OPENSTACK

Cloud Edge Computing: Beyond the Data Center (melampaui pusat data)

Contents

1. PENGANTAR
2. APAKAH CLOUD EDGE COMPUTING?
3. EKSPLORASI MENDALAM PERTIMBANGAN EDGE COMPUTING
4. MENJELAJAHI KARAKTERISTIK DAN KASUS PENGGUNAAN
5. KASUS PENGGUNAAN
6. SKENARIO
7. TANTANGAN
8. KESIMPULAN DAN SERUAN AKSI
9. AUTHORS
10. Translators

PENGANTAR

Selama lebih dari satu dekade, komputasi awan terpusat (centralized cloud computing) telah dianggap sebagai platform pengiriman TI standar (standard IT delivery platform). Meskipun komputasi awan ada di mana-mana, persyaratan dan beban kerja (requirements and workloads) yang muncul mulai memperlihatkan keterbatasannya. Dengan tampilan sentris data center yang kuat, di mana sumber daya komputasi dan penyimpanannya relatif banyak dan terpusat, sedikit atau tidak ada pemikiran diberikan untuk mengoptimalkan hypervisor pendukung dan management platform footprint. Beberapa pengembang awan secara serius mempertimbangkan persyaratan yang dibutuhkan untuk mendukung node yang dibatasi sumber daya (resource-constrained) yang dapat dijangkau hanya dengan koneksi jaringan yang tidak dapat diandalkan atau bandwidth-terbatas (bandwidth-limited), atau memikirkan kebutuhan aplikasi yang membutuhkan bandwidth yang sangat tinggi, latensi rendah, atau kapasitas komputasi yang meluas di banyak tempat.

Aplikasi, layanan, dan beban kerja yang baru semakin menuntut jenis arsitektur yang berbeda, yang dibangun untuk secara langsung mendukung infrastruktur terdistribusi. Persyaratan baru untuk ketersediaan dan kemampuan komputasi awan di lokasi terpencil diperlukan untuk mendukung kedua kebutuhan hari ini (analisis data ritel, layanan jaringan) dan inovasi masa depan (kota cerdas (smart city), AR/VR). Kematangan, ketahanan, fleksibilitas, dan kesederhanaan komputasi awan sekarang perlu diperluas ke beberapa situs dan jaringan untuk mengatasi tuntutan yang terus berkembang.

Baru-baru ini perusahaan mulai menerapkan administrasi dan fleksibilitas arsitektur cloud computing yang disederhanakan untuk mendistribusikan infrastruktur yang menjangkau beberapa situs dan jaringan. Kebutuhan organisasi mulai muncul untuk memanfaatkan kemampuan komputasi awan di jaringan WAN dan penyebarannya semakin kecil ke tepi jaringan (network edge). Meskipun pendekatan ini masih awal, namun semakin jelas bahwa banyak kasus penggunaan dan skenario yang ada akan menguntungkan dari arsitektur terdistribusi (distributed architectures).

Dalam tulisan ini, kita mengeksplorasi kebutuhan yang muncul ini. Telah disebut banyak nama teknologi komputasi: awan terdistribusi (distributed cloud), komputasi kabut (fog computing), pusat data generasi ke-4 (4th generation data centers), namun untuk tujuan dokumen ini, kita akan tetap menggunakan term—cloud edge computing (komputasi tepi awan) yang umum dan mudah dipahami.

The OSF Edge Computing Group melihat evolusi komputasi awan ini sangat menarik, walaupun kami menyadari bahwa kemampuan OpenStack untuk mendukung cloud edge computing masih pada tahap awaln. Berdasarkan minat awal yang diungkapkan community di OpenStack Summit Boston, sebuah lokakarya dua hari diadakan pada bulan September 2017 yang mengumpulkan lebih dari 200 pengguna dan pengembang untuk memulai kerja keras menentukan kasus penggunaan yang relevan dan mempertimbangkan alat dan arsitektur yang dibutuhkan untuk mendukung community. Bukti konsep (proof of concept) telah dilakukan dan masyarakat memiliki beberapa awal penyebaran (deployment) di tempatnya. The Edge Computing Group kini telah mengambil tantangan untuk menjelaskan kebutuhan mendasar dari infrastruktur functional edge computing cloud secara penuh.

Dalam dokumen ini, kami bermaksud untuk menyelesaikan beberapa tugas pokok:

1. 1. Mengembangkan percakapan seputar cloud edge computing, termasuk beberapa definisi dasar, sehingga dapat merangsang minat dan keterlibatan dari komunitas open source.
2. 2. Memandu komunitas open source dan OpenStack yang lebih luas dalam mengembangkan alat dan standar yang dibutuhkan untuk adopsi secara luas.
3. 3. Menjelajahi bagaimana alat, standar dan arsitektur terkini yang mungkin perlu diubah untuk mengakomodasi model awan terdistribusi ini.

Ada banyak pekerjaan yang harus dilakukan untuk mencapai tujuan kami ini, dan kami menyambut serta mendorong seluruh komunitas open source untuk bergabung dalam usaha dan kesempatan untuk membuat atau mengadaptasi alat untuk memenuhi persyaratan baru komputasi awan.

APAKAH CLOUD EDGE COMPUTING?

Perlu disoroti bahwa banyak definisi komputasi tepi (edge computing) yang saling tumpang tindih dan kadang-kadang saling bertentangan - edge computing dapat diartikan berbagai hal bagi berbagai orang. Namun untuk tujuan kami ini, penampilan komputasi tepi yang paling matang adalah pengembang aplikasi dan penyedia menawarkan layanan kemampuan komputasi awan, serta lingkungan layanan TI di tepi jaringan (edge of a network).

