SIC

Serviços e Infraestruturas de Computação

Influx DB and TICK Stack

Telegraph, InfluxDB, Chronograf, Kapacitor

Outline

- Time Series Data
- The Tick Stack framework
- Usage Scenarios

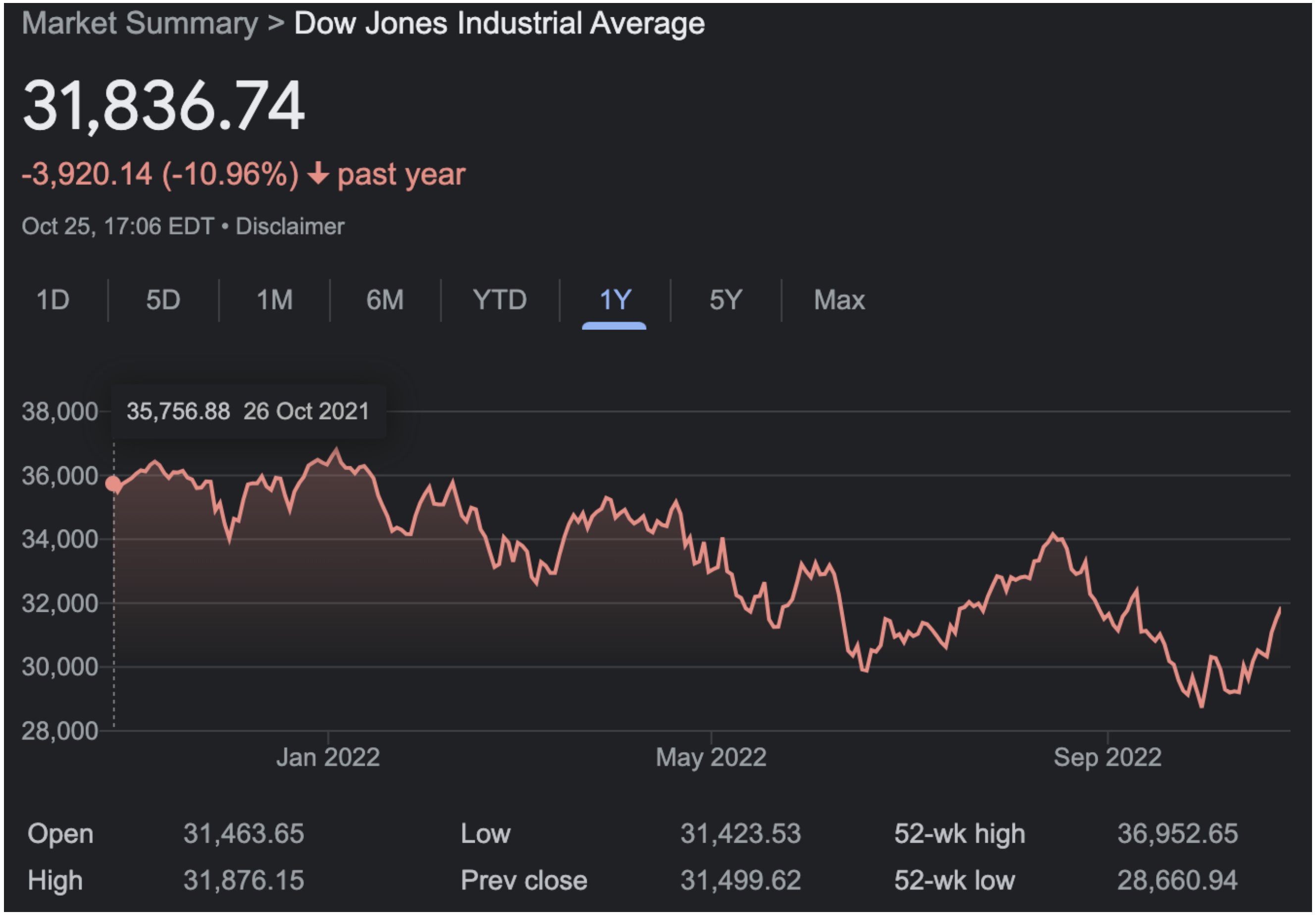
Time Series Data

A series of data points indexed (or listed, or graphed) in chronological order

- Usually, time series consist of data points taken at regular intervals (e.g., temperature reading every hour; instant speed measured every second; plane altitude every second), which are typically designated at discrete time series.
- Typical properties of time series data:
  - Large amounts of data points (e.g., a single sensor producing one reading per second generates 604,800 data points per day).
  - Need for high read and write throughput.
  - Often, the value of data depends on its age (e.g., we keep detailed temperature readings from the last days, but only daily or weekly averages from last year). This leads to regular deletion of large volumes of data, due to data expiration.
  - Mostly regular insert/append operations.

