

plain concepts 

DÍA 4: AGENDA

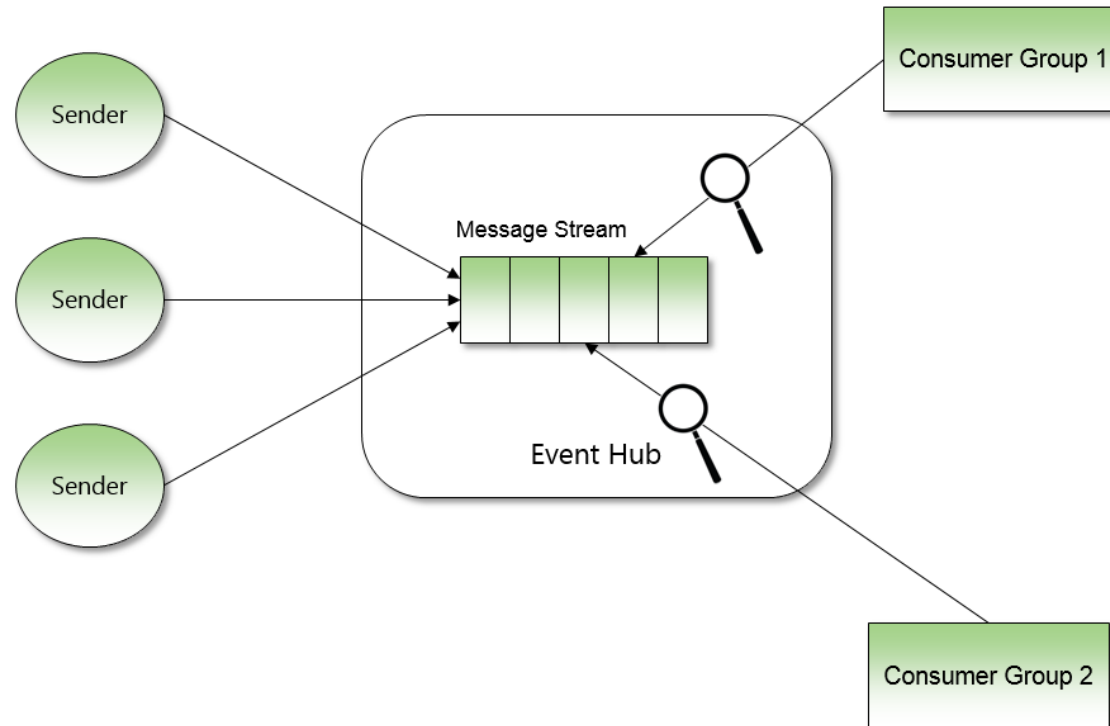
- Event Hub
- Streaming
 - Apache Storm
 - Azure Stream Analytics
- Bases de datos temporales
- Herramientas de Visualización
 - Grafana
 - PowerBI
- **Ejercicio práctico del día**

EVENT HUB

¿QUÉ ES EVENT HUBS?

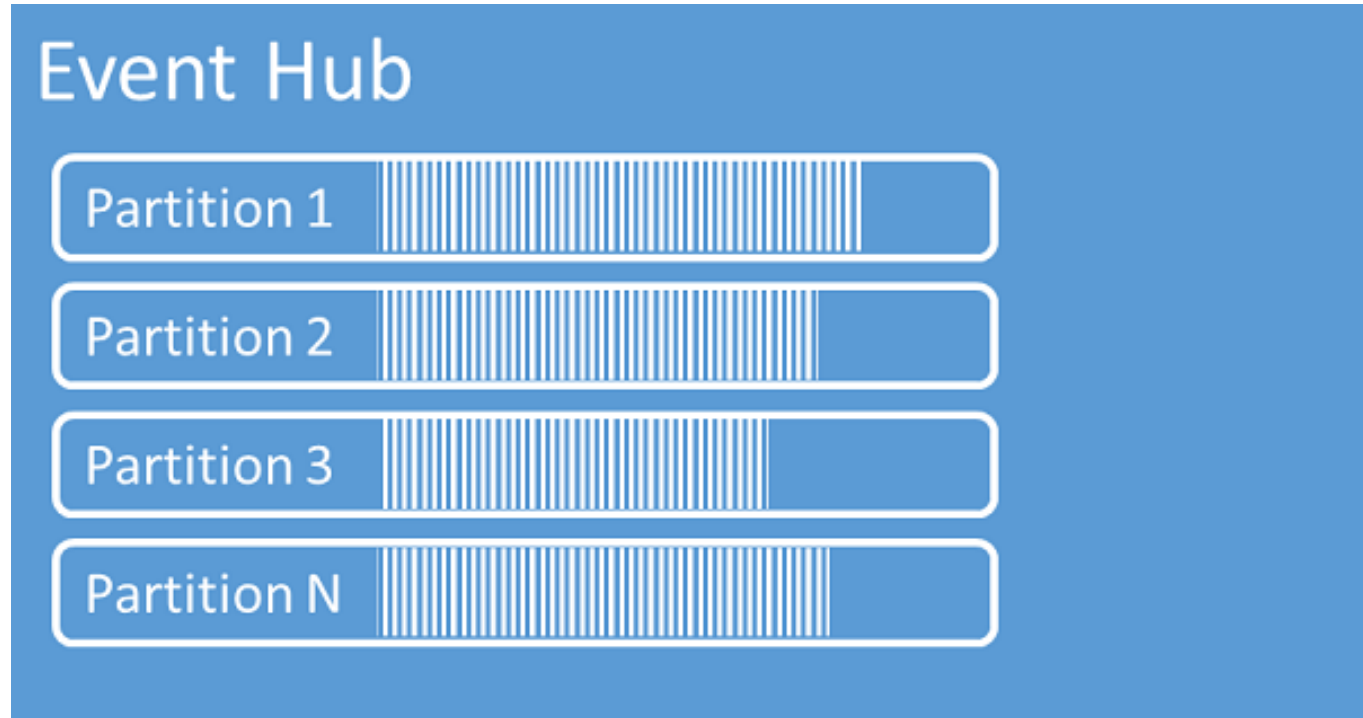
- Simplificando, es una cola de mensajes pero diseñada para trabajar a gran escala
- Provee interfaces HTTP y AMQP
- Internamente implementa un patrón de particionado para permitir un ratio de lecturas alto y un tiempo de retención de eventos más alto
- Se pueden definir grupos de consumidores para así tener varios procesadores de eventos ejecutándose en paralelo. Si solo se necesita un consumidor, se usará el grupo de consumidor por defecto
- El consumidor puede indicar desde cuando quiere empezar a leer eventos. Ejemplos:
 - Un consumidor podrías empezar a leer desde el comienzo hasta el final del stream y esperar a nuevos eventos
 - O bien podrías leer un subconjunto de eventos dentro del HUB

¿QUÉ ES EVENT HUBS?



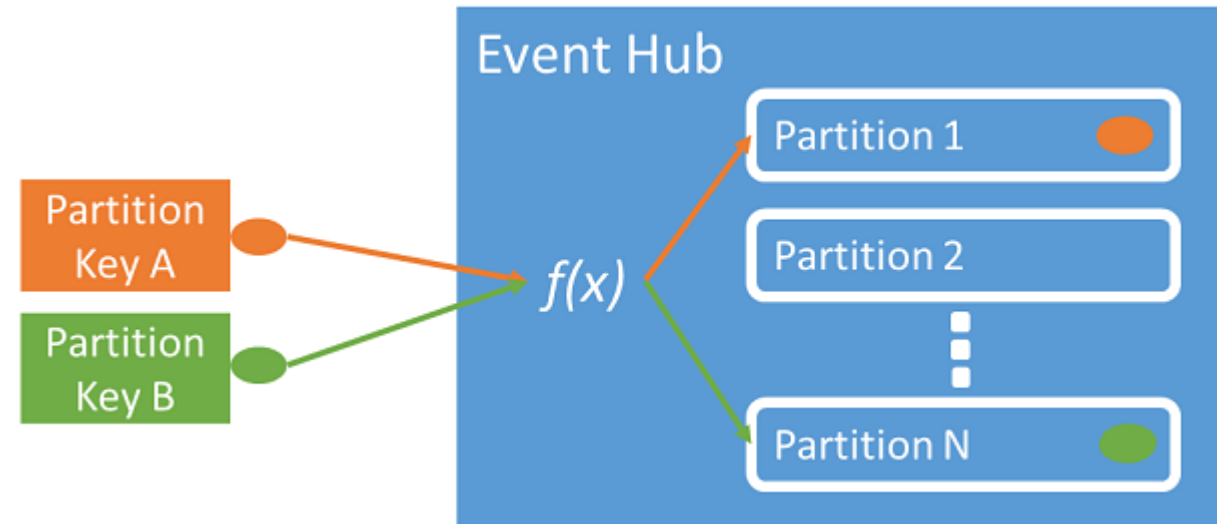
¿QUÉ SON LAS PARTICIONES?

- Una partición es una secuencia de eventos que han sido ingeridas por el Event Hub.
- A medida que van llegando nuevos mensajes, éstos se colocan al final de la partición
- Event Hubs solo asegura el orden de los eventos a nivel de partición



PARTICIONES

- Por defecto son 4, pero se puede aumentar hasta 32.
 - En realidad se pueden pedir más, pero hay que pedirselo directamente al equipo de Azure Service Bus
- Si el sender no provee una clave de partición, entonces se usará el método round robin
- Event Hubs asegura que todos los eventos en una partición serán entregados en orden

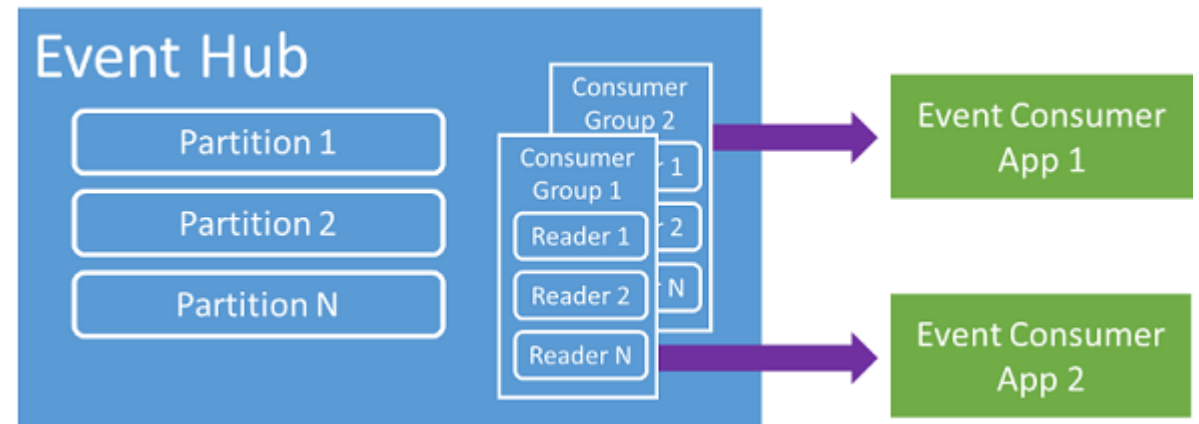


CONSUMIDORES

- Solo debe haber un consumidor activo por partición
- Cada consumidor de eventos debe pertenecer a un consumer group
- Via AMQP los mensajes se van entregando al consumidor a medida que van llegando, por lo que se evita el *polling*

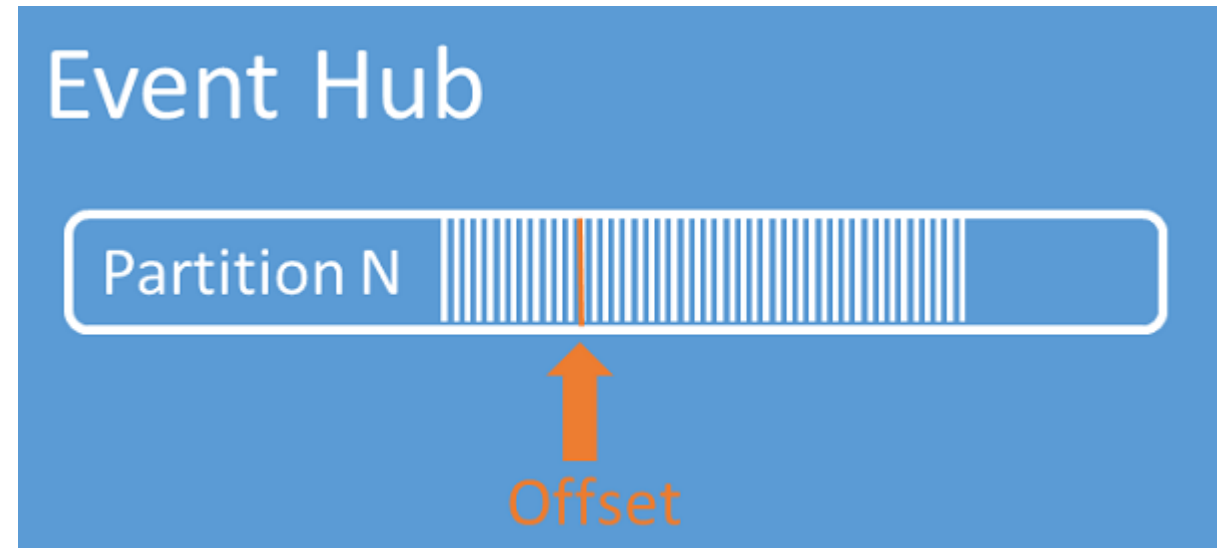
EVENT CONSUMER GROUP

- Un grupo de consumidores es una vista completa de un Event Hub
- Se pueden crear hasta 20 grupos de consumidores
- Los grupos de recursos permiten que diferentes apps consuman eventos de un Hub a su ritmo y empezando desde donde la aplicación considere oportuno (offset)



OFFSETS

- Un offset es la posición de un evento dentro de una partición
- Permite a un consumidor indicar desde donde quiere leer eventos
- Puede especificarse como un entero o un timestamp



CHECKPOINTS

- Es la manera que tiene un consumidor de eventos de cual fue el ultimo evento que leyó
- El cliente debe hacerse cargo de este proceso
- Es una manera de marcar eventos como “completados”

CAPACIDADES Y PRECIOS

- Escritura de eventos
 - 1000 eventos o 1MB por segundo
- Lectura
 - Hasta 2MB por segundo

Basic: Up to 100 connections, no extension Standard: 1000 connections incl.	Price (US Dollars)	
Throughput Unit Hour (Basic)	0.015/0.03	TU per hour (Basic/Standard)
Ingress Events	0.028	per 1,000,000 events
Cost Brokered Connections (0 -1k)	0	Included (Basic/100, Standard/1k)
Cost Brokered Connections (1k-100k)	0.00004	connection/hour
Cost Brokered Connections (100k-500k)	0.00003	connection/hour
Cost Brokered Connections (500k+)	0.00002	connection/hour
Storage Overage >TUs*84GB		local-redundant Azure storage charge-through

plain concepts

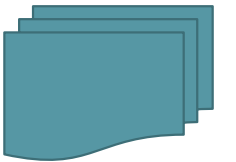
EVENT HUB



STREAMING

TRABAJANDO CON DATOS –TRANSFER

- Permiten obtener y proporcionar datos a herramientas de procesamiento o entre diferentes sistemas.
 - Sistemas de colas
 - Kafka, Rabbit, Message Bus, Flume
 - Hub
 - Event Hub, IoT Hub
 - Legacy
 - Online services
 - File Transfer



TRABAJANDO CON DATOS –STORAGE

- Nos permiten almacenar la información entre diferentes estados de procesamiento
 - NoSQL
 - HIVE, HBASE
 - MongoDB / Redis / CouchDB ..
 - SQL
 - SQL Server / PostgreSQL / Oracle ..
 - Files
 - HDFS



TRABAJANDO CON DATOS –PROCESSING

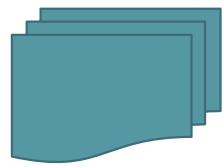
- Realizamos diversos cálculos que intentan extraer información, nueva inteligencia de negocio o simplemente nuevas visualizaciones de los mismos.
 - Hadoop
 - Tez / MR / para realizar procesos sobre grandes cantidades de datos
 - Spark
 - Opciones de visualización sobre un conjunto de datos, aprendizaje automático o procesamiento son algunas de las opciones de Spark.
 - Spark SQL / Mlib / GraphX.
 - Otros



ANALITICA TRADICIONAL

- Data At Rest
- Tenemos gran cantidad de productores de datos: sensores, dispositivos, aplicaciones...
- En un escenario BI tradicional, primero almacenamos los datos y después los analizamos
- Esto no es suficiente para adaptarse a los escenarios emergentes
 - Redes sociales
 - IoT, Internet of Things
- Los datos offline no son suficientes

¿QUE NOS FALTA?



