Endpunkten

  - geänderte Nutzung durch das Web
  - jetzt: Konzentration auf Inhalte
    - Interesse an bestimmten Inhalt → egal wie Inhalt transportiert wird
    - Integrität und Authentizität der Daten wichtig! → anderes Sicherheitskonzept
- Einbeziehung von Speichermedien
  - Transport über die Zeit
  - ermöglicht asynchrone Nutzung, Caching usw.
- Namensstruktur hierarchisch
  - z.B. /parc.com/people/van/presentations/FISS09

Staukontrolle

- Derzeitige Verfahren nicht gut für hohe Geschwindigkeiten oder hohe Latenzen geeignet
  - AIMD zu konservativ für hohe Geschwindigkeiten
  - hohe Heterogenität: langsame und schnelle Links, kleine und große Verzögerung → Skalierbarkeit
- Stabilität?
  - Konvergenz des Verfahrens garantiert?
- Fairness?
  - Flow Rate Fairness
  - Cost Fairness?
- Netzunterstützung?

Trend: Netzvirtualisierung

- Ändern der Infrastruktur aufwändig
  - Ausbringen neuer Protokolle und Mechanismen schwierig (IPv6, Multicast, ECN, usw.)
- Virtuelle Netze: virtuelle Knoten und virtuelle Links
  - mehrere virtuelle Knoten innerhalb eines physikalischen
  - Isolation der virtuellen Netze
- Vorteile
  - Parallele Existenz unterschiedlicher Architekturen
  - Netztopologie kann einfacher geändert werden
  - bessere Ressourcenauslastung/-nutzung durch Multiplexing und Migration
- Forschungsthema in 4WARD und G-Lab

- Viele Parallelen zur Server-Virtualisierung

2.5 Forschungsbedarf im Internet

- Das Internet steht ständig vor neuen Herausforderungen
- Beispiele für Forschungsthemen:
  - Namensgebung
  - Routing
  - Sicherheit
  - Netzwerkmanagement
  - Dienstgüte (QoS)
  - Staukontrolle (für Hochgeschwindigkeitsnetze)
  - Middleboxes
  - Internet-Entwicklung

- Namensgebung: DNS (vor allem Sicherheit), Integration neuer Namensräume
- Routing
  - Inter-Domain-Routing (Skalierbarkeit, Stabilität, Multi-Homing, ...)
  - Integrität und Authentisierung der Routingdaten
  - Neue Routing-Algorithmen
  - Mobile & Ad-Hoc-Routing (u.a. Netzwerkmobilität)
- Sicherheit
  - Formale Methoden für Internet-Sicherheit (z.B. neue Sicherheits- Vertrauensmodelle für Ad-Hoc-Netze etc.), Werkzeuge zur Überprüfung
  - Key-Management (nicht-hierarchisch, für Ad-Hoc-Netze, Multicast)
  - Kryptographie
  - Sicherheit für verteiltes Rechnen
  - Sicherheit im praktischen Einsatz/Gebrauch (perfekte Sicherheit bringt nichts, wenn nicht praktisch umsetzbar)
- Netzwerkmanagement
  - Skalierbares Konfigurationsmanagement für eine große Geräteanzahl
  - Monitoring für viele Geräte oder ganze Netzwerke
  - Management für Netze, nicht nur für einzelne Geräte
- Dienstgüte (QoS)
  - Inter-Domain QoS-Architektur, Warteschlangenmechanismen und Bedienstrategien
- Staukontrolle
  - für neue Anwendungen, Media Streaming und Multicast
  - für drahtlose Netze und für Hochgeschwindigkeitspfade
  - Wechselwirkungen mit Warteschlangenmechanismen in Routern
  - Aggregierte Staus durch verteilte Angriffe oder sog. "Flash Crowds"

FIND – Future Internet Network Design

- Initiative der NSF http://find.isi.edu/
- Zwei Hauptfragestellungen
  - Anforderungen an ein globales Netz in 15 Jahren? Wie sollte ein solches Netz aussehen, was sollte es tun?
  - Wie würde das zukünftige Netz vom heutigen Standpunkt aus konzipiert werden, wenn es komplett neu entwickelt würde?
- Anforderungen
  - Sicherheit und Robustheit
  - Einfacher zu managen
  - Nicht-technische Aspekte berücksichtigen
- Fokus auf Steuerung, Management, usw.