Tujuannya adalah untuk memberikan proses komputasi, tempat penyimpanan, dan penyediaan bandwidth yang lebih dekat dengan input data dan/atau pengguna akhir. Lingkungan komputasi tepi ditandai dengan kemungkinan latensi yang tinggi di antara semua situs dan bandwidth yang rendah dan tidak dapat diandalkan — di samping pemberian layanan yang berbeda dan kemungkinan fungsi aplikasi yang tidak dapat dipenuhi dengan kumpulan sumber daya awan terpusat di pusat data yang jauh. Dengan memindahkan beberapa atau semua fungsi pemrosesan lebih dekat ke pengguna akhir atau titik pengumpulan data, komputasi cloud edge dapat mengurangi efek situs yang didistribusikan secara luas dengan meminimalkan efek latensi pada aplikasi.

Edge computing pertama kali muncul dengan memvisualisasikan layanan jaringan melalui jaringan WAN, mengambil langkah menjauh dari pusat data. Kasus penggunaan awal didorong oleh keinginan untuk memanfaatkan platform yang memberikan fleksibilitas dan alat sederhana yang pengguna komputasi awan telah terbiasa.

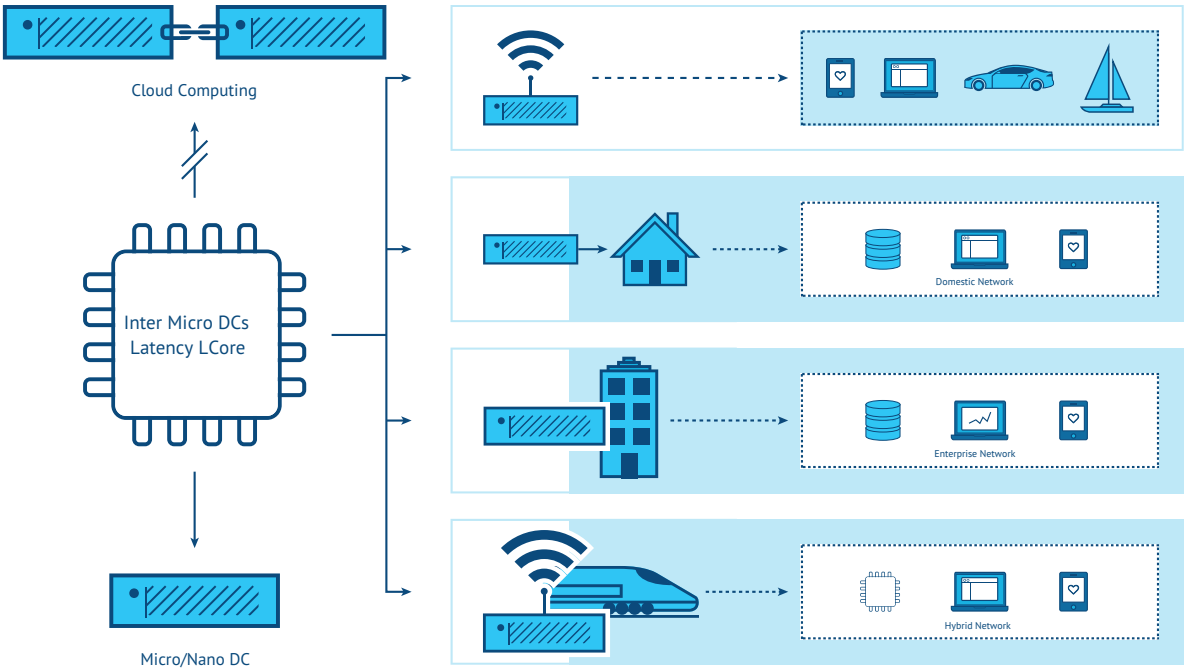
Seiring munculnya kemampuan-kemampuan komputasi baru, kita melihat paradigma yang berubah untuk komputasi — yang tidak lagi harus terikat oleh kebutuhan untuk membangun pusat data terpusat. Sebaliknya, untuk aplikasi tertentu, cloud edge computing mengambil pelajaran virtualisasi dan komputasi awan dan menciptakan kemampuan untuk memiliki ribuan node yang didistribusikan secara besar-besaran yang dapat diterapkan pada beragam kasus penggunaan, seperti IoT industri atau bahkan jaringan pemantauan yang sangat jauh (far-flung monitoring networks) untuk melacak penggunaan sumber daya air waktu nyata selama ribuan, atau jutaan, lokasi.

Banyak kemampuan komputasi tepi open source dan proprietary sudah ada tanpa bergantung pada cloud terdistribusi - beberapa vendor menyebut ini sebagai “device edge.” Komponen dari pendekatan ini mencakup elemen seperti gerbang IoT atau aplikasi NFV. Namun semakin meningkat, aplikasi membutuhkan fleksibilitas cloud di tepi, meskipun alat dan arsitektur yang dibutuhkan untuk membangun infrastruktur terdistribusi masih dalam tahap awal. Pandangan kami adalah bahwa pasar akan terus menuntut kemampuan yang lebih baik untuk cloud computing.

Kemampuan komputasi tepi (edge computing) mencakup berbagai layanan, namun tidak terbatas pada:

- * Paradigma operasi yang konsisten di berbagai infrastruktur.
- * Kemampuan untuk tampil di lingkungan yang terdistribusi secara massal (pikirkan ribuan lokasi global).
- * Kebutuhan untuk memberikan layanan jaringan kepada pelanggan yang berada di lokasi terpencil yang terdistribusi secara global.
- * Aplikasi integrasi, orkestrasi dan persyaratan pengiriman layanan.
- * Keterbatasan hardware dan biaya.
- * Koneksi jaringan terbatas atau terputus-putus.
- * Metode untuk menangani aplikasi dengan persyaratan latensi rendah yang ketat (AR/VR, suara, dan sebagainya).
- * Geofencing dan persyaratan untuk menjaga data pribadi sensitif lokal.

