

**Zeitschrift:** Bulletin Electrosuisse  
**Herausgeber:** Electrosuisse, Verband für Elektro-, Energie- und Informationstechnik  
**Band:** 98 (2007)  
**Heft:** 21

**Artikel:** Peer-to-Peer-Netze  
**Autor:** Braun, Torsten / Brogle, Marc / Lauer, Patrick  
**DOI:** <https://doi.org/10.5169/seals-857494>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

**Download PDF:** 01.04.2025

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Peer-to-Peer-Netze

## Informationen effizient im Internet verbreiten

Wenn Millionen von Computern denselben Dienst bei einem Server im Internet anfragen, zum Beispiel einen Film oder eine Datei herunterladen wollen, muss der Server entweder sehr leistungsfähig sein, oder er wird überlastet, was eine Verzögerung oder einen Ausfall des Dienstes zur Folge haben kann. Effizienter werden Informationen über Peer-to-Peer-Netze im Internet verteilt. Hier werden die Dateien nicht mehr zentral, sondern verteilt bei den Peers gespeichert. Jeder Peer, der im Netz teilnimmt, muss also etwas Speicherplatz und Bandbreite zur Verfügung stellen, kann dafür aber auf alle Informationen im Peer-to-Peer-Netz zugreifen – ohne Engpässe.

Nach der Einführung des World Wide Web (WWW) zu Beginn der 90er-Jahre hat kaum eine Technologie das Internet verändert wie die der Peer-to-Peer-Netze, die sich seit Ende der 90er-Jahre mit dem Aufkommen der Audiotauschbörse Napster

Torsten Braun, Marc Brogle,  
Patrick Lauer

rasant entwickelten. Heute wird der Großteil des Internetverkehrs durch Peer-to-Peer-Anwendungen verursacht. Mit dem Begriff Peer-to-Peer (P2P) verbindet man meist das (illegale) Verteilen von Musikstücken über das Internet. Allerdings ist dies nur eine von vielen möglichen Anwendungen. Eine weitere populäre Anwendung der Peer-to-Peer-Technik ist die Internettelefonie-Software Skype, mit der man seit 2004

kostenlos über das Internet telefoniert. In der jüngsten Vergangenheit entwickeln sich zudem TV-Verteildienste, die im Vergleich zu herkömmlichen terrestrischen oder satellitengebundenen Verteildiensten das Empfangen von TV-Sendern über Computer und Internet ermöglichen.

### Client/Server und Peer-to-Peer

Grundsätzlich bedeutet Peer-to-Peer die Kommunikation zwischen gleichberechtigten Partnern. Peer bedeutet im Englischen «gleichgestellt, ebenbürtig». Im Gegensatz dazu basiert das World Wide Web auf einem Client/Server-Prinzip. Ein Server bietet einen Dienst an, beispielsweise eine Webseite oder eine Datei, der von Client-Systemen abgerufen wird. In diesem Fall sind Client und Server nicht ebenbürtig, da der Dienst exklusiv durch den Server angeboten wird und der Client den Dienst nur konsumiert, aber keine solchen Dienste anbietet (Bild 1). Bei Peer-to-Peer-Netzen können nun Computer (Endsysteme) gleichzeitig eine Client- und Server-Rolle übernehmen. Endsysteme agieren als Client, um eine Datei von einem anderen Endsystem abzurufen, bzw. als Server, um Dateien anderen Endsystemen anzubieten. Das Peer-to-Peer-Prinzip ist daher vor allem für die Kommunikation zwischen gleichartigen Systemen wie Personalcomputern geeignet, während das Client/Server-Prinzip eher leistungsfähigere, bei Diensteanbietern oder Internet Providern platzierte Serversysteme voraussetzt.

Zudem basieren Peer-to-Peer-Systeme meist auf dem Teilen von Ressourcen.

