

TRABAJANDO CON GRAFANA

BIG DATA – BUSINESS INTELLIGENCE – MACHINE LEARNING



Grafana



Índice

1. INTRODUCCIÓN:	2
2. ORIGEN DE DATOS:.....	2
3. PROCESO DE CARGA DE DATOS:.....	3
3.1. SCRIPT SQL:.....	4
3.2. SCRIPT CARGA DE DATOS DESDE ORIGEN DE DATOS EN POSTGRESQL:.....	4
4. DESPLIEGUE Y CONFIGURACIÓN DE GRAFANA:.....	5
5. CONFIGURACIÓN DE LA FUENTE DE DATOS:	5
6. EXPLOTACIÓN DE DATOS:.....	7
7. UTILIDADES DE GRAFANA:.....	12
7.1. SISTEMA DE ALERTAS.....	12
7.2. GESTIÓN DE PERMISOS.....	16
7.3. VISUALIZACIÓN REAL-TIME.....	17
7.4. REPOSITORIO DE <i>PLUGINS</i>	18
1. SOBRE STRATEBI.....	19
TECNOLOGÍAS.....	20
2. DESCRIPCIÓN DE LAS SOLUCIONES.....	21
3. INFORMACIÓN SOBRE STRATEBI.....	22
4. EJEMPLOS DE DESARROLLOS ANALYTICS.....	23

1. Introducción:

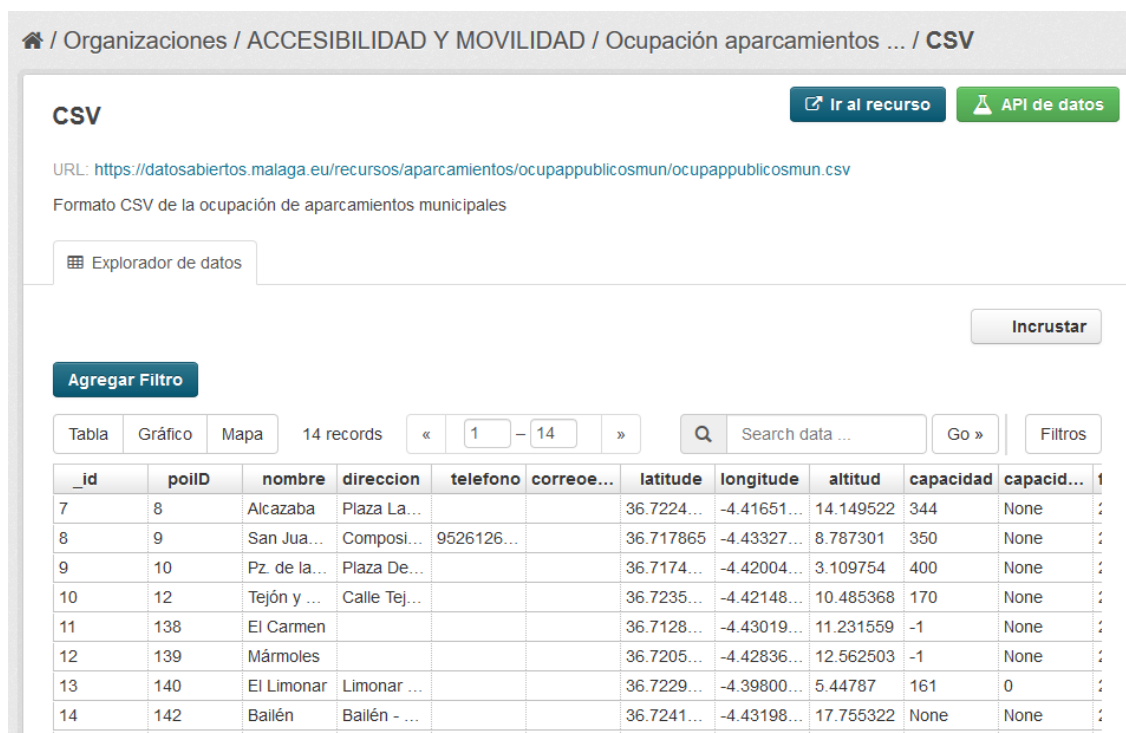
El propósito de este documento es recoger el proceso de creación de un cuadro de mandos que monitorice la situación de los parkings públicos de Málaga en tiempo real utilizando la herramienta Grafana.

Grafana es una herramienta de software libre que permite crear cuadros de mando y gráficas a partir de múltiples fuentes de datos. Suele ser utilizado para la visualización y monitorización de datos en tiempo real.

2. Origen de datos:

En este ejemplo práctico el origen de datos será el portal de datos abiertos del Ayuntamiento de Málaga (<https://datosabiertos.malaga.eu/>), concretamente el conjunto de datos sobre la ocupación de los aparcamientos públicos municipales. Esta información se encuentra en formato CSV y se actualiza cada minuto.

<https://datosabiertos.malaga.eu/dataset/ocupacion-aparcamientos-publicos-municipales>



/ Organizaciones / ACCESIBILIDAD Y MOVILIDAD / Ocupación aparcamientos ... / CSV

CSV [Ir al recurso](#) [API de datos](#)

URL: <https://datosabiertos.malaga.eu/recursos/aparcamientos/ocupappublicosmun/ocupappublicosmun.csv>

Formato CSV de la ocupación de aparcamientos municipales

Explorador de datos Incrustar

Agregar Filtro

Tabla Gráfico Mapa 14 records « 1 - 14 » Search data ... Go » Filtros

_id	poiID	nombre	direccion	telefono	correoe...	latitud	longitud	altitud	capacidad	capacid...	f
7	8	Alcazaba	Plaza La...			36.7224...	-4.41651...	14.149522	344	None	⌵
8	9	San Jua...	Composi...	9526126...		36.717865	-4.43327...	8.787301	350	None	⌵
9	10	Pz. de la...	Plaza De...			36.7174...	-4.42004...	3.109754	400	None	⌵
10	12	Tejón y ...	Calle Tej...			36.7235...	-4.42148...	10.485368	170	None	⌵
11	138	El Carmen				36.7128...	-4.43019...	11.231559	-1	None	⌵
12	139	Mármoles				36.7205...	-4.42836...	12.562503	-1	None	⌵
13	140	El Limonar	Limonar ...			36.7229...	-4.39800...	5.44787	161	0	⌵
14	142	Bailén	Bailén - ...			36.7241...	-4.43198...	17.755322	None	None	⌵

Los datos que incluye este conjunto de datos, entre otros, son los siguientes:

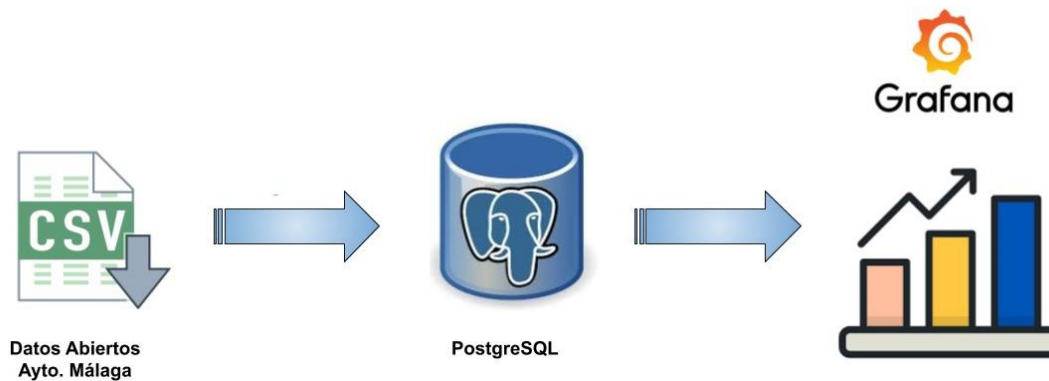
- Identificador numérico del aparcamiento.
- Nombre del parking.
- Dirección del parking.
- Geoposicionamiento con latitud y altitud.
- Capacidad del parking en número de plazas.
- Número de plazas libres.
- Nivel de ocupación.
- Fecha de la consulta.

3. Proceso de carga de datos:

Grafana no soporta directamente la carga de los datos desde la *api* que ofrece el portal de datos abiertos del ayuntamiento de Málaga.

Para poder consumir los datos en Grafana se ha implementado un proceso, mediante un script en *bash*, que carga los datos desde el portal de datos abiertos del ayuntamiento de Málaga a una base de datos en *PostgreSQL*. Este script se ejecuta periódicamente con el propósito de tener un histórico de datos la base de datos *PostgreSQL*.

Posteriormente, desde Grafana, se establecerá una conexión a esta base de datos *PostgreSQL* y se explotarán los datos mediante las herramientas de análisis y visualización que ofrece el propio Grafana.



