



Teoría

ABAP Cloud – Profundizando los Conceptos

SAP S/4HANA Cloud – Modelo de extensibilidad Clean Core





Contenido

1. ABAP Cloud – Profundizando los Conceptos	3
1.1. Cloud Native – Principios	3
1.2. Arquitectura REST	6
1.3. Explorando ABAP Cloud	8
1.4. ABAP Cloud – Artefactos RAP	11
1.5. Exploración de las API publicadas	12



1. ABAP Cloud – Profundizando los Conceptos

1.1. Cloud Native – Principios

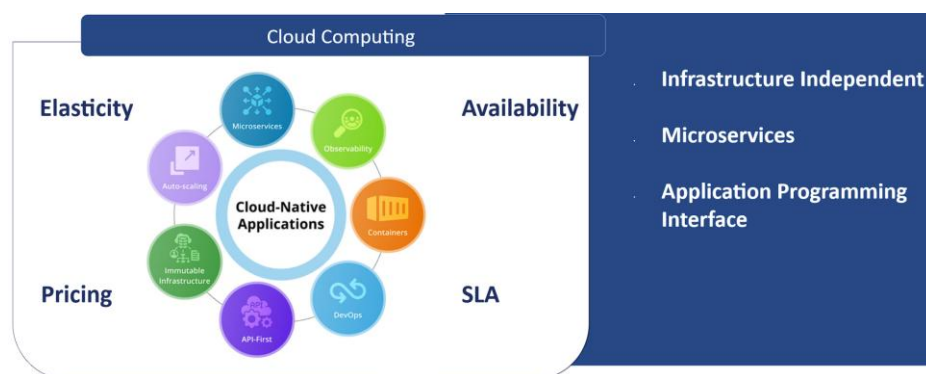
Cloud Native, o computación en la nube, se refiere a la infraestructura tecnológica diseñada para operar en la nube, a diferencia de la infraestructura tradicional de centros de datos locales. Esta arquitectura se caracteriza por alojar componentes de infraestructura como servidores, almacenamiento, sistemas operativos y virtualización en la nube, proporcionados por proveedores externos. Este modelo elimina la necesidad de que las empresas instalen y mantengan equipos físicos, delegando estas responsabilidades al proveedor de la nube.

Componentes y Principios Fundamentales:

- **Elasticidad:** La elasticidad es uno de los principios clave de la computación en la nube. Este concepto se refiere a la capacidad de ajustar dinámicamente los recursos informáticos según la demanda. Las organizaciones suelen experimentar fluctuaciones en el uso de recursos, como un centro de ventas que puede tener picos altos en ciertos periodos y bajos en otros. La elasticidad permite que las empresas paguen solo por los recursos que necesitan en un momento dado, ajustándose automáticamente a las variaciones en la demanda.
- **Precio:** El modelo de precios en la computación en la nube es flexible y varía entre proveedores. Los clientes pueden elegir entre distintos planes de consumo y suscripción, adaptándose a sus necesidades específicas. Por ejemplo, SAP ofrece diversos entornos de ejecución y servicios en su plataforma BTP (Business Technology Platform), permitiendo tanto planes basados en suscripción como en consumo.
- **Disponibilidad:** Es otro componente fundamental del Cloud Native. Dependiendo de la naturaleza del negocio, las empresas pueden necesitar más o menos recursos en diferentes periodos. Este principio se refiere a la capacidad de acceder a los recursos en la nube cuando se necesiten, garantizando que los servicios estén disponibles según la demanda del cliente. Las empresas pueden contratar servicios con disponibilidad 24/7 o ajustar los tiempos de acceso según sus necesidades para optimizar costos y eficiencia.