Jedes Peer-System bringt gewisse Ressourcen in einen Pool ein. Die gesamten Ressourcen werden dann zwischen den einzelnen Peers aufgeteilt, sodass Peers die von anderen Peers eingebrachten Ressourcen mitbenutzen können und diesen wiederum selbst eingebrachte Ressourcen anbieten. Dabei handelt es sich meist um Speicherplatz, aber auch Rechenzeit oder Kommunikationsbandbreite können Ressourcen darstellen.

### Zentralisierte Peer-to-Peer-Netze

Die ersten Peer-to-Peer-Ansätze wie Napster [1] verfolgten einen zentralisierten Ansatz mit einem zentralen Indexverzeichnis, das die Ressourcen (z.B. Dateien, identifiziert über deren Dateinamen) auf die IP-Adressen der Computer abbildet, auf denen die Dateien gespeichert sind. Der Indexserver speichert im Fall von Napster eine Liste von Dateien mit den dazugehörigen IP-Adressen der speichernden Computer (Indexeinträge, siehe Bild 2). Ein Peer, der eine Datei sucht, wendet sich an den Indexserver, der ihm für die gesuchte Datei eine IP-Adresse zurückliefert. Damit kann der suchende Peer den die Datei speichernden

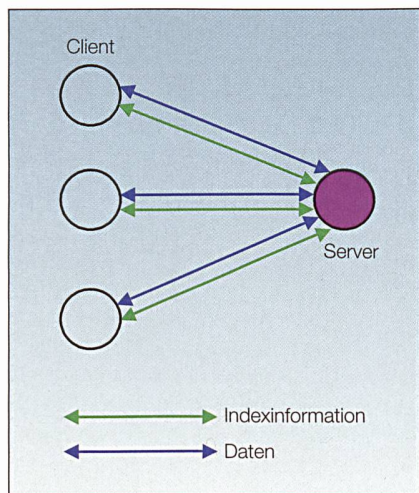


Bild 1 Client/Server-Kommunikation.

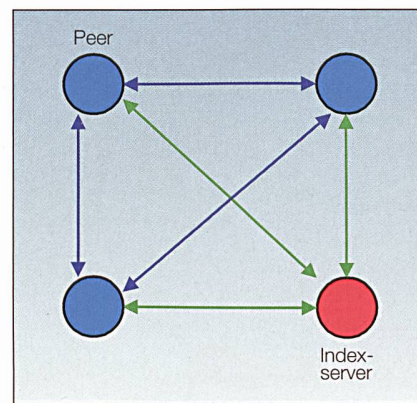


Bild 2 Zentralisiertes Peer-to-Peer-Netz.

Peer kontaktieren und von diesem die Datei abrufen, beispielsweise über HTTP<sup>1)</sup>. Die Existenz des zentralen Indexservers macht den Ansatz jedoch verwundbar, da beim Ausfall des Indexservers das gesamte Peer-to-Peer-Netz nicht mehr funktioniert. Daher sind verteilte Ansätze, bei denen die Indexinformationen ebenfalls verteilt werden, aus Zuverlässigkeitsgründen zu bevorzugen.



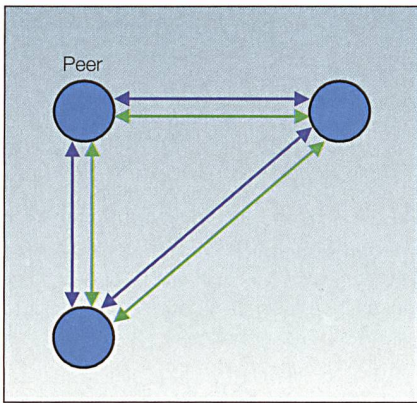


Bild 3 Verteiltes Peer-to-Peer-Netz.

### Verteilte Peer-to-Peer Netze

Bei den verteilten Ansätzen müssen die Indexeinträge geschickt verteilt und vor allem von den anfragenden Peers in einer effizienten Art und Weise gesucht werden (Bild 3). Verteilte Peer-to-Peer-Netze werden weiter in unstrukturierte und strukturierte Peer-to-Peer-Netze unterteilt.