STREAMING

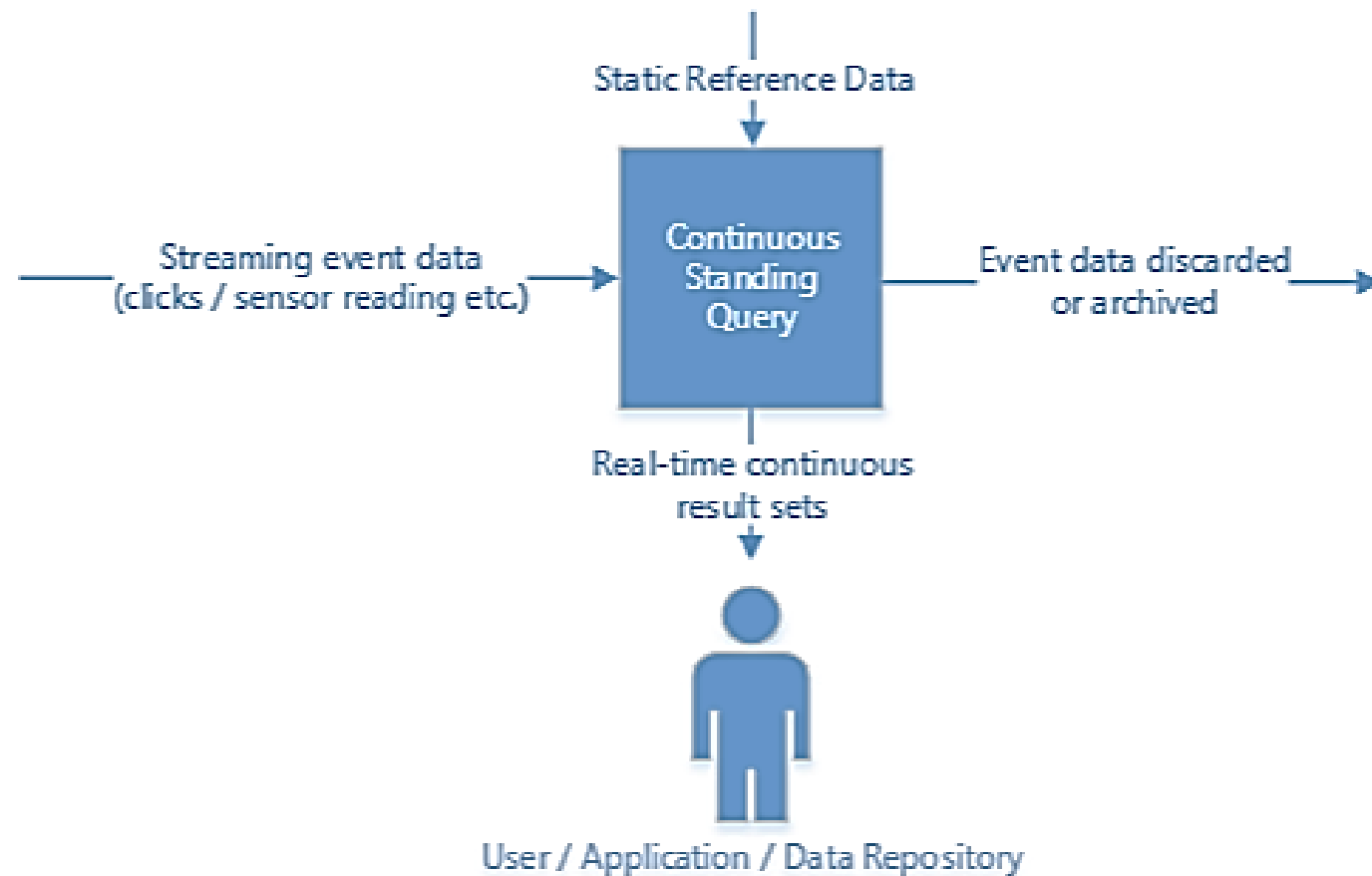
ANALÍTICA EN EL MUNDO MODERNO

- Data In Motion
- Trabajamos con datos en streaming
- Queremos monitorizar y analizar los datos en tiempo (casi) real
- No tenemos tiempo para recibir datos, almacenarlos y procesarlos antes del análisis
 - Necesitamos trabajar con streams

STREAM PROCESSING

- Stream vs Batch
 - Procesamos los datos según van entrando, no necesitamos hacer batch de grandes cantidades.
- Procesamiento por puntos de tiempo
 - Thresholds
 - Sensores
- Procesamiento por ventanas de tiempo
 - Trending
 - Alarmas
 - Agregados

STREAM PROCESSING





¿QUE ES APACHE STORM?

APACHE STORM

- Sistema distribuido de procesamiento de eventos
- Disponible en HDInsight
 - Clusters en Windows o Linux
- <http://storm.apache.org>

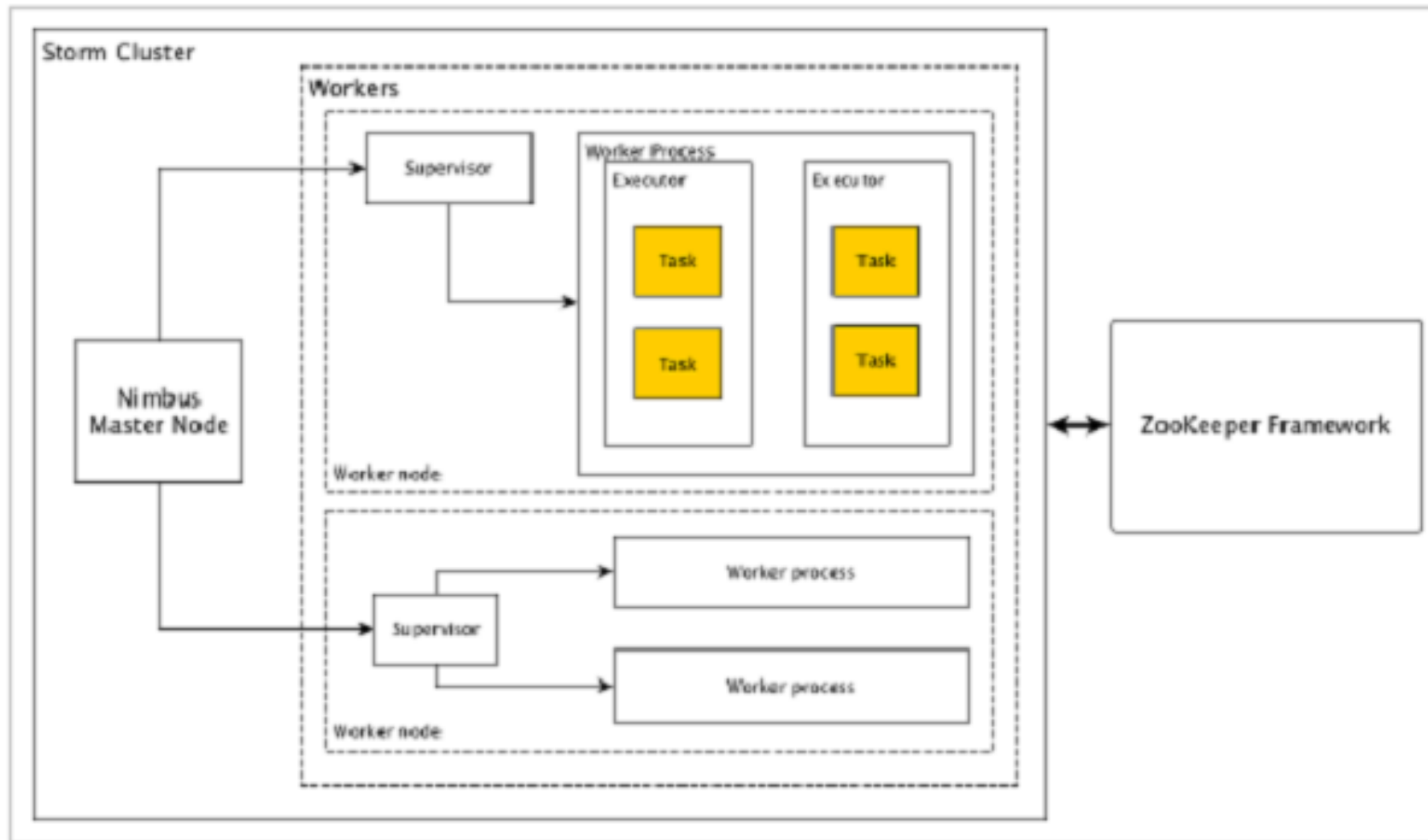
APACHE STORM

- Zookeeper
 - Como en cualquier otro clúster, para coordinación de los diferentes nodos
 - Ya existe en Hadoop
 - Si queremos failover debemos disponer de varios nodos de Zookeeper
- Nimbus
 - Master Node que se encarga de ejecutar nuestras topologías
 - Analiza y divide en tareas
 - Asigna tareas a “Supervisor(s)”

APACHE STORM

- Supervisor
 - Sigue las instrucciones del Nimbus
 - Dispone de múltiples “Worker Process”
 - Gobierna cada “Worker Process”
- Worker Process
 - Ejecuta tareas de una topología dentro de hilos llamados “Executors”
 - Un “Worker Process” tiene potencialmente muchos “Executors”

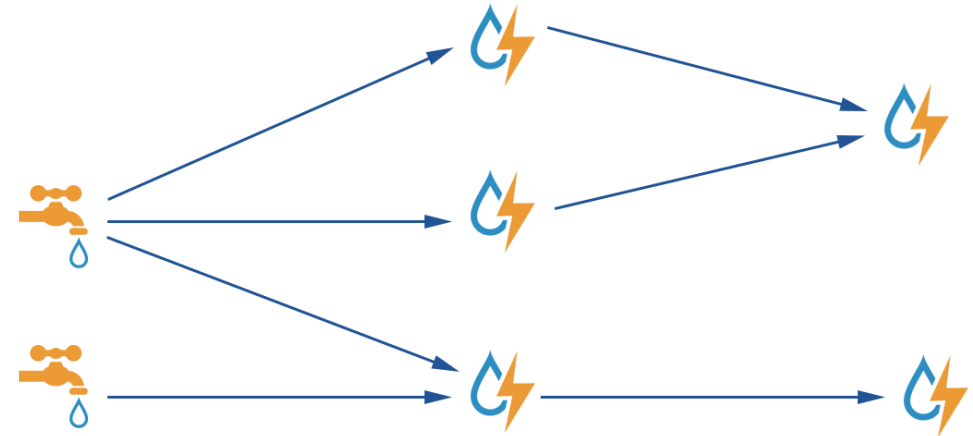
APACHE STORM



CONCEPTOS BASICOS DE APACHE STORM

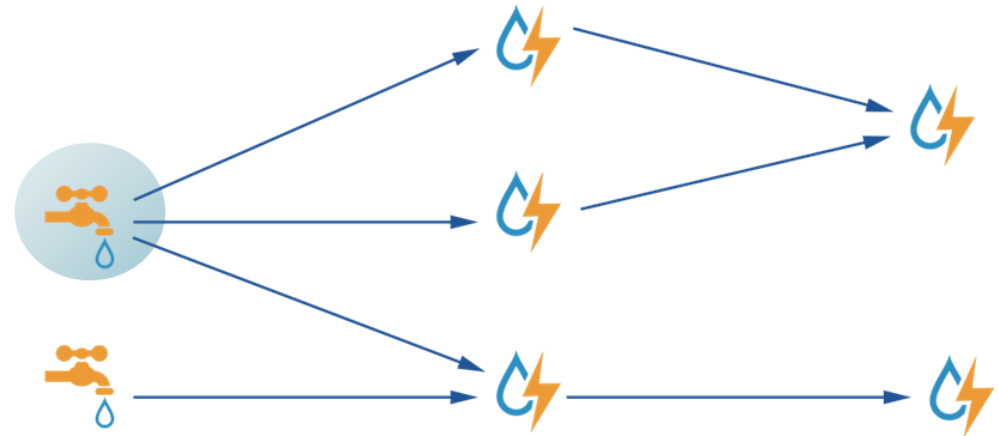
CONCEPTOS BASICOS - TOPOLOGIA

- Un workflow que marca el procesamiento de nuestros Streams
- Básicamente, una topología es una grafo dirigido donde los vértices son elementos de computación y las aristas son los stream de datos



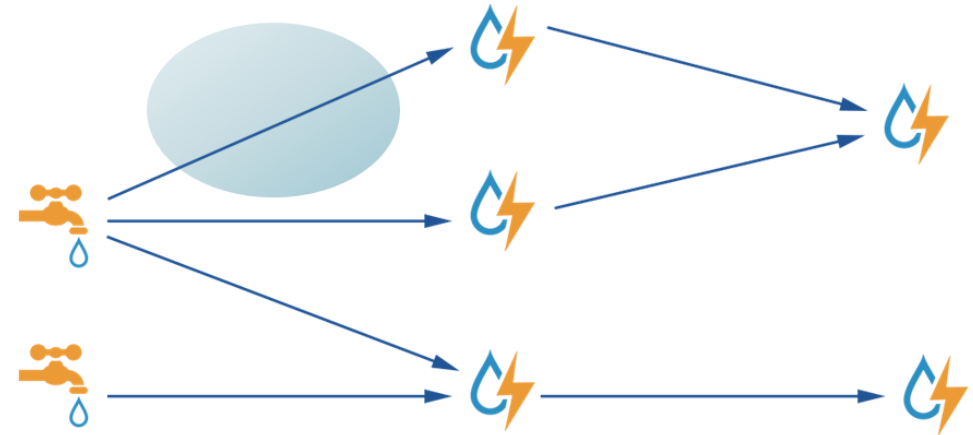
CONCEPTOS BASICOS - SPOUT

- El origen de los datos
- ISpout
 - Kafka
 - Kestrel
 - EventHub
 - ...
- Proporciona **tuplas** a la topología
 - Las tuplas son la estructura de datos básica.
 - Una lista de elementos ordenados