GENI – Global Environment for Network Innovations

- Testeinrichtung für neue Netzwerkarchitekturen
  - Generisch, d.h. nicht auf spezielle Protokolle zugeschnitten, wie sonst bei Testbeds üblich
  - Erlaubt Ausprobieren neuer Netzarchitekturen
- Ziele
  - Erhöhung der Qualität und Quantität experimenteller Forschungsergebnisse in Netzen und verteilten Systemen
  - Beschleunigter Übergang von solchen Ergebnissen zu Produkten
  - Übergangsmöglichkeit zu neuem Netz, welches möglicherweise das Internet ablöst
  - Details: http://geni.net/

- Neue Netzarchitekturen sollten beispielsweise u.a. vorsehen:
  - Eingebaute Sicherheit und Robustheit
  - Einfacher Betrieb und einfache Nutzung
  - Pervasive Computing
  - Steuerung und Management anderer kritischer Infrastrukturen

Weiterentwicklung im Internet: IRTF und IETF

- Internet Research Task Force www.irtf.org
  - Betrachtet längerfristige Entwicklungen
  - Führt Voruntersuchungen durch
  - Eher geschlossene Gruppen, aber offene Mailinglisten
- Internet Engineering Task Force
  - Behebung aktueller Probleme
  - Zeithorizont: Protokollentwicklung in 2–3 Jahren
  - Offene Standardisierung der heutigen Internetprotokolle

Weiterentwicklungen im Internet: die IETF

- Internet Engineering Task Force (IETF) www.ietf.org, edu.ietf.org [RFC3233]
- Ziel: Das Internet besser machen [RFC3935]
- Mission: Produce high quality, relevant technical and engineering documents that influence the way people design, use, and manage the Internet in such a way as to make the Internet work better.
- Behebung aktueller Probleme, Schaffen notwendiger Erweiterungen etc., kurzfristiger umsetzbare Lösungen
- Erstellt gültige Internet-Standards (Request for Comments, RFCs)

- RFCs
  - Protokollstandards sind im sogenannten Standards-Track. Drei Abstufungen: Proposed Std, Draft Std, Full Std
  - Weitere Dokumenttypen (keine Standards im eigentlichen Sinn): Informational, Best Common Practice, Experimental
- Offene, internationale Entwicklergemeinde: „*Open global community of network designers, operators, vendors, and researchers producing technical specifications for the evolution of the Internet architecture and the smooth operation of the Internet.*“
- Credo: „*We reject kings and voting, we believe in rough consensus and running code*“ (D. Clark)
- Diskussion über „offene“ Mailinglisten, jeder Interessierte kann teilnehmen

Organisation:

- Verschiedene Bereiche (Areas): Applications, General , Internet, Operations and Management, Real-time Applications and Infrastructure, Routing, Security, Transport
- Jeder Bereich wird von zwei Area Directors (Ausnahme: General) geleitet
- Innerhalb der Bereiche: Arbeitsgruppen (Working Groups, derzeit 125 insgesamt)
- IESG (Internet Engineering Steering Group)
  - wird aus ADs gebildet
  - verabschiedet Standards (Reviewing)
- ISOC (Internet Society) bildet rechtliche Dachorganisation

2.6 Literatur (1)

[Clark88] D. Clark, „The Design Philosophy of the DARPA Internet Protocols“. Proc SIGCOMM 1988, Sept 1988. http://www.acm.org/sigcomm/ccr/archive/1995/jan95/ccr-9501-clark.html

[Crow07] Jon Crowcroft: “Net Neutrality: The Technical Side of the Debate: A White Paper”, ACM SIGCOMMM Computer Communications Review, Vol. 37, Number 1, Januar 2007

[CWSR02] D. Clark, J. Wroclawski, K. Sollins, R. Braden: „Tussle in Cyberspace: Defining Tomorrow‘s Internet“, ACM SIGCOMM 2002, http://www.acm.org/sigs/sigcomm/sigcomm2002/papers/tussle.pdf