3.1. Script SQL:

```
DROP TABLE IF EXISTS ocupacion;
CREATE TABLE ocupacion (
    poiID INTEGER,
    nombre VARCHAR,
    direccion VARCHAR,
    telefono VARCHAR,
    correoelectronico VARCHAR,
    latitute FLOAT,
    longitude FLOAT,
    altitud FLOAT,
    capacidad INTEGER,
    capacidad_discapitados INTEGER,
    fechahora_ultima_actualizacion timestamp,
    libres INTEGER,
    libres_discapitados INTEGER,
    nivelocupacion_naranja INTEGER,
    nivelocupacion_rojo INTEGER,
    smassa_sector_sare VARCHAR
);
```

3.2. Script carga de datos desde origen de datos en PostgreSQL:

```
1  #!/bin/sh
2
3  set -eu
4
5  CSV_URL='https://datosabiertos.malaga.eu/recursos/aparcamientos/ocupappublicosmun/ocupappublicosmun.csv'
6  CSV_TMP=$(mktemp -u); trap 'rm -f "${CSV_TMP}"' EXIT
7
8  export PGPASSWORD=${POSTGRES_PASSWORD}
9  PSQL="psql --username=${POSTGRES_USER} --host=postgres --port=5432 ${POSTGRES_DB}"
10
11 # Download and transform CSV
12 curl -fsSL "${CSV_URL}" | sed 's|"None"|"0"|g' > "${CSV_TMP}"
13
14 # Import CSV
15 ${PSQL} <<-EOF
16     \copy ocupacion FROM '${CSV_TMP}' DELIMITER ',' CSV HEADER;
17 EOF
18
19 # Cleanup
20 ${PSQL} <<-EOF
21     DELETE FROM ocupacion WHERE fechahora_ultima_actualizacion < (now() - interval '72 hour');
22 EOF
```

En este ejemplo práctico, el script de carga de datos se ejecuta cada 2 minutos mediante crontab y, para evitar un alto volumen de datos en la base de datos, solamente se almacenan datos de las últimas 72 horas, eliminando aquellos que no cumplan esta condición.

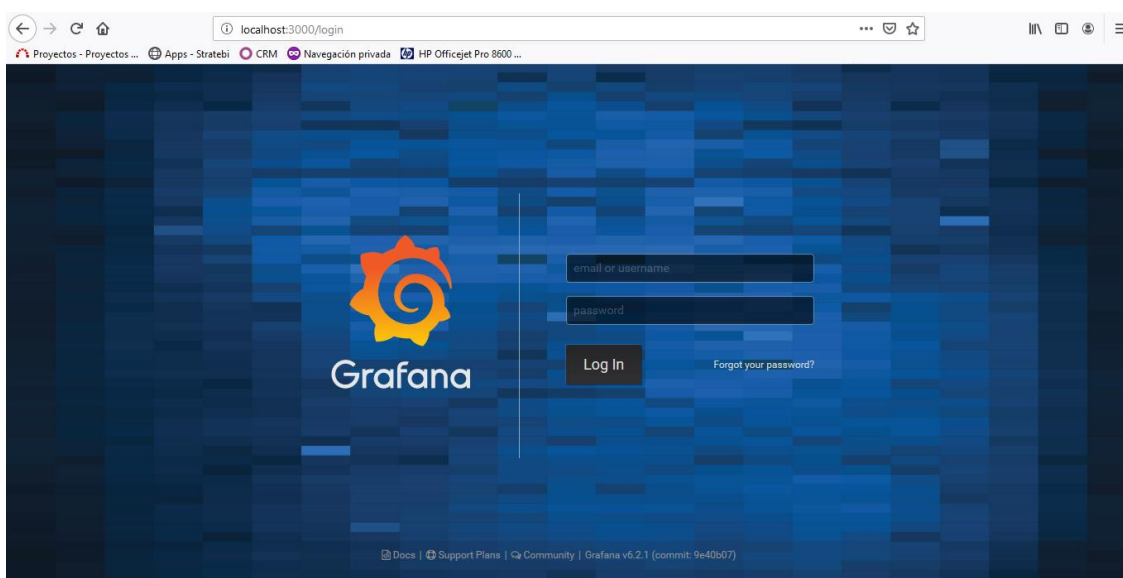
Más información: <http://gitlab.stratebi.com/hector.molinero/demo-grafana/>

4. Despliegue y configuración de Grafana:

Esta documentación se basa en la versión *community* de *Grafana*. Esta herramienta soporta la mayoría de los sistemas operativos actuales: *Linux*, *Windows* y *MAC*.

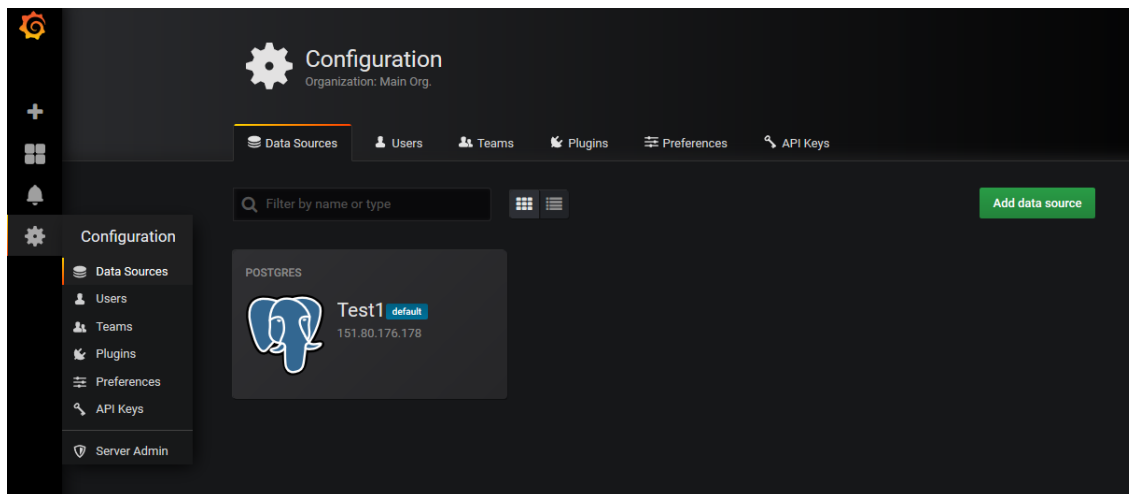
En este enlace se puede encontrar más información de las posibilidades de descarga que brinda Grafana: <https://grafana.com/grafana/download>

Para instalarlo se debe comprimir el archivo comprimido descargado y ejecutar el fichero *grafana-server* dentro de la carpeta */bin/*. Una vez ejecutado, el acceso se realiza a través de <http://localhost:3000>.

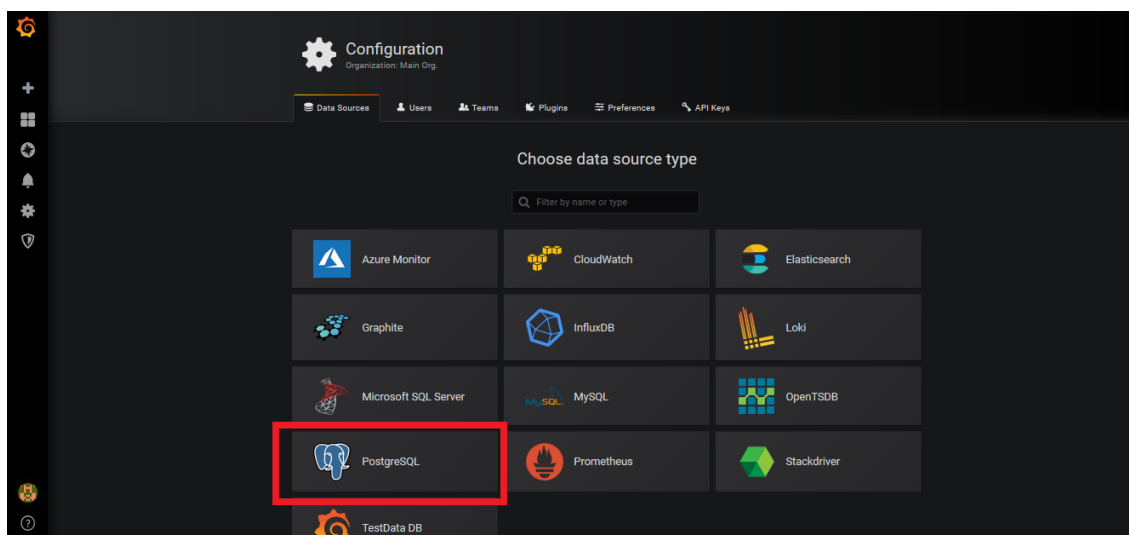


5. Configuración de la fuente de datos:

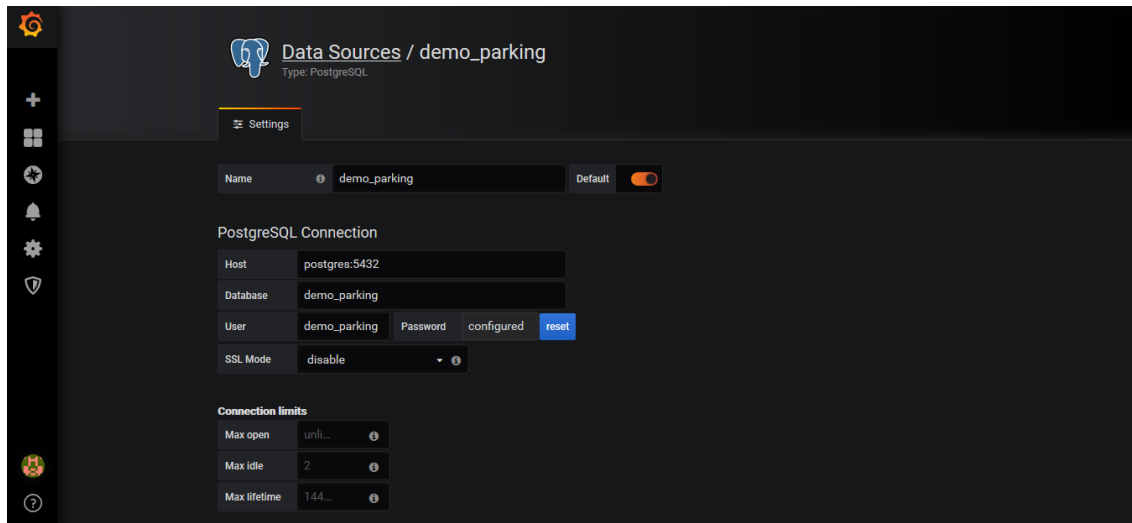
Una vez se accede a Grafana (<https://grafana.demo.stratebi.com>), el primer paso será configurar una nueva fuente de datos. Para ello, se debe desplegar el menú izquierdo y dentro del submenú *Configuration*, seleccionar la opción de *Datasources*.