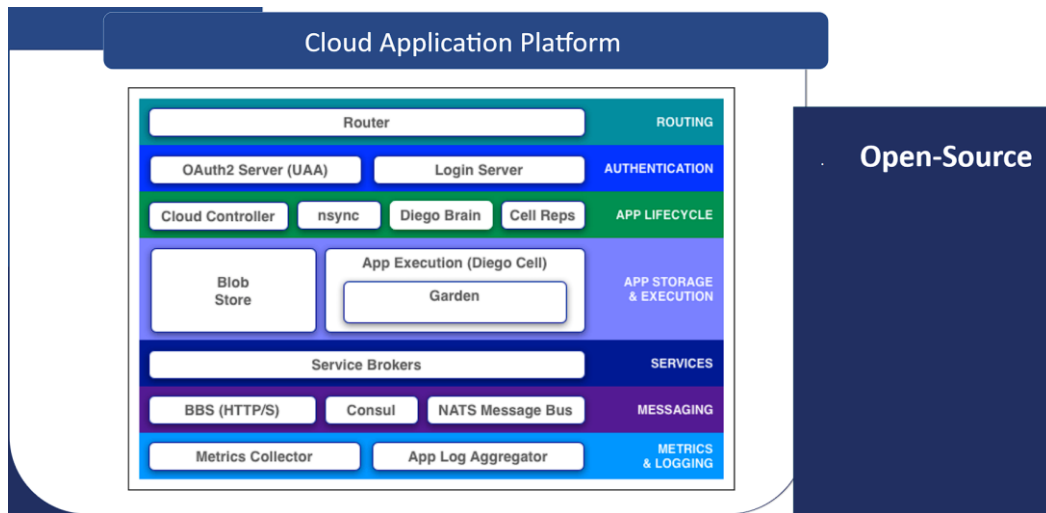
- **SLA (Service Level Agreement):** El SLA, o acuerdo de nivel de servicio, es un componente crítico que define el nivel de servicio esperado del proveedor de la nube. Similar a la disponibilidad, el SLA establece las condiciones bajo las cuales los clientes pueden acceder a los servicios y pagar solo por lo que realmente utilizan. Esto proporciona una transparencia y confiabilidad en la relación entre el proveedor y el cliente, asegurando que se cumplan los estándares de calidad acordados.
- **La infraestructura independiente en Cloud Native:** Se refiere a la separación de los componentes de infraestructura del control directo de la empresa. Esto permite una mayor flexibilidad y adaptabilidad, ya que los recursos pueden ser gestionados y escalados por el proveedor de la nube sin intervención directa del cliente.
- **Microservicios:** Los microservicios son una arquitectura esencial en el Cloud Native, donde las aplicaciones y procesos de negocio se dividen en pequeños servicios independientes que se comunican entre sí a través de APIs. Esto facilita la escalabilidad y el mantenimiento de las aplicaciones, permitiendo a las empresas adaptar rápidamente sus sistemas a las necesidades cambiantes del negocio.
- **Interfaces de Aplicaciones:** Las interfaces de aplicaciones, o APIs, permiten la interacción entre diferentes aplicaciones y servicios en la nube. En el contexto de Cloud Native, las APIs son cruciales para conectar microservicios y otros componentes de la infraestructura, asegurando una integración fluida y eficiente.



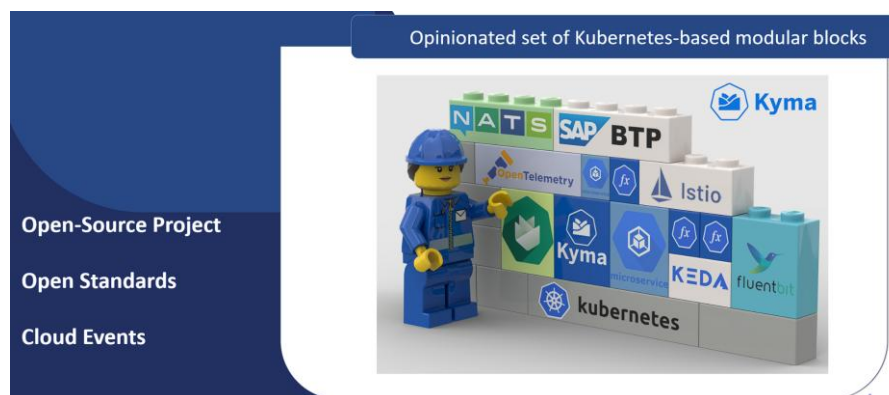
Pilares del BTP (Business Technology Platform):



Cloud Foundry: Es uno de los pilares del Business Technology Platform (BTP) de SAP. Es una plataforma de desarrollo para aplicaciones cloud nativas que ofrece un alto grado de disponibilidad, eficiencia, seguridad y mantenimiento. Para más información sobre Cloud Foundry, se puede acceder a cloudfoundry.org. SAP utiliza Cloud Foundry como una solución Open Source, contribuyendo activamente a su desarrollo y mantenimiento.



