



G N U T E L L A
F Ü R
B E N U T Z E R

Inhalt

Was ist Gnutella, und wie funktioniert es?

Grundlagen von Gnutella

Netzwerk-Modell: Der Ursprung: FoF
Hineinkommen: Der erste Weg: Pong-Caching
Hineinkommen: Der zweite Weg: Aufschreiben, wer antwortet

Aktuelle Veränderungen in Gnutella

Hineinkommen: Der dritte Weg: GWebCaches

Probleme des FoF-Modells -> Veränderungen
Netzwerk-Modell: Wer kontaktiert Wen: Ultrapeers und Blätter
-UPs wirklich kurz
-Ultrapeer und Blätter detaillierter
Netzwerk-Modell: Intra-Ultrapeer QRP
Netzwerk-Modell: Veränderung des Suchvorgangs: Dynamic Querying

Aktuelle Veränderungen Teil 2:

Quellen finden ohne zu suchen aka das Download-Netz

Bessere Downloads Teil 1: Schwärme und Sharen
unvollständiger Dateien
Bessere Downloads Teil 2: Durch Firewalls herunterladen

Datei-Magnete

Pläne für die Zukunft

Wie kann ich es nutzen?

Editor: ARNE BABENHAUSERHEIDE (<http://draketo.de>) Beta-Leser:

Webseite: <http://gnufu.de>

Copyright (c) 2004 Arne Babenhausnerheide.

This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/2.0/de/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Notiz vom Editor: Arne Babenhausnerheide - schrieb den größten Teil des Dokuments und übersetzte es ins Deutsche. Es wurde am 30. November 2003 für die Öffentlichkeit geöffnet. Beiträge von anderen Autoren unterliegen deren Verantwortung und, wenn sie ihren Namen mit E-Mail oder Webseite und einer Info, was sie geändert haben, in die Liste der Mithelfenden eintragen, deren Copyright. I thank Janet for very nice proofreading of the english text and enhancements in style and readability.

Ich danke dem ganzen GDF für zwei Jahre, in denen sie mich als Nicht-Programmierer in ihrem Forum toleriert haben, weil ich einen großen Teil der Informationen in diesem Dokument während dieser Zeit gelernt habe, teilweise durch einfaches, naives Fragen (Ich hoffe, ich habe es trotzdem geschafft etwas Konstruktives dort beizutragen).

Außerdem danke ich stief und et voilà aus den Gnutellaforums, die ohne Bezahlung und mit viel zu wenig Dank dort ihren Job tun.

Was ist Gnutella, und wie funktioniert es?

Gnutella ist ein offenes p2p Filesharing Netzwerk. Das bedeutet, dass, anders als bei den meisten anderen Netzwerken, jeder ein Programm schreiben kann, das auf das GNet zugreift, solange es die öffentlich verfügbaren Spezifikationen erfüllt. Es wurde ursprünglich von Justin Frankel und John Pepper von Nullsoft entwickelt, die es auf ihrer Seite veröffentlicht hatten. Nach nur wenigen Stunden wurde es von der Mutterfirma von Nullsoft, AOL, gelöscht und von der Webseite entfernt, aber es war bereits tausende Male heruntergeladen worden. Wenige Tage später tauchten die ersten Gnutella-Programme von freien Programmierern im Netz auf, die von dessen Idee fasziniert waren. Die Spezifikationen des daraus entstandenen Netzwerkes werden im »GDF (dem Gnutella Development Forum: http://groups.yahoo.com/group/the_gdf) diskutiert und geschrieben, einer offenen Mailingliste, die zur Zeit etwas über 1000 Mitglieder hat. Danach werden sie in »rfc-gnutella (<http://rfc-gnutella.sf.net/>) dokumentiert. Auf diese Art teilen sich alle Programme eine gemeinsame Grundlage, während das Protokoll auch Programmspezifische Optionen erlaubt. Die Entwickler achten dabei so sorgfältig wie möglich darauf, die Kompatibilität mit älteren Programmen zu erhalten.

Obwohl der Name es zu implizieren scheint, ist das Gnutella Protokoll keine GNU-Software. Einige Gnutella-Programme sind allerdings unter der GPL lizenziert. Gnutella ist ein offenes Netzwerk mit offenen Standards und die Quelle des Names kann einfacher bei zu großem Nutella-Genuss als bei GNU gefunden werden, obwohl der Gedanke Gnu+Tella logisch scheint (das heißt: Gnutella ist kein Projekt der FSF und gehört nicht zu den GNU Programmen).

