

# LECTURE NOTES

## MOBI8001 – Mobile Technology & Cloud Computing

### Topik 04 - Offloading in Mobile Cloud Computing

## LEARNING OUTCOMES

1. Peserta mampu menerapkan konsep mobile cloud computing dalam menyelesaikan masalah-masalah teknis di dunia nyata.
2. Peserta memiliki kemampuan dalam menganalisa arsitektur, platform, dan teknologi-teknologi pendukung dari mobile cloud computing.
3. Peserta mampu mengevaluasi kemajuan dan tantangan penelitian dari teknologi mobile cloud computing

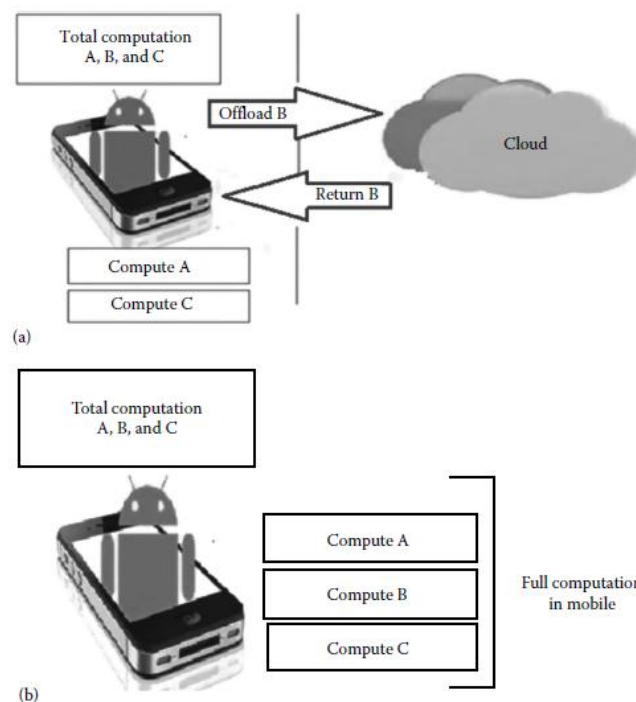
### OUTLINE MATERI :

1. Pendahuluan
2. Offloading Decision
3. Tipe-tipe Offloading
4. Topologi Offloading
5. Offloading di Cloud Computing dan Mobile Cloud Computing
6. Komputasi Adaptif dari Proses Offloading
7. Metode Pemilihan Jalur Cloud
8. Mobile Data Offloading dengan Komputasi oportunis
9. Arsitektur dari Mobile Cloud Computing
10. Persyaratan Offloading Data
11. Kesimpulan

## ISI MATERI

### A. Pendahuluan

Ketika menyangkut perangkat mobile, komputasi, sumber daya penyimpanan mobile, dan kendala daya yang serius karena masa pakai baterai yang terbatas merupakan kontributor utama yang menyebabkan bottleneck. Offloading sebagai komputasi terdistribusi dapat digunakan untuk mengatasi masalah ini. Offloading adalah solusi untuk meningkatkan kemampuan sistem mobile ini dengan melakukan migrasi komputasi ke komputer dengan resource (kecepatan CPU, memory) yang lebih besar, seperti server.



**FIGURE 4.1**  
(a) Computation process performed using offloading to cloud. (b) Computation process performed without using offloading.

**Gambar 1** Proses komputasi dengan menggunakan (a) offloading (b) tanpa offloading

## B. Offloading Decision

Offloading memigrasikan proses komputasi ke komputer dengan resource yang lebih besar. Oleh karena itu dibutuhkan proses pengambilan keputusan apakah perlu dilakukan migrasi dan proses komputasi mana yang dimigrasi. Kegunaan dari offloading decision adalah untuk

### 1. Meningkatkan kinerja sistem

Offloading menjadi solusi menarik untuk memenuhi kebutuhan waktu respon pada sistem mobile karena aplikasi menjadi semakin kompleks, sementara sistem mobile memiliki kecepatan komputasi yang terbatas. Sebagai contoh, robot navigasi mungkin perlu mengenali objek sebelum bertabrakan dengan objek; Jika prosesor robot terlalu lambat, perhitungan mungkin perlu di-offload ke server.

