

RANCANG BANGUN JARINGAN PEER TO PEER DENGAN KONSEP SEMANTIC OVERLAY NETWORKS

Ary Mazharuddin Shiddiqi, A. Zainal Abidin, Waskitho Wibisono

Jurusan Teknik Informatika
Fakultas Teknologi Indformasi

Institut Teknologi Sepuluh Nopember Surabaya

email : ary.shiddiqi@cs.its.ac.id, zhainhart@cs.its.ac.id, waskitho_w@yahoo.co.uk

Abstrak

Peer-to-peer adalah sebuah aplikasi yang menghandle resource dari sejumlah autonomous participant atau user yang terkoneksi secara mandiri. Artinya user dapat mengoneksikan dirinya sesuai dengan keinginannya, tidak terikat oleh struktur jaringan secara fisik. Peer-to-peer menjadi sebuah alternatif aplikasi untuk mencari resource tertentu yang tidak ada di website ataupun alternatif untuk berbagi resource tanpa sebuah web server yang harganya masih tergolong mahal.

Di dalam penelitian ini, dikembangkan sebuah aplikasi peer-to-peer yang menerapkan metode semantic overlay networks dalam membangun jaringan. Metode ini membagi peer yang terkoneksi didalam sebuah jaringan berdasar pada kontent dari file yang dishare. Koneksi tersebut tidak lagi dibagi secara random seperti pada sistem peer-to-peer yang ada sekarang ini. Metode ini menyebabkan peer-peer dengan kesamaan konten berada dalam grup yang sama sehingga kemungkinan untuk mendapatkan file atau resource yang dicari akan semakin besar.

Selain itu, penggunaan konsep jaringan secara hirarki, memungkinkan untuk lebih mengoptimalkan pencarian, karena apabila sebuah resource tidak bisa ditemukan pada sebuah grup, pencarian akan dilakukan digrup yang lebih umum dari grup tersebut. Keunggulan dari dikembangkannya sistem ini adalah dapat mengurangi waktu pencarian resource, juga yang tidak kalah pentingnya adalah mengurangi beban jaringan, karena query yang tidak diperlukan tidak akan dilewatkan dalam jaringan tersebut.

Kata Kunci: *peer-to-peer system, semantic overlay networks*

1. PENDAHULUAN

Peer to Peer (P2P) adalah metode untuk mengkoneksikan dua komputer untuk membentuk jaringan yang sederhana. Pengkoneksian ini dimaksudkan untuk membagi resource yang ada dikomputer yang satu dengan komputer yang lain. Konsep pembangunan jaringan Peer to Peer ini kemudian diterapkan untuk membangun sebuah aplikasi yang berkemampuan untuk mengkoneksikan secara Peer to Peer komputer yang terhubung dengan jaringan internet.

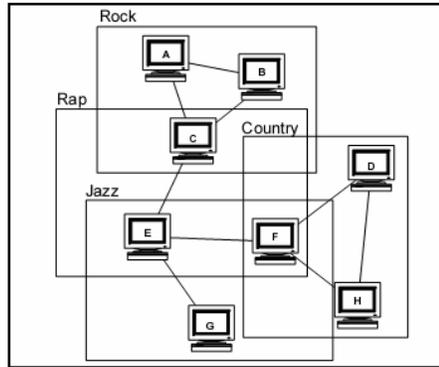
Pada aplikasi Peer to Peer, terdapat sebuah server yang akan membagi user yang terkoneksi pada server tersebut kedalam sebuah atau beberapa sub jaringan yang lebih kecil secara acak. Tujuannya adalah untuk mempercepat proses pencarian resource. Pada sistem ini, setiap request dari user akan diproses kedalam jaringan yang terkoneksi dengannya. Apabila request tidak ditemukan, user bisa merequest untuk mengkoneksikan dirinya dengan jaringan lain melalui server.