Ein Nachfolger Gnutellas verdient besondere Aufmerksamkeit, obwohl einige Entwickler aktueller Programme für Gnutella es verneinen würden. Er wird von den meisten Entwicklern MP (Mikes Protokoll), oder das Shareaza Protokoll genannt, während sein Entwickler ihn Gnutella2 (G2) nannte, was seinem Programm (Shareaza) viel Medienaufmerksamkeit lieferte, und sowie Kontroversen als auch Abneigungen gegenüber ihm im GDF erzeugte und erzeugt. Mehr Informationen darüber hier: <http://mikesprotocol.gnufu.net>

Zurück zum Hauptthema dieses Textes: Den Grundprinzipien von Gnutella, ihre Entwicklung und Evolution während der letzten Jahre (vor allem dem letzten Jahr) und Plänen für zukünftige Entwicklungen.

Grundlagen von Gnutella

Netzwerk-Modell: Der Ursprung: FoF

Sie können sich das ursprüngliche Modell des Gnutella-Netzwerks vorstellen, wie eine Gruppe von Freunden, die sich anrufen, um Informationen zu bekommen. Dabei fragt

einer fünf andere, die wieder jeweils fünf fragen, und so weiter. Nach dem ersten Schritt, werden so 5 Personen erreicht, nach dem zweiten sind es 25, nach dem 5. 3125, nach dem 7. 78.125 und nach dem 14. etwa 6,1 Milliarden. Das wäre also genug, um

jedes menschliche Wesen auf diesem Planeten zu erreichen. Das Ursprüngliche Modell benutzte 7 Schritte (die HTL: Hops to Live, also in etwa Schritte zu Leben, genannt wurden). Ein Problem (zusammen mit einigen Anderen) mit diesem Modell ist, dass sie Teil der Clique sein müssen, bevor sie es nutzen können. Es gab verschiedene Ansätze, dieses Problem zu lösen. Ich werde ihnen drei davon zeigen.

Hineinkommen: Der erste Weg: Pong Caching

Pong-Caching bedeutet, dass der Netzwerkknoten (also Sie) seine Freunde fragt, wer deren Freunde sind. Es bedeutet, dass ihre Freunde sie ihren eigenen Freunden vorstellen, vor allem Freunden, die sie wertschätzen, und dass sie alle neuen Adresse in ihr Telefonbuch eintragen, so dass sie später wissen, wen sie anrufen können, wenn ihre ersten Freunde in Urlaub sind. Das System ist einfach und hat den Vorteil, dass es ihnen sehr zuverlässige Kontakte gibt, aber es ist kein Weg, in das Netzwerk zu kommen, ohne zumindest einen Kontakt zu haben, der bereits darin ist. Das bedeutet, dass sie immer wieder hinein können, aber noch keine Möglichkeit haben eine Verbindung herzustellen, wenn sie es

nicht vorher schon einmal getan haben.

Hineinkommen: Der zweite Weg: Aufschreiben, wer antwortet

Der zweite Weg ist sehr einfach: Wenn einer ihrer fünf Freunde antwortet, um zu sagen, dass Schmidt (die sie vorher nicht kannten) etwas weiß, schreiben sie deren Nummer auf. Wenn Sie sie das nächste Mal als einen ihrer fünf Direktkontakte anrufen, ist die Chance größer, dass Sie die benötigte Information schneller bekommen, weil sie wahrscheinlich Freunde haben wird, die ähnliche Interessen haben, wie sie (sonst hätte sie die Information wahrscheinlich nicht), und bei diesen ist es wahrscheinlicher dass sie weitere Informationen haben, die sie interessieren, als ein zufällig ausgewählter Kontakt. Der Nachteil ist, dass diese antwortenden Kontakte nicht unbedingt immer zu Hause sind, weswegen sie einen Kontakt mit großem Wissen (mit vielen interessanten Dateien) finden könnten, den sie nie wieder erreichen können. Das ist allerdings immer noch kein Weg zum ersten Einstieg in das Netz. Und damit kommen wir zu einer jüngeren Neuerungen in Gnutella: GWebCaches. Ich werde sie im nächsten Teil erläutern.