### 2. Menghemat energy

Akhir-akhir ini, mengembangkan sistem smartphone yang memiliki fungsi dalam jumlah besar mengkonsumsi lebih banyak tenaga dan memperpendek masa pakai baterai. Offloading dapat memperpanjang masa pakai baterai dengan mentransfer bagian perhitungan intensif energi ke server.

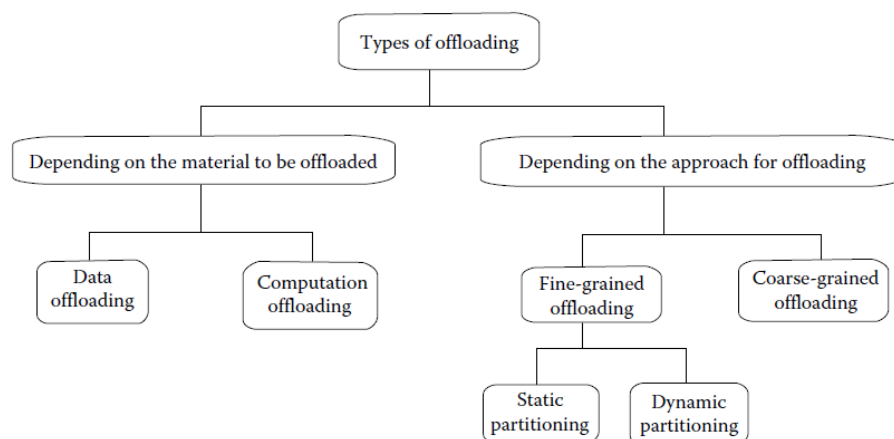


FIGURE 4.3  
Types of offloading.

## Gambar 2 Tipe-tipe offloading

### C. Tipe-tipe offloading

Seperti ditampilkan oleh gambar 2, proses offloading dapat dikategorikan ke dalam dua grup yaitu

- **Berdasarkan material yang di-offload**

Berdasarkan material yang dioffload, ada 2 dua jenis offloading:

1. **Offloading data:** pada skema ini, data dimigrasikan dari satu jaringan padat ke jaringan lain
2. **Offloading komputasi:** pada skema ini, proses komputasi yang berat dimigrasikan dari perangkat mobile ke server dengan cloud untuk meningkatkan performa dan masa pakai baterai.

- **Berdasarkan pendekatan pengurangan waktu**

Berdasarkan pendekatan pengurangan waktu, ada 2 jenis offloading:

1. **Full offloading/coarse-grained offloading:** program penuh dimigrasikan ke infrastruktur dan programmer selanjutnya tidak dapat memodifikasi source-codenya.
2. **Partial offloading/fine-grained offloading:** mempartisi sebuah program dan hanya membebani bagian aplikasi yang membutuhkan daya besar.

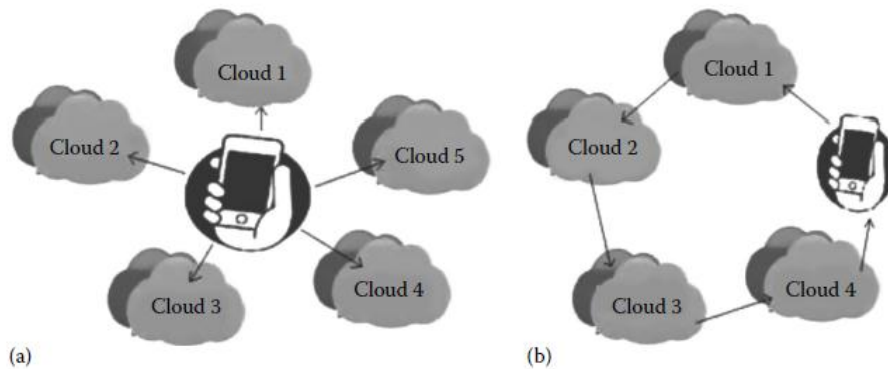


FIGURE 4.5  
(a) MCC offloading star topology and (b) MCC offloading ring topology.