Una vez dentro, se debe pulsar el botón de *Add data source* y se abrirá una interfaz para seleccionar el tipo de fuente de datos. Se debe elegir *PostgreSQL*.



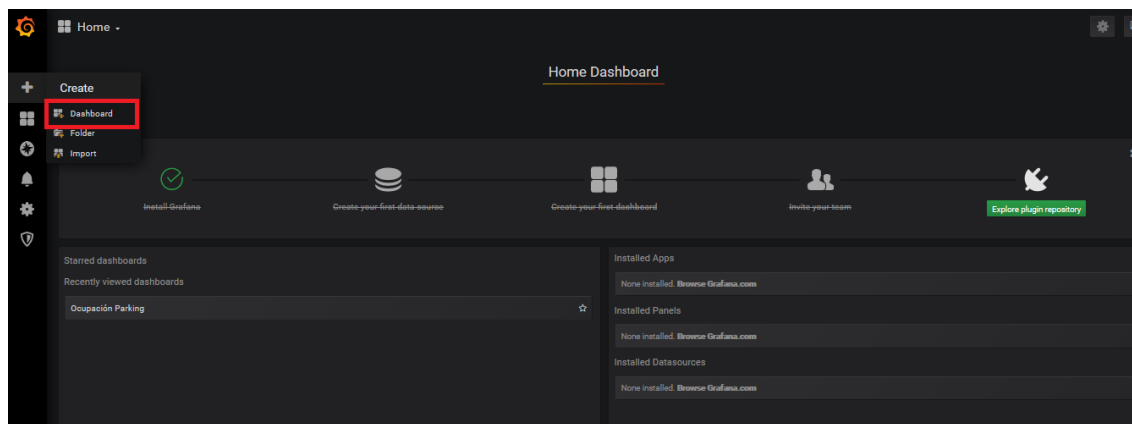
El último paso será configurar los parámetros de conexión a la base de datos.



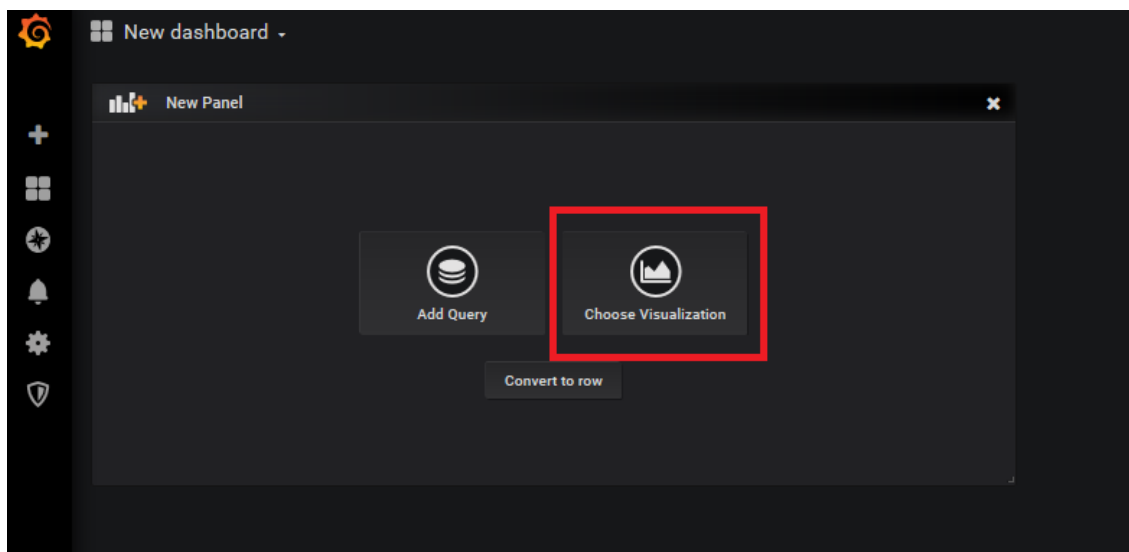
6. Explotación de datos:

El objetivo de este ejercicio es monitorizar la ocupación de cuatro aparcamientos públicos de Málaga: parking Salitre, parking Cervantes, parking Avenida Andalucía y parking Plaza de la Marina. Para ello, se implementará un cuadro de mandos que monitorice toda esta actividad en un determinado periodo de tiempo.

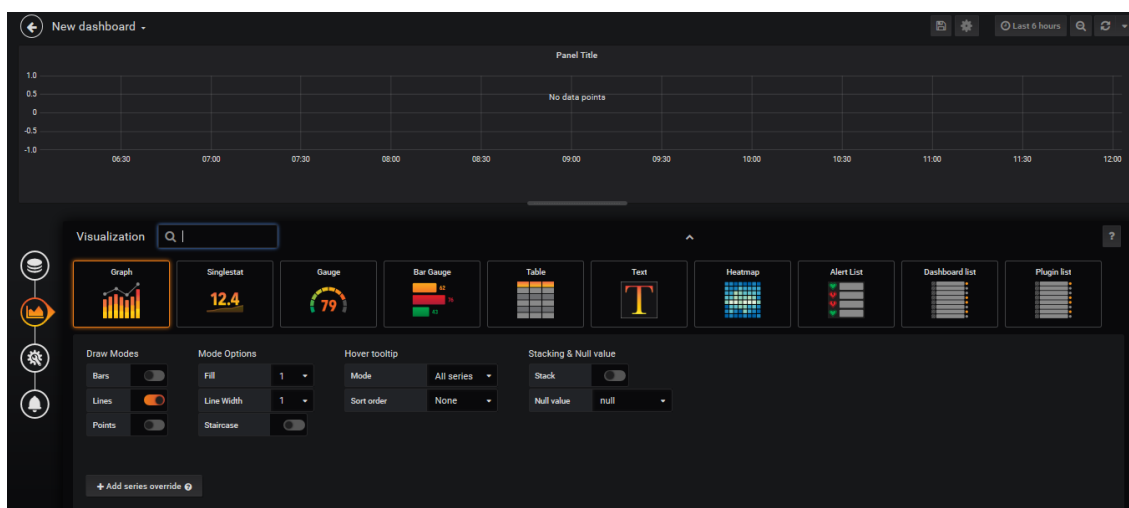
Para la explotación de los datos en un cuadro de mandos (*dashboard*) se debe desplegar nuevamente el menú izquierdo, pulsar el botón (+) y seleccionar la opción de *Create Dashboard*.



El proceso de creación de un cuadro de mandos con Grafana es muy sencillo. En primer lugar, se debe seleccionar la opción *Choose Visualización* para elegir el elemento visual que se desea incluir en el cuadro de mandos.



En este caso de ejemplo se debe seleccionar la opción *Graph*. Con este tipo de gráfico se busca pintar la evolución de la ocupación de los aparcamientos en los parkings del ayuntamiento de Málaga en un periodo de tiempo. Una vez seleccionada la opción se pueden configurar distintos parámetros de visualización del gráfico.

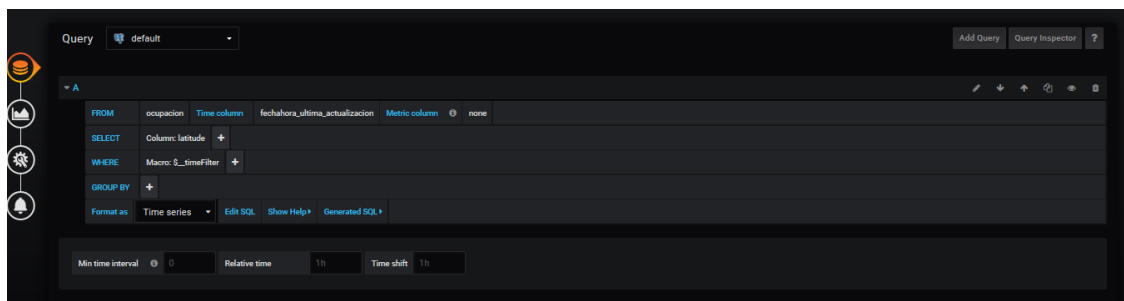


Tras hacer click sobre el tipo de gráfico, se aprecia un nuevo submenú en la parte inferior izquierda que incluye las siguientes opciones:

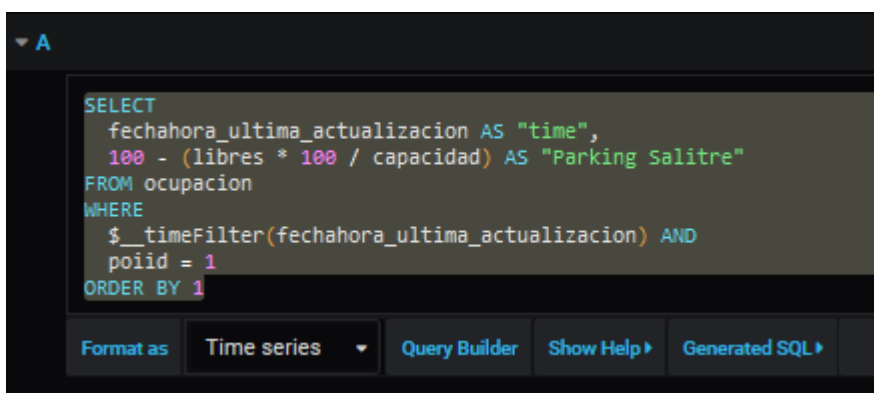


- *Queries*: Aquí se implementan las consultas que se mostrarán en los elementos de visualización.
- *Visualization*: Incluye opciones de configuración a nivel de visualización del elemento seleccionado.
- *General*: Configuración del elemento de visualización. Permite definir nombre y descripción del elemento.
- *Alert*: Permite la creación de alertas. Más información en capítulo 7.1.