Recent Changes in Gnutella

Hineinkommen: Der dritte Weg: GWebCaches

Im Bild bleibend, ist ein GWebCache ein Kontakt, der seine Telefonnummer in die Zeitung setzt, und sich aufschreibt, wer anruft. Jedes Mal, wenn Sie eine Weile weg waren und merken, dass Ihre Kontakte nicht mehr die selbe Handynummer haben, rufen Sie den öffentlich bekannten Kontakt an.

Bevor er ihnen weitere Telefonnummern gibt, wird er Sie fragen: "Kennen Sie andere öffentlich bekannte Kontakte? Wenn ja, bitte sagen sie mir ihre Nummern."

Das wird gemacht, weil die öffentlichen Kontakte nicht alle Zeitungen lesen können, und Sie es die ganze Zeit tun, ohne hart dafür zu arbeiten. Auf die Art können die Kontakte die Übersicht über die anderen öffentlichen Kontakte behalten. Dann gibt Ihnen der Kontakt einige Nummern, die Sie

anrufen können, und die Nummern anderer öffentlicher Kontakte, die er kennt (GWebCaches), und notiert ihre Nummer, um sie jemand Anderem geben zu können. Das beschreibt ungefähr, wie GWebCaches funktionieren.

Wie gesagt sind sie eine der Neuentwicklungen in Gnutella, und darum werde ich nun zu einigen weiteren Entwicklungen in Gnutella und zu Plänen für die Zukunft kommen.

Notiz: GWebCaches sind nur für die erste Verbindung nötig, und nur, wenn ihr lokaler Host-Cache leer ist, aber sie dürfen ihrem lokalen Adressbuch nicht vorgezogen werden (ansonsten werden sie überlastet).

Probleme des FoF-Modells

-> Veränderungen

Das Freund-eines-Freundes Modell hat bestimmte Nachteile, die ihre Quelle in der Art haben, wie die Suchen ablaufen. Wenn eine Suchanfrage zu viele Ergebnisse bringt, können die Knoten, über die Sie verbunden sind (ihre 5 nächsten Freunde) überladen werden, weil sie ihre "Telefonnummer" nicht mitschicken, sondern ihre eigene, und so jede Antwort durch sie geht und sie sie weiterleiten. Wenn Sie beispielsweise auf dem Kampus nach dem Leiter der Universität fragen, werden sie in der Realität hunderte von Antworten bekommen, im Netz Tausende bis Millionen. Wenn außerdem jede Frage an Jeden in einem 75.000 bis 600.000 Computer Netzwerk weitergeleitet wird, und jeder Computer nur einmal pro Stunde fragt, muss jeder von ihnen zwischen 130 und 1600 Fragen pro Sekunde beantworten. Zusätzlich müssen sie sie weiterleiten, nicht zu vergessen die Antworten, die auch weitergeleitet werden müssen. Obwohl Computer schnell sind und heutige Internetverbindungen im Vergleich zu Verbindungen vor einem Jahr weit mehr leisten, ist das selbst für sie zu viel.

Stellen Sie sich vor, Ihr Telefon würde den ganzen Tag wegen allen Arten von Fragen und ohne Pause klingeln.

Um dieses Problem zu lösen, wurden einige Veränderungen am FoF-Modell nötig.

Netzwerk-Modell: Wer kontaktiert Wen: Ultrapeer und Blätter

-UPs wirklich kurz

Sicher haben Sie Freunde, die viele andere Leute kennen und die Sie alles mögliche fragen und sicher sein können, dass sie genau die Person kennen, die ihnen die Antwort geben kann. In Gnutella werden sie Ultrapeer genannt. Ein Ultrapeer muss selbst nicht viel wissen, sondern nur, wer es weiß. In Gnutella bedeutet das, dass ein guter Ultrapeer nicht unbedingt viele Dateien zu haben braucht, um dem Netzwerk zu nutzen. Wenn sie Angst haben viel zu sharen sollten sie Ultrapeer in Gnutella werden.

