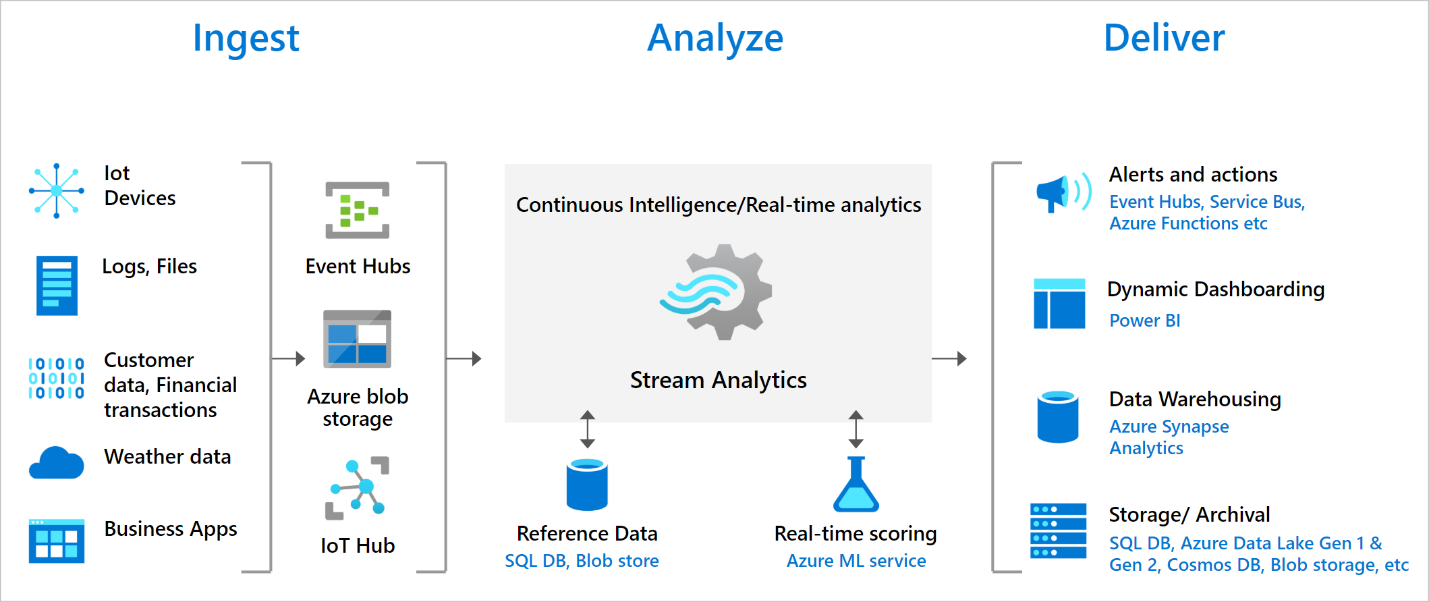
Azure Stream Analytics

Azure Stream Analytics adalah sebuah *event processing engine* yang digunakan untuk melakukan analisa secara *realtime*. Azure Stream Analytics bisa memproses data dalam jumlah besar dari berbagai sumber secara bersamaan. Bayangkan kita mengeksekusi sebuah SQL Query untuk melakukan SELECT dari satu atau lebih tabel, dilakukan berulang kali dengan periode tertentu, dengan *lantency* yang sangat rendah.

Penjelasan lebih lanjut terkait Azure Stream Analytics bisa dilihat langsung di [dokumentasi Microsoft](https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction). Namun, untuk mempermudah kita dalam memahami, perlu adanya contoh penerapan dari Azure Stream Analytics. Untuk itu, [Dinesh Priyankara](https://www.linkedin.com/in/dineshpriyankara/?originalSubdomain=lk) melalui webinar *Data Engineering*-nya pernah memberi sebuah contoh penggunaan Azure Stream Analytics.

**Contoh Kasus:** Dinas Kepolisian di sebuah kota ingin memantau kecepatan rata-rata dari kendaraan yang lewat di beberapa persimpangan jalan. Nantinya, hal ini dapat digunakan untuk memantau pelanggar lalu lintas yang melewati batas kecepatan. Terdapat sejumlah *speed camera* yang dipasang di beberapa persimpangan jalan. Pihak kepolisian ingin memantau kecepatan rata-rata kendaraan secara *realtime* setiap 30 detik dan menampilkannya di PowerBI Dashboard.

Arsitektur dari contoh kasus tersebut kurang lebih sama seperti yang ada pada dokumentasi Microsoft terkait Azure Stream Analytics.