[ESA96] „Flight 501 Failure Report“ http://ravel.esrin.esa.it/docs/esa-x-1819eng.pdf, 19. July, 1996

Literatur (2)

[FlJa94] S. Floyd, V. Jacobson, „The Synchronization of Periodic Routing Messages“, IEEE/ACM Transactions on Networking, Vol. 2, No. 2, April, 1994, http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=298431

[FFLM09] D. Farinacci, V. Fuller, D. Lewis, D. Meyer: Locator/ID Separation Protocol (LISP), draft-farinacci-lisp-12.txt, März 2009, http://tools.ietf.org/html/draft-farinacci-lisp

[Jaco88] V. Jacobson, „Congestion Avoidance and Control“, Proceedings of SIGCOMM 1988, pp. 273–288, http://portal.acm.org/citation.cfm?id=52324.52356

[JSTP+09] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, R. Braynard: “Networking Named Content”, ACM CoNEXT’09, December 1–4, 2009, Rome, Italy. http://conferences.sigcomm.org/co-next/2009/papers/Jacobson.pdf

[Neum90] P. G. Neumann, “Cause of AT&T network failure”, Januar 1990, http://catless.ncl.ac.uk/Risks/9.62.html#subj2

Literatur (3)

[RFC 1122] R. Braden. Requirements for Internet Hosts - Communication Layers. RFC 1122 (Standard), Oktober 1989. Updated by RFCs 1349, 4379. URL: http://www.ietf.org/rfc/rfc1122.txt.

[RFC 1958] B. Carpenter. Architectural Principles of the Internet. RFC 1958 (Informational), Juni 1996. Updated by RFC 3439. URL: http://www.ietf.org/rfc/rfc1958.txt.

[RFC 2775] B. Carpenter. Internet Transparency. RFC 2775 (Informational), Februar 2000. URL: http://www.ietf.org/rfc/rfc2775.txt.

[RFC 3233] P. Hoffman und S. Bradner. Defining the IETF. RFC 3233 (Best Current Practice), Februar 2002. URL: http://www.ietf.org/rfc/rfc3233.txt.

Literatur (4)

[RFC 3238] S. Floyd und L. Daigle. IAB Architectural and Policy Considerations for Open Pluggable Edge Services. RFC 3238 (Informational), Januar 2002. URL: http://www.ietf.org/rfc/rfc3238.txt.

[RFC 3426] S. Floyd. General Architectural and Policy Considerations. RFC 3426 (Informational), November 2002. URL: http://www.ietf.org/rfc/rfc3426.txt.

[RFC 3439] R. Bush und D. Meyer. Some Internet Architectural Guidelines and Philosophy. RFC 3439 (Informational), Dezember 2002. URL: http://www.ietf.org/rfc/rfc3439.txt.

[RFC 3724] J. Kempf, R. Austein und IAB. The Rise of the Middle and the Future of End-to-End: Reflections on the Evolution of the Internet Architecture. RFC 3724 (Informational), März 2004. URL: http://www.ietf.org/rfc/rfc3724.txt.

Literatur (5)

[RFC 4984] D. Meyer, L. Zhang und K. Fall. Report from the IAB Workshop on Routing and Addressing. RFC 4984 (Informational), September 2007. URL: http://www.ietf.org/rfc/rfc4984.txt.

[SaRC81] Saltzer, J., Reed, D., and D. Clark, End-To-End Arguments in System Design. 2nd International Conf on Dist Systems, Paris France, April 1981. ACM Transactions in Computer Systems 2, 4, November, 1984, pages 277–288. http://portal.acm.org/citation.cfm?id=357402

[Thal09] D. Thaler, Evolution of the IP Model, IETF Journal, Vol. 4, Issue 3, February 2009, http://www.isoc.org/tools/blogs/ietfjournal/wp-content/uploads/2009/02/IETFJournal0403.pdf

[WiDo02] W. Willinger, J. Doyle, „Robustness and the Internet: Design and evolution”, March 2002, http://netlab.caltech.edu/internet/