Metode yang telah ada dirasa kurang efektif dan membutuhkan resource yang besar, karena harus mencari kesemua node dengan komposisi yang acak. Oleh karena ini dalam penelitian ini akan didesain jaringan pada aplikasi Peer to Peer yang akan membagi user yang terkoneksi dengan server kedalam sebuah atau beberapa sub jaringan yang lebih kecil dengan menerapkan metode *semantic overlay networks*. Metode untuk membagi user sesuai dengan kemiripan konten yang dimiliki atau dishare. Selain itu diterapkan juga metode hierarchial (parent-child) untuk mempercepat proses pencarian.

Semantic Overlay Networks pada sistem peer-to-peer

Penerapan Semantic overlay networks dalam proses pembentukan jaringan peer-to-peer diharapkan akan lebih meminimalisasi *flooding* saat proses pencarian. Konsep semantic akan mengelompokkan peer-peer yang memiliki kesamaan konten kedalam sebuah jaringan yang sama. Dengan ini, metode diharapkan kemungkinan untuk menemukan sebuah resource akan lebih besar ketika proses pencarian dilakukan pada kelompok dimana resource itu mungkin berada [CRE03].

Sebagai contoh, dalam sekelompok peer yang memiliki banyak file audio yang disharing, katakanlah A-H. Tiap-tiap peer dapat dikelompokkan berdasarkan mayoritas genre file audio yang dimiliki. Tiap-tiap peer akan dihubungkan sesuai dengan konten yang dimiliki, seperti yang ada pada gambar dibawah ini :



Gambar 1. Contoh Penerapan Semantic Overlay Networks[CRE03]

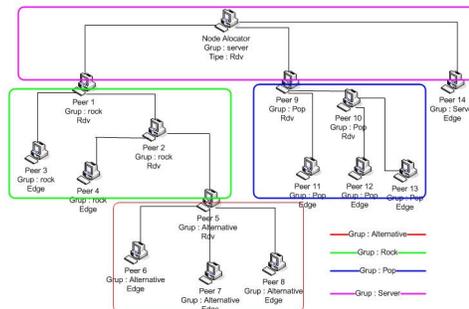
Peer A, B, C memiliki mayoritas file audio bergenre Rock, sehingga dikelompokkan dalam satu jaringan 'Rock'. Peer C, E dan F juga memiliki file-file audio bergenre Rap, jadi juga dikelompokkan dalam jaringan 'Rap'. Dalam sistem ini tidak menutup kemungkinan terdapat peer yang bergabung dengan dua atau lebih kelompok jaringan. Dalam proses pencarian, sebelum query message dikirimkan, query message akan diproses terlebih dahulu untuk menentukan dimana seharusnya mencari resource tersebut. Query message hanya akan dilewatkan kepada peer-peer yang tergabung dengan jaringan tersebut, sehingga jaringan lain tidak akan dibebani dengan pencarian yang memiliki kemungkinan yang kecil untuk menemukan sebuah resource.

2. Perancangan Arsitektur Sistem

Selayaknya sistem peer-to-peer, setiap peer yang ada pada sistem dapat berkomunikasi satu dengan yang lainnya. Yang membedakan sistem ini dengan sistem peer-to-peer yang ada sekarang ini adalah pengelompokan tiap peer sesuai dengan konten yang dimiliki. Peer yang mengandung banyak konten file audio bergenre rock akan menjadi satu grup dengan peer lain yang memiliki mayoritas genre rock juga.

Sistem peer-to-peer ini dikembangkan dengan menggunakan bahasa pemrograman Java dengan framework JXTA (juxtapose) yang merupakan framework open source yang menyediakan fungsi-fungsi standart untuk membangun sebuah sistem peer-to-peer.