EKSPLORASI MENDALAM PERTIMBANGAN EDGE COMPUTING “Edge” dalam komputasi tepi (edge computing) mengacu pada pinggiran (outskirt) dari domain administratif, sedekat mungkin dengan sumber data diskrit atau dengan pengguna akhir. Konsep ini berlaku untuk jaringan telekomunikasi, untuk perusahaan besar yang mempunyai titik terdistribusi seperti ritel, atau untuk aplikasi lainnya, khususnya dalam konteks IoT.



Salah satu karakteristik komputasi tepi (edge computing) adalah aplikasi sangat terkait dengan lokasi tepi. Untuk telekomunikasi, “the edge” akan merujuk ke titik yang dekat dengan pengguna akhir namun tetap dikendalikan oleh penyedia, "the edge" berpotensi memiliki beberapa elemen beban kerja yang berjalan pada perangkat pengguna akhir. Untuk perusahaan besar, “the edge” adalah titik di mana aplikasi, layanan atau beban kerja digunakan (misalnya toko ritel atau pabrik). Untuk tujuan definisi ini, the edge bukanlah perangkat akhir dengan kapasitas yang sangat terbatas untuk mendukung bahkan arsitektur awan minimal, seperti perangkat IoT atau sensor. Pembedaan inii adalah pertimbangan penting, karena banyak diskusi tentang komputasi tepi tidak membuat perbedaan itu.

Komputasi tepi (edge computing) mirip dengan komputasi data center dalam hal:

- * Edge computing ini termasuk sumber daya komputasi, penyimpanan dan jaringan.
- * Sumbernya bisa dibagi oleh banyak pengguna dan banyak aplikasi.
- * Edge computing ini manfaatkan virtualisasi dan abstraksi dari kolam sumber daya (resource pool).
- * Edge computing ini manfaatkan kemampuan untuk meningkatkan perangkat keras komoditas.
- * Menggunakan API untuk mendukung interoperabilitas.

Komputasi tepi (edge computing) berbeda dari komputasi di pusat data yang besar:

- * Situs tepi (edge site) sedekat mungkin dengan pengguna akhir. Edge site memperbaiki kejadian melalui latensi tinggi dan koneksi yang tidak dapat diandalkan.
- * Mungkin memerlukan perangkat keras khusus, seperti platform GPU/FPGA untuk fungsi AR/VR.
- * Edge bisa berskala ke sejumlah besar situs, didistribusikan di lokasi yang berbeda.

- * Lokasi situs tepi (edge site’s location) dan identitas tautan akses yang diakhirinya sangat penting. Aplikasi yang perlu dijalankan dekat dengan penggunanya, aplikasi perlu berada di bagian tepi (edge) yang cocok. Sudah umum, lokasi aplikasi menjadi masalah dalam komputasi tepi (edge computing).
- * Seluruh kolam situs (pool of site) bisa dianggap dinamis. Karena pemisahan fisik pool of site, dalam beberapa kasusmasing masing situs tepi (edge site) akan saling terhubung satu sama lain dan inti terhubung dengan koneksi WAN. Situs tepi akan bergabung dan meninggalkan kolam infrastruktur dari waktu ke waktu.
- Situs tepi (edge site) jarak jauh dan berpotensi tak berawak, dan karenanya harus dikelola dari jarak jauh. Alat perlu mendukung akses jaringan intermiten (terputus-putus) ke situs.
- * Edge mendukung perbedaan besar dalam ukuran dan skala situs, mulai dari skala pusat data besar hingga satu perangkat kecil.
- * Situs tepi (edge site) dibatasi sumber daya; penambahan kapasitas ke situs yang ada terbatas karena persyaratan ruang atau daya.
- * Multi-tenancy dalam skala besar diperlukan untuk beberapa kasus penggunaan.
- * Isolasi komputasi tepi (edge computing) dari awan pusat data (data center cloud) mungkin diperlukan untuk memastikan bahwa kompromi di domain “external cloud” tidak dapat mempengaruhi layanan.

Konsep komputasi tepi (edge computing) harus mencakup situs tepi (edge site) (misal, infrastruktur, komputasi, jaringan dan penyimpanan), namun juga aplikasi (workload) yang berjalan di atasnya. Perlu dicatat bahwa setiap aplikasi di lingkungan komputasi tepi berpotensi memanfaatkan sebagian atau semua kemampuan yang disediakan oleh komputasi awan, penyimpanan blok, penyimpanan objek, jaringan virtual, perangkat nyata (bare metal), atau kontainer.

Fitur penting yang mendefinisikan dan memisahkan komputasi tepi (edge computing) dari komputasi awan (cloud computing) adalah:

- * kemampuan edge computing untuk mendukung kumpulan dinamis dari beberapa situs yang berpotensi didistribusikan secara luas,
- * koneksi jaringan edge computing yang berpotensi tidak dapat diandalkan, dan
- * kemungkinan kendala sumber daya edge computing yang sulit diselesaikan di situs-situs di seluruh jaringan.

MENJELAJAHI KARAKTERISTIK DAN KASUS PENGGUNAAN

Jadi, apa yang kita ketahui sejauh ini tentang karakteristik komputasi tepi (edge computing), kasus penggunaan, dan skenario?