-Ultrapeer und Blätter detaillierter

In der Computerwelt gibt es, wie in der physischen Welt, Kontakte, die mehr Anrufe bearbeiten können und andere, die weniger schaffen (oder sich die Rechnungen nicht leisten können). In der echten Welt liegt das daran, dass sie mehr freie Zeit oder mehr Geld haben, während in der Computerwelt schnellere Verbindungen (wie DSL, Kabel, T1, T3 oder ähnliches Breitband) der Grund sind. Nachdem sie das erkannt und überdacht hatten, entschieden sich die Programmierer die Topologie des Netzwerks, das heißt wie das Netzwerk von Außen betrachtet aussieht, zu verändern. Jetzt kontaktieren sie nicht einfach irgendeinen ihrer Freunde, sondern nur diejenigen, von denen sie wissen, dass sie die Zeit haben ihren Anruf anzunehmen und an Andere weiterzuleiten. Um Sie vor zu

vielen Anrufen zu bewahren, fragen diese dann, welche Arten von Informationen Sie haben oder um es menschlicher auszudrücken, welches ihre Spezialität ist.

In der Computerwelt bedeutet das, ihr Computer verschickt eine Liste mit allen gesharten Dateien an den Ultrapeer (wie diese Kontakte genannt werden). Diese Liste enthält Zusammenfassungen (Hash-Strings) aller Dateien, die Sie für andere zum Download freigegeben haben und über die die Herunterladende überprüfen kann, ob sie wirklich die Dateien sind, die sie will. Wann immer ein Anruf einen Ultrapeer erreicht, überprüft er, ob Sie die Antwort wissen können und ruft Sie nur in dem Fall an.

Diese Ultrapeers haben viele Verbindungen zu Anderen, was bedeutet, dass sie ein wirklich großes Adressbuch haben. Normalerweise bleiben sie in aktivem Kontakt mit 16 anderen Ultrapeers aus ihrem Adressbuch denen sie Fragen stellen, die diese dann an 16 weitere weiterleiten. Außerdem haben sie etwa 16 Blätter, die nicht so viel telefonieren können oder wollen, von denen sie Anrufe annehmen, und deren Dateien oder, in der menschlichen Welt, Spezialitäten sie kennen.

Dies mag wie ein fauler Handel für die Ultrapeer erscheinen, die weit mehr Ressourcen investieren als die Blätter, um das Netzwerk intakt zu halten, aber effektiv ist der Handel für beide nützlich. Während die Ultrapeer (UP) viel ihrer Zeit nutzen, um das Netzwerk am Laufen zu halten, spezialisieren sich Blätter darin Informationen zu sammeln und weiterzugeben. Auf diese Art kann jede, die etwas wissen will, einfach einen Anruf starten, und ein spezialisiertes Blatt kann es ihr erklären. Das heißt die Leute spezialisieren sich, damit am Ende mehr für alle herauskommt.

Netzwerk Modell: Intra-Ultrapeer QRP

Während mit den Ultrapeer nicht mehr jeder beim weiterleiten der Fragen mithelfen muss, und die Leute sich stattdessen darauf spezialisieren können, ihre Informationen anderen zugänglich zu machen, würden die Ultrapeer immernoch jede Frage an jeden schicken, egal, ob dieser Ultrapeer überhaupt Blätter hat, die diese Art von Informationen haben. Das hört sich normal an, denn wie kann ein Ultrapeer wissen, welche Dateien die anderen Ultrapeer haben? Die Antwort findet sich, erneut, im wirklichen Leben. Eine normale Person kennt ihre Freunde, und sie weiß, welche von ihnen die Antwort auf bestimmte Fragen wissen könnten, und welche sehr wahrscheinlich nicht. Im wirklichen Leben wird das größtenteils durch lockere Unterhaltungen möglich.

Normalerweise betreiben Computer jedoch keinen Small Talk, daher tauschen sie diese Informationen nicht einfach nebenbei aus. Deshalb wurde das Query Routing Protokoll (Suchanfragen Leitungsprotokoll) entwickelt. Dabei sagt jedes Blatt seinen Ultrapeer, welche Dateien es hat, aber statt die Namen zu nennen, die zu viel Platz verbrauchen würden, wird jedes Wort, das Teil des Dateinamens ist, als eine Zahlenfolge gespeichert (Immerhin reden wir von Computern). Sie können sich den darauf folgenden Ablauf wie ein Spiel Schiffe versenken vorstellen (Die Zahlen bilden das Spielbrett mit zwei Koordinaten). Ein Ultrapeer leitet nicht alle Fragen an das jeweilige Blatt weiter, sondern nur die, die es beantworten könnte (deren Suchworte ein Schiff treffen), und auf diese Art bekommen Blätter weit weniger unnütze Anrufe.