Examples of Time Series Data

Stock Market tickers



Market Summary > Dow Jones Industrial Average

31,836.74

-3,920.14 (-10.96%) ↓ past year

Oct 25, 17:06 EDT • Disclaimer

Open	31,463.65	Low	31,423.53	52-wk high	36,952.65
High	31,876.15	Prev close	31,499.62	52-wk low	28,660.94

Examples of Time Series Data

Daily Sales

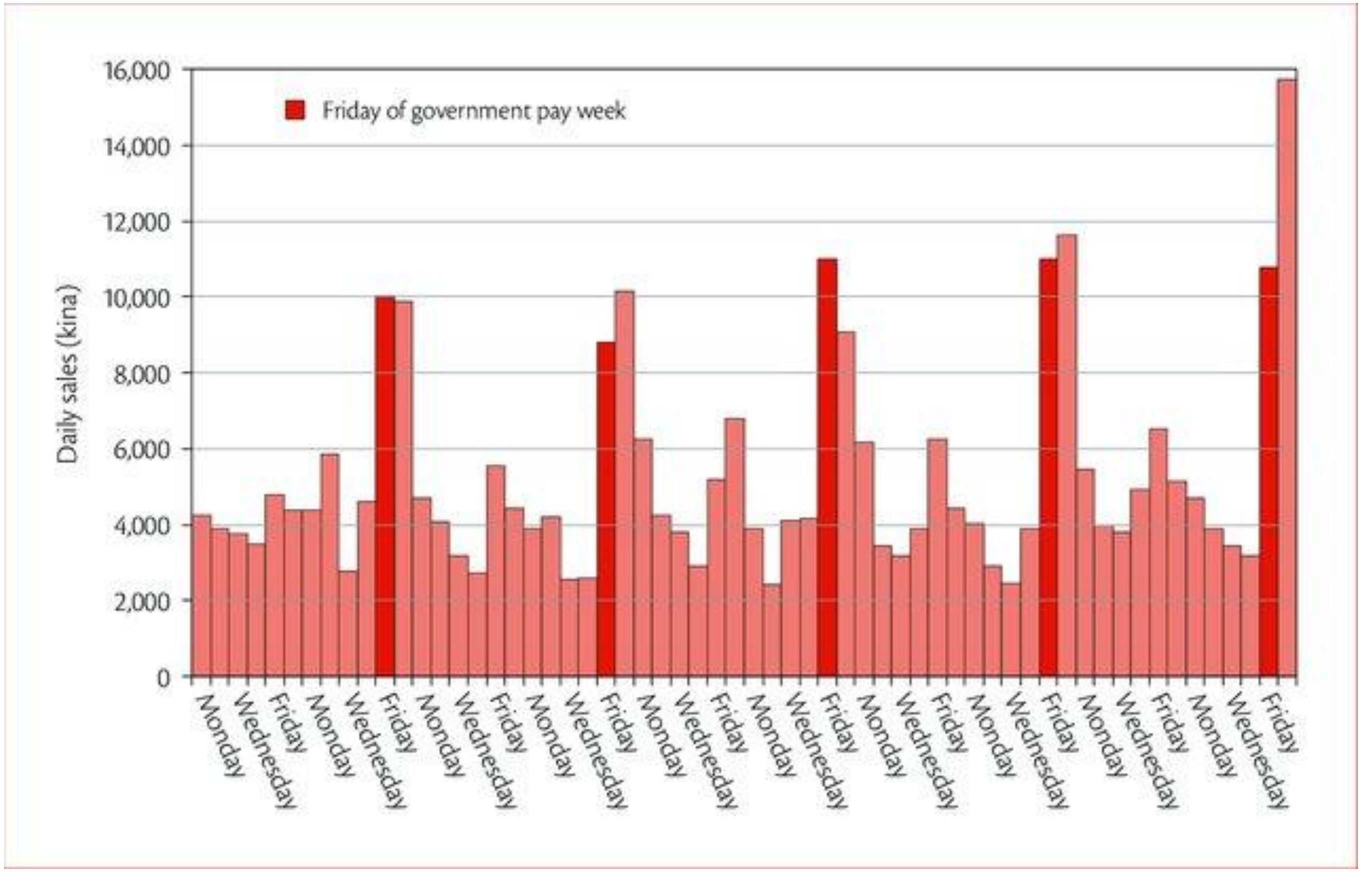


Image Source: https://www.researchgate.net/publication/327247683_The_Broader_Economy/figures?lo=1

Why specific tools for time series data?

- Traditional tools are not specifically tailored to handle this type of data, with very frequent insert/append operations and regular (and likely automated) deletion of data due to data expiration.
- Streamlining the process of collecting and storing this data would be nice.
- It would also be nice to have specific tools for visualizing such data, for processing this data.
- It would also be nice to have ways of making it easier (and faster) to perform queries on data time series data.

The TICK Stack

A collection of associated technologies combined to deliver a platform for storing, capturing, monitoring, and visualizing data that is in time series. The TICK stack consists of the following technologies:

- Telegraf – collection of tie sequential data from a range of sources including IoT devices.
- InfluxDB – high performance and efficient database store for handling high volumes of time-series data.
- Chronograf – real-time visualization of InfluxDB data.
- Kapacitor – monitoring and alerting based on views of InfluxDB data and anomalies contained within those views.