### Unstrukturierte Peer-to-Peer-Netze

Bei den unstrukturierten Peer-to-Peer-Netzen schickt ein suchender Peer einfach eine Anfrage ins Netz. Die Anfrage wird dann solange im Netz verteilt, bis ein Peer einen der Anfrage entsprechenden Indexeintrag besitzt. Die einzelnen Peers sind teilweise miteinander vermascht, d.h., sie etablieren zu einer kleinen Anzahl anderer Peers Kommunikationsbeziehungen, die meist auf TCP<sup>2)</sup>-Verbindungen oder auf UDP<sup>3)</sup>-Flüssen basieren.

Eine einfache, aber ineffiziente Möglichkeit, Anfragen zu verteilen, besteht in einem Flutmechanismus. Eine Anfrage wird mit einem Wert für die Lebenszeit versehen und an alle Nachbarn geschickt. Die Nachbarn dekrementieren die Lebenszeit und schicken die Nachricht weiter an ihre Nachbarn. Sobald die Lebenszeit der Nachricht abgelaufen ist oder dieselbe Anfrage schon einmal weitergeleitet wurde, wird sie gelöscht. So werden unendlich lang kreisende Anfragen vermieden. Eine andere Möglichkeit, die Anfrage zu verteilen, besteht darin, die Nachrich-

ten über einen oder wenige andere Nachbarn in einer zufälligen Weise weiterzuleiten.

Der Vorteil von unstrukturierten Peer-to-Peer-Netzen liegt darin, dass die Anfragen nahezu beliebig gestaltet und verarbeitet werden können. Dies erlaubt zum Beispiel die Formulierung von komplexeren logischen Suchausdrücken oder Schlüsselwortsuchen. Dateien können zusammen mit beschreibenden Zusatzinformationen (Metadaten, z.B. Autor, Erstellungsdatum, Schlüsselworte, Musikrichtung etc.) gespeichert werden. Ein Peer, der die Anfrage verarbeitet, analysiert entsprechend die Metadaten. Bei einer Übereinstimmung von Suchausdruck und Metadaten sendet er dem suchenden Peer (direkt oder indirekt über die weiterleitenden Peers) eine Antwort mit den passenden Indexeinträgen. Der anfragende Peer kann daraufhin die gewünschte Datei beim gefundenen Peer anfordern und herunterladen.

Das wesentliche Problem unstrukturierter Ansätze ist der grosse Kommunikationsaufwand beim Suchen von Ressourcen. Eine Möglichkeit, diesen Aufwand zu reduzieren, besteht in der Einführung spezieller Peers, die oft als Super Peers bezeichnet werden. Diese Super Peers sind typischerweise leistungsfähige Computer mit hoher Rechen-, Speicher- und Kommunikationskapazität. Normale Peers werden einem Super Peer zugeordnet, dem sie Informationen über die gespeicherten Dateien mitteilen. Die Super Peers verbinden sich untereinander auf einer höheren Hierarchiestufe (Bild 4). Sucht ein normaler Peer eine Datei, leitet er zunächst die Anfrage an seinen zugeordneten Super Peer, der dann im Netz der Super Peers nach der gewünschten Datei sucht. Die Anfrage kann in der

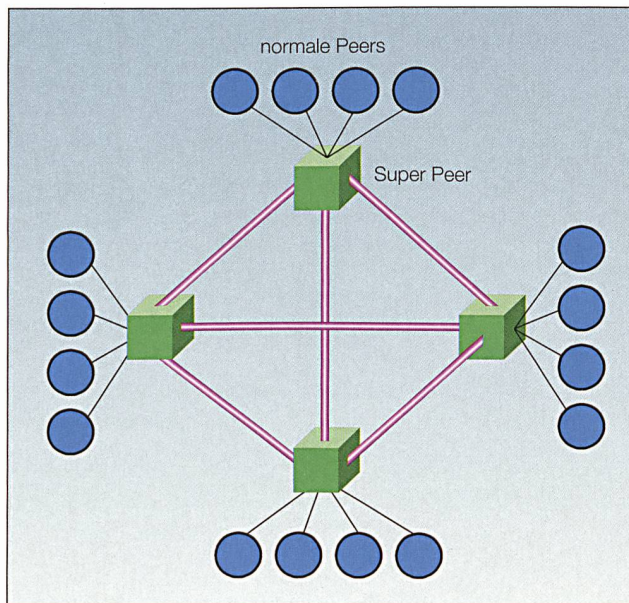


Bild 4 Super Peers.