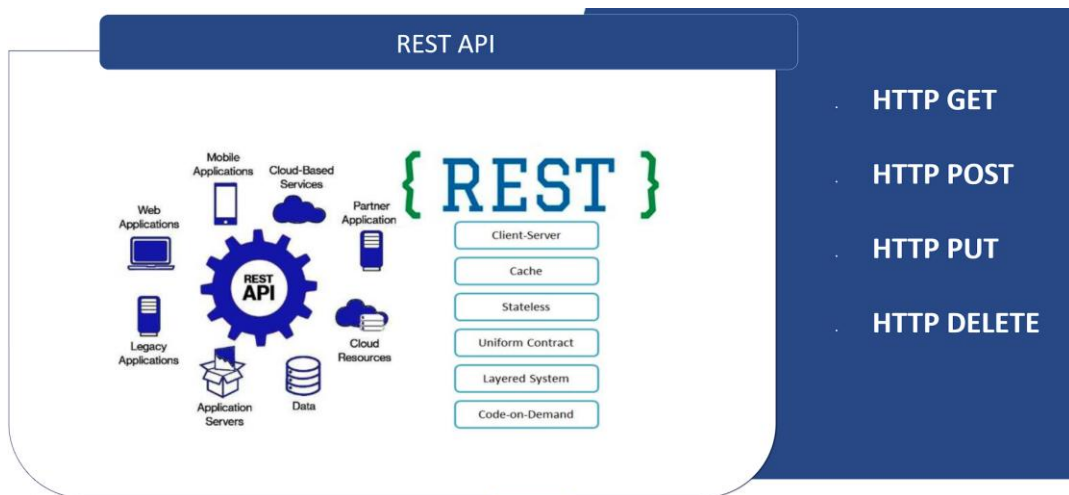
Kyma Project: Otro pilar del BTP es Kyma Project, una plataforma basada en Kubernetes, el orquestador de contenedores mantenido por Google. Kyma permite modularizar y trabajar de manera simple sobre Kubernetes, proporcionando un entorno de ejecución diferente disponible en BTP para la activación de los clientes. Para más información sobre Kyma Project, se puede visitar kyma-project.io. Al igual que con Cloud Foundry, SAP es un contribuidor activo en el desarrollo y mantenimiento de Kyma Project.





1.2. Arquitectura REST

REST, es el acrónimo de Representational State Transfer, es un estilo de arquitectura de software utilizado principalmente para la comunicación de APIs (Interfaces de Programación de Aplicaciones) en entornos distribuidos, como la World Wide Web. Este término fue acuñado en el año 2000 por Roy Fielding en su tesis doctoral sobre la web. REST se basa en un protocolo para la representación de los estados de los recursos, que son esencialmente los datos necesarios para la comunicación entre diferentes clientes, dispositivos y sistemas.



Principios Fundamentales de la Arquitectura REST

- **Interfaz Uniforme de Recursos:** Uno de los principios clave de REST es la interfaz uniforme de recursos, donde cada recurso se identifica de forma única a través de un mecanismo de direccionamiento. Esto generalmente implica un Host más una ruta relativa para acceder al servicio, conocido como el endpoint de la API. La uniformidad en la identificación de los recursos facilita su acceso y manipulación.
- **Conexión Cliente-Servidor:** REST adopta un enfoque de diseño de microservicios mediante una conexión cliente-servidor. En este contexto, el cliente es donde se ejecuta la aplicación de la interfaz de usuario, mientras que el servidor alberga la API publicada. Estas dos capas separadas se comunican entre sí a través de Internet utilizando el protocolo REST. Esta separación permite una mayor modularidad y escalabilidad en las aplicaciones.



- **Protocolo HTTP y Stateless:** REST utiliza el protocolo HTTP para la transferencia de información entre varios clientes y servidores. HTTP es un protocolo sin estado (stateless), lo que significa que cada solicitud del cliente al servidor debe contener toda la información necesaria para entender y procesar la petición. Esto asegura que las interacciones sean independientes y no se mantenga información de estado entre solicitudes.
- **Operaciones CRUD mediante HTTP:** REST proporciona un conjunto de operaciones estándar para manipular recursos, que se traducen en operaciones SQL:
 - **GET:** Se utiliza para leer o recuperar información de un recurso.
 - **POST:** Se usa para crear un nuevo recurso.
 - **PUT:** Se emplea para actualizar un recurso existente.
 - **DELETE:** Se utiliza para eliminar un recurso.