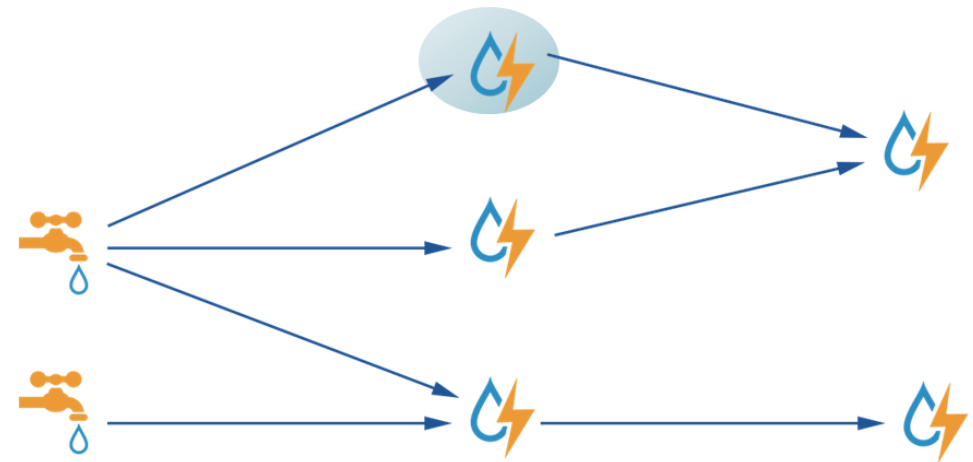
CONCEPTOS BASICOS - STREAM

- Una secuencia no ordenada de tuplas que salen de cada vértice
- Pueden estar en diferentes formatos
 - Json es habitual



CONCEPTOS BASICOS - BOLTS

- Unidades de procesamiento
- Pueden hacer operaciones simples o más complejas
 - Filtros, agregaciones, uniones..
 - Interactuar con datos referenciales como ficheros, bases de datos etc.
- IBolt
 - Lo habitual es desarrollar estas piezas aunque hay elementos reusables

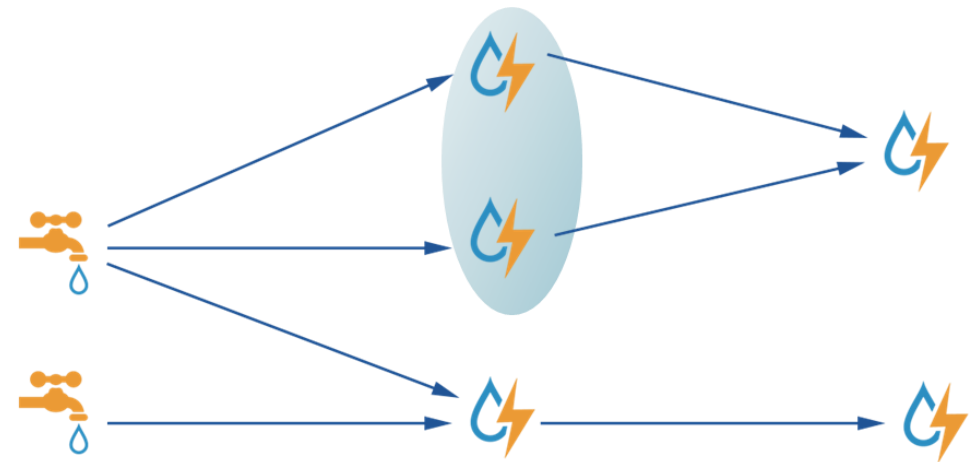


CONCEPTOS BASICOS – STREAM GROUPING

- Necesitamos estructurar la topología
- ¿Que streams recibe cada bolt?
- Podemos agrupar estas tuplas, según lo necesitemos, de varias formas diferentes:
 - Shuffle Grouping
 - Field Grouping
 - Global Grouping
 - All Grouping
 - Otros

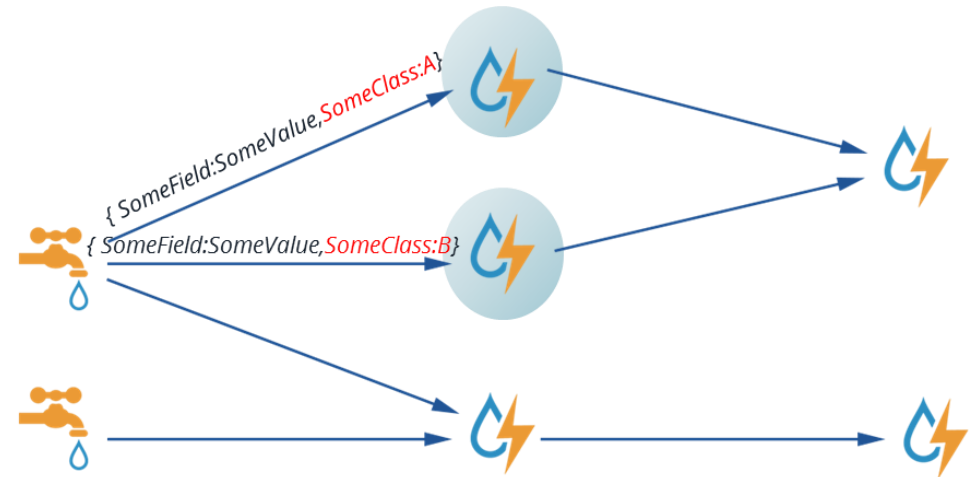
CONCEPTOS BASICOS - SHUFFLE GROUPING

- Las tuplas se dividen de forma aleatoria entre los diferentes bolts de destino.
- El numero de bolts para cada tarea es configurable



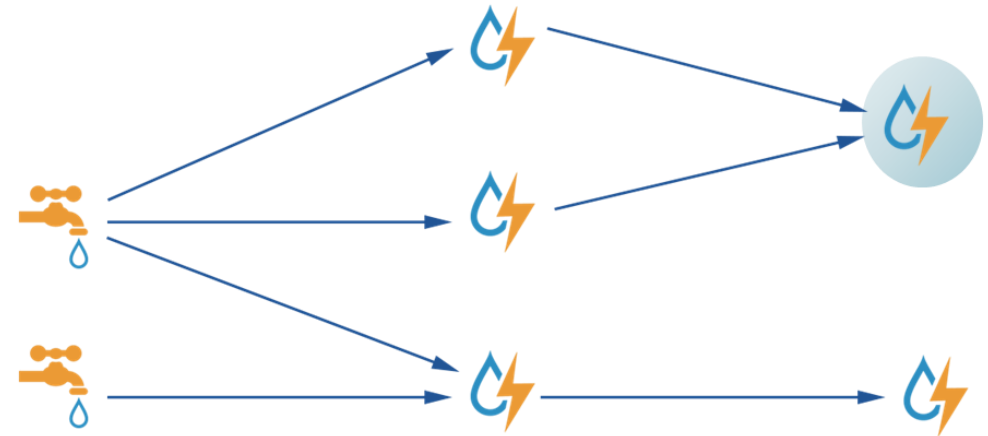
CONCEPTOS BASICOS – FIELD GROUPING

- La distribución depende de algún valor de los elementos de cada tupla
 - { *SomeField:SomeValue*, *SomeClass:A* }
 - { *SomeField:SomeValue*, *SomeClass:B* }



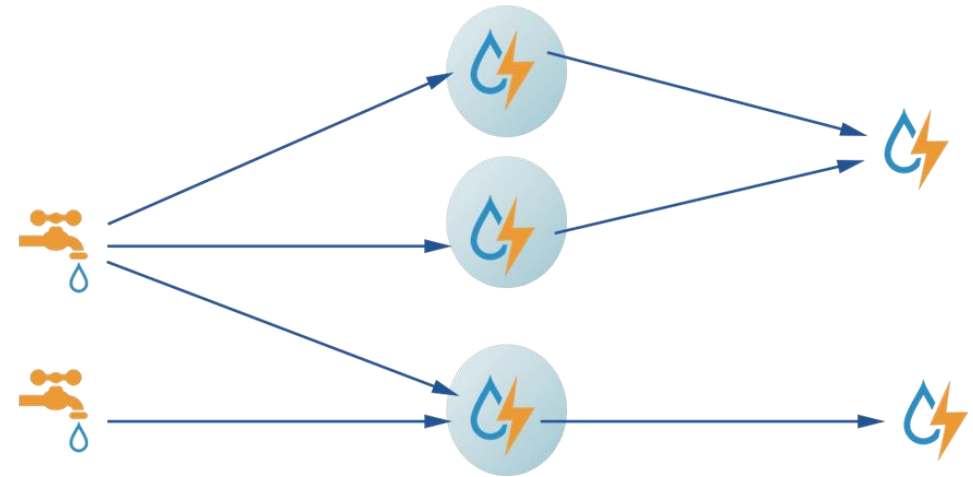
CONCEPTOS BASICOS – GLOBAL GROUPING

- Todos los streams son agrupados en un único bolt



CONCEPTOS BASICOS – ALL GROUPING

- Capa tupla es copiada y enviada a cada bolt



plain concepts

APACHE STORM



AZURE STREAM ANALYTICS

AZURE STREAM ANALYTICS

- Azure Stream Analytics es un motor de proceso de eventos
- Nos permite describir las transformaciones que queremos aplicar utilizando una sintaxis SQL
- Esta integrado con la infraestructura de Azure para gestión de colas de eventos de Azure Event Hubs

AZURE STREAM ANALYTICS

Analisis en
tiempo real

Facilidad de
escalado

Manejado
(PaaS)

Disponibilidad

Bajo coste

Desarrollo
rapido

ANALISIS EN TIEMPO REAL

- Ingesta de millones de eventos por Segundo
 - Hasta 1GB/s
- Nivel de carga variable
 - Escalado para acomodarse a cargas variables
 - Baja latencia en el procesamiento
- Transformaciones, enriquecimiento de los datos, correlación...
 - Correlación entre diferentes streams o con datos de referencia
 - Búsqueda de patrones en tiempo real

FACILIDAD DE ESCALADO

- Aprovecha las capacidades de Azure para el escalado
 - Aumento del numero de recursos
 - Escalado bajo demanda
 - Arquitectura distribuida

streaming unit pool

Scale settings can't be edited while a job is running.

STREAMING UNITS

A Stream Analytics job can be scaled through Streaming Units, which define the amount of processing power a job receives. Each Streaming Unit corresponds to roughly 1 MB/second of throughput.

MANEJADO

- Paradigma PaaS (Platform as a Service)
 - No es necesario tener experiencia en el despliegue
 - No es necesario mantener el software
 - No es necesario hacer ajustes para mejorar el rendimiento

DISPONIBILIDAD

- Entrega de eventos garantizada
 - No se pierden eventos
 - Los eventos se entregan una sola vez
- Disponibilidad garantizada
 - SLA con 99'9% de uptime
 - Auto recuperación en caso de error
 - Gestión del estado embebida, para simplificar y acelerar la recuperación en caso de error

BAJO COSTE

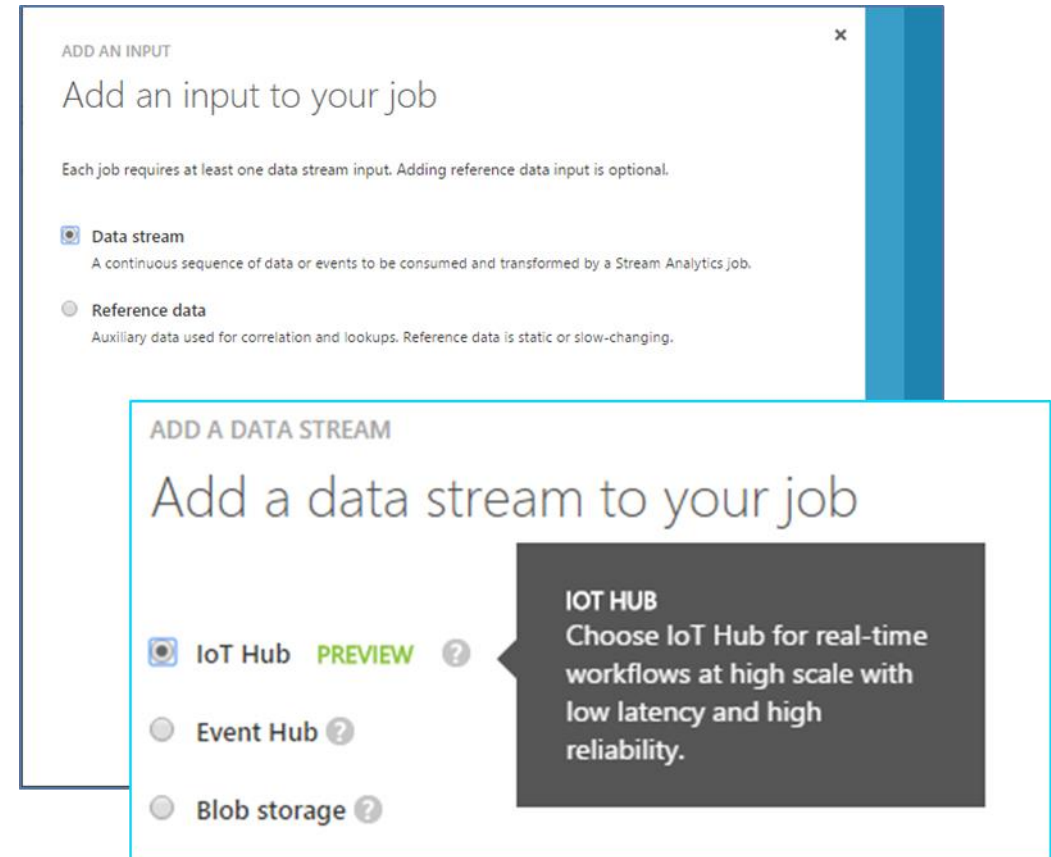
- Pago por uso
 - Si no estamos procesando, no pagamos
- Ventajas de Azure
 - Bajos costes iniciales
 - Posibilidad de escalar de forma incremental

DESARROLLO RÁPIDO

- Lenguaje declarativo similar a SQL
 - De alto nivel
 - Conciso
 - Soporte de primera clase a los streams de eventos y los datos de referencia
- Semánticas temporales incorporadas
 - Windowing y joining
 - Configuración sencilla para tratar con eventos retrasados y/o fuera de orden temporal