### Gambar 3 Topologi Offloading

#### D. Topologi Offloading

Ada 2 jenis topologi offloading, yaitu topologi star dan ring seperti ditunjukkan oleh gambar 3. Pada topologi star, satu perangkat mobile berkomunikasi dengan banyak server secara bersamaan sehingga banyak waktu dan energy yang terpakai untuk komunikasi dengan banyak server. Di sisi lain, pada topologi ring, dua nodes berkomunikasi satu sama lain dan saling tergantung satu sama lain. Ketika kegagalan terjadi di tengah-tengah nodes, maka program tidak bisa dieksekusi.

#### E. Offloading di Cloud Computing dan Mobile Cloud Computing: persamaan dan perbedaan

Kata “mobile computing” mempunyai arti proses komputasi yang terkait dengan perangkat mobile. Untuk lebih jelasnya, mobile computing berkaitan dengan data dan komputasi yang terkait dengan perangkat yang tidak fixed dan dapat merubah jaringan ketika diperlukan. Di lain pihak, cloud computing adalah proses komputasi yang melingkupi perangkat mobile dan non-mobile. Sehingga bisa dikatakan cloud computing adalah super set

Dengan analogi yang sama, offloading di cloud computing adalah superset dari offloading di mobile cloud computing. Semua arsitektur dan algoritma yang digunakan adalah sama, perbedaan yang adalah mengenai mobilitas dari perangkat komputasi di mobile cloud computing. Pada MCC, perubahan jaringan seperti ditunjukkan oleh gambar 4 menyebabkan server yang diakses oleh perangkat mobile selalu berubah dan membutuhkan perhitungan energi secara konstan. Masalah optimisasi akan menjadi salah satu pertimbangan dalam kasus ini.

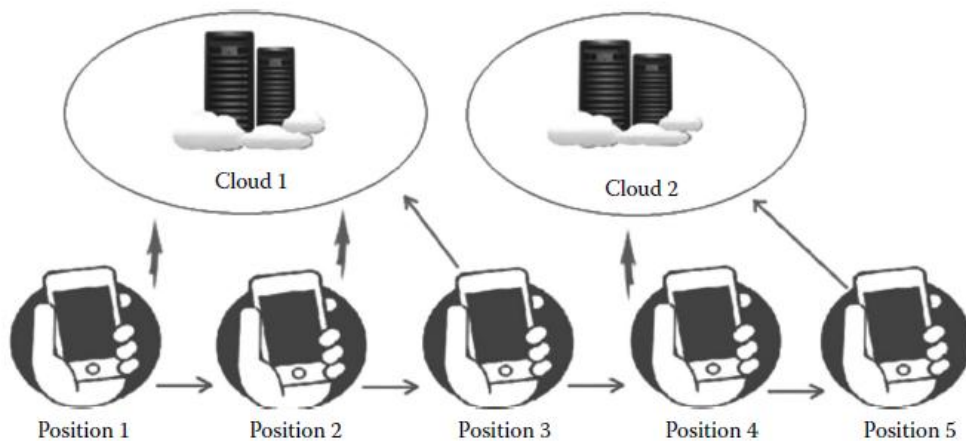


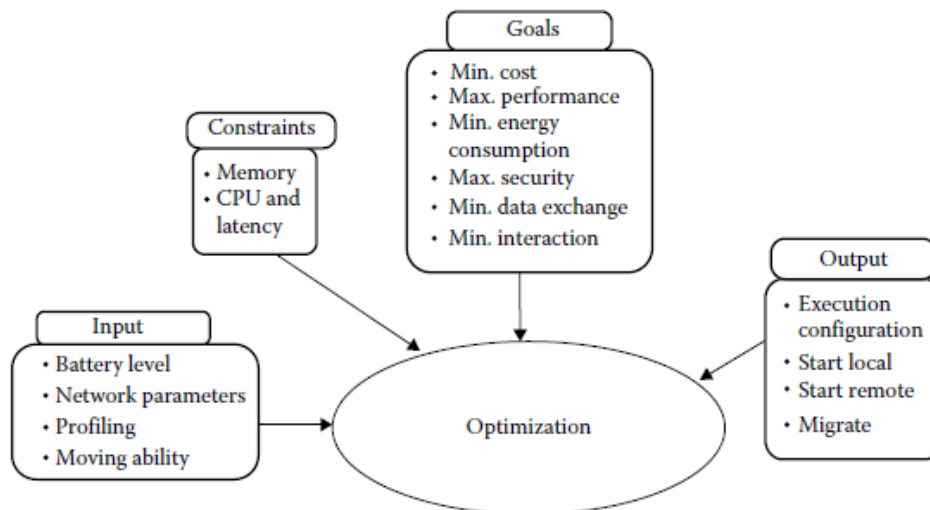
FIGURE 4.7  
Offloading in MCC deals with moving devices.