Regel ohne den Einbezug normaler Peers bearbeitet werden. Die Last zum Suchen und Verarbeiten von Anfragen wird auf die leistungsfähigen Super Peers beschränkt. Etlliche Peer-to-Peer-Systeme in der Praxis, wie FastTrack/Kazaa [2], neuere Versionen von Gnutella [3] oder Skype [4], basieren auf diesem Prinzip. Allerdings ist ein solches Netz abhängig von den Super Peers, und deren Ausfälle können – wie bei der massiven Störung im Skype-Netz am 16. August 2007 – schwerwiegende Folgen haben.

### Strukturierte Peer-to-Peer-Netze

Das Problem der eingeschränkten Skalierbarkeit wird durch strukturierte Peer-to-Peer-Netze adressiert. Peers und Ressourcen haben üblicherweise einen eindeutigen Namen, der durch identische Hash-Funktionen auf eine Position in einer bestimmten Struktur abgebildet wird. Ein Beispiel für eine solche Struktur ist ein Ring, auf dem alle möglichen Werte des Bildbereichs der Hash-Funktion (Identifikatoren) der Reihe nach angeordnet sind. Da jeder Peer die gleiche Hash-Funktion verwendet, spricht man auch von verteilten Hash-Tabellen (Distributed Hash Table, DHT). Mehrdimensionale Hash-Funktionen resultieren in einem mehrdimensionalen Raum. Jeder Name wird auf eine bestimmte Position in diesem Raum abgebildet. So nimmt jeder Peer eine gewisse Position in diesem Ring oder Raum ein und erhält einen entsprechenden Identifikator. Die Peers etablieren dann Kommunikationsbeziehungen zu anderen Peers oder speichern zumindest Informationen zu ihrer Erreichbarkeit, beispielsweise IP-Adressen. Ein Peer sammelt also insbesondere über die Nachbarn auf dem Ring oder im mehrdimensionalen Raum Informationen. Über weiter entfernt liegende Peers werden weniger Informationen gespeichert. Bild 5 zeigt einen Ring, wie er bei Pastry [5] verwendet wird. Ein Peer speichert hauptsächlich Informationen (IP-Adressen) über Peers in seiner Nähe, er kennt aber auch einige weiter entfernt liegende Peers.

In derselben Art wie die Namen der Peers werden die Namen der Ressourcen (z.B. Dateinamen) per Hash-Funktion auf einen Identifikator abgebildet. Will ein Peer nun eine Datei ins Netz einbringen, bildet er den Dateinamen per Hash-Funktion auf eine Position bzw. einen Identifikator ab. Die Datei oder die Metadaten zum Auffinden der Datei werden auf jenem Peer gespeichert, der dieser sich aus dem Dateinamen ergebenden Position am nächsten ist. Dazu sendet der die Datei einbringende Peer eine Nachricht zum speichernden Peer. Ein nach einer Datei suchender Peer



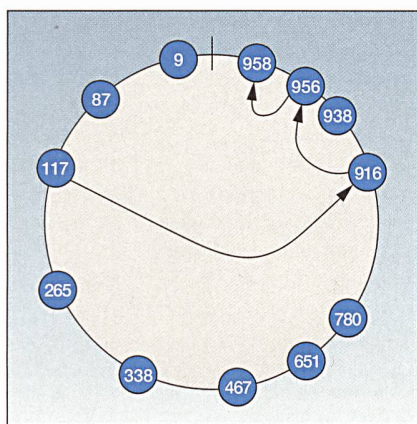


Bild 5 Pastry.

wendet dieselbe Hash-Funktion auf den Dateinamen an, erhält dieselbe Position wie der einfügende Peer und schickt eine Anfrage zur Position des am nächsten liegenden Peers, der die Datei oder deren Metadaten gespeichert haben sollte. Der speichernde Peer kann daraufhin die Datei direkt übertragen oder Informationen zur Verfügung stellen, wo diese heruntergeladen werden kann.