Kebutuhan yang menentukan yang mendorong cloud edge computing adalah kebutuhan akan pemberian layanan agar lebih mendekati pengguna atau sumber data titik akhir (end-point data sources). Lingkungan komputasi edge akan bekerja dalam hubungannya dengan

kapasitas inti, namun lingkungan komputing edga bertujuan untuk memberikan pengalaman pengguna akhir yang meningkat tanpa memasukkan tuntutan konektivitas yang tidak masuk akal ke intinya. Hasil perbaikan lingkungan komputasi edge dari:

1. **Reducing latency:**Mengurangi latensi: Latensi ke pengguna akhir bisa lebih rendah dibandingkan komputasinya dilakukan lebih jauh - misalnya membuat remote desktop seresponsif mungkin, atau AR yang sukses, atau game yang lebih baik.
2. **Mitigating bandwidth limits:**Kemampuan untuk memindahkan beban kerja lebih mendekati pengguna akhir atau data collection points akan mengurangi efek bandwidth terbatas di suatu situs. Hal ini sangat berguna jika layanan pada node tepi (edge node) mengurangi kebutuhan untuk mengirimkan sejumlah besar data ke inti pemrosesan, seperti yang sering terjadi pada beban kerja IoT dan NFV. Pengurangan data dan pemrosesan lokal dapat diterjemahkan ke dalam aplikasi yang lebih responsif dan mengurangi biaya pengangkutan terabyte data dalam jarak yang jauh.

Tetapi ada timbal baliknya. Untuk memberikan komputasi tepi, institusi perlu meningkatkan jumlah penyebaran. Institusi ini merupakan tantangan yang signifikan terhadap penyebaran luas di seluruh penjuru negeri. Jika mengelola satu awan membutuhkan sepuluh tim, bagaimana sebuah organisasi bisa mengatasi ratusan atau bahkan ribuan awan kecil? Beberapa persyaratan organisasi meliputi:

1. 1. Konsistensi standardisasi dan infrastruktur sangat dibutuhkan. Setiap lokasi harus serupa; kuantitasnya yang harus diketahui.
2. 2. Manageability (pengelolaan) perlu diotomatisasi; pengelolaan penggelaran, penggantian dan pemulihan kegagalan harus sederhana dan mudah.
3. 3. Rencana sederhana dan hemat biaya perlu diletakkan apabila perangkat keras gagal.
4. 4. Desain toleransi kesalahan lokal mungkin penting, terutama di lingkungan yang terpencil atau tidak terjangkau - infrastruktur sentuh nol (zero touch infrastructure) sangat diharapkan. Ini adalah pertanyaan penyeimbangan biaya untuk membeli dan menjalankan perangkat keras yang berlebihan dibandingkan dengan biaya penghentian dan perbaikan keadaan darurat. Pertimbangan desain toleransi kesalahan lokal meliputi:
    1. 1. Apakah lokasi ini perlu swasembada?
    2. 2. Jika lokasi mengalami kegagalan, tidak ada yang akan menunggu di tempat untuk memperbaikinya, dan suku cadang lokal tidak mungkin terjadi.
    3. 3. Apakah perlu mentolerir kegagalan? Dan jika ya, berapa lama waktu yang akan terjadi sebelum seseorang akan siap memperbaikinya-dua jam, seminggu, satu bulan?
5. 1. Perawatan perlu teknisi yang tidak terlatih langsung melakukan perbaikan dan penggantian manual, dan sementara administrator jarak jauh yang terampil menginstal ulang atau merawat perangkat lunak.
6. 2. Desain fisik mungkin perlu dipikir ulang secara keseluruhan. Sebagian besar lingkungan komputasi tepi tidak akan ideal-daya terbatas, kotoran (dirt), kelembaban dan getaran harus dipertimbangkan.

KASUS PENGGUNAAN Mungkin ada lusinan cara untuk mengkarakterisasi kasus penggunaan dan makalah ini terlalu pendek untuk memberikan daftar yang lengkap. Tapi berikut ini adalah beberapa contoh untuk membantu memperjelas pemikiran dan menyoroti kesempatan untuk berkolaborasi.

Empat kategori utama persyaratan beban kerja (workload) yang mendapatkan keuntungan dari arsitektur terdistribusi adalah analisis, kepatuhan, keamanan, dan NFV.

DATA COLLECTION AND ANALYTICS

IoT, dimana data sering dikumpulkan dari jaringan besar microsites, merupakan contoh aplikasi yang mendapat keuntungan dari model komputasi tepi. Mengirimkan banyak data melalui koneksi jaringan yang seringkali terbatas ke mesin analisis yang berada di pusat data terpusat bersifat kontraproduktif; Mungkin tidak cukup responsif, bisa berkontribusi pada latensi yang berlebihan, dan menghabiskan bandwidth yang berharga. Karena perangkat tepi juga dapat menghasilkan terabyte data, analisis lebih dekat ke sumber data di tepinya dapat lebih hemat biaya dengan menganalisis data di dekat sumber dan hanya mengirim sejumlah kecil informasi terkondensasi kembali ke sistem terpusat. Ada tradeoff di sini-penyeimbangan biaya pengangkutan data ke inti dibandingkan dengan kehilangan beberapa informasi.

Security (keamanan)

Sayangnya, karena perangkat tepi berkembang cepat (edge devices proliferate) - termasuk handset mobile dan sensor IoT - vektor serangan baru muncul memanfaatkan penyebaran (proliferasi) titik akhir. Komputasi tepi menawarkan kemampuan untuk memindahkan elemen keamanan lebih dekat ke sumber serangan yang berasal, memungkinkan aplikasi keamanan kinerja yang lebih tinggi, dan meningkatkan jumlah lapisan yang membantu mempertahankan inti dari pelanggaran dan risiko.

Kepatuhan

Kepatuhan mencakup berbagai persyaratan, mulai dari geofencing, kedaulatan data, dan penegakan hak cipta. Membatasi akses data berdasarkan geografi dan batas politik, membatasi arus data tergantung pada keterbatasan hak cipta, dan menyimpan data di tempat dengan peraturan tertentu dapat dicapai dan dapat diterapkan dengan infrastruktur komputasi tepi.