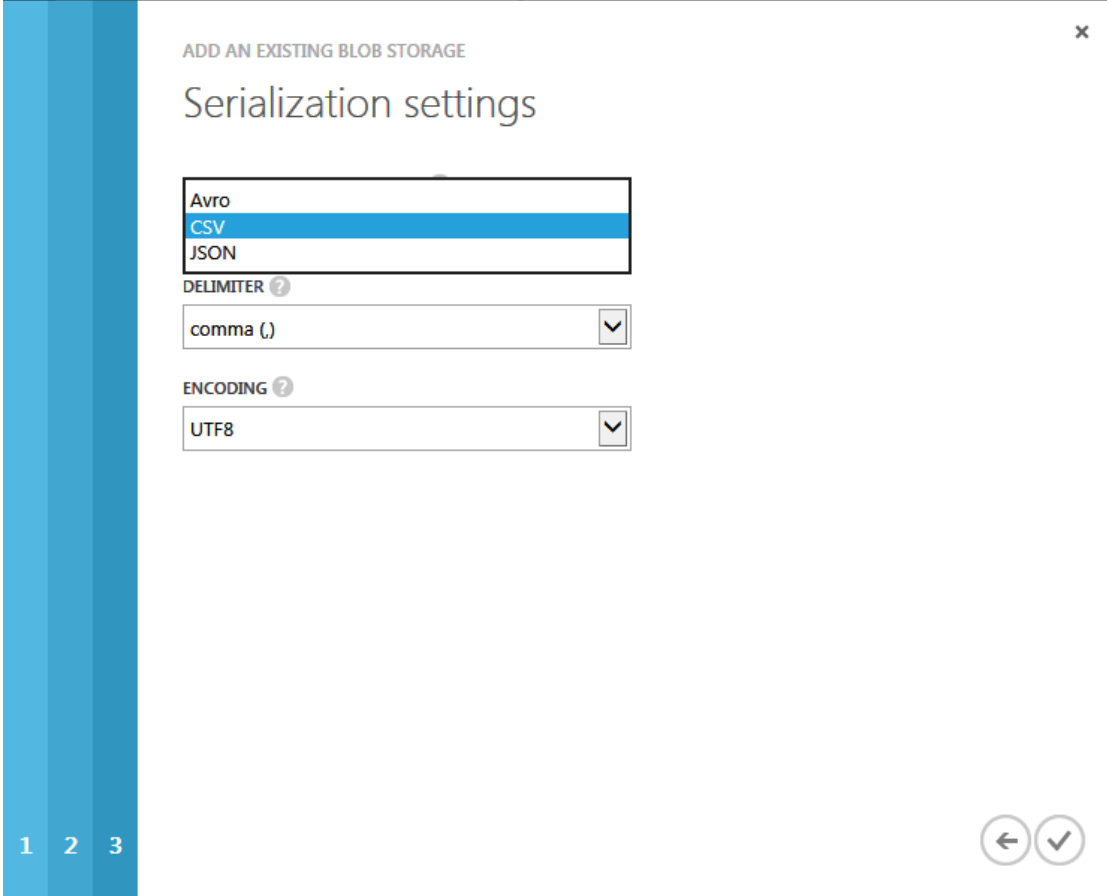
ENTRADAS

- Como entradas de datos de stream
 - Azure Event Hub
 - Azure IoT Hub
 - Azure Blob Storage
- Como entradas datos de referencia
 - Azure Blob Storage.
 - Cacheado para mejorar el rendimiento



DEFINIENDO UN ESQUEMA

- Debemos definir el formato y encoding de los datos de entrada
 - El formato puede ser CSV, JSON o AVRO
 - Para CSV disponemos de distintos delimitadores
 - En el caso de usar CSV o AVRO podemos definir el esquema de los datos



ADD AN EXISTING BLOB STORAGE

Serialization settings

Format selection: Avro, CSV (selected), JSON

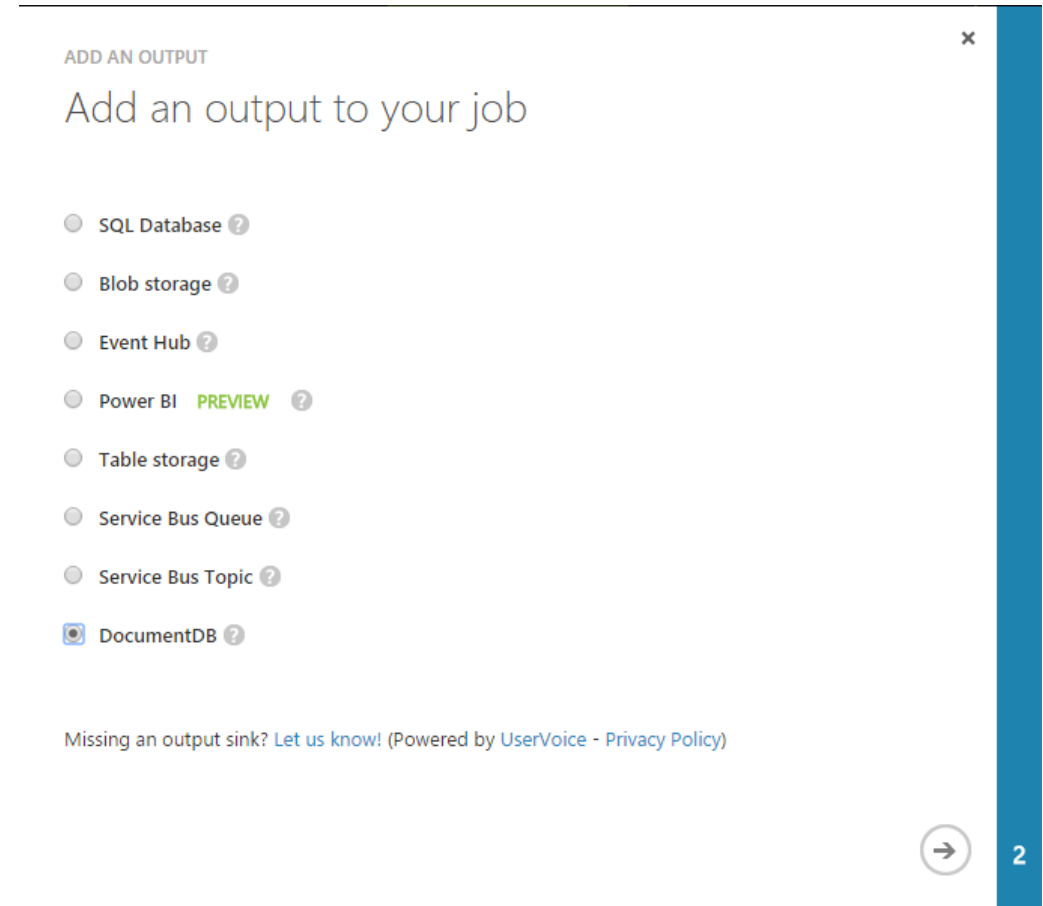
DELIMITER ?
comma (,)

ENCODING ?
UTF8

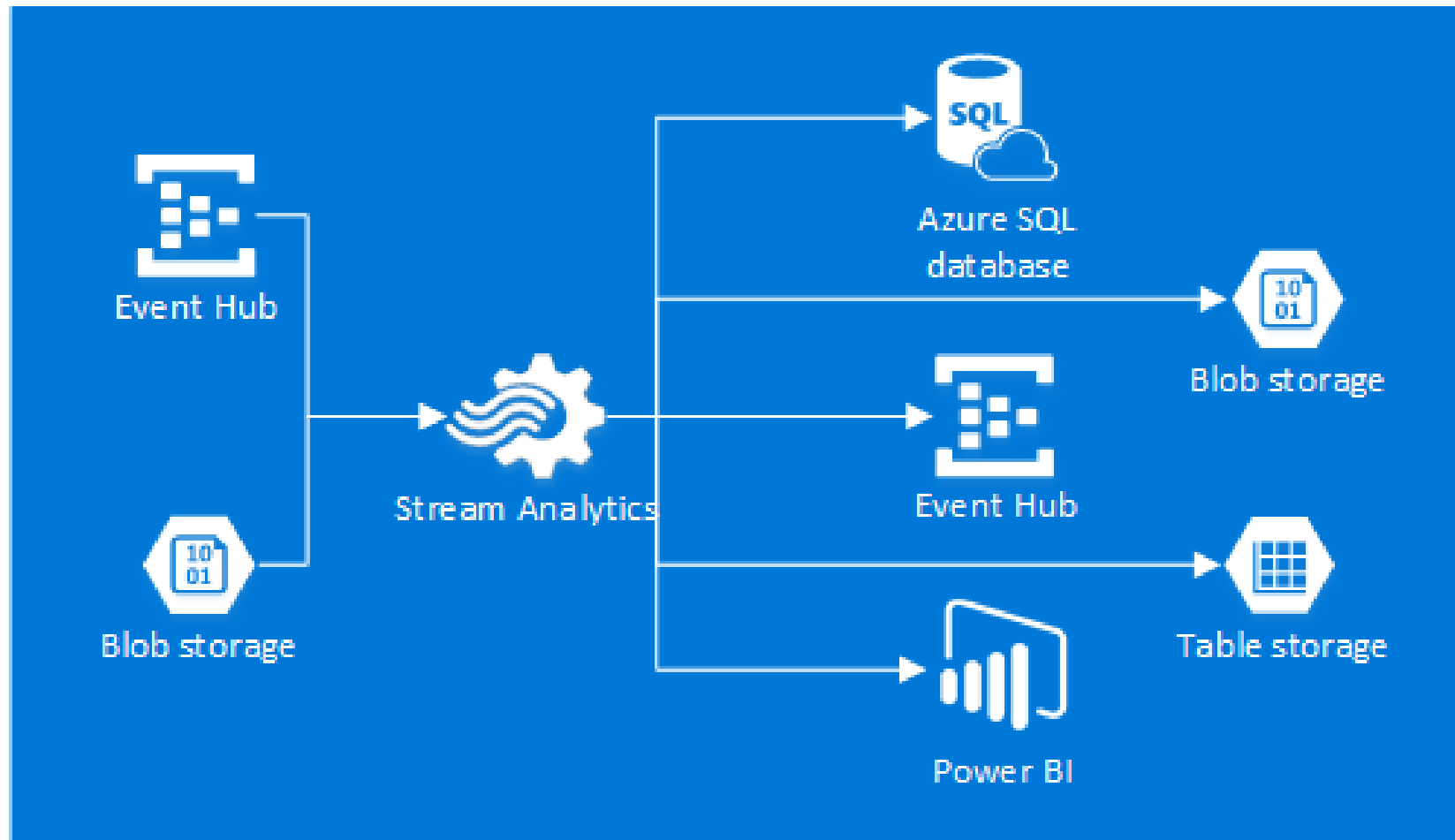
Navigation: 1 2 3 (step 2 is active), back, confirm

SALIDAS

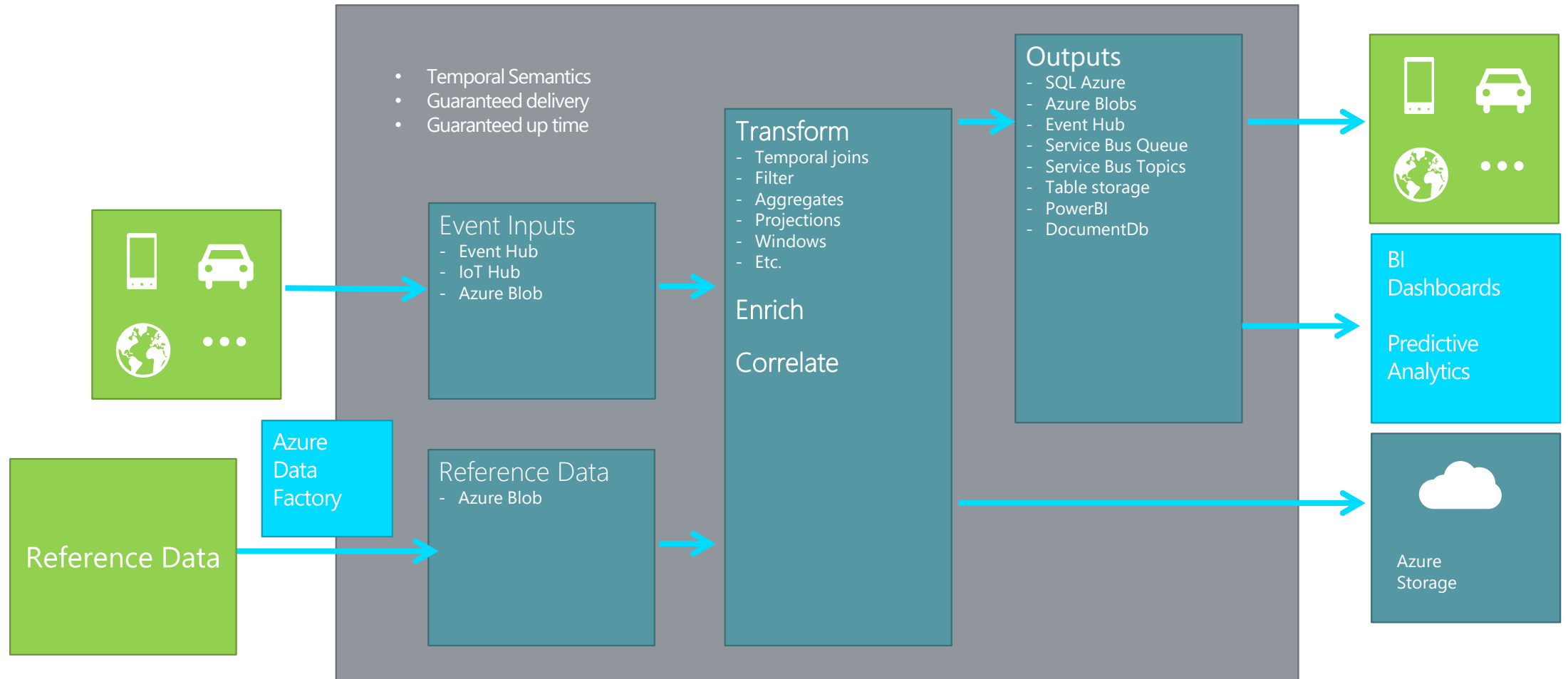
- Azure Blob storage
- Azure Table storage
- SQL database. Para reporting tradicional
- Event hub. Para alertas o notificaciones
- Service Bus Queue
- Service Bus Topic
- PowerBI.com
- DocumentDB



ARQUITECTURA



ARQUITECTURA



EL LENGUAJE

SAQL

DML

- SELECT
- FROM
- WHERE
- GROUP BY
- HAVING
- CASE WHEN THEN ELSE
- INNER/LEFT OUTER JOIN
- UNION
- CROSS/OUTER APPLY
- CAST
- INTO
- ORDER BY ASC, DSC

Scaling Extensions

- WITH
- PARTITION BY
- OVER

Date and Time Functions

- DateName
- DatePart
- Day
- Month
- Year
- DateTimeFromParts
- DateDiff
- DateAdd

Temporal Functions

- Lag, IsFirst
- CollectTop

Windowing Extensions

- TumblingWindow
- HoppingWindow
- SlidingWindow

Aggregate Functions

- Sum
- Count
- Avg
- Min
- Max
- StDev
- StDevP
- Var
- VarP

String Functions

- Len
- Concat
- CharIndex
- Substring
- PatIndex

TIPOS DE DATOS

- Los datos de entrada se castean a uno de estos tipos
- Podemos establecerlos mediante un CREATE TABLE
 - No estemos creando una tabla, simplemente un mapeo de tipo de datos

Tipo	Descripcion
bigint	Integers in the range -2^{63} (-9,223,372,036,854,775,808) to $2^{63}-1$ (9,223,372,036,854,775,807).
float	Floating point numbers in the range - 1.79E+308 to -2.23E-308, 0, and 2.23E-308 to 1.79E+308.
nvarchar(max)	Text values, comprised of Unicode characters. Note: A value other than max is not supported.
datetime	Defines a date that is combined with a time of day with fractional seconds that is based on a 24-hour clock and relative to UTC (time zone offset 0).

