

Perancangan Sistem Pencarian File Dokumen Dengan Agen Bergerak Metode *Branching* Pada Sistem Terdistribusi

Muhammad Hasbi ¹⁾

Abstract

Existence of the file documents in an agency is required by the leadership. The position of the file can be spread over several units (sections) in the agency, requiring the search process to determine the existence of the file. Search process is performed using mobile agent that is aglet. Aglet can move to form a 'slave' aglet according to its function as work aglet to search the files in the units with Branching. Designing a search system files on distributed systems using the aglet has been created to produce an analysis aglet role in the search process in a distributed system.

Keywords : Search, mobile agent, aglet, distributed, branching

I. Pendahuluan

Penyimpanan dokumen dalam suatu instansi sudah merupakan pekerjaan rutin, keberadaan dokumen dalam hal ini berupa suatu file yang di simpan dalam media penyimpanan tertentu sudah suatu hal yang sering dilakukan suatu instansi. File dokumen ini sangat penting untuk menunjang proses kegiatan di dalam instansi, dalam hal ini file dokumen ini berupa standard operating prosedur (SOP) di instansi tersebut. File sop dalam institusi berada tersebar di unit-unit yang harus diketahui keberadaannya, sehingga diperlukan suatu proses untuk mengetahui keberadaan file sop tersebut. Proses pencarian merupakan cara untuk mengetahui keberadaan file sop yang tersebar tersebut. Proses pencarian pada masalah ini dilakukan dengan menggunakan agen bergerak aglet .

¹⁾ Staf Pengajar STMIK Sinar Nusantara Surakarta

II. Tujuan

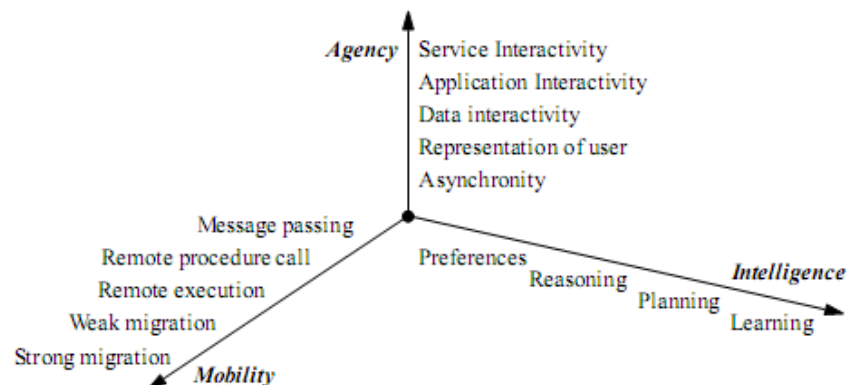
Merancang system pencarian file dokumen di jaringan sistem terdistribusi dengan agen bergerak (aglet).

III. Metode Penelitian

- a. Studi pustaka
Mengambil informasi dari sumber pustaka yang berhubungan dan mendukung system agen bergerak yang sedang dibangun.
- b. Analisis
Mengkaji lebih jauh berdasarkan referensi yang diperoleh permasalahannya dan merancang solusinya.
- c. Desain
Merancang konsep solusinya berdasarkan tahap analisis. Tahap analisis dan desain ini dapat berjalan bersamaan untuk memperoleh solusi yang paling tepat.
- d. Implementasi
Mengimplementasikan konsep yang diperoleh dengan instansi virtual berdasarkan teori agen bergerak yang dibahas sebelumnya.

