

TUGAS KAPITA SELEKTA



OLEH :

DIAN OKTAVIAN

09011181320046

**SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
INDRALAYA**

2016

Tugas : Analisis Adaptive Middleware (Adaptive middleware for context-aware applications in smart-homes)

Pada “SUMMARY OF THE IOT MIDDLEWARES SUPPORTED ARCHITECTURAL REQUIREMENTS”.

1. PENDAHULUAN

konteks-Aware: Konteks adalah kunci dalam IOT dan aplikasi. Sejumlah besar sensor akan menghasilkan jumlah besar data, yang tidak memiliki nilai apapun kecuali dianalisis, ditafsirkan, dan dipahami. Konteks-Aware komputasi menyimpan informasi konteks yang berkaitan dengan sensor data, mengurangi interpretasinya. *Konteks -Aware* (terutama dalam konteks temporal dan spasial) memainkan peran penting dalam perilaku adaptif dan otonom dari hal-hal di IOT [20], [58]. Perilaku tersebut akan membantu untuk menghilangkan manusia sentris mediasi di IOT, yang akhirnya membuat lebih mudah untuk melakukan komunikasi M2M, inti unsur visi IOT ini.

Proyek *Smart-Home* di rancang untuk membayangkan rumah masa depan dengan di penuh sensor – sensor yang dapat menentukan berbagai jenis informasi dan aktifasi dari *Smart-Home* seperti lokasi dan aktivitas-sementara, aplikasi dalam penggunaan *Smart-home* ini untuk memberikan konteks sensitif pelayanan-pelayanan untuk penghuni dan untuk otomatisasi rumah *konteks-aware*. Aplikasi ini di buat bertujuan untk mempermudah dan mendukung orang-orang yang dengan masalah kesehatan, seperti hidup tua sendirian dirumah, bisa juga perbaikan gaya hidup secara umum misalnya : kulkas Pintar yang dapat memonitoring isi di dalamnya serta dapat memberi peringatan kepada penghuni rumah jika barang ataupun isi di dalamnya sudah habis atau pun hampir habis.

2. CONTEXT-AWARENESS

Context-awareness adalah kemampuan aplikasi untuk menyesuaikan diri dengan konteks user. Konteks Seorang pengguna dapat didefinisikan secara luas sebagai keadaan atau situasi di mana tugas komputasi berlangsung [6]. Salah satu konteks yang paling umum adalah lokasi pengguna (atau obyek yang menarik). Dalam *smart-home*, lokasi dapat diperoleh dengan menggunakan berbagai jenis sensor alternatif yang berbeda, termasuk ultrasonik, RFID-tag, kamera video dan bahkan sensor tekanan di lantai [5]. Kualitas (yang kuantitatif-masing aplikasi-spesifik) dari informasi lokasi yang diperoleh oleh sensor yang berbeda akan tetapi berbeda. Misalnya, ultrasonik.

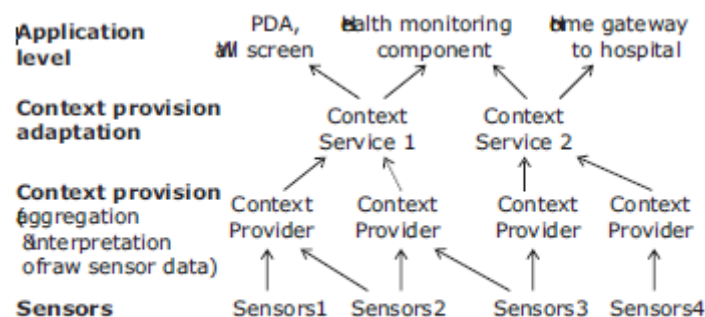


Figure 1: Layers of context provision in our middleware

Kita adapat menentukan lokasi dengan presis hingga 3 cm, sedangkan RF lateration terbatas pada 1-3 m. Dengan demikian, kita dapat mendefinisikan sifat yang kita sebut Kualitas Konteks (QoC) atribut yang menjadi ciri kualitas konteks data yang diterima. QoC

adalah, seperti yang akan kita lihat dalam Bagian 4.3, penting untuk middleware dalam memilih alternative terbaik di antara yang tersedia untuk memberikan jenis konteks tertentu.

