

Data Quality

Índice

Introducción.....	2
Funcionamiento.....	4
Interfaz de configuración de reglas.....	4
Hoja de instrucciones.....	4
Hoja de Reglas.....	5
Hoja de Tablas.....	6
Hoja de Matrix Input.....	6
Hoja de Correos.....	7
Visualización de la calidad de los datos.....	8
Home.....	8
Resumen de Calidad.....	9
Evolución de Alertas.....	10
Alertas.....	11
Calidad de Metadatos.....	12
Arquitectura.....	13
Matrix Input.....	13
BigQuery Loader.....	13
Backend (DAG).....	14
Generación del YAML.....	14
Creación Tarea Dataplex.....	14
Ejecución QID.....	15
Ejecución QAE.....	15
Metadata Core Engine.....	15

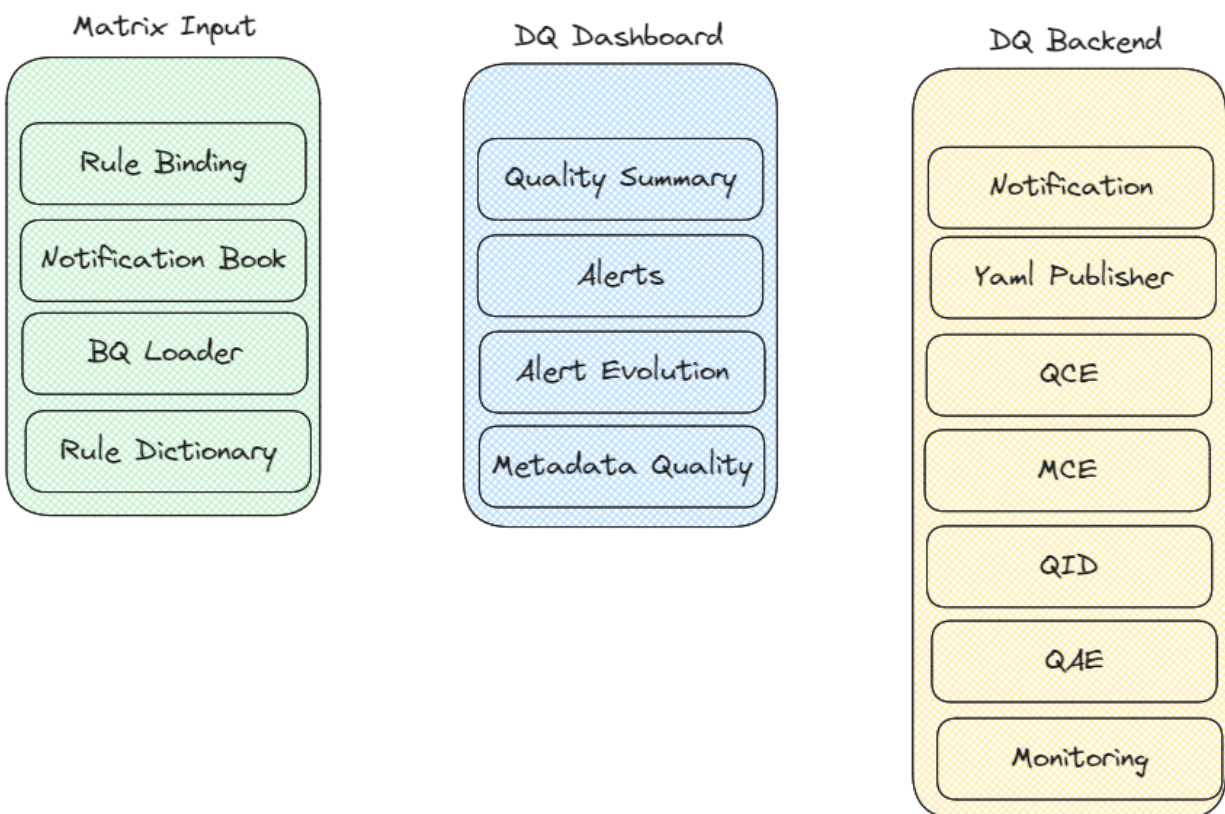
Introducción

Nuestras organizaciones buscan ser Data Driven, pero tenemos un problema muy común que es la falta de calidad en el dato, debido a ello tener confianza sobre ellos resulta difícil. Este es un problema difícil de resolver, que supone una gran carga de trabajo y que no está exenta de errores humanos.

Es por ello que hemos desarrollado Data Quality, es una plataforma cuyo principal objetivo es monitorizar la Calidad de los Datos y de los Metadatos ofreciendo una visión clara del estado de salud de los datos.

Uno de nuestros objetivos es hacerla muy sencilla de utilizar para que incluso perfiles de negocio sin conocimiento técnico sean capaces de hacer uso de esta plataforma.

Aquí presentamos un diagrama funcional de esta plataforma:



Matrix Input es la interfaz de entrada con la que interactúa el usuario. Tiene algunas funcionalidades como la aplicación de reglas de calidad (Rule Binding), la personalización de notificaciones (Notification Book), la carga automática de datos de configuración (BQ Loader) y el repositorio de reglas de calidad (Rule Dictionary). Se explicará en detalle posteriormente.

DQ Dashboard es el cuadro de mando donde el usuario puede ver todos los resultados de sus ejecuciones de Calidad, se disponen de varias hojas con distintos gráficos que se explicarán a continuación.