Wenn dies so viel Druck von den Blättern nimmt, warum es nicht erweitern? Genau das wurde getan. Alle Ultrapeer schicken

ihre Spielbretter, auf denen ihre Schiffe und die Schiffe ihrer Blätter vermerkt sind, an ihre direkten Nachbarn. Dann leiten sie von den Suchanfragen, die nur noch einen weiteren Schritt laufen, nur jene an die Ultrapeer weiter, auf deren Spielbrettern sie einen Treffer erzielen. Das heißt die letzten zwei Schritte einer Suche (zum letzten Ultrapeer und zu dessen Blättern) werden nur dann gemacht, wenn es eine Chance gibt, dass sie Ergebnisse bringen.

Sie können sehr leicht erkennen, warum dadurch die benötigte Bandbreite deutlich gesenkt wird, indem sie sich einen Baum vorstellen. Einen normalen Baum, nicht eines dieser mathematischen Kontrukte. Wenn sie versuchen die Blätter zu zählen, haben sie fast keine Chance jemals fertig zu werden. Wenn sie aber die Blätter entfernen und nur noch die kleinen Ästchen zählen, liegt bereits weit weniger Arbeit vor ihnen. Wenn sie nun auch noch all die winzigen Ästchen weg nehmen, können sie wirklich mit dem Zählen der verbleibenen Äste beginnen. QRP entfernt nicht alle Blätter und Ästchen, aber es entfernt diejenigen, die ihnen keine Antwort liefern könnten. Da jeder Teil (Ast oder Blatt) durch den eine Suchanfrage geleitet werden muss, Bandbreite verbraucht, und es weit mehr Blätter als Äste (Ultrapeer) gibt, führt die Wegnahme eines Großteils der letzten beiden Schritte (Blätter und kleine Ästchen) zu enormen Bandbreitensparnissen. Dieses Beispiel funktioniert nicht für alle Teile von Gnutella, aber hier passt es sehr gut. Die Programmierer LimeWres sprechen von Ersparnissen um 70%-80% alleine durch QRP.

Netzwerk Modell: Veränderung des Suchens: Dynamic Querying

Während das Ultrapeer-Modell und QRP einen Teil des Problems lösen, dass sie keine Zeit haben, etwas jemand Anderem richtig zu erklären oder es erklärt zu bekommen, weil das Telefon ständig mit Fragen nervt, auf die sie keine Antwort kennen (oder in Tech-Sprech: Weil die Netzwerkbelastung die Geschwindigkeit ihrer Verbindung voll auslastet), gibt es immernoch ein anderes Problem, das normalerweise kaum sichtbar würde, selbst wenn sie direkt darauf schauen. In der wirklichen Welt wird ein Ultrapeer nach einer Spezialistin suchen, die ihnen die Info geben kann, bis er eine Passende findet, und danach aufhören. In der Computerwelt wird die Frage immer zu so vielen Kontakten wie möglich weitergeleitet, ohne darauf zu achten, ob es bereits Antworten gibt.

Mit Dynamic Querying (Dynamischen Suchanfragen) ändert sich das. Nun fragen die Ultrapeer nur einen anderen Ultrapeer auf einmal und warten etwas, um zu sehen, ob sie Antworten bekommen (deswegen scheinen Gnutella-Suchanfragen für seltene Dateien seit kurzem etwas langsamer zu sein, dafür erhöht sich aber die Chance die Dateien wirklich zu finden). Wenn sie eine zufriedenstellende Zahl von Antworten haben (normalerweise etwa 250), beenden sie die Suche.

Das hört sich eigentlich völlig normal und logisch an, war aber ein wirklich großer Schritt für Gnutella, weil es Ressourcen spart, die für sehr verbreitete Dateien verschwendet wurden. Ich nehme erneut das Beispiel des Leiters der Universität: Wenn sie nun nach dem Dekan fragen, werden ihre Ultrapeer erst überprüfen, ob sie direkt jemanden kennen, der ihre Frage beantworten kann. Dann werden sie ihnen