Selain terdiri atas peer seperti aplikasi peer-to-peer pada umumnya, pada sistem ini, terdapat sebuah node alocator yang bertugas untuk mengalokasikan tiap-tiap peer yang terkoneksi kedalam sistem peer-to-peer sesuai dengan konten yang dimiliki. Contoh sebuah jaringan peer-to-peer dengan menerapkan semantic overlay network bisa dilihat pada gambar dibawah ini



Gambar 2. Contoh jaringan peer-to-peer dengan menerapkan metode semantic overlay networks

3. Skenario Uji Coba Performa

Dalam uji coba performa, akan dihitung kecepatan proses koneksi, proses pencarian, proses join group dan kecepatan serta transfer rate pada proses download. Uji coba hanya dilakukan pada aplikasi peer, karena node alocator hanya berfungsi untuk mengalokasikan peer kepada grup dan mempropagasi message menuju grup yang memiliki konten terbesar.

Uji coba dilakukan dengan menggunakan 2 komputer yang dihubungkan secara peer-to-peer dengan IP : 192.168.0.2 dan 192.168.0.3 dengan menjalankan 1 aplikasi node alocator, 18 edge peer dan 3 rendezvous peer. Detail dari konfigurasi aplikasi dapat dilihat dalam tabel dibawah ini :

Tabel 1 : Konfigurasi peer dalam sistem

Nama Peer	Tipe	Port
NodeAlocator	Node Alocator	9701
EdgePop1	Edge	9703
EdgePop2	Edge	9704
EdgePop3	Edge	9705
EdgePop4	Edge	9706
EdgePop5	Edge	9707
EdgePop6	Edge	9708
RdvPop1	Rendezvous	9709
RdvPop2	Rendezvous	9710
EdgeRock1	Edge	9711
EdgeRock2	Edge	9712
EdgeRock3	Edge	9713
EdgeAlter1	Edge	9714
EdgeAlter2	Edge	9715
EdgeAlter3	Edge	9716
RdvAlter1	Rendezvous	9717
EdgeHard1	Edge	9718
EdgeHard2	Edge	9719
EdgeHard3	Edge	9720
EdgeClass1	Edge	9721
EdgeClass2	Edge	9722
EdgeClass3	Edge	9723

3.1. Uji kecepatan koneksi

Uji kecepatan koneksi dilakukan dengan menghitung waktu yang diperlukan bagi tiap-tiap peer untuk masuk kedalam sistem. Waktu dihitung mulai dari membangun koneksi dengan node alocator hingga peer bergabung dengan salah satu grup yang ada. Hasil pengujian dapat dilihat dibawah ini :

Tabel 2. Kecepatan koneksi peer kedalam sistem (dalam milidetik)

Nama Peer	Waktu 1	Waktu 2	Rata-rata
EdgePop1	8673	4206	6439,5
EdgePop2	8433	1242	4837,5
EdgePop3	8593	1422	5007,5
EdgePop4	2508	7748	5128
EdgePop5	6548	7650	7099
EdgePop6	6658	2534	4596
RdvPop1	9293	7682	8487,5
RdvPop2	8990	8732	8861
EdgeRock1	8863	6878	7870,5
EdgeRock2	9012	8701	8856,5
EdgeRock3	6700	10213	8456,5
EdgeAlter1	2133	12258	7195,5
EdgeAlter2	8433	7221	7827
EdgeAlter3	7765	2341	5053
RdvAlter1	7245	3365	5305
EdgeHard1	8762	8562	8662
EdgeHard2	8762	8876	8819
EdgeHard3	8732	8762	8747
EdgeClass1	8563	4291	6427
EdgeClass2	8622	8976	8799
EdgeClass3	8664	8312	8488
Rata-Rata	7712	6665	7188

Dari ujicoba diatas dapat diambil kesimpulan, bahwa proses koneksi peer kepada sistem mengalami pasang surut, artinya kadang cepat tetapi kadang juga lambat, tergantung dari secepat apa peer memperoleh rendezvous advertisement. Tetapi dapat diambil rata-rata yaitu waktu yang dibutuhkan sebuah peer untuk bergabung kedalam sistem adalah 7 detik, tidak dipengaruhi banyaknya peer yang bergabung didalamnya.