IV. Pengertian aglet

a. Agen mobile

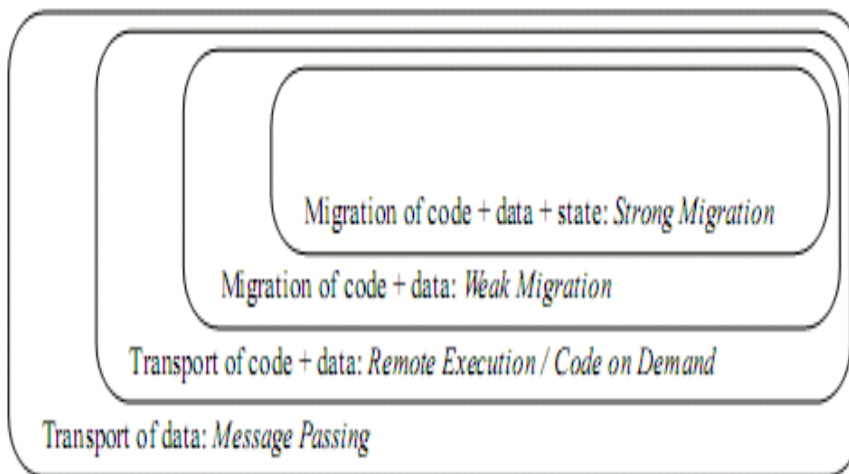


Gambar 1. Area software agen
(Kurt Rothermel and Markus Schwehm, 1998)

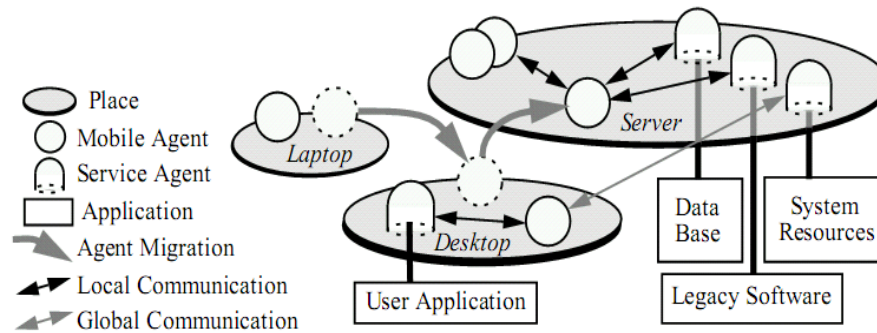
Dimensi 1: **Cerdas**: Agen cerdas dapat di klasifikasikan menurut kemampuan untuk mengekspresikan preference, kepercayaan, dan emosi, dan kemampuan untuk melaksanakan tugas dengan penalaran, perencanaan, pembelajaran.

Dimensi 2: **Agen**: Tingkatan autonomous dan otoritas pribadi dalam agen dan dapat diukur paling tidak dengan jumlah interaksi secara alami antara agen dan entitas lain dalam sistem. Minimum suatu agen harus berjalan secara asinkronis. Tingkatan agen ditangani jika suatu agen merepresentasikan user dalam berbagai cara. Lebih jauh lagi agen dapat berinteraksi dengan data, aplikasi, layanan dan agen lain.

Dimensi 3: **Mobilitas (bergerak)**: Goal dari mobilitas adalah remote aksi dan mobilitas data dan komputasi. Agen mobile dapat diklasifikasikan menurut cakupannya menjadi agen desktop (agen statik), agen intranet dan agen internet (agen jaringan).



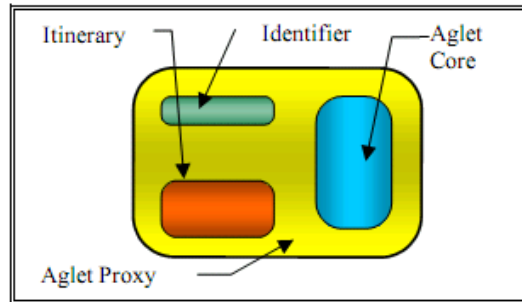
Gambar 2. Tingkatan pergerakan
(Kurt Rothermel and Markus Schwehm, 1998)



Gambar 3: Sistem agen bergerak
(Kurt Rothermel and Markus Schwehm, 1998)

Seperti diilustrasikan dalam Gambar 3, sistem agen terdiri dari sejumlah (abstrak) tempat, menjadi rumah berbagai layanan. Agen merupakan entita aktif, yang dapat berpindah dari satu tempat ke tempat lain untuk bertemu agen lain dan untuk mengakses layanan yang disediakan di tempat-tempat lain. State dan kode agen tersebut dipindahkan ke tempat baru ketika terjadi migrasi agen. Tempat menyediakan lingkungan yang aman untuk melaksanakan agen lokal serta pengunjung agen. Beberapa model agen membedakan antara agen mobile dan apa yang disebut agen layanan. Layanan agen diam dan antarmuka layanan yang tersedia di tempat. Layanan tersebut mungkin meliputi layanan sistem, seperti file atau akses direktori, serta tingkat layanan aplikasi, seperti pemesanan hotel atau jasa pengiriman bunga. Layanan agen encapsulate sembarang layanan dan mewakili mereka dalam dunia agen. Dari segi teknis, agen layanan peta permintaan layanan disajikan dalam bahasa agen ke antarmuka pelayanan individual. Catatan bahwa konsep ini memungkinkan sistem warisan untuk dimasukkan.