Estas operaciones permiten a los desarrolladores realizar las cuatro operaciones básicas de persistencia (Crear, Leer, Actualizar y Eliminar) de manera eficiente y estandarizada.

- **Navegación y Selección de Recursos:** REST también permite la navegación y selección de múltiples recursos mediante la utilización de modelos de programación como el RAP (Restful Application Programming). Este enfoque facilita el acceso y manipulación de datos a través de las APIs, mejorando la flexibilidad y eficiencia en la gestión de los recursos.

Aplicaciones y Transformación en Sistemas SAP:

En el contexto de SAP, REST se ha convertido en una de las arquitecturas de APIs más populares, especialmente en la nube. Se utiliza ampliamente en sistemas S/4HANA, tanto en ediciones privadas como públicas, así como en entornos ABAP en BTP (Business Technology Platform). La transformación y actualización de los procesos de negocio en SAP se basan en gran medida en aplicaciones RAP Restful, que utilizan este protocolo para la comunicación eficiente y efectiva.



La arquitectura REST ofrece un marco robusto y flexible para la comunicación entre sistemas distribuidos, aprovechando principios como la interfaz uniforme de recursos, la conexión cliente-servidor, el protocolo HTTP sin estado y las operaciones CRUD estándar. Estos principios permiten la creación de aplicaciones escalables, modulares y fáciles de mantener, haciendo de REST una opción popular para el desarrollo de APIs en la nube y en sistemas empresariales como SAP. La adopción de REST en entornos SAP refleja su eficacia y capacidad para soportar la transformación digital y la modernización de los procesos de negocio.

1.3. Explorando ABAP Cloud

Con la llegada de REST y el uso del Cloud nativo, el entorno ABAP ha tenido que aplicar cambios para aprovechar el nuevo enfoque basado en la nube. Esta necesidad se ha manifestado en el modelo de desarrollo ABAP Cloud, que integra diversos aspectos y componentes básicos esenciales para un entorno nativo en la nube. ABAP Cloud ofrece todas las herramientas que un desarrollador necesita para trabajar en este nuevo paradigma.

Principales Características de ABAP Cloud:

- **ABAP for Cloud Development:** Es un lenguaje nativo en la nube, optimizado para este entorno. Se enfoca en modelos de programación nativos en la nube, como el modelo de programación de aplicaciones ABAP Restful (RAP) y el análisis de ABAP, que deben seguirse en la nube. Este lenguaje proporciona una versión adaptada y optimizada para el desarrollo en la nube, permitiendo a los desarrolladores crear aplicaciones más eficientes y escalables.
- **Plataforma como Servicio (PaaS):** ABAP Cloud también incorpora características de una plataforma como servicio (PaaS), ofreciendo servicios de reutilización, gestión de identidad y acceso, y conectividad. Esta plataforma facilita el desarrollo y despliegue de aplicaciones en la nube, proporcionando un entorno completo y cohesionado para los desarrolladores.