WHERE

- Especifica condiciones para las filas devueltas en un SELECT
- Nos permite establecer filtros
- No hay limite al numero de predicados en el WHERE

```
SELECT UserName, TimeZone  
FROM InputStream  
WHERE Topic = 'XBox'
```


INTO

- Canaliza datos entre una entrada y una salida
- Podemos tener múltiples salidas
 - Utilizamos el INTO para llevar cada SELECT a su destino
 - Por ejemplo, algunos eventos se van al Blob Storage para analizarse posteriormente, otros van directamente a Event Hub

```
SELECT UserName, TimeZone  
INTO Output  
FROM InputStream  
WHERE Topic = 'XBox'
```

JOIN

- Nos permite combinar múltiples streams, o un stream con datos de referencia
 - Podemos especificar la ventana temporal en la que lo aplicaremos, mediante `DateDiff`

query

```
1 SELECT events.id, events.content, events.tempo, heartbeat.level
2 FROM [events]
3 TIMESTAMP BY EventEnqueuedUtcTime
4 JOIN [Heartbeat]
5 TIMESTAMP BY MeasureDate
6 ON DateDiff(Minute, events, heartbeat) between 0 and 1
```

DATOS DE REFERENCIA

- Información accesoria al stream
 - Estática o lentamente cambiante
 - Almacenada en ficheros CSV o JSON en Azure Blob Storage
- Podemos utilizarla junto con los datos del stream

```
SELECT myRefData.Name, myStream.Value
FROM myStream
JOIN myRefData
      ON myStream.myKey =
myRefData.myKey
```

UNION

- Combina los resultados de dos o mas queries en un único conjunto de resultados que incluye filas de ambas
- El número de columnas y su orden debe ser el mismo en todas las queries
- Los tipos de datos deben de ser compatibles
- Si no utilizamos UNION ALL los duplicados se eliminan

UNION

TollId	EntryTime	LicensePlate	...
1	2014-09-10 12:01:00.000	JNB 7001	...
1	2014-09-10 12:02:00.000	YZZ 1001	...
3	2014-09-10 12:02:00.000	ABC 1004	...

TollId	ExitTime	LicensePlate
1	2009-06-25 12:03:00.000	JNB 7001
1	2009-06-25 12:03:00.000	YZZ 1001
3	2009-06-25 12:04:00.000	ABC 1004

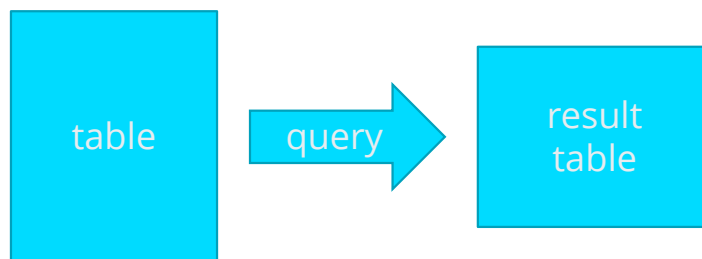
SELECT TollId, ENTime AS Time , LicensePlate FROM EntryStream TIMESTAMP BY ENTime
UNION
SELECT TollId, EXTime AS Time , LicensePlate FROM ExitStream TIMESTAMP BY EXTime

TollId	Time	LicensePlate
1	2014-09-10 12:01:00.000	JNB 7001
1	2014-09-10 12:02:00.000	YZZ 1001
3	2014-09-10 12:02:00.000	ABC 1004
1	2009-06-25 12:03:00.000	JNB 7001
1	2009-06-25 12:03:00.000	YZZ 1001
3	2009-06-25 12:04:00.000	ABC 1004

LA GESTIÓN TEMPORAL

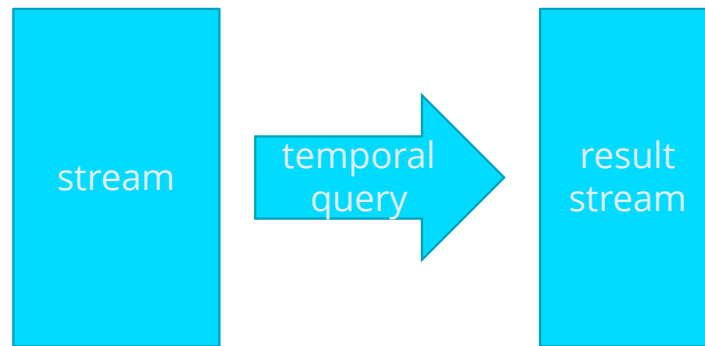
CONSULTAS TRADICIONALES

- Las consultas tradicionales asumen que los datos no cambian mientras las consultas
 - La consulta trabaja sobre un estado fijo
 - En el caso de que los datos varíen, trabajamos sobre un snapshot o una transacción
 - Consultando un estado finito, la consulta termina en un tiempo finito



CONSULTAS TEMPORALES

- Cuando analizamos un stream de datos, trabajamos con una cantidad de datos potencialmente infinita
 - La consulta nunca terminaria
- Para solucionar esto trabajamos con ventanas temporales



ARRIVAL TIME VS APPLICATION TIME

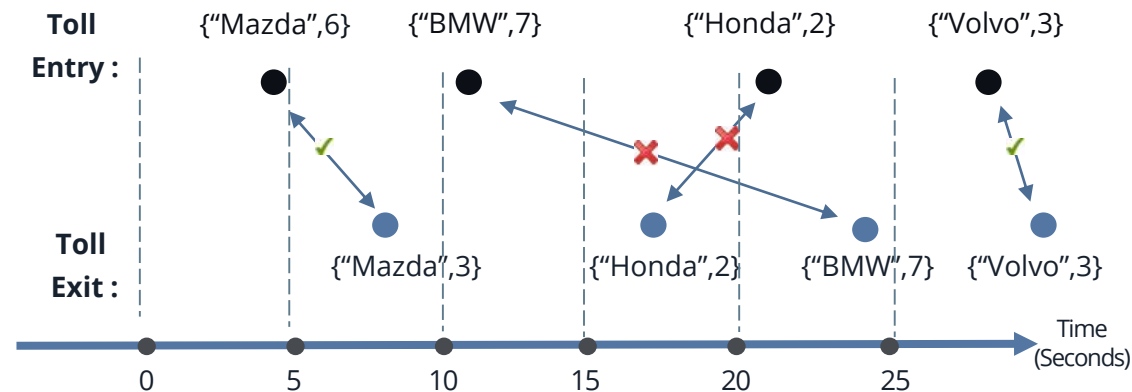
- Todos los eventos tienen un timestamp, accesible mediante el campo `System.Timestamp`
- Por defecto es el tiempo de entrada al sistema
 - Arrival time
 - Para eventos, el que le asigne el Event Hub en su entrada
 - Para datos en blob storage, el Last Modified del blob
- En muchas ocasiones este no es el timestamp que nos interesa
 - Application time
 - Generado mediante la expresión `TIMESTAMP BY`

JOINS TEMPORALES

- Hemos visto como utilizar JOIN para combinar eventos de una o mas entradas
- En Stream Analytics, JOIN es temporal
 - Cada JOIN debe especificar limites temporales para la unión
 - En la clausula ON, utilizando DATEDIFF

JOINS TEMPORALES

```
SELECT Make
FROM EntryStream ES TIMESTAMP BY EntryTime
JOIN ExitStream EX TIMESTAMP BY ExitTime
ON ES.Make= EX.Make
AND DATEDIFF(second,ES,EX) BETWEEN 0 AND 10
```



"Honda" – Not in result because event in Exit stream precedes event in Entry Stream

"BMW" – Not in result because Entry and Exit stream events > 10 seconds apart

Query Result = [Mazda, Volvo]

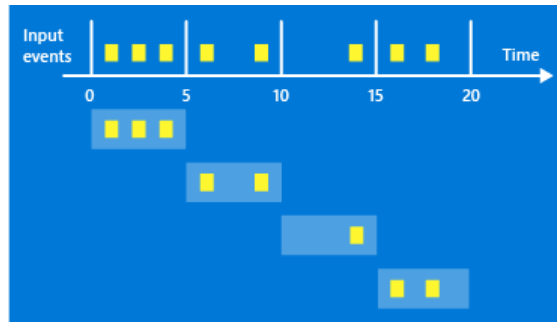
FUNCIONES DE VENTANA

- A la hora de procesar operaciones sobre conjuntos que llegan en tiempo real, un escenario común es tener un subconjunto temporal
 - Suma de todos los datos en una hora determinada
- ¿Como definimos un subconjunto temporal en un stream?
 - Mediante funciones de ventana

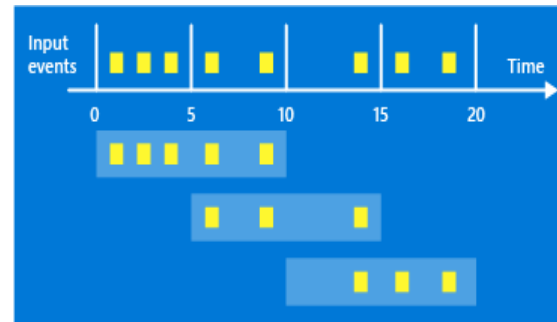
FUNCIONES DE VENTANA

- Una ventana contiene datos de eventos a lo largo de una escala de tiempo
- Cada operación de ventana genera un evento al final de la ventana
 - Con el TimeStamp de la ventana
- Todas tienen una longitud fija
- Se generan con la clausula GROUP BY

FUNCIONES DE VENTANA



Tumbling window
Aggregate per time interval



Hopping window
Schedule overlapping windows



Sliding window
Windows constant re-evaluated

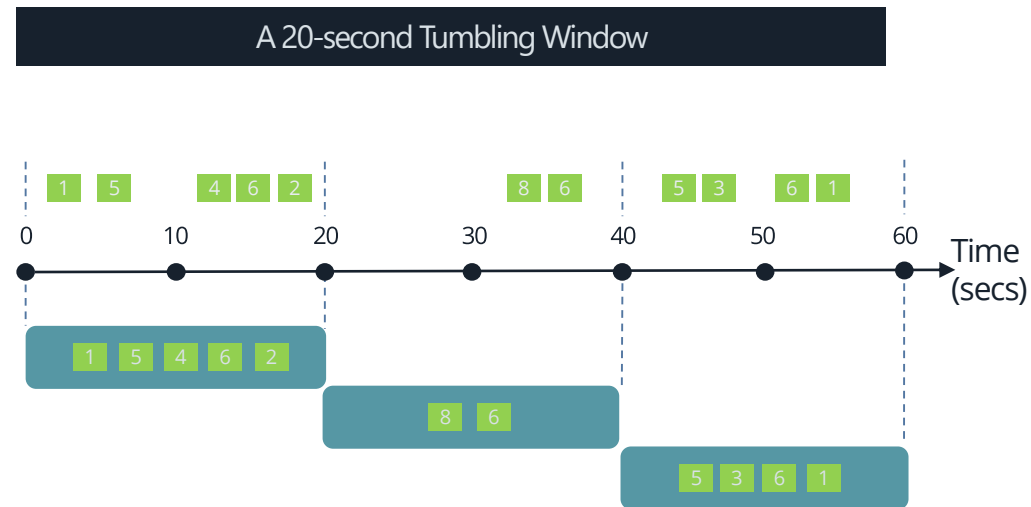
TUMBLING WINDOW

- Se repiten
- No se superponen
- Un evento pertenece a una sola ventana

Query: Count the total number of vehicles entering each toll booth every interval of 20 seconds.

```
SELECT TollId, COUNT(*)  
FROM EntryStream TIMESTAMP BY EntryTime  
GROUP BY TollId, TumblingWindow(second, 20)
```

TUMBLING WINDOW



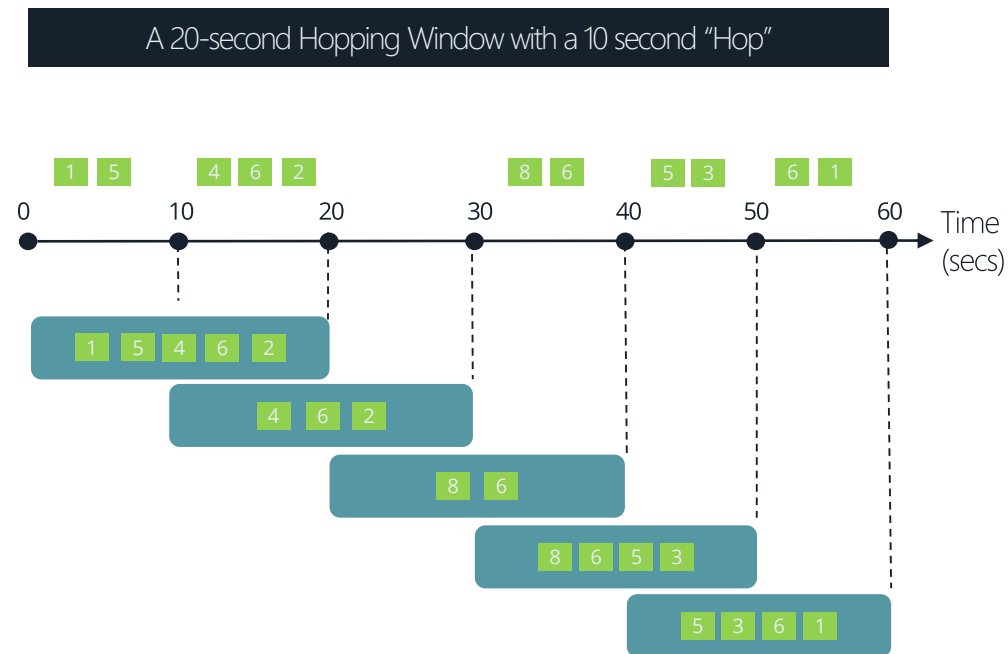
HOPPING WINDOW

- Se repiten
- Se pueden superponer
- Se mueven hacia adelante en saltos fijos
 - Si el tamaño del salto es igual al de la ventana, es igual a la anterior

QUERY: Count the number of vehicles entering each toll booth every interval of 20 seconds; update results every 10 seconds

```
SELECT COUNT(*), TollId  
FROM EntryStream TIMESTAMP BY EntryTime  
GROUP BY TollId, HoppingWindow (second, 20,10)
```

HOPPING WINDOW



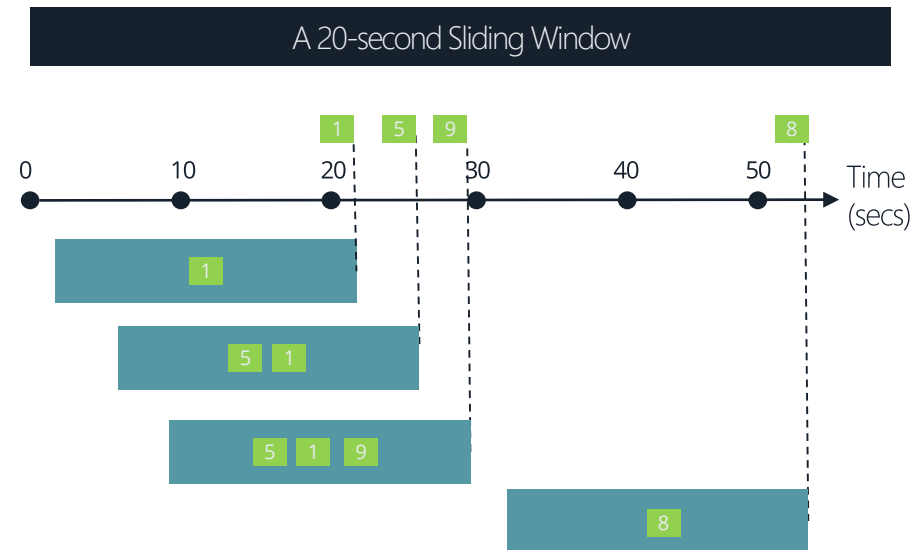
SLIDING WINDOW

- Se mueve continuamente hacia delante de ϵ (epsilon) en ϵ
 - $\epsilon = 1/100$ de nanosegundo
- Genera una salida solo durante la ocurrencia de un evento
- Cada ventana tiene al menos un evento
- Los eventos pueden pertenecer a mas de una ventana