einfach einige Nummern von Leuten geben, die sie auf dem Campus kennen. Sie werden immernoch mehr als eine Antwort bekommen, weil sie ihnen mehr als eine Nummer geben, schließlich können die Ultrapeer nicht sicher sein, dass Sie jede der Nummern erreichen können. Aber sie werden nicht mehr tausende von Nummern bekommen (eine von jedem Studenten auf dem Campus). Erstens weil damit die Ultrapeer ihre Zeit mit etwas verschwenden würden, das Ihnen keinen zusätzlichen Vorteil bringt, zweitens, weil sie niemals alle der Leute anrufen könnten und drittens, weil sie dann Ihre Ultrapeer nicht mehr erreichen würden, weil diese zu beschäftigt damit wären, Antworten von anderen anzunehmen, die ihnen ihre Nummern geben und ihre Fragen an andere Ultrapeer weiterzuleiten. Den Statistiken der Bearshare-Programmierer folgend, spart das weitere 60% der Netzwerkbelastung.

Quellen finden ohne zu suchen aka das Download-Netz

Nun mögen sie denken, "aber ich kann von diesen drei Quellen nicht herunterladen, weil es schon andere machen. Ich will alle Adressen, von denen ich herunterladen kann" (Und sie sind damit nicht alleine. Ich fühle dasselbe). Mit einem Blick auf die wirkliche Welt können wir jedoch auch für dieses Problem eine Lösung finden, ohne allzu viele Ressourcen darauf verschwenden zu müssen. Wenn sie dort (in der wirklichen Welt) eine Spezialistin bitten, ihnen etwas zu erklären und diese Spezialistin beschäftigt ist, wird sie andere Spezialistinnen kennen (weil sie sich gegenseitig kennen) die momentan mehr Zeit haben.

Dieses System in Gnutella zu integrieren ist nicht so einfach wie bei dem Ultrapeer-Blatt Modell oder dem Dynamischen Suchsystem.

Aber die Programmierer fanden einen Weg. Wie ich im Dynamischen Suchsystem geschrieben habe, bekommen sie mehr als eine Adresse bei der sie Anfragen können. Wenn sie nun jemanden fragen, der die Antwort kennen sollte, geben sie ihm die anderen Adressen, die sie kennen. Auf diese Art lernen diese Spezialisten von der Existenz der Anderen (auf die selbe Art wie die GWebCaches, von denen ich bereits gesprochen habe). Indem jeder der anfragt auch sein eigenes Set von Adressen liefert, kennen die Spezialisten mehr und mehr zusätzliche Adressen, und wenn Sie sie bitten etwas zu erklären, und sie momentan keine Zeit haben, geben sie ihnen die Adressen (sie tun es auch, wenn sie Zeit haben, einfach für den Fall, dass sie unterbrochen werden, und weil sie in Gnutella von mehr als einer Quelle auf einmal herunterladen können, wie sie es vielleicht von dem Overnet-Netzwerk kennen (das es ins Extreme treibt, aber nur für große Dateien wirklich effizient ist)). Zusätzlich fügen die Spezialisten ihre Nummer zu ihrer Liste von Alternativen Kontakten hinzu, sobald sie genug wissen, um anderen etwas beizubringen.

Das ist der Grund, warum oft viele Leute Dateien von ihnen herunterladen, die sie gerade selbst heruntergeladen haben.

Bessere Downloads Teil1: Schwärme und weitergeben unvollständiger Dateien

Swarming lässt sich leicht erklären (aber nur schwer im Freund-eines-Freundes Modell, deswegen lasse ich es für diesen Teil weg). Es funktioniert indem einfach eine Datei von mehr als einer Quelle gleichzeitig heruntergeladen wird und die einzelnen Teile wie ein Ameisenschwarm an ihren Platz finden und die fertige Datei formen. Dafür wird die Datei einfach in

mehrere Teile geteilt, als wollten sie ein Buch von mehreren Freunden bekommen, und jeder von ihnen würde nur ein paar von dessen Seiten kopieren. Wenn sie jeden von ihnen bitten einen anderen Teil des Buches zu kopieren, werden sie das gesamte Buch bekommen und jeder von ihren Freunden hat nur wenig Arbeit (und falls Einer gerade keine Zeit hat um es zu machen, übernimmt einfach ein Anderer).

Swarming funktioniert am besten mit dem Download-Netz und Partial File Sharing (PFS, Sharen unvollständiger Dateien), das es den Nutzern erlaubt Anderen den teilweisen Download von Dateien zu ermöglichen, die sie selbst gerade herunterladen. Sie können die Seiten, die sie bereits haben, kopieren ohne das ganze Buch haben zu müssen, weil die Seiten nummeriert sind und ihre Freunde sie bitten können bestimmte Seiten zu kopieren.