El siguiente paso será definir las consultas para ese gráfico. Para ello se debe pulsar el icono de las *Queries* que permite la creación de consultas mediante un sencillo asistente sobre el que se pueden elegir de múltiples opciones en la consulta.

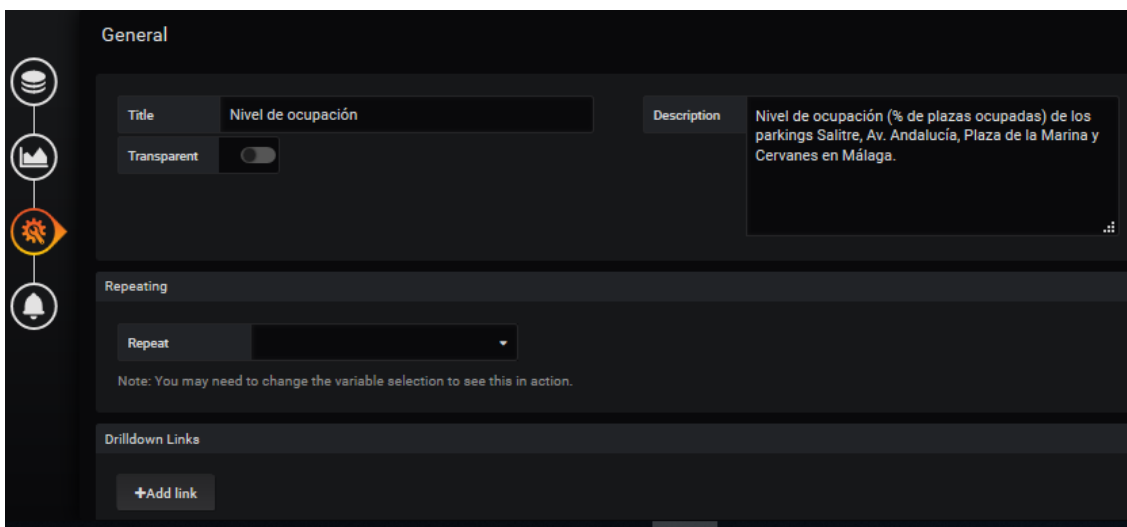


Este asistente también permite introducir manualmente consultas SQL a través de la opción *Edit SQL*. En la construcción de este gráfico se elegirá esta opción y se debe introducir la siguiente consulta para obtener el % de ocupación del parking Salitre (*pooid* = 1):

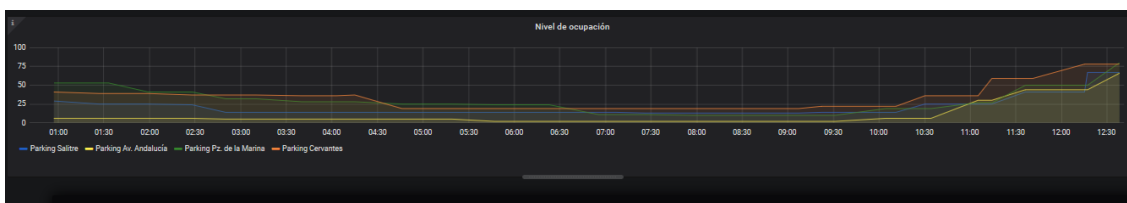


Este proceso se debe repetir, pulsando el botón de *Add Query*, para los parkings de Av. Andalucía (*pooid* = 4), Cervantes (*pooid* = 2) y Plaza de la Marina (*pooid* = 10).

Antes de terminar, pulsando el icono *General*, se debe introducir un nombre y una descripción del gráfico que se acaba de crear.



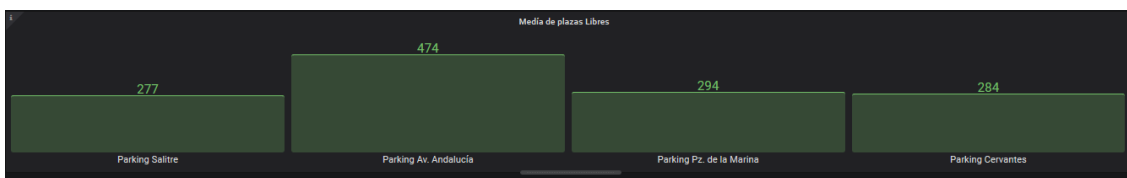
Si todo ha ido correctamente se obtiene una gráfica similar a la que aparece a continuación:



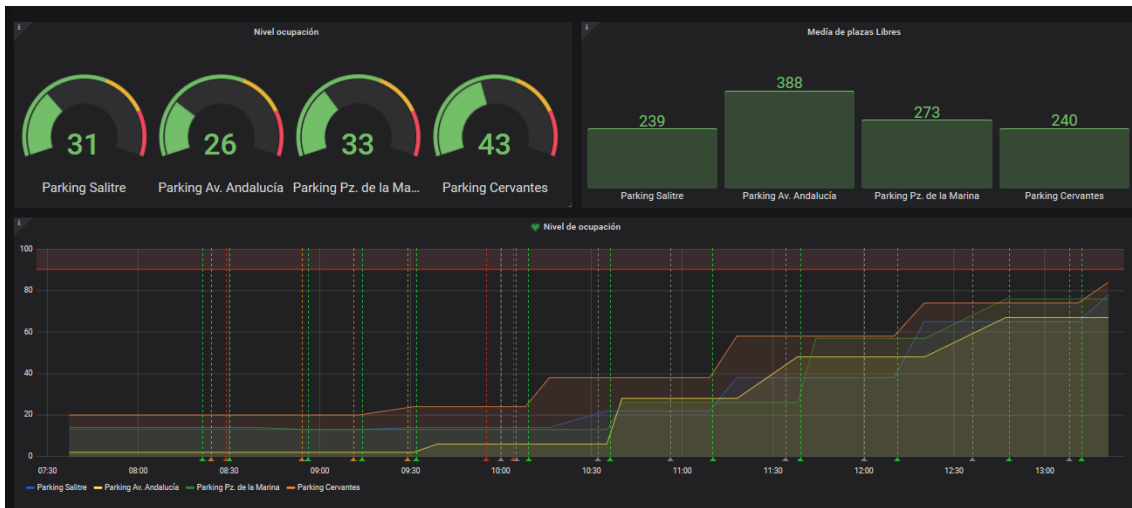
El siguiente paso será crea un gráfico de tipo *Gauge* para analizar el nivel medio de ocupación de los cuatro parkings. En la configuración del elemento de visualización se debe configurar un *Thresholds* que muestre por colores valor del nivel de ocupación: verde (menor del 60%, amarillo entre 60% y 80% y rojo superior al 80%).



Por último, se debe agregar un *Bar Gauge* en posición vertical con el número medio de plazas libres por aparcamiento. En la configuración del elemento de visualización se debe configurar un *Thresholds* que muestre por colores valor del número de plazas libres: verde (mayor de 150 plazas, amarillo entre 75 y 150 plazas y rojo inferior a 75 plazas).