Eine noch offene Frage ist, wie einfügende und suchende Peers mit dem speichernden Peer kommunizieren. Zwar erhalten einfügende und suchende Peers aus der Hash-Funktion die Position bzw. den Identifikator des speichernden Peers. Allerdings kennen sie nicht dessen IP-Adresse und können daher nicht direkt mit ihm kommunizieren. Typischerweise kennen sie aber andere Peers mit Identifikatoren, die näher beim speichernden Peer liegen. Nachrichten zum Speichern oder Suchen von Dateien werden dann an die einem Peer bekannten Peers weitergeleitet, die bezüglich der Identifikatoren so nahe wie möglich beim speichernden Peer liegen. Dieses Weiterleiten wird so lange wiederholt, bis der speichernde Peer erreicht und ermittelt ist. In Bild 5 sendet Peer 117 eine Nachricht an Peer 916, der diese dann über Peer 956 an den speichernden Peer 958 weiterleitet. Die Datei oder die Metadaten werden dann auf Peer 958 gespeichert und können später von anderen Peers abgerufen werden, indem Anfragen (ggf. über andere Peers) an Peer 958 geschickt werden.

Vorteil des strukturierten Ansatzes ist die effiziente Suche. Allerdings sind die Suchmöglichkeiten eingeschränkt, da Anfragen auf festgelegte Namen, z.B. Dateinamen, limitiert sind. Suche nach Schlüsselwörtern oder komplexe Suchausdrücke sind nicht möglich, da diese nicht per Hash-Funktion auf eindeutige Identifikatoren abgebildet werden können. Auch ist es notwendig, bei häufigem Neueintreten und Austreten von

Peers aus dem Netz die Struktur des Netzwerkes anzupassen. Dies kann dazu führen, dass Daten von einem zum anderen Peer verlagert werden müssen.

## Multicast

Peer-to-Peer-Netze können auch zur Gruppenkommunikation (Multicast) eingesetzt werden. Multicast bedeutet, dass beispielsweise ein Videostrom gleichzeitig an mehrere bestimmte Computer im Internet geschickt wird, aber nicht generell an alle Computer, was einem Broadcast entsprechen würde. Das Application Level Multicast genannte Konzept setzt auf Peer-to-Peer-Netzen auf, indem sich Peers miteinander verbinden und dadurch ein Overlay-Netz etablieren. Zur Verteilung von Multicast-Daten bietet sich der Aufbau von Verteilbäumen an. Ein Wurzelknoten sendet die Multicast-Daten an seine Kinder-Knoten im Baum, die Kinder-Knoten verteilen die Daten entsprechend weiter. So setzt Scribe [6] auf Pastry auf und etabliert einen Verteilbaum. Dabei wird der Name der Gruppe auf einen Pastry-Identifikator abgebildet. Der diesem Identifikatorwert am nächsten liegende Peer wird zum Wurzelknoten des Verteilbaums. Über den oben beschriebenen Pastry-Mechanismus werden Nachrichten zum Beitreten zu der Gruppe an den Wurzelknoten gesendet. Alle dazwischenliegenden Knoten, die diese Nachrichten weiterleiten, werden in den Verteilbaum aufgenommen. Bild 6 zeigt einen Baum, nachdem in Bild 5 Peer 117 eine Nachricht zum Beitreten in die Gruppe über zwei Zwischenknoten an den Wurzelknoten 958 geschickt hat.