3.2. Uji Kecepatan pencarian

Pencarian resource dilakukan dalam dua scope, yaitu dalam grup dimana peer tersebut berada dan pencarian pada grup lain. Waktu dihitung dari pada saat tombol search ditekan hingga hasil pencarian ditampilkan. Hasil uji coba dapat dilihat pada tabel dibawah ini :

Tabel 3.Kecepatan pencarian resource (dalam milidetik)

Nama Peer	Lokal	Grup lain
EdgePop1	2210	4280
EdgePop2	1250	2960
EdgePop3	1820	2106
EdgePop4	2060	2340
EdgePop5	1830	2970
EdgePop6	3020	2780
RdvPop1	1620	2400
RdvPop2	3110	3320
EdgeRock1	2100	1930
EdgeRock2	3010	2010
EdgeRock3	1260	1890
EdgeAlter1	2320	2430
EdgeAlter2	2100	2120
EdgeAlter3	2180	2740
RdvAlter1	1920	2530
EdgeHard1	1870	2210
EdgeHard2	2160	2340
EdgeHard3	2170	2520
EdgeClass1	2360	2540
EdgeClass2	2150	2010
EdgeClass3	2710	2530
Rata-Rata	2153,81	2521,714

Dari uji coba diatas dapat diambil kesimpulan bahwa tidak ada perbedaan waktu yang mencolok antara mencari resource pada group sendiri dibandingkan dengan mencaripada grup lain, perbedaan waktu yang muncul hanya dikarenakan pada pencarian dalam grup lain, message terlebih dulu dikirim kepada node alocator untuk dialokasikan kegroup lain.

Rata-rata waktu yang diperlukan dalam pencarian ini adalah 2153 milidetik untuk pencarian lokal dan 2521 milidetik untuk pencarian yang ditujukan kepada grup lain.

3.3. Uji Join grup

Peer dapat bergabung dengan grup lain dengan cara mengirimkan joinExtendedGroup kepada node alocator, waktu yang diperlukan tiap peer untuk join pada grup lain, diuji cobakan dengan hasil dibawah ini :

Tabel 4. Kecepatan join grup lain

Nama Peer	Waktu 1	Waktu 2	Rata-rata
EdgePop1	6700	5770	6235
EdgePop2	6450	6520	6485
EdgePop3	5620	6640	6130
EdgePop4	7430	7820	7625
EdgePop5	7210	8740	7975
EdgePop6	6730	6710	6720
RdvPop1	7710	8400	8055
RdvPop2	6520	6630	6575
EdgeRock1	8840	5780	7310
EdgeRock2	6630	8310	7470

EdgeRock3	8540	8540	8540
EdgeAlter1	5720	5880	5800
EdgeAlter2	6690	6120	6405
EdgeAlter3	5760	5960	5860
RdvAlter1	5870	6120	5995
EdgeHard1	6010	6520	6265
EdgeHard2	6870	7010	6940
EdgeHard3	7800	7940	7870
EdgeClass1	8230	8110	8170
EdgeClass2	8330	7870	8100
EdgeClass3	6740	7410	7075
Rata-Rata	6971	7085	7028

Dari hasil uji coba diatas dapat diambil kesimpulan waktu yang diperlukan oleh peer untuk bergabung dengan sebuah grup external adalah kira-kira 6-7 detik, sedikit berbeda . dengan waktu yang diperlukan peer untuk koneksi kedalam system. Perdeaan waktu yang mendasar dikarenakan traffik jaringan yang kadang padat kadang tidak.