b. Aglet

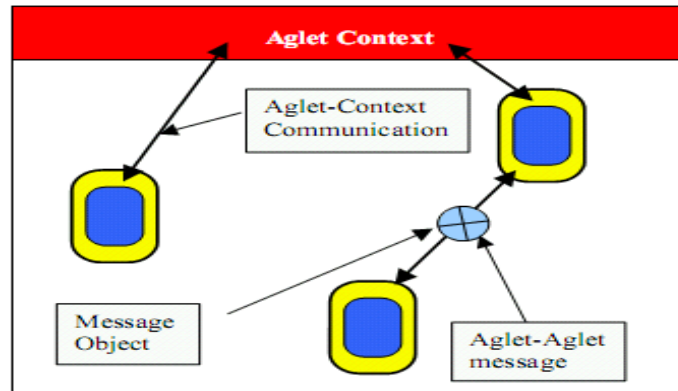


Gambar 4. Struktur Aglet
(P.E.Clements, T.Papaioannou, J.Edwards, 1997)

Aglet dapat bergerak sekitar jaringan sesuai dengan rencana perjalanannya (itinerary). Aglet workbench mendukung mobilitas dan itinerary.

Struktur aglet terdiri dari dua bagian yang terpisah yaitu aglet inti dan aglet proxy. Inti merupakan jantung dari aglet dan mengandung semua variabel dan method yang dimiliki aglet. Inti ini menyediakan melalui interface yang memungkinkan aglet berkomunikasi dengan lingkungannya.

Aglet inti dibungkus oleh aglet proxy yang bertindak sebagai pelindung melawan dari usaha akses langsung method dan variabel privatenya, dan dapat juga menyembunyikan lokasi aglet sebenarnya dari aglet jahat. Aglet juga dapat mengandung identifier dan itinerary.



Gambar 5. Lingkungan Aglet
(P.E.Clements, T.Papaioannou, J.Edwards, 1997)

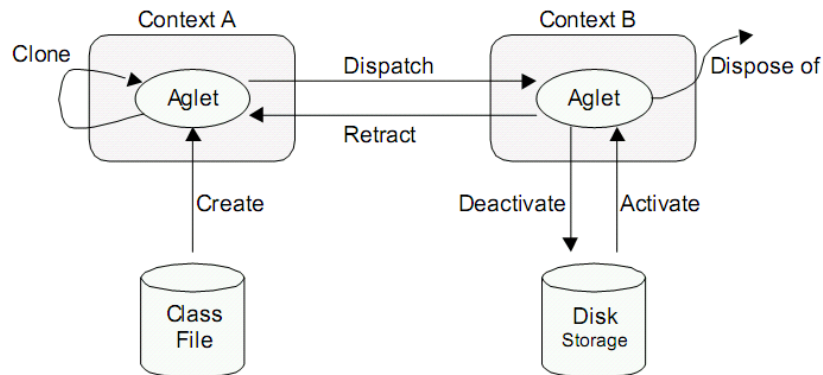
Salah satu perbedaan dengan sistem melewati pesan secara transisional dengan sistem berbasis agen. Agen mobile dapat diijinkan merubah tingkahlaku pada server tanpa sebelum persetujuan pemilik server.

Tidak seperti kebanyakan agen mobile yang melakukan perjalanan di internet, aglet tidak dapat bermigrasi ke sistem host yang tidak mempunyai server kompatible, ke dalam dimana aglet dapat dimasukkan. Objek server ini dikenal sebagai Context, bertindak sebagai gudang (warehouse) atau tempat kerja dimana aglet dari semua tipe dapat berkomunikasi dengan aglet lain. (lihat gambar). Suatu Context merupakan objek stasioner yang menyediakan alat untuk hosting dan mengolah aglet dalam suatu lingkungan yang mengamankan dari aglet jahat.