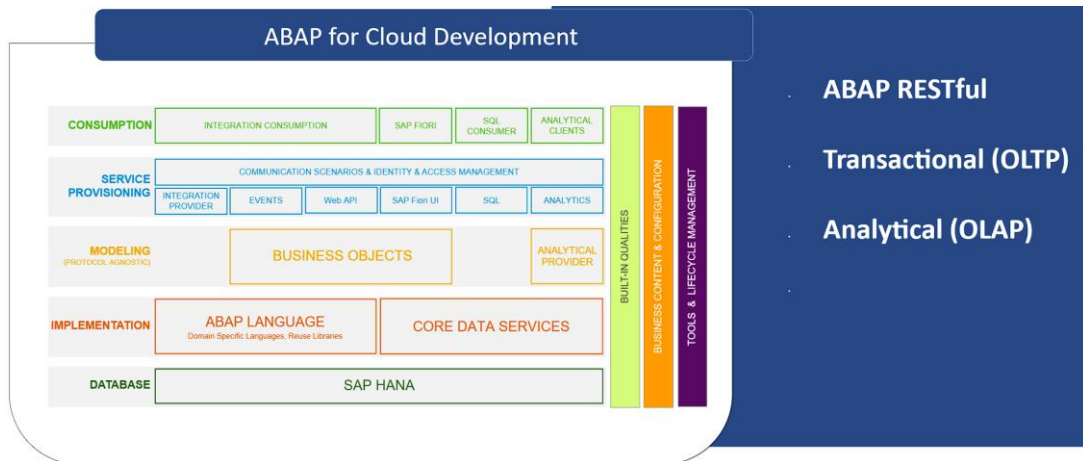


- **Funciones Integradas en SAP S/4HANA Cloud:** Dentro de SAP S/4HANA Cloud, ABAP Cloud integra funciones como APIs empresariales, puntos de extensión y eventos. Esto permite a los desarrolladores conectar y extender las funcionalidades de SAP S/4HANA Cloud, facilitando la integración de diferentes servicios y aplicaciones.
- **Herramientas de Desarrollo:** Las herramientas de desarrollo en ABAP Cloud se incluyen en el editor de código Eclipse con el ADT (ABAP Development Tools). Estas herramientas proporcionan un entorno robusto y eficiente para el desarrollo de aplicaciones en ABAP Cloud, permitiendo a los desarrolladores trabajar de manera más productiva.

Modelos de Programación en ABAP Cloud:

En el núcleo del modelo de desarrollo ABAP Cloud se encuentran los modelos de programación que definen la arquitectura de diseño y ejecución de todas las extensiones, servicios y aplicaciones. Generalmente, se consideran tres casos de uso o escenarios diferentes:

- **Transaccional (OLTP):** ABAP Cloud permite crear objetos de negocio y exponerlos como servicios para ser consumidos en interfaces de usuario y escenarios de integración. Esto incluye la implementación de las operaciones estándar de creación, lectura, actualización y eliminación (CRUD) mediante el modelo de programación de aplicaciones ABAP Restful (RAP).
- **Analítica (OLAP):** ABAP Cloud también está equipado para crear servicios e interfaces de usuario para el análisis de datos en múltiples dimensiones. Este enfoque se integra principalmente con SAP Analytics Cloud (SAC) y se basa en la capa analítica de ABAP propagada a la base de datos HANA.
- **Integración:** Ambos aspectos anteriores se complementan con una fuente de integración de datos y aplicaciones, lo que permite dar respuesta a entornos orientados a servicios en la nube. Esto facilita la integración de diferentes servicios y aplicaciones, proporcionando una solución completa y cohesiva.

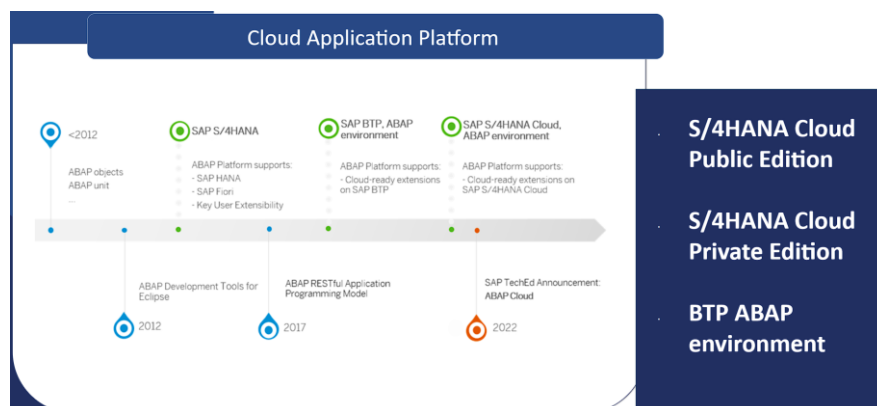


Historia e Innovaciones en ABAP Cloud

El enfoque de ABAP Cloud es el resultado de una serie de innovaciones a lo largo de los años. Algunas de las principales etapas en su desarrollo incluyen:

- **2012:** Introducción de herramientas de desarrollo ABAP para Eclipse.
- **2015:** Soporte para SAP S/4HANA, incluyendo Fiori Cloud y Key User Extensibility.
- **2018:** Presentación de ABAP en BTP (Business Technology Platform), con diversas extensiones disponibles.
- **2022:** Lanzamiento de un entorno nativo en ABAP Cloud, integrando instancias públicas y privadas de SAP S/4HANA y el entorno ABAP en BTP.