NETWORK FUNCTION VIRTUALIZATION (NFV)

Network Function Virtualization (NFV), intinya adalah aplikasi komputasi tepi klasik karena menyediakan fungsionalitas infrastruktur. Operator telekomunikasi mencari untuk mengubah model pengiriman layanan mereka dengan menjalankan fungsi jaringan virtual sebagai bagian, atau berlapis-lapis di atas, dari infrastruktur komputasi yang canggih. Untuk memaksimalkan efisiensi dan meminimalkan biaya/kompleksitas, menjalankan NFV pada infrastruktur komputasi yang rumit masuk akal.

Real-time

Aplikasi real-time, seperti AR/VR, mobil yang terhubung, telemedicine, internet industri sentuhan 4,0 dan kota cerdas, tidak dapat mentolerir lebih dari beberapa milidetik latency dan dapat sangat sensitif terhadap variasi jitter, atau latency. Sebagai contoh, mobil yang terhubung memerlukan latency rendah dan bandwidth tinggi, dan bergantung pada komputasi dan caching konten di dekat pengguna, menjadikan kapasitas tepi (edge capacity) sebagai kebutuhan. Dalam banyak skenario, terutama di mana otomasi loop tertutup (closed-loop automation) digunakan untuk menjaga ketersediaan tinggi, waktu respon dalam puluhan milidetik dibutuhkan, dan tidak dapat dipenuhi tanpa infrastruktur komputasi tepi.

Immersive (tampilan gambar tiga dimensi yang tampak mengelilingi pengguna).

Edge computing expands bandwidth capabilities, unlocking the potential of new immersive applications. Some of these include AR/VR, 4K video, and 360° imaging for verticals like healthcare. Caching and optimizing content at the edge is already becoming a necessity since protocols like TCP don’t respond well to sudden changes in radio network traffic. Edge computing infrastructure, tied into real-time access to radio/network information can reduce stalls and delays in video by up to 20% during peak viewing hours, and can also vary the video feed bitrate based on radio conditions.

Network Efficiency (Efisiensi Jaringan)

Many applications are not sensitive to latency and do not require large amounts of nearby compute or storage capacity, so they could theoretically run in a centralized cloud, but the bandwidth requirements and/or compute requirements may still make edge computing a more efficient approach. Some of these workloads are common today, including video surveillance and IoT gateways, while others, including facial recognition and vehicle number plate recognition, are emerging capabilities. With many of these, the edge computing infrastructure not only reduces bandwidth requirements, but can also provide a platform for functions that enable the value of the application—for example, video surveillance motion detection and threat recognition. In many of these applications, 90% of the data is routine and irrelevant, so sending it to a centralized cloud is prohibitively expensive and wasteful of often scarce network bandwidth. It makes more sense to sort the data at the edge for anomalies and changes, and only report on the actionable data.

Operasi situs mandiri (self-contained) dan otonom

Banyak lingkungan, bahkan saat ini, memiliki konektivitas yang terbatas, tidak dapat diandalkan atau tidak dapat diprediksi. Ini bisa termasuk transportasi (pesawat terbang, bus, kapal), operasi penambangan (rig minyak, jaringan pipa, tambang), infrastruktur listrik (ladang angin, pembangkit tenaga surya), dan bahkan lingkungan yang biasanya memiliki konektivitas yang baik, seperti toko. Komputasi tepi mendukung lingkungan seperti itu dengan membiarkan situs tetap semi otonom dan fungsional bila diperlukan atau bila konektivitas jaringan tidak tersedia. Contoh terbaik dari pendekatan ini adalah perlunya lokasi ritel mempertahankan sistem point of sale (POS) mereka, bahkan saat sementara tidak ada konektivitas jaringan.

Privasi

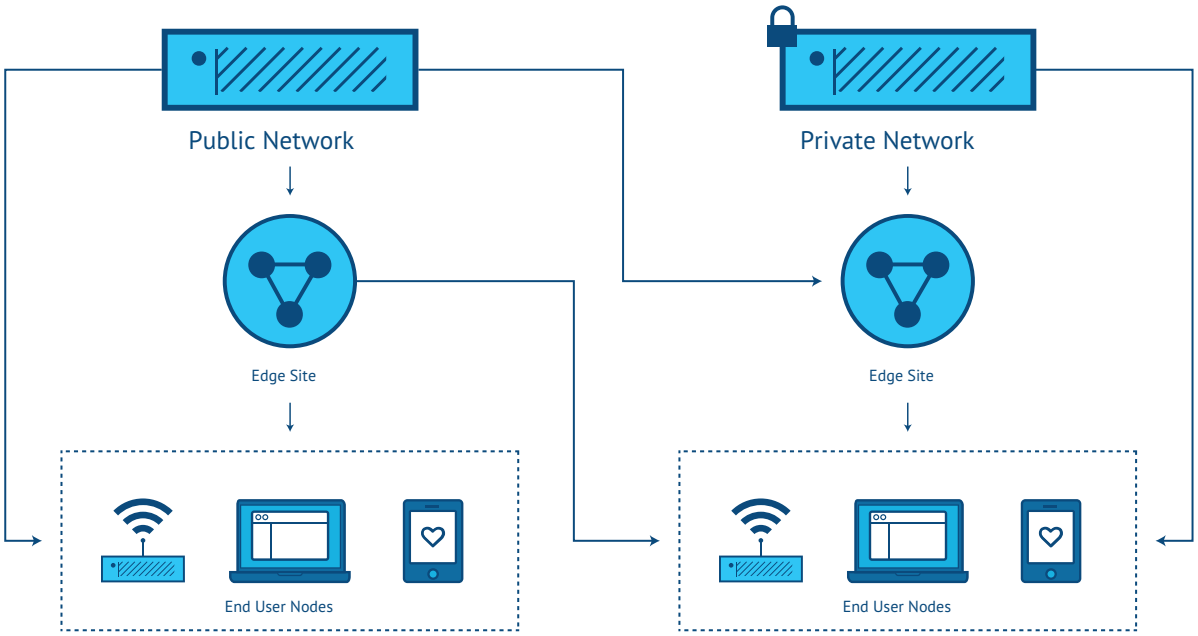
Perusahaan mungkin memiliki kebutuhan untuk kapasitas komputasi tepi tergantung pada beban kerja, batasan konektivitas dan privasi. Misalnya, aplikasi medis yang perlu menganonimkan informasi kesehatan pribadi personal health information (PHI) sebelum mengirimnya ke awan bisa melakukan ini dengan memanfaatkan infrastruktur komputasi yang canggih.