### Gambar 4 Proses offloading pada MCC

#### F. Komputasi Adaptif dari Proses Offloading

Proses offloading aplikasi intensif dari mobile platform ke remote cloud dapat mengatasi permasalahan keterbatasan masa pakai baterai dan keterbatasan kemampuan pemrosesan dari perangkat mobile. Saat ini, sebagian besar aplikasi mobile dikembangkan dengan menggunakan platform Android. Android sudah mapan sebagai platform ponsel yang paling menonjol. Mobile Augmentation Cloud Services (MACS)

adalah middleware yang memungkinkan extension secara adaptif atau optimasi, seperti yang ditunjukkan pada Gambar 4.8, penerapan aplikasi Android dari klien mobile ke dalam cloud.



**Gambar 5 Permasalahan optimisasi dari Mobile Augmentation Cloud Services**

## G. Metode Pemilihan Jalur Cloud

Ada beberapa jenis pemilihan jalur optimal dari cloud ketika lebih dari satu jalur tersedia, yaitu sebagai berikut:

1. Secara acak: pemilihan jalur cloud secara acak
2. Berdasarkan bandwidth: pilih jalur cloud dengan bandwidth tertinggi.
3. Berdasarkan tingkat kegagalan link: pilih jalur dengan tingkat kegagalan link rendah
4. Berdasarkan faktor kecepatan: pilih jalur cloud dengan faktor kecepatan tertinggi
5. Berdasarkan biaya: pilih jalur cloud dengan biaya terendah.

**Ada beberapa isu dalam pemilih jalur cloud, yaitu:**

1. Bandwidth: Tergantung kondisi koneksi wireless antara perangkat mobile dan cloud.

Ketika koneksi wireless bagus, banyak eksekusi aplikasi dan data yang bisa



dialihkan ke cloud, sebaliknya bila koneksi wireless buruk, hanya sedikit dari eksekusi aplikasi dan data yang bisa dialihkan ke cloud.

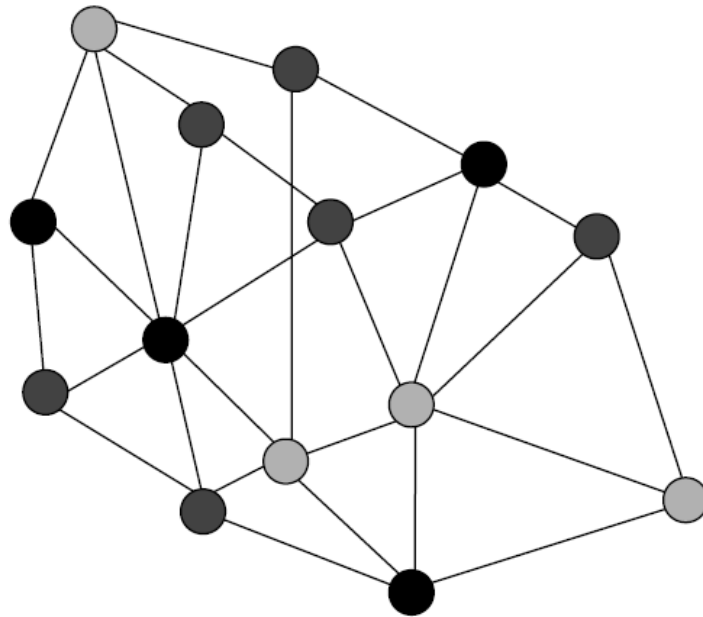
2. Harga: Bagi provider servis mobile cloud, penting untuk membangun layanan cloud dengan harga termurah
3. Kecepatan: Seberapa cepat server bisa melakukan komputasi di cloud
4. Keamanan: Pertama-tama, memindahkan semua data dan komputasi ke cloud itu berbahaya. Misalnya, melacak individu melalui data navigasi berbasis lokasi yang dibongkar ke awan. Selain itu, pengaturan keamanan dan privasi bergantung pada penyedia awan karena data disimpan dan dikelola di dalam awan.
5. Ketersediaan: berhubungan dengan kegagalan jalur dan ketersediaan cloud selama proses offloading. Kegagalan jaringan dapat terjadi karena mobilitas dari perangkat mobile dan koneksi jaringan yang tidak stabil.