Estas innovaciones, cada una con su propio valor, se han sumado para formar el entorno robusto y avanzado que es ABAP Cloud en la actualidad.





1.4. ABAP Cloud – Artefactos RAP

ABAP Cloud es una plataforma que incluye varias herramientas, tecnologías y componentes de integración para correos y comunicación, haciendo el entorno de ejecución más completo y versátil. Este entorno está basado en el modelo de programación RAP (RESTful ABAP Programming) que está compuesto por cuatro pilares fundamentales: Data Access, Domain Model and Implementation, Business Services y Business Services Consumption.

Aspectos Importantes:

- **Data Access:** ABAP Cloud permite el acceso y almacenamiento de datos en tablas de persistencia utilizando almacenamiento en columnas y tecnologías de acceso en tiempo real. Esto facilita el trabajo con modelos de datos que integran Machine Learning y otras tecnologías avanzadas. El primer pilar, Data Access, es esencial para la estructura del modelo RAP, permitiendo una gestión eficiente y segura de los datos.

- **Domain Model and Implementation (Business Object):** Aquí se modelan y relacionan las diferentes entidades de negocio como facturas, pedidos y órdenes de compra. Se utilizan múltiples lenguajes y tecnologías como:

- TDL (Type Definition Language).
- DDL (Data Definition Language).
- SDL (Service Definition Language).
- DCL (Data Control Language).

Estos lenguajes ayudan en la definición de datos, servicios, controles de acceso y anotaciones. Este pilar permite una integración fluida y coherente de los objetos de negocio, facilitando operaciones como facturación, pedidos y órdenes de compra.

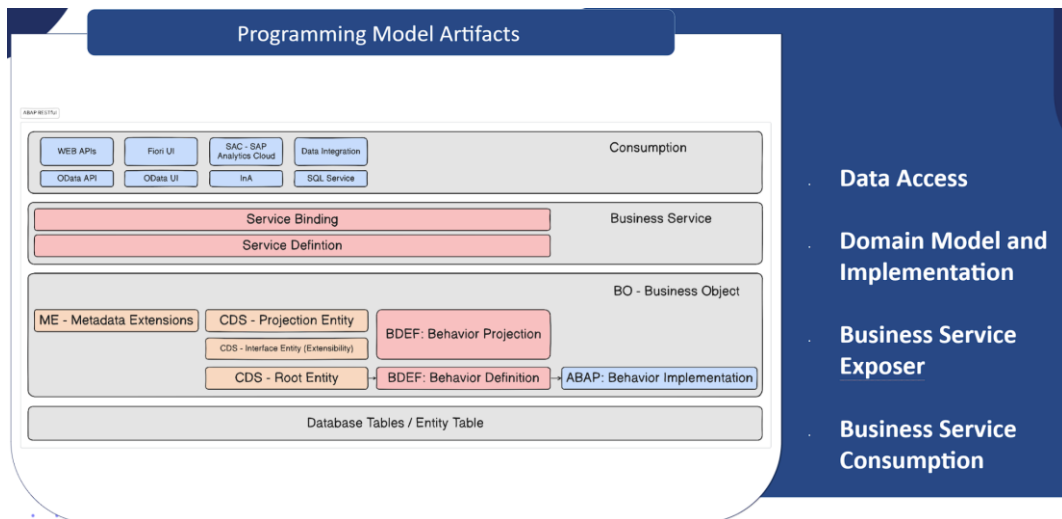
- **Business Service Exposer:** En este pilar se expone la lógica de negocio a través de servicios utilizando protocolos como OData para APIs y la interfaz de usuario. Se mencionan herramientas como:

- Service Definition (SDL)