The TICK Stack

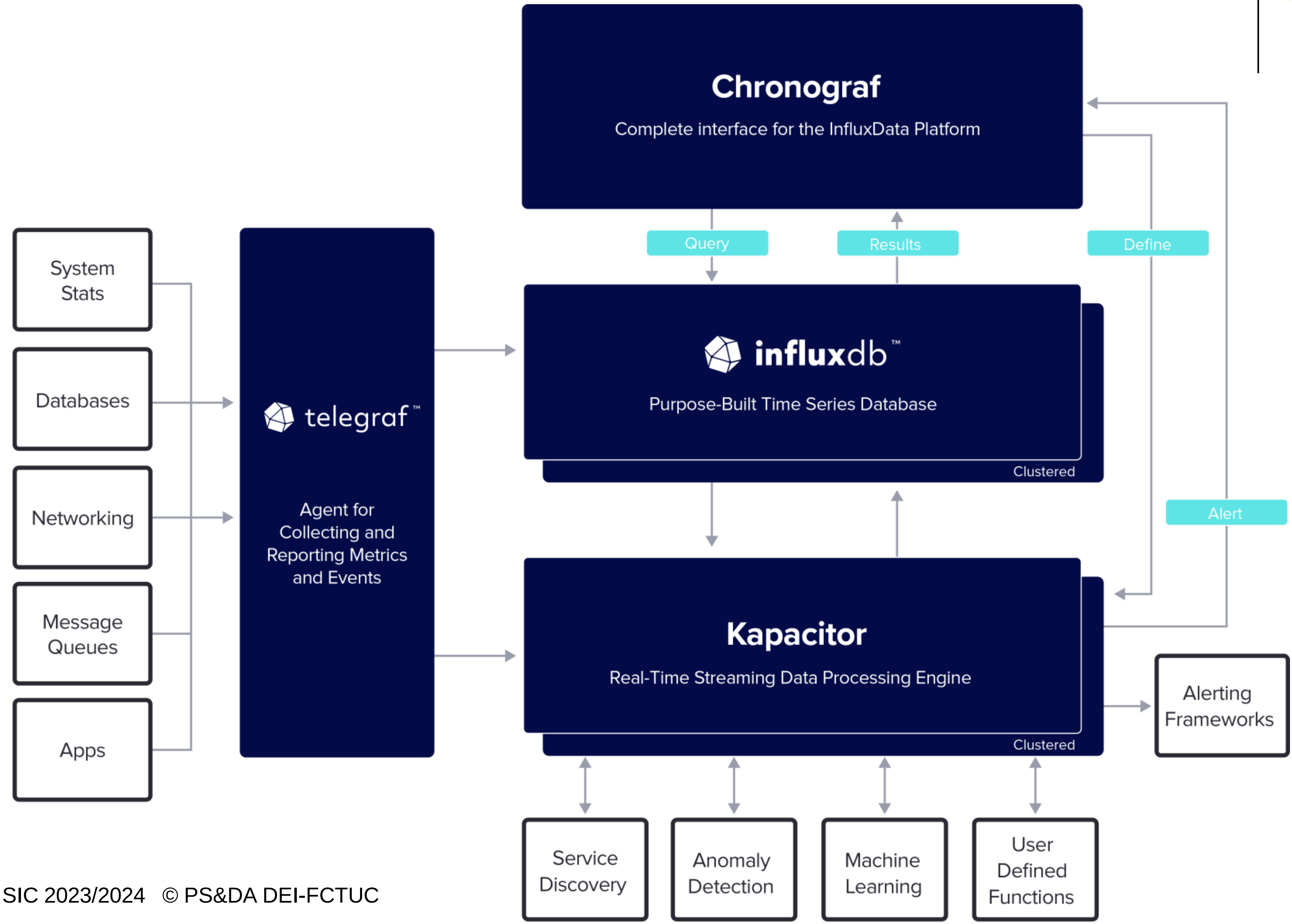
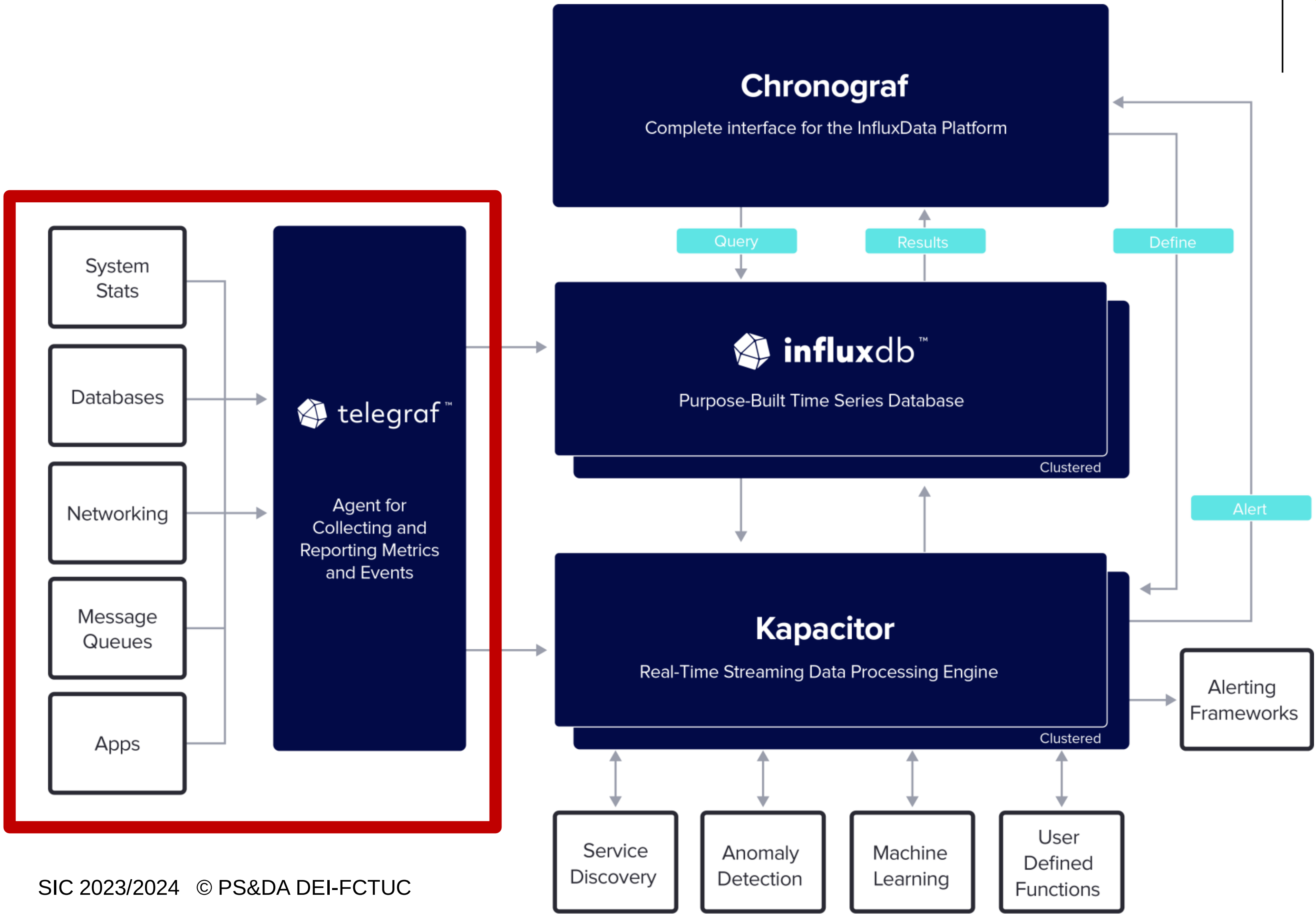