Query: Find all the toll booths which have served more than 10 vehicles in the last 20 seconds

```
SELECT TollId, Count(*)  
FROM EntryStream ES  
GROUP BY TollId, SlidingWindow (second, 20)  
HAVING Count(*) > 10
```

SLIDING WINDOW



ESCALANDO EL ANÁLISIS

STREAMING UNIT

- Medida de los recursos de computación disponibles para procesar un trabajo
 - Una Streaming Unit procesa hasta 1Mb/Segundo
- Por defecto, cada trabajo consiste de una Streaming Unit
- El número total depende de
 - Ritmo de entrada de eventos
 - Complejidad de la consulta

SUBCONSULTAS

- Una consulta puede tener mas de un paso
 - Un paso es una subconsulta definida utilizando WITH
 - CTE, Common Table Expression
- La única query fuera del WITH es también un paso
- Nos permite desarrollar queries complejas al utilizar resultados intermedios
 - La salida de cada paso se puede enviar a múltiples lugares utilizando INTO

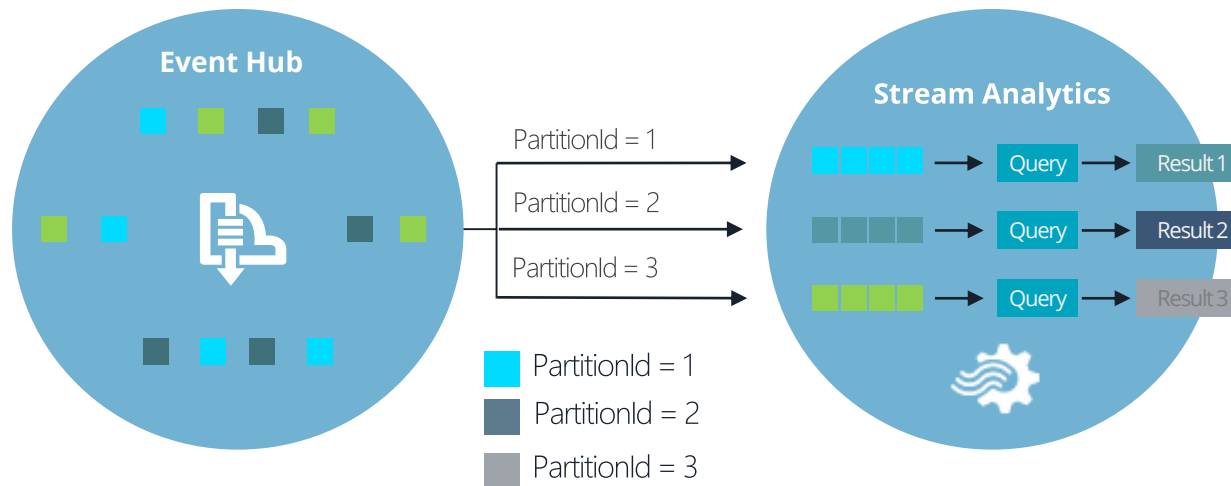
```
WITH Step1 AS (  
    SELECT Count(*) AS CountTweets, Topic  
    FROM TwitterStream PARTITION BY PartitionId  
    GROUP BY TumblingWindow(second, 3), Topic, PartitionId  
)  
Step2 AS (  
    SELECT Avg(CountTweets)  
    FROM Step1  
    GROUP BY TumblingWindow(minute, 3)  
)  
SELECT * INTO Output1 FROM Step1  
SELECT * INTO Output2 FROM Step2  
SELECT * INTO Output3 FROM Step2
```

PARTICIONADO

- Cuando una consulta esta particionada, los eventos de entrada se procesan y agregan en grupos separados
 - Se genera un evento de salida por cada partición
- La consulta debe utilizar la sintaxis PARTITION BY
- Si la entrada es un Event Hub particionado, podemos escribir consultas y subconsultas particionadas
- Nos permite maximizar el numero de Streaming Units a utilizar

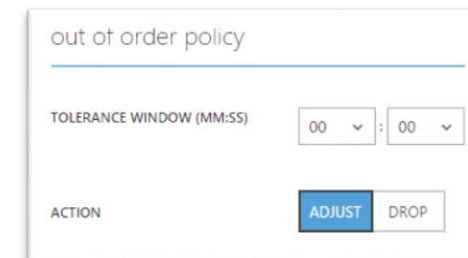
PARTICIONADO

```
SELECT Count(*) AS Count, Topic
FROM TwitterStream PARTITION BY PartitionId
GROUP BY TumblingWindow(minute, 3), Topic, PartitionId
```



ENTRADAS DESORDENADAS

- Podemos configurar una tolerancia en la Out of Order Policy para tratar con eventos desordenados
 - Por defecto es 0, lo que indica que esperamos que los eventos siempre lleguen en orden
- Utilizar un valor mayor que 0 permite a Azure Stream Analytics corregir el desorden
 - Habrá un buffer del tamaño especificado y dentro de ese buffer se reordenaran los eventos en base al TimeStamp seleccionado
 - Esto ocasiona un retraso en la salida



The screenshot shows a configuration window titled "out of order policy". It contains a "TOLERANCE WINDOW (MM:SS)" field with two dropdown menus, both set to "00". Below this is an "ACTION" section with two buttons: "ADJUST" (highlighted in blue) and "DROP".

plain concepts

AZURE STREAM ANALYTICS



STREAM ANALYTICS VS APACHE STORM



OPENTSDDB

SERIES TEMPORALES

- Las series temporales son secuencias de datos medidos en un momento determinado y ordenados cronológicamente
 - Eventos generados por sensores o software, ficheros de log...
- Gran cantidad de series temporales se generan cada día
 - Infraestructuras en la nube
 - Sensores
 - Dispositivos IoT

SERIES TEMPORALES

- La utilidad de las series temporales
 - Aprovechar el conocimiento de eventos pasados (vista histórica)
 - Junto con eventos del presente (vista en tiempo real)
 - Para predecir que ocurrirá en el futuro (análisis predictivo)
- Las series temporales se utilizan en múltiples ámbitos
 - Mercados financieros y economía
 - Seguros
 - Climatología
 - Centros sanitarios
- La aplicación de Big Data a las series temporales es un paso lógico

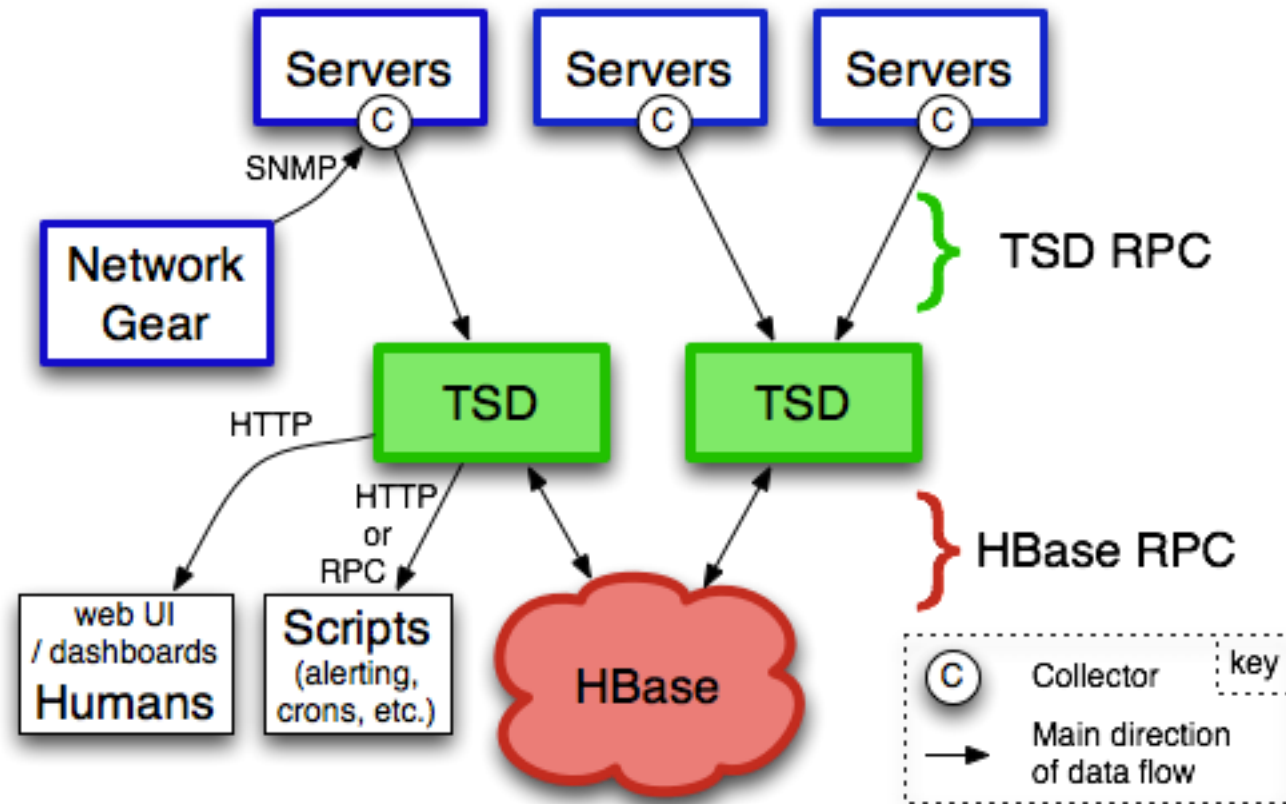
PUNTOS DE DATOS

- Las series temporales se componen de puntos de datos
- Cada uno contiene
 - La medida a la que se refiere
 - Etiquetas (pares clave/valor) con información adicional sobre la medida
 - El instante del tiempo en que se tomo la medida
 - El valor

OPENTSDDB

- Open Source Time Series Database
- Una base de datos para almacenar series temporales
- Almacena trillones de puntos de datos
 - A gran velocidad
 - Sin perder precisión
- Escala utilizando HBase

OPENTSDB



DESPLEGANDO OPENTSDDB

- OpenTSDB no viene por defecto en CDH
- Necesitamos desplegarlo y configurarlo por nuestra cuenta
- Es un proceso sencillo
 - Siempre que todo vaya bien 😊

DESPLEGANDO OPENTSDDB

- Lo primero es descargar y compilar OpenTSDB

```
sudo ln -sf /usr/share/zoneinfo/UTC /etc/localtime
```

```
sudo yum install git automake gnuplot
```

```
sudo yum install java-devel
```

```
git clone https://github.com/OpenTSDB/opentsdb.git
```

```
cd opentsdb
```

```
./build.sh
```

DESPLEGANDO OPENTSDDB

- OpenTSDB almacena los datos Hbase
 - Una BD No-SQL
- Necesitamos configurarlo en el cluster
 - Configuramos Hbase desde Cloudera Manager
 - Reiniciamos Hbase
 - Reiniciamos Zookeeper

DESPLEGANDO OPENTSDDB

- Configuramos y arrancamos OpenTSDB

```
export JAVA_HOME=/usr/java/jdk1.7.0_67-cloudera
env COMPRESSION=NONE HBASE_HOME=/opt/cloudera/parcels/CDH/lib/hbase
./src/create_table.sh
mkdir /tmp/tsd
nohup ./build/tsdb tsd --port=4242 --staticroot=build/staticroot/ --
cachedir=/tmp/tsd/ --zkbasedir=/hbase --zkquorum=127.0.0.1:2181 --auto-
metric &
```

DESPLEGANDO OPENTSDDB

- Verificamos que esta arrancado
 - `netstat -lpn | grep 4242`
- Abrimos el Puerto 4242 en el master node
 - Security Group - Inbound Rules
 - VNet - Inbound Rules
- <http://mstrainingmadrid-mn0.northeurope.cloudapp.azure.com:4242>

ENVIANDO DATOS A OPENTSDDB

- API HTTP
 - `curl -i -H "Content-Type: application/json" -X POST -d '{"metric": "Room.Temperature", "timestamp": 1462870380, "value": 18, "tags": { "sensor": "s0" }}'` <http://mstrainingmadrid-mn0.northeurope.cloudapp.azure.com:4242/api/put/?details>
- C# App
 - <https://github.com/hanuk/tsdbwriter>

plain concepts

OPENTSDDB



GRAFANA

GRAFANA

- Sistema de visualización de datos mediante dashboards
- Dispone de un potente sistema de plugins
 - Para obtener datos
 - Para visualizarlos

DESPLEGANDO GRAFANA

- Grafana tampoco viene por defecto en CDH

```
sudo yum install numpy
```

```
sudo yum install
```

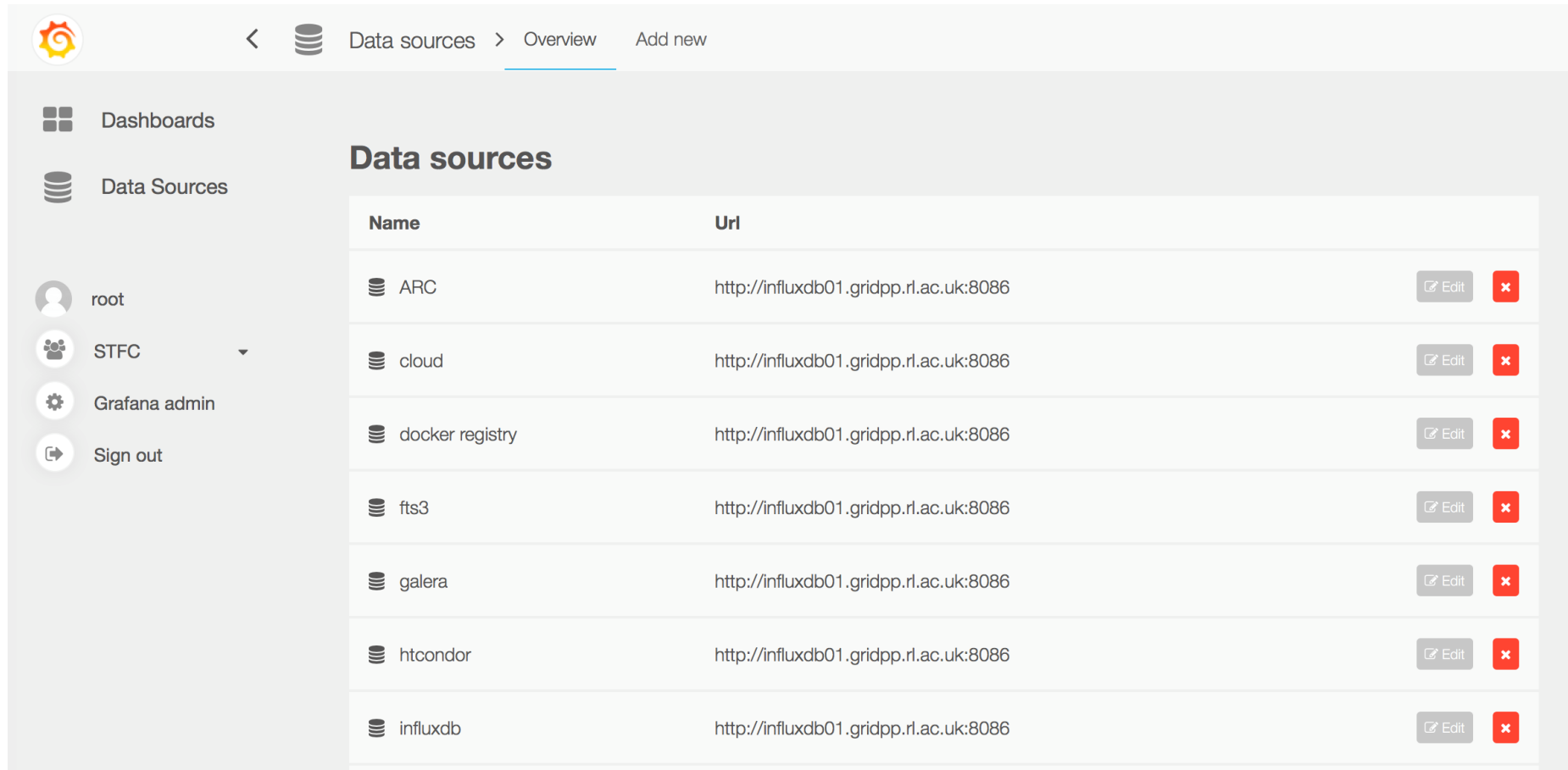
https://grafanarel.s3.amazonaws.com/builds/grafana-3.1.1-1470047149.x86_64.rpm

```
sudo service grafana-server start
```