Cara lain untuk melihat persyaratan yang akan mendapat keuntungan dari cloud edge computing adalah dengan menentukan jenis perusahaan yang akan menerapkannya. Aplikasi operator menjadi workload (beban kerja) yang diletakkan pada infrastruktur komputasi yang dibangun dan dikelola oleh operators—perusahaan telekommunikasi, sebagai contoh. Aplikasi pihak ketiga dibangun oleh organisasi untuk berjalan di infrastruktur tepi yang ada, untuk memanfaatkan infrastruktur komputasi terdepan lainnya. Perlu dicatat bahwa aplikasi apapun dapat memanfaatkan salah satu atau semua kemampuan yang disediakan oleh cloud-compute, block storage, object storage, virtual networking, bare metal, atau containers.

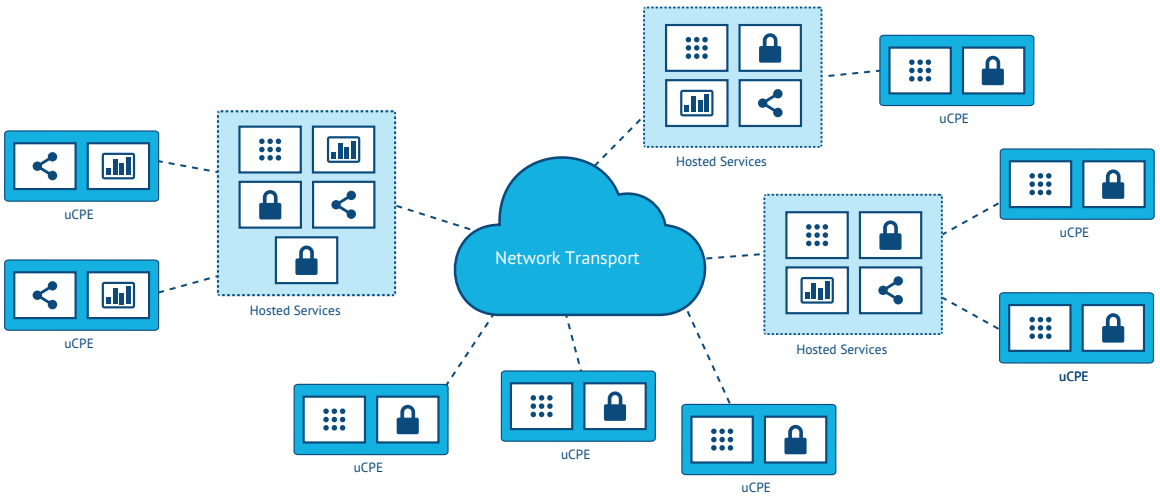
SKENARIO Karakteristik dasar dari paradigma komputasi tepi adalah bahwa infrastruktur berada dekat dengan pengguna akhir, bahwa skala distribusi situs tinggi dan simpul tepi terhubung oleh koneksi jaringan WAN. Memeriksa beberapa skenario lebih mendalam membantu kami mengevaluasi kemampuan saat ini yang memetakan ke kasus penggunaan, serta menyoroti kelemahan dan peluang untuk perbaikan.

1. Ritel / keuangan / lokasi terpencil “cloud in a box”: Infrastruktur komputasi tepi yang mendukung rangkaian aplikasi yang disesuaikan dengan perusahaan tertentu atau vertikal industri. Seringkali digunakan oleh perusahaan, infrastruktur komputasi tepi, akhirnya digabungkan menjadi infrastruktur terdistribusi, untuk mengurangi jejak perangkat keras (hardware footprint), membuat standar penerapan di banyak tempat, memberikan fleksibilitas yang lebih besar untuk menggantikan aplikasi yang berada di tepi (dan memiliki aplikasi yang sama yang berjalan secara seragam di semua node terlepas dari HW), meningkatkan ketahanan, dan mengatasi kekhawatiran tentang koneksi WAN terputus putus (intermiten). Menyembunyikan konten atau menyediakan komputasi, penyimpanan, dan jaringan untuk aplikasi mandiri menjadi pemanfaatan yang jelas untuk komputasi tepi dalam pengaturan dengan konektivitas terbatas.

2. Konektivitas mobile: Jaringan mobile / nirkabel cenderung menjadi elemen lingkungan yang umum digunakan untuk komputasi awan, karena jaringan seluler akan tetap dicirikan oleh bandwidth yang terbatas dan tidak dapat diprediksi, setidaknya sampai 5G tersedia secara luas. Aplikasi seperti augmented reality untuk perbaikan jarak jauh (remote repair) dan telemedicine, perangkat IoT untuk menangkap utilitas (air, gas, listrik, manajemen fasilitas), persediaan, rantai pasokan dan solusi transportasi, kota cerdas, jalan cerdas dan aplikasi keamanan jarak jauh akan bergantung pada jaringan seluler ke tingkat yang lebih besar atau lebih rendah. Mereka semua akan mendapatkan keuntungan dari kemampuan komputasi tepi untuk memindahkan beban kerja lebih dekat ke pengguna akhir.