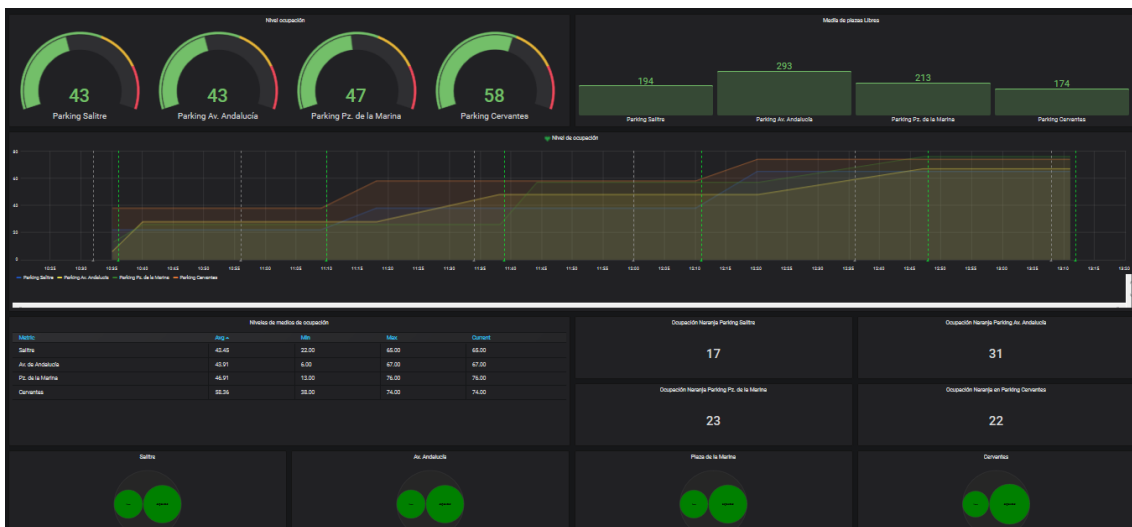
Los tres elementos de visualización creados de deben distribuir en el *dashboard*. El resultado final debe ser similar al que se muestra a continuación:



Como implementaciones adicionales, se propone:

- Una tabla, con el elemento de visualización *Table*, que muestre los datos medios de ocupación, datos mínimos y máximos y dato actual de los cuatro parkings
- Los valores de ocupación naranja de los tres parkings usando el elemento de visualización *Singlestat*.
- Un diagrama de burbujas para cada parking que indique la capacidad del parking y el número medio de plazas libres. Se debe descargar el plugin *Bubble Chart*. En este ejemplo se debe configurar un *Threshold* similar al creado en el elemento gráfico *Bar Gauge*.

El resultado final debe ser similar al de la imagen que se muestra a continuación:



7. Utilidades de Grafana:

7.1. Sistema de alertas

Grafana trae implícito un sistema de alertas sobre los cuadros de mando. El sistema permite establecer reglas de alerta y configurar la frecuencia con la que se debe evaluar la regla de alerta, así como las condiciones que deben cumplirse para que la alerta cambie de estado y active sus notificaciones vía email.

A modo de ejemplo, se va a implementar una alerta sobre un el elemento de visualización tipo *Graph* que envíe un mensaje de error cuando la ocupación del parking Cervantes sea superior al 95%.

En primer lugar, se deben configurar las variables de envío de correo *SMTP* en el fichero `/conf/defaults.ini`. Tras esto, dentro de Grafana, se deben configurar las notificaciones. En este punto es muy importante tener en cuenta que cuando una alerta cambia de estado envía notificaciones.

Para ello, se debe configurar el canal de notificación accediendo a *Alerting >> Notification channels* y crear un nuevo canal (*New channel*) con una serie de parámetros de configuración de la alerta:

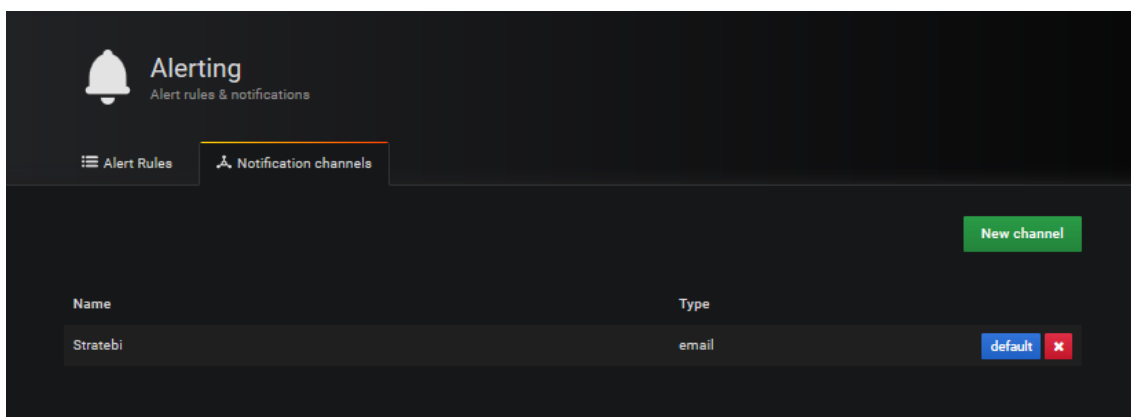
- *Name*: Nombre de la alerta.
- *Type*: Tipo de alerta (email, telegram, slack...).
- *Default (send on all alerts)*: Cuando está marcada, esta opción notificará toda las reglas de alerta existentes sobre ese canal.
- *Include image*: Si se marca, permite incluir una imagen del cuadro de mandos en el momento actual en que se envía la alerta.
- *Disable resolve messages*: Cuando está marcada, esta opción deshabilitará el mensaje de resolución [OK] que se envía cuando el estado de alerta vuelve a falso.
- *Send reminders*: Cuando esta opción esté marcada, se enviarán notificaciones adicionales (recordatorios) para las alertas activadas. Se puede especificar la frecuencia con la que se deben enviar los recordatorios usando el número de segundos (s), minutos (m) o incluso horas (h). Es muy importante, en este punto, tener en cuenta que los avisos de alerta se envían después de evaluar las reglas de las alertas. Por tanto, nunca se podrá enviar un recordatorio con más frecuencia que un intervalo de evaluación de la regla configurada.
- *Email addresses*: En caso de que el tipo de notificación sea un email, como en el ejemplo que se muestra a continuación, se definen las direcciones de los receptores de las notificaciones. Se pueden introducir varias direcciones separándolas por una coma.

The screenshot shows the 'Alerting' configuration page in Grafana. The 'Notification channels' tab is active. The 'Edit Notification Channel' form is displayed with the following settings:

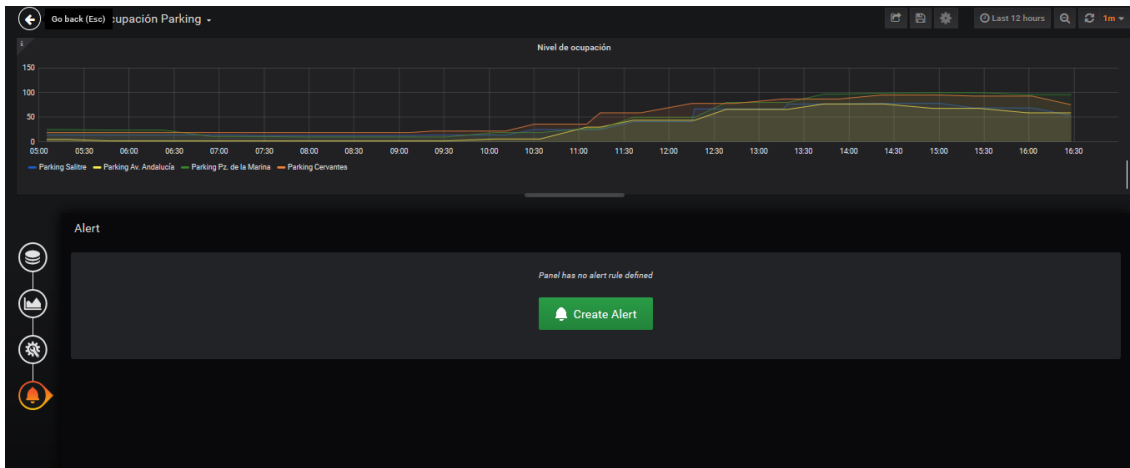
- Name: Stratebi
- Type: Email
- Default (send on all alerts):
- Include image:
- Disable Resolve Message:
- Send reminders:
- Send reminder every: 15m

A blue informational box at the bottom states: "Alert reminders are sent after rules are evaluated. Therefore a reminder can never be sent more frequently than a configured alert rule evaluation interval."

Si todo ha ido correctamente, el canal de notificación creado debe aparecer en la pestaña de *Notification channels*.



Una vez creado, se debe acceder al cuadro de mandos ya creado sobre el cual se quiere activar la alerta. Para ello, se accede al modo de edición del panel del nivel de ocupación (tipo *Graph*) y pulsar el botón de *Alert*. Una vez dentro, se debe hacer *click* sobre el botón *Create Alert*.



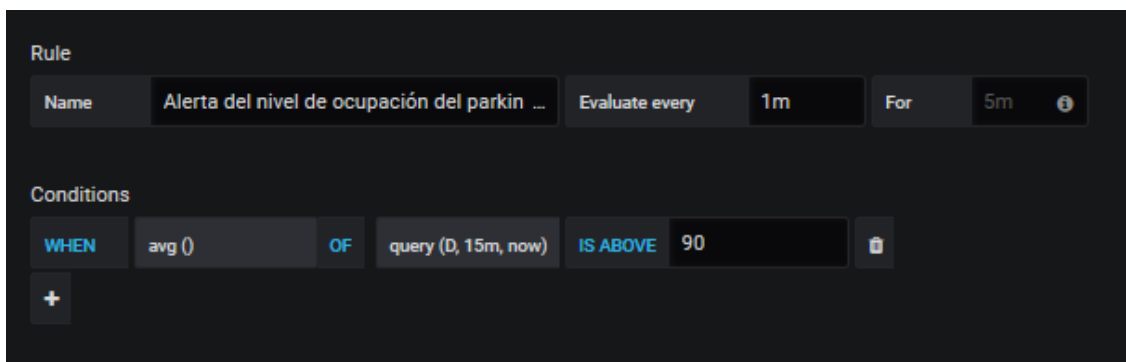
Para crear una regla de alerta se despliegan una serie de configuraciones entre las que se debe tener en cuenta:

- *Name*: Nombre de la regla de la alerta.
- Intervalo de evaluación: Aquí se especifica con qué frecuencia se debe evaluar la regla de la alerta. Si una regla de alerta tiene configurado el parámetro *For* y la condición de consulta sobrepasa el umbral configurado, la alerta pasará de *Ok* a *Pending*, antes de enviarse. Pasando de *Ok* a *Pending* Grafana no enviará ninguna notificación y una vez que la regla de alerta se haya activado durante más de lo que indica la etiqueta *For*, la alerta cambiará a estado *Alerting* y enviará notificaciones de alerta.
- *Conditions*: Permite configurar la condición de la alerta definiendo una consulta, un rango de tiempo y una función de agregación.