Der Name Schwärmen hat keine weitere weitere Bedeutung, ruft aber schön das Bild von ameisenähnlichen Dateifragmenten hervor, die zu ihrem Computer hasten.

Bessere Downloads Teil2: Durch Firewalls herunterladen

Stellen sie sich vor es gäbe Leute, die nicht angerufen werden könnten, sondern nur andere anrufen könnten (vielleicht, weil sie nur öffentliche Telefone benutzen, oder weil ihre Nummer nicht auf anderen Telefonen angezeigt wird und sie sie ungerne herausgeben, weil sie es nicht mögen von Telefonverkäufern belästigt zu werden oder sich Sorgen um Telefonterror machen). In Gnutella sind das Computer, die hinter einer Firewall sind. Sie können andere anrufen und von diesen Informationen bekommen, aber niemand kann sie anrufen.

Eine Lösung für dieses Problem ist, dass die Leute hinter einer Firewall ihre Ultrapeer regelmäßig anrufen und falls jemand sie anrufen will, dieser einfach den Ultrapeer anruft, der dann zwei Telefonhörer zusammenhält, einen, auf den die Person hinter der Firewall (die nicht angerufen werden kann) angerufen hat und denjenigen, auf den sie angerufen haben. Auf die Art können sie mit der Person hinter der Firewall sprechen. Allerdings benötigt das zwei gleichzeitige Anrufe, was in der Copmuterwelt bedeutet, dass die doppelte Bandbreite benötigt wird. Leute hinter einer Firewall halten ihre Verbindungen zu den Ultrapeer meistens aufrecht, während die Ultrapeer darüber Informationen oder Daten weiterleiten.

Es gibt Pläne die Ultrapeer von dieser zusätzlichen Bandbreitennutzung zu entlasten, indem andere Leute das Telefonverbinden übernehmen. Wenn dann jemand Informationen von einer Spezialistin hinter einer Firewall möchte, sagt der Ultrapeer der Spezialistin und dem Anfragenden die Nummer einer dritten Person, die dann die beiden Telefonhörer zusammenhält. Da in Gnutella die meisten Leute drei bis fünf Telefone haben, würde das kein großes Problem darstellen. Diese Kontakte werden höchstwahrscheinlich Routing-Peers genannt werden (Verbindungs-Kontakte).

Datei-Magnete

Dateimagnete lösen sich vom Freund-eines-Freundes Konzept. Sie sind Links auf Webseiten, die sie einfach anklicken können, und die ihrem Filesharing Programm sagen, dass es Gnutella (oder andere Netzwerke) nach einer bestimmten Datei durchsuchen und exakt diese herunterladen soll.

Sie können sich das wie einen Artikel in einer Zeitung vorstellen, der ihnen die Information gibt, mit der ihr Ultrapeer genau weiß, was die Spezialistin wissen oder können muss, von der sie lernen wollen. In der wirklichen Welt würden sie höchstwahrscheinlich eine Spezialistin und alle diejenigen finden, die von ihr gelernt haben.

Mit einem Dateimagneten können sie es vermeiden korrupte und falsch benannte oder sonstwie falsche Dateien zu bekommen, weil sie einen Hash-String benutzen, der eine Art Zusammenfassung der Informationen der Spezialistin ist. Wenn sie anfängt ihnen Mist zu erzählen, werden sie sofort merken, dass es nicht zur Zusammenfassung passt. In Gnutella startet das Programm eine Suchanfrage nach Dateien, denen die Leute, die sie haben, die Zusammenfassung (also den Hash-String) gegeben haben, die der Magnet enthält. Nach dem Herunterladen erstellt das Programm seine eigene Zusammenfassung der Datei und überprüft, ob sie wirklich zusammenpassen. Wenn sie nicht passen, sagt es ihnen, dass die Datei korrumpiert wurde. Die Zusammenfassungen derselben Dateien sind immer exakt gleich, weil sie mithilfe spezifischer mathematischer Methoden erstellt werden, die für dieselben Daten (also Informationen) immer dasselbe Ergebnis liefern.