Por último el DQ Backend, hace referencia a toda la lógica y procesos que se ejecutan en Google Cloud. Entre ellos destacamos la funcionalidad de Notificación para notificar a los usuarios a partir del Notification Book, el Yaml Publisher que crea un yaml con la información de Matrix Input, QCE (Quality Core Engine) es el componente que ejecuta las reglas de calidad, QME (Metadata Core Engine) es el componente que se encarga de la Calidad de Metadatos. Luego disponemos del QID (Quality Intelligent Decision) que identifica los errores de Calidad y enriquece los datos y el QAE (Quality Action Engine) que es el componente que se encarga de notificar.

Para todos estos procesos dentro de Google Cloud disponemos del componente “Monitoring” con el que se puede ver cómo avanza el flujo, los errores, tiempo de ejecución de cada componente, etc.

Funcionamiento

Data Quality permite a los usuarios crear reglas, aplicar reglas a campos y consultar los resultados de esas reglas a través de unos cuadros de mando interactivos. Además, se ofrece la posibilidad de Alertar mediante correo electrónico o Google Chat a los usuarios especificados cuando se incumple alguna regla.

Interfaz de configuración de reglas

La configuración de la aplicación de las reglas de calidad a los distintos campos y tablas, así como la configuración de las alertas se realizan en la interfaz de Google Sheets, también llamado componente Matrix Input, donde las configuraciones pueden realizarse de forma sencilla y ágil. A continuación detallaremos cada una de sus hojas.

Hoja de instrucciones

Objetivo		Listar la relación de capas, tablas y campos que forman parte del proyecto y asignar la regla que aplique para garantizar la calidad del dato. Para ello se han identificado reglas de calidad básicas y avanzadas que se categorizan en Dimensiones.	
Dimensiones			
Dimensión	Descripción	Ejemplo	
Exactitud	Se mide el grado en el que los datos representan correctamente el objeto del mundo real o un evento que se describe.	La dirección de envío de pedidos a un cliente en la base de datos de clientes es la dirección real.	
Complejidad	Refiere el grado en el que el dato tiene el valor esperado y cumple con los requerimientos marcados.	Podemos establecer que los clientes tendrán sus datos completos si hemos registrado su nombre, primer apellido, número de identificación, etc.	
Consistencia	Mide si los datos están libres de contradicción y tienen coherencia lógica, de formato o temporal.	Para un cliente determinado tenemos ventas registradas pero no nos consta ninguna orden de pedido.	
Integridad	La integridad refiere a la integridad referencial (consistencia entre índices de tablas) o a la coherencia interna para que no falten datos.		
Disponibilidad	Refiere a la disponibilidad de un dato en el momento esperado o que se refresque con una determinada frecuencia para garantizar que el valor es vigente.		
Unicidad	La unicidad o deduplicación establece que no existe más de una entidad en el mismo conjunto de datos.	En nuestra base de datos podemos tener dos clientes que se registraron como «Fran García» y «Francisco Juan García», siendo la misma persona pero sólo el último contiene todos los datos completos.	
Validez	La validez indica si los valores son consistentes con los definidos dentro del dominio de valores definido.		
Instrucciones para completar las pestañas:			
Tablas	Listado de tablas que forman parte del proyecto. Se debe indicar la descripción, proyecto y dataset.		
Journey	Se refiere a documentar el linaje de los datos por las diferentes capas, es decir, indicar el origen y destino de las tablas.		
Reglas	Listado de reglas de definidas agrupadas por Dimensión.		
Matriz	Asignar la reglas que aplique a la tabla/campo del proyecto.		
Valores a completar			
Tabla	Nombre de la tabla		
Campo	Nombre de campo		
Descripción	Descripción de campo		
Tipo	Tipo de dato (Número entero, cadena de caracteres...)		
Capa	Capa de persistencia de la tabla		
BU	Business Unit		
Tipo de Motor	Core o Advanced		
Reglas	Marcar con X o Indicar el umbral que aplica al campo de la tabla		
Severidad		Acción	
1 (Baja)	0 (Sin acción)		
2 (Media)	1 (Parada y se muestra la alerta en el CdM)		
3 (Alta)	2 (Notificación y se muestra la alerta en el CdM)		
	3 (No Notifica pero se muestra la alerta en el CdM)		

- Severidad 0 - No hacer nada pero tenerla registrada
- Severidad 1 - Notificar en el Cuadro de Mando pero no por Correo
- Severidad 2 - Notificar por Correo y mostrarla en el Cuadro de Mando


Hoja de Reglas

En esta hoja se ve el registro completo de reglas que se disponen para aplicar en la plataforma. Contiene información como: dimensión, nombre de la regla, descripción, ejemplo, si la regla tiene o no parámetros y cuales serían, la definición de la regla en código YML y la oportunidad de establecer valores de Severidad y Acción personalizados para cada regla.

En esta hoja se pueden crear nuevas reglas insertando nuevas filas. Estas reglas también se pueden personalizar para adecuarse más a los casos de uso del cliente.