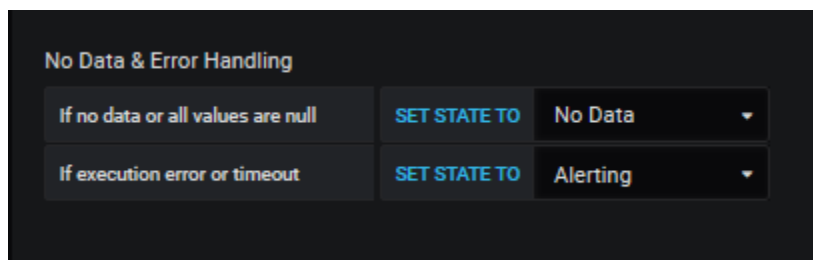
En este caso de ejemplo, los datos de la configuración serán los siguientes:

- *Name*: Alerta del nivel de ocupación del parking Cervantes
- *Evaluate every* 1m
- *Conditions*: WHEN avg() OF query (D, 15m, now) IS ABOVE 95

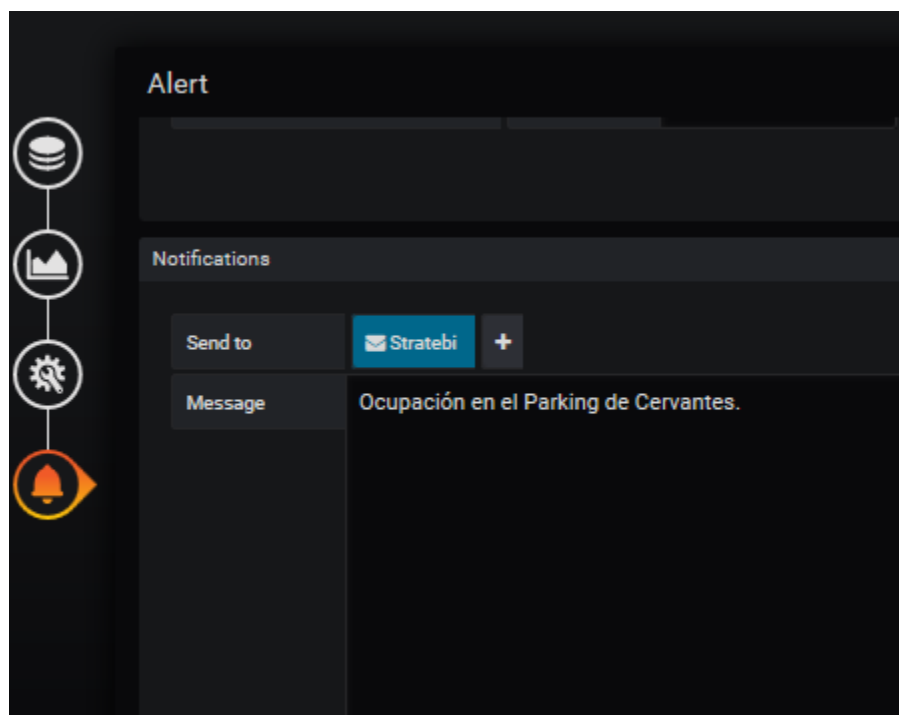
Lo que se está configurando en esta regla de alerta es una evaluación de la misma cada minuto, cuando la media de ocupación del parking Cervantes (*query D*), en los últimos 15 minutos (rango de tiempo 15m, *now*) sobrepase el umbral del 95%. En este caso, no se indica ningún valor a la etiqueta *For* para que grafana envíe la notificación directamente en caso de que se sobrepase el umbral de la condición de la regla.



A continuación, se establece el estado de la alerta a NoData en caso de que no haya datos o se encuentren valores nulos. Por su parte, si hubiera un error o *timeout* en la ejecución se establece el estado de la regla como *Alerting*.



Por último, se debe configurar el mensaje que el destinatario recibe con la alerta. En este caso de ejemplo se incluirá un mensaje relevante de la alerta.



Más información en: <https://grafana.com/docs/alerting/rules/>.

7.2. Gestión de permisos

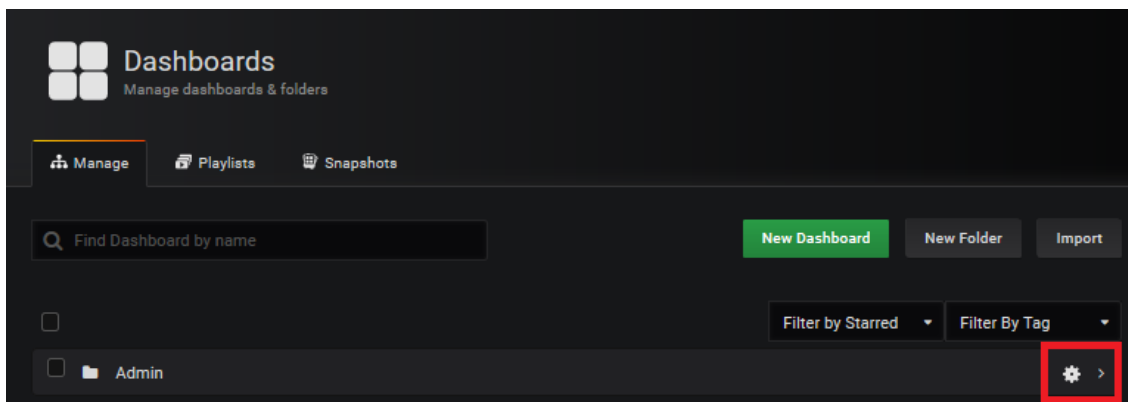
Por defecto, existen tres roles de usuario en la aplicación:

- *Admin*: Usuario con permisos totales en la aplicación.
- *Editor*: Usuario con permiso de edición sobre cuadros de mandos.
- *Viewer*: Usuario que únicamente tiene permisos de lectura en cuadros de mando.

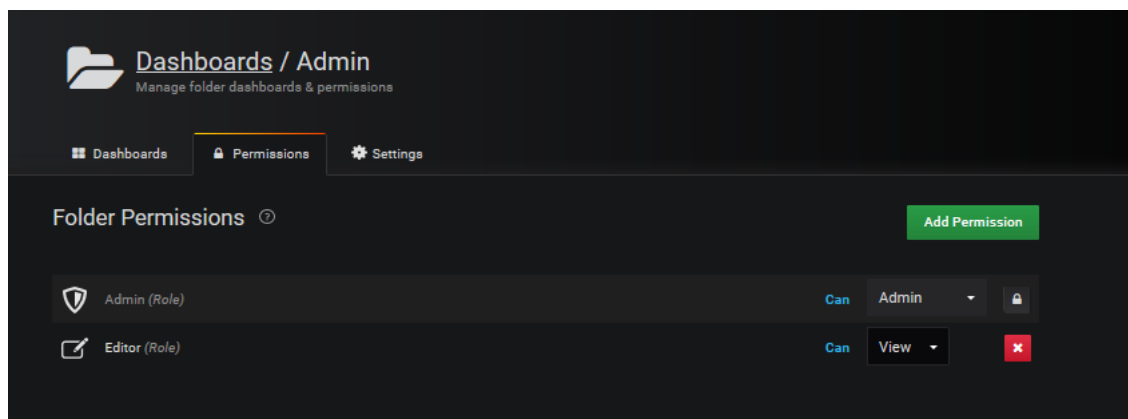
También se pueden crear grupos de usuarios dentro de un equipo de trabajo a través de la opción *Teams* dentro de *Configuration*.

Otra utilidad importante de la herramienta es que se pueden definir directorios – que contentan dashboards – y sobre estos directorios establecer permisos de acceso a nivel de usuario, rol o incluso por equipo de trabajo (*team*).