3. Network-as-a-Service (NaaS): Datang dari kebutuhan untuk memberikan pengalaman aplikasi layanan jaringan yang identik di lingkungan yang sangat berbeda, kasus penggunaan NaaS memerlukan jejak kaki kecil (small footprint) dari platform terdistribusi di tepinya, dan . alat manajemen terpusat kuat yang melewati koneksi jaringan WAN yang tidak dapat diandalkan atau terbatas untuk mendukung layanan di tepi. Karakteristik utama dari skenario ini adalah: tapak perangkat keras kecil (small hardware footprint), bergerak (mengubah koneksi jaringan) dan terus-menerus mengubah beban kerja, lokasi hibrida data dan aplikasi. Ini adalah salah satu kasus yang membutuhkan infrastruktur untuk mendukung mikro node - dosis kecil (micro nodes—small doses) yang dihitung dalam paket non-tradisional (tidak semua 19in rack di pusat data yang didinginkan). NaaS akan membutuhkan dukungan ribuan atau puluhan ribu node di tepi dan harus mendukung arsitektur hirarki dan / atau bertautan (mesh) serta lokasi permintaan yang mungkin berputar sesuai kebutuhan dan shutdown saat selesai. API dan GUI harus berubah untuk mencerminkan bahwa sejumlah besar node komputasi akan memiliki lokasi yang berbeda dari kehadirannya di pusat data yang sama.



4. Universal Customer Premises Equipment (uCPE): Skenario ini, yang telah digunakan hari ini, menuntut dukungan untuk jejak perangkat keras seukuran alat dan ditandai oleh koneksi jaringan yang terbatas dengan beban kerja yang umumnya stabil yang membutuhkan ketersediaan tinggi. Ini juga memerlukan metode untuk mendukung lokasi hibrida data dan aplikasi di ratusan atau ribuan simpul dan penskalaan penempatan uCPE yang ada akan menjadi persyaratan yang muncul.

Hal ini terutama berlaku untuk aplikasi NFV di mana berbagai situs mungkin memerlukan serangkaian aplikasi layanan yang terhubung, atau situs dengan serangkaian aplikasi lain yang diperlukan yang masih perlu dilakukan dalam konser. Arsitektur hierarkis atau mesh perlu didukung dengan kapasitas lokal dan kebutuhan untuk menyimpan dan meneruskan pemrosesan data karena koneksi jaringan terputus-putus. Self-healing dan self-administration dikombinasikan dengan kemampuan untuk mengelola simpul dari jarak jauh adalah keharusan.

5. Satellite enabled communication (SATCOM): Skenario ini ditandai oleh banyak perangkat terminal yang mampu, sering didistribusikan ke kondisi yang paling jauh dan sulit. Pada saat yang sama, masuk akal untuk memanfaatkan platform terdistribusi ini untuk layanan hosting, terutama mengingat latency yang sangat tinggi, bandwidth yang terbatas dan biaya komunikasi melalui satelit. Contoh spesifik dari kasus penggunaan semacam itu mungkin mencakup kapal-kapal (dari kapal penangkap ikan sampai kapal tanker), pesawat terbang, rig minyak, operasi penambangan atau infrastruktur kelas militer.

TANTANGAN Meskipun ada banyak contoh penerapan teknologi tepi yang telah berlangsung di seluruh dunia, adopsi yang luas akan memerlukan cara berpikir baru untuk memecahkan tantangan dan keterbatasan yang muncul dan sudah ada.

Kami telah menetapkan bahwa platform komputasi tepi harus dirancang secara lebih toleran dan lebih kuat daripada awan sentris data center tradisional, baik dari sisi perangkat keras maupun layanan platform yang mendukung siklus hidup aplikasi. Kita tidak dapat mengasumsikan bahwa kasus penggunaan tepi seperti itu akan memiliki fasilitas pemeliharaan dan dukungan yang infrastruktur data center standar. Suplai zero touch, otomasi, dan orkestrasi otonom di semua tumpukan infrastruktur dan platform merupakan persyaratan penting dalam skenario ini.

Tetapi masih ada tantangan lain yang perlu dipertimbangkan.

Untuk satu, sistem pengelolaan sumber daya tepi harus memberikan serangkaian mekanisme tingkat tinggi yang hasil rakitannya dalam sistem yang mampu beroperasi dan menggunakan infrastruktur IaaS yang didistribusikan secara geografi dengan mengandalkan interkoneksi WAN. Dengan kata lain, tantangannya adalah merevisi (dan memperluas bila diperlukan) layanan inti IaaS supaya dapat menangani hal-hal yang spesifik di atas - pemutusan jaringan / bandwidth, kapasitas terbatas dalam hal perhitungan dan penyimpanan, pemasangan tanpa awak, dan sebagainya.

Beberapa kebutuhan yang dapat diperkirakan meliputi:

- Manajer virtual-machine/container/bare-metal yang bertugas mengelola siklus hidup machine/container (konfigurasi, penjadwalan, penyebaran, penundaan / resume, dan shutdown).
- * Seorang manajer image yang bertanggung jawab atas file template (a.k.a. virtual-machine / container images).
- * Seorang manajer jaringan yang bertugas menyediakan konektivitas ke infrastruktur: jaringan virtual dan akses eksternal untuk pengguna.
- * Manajer penyimpanan, menyediakan layanan penyimpanan ke aplikasi tepi.
- * Alat administrasi, menyediakan antarmuka pengguna untuk mengoperasikan dan menggunakan infrastruktur yang tersebar.