#### **H. Mobile Data Offloading dengan Komputasi oportunistis**

Karena semakin populernya berbagai aplikasi untuk smartphone, jaringan seluler 3G menjadi kelebihan beban. Offloading data mobile menggunakan komunikasi oportunistik merupakan salah satu solusi yang paling menjanjikan untuk masalah ini. Dengan menggunakan komunikasi oportunistik, kita dapat mempromosikan penyiaran informasi di jejaring sosial mobile (MoSoNets) untuk mengurangi jumlah lalu lintas data seluler. Aplikasi MoSoNets telah berkembang dari situs jejaring sosial online hingga berbagai aplikasi mobile dan perangkat lunak sosial mobile.

Pada MoSoNets (gambar 7), penyedia layanan mengirimkan informasinya hanya kepada beberapa pelanggan terpilih atau menargetkan pengguna untuk meminimalkan lalu lintas data seluler dan dengan demikian mengurangi biaya operasi. Kemudian,

informasi tersebut dapat dihasilkan lebih lanjut oleh pengguna target di antara semua pelanggan lain melalui partisipasi sosial, saat smartphone mereka berada dalam jangkauan komunikasi dan komunikasi oportunistik.



**Gambar 6 MoSoNets Social Graph**

## I. Arsitektur dari Mobile Cloud Computing

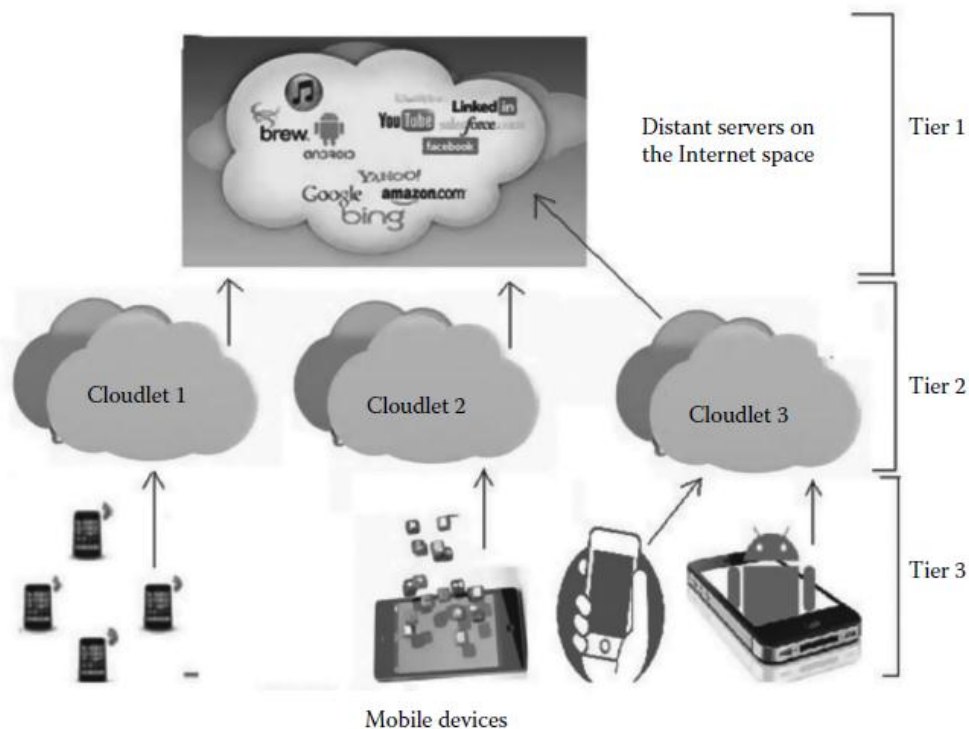


FIGURE 4.12  
Three-tier architecture of mobile cloud computing.