DESPLEGANDO GRAFANA

- Verificamos que esta arrancado
 - `netstat -ltn | grep 3000`
- Abrimos el Puerto 3000 en el master node
 - Security Group - Inbound Rules
 - VNet - Inbound Rules
- <http://mstrainingmadrid-mn0.northeurope.cloudapp.azure.com:3000>


GRAFANA – DATA SOURCES





The screenshot shows the Grafana web interface. At the top, there's a navigation bar with the Grafana logo, a back arrow, a database icon, and the text 'Data sources > Overview Add new'. On the left sidebar, there are links for 'Dashboards' and 'Data Sources' (which is active), along with user information for 'root' and a list of roles: 'STFC', 'Grafana admin', and 'Sign out'. The main content area is titled 'Data sources' and contains a table with two columns: 'Name' and 'Url'. The table lists seven data sources, all of which are InfluxDB instances with the same URL. Each row has an 'Edit' button and a delete button (a red square with a white 'x').


Name	Url	Actions
ARC	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
cloud	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
docker registry	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
fts3	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
galera	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
htcondor	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete
influxdb	http://influxdb01.gridpp.rl.ac.uk:8086	Edit Delete


GRAFANA – ADDING A DATABASE





<  Data sources > Overview Add new Edit


 Dashboards

 Data Sources

 root

 STFC ▾

 Grafana admin

 Sign out

Edit data source

Name	galera	Default	<input type="checkbox"/>
Type	InfluxDB 0.9.x		

Http settings

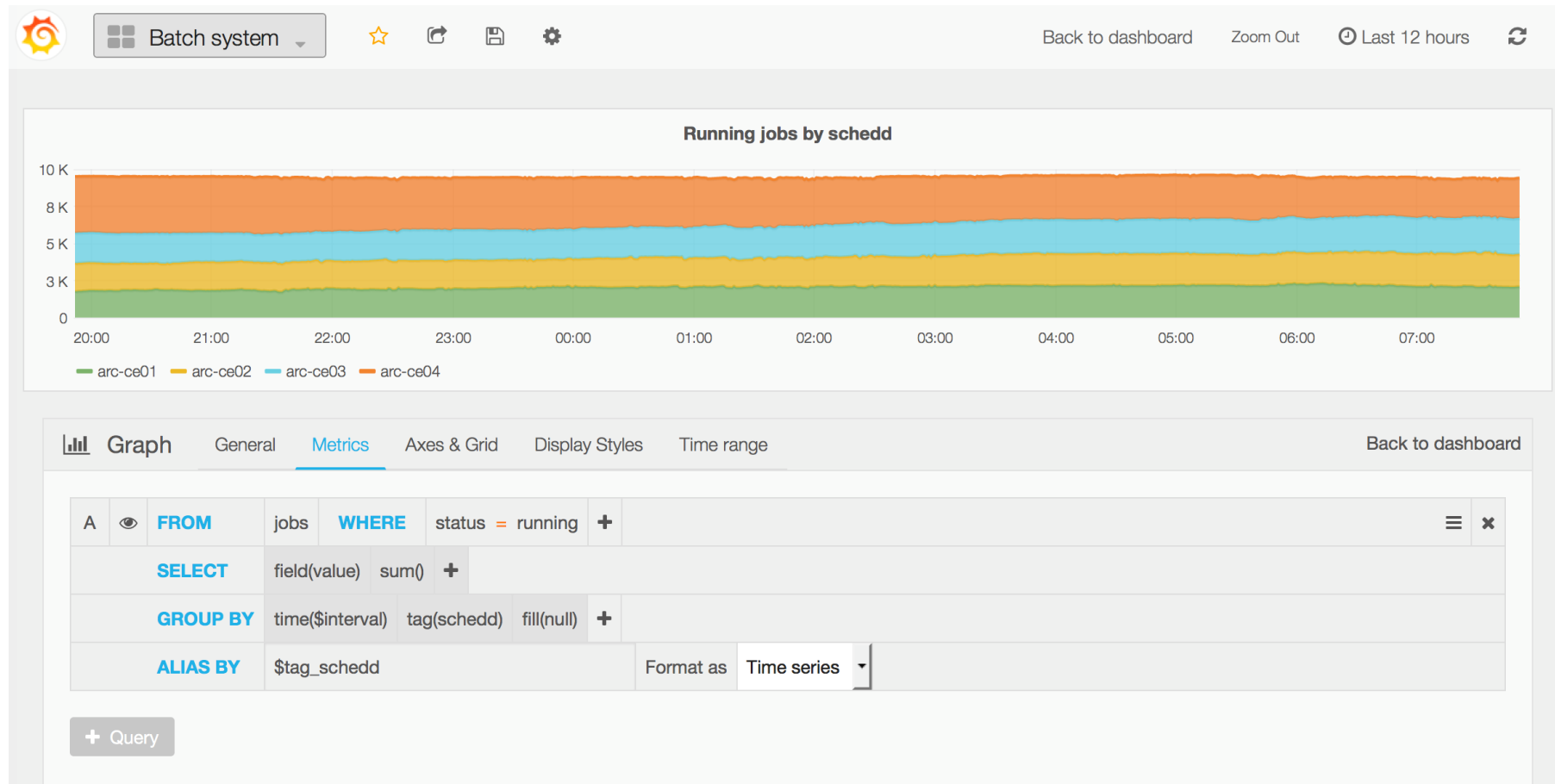
Url	http://influxdb01.gridpp.rl.ac.uk:8086	Access ?	proxy
Http Auth	Basic Auth <input type="checkbox"/>	With Credentials <input type="checkbox"/>	

InfluxDB Details

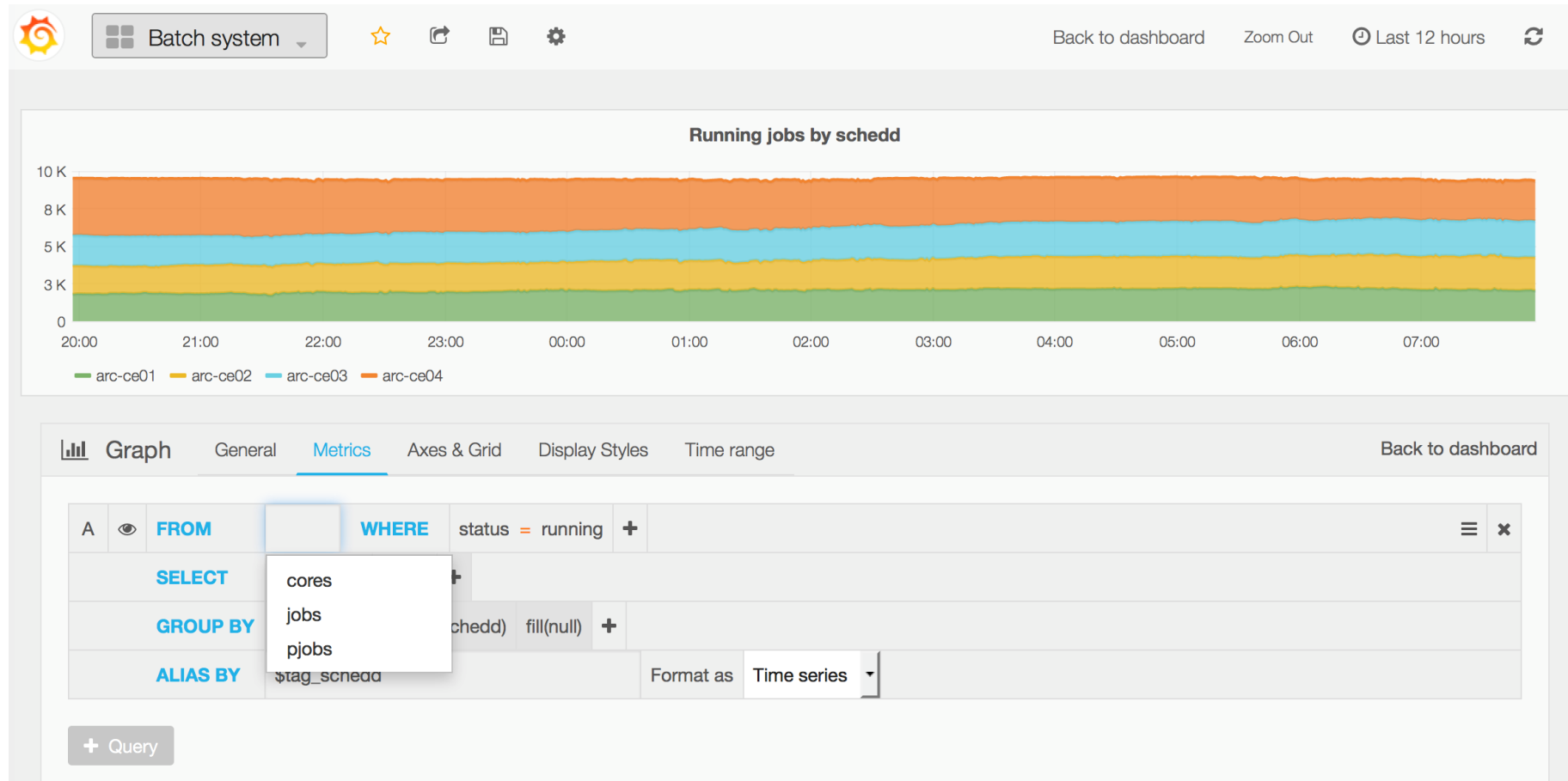
Database	galera		
User	reader	Password

Save Test Connection Cancel







GRAFANA – MAKING A PLOT




GRAFANA – MAKING A PLOT



TEMPLATING




 ARC CEs     Zoom Out Last 12 hours 


</> Templating Variables host + New 

Variable

Name	host	Type	query	Data source	ARC
------	------	------	-------	-------------	-----

Value Options

Query	show tag values from jobs with key=host				
Regex 	/.*-(.*)-.* /				
All value 	(arc-ce01 arc-ce02 arc-ce03 arc-ce04)	All format	regex values		
Refresh on load <input type="checkbox"/> 					

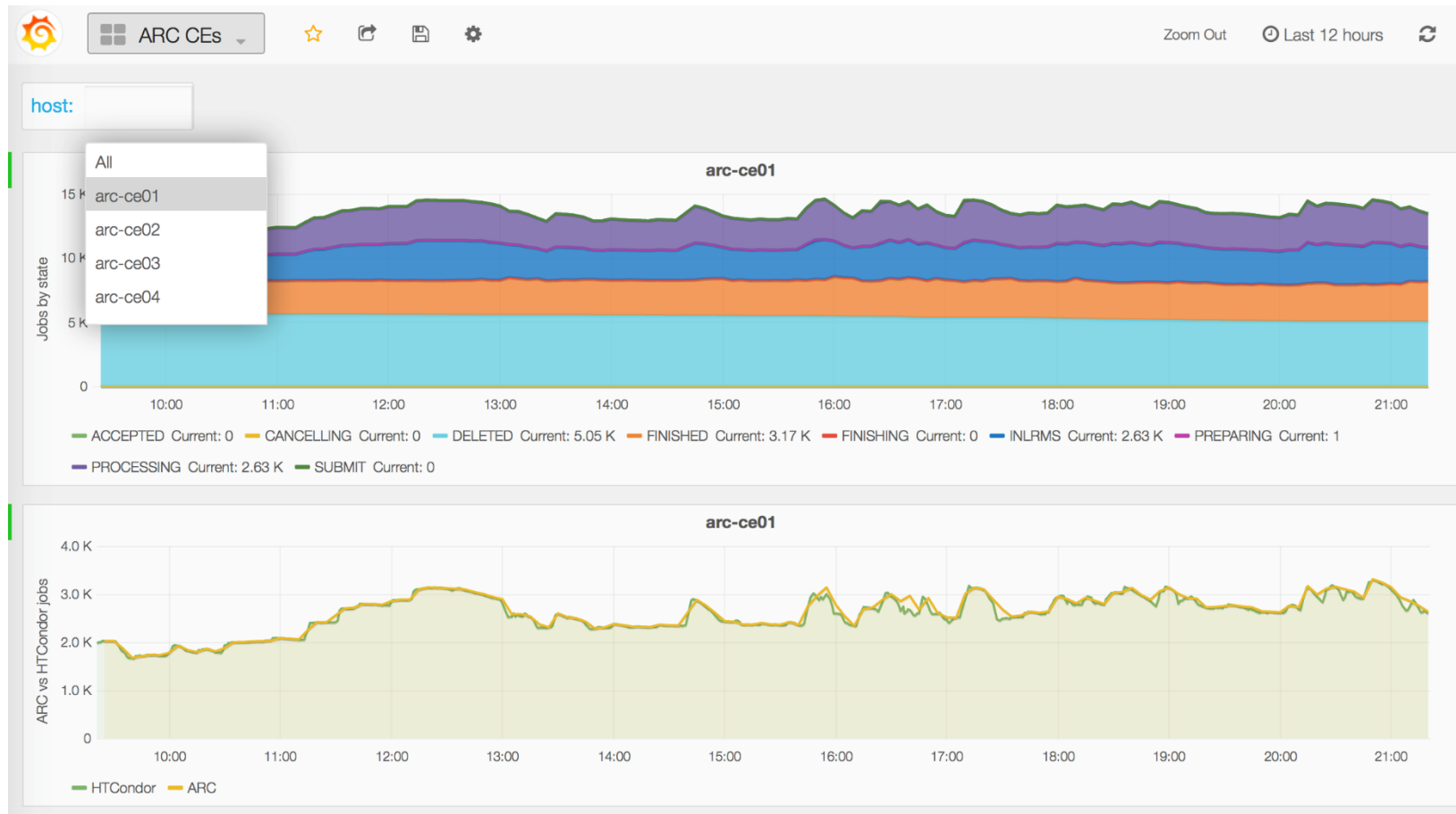
Multi-value selection  **Display options**