Image source: https://www.influxdata.com

Telegraph


Chronograf
Complete interface for the InfluxData Platform
Query
Results
Define
System Stats
Databases
Networking
Message Queues
Apps
telegraf™
Agent for Collecting and Reporting Metrics and Events
influxdb™
Purpose-Built Time Series Database
Clustered
Alert
Kapacitor
Real-Time Streaming Data Processing Engine
Clustered
Alerting Frameworks
Service Discovery
Anomaly Detection
Machine Learning
User Defined Functions

Telegraph

Agent for collecting metrics and reporting metrics and events

- Set of agent plug-ins for collecting and reporting metrics
- There are already hundreds of plug-ins available for popular services and applications
  - https://docs.influxdata.com/telegraf/v1.24/plugins
- Typically, agents have a minimal footprint
- New inputs and outputs can easily be added
- Can work with any external scripts

Telegraph

Some examples of application domains where Telegraph can be used:

IoT Sensors:

- Collect critical stateful data (pressure levels, temp levels, etc.) with popular protocols like MQTT, ModBus, OPC-UA, and Kafka.

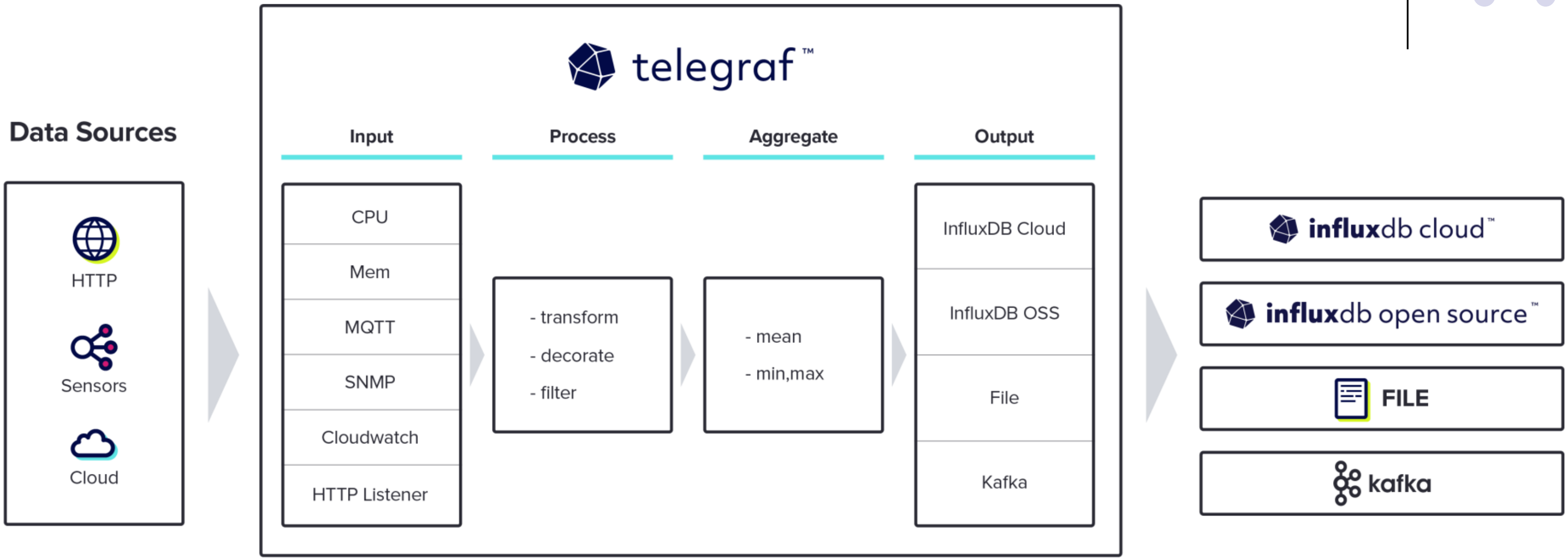
System Telemetry:

- Metrics from system tools (such as iptables, Netstat, NGINX, SNMP, and HAProxy) to monitor IT infrastructure and services.