3.4. Uji download

Dalam uji proses download, dibagi dengan 2 skenario, download file yang berbeda dan download 1 buah secara simultan (1 resource didownload banyak orang). Hasil yang diamati adalah rata-rata transfer data dan waktu. Hasil uji coba ada dibawah ini :

Tabel 5. Kecepatan dan transfer rate download

Nama Peer	Waktu	Transfer Rate
EdgePop1	5310	1420 KB/s
EdgePop2	5470	1022 KB/s
EdgePop3	4562	987 KB/s
EdgePop4	6508	802 KB/s
EdgePop5	5564	1340 KB/s
EdgePop6	7549	1440 KB/s
RdvPop1	5420	1029 KB/s
RdvPop2	4550	1120 KB/s
EdgeRock1	6070	876 KB/s
EdgeRock2	8090	854KB/s
EdgeRock3	5570	1021 KB/s
EdgeAlter1	5430	1025 KB/s
EdgeAlter2	6503	988 KB/s
EdgeAlter3	4442	1420KB/s
RdvAlter1	3040	1720 KB/s
EdgeHard1	4958	1640 KB/s
EdgeHard2	5430	1211 KB/s
EdgeHard3	7023	932 KB/s
EdgeClass1	7540	810 KB/s
EdgeClass2	4560	1429 KB/s
EdgeClass3	5644	1289 KB/s

Dari dua skenario diatas dapat diambil kesimpulan bahwa waktu dan transfer rate bergantung pada kemampuan tiap- tiap peer untuk memberikan layanan kepada peer yang lain. Juga bergantung dari seberapa banyak peer yang meminta layanan.

4. Kesimpulan

Di dalam sistem ini digunakan konsep *semantic overlay networks* didalam membangun jaringan peer-to-peer. Setelah melakukan scenario percobaan uji fungsionalitas, dapat diperoleh kesimpulan sebagai berikut :

- Kecepatan proses pencarian resource, join grup download untuk file yang berbeda tidak ditentukan oleh banyak peer yang tergabung kedalam sistem. Sedikit atau banyak peer yang tergabung dengan sistem, kecepatannya sama.

- Pencarian dalam ruang lingkup grup sendiri akan memperkecil beban jaringan. Hal ini dikarenakan peer-peer yang berada diluar grup tidak akan menerima query message.
- Sebuah resource yang didownload secara bersamaan akan berkurang kecepatan transfernya bisa dibandingkan dengan download tunggal, hal ini dikarenakan peer host memerlukan lebih banyak memori dalam proses transfernya.

DAFTAR PUSTAKA

- [CRO02] Crowcroft,Jon.,Moreton,Tim.,Pratt,Ian.,Twigg,Andrew.2002.**Peer-to-peer Technology**.
- [WAH03] Wahono, Romi Satria.2003.**Software P2P: Riwayatmu Dulu dan Nasibmu Kini**. IlmuKomputer.com.
- [WIL02] Wilson, Brendon J.2002.**JXTA**.New Riders Publishing.
- [CRE03] Crespo, Arturo.,Molina, Hector Garcia.2003. **Semantic Overlay Networks for P2P System**.Stanford University.
- [WIJ07] Wikipedia, Juni.2007.**Genre**.<URL: <http://en.wikipedia.org/wiki/Genre> >
- [WIM07] Wikipedia, Mei.2007.**Semantic**.<URL: <http://en.wikipedia.org/wiki/Semantic> >
- [WIK07] Wikipedia, April.2007.**Music Genre**. <URL: http://en.wikipedia.org/wiki/Music_Genre >
- [WIA07] Wikipedia, April.2007.**ID3**.<URL: <http://en.wikipedia.org/wiki/ID3> >
- [SUN07] Sun, Micro.2007.**JXTA v2.3.x: Java Programmer's Guide**.Sun Microsystem,Inc.
- [DAR04] Darwin, Ian F.2004.**Java Cookbook, 2nd Edition**.O'Relly
- [WIB07] Wibisono, Waskito.2007.**An Overview of Peer-to-Peer**. Informatika ITS
- [SUT03] Sutrisno, Eko.2003.**Proyek JXTA**. IlmuKomputer.Com
- [LI204] Li, Mei.,Lee, Wang Cien., Sivasubramaniam, Anand.**Semantic Small World: An Overlay Netwok for Peer-to-Peer Search**. Pennsylvania State University
- [YUL07] Hendriyana, Yulian F.2007.**Metadata**. <URL: <http://yulian.firdaus.or.id/2007/02/06/metadata/> >