Hoja de Tablas


Datos del Proyecto. (cumplimenta DGQO)		(todo en minúsculas)	
Nombre del Producto:	Producto_1		
Motor Reglas	CORE		
Entorno	Test		
Proyecto CDP DGQOffice asignado al producto:	proyecto_1		
Dataset GCP DGQOffice asignado al producto:	dataset_1		
Localización	europa-west3	(FRÁNCFORT ALEMANIA)	
Lakes			
Zones			
Documento de referencia (Owner BDP)			



Tablas	Descripción	Proyecto	Dataset
Ventas		proyecto_1	dataset_1
Tienda		proyecto_1	dataset_1
Producto		proyecto_1	dataset_1
Marca		proyecto_1	dataset_1

Esta hoja actúa como hoja de configuración para registrar en qué proyecto y en qué dataset quieres cargar los resultados de las reglas de Calidad. Además, deberás indicar el entorno que validará esta Matrix Input, recomendando que haya un documento Matrix Input por entorno. También quedarán registrados los proyectos, datasets y tablas que se quieren auditar. Pueden ser tablas de distintos proyectos y distintos datasets. Todos los datos necesarios del proyecto indicado se cargan automáticamente.

Hoja de Matrix Input

<div>  </div>															
										Reglas					
Tabla	Campo	Descripción Y Comentarios	Prioridad	Tipo	Capa	BU	TipoMotor	NOT_NULL_SIMPLE	NOT_BLANK	ACCEPTED_GROUP_FIX	VALUES_ALWAYS_EXPECTED	ACCEPTED_COUNTRY_FIX	VALUE_POSITIVE		
Tabla	Campo	Descripción Y Comentarios	Prioridad	Tipo	Capa	BU	TipoMotor			group_fix	cardinality	country_column			
Ventas	id	id de las ventas	alta	INTEGER	SILVER	ES	CORE	x						x	
Tienda	nombre	nombre de la tienda	alta	STRING	BRONZE	ES	CORE	x	x						
Producto	precio	precio del producto	alta	FLOAT	BRONZE	ES	CORE	x						x	
Marca	nombre	nombre de la marca	media	STRING	GOLD	ES	CORE	x	x						
Ventas	fecha	fecha de venta	baja	DATE	GOLD	ES	CORE	x							
Ventas	unidades	unidades de venta	media	INTEGER	GOLD	ES	CORE	x						x	
Producto	idmarca	id de la marca del pr	baja	INTEGER	GOLD	ES	CORE	x						x	

En esta hoja es donde se asigna qué reglas de calidad debe cumplir cada campo de cada tabla de manera tan simple como marcando con una 'x' o con un valor dependiendo del tipo de regla.

Los campos y las tablas se indican a la izquierda por filas. Ofrecemos un campo de comentarios para dejar comentarios sobre por qué se aplica la regla o similares. También se puede especificar un valor de prioridad ofreciendo la posibilidad de filtrar sobre los bindings (llamamos binding a la relación entre reglas y campos, concretamente una fila).


Luego disponemos de los campos tipo de dato, capa de almacenamiento a la que pertenece, business unit y tipo de motor. Estos campos forman parte de los metadatos del binding y también son personalizables de cara al cliente.

En la parte derecha de la imagen tenemos todas las reglas definidas en la hoja de Reglas (también denominada Rule Dictionary o diccionario de reglas). Para aplicar estas reglas sobre un campo basta con marcar con una 'x' la casilla de esa reglas para esa fila (siendo la fila el campo). Es importante mencionar que sólo se marcará una 'x' si la regla no requiere de parámetros. En otras palabras, la fila de arriba contiene el nombre de la regla y la de abajo contiene los nombres de los parámetros para esa regla. Si la celda se encuentra vacía y en gris quiere decir que no tiene parámetros y se marcará con una 'x'. En caso contrario, si la celda tiene valor y está de color verde, habrá que marcar la casilla con un valor, siendo este el valor del parámetro para esa regla.

Si una regla requiere o no parámetro está especificado tanto en Matrix Input (comentado en el párrafo anterior) como en la hoja de Reglas en la columna de 'parámetros'.

No hay límite de reglas a aplicar en un binding. Aunque sí es importante mencionar que hay reglas específicas para un tipo de dato. Por lo que algunas reglas no se podrán aplicar si están pensadas para tipo fecha y el campo es de tipo numérico. Estas restricciones se muestran directamente en la hoja de cálculo sombreando las celdas de color gris, indicando que no se puede aplicar esa regla sobre ese campo.

Hoja de Correos

			
nombre	correo	entorno	severidad
Example	example@gmail.com	Test	1 (Baja)
Example	example@gmail.com	Pro	1 (Baja)
Example	example@gmail.com	Test	2 (Media)
Example	example@gmail.com	Pro	2 (Media)
Example	example@gmail.com	Test	3 (Alta)
Example	example@gmail.com	Pro	3 (Alta)

En esta hoja permitimos al cliente personalizar los usuarios que deben ser notificados en base a la severidad y al entorno de ejecución. Un usuario puede ser notificado para varias severidades o varios entornos así como para un solo entorno y una severidad.