Healthcare:

- Time series data from medical sensors (fitness sensors, ECG, glucose levels, etc.).

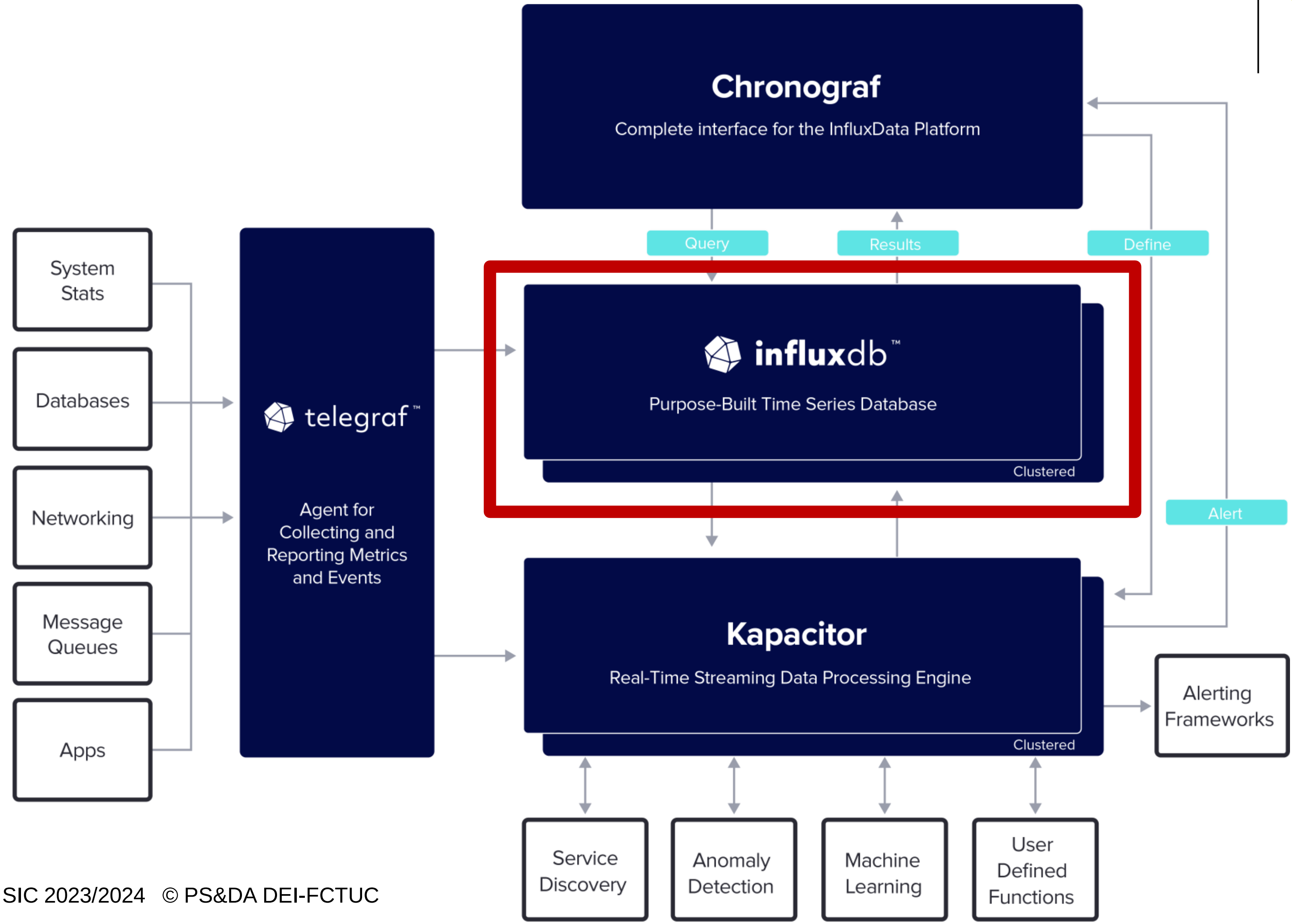
Telegraph overview



- Input plug-ins: collect metrics from the system, services, and 3rd parties.
- Process plug-ins: transform, decorate, and filter metrics (sanitize)
- Aggregate plug-ins: create aggregate metrics (e.g., average).
- Output plug-ins: write to datastores, services, and message queues, such as InfluxDB, Kafka, MQTT, and others.

Image source: https://www.influxdata.com/time-series-platform/telegraf

InfluxDB


Chronograf
Complete interface for the InfluxData Platform
Query
Results
Define
System Stats
Databases
Networking
Message Queues
Apps
telegraf™
Agent for Collecting and Reporting Metrics and Events
influxdb™
Purpose-Built Time Series Database
Clustered
Alert
Kapacitor
Real-Time Streaming Data Processing Engine
Clustered
Alerting Frameworks
Service Discovery
Anomaly Detection
Machine Learning
User Defined Functions

InfluxDB

Scalable storage for time series (metrics, events, real-time analytics)

- Specifically designed for high-performance datastore of time series
- Fast and simple HTTP/HTTPS APIs for write and query operations
- Plugins support for other data sources
- Retention policies of automatic expiration of (older) data
- SQL-like query language, to enable queries of aggregated data
- Tags for indexation of time series, for fast and efficient queries
- Possibility of creating *continuous queries*, which automatically compute aggregate data to improve the efficiency of frequent queries.