### Gambar 7 Arsitektur dari mobile Cloud Computing

Kasus umum arsitektur, seperti yang ditunjukkan pada gambar 7, dimaksudkan untuk mendukung penghitungan offloading untuk node seluler di mana tier 2 terdiri dari cloud lokal dengan kemampuan terbatas (cloudlet) di dekatnya dan tier 1 yang terdiri dari server cloud yang tersedia di internet. Untuk arsitektur seperti itu, ada dua jenis skenario, yaitu skenario penggunaan yang dikelola (managed) dan tidak dikelola (unmanaged).

1. Skenario penggunaan yang dikelola biasanya sesuai dengan kasus di mana penyedia layanan wireless menyebarkan infrastruktur cloudletnya sendiri di AP miliknya sendiri untuk digunakan oleh pelanggan selulernya. Oleh karena itu, WSP dapat

diharapkan untuk secara terpusat mengatur akses ke cloudlet dengan tujuan menawarkan pengalaman pelayanan yang baik kepada pelanggannya dan untuk memenuhi tujuan utilitasnya sendiri.

2. Skenario penggunaan tidak dikelola (unmanaged) sesuai dengan kasus di mana cloudlet ditambah Wi-Fi hot spot dikerahkan oleh otoritas publik atau pengusaha swasta di fasilitas seperti bandara, stasiun kereta api, bangunan umum, dan kafe untuk kepentingan warganya atau pelanggan [12]. Skenario ini merupakan perpanjangan dari rencana saat ini dimana akses Wi-Fi gratis ditayangkan hanya sebagai layanan tambahan, atau sebagai cara untuk menarik lebih banyak pelanggan dengan dasar best-effort dan tanpa usaha untuk mengatur penggunaannya.

## **J. Persyaratan Offloading Data**

Optimalisasi energi adalah salah satu alasan penting untuk offloading. Mobile cloud computing melalui offloading menghemat energi pemrosesan, yang merupakan salah satu cara yang paling signifikan untuk meminimalkan konsumsi daya. Solusi yang diusulkan untuk berbagai perubahan lingkungan komputasi mobile mempertimbangkan tiga perubahan lingkungan yang umum: tingkat kekuatan sisi klien, status koneksi, dan bandwidth. Prioritas tertinggi diberikan pada tingkat daya sisi klien karena faktor ini tidak dapat digantikan.

## SIMPULAN

Offloading melibatkan konsumsi daya dalam menjalankan aplikasi dan selama berkomunikasi dengan remote cloud. Untuk lebih memanfaatkan sumber daya di atas awan, ada trade-off antara biaya energi transmisi dan biaya energi eksekusi lokal. Meskipun banyak karya telah berusaha untuk menemukan titik keseimbangan, hanya sebagian dari studi tersebut yang berfokus pada efisiensi energi, sementara dalam banyak kasus mereka hanya berfokus pada waktu respon dan konsumsi sumber daya lainnya.

Sebagian besar penelitian menggunakan pemodelan dan simulasi. Beberapa model hanya melepaskan sebagian dari program ke server, yang mengurangi biaya transmisi, sementara pada saat yang sama membawa overhead partisi ekstra. Karya-karya lain berfokus pada transplantasi keseluruhan OS atau aplikasi runtime ke cloud karena ini memudahkan beban pada pengembang aplikasi eksternal.

## DAFTAR PUSTAKA

1. DE, Debashis. Mobile cloud computing: architectures, algorithms and applications. CRC Press, 2016.
2. Ma, Y. Cui, L. Wang, and I. Stojmenovic, Energy optimizations for mobile terminals via computation offloading, in *Second IEEE International Conference on Parallel Distributed and Grid Computing*, Solan, India, pp. 236–241, 2012.
3. K. Kumar, J. Liu, Y. H. Lu, and B. Bhargava, A survey of computation offloading for mobile systems, *Mobile Networks and Applications*, 18(1), 129–140, 2013.