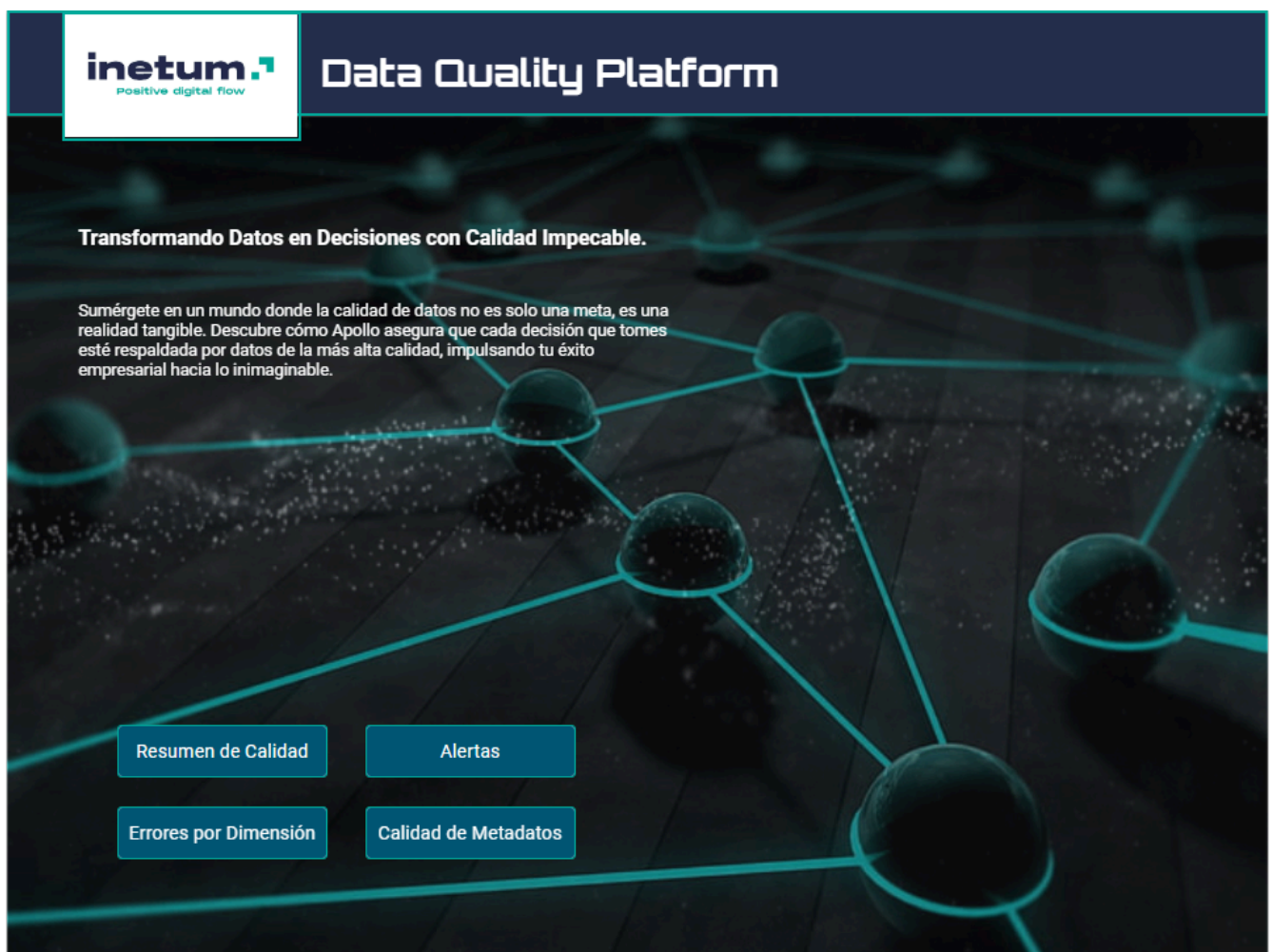
El usuario sólo será notificado si se incumple alguna regla del entorno indicado y si alguna de esas reglas tiene la severidad establecida en esta hoja.

Visualización de la calidad de los datos