Melalui Context ini aglet dapat mengambil informasi tentang lingkungannya dan mengirim dan menerima pesan ke lingkungannya dan ke aglet lain yang sedang aktif berdasarkan kedekatannya.

Jika usaha yang dibuat aglet jahat untuk melewati aglet proxy dan mengakses data private, menager keamanan akan mengesampingkan usaha tersebut dan menghentikan akses aglet jahat tersebut.

c. Model Siklus aglet



Gambar 6: Model siklus hidup Aglet
(Danny B. Lange and Mitsuru Oshima, 1998)

Penciptaan. Penciptaan sebuah aglet terjadi dalam konteks. The aglet baru ditentukan identifiernya, dimasukkan ke dalam konteks, dan diinisialisasi. aglet ini mulai dijalankan segera setelah telah berhasil diinisialisasi.

Kloning. Kloning dari sebuah aglet menghasilkan salinan hampir identik dari aglet asli dalam konteks yang sama. Satu-satunya perbedaan adalah penentuan identitasnya dan fakta bahwa eksekusi ulang dalam aglet baru. Perhatikan bahwa thread eksekusi tidak diklon.

Mengirimkan. Mengirimkan sebuah Aglet dari satu konteks ke yang lain akan menghapusnya dari konteks saat ini dan memasukkannya ke dalam konteks tujuan, di mana ia akan restart eksekusi (pelaksanaan thread tidak bermigrasi). Kami mengatakan bahwa aglet telah “didorong” ke konteks baru.

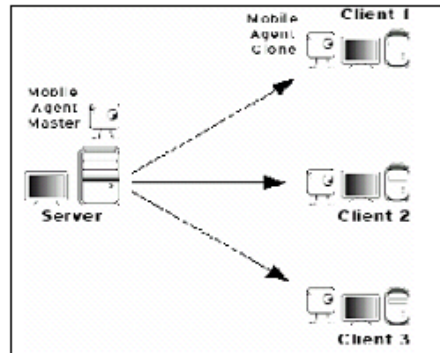
Pencabutan. Pencabutan dari sebuah aglet akan menarik (menghilangkan) dari konteks saat ini dan memasukkannya ke dalam konteks yang meminta pencabutan.

Aktivasi dan deaktivasi. Deaktivasi aglet adalah kemampuan untuk sementara menghentikan pelaksanaan dan menyimpan kondisi di penyimpanan sekunder. Aktivasi aglet suatu akan mengembalikannya dalam konteks.

Pembuangan. Pembuangan suatu aglet akan menghentikan eksekusi dan menghapusnya dari konteks saat ini.

Pesan. Pesan antara aglets melibatkan pengiriman, penerimaan, dan penanganan pesan sinkron maupun asinkron.

d. Pola Branching



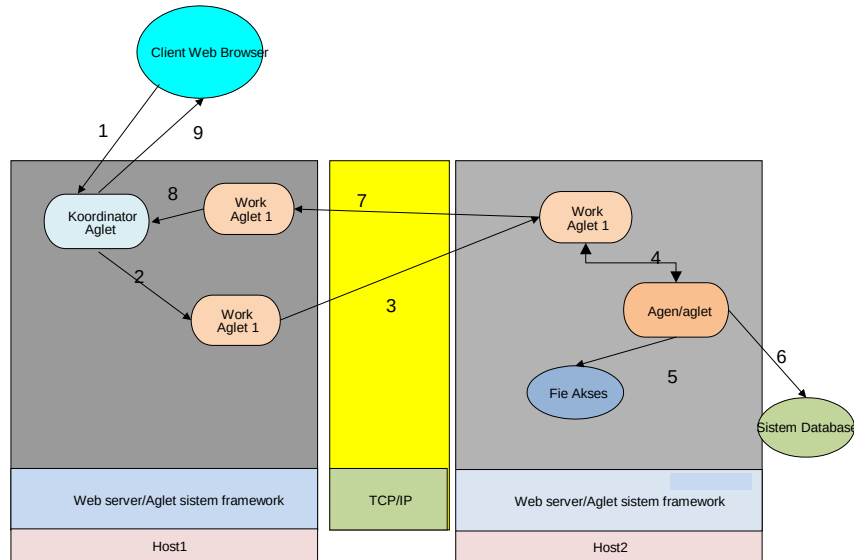
Gambar 7: Pola Perpindahan *Branching*

Gambar 7 pola ini, *mobile agent* juga dengan sebuah panduan berupa daftar *host* yang harus dikunjungi secara terurut namun tidak mengunjunginya satu per satu melainkan menggandakan diri sebanyak jumlah *host* dalam daftar dan setiap *mobile agent* hasil penggandaan bermigrasi ke salah satu *host* dalam daftar dan melakukan tugasnya. Pada pola ini tidak ada *host* dalam daftar yang dikunjungi oleh lebih dari satu *mobile agent*. (Tafta Zani, Fazmah Arif Y, Dana S. Kusumo, 2007).