## Anwendungen

Die Anwendungen von Peer-to-Peer-Netzen sind mannigfaltig und nicht nur auf das (ggf. illegale) Verteilen von Musikstücken oder Videofilmen beschränkt. Grundsätzlich können mit Peer-to-Peer-Netzen in einer eleganten Art und Weise verteilte und damit robuste sowie skalierbare Datenbanken aufgebaut werden, um beliebige Daten verteilt zu speichern. Dies kann neben er-

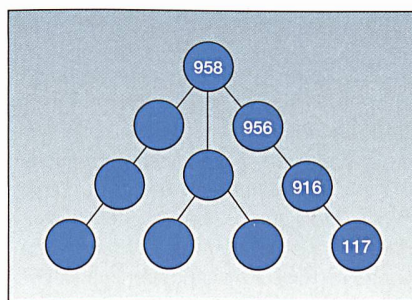


Bild 6 Scribe.

höhter Robustheit gegenüber Rechner- oder Netzausfällen auch zu Leistungssteigerungen durch parallelen Zugriff führen. BitTorrent [7] zerlegt dazu eine Datei in kleinere Teile, sodass ein Peer die einzelnen Teile von verschiedenen Peers gleichzeitig beziehen kann. Damit lässt sich die Rechen- und Netzlast zwischen Peers gleichmäßig verteilen.

Skype ermöglichte erstmals Internet-Telefonie mithilfe von Peer-to-Peer-Netzen. Skype ist inzwischen weit verbreitet (typischerweise fast 10 Millionen gleichzeitig aktive Benutzer) und bietet neben Audio/Video-Telefonie auch Chat- und Dateiübertragungsdienste an. Als Plattformen werden nicht nur herkömmliche Computer, sondern auch persönliche digitale Assistenten (PDAs) oder Mobiltelefone unterstützt. Skype benutzt allerdings ein proprietäres, bislang unveröffentlichtes Protokoll und ist damit nicht zu anderen Systemen kompatibel. Daher haben die Entwickler des Internetstandards SIP<sup>4)</sup> einen neuen, Peer-to-Peer-basierten Ansatz zur Signalisierung vorgeschlagen [8]. Mithilfe von Peer-to-Peer-Mechanismen sollen dabei die möglicherweise mobilen Endbenutzer lokalisiert werden. Dies bedeutet, dass anstatt für einen Dateinamen für einen Benutzernamen eine IP-Adresse des aktuellen Endsystems des Benutzers ermittelt werden muss.

In der jüngsten Vergangenheit werden Peer-to-Peer-Netze auch zunehmend für Videostreaming oder gar TV-Broadcast-Dienste vorgeschlagen. Zattoo [9] bietet in einigen Ländern, darunter die Schweiz, eine Auswahl an TV-Programmen an. Bei solchen Videoverteildiensten senden die Peers den empfangenen Datenstrom an einen oder mehrere andere Peers weiter. Dabei werden mithilfe des Peer-to-Peer-Netztes spezielle Verteilbäume aufgebaut. Die Rechnerlast wird damit auf viele Peers verteilt. Das Problem ist jedoch, dass typische Breitbandanschlüsse – zumindest in Europa – derzeit noch geringe Bandbreiten vom Endbenutzer zum Netz hin unterstützen. Damit können derzeit nur wenige Videostreams von einem Endsystem zu anderen Endsystemen gesendet werden (bei weniger als 1 Mbit/s Uplink-Bandbreite nur ein Videostrom mit geringer Qualität). Gerade wenn Internet-Service-Provider TV-Dienste über eigene Server anbieten, sind diese eher an hohen Bandbreiten in Richtung der Kunden als in die Gegenrichtung interessiert.