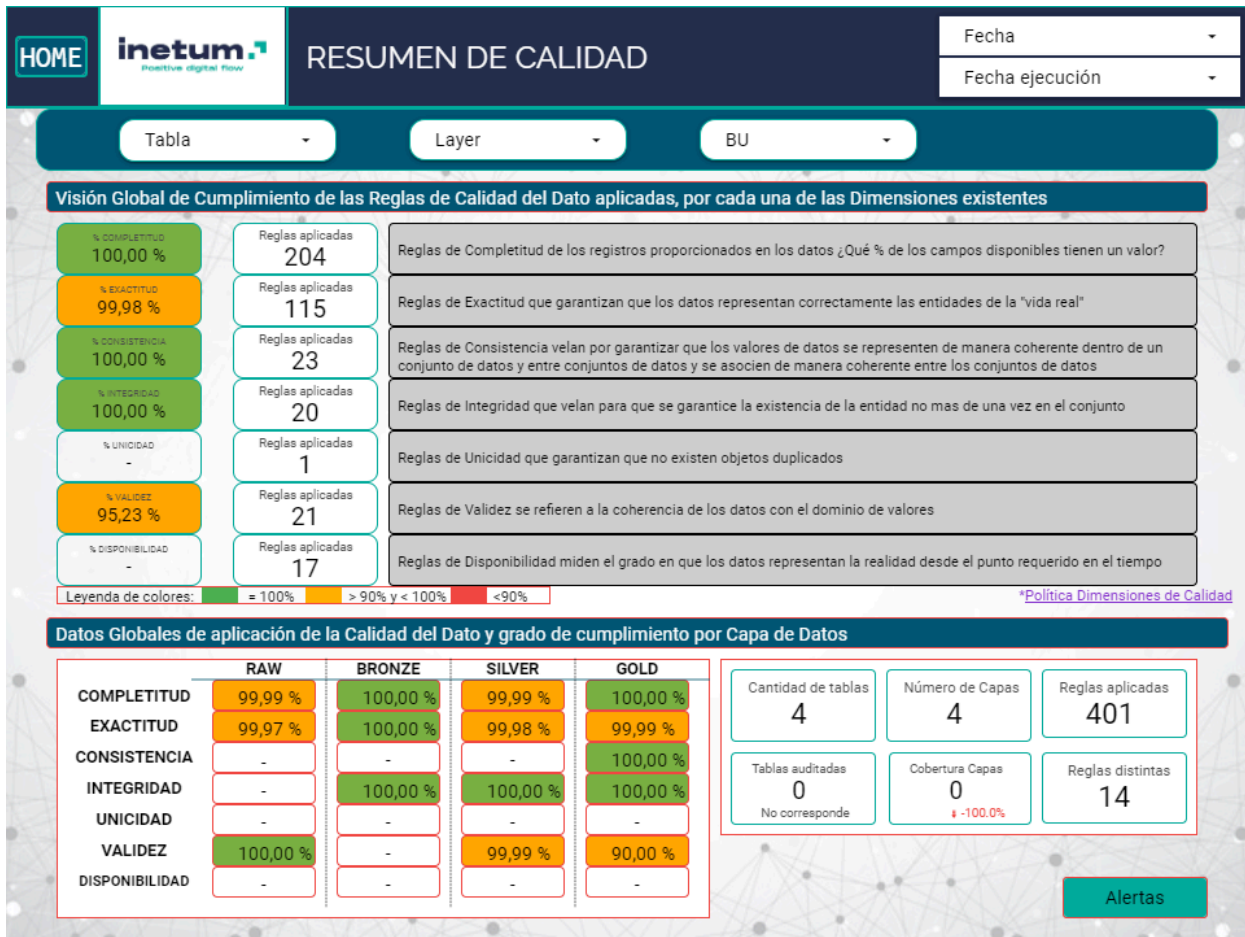
Ahora vamos a presentar la interfaz de salida, es decir, los cuadros de mando que muestran los resultados de las reglas aplicadas. A continuación vamos a explicar cada una de las páginas:

Home

Nada más entrar se nos recibe con la pantalla principal (Home) que sirve como índice para movernos por los cuadros de mando. Se nos muestran 4 botones para navegar hacia las otras páginas.



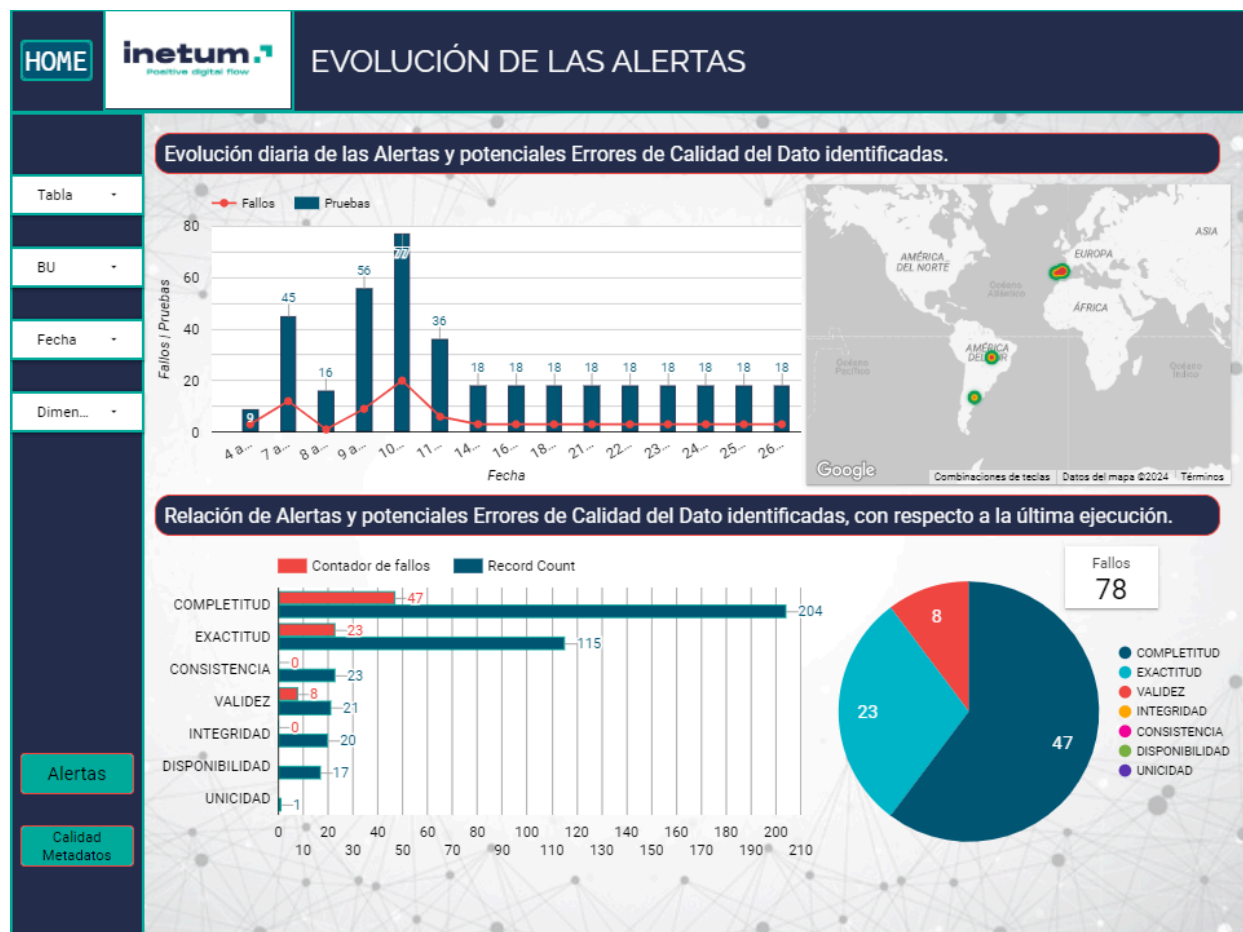
Resumen de Calidad



Esta página muestra los porcentajes de reglas cumplidas y el número de reglas aplicadas agrupado por dimensión de calidad. La parte inferior contiene un desglose de los porcentajes por las distintas capas de Almacenamiento y algunas métricas extra como el número total de reglas aplicadas, las reglas distintas que han sido aplicadas, el número de Capas de Almacenamiento, las capas de almacenamiento auditadas y de la misma forma con las tablas.

Cabe mencionar que en todas las hojas se disponen de filtros para mejorar la experiencia del usuario y facilitar la detección de problemas o insights.

Evolución de Alertas



En esta hoja se muestra un histórico de las reglas ejecutadas y las reglas incumplidas a lo largo del tiempo. En la parte superior derecha se ve un mapa de calor por las distintas Business Units que dispone el cliente y en la parte inferior más gráficos exponiendo información de las reglas incumplidas sobre las totales agrupadas por dimensión.

También se disponen de filtros para visualizar la serie temporal por el rango de fechas deseado, filtros de dimensión y tabla entre otros. Por último, se nos ofrece la capacidad de navegar libremente por el mapa de calor con capacidad de Zoom o desplazamiento.

Alertas

[HOME](#)

LISTADO DE ALERTAS

Fecha ▼

Fecha ejecución ▼

Columna ▼

Layer ▼

BU ▼

Dimensión ▼

Listado de Alertas y potenciales Errores de Calidad del Dato identificadas.

	message	Dimensión	Tabla	Columna	Regla incump...	Severidad	Acción	Mensaje	Nº fallos
1.	Hay algún valor nulo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: unidades	COMPLETITUD	ceep-394706.conjuntopruebaceep.Ventas	unidades	NOT_NULL_SIMPLE	2	2	Hay algún valor nulo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: unidades	1
2.	Hay algún valor nulo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: ID	COMPLETITUD	ceep-394706.conjuntopruebaceep.Ventas	ID	NOT_NULL_SIMPLE	2	2	Hay algún valor nulo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: ID	1
3.	Hay algún valor no positivo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: ID	EXACTITUD	ceep-394706.conjuntopruebaceep.Ventas	ID	VALUE_POSITIVE	1	1	Hay algún valor no positivo en: ceep-394706.conjuntopruebaceep.Ventas y en campo: ID	1
4.	null	VALIDEZ	ceep-394706.conjuntopruebaceep.Marca	Nombre	REGEX_VALID_UPPER_STRING	null	null	null	4
5.	null	VALIDEZ	ceep-394706.conjuntopruebaceep.Ventas	fecha	ACCEPTED_DATE_RANGE	null	null	null	4

1 - 5 / 5 < >

Query resultado ▼

1.

```
WITH  
  zero_record AS (  
    SELECT  
      'DGOO_VENTAS_UNIDADES' AS rule_binding_id,  
    ),  
  data AS (  
    SELECT  
      'DGOO_VENTAS_UNIDADES' AS rule_binding_id,  
    FROM  
      'ceep-394706.conjuntopruebaceep.Ventas' d  
    WHERE  
      True  
  ),  
  last_mod AS (  
    SELECT  
      project_id || ':' || dataset_id || ':' || table_id AS internal_table_id,  
      TIMESTAMP_MILLIS(last_modified_time) AS last_modified  
    FROM 'ceep-394706.conjuntopruebaceep._TABLES_'
```