V. Konsep dan Perancangan

Agen dari sebuah host pindah ke host lain akan melewati lapisan ATCI yang mengacu pada transfer Agen dan antarmuka komunikasi. Dalam lapisan ATCI (dapat ATP, CORBA, RMI, dan seterusnya)

menyediakan komunikasi dasar dan perjanjian untuk agen migrasi. Agen kemudian migrasi ke Host lain melalui TCP/IP (sistem agen), lalu ke ATCI dan agen di host tujuan. (Ma Yubao, Ding Renyuan, 2009).

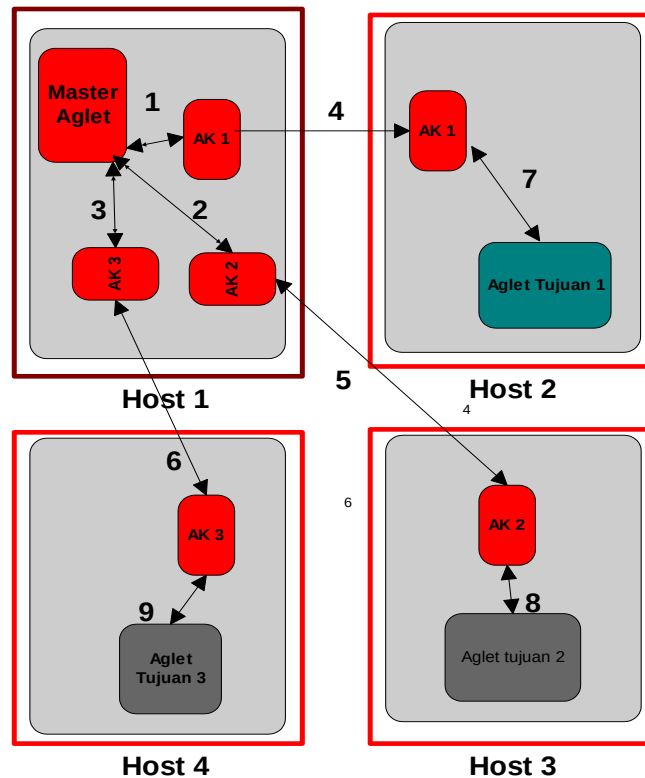


Gambar 8: konsep proses pencarian file dengan Aglet

Proses pencarian:

1. Client membuat Koordinator Aglet (KA)
2. Di Web Server Koordinator membuat Aglet Kerja (AK)
3. AK dikirim ke Remote server
4. AK berkomunikasi dengan Aglet tujuan
- 5,6 Aglet tujuan berhubungan dengan File Akses dan Sistem Database
7. AK kembali dengan membawa hasil ke asal
8. AK melaporkan hasil ke KA
9. KA menerima laporan dari Aglet yang dibentuk dan mengakumulasi laporan dan melaporkan ke Client Web Browser

VI. Implementasi



Gambar 9: Model pencarian file dengan metode Braching

Proses pencarian file dokumen dengan agen bergerak aglet ini dilakukan dengan mengikuti jumlah komputer (host) yang akan dituju dan jumlah ini telah ditentukan terlebih dahulu. Master aglet akan membuat aglet baru (*slave*) sejumlah komputer yang akan dituju seolah-olah bercabang (*branching*). Setiap *slave* akan migrasi ke *Host* yang telah ditentukan dan mencari file dokumen, hasilnya akan dilaporkan ke Master Aglet. Hasil dari beberapa *slave* (sesuai jumlah *host* yang dituju) di terima oleh master aglet yang merupakan hasil akhir pencarian file dokumen. Dalam contoh Gambar 9 di perhatikan

bahwa aglet yang bergerak ke komputer lain sudah ditentukan sejumlah 3 yaitu ke *host* 1, 2 dan 3.