Keseluruhan proses analisa data kecepatan terbagi dalam tiga tahap:

1. *Ingest*. Kamera pada persimpangan jalan adalah *IoT Device* yang akan melakukan pengiriman data ke Azure Event Hubs. Azure Event Hubs memiliki konsep yang sama dengan Apache Kafka. Penjelasan lebih lanjut mengenai bagaimana Azure Event Hubs bekerja dapat dilihat di [dokumentasi Microsoft](https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-about) atau penjelasan mengenai Apache Kafka dari [Learning Journal](https://youtu.be/udnX21__SuU).
2. *Analyze*. Azure Stream Analytics akan mengambil data dari Azure Event Hubs setiap 30 detik, kemudian mengirimnya ke PowerBI Stream Dataset.
3. *Deliver*. Agar pihak kepolisian dapat melihat kecepatan rata-rata setiap kamera maka diperlukan sebuah visualisasi. Visualisasi akan dilakukan menggunakan PowerBI Dashboard.

# Memproses Data dengan Azure Stream Analytics

## Persiapan

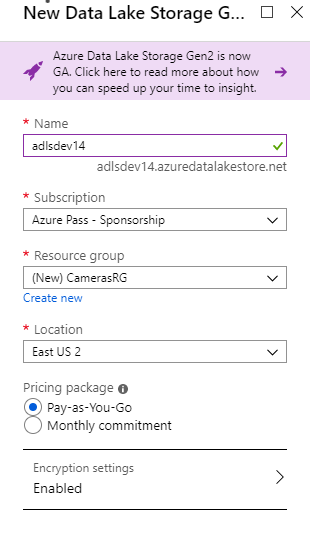
Sebelum memulai mengerjakan contoh kasus ini, kita perlu mempersiapkan beberapa hal:

1. *Azure Subscription*. Untuk mendapatkannya kita dapat membuka halaman resmi Azure. Azure menyediakan *Free Trial* 30 Hari apabila kita belum pernah menggunakan Azure samasekali.
2. NodeJs di komputer. Kita akan menggunakan sebuah program sederhana untuk menggantikan peran *Speed Camera* pada contoh kasus ini.
3. Program *Speed Camera Driver* yang bisa diunduh dari GitHub Repository [ini](https://github.com/faisalwilmar/cameradriver-event-hubs). Jangan lupa untuk membaca panduan instalasi dan cara menjalankannya.

## Membuat Data Lake Storage

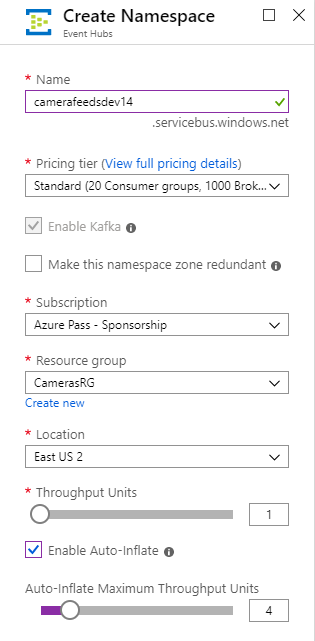
Kita perlu membuat Data Lake Storage untuk nantinya menyimpan hasil dari Stream Analytics.

1. Buka Azure Portal di [http://portal.azure.com](http://portal.azure.com/) dan masuk mengunakan akun yang sudah memiliki *Azure Subscription*.
2. Klik **+ Create a resource**, pilih **Storage**, lalu **Data Lake Storage Gen1**.
3. Di bagian **New Data Lake Storage Gen1**, di kotak **Name**, isikan dengan **adls<nama><tanggal>**. Nama ini bersifat unik untuk seluruh pengguna Azure, sehingga nama dan tanggal akan memastikan bahwa tidak ada nama yang sama dengan milik orang lain.
4. Pada bagian **Resource Group**, klik **Create New**, beri nama dengan **CamerasRG**, lalu klik **Ok**. Langkah ini akan membuat Resource Group baru pada Azure milik kita. Resource Group adalah cara Azure dalam mengelompokkan *resource* sehingga mudah untuk dikelola terutama dalam hal *billing*, menonaktifkan layanan, dan sebagainya.
5. Pada bagian **Location**, pilih region yang paling dekat dengan lokasi kita. Bila sedang berada di Indonesia, bisa pilih region *South East Asia* atau *East Asia*.
6. Biarkan pengaturan lainnya sebagai nilai *default*, lanjutkan dengan klik **Create**.