1 - 1 / 69 < >

Resumen Calidad

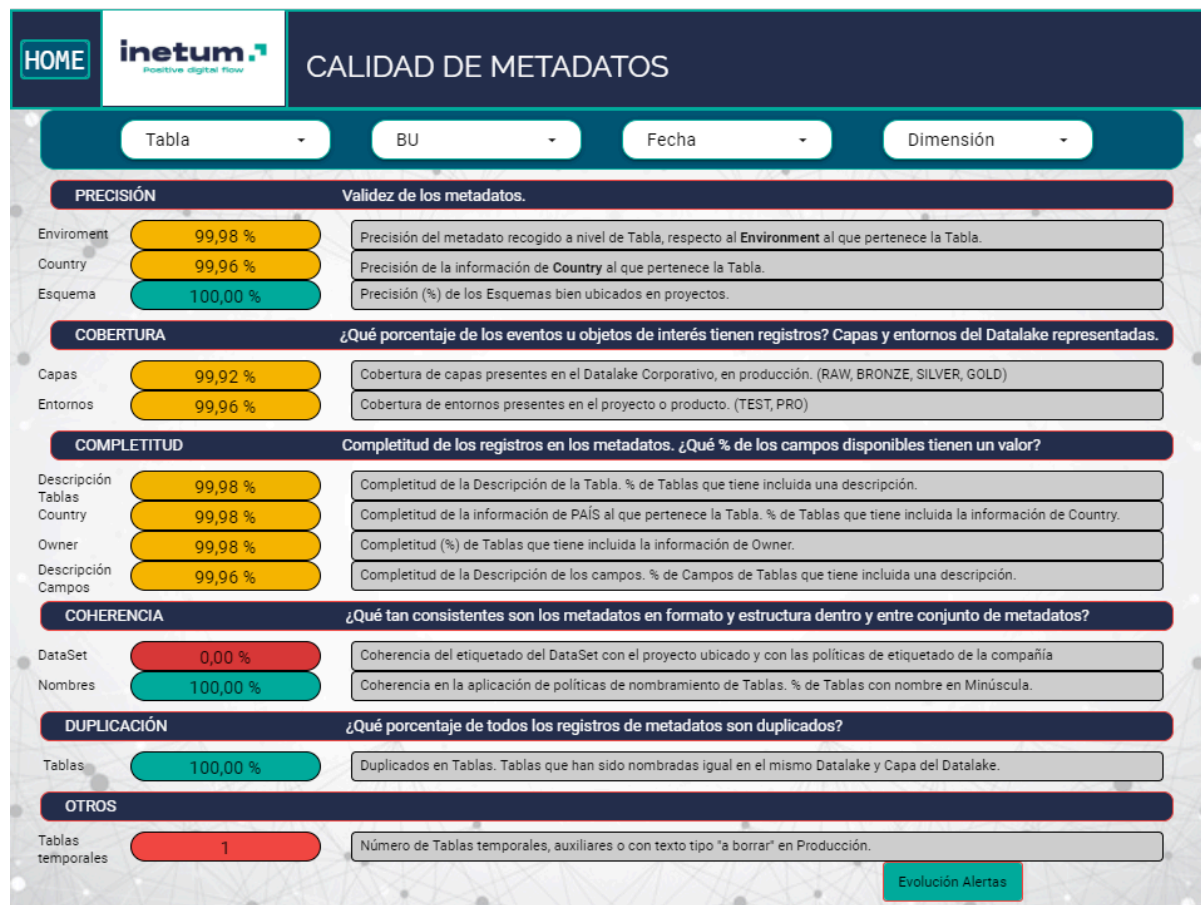
Evolución Alertas

Esta página está más orientada a perfiles técnicos y contiene información sobre las reglas incumplidas. Algunas de las columnas que contiene son: la tabla y columna auditada, la regla ejecutada, el nivel de severidad y acción, y número de fallos, entre otras.

En la parte inferior se encuentra una sentencia SQL con la que el usuario podrá recuperar los registros que no superaron con éxito la regla indicada. De esta forma el usuario en cuestión podrá tomar medidas para solucionar o investigar esos registros.

Es importante mencionar que sólo se muestra la sentencia SQL y no se muestran los datos directamente. Esto se debe a la Ley de Protección de Datos. Como no se sabe exactamente qué usuarios van a tener acceso a este cuadro de mando, se opta por no mostrar los datos explícitamente por si algún usuario no debería ver los datos. Por esto proporcionamos una sentencia que el usuario puede copiar y pegar en el espacio de BigQuery para recuperar esas filas afectadas y tomar las medidas necesarias. Solo el usuario con los permisos suficientes en BigQuery podrá ejecutar esa sentencia y visualizar los datos.

Calidad de Metadatos



Esta última página muestra toda la información relevante a la parte de Calidad de Metadatos. Entre estas métricas se destacan algunas como: el porcentaje de tablas o campos que tienen descripción, las tablas que no pertenecen al entorno en el que se encuentran, las tablas que no contienen la información de country o capa...

También se valida que la nomenclatura especificada por el cliente se esté respetando en los datasets y las tablas. Esto se realiza de forma conjunta con el cliente y de momento no se dispone de interfaz para que el usuario pueda modificar las reglas de Calidad de metadatos una vez definidas.

Por último mencionar que la ejecución de estas reglas se realiza de forma automática y el usuario únicamente interactúa con la Matrix Input y los cuadros de mando. Este proceso se realiza en Google Cloud y se puede personalizar la frecuencia de ejecución de las reglas.

Arquitectura

Matrix Input

Como hemos visto, Matrix Input es la interfaz de configuración de Data Quality y está construida en Google Sheets, pero todas las funcionalidades (listas desplegables y la validación de bindings duplicados entre otras) están programadas completamente desde Apps Script.