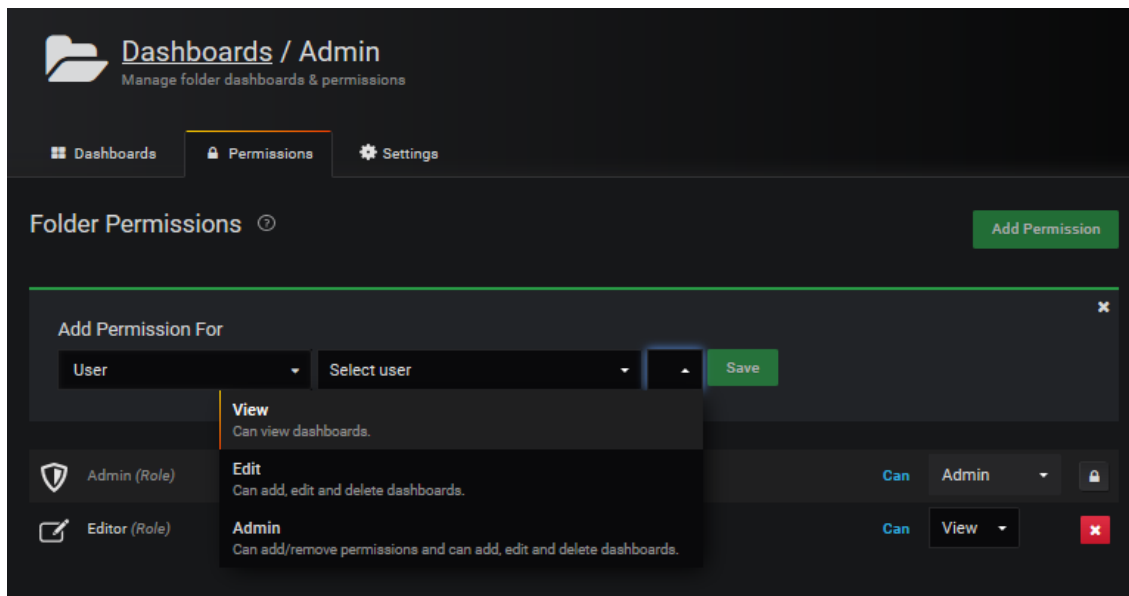
Para establecer permisos de trabajo sobre un directorio, debemos situarnos con el cursor encima del directorio y nos aparecerá a la derecha una rueda de engranaje la cual hay que pulsar para acceder a la configuración.



Tras acceder, se debe pulsar la pestaña de *Permissions* y, a través del botón *Add Permission*, configurar los permisos sobre ese directorio.



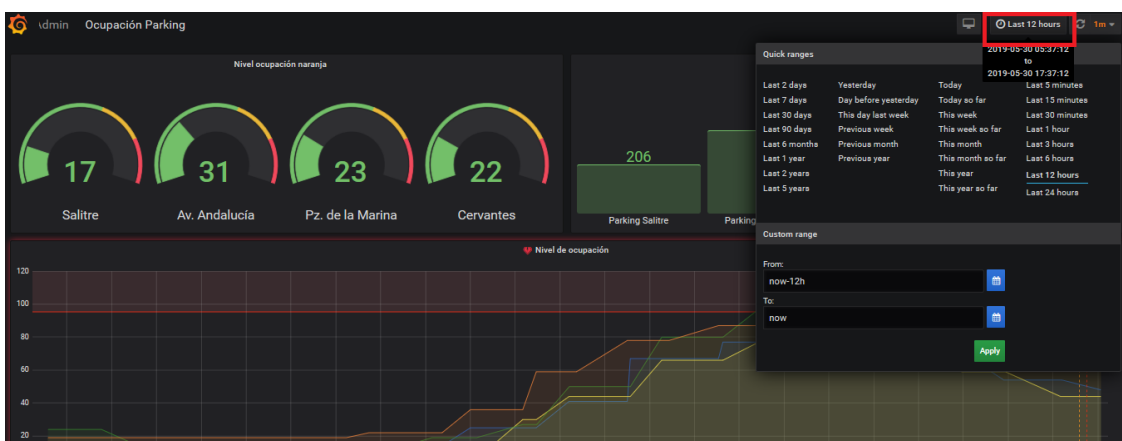
Sobre ese directorio se pueden configurar permisos por equipo o por un usuario en concreto a nivel de rol.



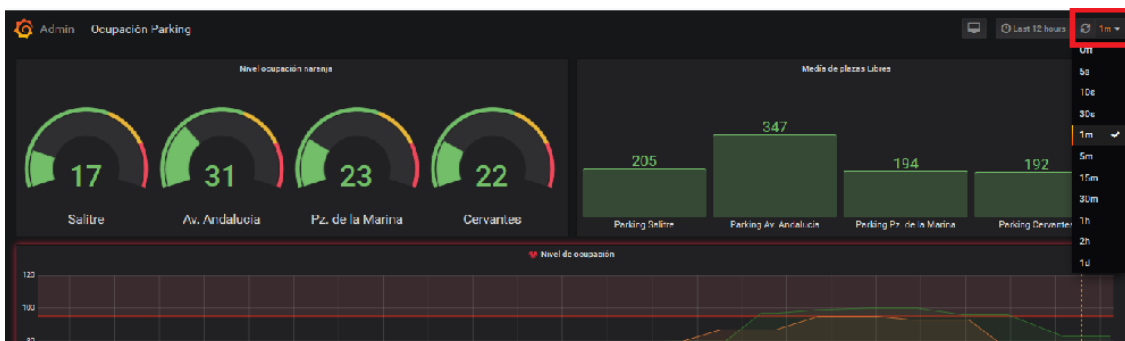
7.3. Visualización real-time

El propósito general de Grafana es la visualización de datos en tiempo real. La herramienta permite configurar la periodicidad de recarga del cuadro de mandos, así como el intervalo de tiempo de consulta.

En la parte superior de la pantalla podemos seleccionar el rango de consulta.



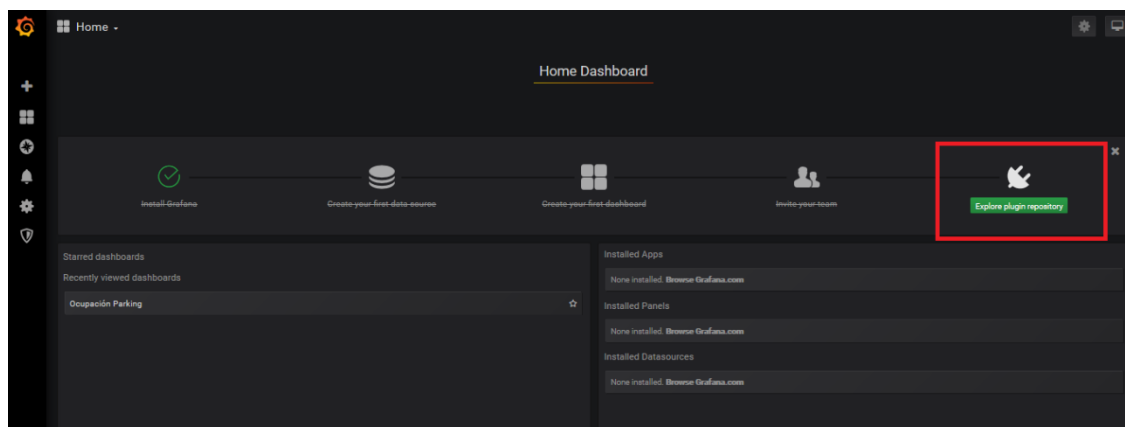
Por su parte, también podemos seleccionar la configuración del tiempo de recarga automática del cuadro de mandos.



Se debe tener en cuenta que la configuración que se guarde al crear el cuadro de mandos es la que aparecerá siempre por defecto a la hora de consultar el *dashboard*. En este caso de ejemplo, se visualizan datos de las últimas horas y la página se recarga cada minuto.

7.4. Repositorio de *plugins*

Grafana ofrece un repositorio de *plugins* que mejora las prestaciones que ofrece la herramienta. Para acceder, desde el *Home*, se debe pulsar la opción *Explore plugin repository*.



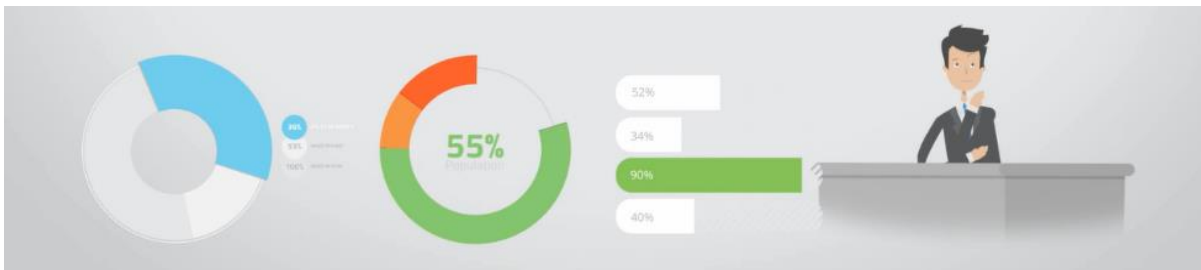
Este botón enlace a la página oficial grafana, concretamente a la comunidad de plugins: <https://grafana.com/plugins>. En el menú de la parte izquierda de la web se puede filtrar por tipo de plugin (*panel*, *data source* o *app*).

La instalación de un plugin se puede hacer descargando un fichero *.zip* y descomprimiéndolo en la carpeta */plugins/* o por línea de comandos usando la herramienta *grafana-cli*.

1. SOBRE STRATEBI

En Stratebi ofrecemos **gran cantidad de soluciones analíticas** por una compañía de **rápido crecimiento**, innovando en las áreas tecnológicas de mayor desarrollo en la actualidad: **Business Intelligence, Big Data y Machine Learning**, muchas de ellas, basadas en soluciones **Open Source**.

Además, somos **Partners Certificados en Microsoft PowerBI, Talend y Vertica...**, con gran número de proyectos con ambas tecnologías y los principales especialistas en **Pentaho** y muchas otras soluciones Open Source



Desarrollamos nuevas soluciones analíticas basadas en Open Source, para la generación de Cuadros de Mando en tiempo real, con tecnologías IoT para SmartCities, machine learning, etc...