InfluxDB

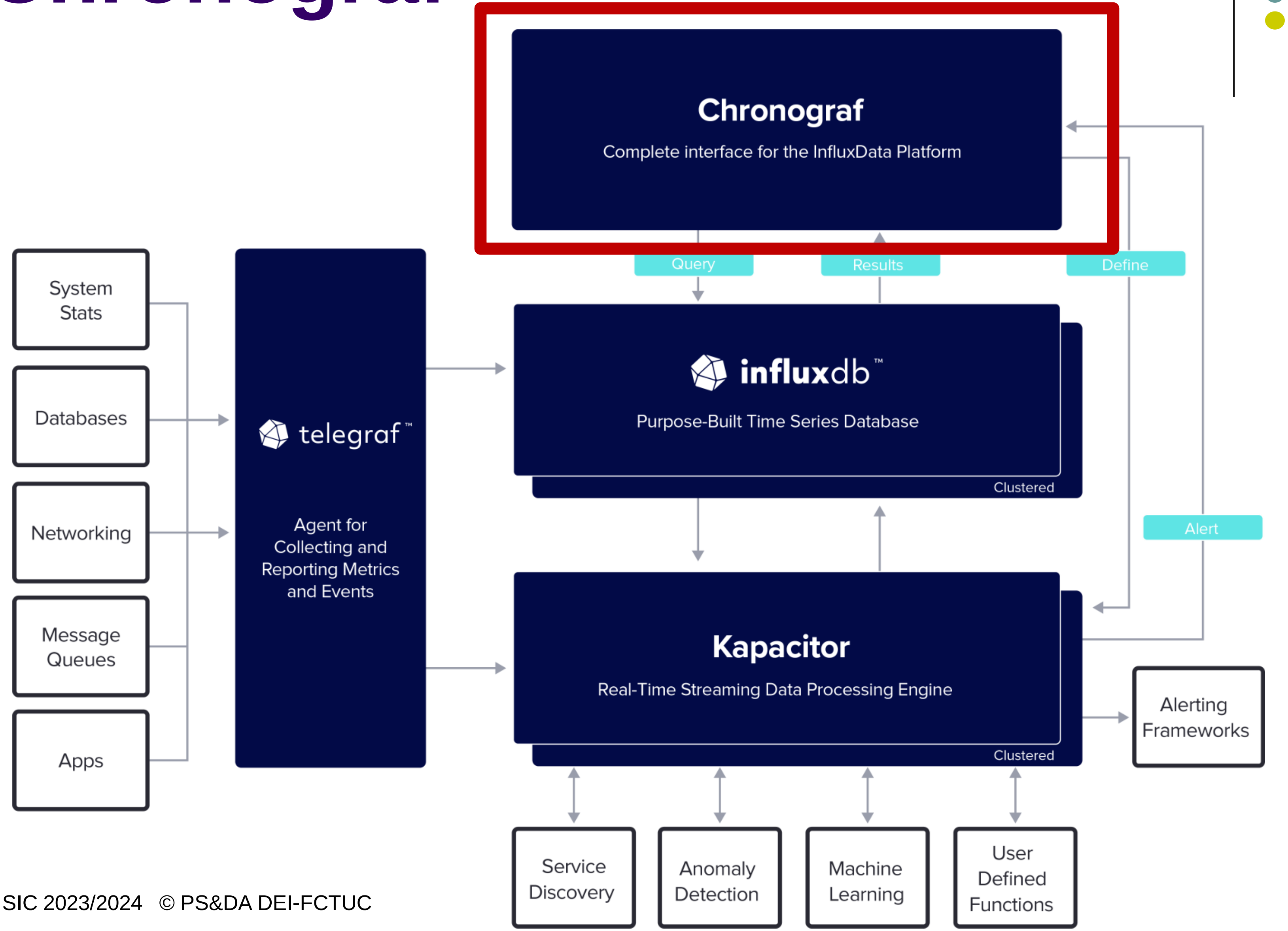
Examples of queries

```
SELECT <field_key>[,<field_key>,<tag_key>] FROM <measurement_name>[,<measurement_name>]
```

```
> SELECT "level description","location","water_level" FROM "h2o_feet"
name: h2o_feet
--------------
time                    level description     location       water_level
2015-08-18T00:00:00Z    below 3 feet          santa_monica   2.064
2015-08-18T00:00:00Z    between 6 and 9 feet coyote_creek    8.12
[...]
2015-09-18T21:36:00Z    between 3 and 6 feet santa_monica    5.066
2015-09-18T21:42:00Z    between 3 and 6 feet santa_monica    4.938
```

More examples at: https://docs.influxdata.com/influxdb/v1.8/query_language/explore-data/

Chronograf


Chronograf
Complete interface for the InfluxData Platform
Query
Results
Define
System Stats
Databases
Networking
Message Queues
Apps
telegraf™
Agent for Collecting and Reporting Metrics and Events
influxdb™
Purpose-Built Time Series Database
Clustered
Alert
Kapacitor
Real-Time Streaming Data Processing Engine
Clustered
Alerting Frameworks
Service Discovery
Anomaly Detection
Machine Learning
User Defined Functions