Proses pencarian terlihat pada Gambar 9:

1. Master membuat slave 1 (Aglet Kerja 1 /AK 1)
2. Master membuat slave 2 (Aglet Kerja 2 /AK 2)
3. Master membuat slave 3 (Aglet Kerja 2 /AK 2)
- 4, 5, 6 AK 1, AK2, dan AK 3 di kirimkan berturut-turut ke Host 1, Host 2, dan Host 3.
7. AK 1 setelah sampai di Host 1 berkomunikasi dengan Aglet tujuan 1, untuk mencari file dokumen yang dicari di Host 1. Aglet Tujuan 1 akan berkomunikasi dengan host 1, tentang keberadaan file dokumen yang dicari. Hasil pencarian berupa keberadaan file di Host 1 akan dilaporkan ke Master Aglet melalui AK 1 dengan alur yang sama tetapi arahnya berlawanan, kemudian hasil ini akan diteruskan ke User.
8. AK 2 setelah sampai di Host 2 berkomunikasi dengan Aglet tujuan 2, untuk mencari file dokumen yang dicari di Host 2. Aglet Tujuan 1 akan berkomunikasi dengan host 2, tentang keberadaan file dokumen yang dicari. Hasil pencarian berupa keberadaan file di Host 2 akan dilaporkan ke Master Aglet melalui AK 2 dengan alur yang sama tetapi arahnya berlawanan, kemudian hasil ini akan diteruskan ke User.
9. AK 3 setelah sampai di Host 3 berkomunikasi dengan Aglet tujuan 3, untuk mencari file dokumen yang dicari di Host 3. Aglet Tujuan 3 akan berkomunikasi dengan host 3, tentang keberadaan file dokumen yang dicari. Hasil pencarian berupa keberadaan file di Host 3 akan dilaporkan ke Master Aglet melalui AK 3 dengan alur yang sama tetapi arahnya berlawanan , kemudian hasil ini akan diteruskan ke User.

VIII. Kesimpulan

Proses yang digambarkan dalam makalah ini untuk mendukung teori tentang proses kerja aglet yang dapat bergerak, inteligen, dan sebagai agen.

Makalah ini mendemonstrasikan secara logika proses pencarian file yang dilakukan oleh aglet. Pencarian file dokumen di dalam jaringan computer (jaringan local) dapat dilakukan dengan agen mobile yaitu aglet. Proses pencarian dilakukan dengan cara aglet mengkloning menjadi aglet kerja sejumlah host yang dikunjungi dan mengirimkan aglet kerja ini ke host lain untuk mencari file dokumen yang dikehendaki. Aglet ini menelusuri host yang dituju untuk mencari file yang dikehendaki, kemudian hasil pencarian dilaporkan ke Aglet asal (Master Aglet).

IX. Future Work

Proses pencarian dengan menggunakan aglet ini masih dapat dilakukan pada sistem terdistribusi dengan skala lebih luas yaitu pada internet.

X. Daftar Pustaka

- .Danny B. Lange and Mitsuru Oshima, **Mobile Agents with Java: The Aglet API**, To appear in *World Wide Web* (Baltzer Science Publishers, The Netherlands).1998.
- Kurt Rothermel and Markus Schwehm, **Mobile Agents**, In: A. Kent and J. G. Williams (Eds.): *Encyclopedia for Computer Science and Technology*, New York: M. Dekker Inc., 1998 (accepted for publication).
- Ma Yubao, Ding Renyuan, **Mobile Agent technology and Its Application in Distributed Data Mining**, 2009 First International Workshop on Database Technology and Applications. 2009 IEEE DOI 10.1109/DBTA.2009.22.
- P.E.Clements, T.Papaioannou, J.Edwards, **Aglets: Enabling the Virtual Enterprise**, Published and Presented at ME-SELA '97, ISBN 1 86058 066 1, p425.
- Tafta Zani, Fazmah Arif Y, Dana S. Kusumo, **aplikasi berbasis agent untuk deteksi perangkat keras komputer melalui jaringan lokal**, Departemen Teknik Informatika – Sekolah Tinggi Teknologi Telkom, Seminar Nasional Sistem dan Informatika 2007; Bali, 16 November 2007