2.1 Quality konteks

Sementara berbagai jenis konteks akan memiliki QoC atribut specific, ada atribut umum tertentu yang dapat digunakan untuk kebanyakan konteks[2]. Berdasarkan identifikasi atribut umum berikut:

Tindakan Presis seberapa akurat informasi konteks akurat realitas, misalnya lokasi presis benar. Tindakan kebenaran probabilitas bahwa sebagian informasi konteks yang Misalnya, menentukan informasi postur badan saat seseorang (duduk, berdiri, berbaring di lantai dalam kesulitan) menggunakan kamera video memiliki probabilitas yang berbeda dari kebenaran dengan menggunakan sensor tekanan di benda –benda perabotan. Resolusi menunjukkan granularity informasi. Hal ini dapat misalnya di tunjukkan dengan kita memasang sebuah HOTSPOT untuk mendeteksi suhu tinggi yang berada pada ruangan dapur, seperti suhu yang di keluarkan oleh alat seperti (oven, kompor, pemanggang roti) yang tidak dapat ditangkap oleh sensor suhu di ruangan jika tidak di bantu dengan fitur tambahan lainnya. Up-to-dateness menentukan usia konteks informasi. Yang juga menarik adalah seberapa besar kemungkinan pengukuran masih akurat. Refresh rate berkaitan dengan up-to-dateness, dan menggambarkan seberapa sering atau yang kita inginkan untuk menerima pengukuran baru. QoC berbeda dari QoS, karena konteks informasi memiliki metrik kualitas bahkan ketika itu tidak disediakan sebagai layanan untuk setiap klien. Dalam middleware ini, penyedia konteks perlu untuk menentukan QoC atribut untuk informasi konteks yang diberikan. Atribut ini dapat bervariasi dari waktu ke waktu karena itu harus diperbarui secara berkala.

3. STRUCTURE OF OUR MIDDLEWARE

Pada bagian bawah adalah sensor yang mana untuk memberikan data sensor yang belum diolah. Ini bisa menjadi jaringan wireless sensor, lanjutan ke ultrasonik untuk lokasi, tag RFID untuk identifikasi, kamera video untuk pelacakan, atau orang. Ini menghasilkan data sensor baku, Data-data ini diteruskan naik satu tingkat dalam rangka untuk penyedia konteks.

Contexts Proprietary (CP) merupakan komponen (software atau hardware) dan menafsirkan data sensor untuk menghasilkan beberapa tingkat konteks yang lebih tinggi, misalnya lokasi, identitas, jenis kegiatan, kesehatan kondisi seseorang. CP dapat mengakses lebih dari satu kondisi dari kelompok yang sama didalam sensor. Demikian pula untuk satu CP dapat menggunakan data sensor dari sekelompok sensor, sebagai data redundancy, Ini agak lebih umum dibandingkan dengan pendekatan yang khas dalam Toolkit Konteks, di mana salah satu sensor biasanya sama dengan satu konteks widget, dan komponen terpisah yang disebut agregator dan interpreter mengumpulkan dan membedakan data sensor. Namun, sebagai contoh, kita bisa membayangkan Toolkit Konteks yang digunakan untuk mengimplementasikan lapisan penyediaan konteks middleware. Dalam kasus apapun CP untuk dibagi menjadi sub - komponen dengan kemampuan dapat digunakan kembali.

4. Analisa

Toolkit adalah sebuah kerangka kerja yang bertujuan untuk memfasilitasi pengembangan dan penyebaran aplikasi *Contexts-Aware*. misalnya layanan lokasi, dari sensor yang memperoleh data yang diperlukan untuk memberikan layanan. Dengan demikian, dalam lingkungan yang berbeda, penyedia jenis konteks dapat ditukar dengan pelayanan lain yang mungkin menggunakan jenis fundamental berbeda dari sensor, namun aplikasi tidak perlu dimodifikasi untuk mengakomodasi sumber yang berbeda dari konteks yang sama. Namun, tidak ada mekanisme yang memungkinkan layanan konteks untuk beradaptasi dan bereaksi