Chronograf

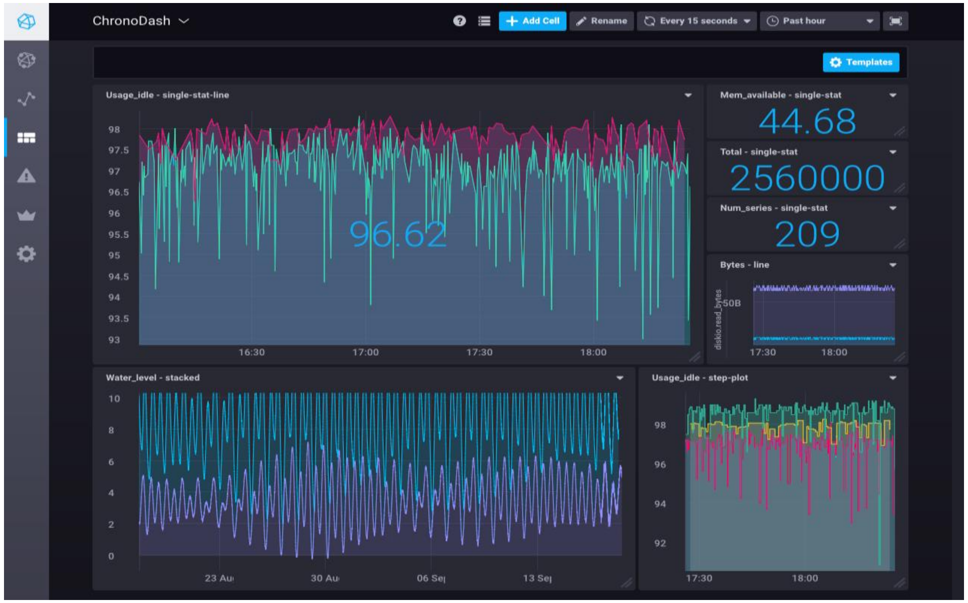
Web interface for monitoring, visualization, and management

- Data visualization
- Database management
- Infrastructure monitoring/overview
- Alert management

Chronograf

Web interface for monitoring, visualization, and management

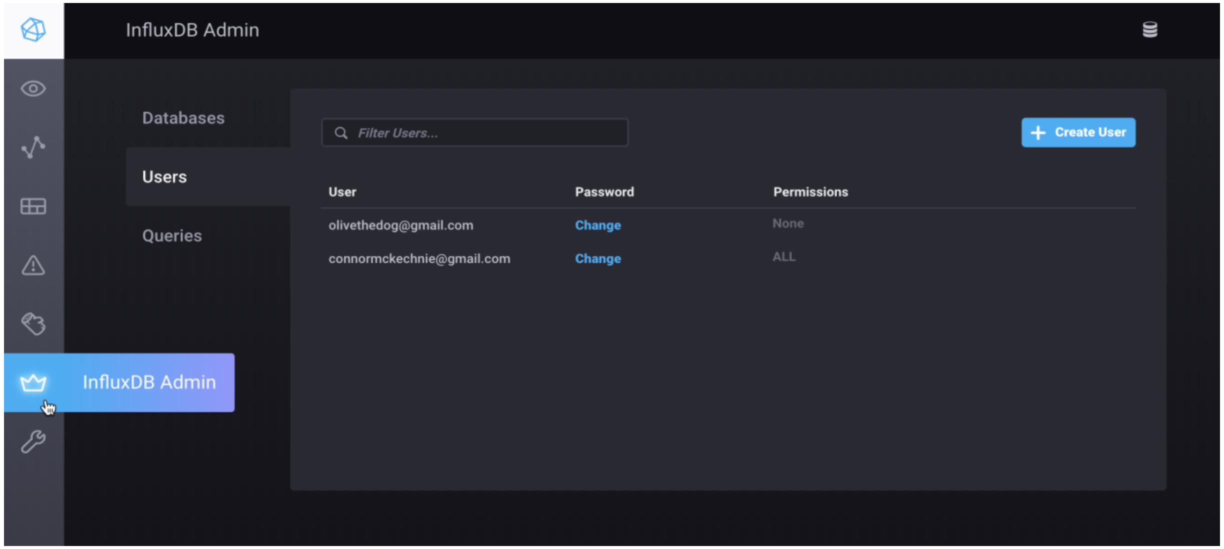
- Data visualization
  - Monitor application data with (pre-created) dashboards
  - Create customized dashboards complete with various graph types
  - Investigate data with data explorer and query templates



Chronograf

Web interface for monitoring, visualization, and management

- Database management
  - Create and delete databases and retention policies
  - View and manage currently-running queries
  - Create, delete, and assign permissions to users



User	Password	Permissions
olivethedog@gmail.com	Change	None
connormckechnie@gmail.com	Change	ALL