Ein Nachteil vieler Peer-to-Peer-basierten Videoverteildienste ist, dass proprietäre Protokolle eingesetzt werden. Dieser Nachteil wird zum Beispiel durch die im europäischen Forschungsprojekt EuQoS [10] entwickelte, sogenannte Multicast-Middleware [11] vermieden. Mithilfe der Multicast-



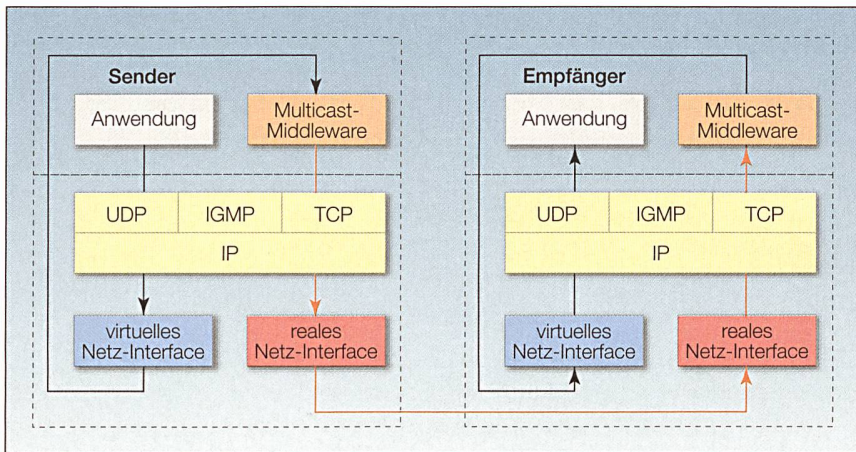


Bild 7 Multicast-Middleware.

Middleware kann jede beliebige IP-Multicast-kompatible Anwendung unterstützt werden. Die Multicast-Middleware greift dabei in einem Endsystem IP-Multicast-Pakete ab und verteilt sie über einen beliebigen Application-Level-Multicast-Mechanismus, z.B. Scribe/Pastry. Multicast-Nachrichten, die bei einem Peer über Application Level Multicast ankommen, werden an den Internet-Protokoll-Stack des Betriebssystems übergeben und über diesen an die IP-Multicast-Anwendung weitergeleitet (Bild 7).

**Sicherheit**

Häufig werden im Zusammenhang mit Peer-to-Peer-Netzen Sicherheitsbedenken geäußert. Es ist sicher richtig, dass durch Kommunikationsbeziehungen über Peer-to-Peer-Netze schädliche Software einfacher auf einen Computer übertragen werden kann. Allerdings können solche Angriffe durch E-Mails oder schadhafte Webseiten im gleichen Mass erfolgen. Häufig müssen Benutzer zur Teilnahme an Peer-to-Peer-Netzen auf Webseiten bereitgestellte Software herunterladen und auf ihrem Computer installieren, ohne zu wissen, was diese Programme wirklich tun. Hier bleibt den Benutzern nichts anderes übrig, als den Anbietern zu vertrauen – wie sie dies auch bei anderen kommerziellen Softwareanbietern tun. Es

sind auch Fälle bekannt, bei denen grosse Telekommunikationsanbieter virenverseuchte Software auf CDs ausgeliefert haben. Ansonsten wird wie bei anderen verteilten Applikationen durch die Anwendung von Standard-Kryptografie-Verfahren hohe Sicherheit bezüglich Manipulation oder Mit-hören von Nachrichten erzielt. Protokollanalysen haben beispielsweise Skype im Vergleich zur herkömmlichen Telefonie über ISDN oder analoge Telefonnetze einen hohen Sicherheitsstandard bestätigt [12]. Ein anderes Sicherheitsproblem besteht in sich nicht fair verhaltenden Peers, d.h. Peers, die versuchen, sich durch Fehlinformation oder Fehlverhalten Vorteile bezüglich der Ressourcennutzung zu verschaffen. Andere Peers sollten in solchen Fällen versuchen, Fehlverhalten und -informationen zu erkennen und ggf. solche Knoten zu bestrafen, beispielsweise durch Ausschluss aus dem Peer-to-Peer-Netz. Die Entwicklung geeigneter Verfahren stellt ein aktuelles Forschungsthema dar.