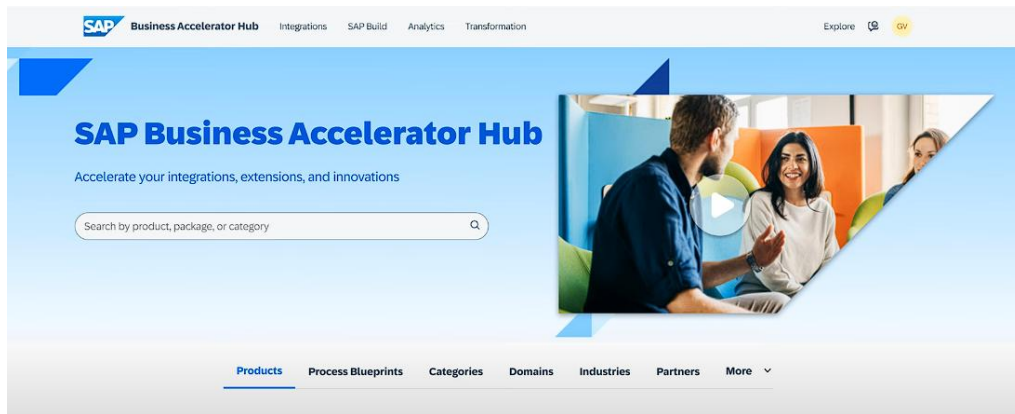
- Service Binding: Vincula el protocolo a utilizar para la exposición del servicio.
- Metadata Extension y User Interface Annotations: Ayudan a crear interfaces de usuario desde el backend.

● **Business Services Consumption:** El consumo de servicios expuestos es el último pilar del modelo RAP. Protocolos como web APIs y OData APIs permiten la interacción con aplicaciones e interfaces de usuario como Fiori. La capa analítica de la base de datos HANA proporciona capacidades avanzadas para la integración y análisis de datos, mejorando la funcionalidad y eficiencia de las aplicaciones estándar en entornos S4HANA, tanto públicos como privados.



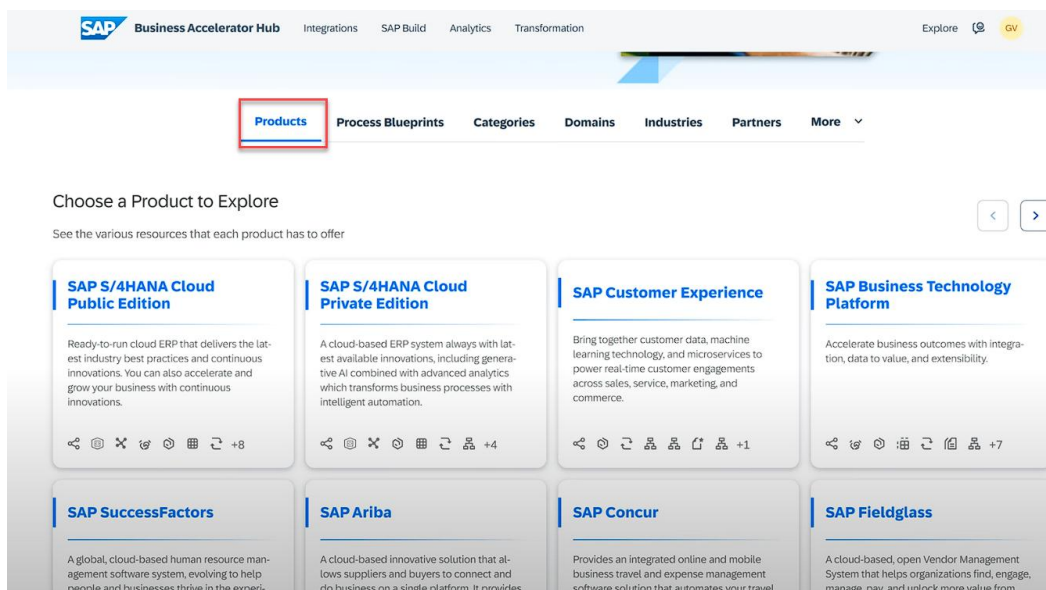
1.5. Exploración de las API publicadas

La plataforma SAP Business Accelerator Hub es la página oficial de SAP donde se puede encontrar información sobre las diferentes APIs liberadas. Estas APIs son esenciales para los proyectos empresariales y se utilizan en diversos aspectos como integración, extensibilidad y objetos de tipo OData y CDs. El Business Accelerator Hub se actualiza constantemente para proporcionar la información más reciente y relevante. Se puede acceder a la página por medio del siguiente enlace: hub.sap.com.