Chronograf

Web interface for monitoring, visualization, and management

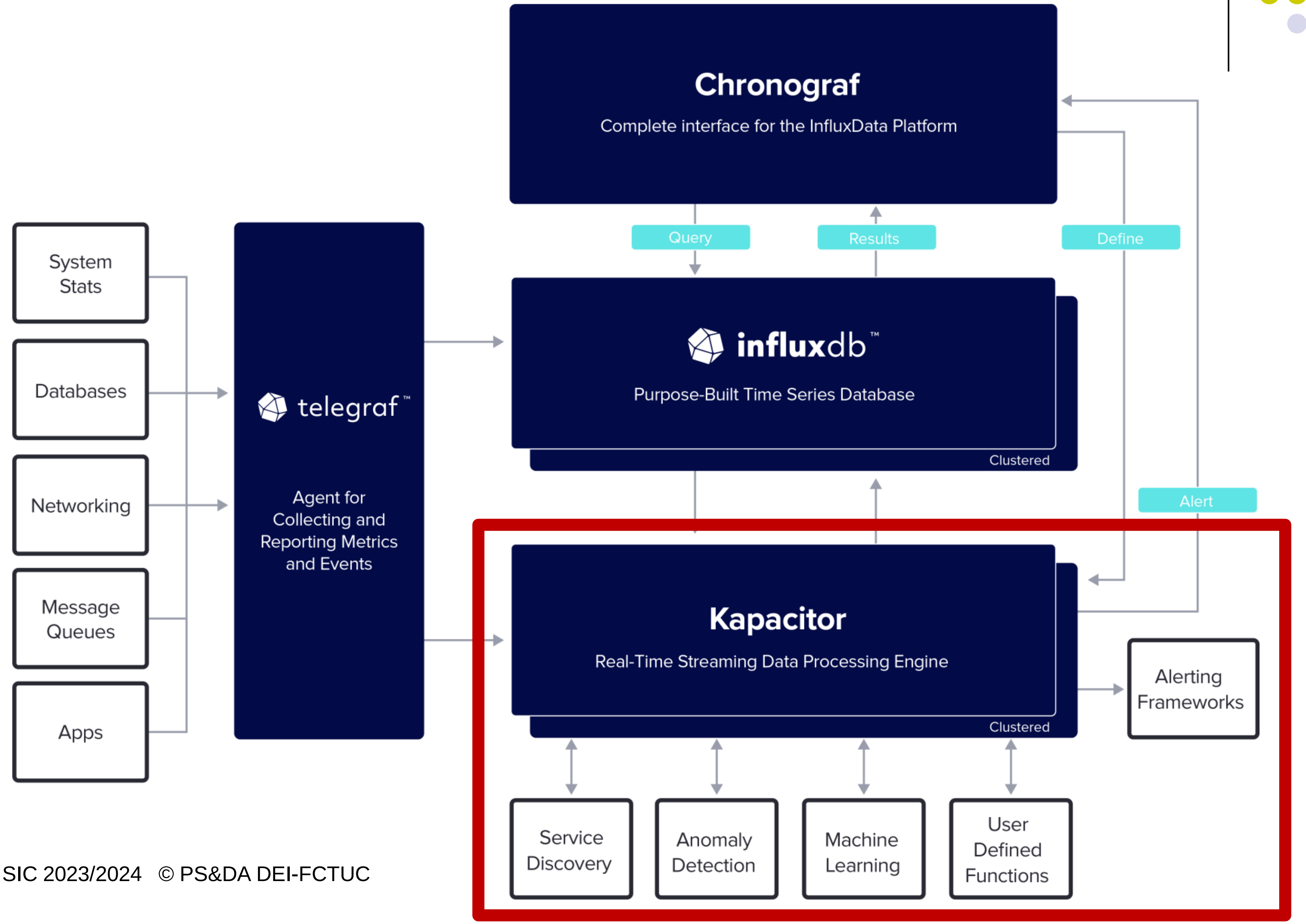
- Infrastructure monitoring/overview
  - View hosts and status in the infrastructure
  - View the configured applications on each host
  - Monitor applications with dashboards

Chronograf

Web interface for monitoring, visualization, and management

- Alert management
  - Generate threshold, relative, and deadman alerts on time series data
  - Enable/disable alert rules; view active alerts on a dashboard
  - Send alerts to event handlers such as Slack

Kapacitor


Chronograf
Complete interface for the InfluxData Platform
Query
Results
Define
System Stats
Databases
Networking
Message Queues
Apps
telegraf™
Agent for Collecting and Reporting Metrics and Events
influxdb™
Purpose-Built Time Series Database
Clustered
Alert
Kapacitor
Real-Time Streaming Data Processing Engine
Clustered
Alerting Frameworks
Service Discovery
Anomaly Detection
Machine Learning
User Defined Functions

Kapacitor

Processing, monitoring, and alert generation based on time-series data

- Process both streaming and batch data *(familiar with these concepts?)*
- Data inputs:
  - Regular queries to the InfluxDB, or receive data from line protocol
- Store transformed data in InfluxDB
- Support customer-defined functions to detect anomalies
- Allows creating data processing pipelines, using the Tickscript DSL
- Integrates with external tools (Slack, HipChat, Sensu…)

Kapacitor

Processing, monitoring, and alert generation based on time-series data

- Action-oriented
  - Kapacitor’s alerting system follows a publish-subscribe design pattern
  - Alerts are published to topics and handlers subscribe to a topic
- Streaming analytics
  - Allows the scripting to be done using lambda expressions to define transformations on data points as well as define boolean conditions that act as the filter
- Anomaly detection
  - Kapacitor provides a simple plugin architecture that allows it to integrate with any anomaly detection engine
    - ML libraries, pattern matching engines, rules engines

Kapacitor

Example of TICKscript DSL

```
stream |eval(lambda: "error_count" / "total_count") .as('error_percent’)
```

(adds a new field “error_percent” to each data point with the result of “error_count” / “total_count”) where error_count and total_count are existing fields on the data point

Use Cases

- Infrastructure Monitoring
- Anomaly detection for cybersecurity purposes
- Anomaly detection for fault management
- Handling of IoT data
- …

Further Reading

- InfluxDB documentation: https://docs.influxdata.com