Enable <input type="checkbox"/>	Variable Label	Hide label <input type="checkbox"/>
---------------------------------	----------------	-------------------------------------

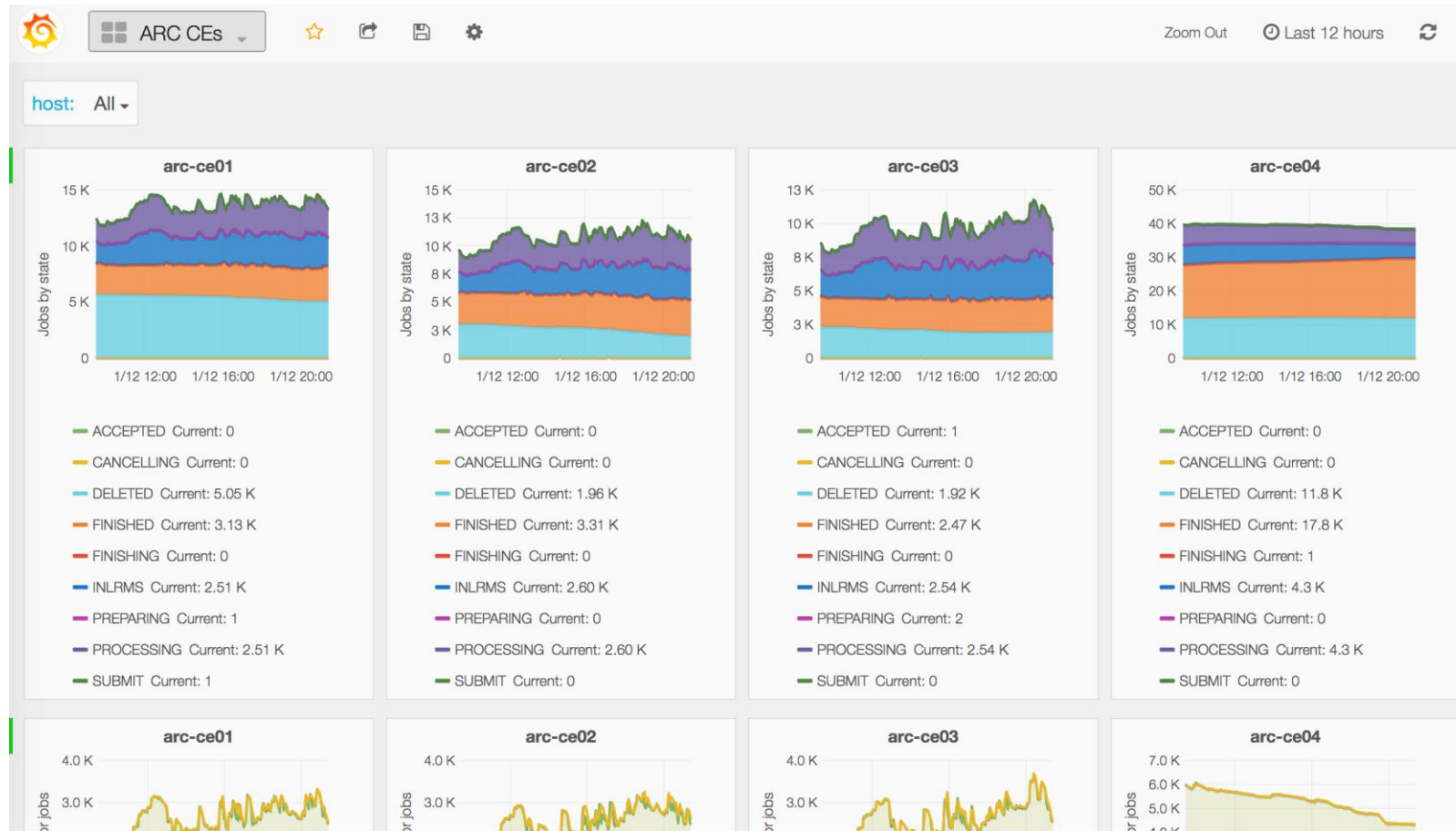
Value groups/tags (Experimental feature)

Enable <input type="checkbox"/>

TEMPLATING



TEMPLATING



plain concepts

GRAFANA



POWER BI

POWERBI - ANTES

PowerQuery
(SSIS)

PowerPivot
(SSAS)

PowerView
(SSRS)

POWERBI - AHORA

Power BI
Website

Power BI
Desktop

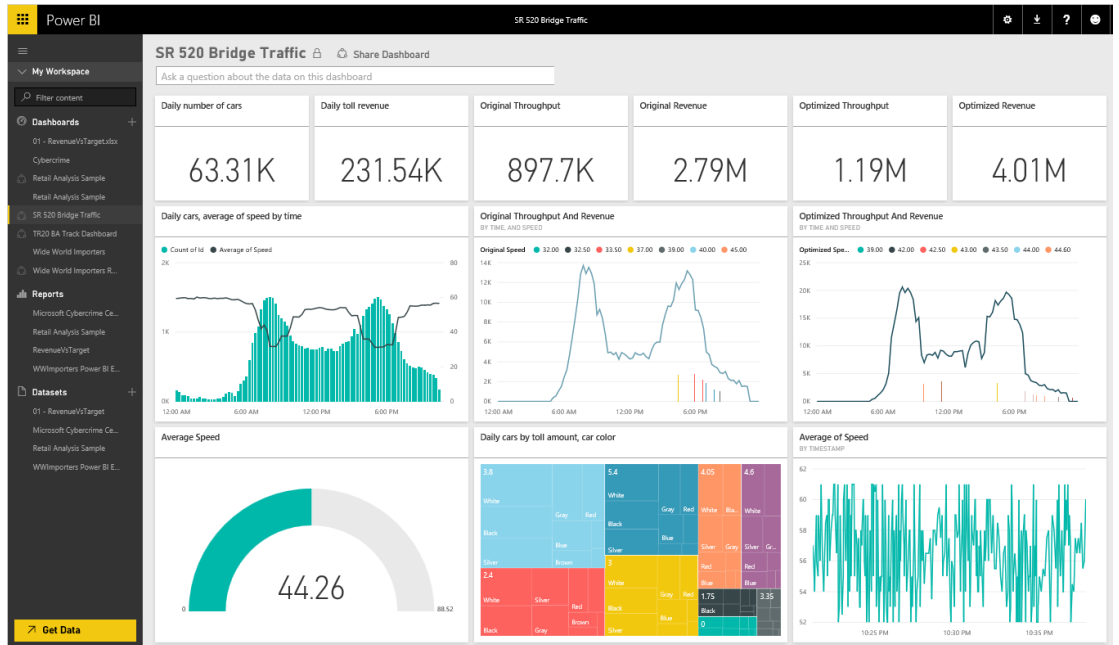
Power BI
Mobile

¿QUE ES POWER BI?

- La nueva generación de herramientas de BI de Microsoft
 - Tanto para frontend (dashboards, reportes)
 - Como para backend (ETL, mashups)
- Reemplaza (de forma no oficial) a PowerView, PowerPivot y PowerQuery
 - Difíciles de instalar
 - Difíciles de utilizar

¿QUE ES POWER BI?

- Servicio de Business Intelligence en la nube
- Acceso rápido y sencillo a nuestros datos
- Exploración y descubrimiento de datos
- Obtención de insights desde distintos dispositivos (web, móvil, escritorio...)
- Adaptado a entornos colaborativos



BENEFICIOS DE POWER BI



Dashboards y reportes predefinidos



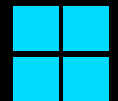
Actualizaciones del dashboard en tiempo real



Conexión segura a las fuentes de datos, on-premises o en la nube



Exploración de datos intuitiva utilizando lenguaje natural





Integrado con Azure









ESQUEMA POWER BI


Fuentes de datos

-  SaaS solutions
e.g. Salesforce, GitHub, Google
-  On-premises data
e.g. Analysis Services
-  Content packs
Corporate data sources or external data services
-  Azure services
Azure SQL, Stream Analytics...
-  Excel files
Workbook data / data models
-  Power BI Desktop files
Data from files, databases, Azure, and other sources

Power BI

-  Content packs
-  Live dashboards
-  Visualizations
-  Reports
-  Datasets
-  Data refresh

 Natural language query

 Sharing & collaboration



COMPONENTES POWER BI

Power BI
Website

Power BI
Desktop

Power BI
Mobile

COMPONENTES POWER BI

- Herramienta web (<http://powerbi.com>) para conectarnos a un conjunto de datos y crear un reporte
- Acceso a los dashboards y reportes desde cualquier lugar
- Q&A, consultas contra los conjuntos de datos para crear elementos del dashboard
- Requiere una cuenta corporativa para crear una cuenta

Power BI
Website

Power BI
Desktop

Power BI
Mobile

COMPONENTES POWER BI

- Herramienta de escritorio
- Permite modelado mas intenso, ETL
- Mas opciones de Fuente de datos
- Permite publicar a la web o compartir el fichero generado

Power BI
Website

Power BI
Desktop

Power BI
Mobile

COMPONENTES POWER BI

- Herramienta para clientes móviles
 - iOS, Android, Windows Phone
- Visualización de dashboards personalizados

Power BI
Website

Power BI
Desktop

Power BI
Mobile

plain concepts
COMPONENTES
POWER BI

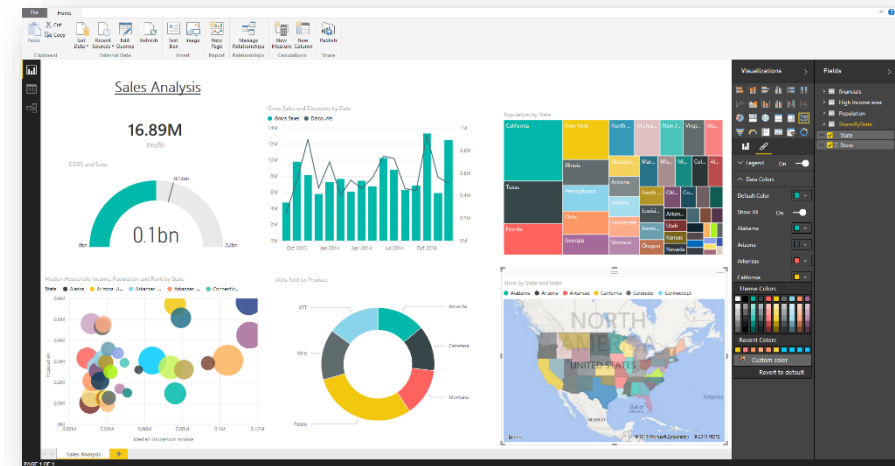


POWER BI DESKTOP

- Aplicación gratuita de escritorio para trabajar con los servicios de Power BI
- Combina las capacidades de Power Query, Power Pivot y Power View
- Especializada en visualización interactiva y análisis de datos
- Genera reportes interactivos y los publica online
- Actualizaciones mensuales

TRABAJANDO CON POWER BI DESKTOP

- Obtener y preparar los datos
- Establecer su estructura y las transformaciones necesarias
- Explorar los datos
- Generar reportes y visualizaciones
- Publicar los reports interactivos



OBTENCIÓN DE DATOS

- Mediante la funcionalidad de consultas podemos conectarnos a distintas fuentes de datos
- La navegación entre los datos es rápida gracias al cacheado en memoria
- Al editar las consultas antes de cargar los datos podemos filtrar solo lo necesario
- Si los datos se cargan desde una base datos, podemos encontrar tablas relacionadas de forma automática

OBTENCIÓN DE DATOS

File	Database	Azure	Other
<ul style="list-style-type: none">• Excel• CSV• XML• Text• Folder	<ul style="list-style-type: none">• SQL Server• Direct Query for SQL Server• Access• SQL Server Analysis Services• Oracle• IBM DB2• MySQL• PostgreSQL• Sybase• Teradata	<ul style="list-style-type: none">• SQL Database• Direct Query for SQL Database• SQL Data Warehouse• Marketplace• HDInsight• Blob Storage• Table Storage• HDInsight Spark• DocumentDB	<ul style="list-style-type: none">• Web• SharePoint List• Odata Feed• Hadoop File (HDFS)• Active Directory• Microsoft Exchange• Dynamics CRM Online• Facebook• Google Analytics• Salesforce Objects• Salesforce Reports• ODBC• appFigures• GitHub• QuickBooks Online• SweetIQ• Twilio• Zendesk• Spark• Blank Query• Mail Chimp

ESTRUCTURADO Y TRANSFORMACIÓN

- Aplicación de transformaciones a los datos a través de la UI
 - Eliminar tablas/filas/columnas
 - Cambio de tipo de datos
 - Eliminar relaciones
 - Corrección de errores
 - Fusión de consultas
- Muy útil para escenarios avanzados como mashups

ESTRUCTURADO Y TRANSFORMACIÓN

- Creación automática del modelo al importar los datos
 - Detección de relaciones, categorización y agrupación por defecto
- Refinado de modelos para cálculos complejos
 - Relaciones entre tablas, manuales o mediante la herramienta AutoDetect
- Definición de cálculos (medidas) para generar nuevos campos
 - Medidas generadas automáticamente, medidas personalizadas o creación de tablas mediante DAX
- Desarrollo de análisis avanzados utilizando medidas y relaciones

EXPLORACIÓN DE DATOS

Selección y
ordenación

Filtrado

Drill

Pivot y slice

Distintas
visualizaciones

Copia y pega
consultas de
Excel

GENERACIÓN DE REPORTE

Visualizaciones
interactivas

Creación de
relaciones con
Drag&Drop

Ajuste al
tamaño de
pantalla


Visualizaciones
personalizadas

Personalización
de colores,
formatos...

PUBLICACIÓN Y COMPARTICIÓN

- Los ficheros generados por Power BI Desktop
 - Se pueden compartir con otros usuarios
 - Pueden ser publicados online
- Los cambios en los dashboards se sincronizan automáticamente entre usuarios

LICENCIAMIENTO DE POWER BI

Euro (€) 	FREE POWER BI	€8.40 / user / month POWER BI PRO
	Sign up	Purchase
<u>Data capacity limit</u>	1 GB/user	10 GB/user ⓘ
Create, view and share your personal dashboards and reports with other Power BI users	•	•
Author content with the Power BI Desktop	•	•
Explore data with Natural Language ¹	•	•
Access your dashboards on mobile devices using native apps for iOS, Windows, and Android	•	•
Consume curated content packs for services like Dynamics, Salesforce, and Google Analytics	•	•
Import data and reports from Excel, CSV and Power BI Desktop files	•	•
Publish to web ²	•	•

LICENCIAMIENTO DE POWER BI

Data Refresh

Consume content that is scheduled to refresh	Daily	Up to 8 times per day
Consume streaming data in your dashboards and reports ⓘ	10K rows/hour	1M rows/hour ⓘ
Consume live data sources with full interactivity ⓘ		•
Access on-premises data using the Data Connectivity Gateways (Personal and Data Management)		•

Collaboration

Collaborate with your team using Office 365 Groups in Power BI		•
Create, publish and view organizational content packs		•
Manage access control and sharing through Active Directory groups		•
Shared data queries through the Data Catalog		•

plain concepts

POWER BI



¿PREGUNTAS?

plain concepts

EJERCICIO PRÁCTICO





GRACIAS

Barcelona



Bilbao



Madrid



Sevilla



Dubai



London



Seattle