Anders als Magnetlinks sind KaZaA-Links und eDonkey-Links nicht sicher, weil sie Methoden zur Zusammenfassung benutzen, die mit falschen Dateien hereingelegt werden können (KaZaA-Links zum Beispiel suchen nach einer Zusammenfassung, die nur die Einführung und den ersten Teil der Information enthält, aber den Rest ignoriert, um die Zusammenfassung schneller erstellen zu können. Natürlich ist es sehr einfach

ihnen damit falsche Informationen zu geben, weil die Spezialisten ihnen nur am Anfang die Wahrheit erzählen müssen und dann lügen oder fantasieren können, soviel sie wollen).

Dateimagnete können außerdem Suchwörter enthalten, über die Dateien gesucht werden können, als würden sie eine normale Suchanfrage in Gnutella starten, manche enthalten sogar Webadressen, von denen die Dateien heruntergeladen werden können, falls sie in dem Filesharing-Netzwerk nicht aufzufinden sind.

Weitere Informationen über Datei-Magneten können über die folgenden Links gefunden werden: »Magnet-Uri (<http://magnet-uri.sf.net>) liefert die Spezifikationen und »MagnetLink.org weitere nützliche Infos.

Es gibt inzwischen eine neue Version von magnet-links (aka Dateimagneten): KaZaA Magnete. Leider sind diese nicht sicher, weil sie das KaZaA Hashing-System (die unvollständige Zusammenfassung) mit einigen Veränderungen benutzen (sie fügen nun eine weitere kleine Zusammenfassung hinzu, die Informationen über die fehlenden Teile enthält, aber sie haben noch nicht veröffentlicht wie sie sie erstellen). Sollten KaZaA-Magnete Informationen über Suchwörter enthalten, können sie mit Gnutella funktionieren, werden aber nicht sicherstellen, dass Sie bekommen, was ihnen angeboten wurde. Wenn sie das Wort "kzhash" in dem Link finden, ist der Link möglicherweise nicht sicher (abgesehen von dem etwas missratenen Namen).

Sie werden einige Dateimagnete auf der Magnet-Lists-Seite finden (<http://magnetlists.gnufu.net>)

Pläne für die Zukunft (Stand: 15.5.2004)

- Community-Feature, Region-Preferencing.
- Encryption (Verschlüsselung) - Es dem Provider unmöglich machen zu sehen, was sie herunterladen. Bearshare verschlüsselt bereits bei der Kommunikation mit anderen BearShare-Programmen.
- Beliebte Inhalte zwischenspeichern.
- Magma-Listen - Vielfach-Magnete, ähnlich wie Spiellisten (<http://magma.gnufu.net>)
- Privacy (Anonymität und Privatsphäre schaffen) (Siehe <http://anognut.gnufu.net> (englisch)).
- Routing Peers und bessere Firewall Unterstützung.
- Was gibt's Neues? - Neue Dateien in Gnutella finden (in LimeWire integriert).

Wie kann ich Gnutella nutzen?

Um Teil des Gnutella Netzwerks zu werden, können sie eins der Programme von <http://www.gnutella.com/connect/> nutzen (sollte diese Seite offline sein, benutzen sie einfach die Liste auf dmoz (http://dmoz.org/Computers/Software/Internet/Clients/File_Sharing/Gnutella/), oder besser noch: Lernen sie zu programmieren und helfen sie bei Weiterentwicklung einiger der Open-Source-Programme.

Auf der GnuFU Webseite finden sich Links zu mehreren Open Source und Closed Source Programmen:
<http://gnufu.net>

Da GnuFU durch Probleme bei meinem Provider letzte Woche mehrfach down war, habe ich mich entschlossen, nicht länger vollständig von ihm Abhängig zu sein. Sollte diese Seite aus dem Netz verschwinden, weswegen auch immer existiert doch eine Kopie des PWP-Codes in Freenet. Diese Kopie kann nur über das Freenetproject abgerufen werden, ein verteiltes, vollständig anonymisiertes Informationsnetzwerk. Das zum Zugriff darauf nötige Programm kann von »<http://freenetproject.org> heruntergeladen werden. Der Schlüssel zu GnuFU auf Freenet ist: SSK@1~6U-1UApvA5hld50tMsau3O5tEPAgM,kr~pLjSLxfECXC2Mvt3RKw/gnufu/3//gnufu-de.html

Bei den meisten Systemen wird der Link "http://free.gnufu.net" sie dorthin bringen. Um auf diesen Spiegel zuzugreifen, benötigen sie das Programm von »<http://freenetproject.org>.