BigQuery Loader

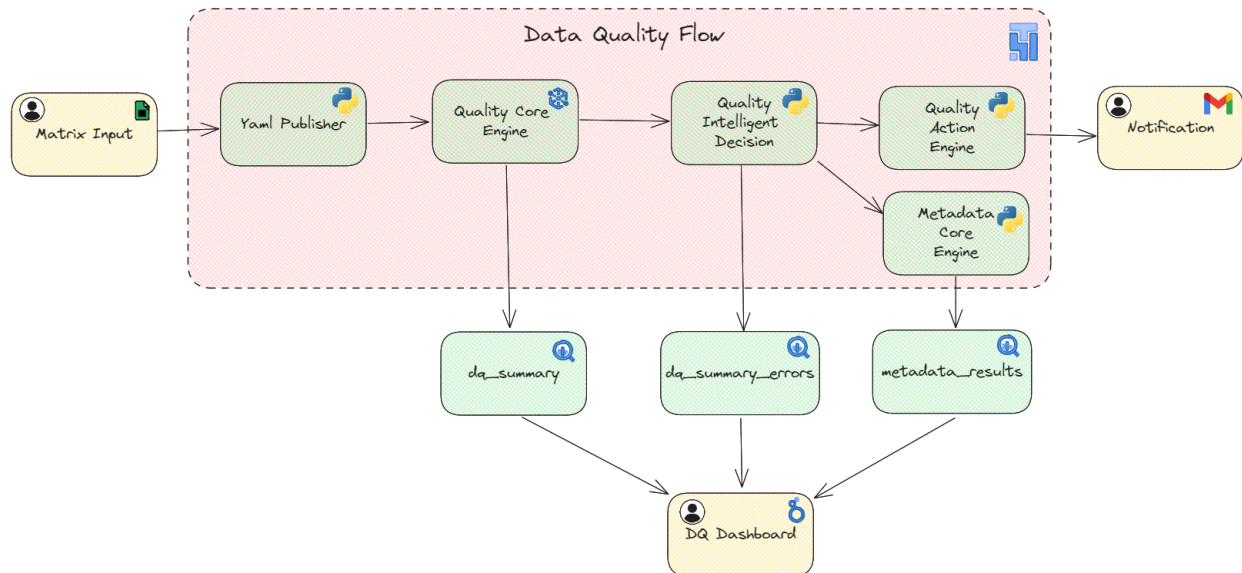
Para cargar la información sobre el proyecto de Google Cloud sobre el que se va a aplicar la calidad de datos en Google Sheets disponemos de un componente “BigQuery Loader” que carga automáticamente todos los datos necesarios del proyecto indicado.

Este componente consiste en una cloud function que accede a toda la información de BigQuery y la inserta directamente en la Hoja de Tablas en las columnas correspondientes. Se inserta el proyecto, con todos sus datasets, todas las tablas de estos datasets y todos los campos de estas tablas. Esto previene que el usuario pueda cometer fallos a la hora de completar los campos y agiliza mucho el proceso.

Backend (DAG)

Toda la ejecución de Data Quality, la aplicación de reglas, recopilación de resultados y la gestión de alertas se hace a través de un DAG de Composer. Un DAG es un grafo acíclico dirigido que permite orquestar el flujo de distintas tareas.

A grandes rasgos y alta abstracción se podría definir el DAG de DQ como:



Englobar todo este procesamiento en un DAG permite la monitorización de todas las tareas de Data Quality y la automatización de las mismas.

La ejecución del DAG se puede programar con una expresión CRON que de momento se hará a mano para el cliente ya que aún no existe interfaz para poder cambiarlo.

A continuación vamos a analizar en detalle cada una de las tareas del DAG:

Generación del YAML

La primera tarea se encarga de construir el yaml que contiene las reglas de calidad a evaluar por Dataplex. Ésto se hace directamente con un script de python que accede a la Matrix Input y construye el yaml en función de las asignaciones de reglas a campos. Luego guarda ese yaml en un bucket de Cloud Storage para que lo pueda recuperar Dataplex.

Creación Tarea Dataplex

En este paso se crea la tarea de Dataplex a partir del yaml generado por el paso anterior. Dataplex entonces ejecuta las tareas definidas y una vez finaliza la ejecución se guardan los resultados en la tabla "dq_summary" en BigQuery.

Ejecución QID

El script de QID (Quality Intelligent Decision) consiste en un script SQL que trunca la tabla de “dq_summary_errors” e inserta los registros que se marquen como error en “dq_summary” enriqueciéndolos con valores sacados de Matrix Input, como puede ser la severidad o la acción. Una vez generado el script SQL de QID, se ejecuta sobre BigQuery. Una vez finalizado ese proceso, se ejecutan las 2 siguientes tareas en paralelo (Ejecución QAE y MCE).

Ejecución QAE

El QAE (Quality Action Engine) ejecuta otro script SQL para identificar si ha habido errores en el día o más concretamente en la tarea que se acaba de ejecutar. Luego, en caso de que haya errores en ese día, con la ayuda de la Hoja de Correos de la Matrix Input, se notifica a los usuarios correspondientes. Esta notificación se puede hacer por email, google chat o como requiera el usuario.

Metadata Core Engine

La otra tarea que se ejecuta en paralelo es la encargada de la calidad de metadatos. Este proceso crea tablas para almacenar información y métricas sobre la Calidad de Metadatos. Se tratan de consultas que se realizan sobre los esquemas de metadatos que ofrece BigQuery sacando KPIs y valores que se deben definir previamente con el cliente. Estas tablas están construidas de tal forma que la conexión con Looker Studio (donde se encuentran los cuadros de mando) sea lo más sencilla y requiera el menor tiempo posible. Se ha modularizado este proceso para que toda la lógica de consulta esté alojada en el DAG y no haya nada o lo más mínimo en Looker Studio. Así se favorece también la reutilización del cuadro de mando y sus consultas y se agiliza el despliegue de la plataforma reduciendo el tiempo y complejidad del cuadro de mando de Looker.

Siguiendo esa filosofía y por razones de seguridad no se relaciona el cuadro de mando de Looker directamente con las tablas de “dq_summary” o “dq_summary_errors”, sino con vistas sobre éstas con los campos requeridos para crear los filtros y los gráficos del cuadro de mando.

Importante mencionar que para las consultas de calidad de metadatos, estamos utilizando tablas para almacenar los resultados y NO vistas. Esto se debe a que para crear y visualizar esas vistas se requieren permisos de BigQuery Admin a los usuarios nominales. Sin embargo si materializamos esos resultados directamente en tablas no existe ese problema, la única pega es que los resultados de calidad de metadatos se actualizarán con cada ejecución del DAG en vez de automáticamente como se haría con una vista.