terhadap kegagalan atau degradasi infrastruktur sensor yang mendasari, misalnya dengan beralih ke cara alternatif memperoleh jenis yang sama dari konteks. Bahkan, situasi ini di mana beberapa cara memperoleh konteks yang sama mungkin dinamis hadir dalam sistem ini tidak dianggap. dimulai dengan asumsi ini dan kemudian membahas masalah manajemen adaptasi. Ada juga pendekatan middleware lain untuk pada konteks kesadaran middleware. Misalnya, [7] menjelaskan infrastruktur yang menyediakan dukungan konteks kesadaran untuk aplikasi. Menggunakan predikat logika orde pertama untuk model konteks dan memungkinkan pengurang konteks tingkat yang lebih tinggi dengan menggunakan pendekatan berbasis aturan. Efektif, aplikasi dapat mendelegasikan konteks kesadaran untuk infrastruktur mendatang, yang mengurus memperoleh konteks, mengevaluasi aturan (yang disediakan oleh aplikasi) dan melakukan 1 tindakan. Sebaliknya, dalam aplikasi solusi delegasi ke middleware pilihan penyedia konteks yang paling tepat. Seperti adaptasi di middleware, Bhatti dan Ksatria [1] telah diusulkan middleware QoS untuk distribusi multimedia di Internet yang juga menggunakan fungsi utilitas untuk menentukan alternatif terbaik untuk memberikan media untuk aplikasi, untuk Misalnya memilih codec yang paling tepat untuk pengiriman audio yang mempertimbangkan sifat sewa skr jaringan, yaitu bandwidth dan latency. Bagaimanapun, konteks adaptasi middleware mereka sangat berbeda. QoS middleware biasanya mengatasi masalah ini mengadaptasi ery telah dijalankan data di seluruh jaringan dalam menanggapi perubahan sifat jaringan. Dalam kasus kami, kami memilih sumber data yang diberikan berbagai alternatif dengan kualitas yang berbeda-beda (di mana kualitas secara kuantitatif aplikasi spesifik konsep). Selain itu, alternatif kami tidak tetap, tapi secara dinamis ditentukan oleh penyedia konteks saat ini tersedia dalam jaringan (pos- sibly dinamis) atribut QoC. Hal ini memerlukan protokol penemuan layanan yang bereaksi dengan cepat untuk perubahan penyedia, sehingga mesin adaptasi memiliki pandangan yang akurat dari penyedia. Terlebih lagi, sementara Bhatti dan Ksatria menjelaskan fungsi utilitas tetap untuk memutuskan adaptasi, kita menggunakan fungsi utilitas aplikasi-spesifik, sebagai QoC sebuah aplikasi keinginan khusus untuk tujuan dan pelaksanaan aplikasi, dan karena itu tidak dapat statis diasumsikan di muka, karena dengan sifat pengiriman jaringan, tetapi harus baik disediakan oleh aplikasi atau belajar melalui umpan balik penerapan. Aspek desain menarik dalam middleware adalah bagaimana fungsi utilitas mereka untuk mencegah negara menyusun, yaitu situasi di mana QoS jaringan dekat utilitas fungsi threshold tion, menyebabkan middleware untuk terus menyusun tween dua lokasi untuk adaptasi.

Tujuan pembuatan Smart-home adalah untuk memperkenalkan gagasan tentang komputasi otonom dalam middleware untuk aplikasi *konteks-aware*. Sementara ada banyak penelitian pada komputasi otonom, belum terealisasi dalam proses pembuatan spesifikasi diterapkan untuk middleware untuk aplikasi *konteks-aware*. Namun, khususnya di lingkungan *Smart-home*, manajemen dari sistem ini sangat penting karena kita tidak dapat berasumsi bahwa akan menjadi hasil administrator teknis sepanjang jam untuk memecahkan masalah yang mungkin timbul dan menyempurnakan sistem. Dalam solusi ada usulan untuk mengasumsikan bahwa CP dapat memperkirakan QoC (atau setidaknya sebagian besar dari Qoc). Ini tetap dilihat dalam praktek seberapa baik asumsi ini berlaku. Meskipun middleware kami dirancang dalam konteks SmartCar skenario aplikasi rumah dan konteks kesadaran, bisa *generalised* untuk kasus di mana ada beberapa penyedia layanan untuk layanan yang sama, misalnya jasa percetakan, dan berdasarkan keinginan aplikasi memilih penyedia layanan yang optimal untuk setiap aplikasi, misalnya printer berkualitas terbaik. Kemudian, jika layanan saat ini digunakan (misalnya printer) harus gagal (misalnya kertas macet, tidak ada toner) middleware dapat secara otomatis beralih ke yang terbaik tersedia. Secara umum, dapat di percaya fungsi utilitas untuk menjadi lebih cocok untuk penyediaan layanan adaptif

dibandingkan permintaan berbasis penemuan layanan, karena semua alternatif yang tersedia dengan utilitas dan memungkinkan middleware untuk mandiri menentukan alternatif terbaik pada kegagalan layanan.

5. Referensi

- [1] S. N. Bhatti and G. Knight. Enabling QoS adaptation decisions for Internet applications. *Computer Networks*, 31(7):669–692, 1999.
- [2] T. Buchholz, A. Kupper, and M. Schiffers. Quality of context: What it is and why we need it. In *Proceedings of the Workshop of the HP OpenView University Association 2003 (HPOVUA 2003)*, Geneva, 2003.
- [3] A. K. Dey and G. D. Abowd. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16:97–166, 2001.
- [4] E. Frank and I. H. Witten. Selecting multiway splits in decision trees.
- [5] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer, IEEE*, 34(8):57–66, Aug. 2001.
- [6] S. Meyer and A. Rakotonirainy. A survey of research on context-aware homes. In *Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*, pages 159–168. Australian Computer Society, Inc., 2003.
- [7] A. Ranganathan and R. H. Campbell. An infrastructure for context-awareness based on first order logic. *Personal Ubiquitous Comput.*, 7(6):353–364, 2003.
- [8] S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2nd edition, 2003.