## Membuat Event Hub Namespace dan Hub

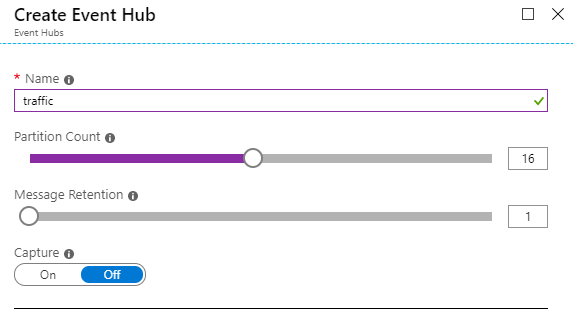
1. Di Azure Portal, klik **+ Create a Resource**, pilih **Internet of Things**, lalu pilih **Event Hubs**.
2. Pada bagian **Create Namespace**, di kotak **Name**, isikan dengan **camerafeeds<nama><tanggal>**.
3. Pada bagian **Pricing Tier** pilih **Standard**.
4. Pada bagian **Resource Group**, pilih resource groupyang sudah kita buat tadi yaitu **CamerasRG**.
5. Pada bagian **Location**, pilih lokasi yang sama dengan yang kita pilih untuk Data Lake Storage.
6. Atur **Throughput Units** ke 1. **Throughput Units** adalah satuan yang digunakan Azure untuk Event Hubs. 1 TU berarti Event Hubs dapat menerima 1MB *message* perdetik atau 1000 *message* perdetik, mana yang tercapai terlebih dahulu, serta mengirim *message* dengan jumlah duakalinya.
7. Pilih **Enable Auto-Inflate** dan atur **Auto-Inflate Maximum Throughput Units** ke 4. *Auto-Inflate* adalah mekanisme *scale up* untuk Event Hubs. Kita akan menjalankan Event Hubs dengan kapasitas minimal yaitu 1 TU, kemudian apabila *traffic*-nya meningkat maka Azure akan otomatis melakukan *scale up* hingga maksimal 4 TU. Event Hubs sendiri memiliki kapasitas maksimal 20 TU.
8. Klik **Create**.



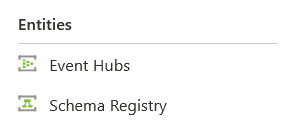
1. Tunggu hingga Event Hub Namespace berhasil dibuat.
2. Pilih **All Resources**, lalu pilih Event Hub Namespace yang baru saja dibuat, lali klik **+ Event Hub**.

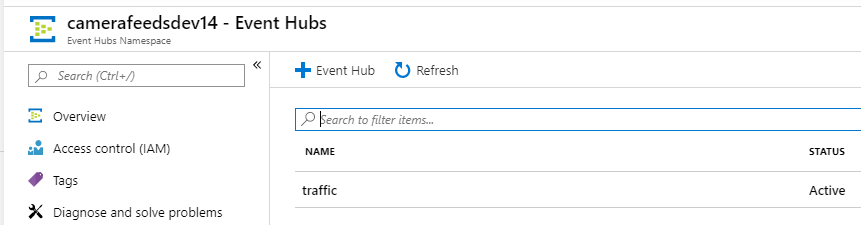


1. Pada bagian **Create Event Hub**, di kotak **Name**, isi dengan **traffic**.
2. Atur **Partition Count** menjadi **16**. Partisi digunakan oleh Event Hubs untuk memecah *load* dari Event Hubs itu sendiri. Banyaknya jumlah partisi akan selaras dengan meningkatnya jumlah pengguna dari Event Hubs. Hal ini dikarenakan partisi berkaitan dengan sistem kerja parallel milik Event Hubs. Semakin banyak pengguna yang mengakses dalam waktu bersamaan (*concurrent users*) maka semakin banyak partisi yang harus melayaninya. Jumlah partisi untuk satu Event Hubs bisa mencapai 32. Partisi pada Event Hubs tidak bisa dirubah kecuali aplikasi di-*restart*. Untuk itu penting untuk memperhatikan sebanyak apa *concurrent users* kedepannya sebelum melaukan pengaturan partisi. Dalam hal ini kita melakukan pengaturan partisi hingga 16 dengan anggapan kedepannya akan semakin banyak *speed camera* yang terpasang di kota tersebut.
3. Biarkan pengaturan lain pada posisi *default*, lalu klik **Create**.



1. Tunggu hingga Event Hub selesai dibuat.
2. Pada bagian Event Hub Namespace **camerafeeds<nama><tanggal>**, dibagian menu **Entities**, klik **Event Hubs**, lalu klik **traffic**.

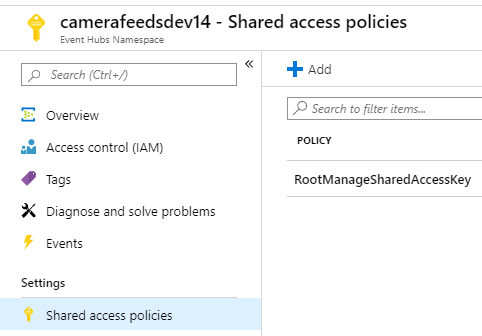




1. Pada bagian **traffic**, klik **+ Consumer Group.**



1. Pada bagian **Create Consumer Group**, isikan **Name** dengan **cameradatafeed**, lalu klik **Create**.
2. Lakukan Langkah 16 – 17 namun beri nama **cameradatafeed2**.
3. Pada bagian Event Hub Namespace **camerafeeds<nama><tanggal>**, pada bagian **Settings**, klik **Shared Access Policies**, lalu **RootManageSharedAccessKey**.



1. Klik **Click to Copy** pada bagian kanan **Primary Key**. Beri nama **Event Hub Primary** Key lalu simpan di notepad untuk digunakan nantinya.