**Plattform für neue Dienste**

Peer-to-Peer-Netze bilden eine leistungsfähige und robuste Technik für diverse Kommunikationsdienste und Anwendungen. Sie erlauben insbesondere, neue Dienste unabhängig von traditionellen An-

biern und Netzbetreibern zu realisieren. Der Bedarf für zentrale Servereinrichtungen wird stark reduziert und beschränkt sich oft auf Sicherheitsfunktionen sowie Funktionen zur Initialisierung oder Softwareaktualisierung. Speicher-, Rechen- und Kommunikationsressourcen werden zwischen den beteiligten Anwendern geteilt, was zu einer hohen Skalierbarkeit führt.

**Referenzen**

- [1] M. Parameswaran, A. Susarla, A. B. Whinston: P2P Networking: An Information-Sharing Alternative. IEEE Computer, S. 31–38, July 2001.
- [2] N. Leibowitz, M. Ripeanu, A. Wierzbicki: Deconstructing the Kazaa Network. The Third IEEE Workshop on Internet Applications (WIAPP), S. 112–120, 2003.
- [3] G. Kan: Kapitel über Gnutella im Buch von A. Oram: Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology, S. 94–122, 2001.
- [4] Skype: Skype – Einfach reden. Abgerufen am 28. August 2007 von <http://www.skype.com/intl/de>.
- [5] A. Rowstron, P. Druschel: Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. IFIP/ACM Middleware, Heidelberg, 2001.
- [6] M. Castro, P. Druschel, A.-M. Kermarrec, A. Rowstron: Scribe: a large-scale and decentralized application-level multicast infrastructure. Selected Areas in Communications, IEEE Journal, S. 1489–1499, October 2002.
- [7] M. Izal et al.: Dissecting BitTorrent: Five months in a torrent's lifetime. 5th Passive and Active Measurement Workshop, 2004.
- [8] K. Singh, H. Schulzrinne: Peer-to-peer Internet Telephony using SIP. NOSSDAV, Skamania, 2005.
- [9] G. Santner: Peer-to-Peer-Fernsehen über das Internet. Bulletin SEV/VSE, S. 19–21, Mai 2007.
- [10] EuQoS: End-to-End Quality of Service over Heterogeneous Networks, Abgerufen am 27. August 2007 von [www.euqos.eu](http://www.euqos.eu).
- [11] M. Brogle, D. Milic, T. Braun: Supporting IP Multicast Streaming Using Overlay Networks. QShine, Vancouver, 2007.
- [12] S. L. Garfinkel: VoIP and Skype Security. Skype Security Overview, 2005.

**Angaben zu den Autoren**

Prof. Dr. **Torsten Braun** ist seit 1998 ordentlicher Professor am Institut für Informatik und angewandte Mathematik (IAM) der Universität Bern. Er leitet dort die Forschungsgruppe Rechnernetze und Verteilte Systeme und ist seit 2007 geschäftsführender Direktor des IAM. Seit dem Jahr 2000 ist er Mitglied des Stiftungsratsausschusses von Switch.

Universität Bern, IAM, 3012 Bern, [braun@iam.unibe.ch](mailto:braun@iam.unibe.ch)

**Marc Brogle**, dipl. Inf., ist seit September 2004 Doktorand und Assistent in der Forschungsgruppe Rechnernetze und Verteilte Systeme an der Universität Bern. Sein Forschungsgebiet umfasst unter anderem Quality of Service für Peer-to-Peer-Netze und Application Layer Multicast.

Universität Bern, IAM, 3012 Bern, [brogle@iam.unibe](mailto:brogle@iam.unibe)

**Patrick Lauer**, dipl. Inf., ist seit Anfang 2007 Mitarbeiter in der Forschungsgruppe Rechnernetze und Verteilte Systeme an der Universität Bern, wo er sich mit verschiedenen Aspekten von P2P-Netzen beschäftigt.

Universität Bern, IAM, 3012 Bern, [lauer@iam.unibe](mailto:lauer@iam.unibe)

<sup>1)</sup> Hypertext Transfer Protocol.  
<sup>2)</sup> Transport Control Protocol.  
<sup>3)</sup> User Datagram Protocol.  
<sup>4)</sup> Session Initiation Protocol.