Kebutuhan ini relatif jelas dan kemungkinan bisa dipenuhi dengan memanfaatkan dan menyesuaikan proyek yang ada. Tapi kebutuhan komputasi tepi lainnya lebih menantang. Ini termasuk, namun tidak terbatas pada:

- * Mengatasi latensi penyimpanan melalui koneksi WAN.
- * Keamanan yang diperkuat di edge—monitoring integritas fisik dan aplikasi masing-masing lokasi, dengan kemampuan untuk secara mandiri mengaktifkan tindakan perbaikan bila diperlukan.
- * Pemantauan pemanfaatan sumber daya di semua node secara bersamaan.
- * Alat orkestrasi yang mengelola dan mengkoordinasikan banyak situs tepi dan beban kerja, berpotensi menuju ke arah peering control plane atau “self-organizing edge.”

- * Orkestrasi federasi platform tepi (atau cloud-of-clouds) harus dieksplorasi dan diperkenalkan pada layanan inti IaaS.
    - * Automated edge commission/decommission operations, termasuk penyebaran perangkat lunak awal dan peningkatan komponen sistem manajemen sumber daya.
    - * Data otomatis dan relokasi beban kerja - load balancing di perangkat keras yang terdistribusi secara geografis.
- * Beberapa bentuk sinkronisasi propagasi negara abstrak harus dibutuhkan di "core" infrastruktur untuk mengatasi hubungan jaringan yang tidak terputus-putus.
- * Cara baru untuk mengatasi masalah pembagian jaringan karena keterbatasan konektivitas - mengatasi pemutusan hubungan pendek dan pemutusan yang panjang.
- * Alat untuk mengelola siklus hidup aplikasi tepi, termasuk:
    - * Definisi kendala penempatan lanjutan untuk mengatasi persyaratan latensi komponen aplikasi.
    - * provisioning/scheduling aplikasi untuk memenuhi persyaratan penempatan (penempatan awal).
    - * Data dan relokasi beban kerja menurut kejadian internal/external (kasus penggunaan mobilitas, kegagalan, pertimbangan kinerja, dan lain-lain).
- * Integrasi kesadaran lokasi: Tidak semua penerapan tepi akan memerlukan aplikasi yang sama pada saat bersamaan. Kesadaran lokasi dan permintaan merupakan kebutuhan yang mungkin.
- * Perangkat keras diskrit dengan sumber daya terbatas dan kemampuan terbatas untuk memperluas lokasi terpencil perlu dipertimbangkan saat merancang keseluruhan arsitektur pada tingkat makro dan alat administratif. Konsep untuk dapat mengambil sumber daya jarak jauh berdasarkan permintaan dari situs lain, baik tetangga melalui jaringan mesh atau dari elemen inti dalam jaringan hierarkis, berarti bahwa fluktuasi permintaan lokal dapat dipenuhi tanpa inefisiensi penerapan perangkat keras.

KESIMPULAN DAN SERUAN AKSI

Halaman web OpenStack OSF Edge Computing

Tempat utama untuk menemukan petunjuk pada video tentang kejadian, artikel, dan konten terdepan sebelumnya di komputasi tepi.

Milis OpenStack OSF Edge Computing

Forum diskusi tentang komputasi tepi dan tepi (tidak eksklusif untuk OpenStack) dan menerima informasi tentang aktivitas yang sedang berlangsung dan seruan untuk bertindak.

OpenStack Design Summit (Konferensi Tingkat Tinggi / KTT perancangan OpenStack)

Konferensi empat hari untuk para pemimpin bisnis TI, operator dan pengembang awan yang meliput lanskap infrastruktur terbuka. Lihatlah trek dan sesi yang terkait dengan tepi pada KTT mendatang.

IRC Meeting (pertemuan chat IRC)

Fog/Edge/Massively Distributed Clouds (FEMDC) Pertemuan SIG IRC (online) diadakan setiap dua minggu (minggu yang ganjil), 1500 UTC pada hari Rabu, di saluran IRC # openstack-meeting.

OSF events

Jadwal acara global termasuk OpenStack Summits yang populer dan OpenStack Days regional.

OpenStack Marketplace (https://www.openstack.org/marketplace/)

One-stop resource untuk ekosistem global yang terampil untuk distribusi, driver, pelatihan, layanan dan lainnya.

Dokumentasi OpenStack

Indeks ke semua dokumentasi, untuk setiap peran dan langkah dalam merencanakan dan mengoperasikan awan OpenStack.

Selamat datang di komunitas! (https://www.openstack.org/community/)

Bergabunglah dengan milis dan saluran IRC, cari pekerjaan dan acara, akses kode sumber dan banyak lagi.

Kelompok pengguna

Temukan kelompok pengguna di dekat Anda, hadirilah pada pertemuan dan hackathons-atau selenggarakan sesuatu!

AUTHORS

KONTRIBUTOR

- Beth Cohen, Distinguished Member of Technical Staff, Verizon
- Gnanavelkandan Kathirvel, Director – Cloud Strategy & Architecture, AT&T and Board of Directors, OpenStack Foundation (OSF)
- Gregory Katsaros, Senior System Architect, Inmarsat
- Adrien Lebre, Ass. Prof., IMT Atlantique/Inria/LS2N, France
- Andrew Mitry, Sr. Distinguished Engineer, Walmart Labs
- Christopher Price, President, Ericsson Software Technology
- Paul-André Raymond, SVP Technology, B.Yond
- Alex Reznik, Enterprise Architect, HPE and Chair, ETSI ISG MEC
- Pasi Vaananen, Systems Architect, NFV, Red Hat

- Ildiko Vansca, Ecosystem Technical Lead, OpenStack Foundation (OSF)
- Ian Wells, Distinguished Engineer, Cisco

TECHNICAL WRITER

- Brian E. Whitaker, Founder, Zettabyte Content

Translators

- suhartono, <cloudsuhartono@gmail.com>