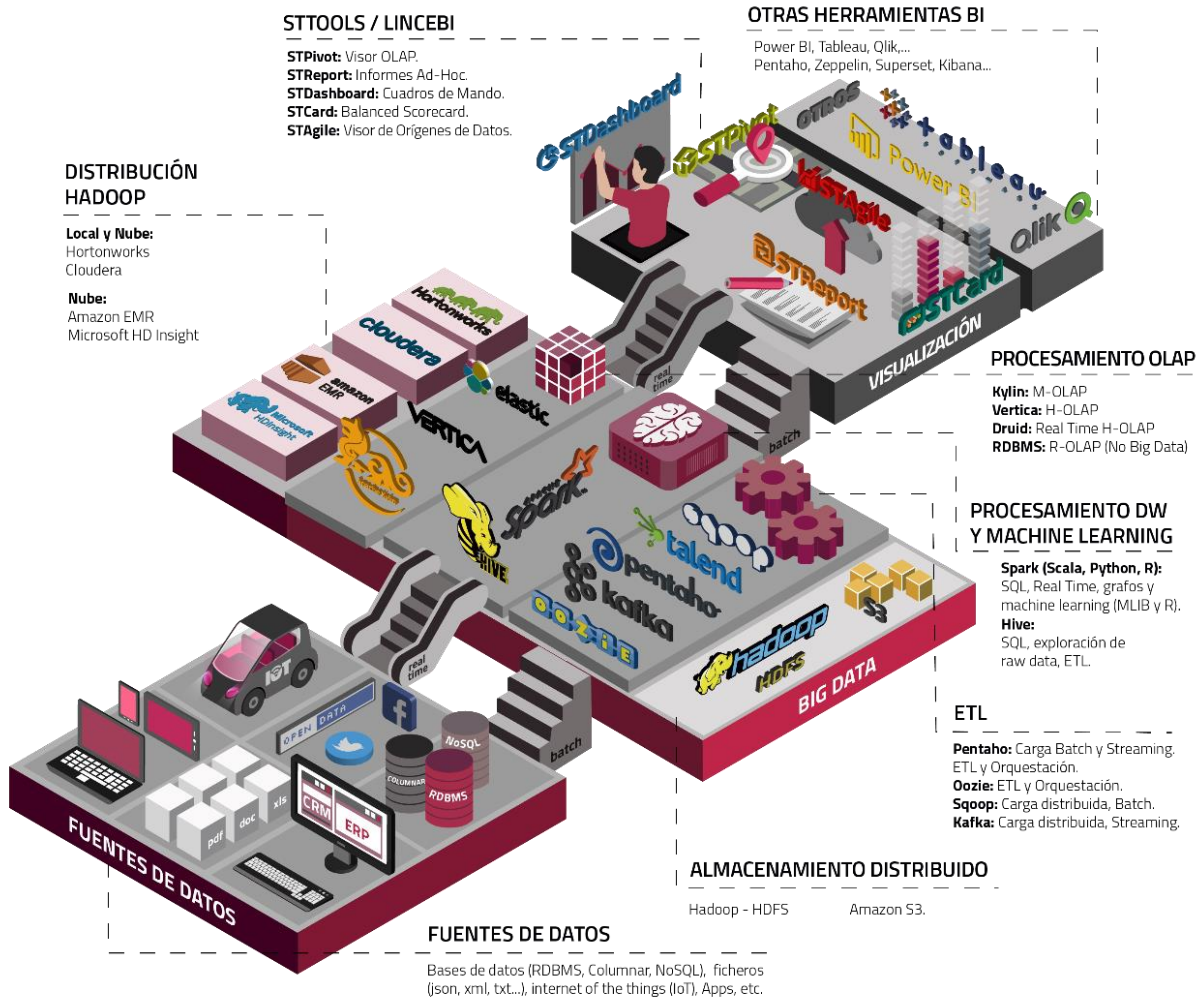


TECNOLOGÍAS

Desde hace tiempo somos Partners Certificados de Vertica, Talend, Hortonworks, Kylin y Microsoft PowerBI...



2. DESCRIPCIÓN de las soluciones



Stratebi son los creadores de la solución LinceBI.com (Big Data Analytics.)



- Big Data
- ETL y Data Quality
- Business Intelligence
- Machine Learning
- Soluciones Verticales

3. INFORMACIÓN SOBRE STRATEBI



Stratebi es una empresa española, con sede en Madrid y oficinas en Barcelona, Alicante y Sevilla, creada por un grupo de profesionales con amplia experiencia en sistemas de información, soluciones tecnológicas y procesos relacionados con soluciones de Open Source y de inteligencia de Negocio.

Esta experiencia, adquirida durante la participación en proyectos estratégicos en compañías de reconocido prestigio a nivel internacional, se ha puesto a disposición de nuestros clientes

Stratebi es la única empresa española que ha estado presente en todos los Pentaho Developers celebrados en Europa, siendo partners de Vertica, Talend, Microsoft PowerBI, Apache Kylin, Hortonworks...

En Stratebi nos planteamos como **objetivo** dotar a las compañías e instituciones, de herramientas escalables y adaptadas a sus necesidades, que conformen una estrategia Business Intelligence capaz de rentabilizar la información disponible. Para ello, nos basamos en el desarrollo de soluciones de Inteligencia de Negocio, mediante tecnología Open Source.

Stratebi son **profesores y responsables de proyectos** del Master en Business Intelligence de la Universidad UOC.

Los profesionales de Stratebi son los creadores y autores del primer weblog en español sobre el mundo del Business Intelligence, Data Warehouse, CRM, Dashboards, Scorecard y Open Source.

Todo Bi, se ha convertido en una referencia para el conocimiento y divulgación del Business Intelligence en español.

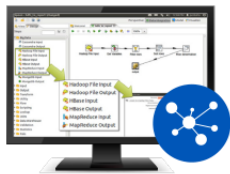
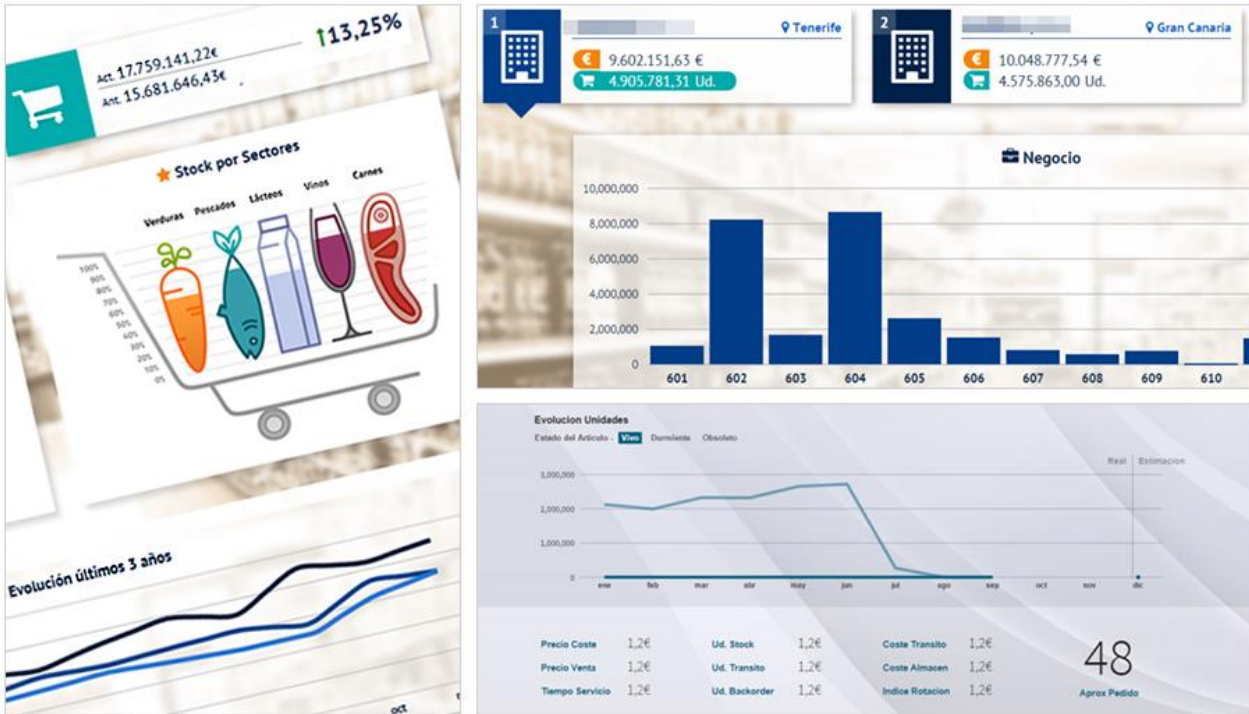
Stratebi ha sido elegida como **Caso Éxito del Observatorio de Fuentes Abiertas de Cenatic**.

Observatorio Nacional del
Software de Fuentes Abiertas



4. EJEMPLOS DE DESARROLLOS ANALYTICS

A continuación se presentan **ejemplos de algunos screenshots** de cuadros de mando diseñados por Stratebi, con el fin de dar a conocer lo que se puede llegar a obtener, así como Demos Online en la web de Stratebi:



Data Ingestion
Manipulation
Integration



Enterprise and
Ad Hoc Reporting



Data Discovery
Visualization



Predictive
Analytics

Pentaho Analytics Platform

Hadoop

NoSQL

Analytic
Databases

Relational