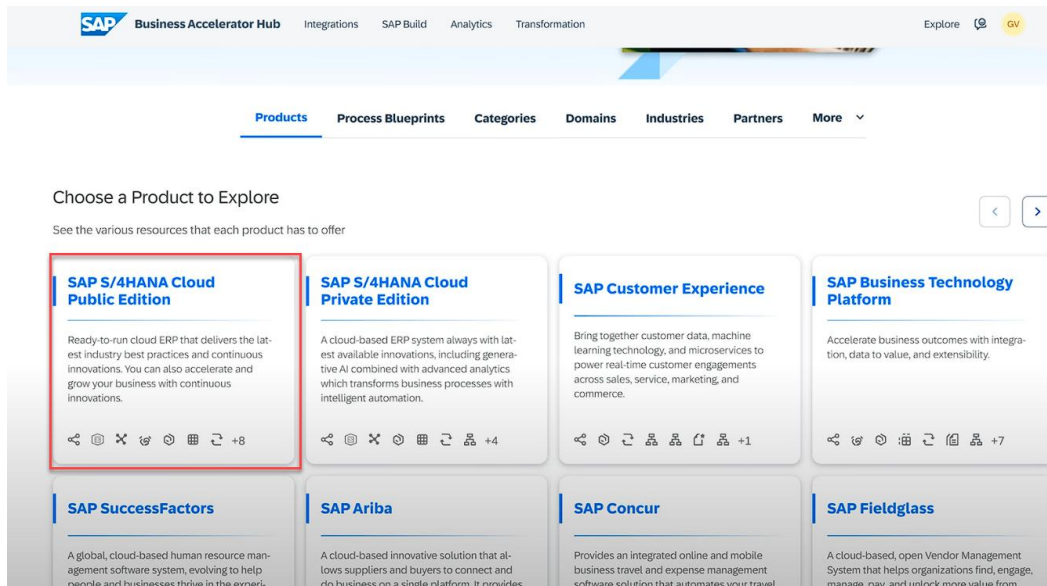


Aspectos Importantes:

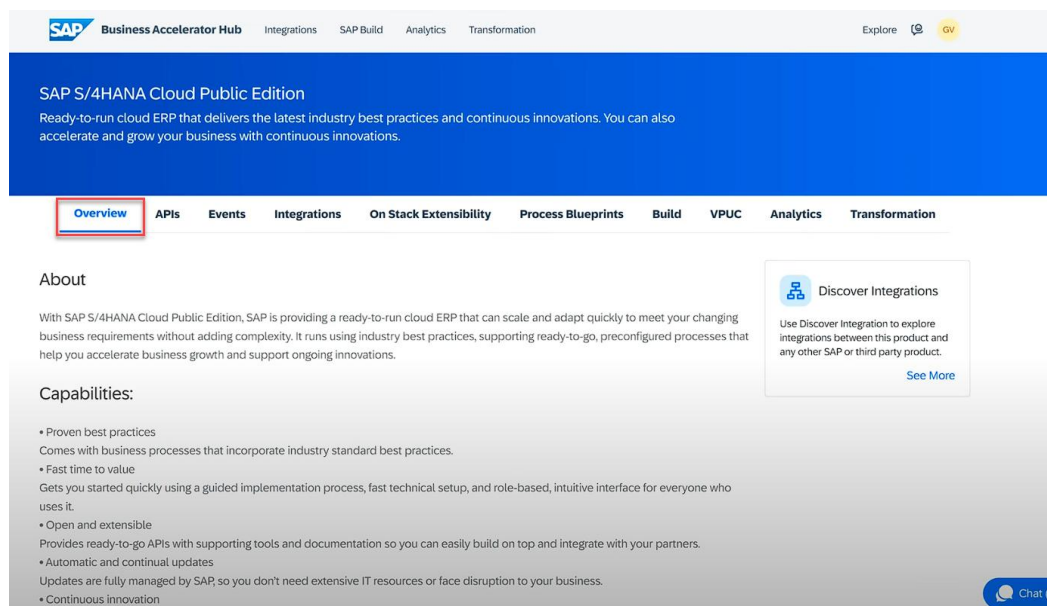
Publicación de APIs: La publicación de APIs se refiere a la liberación de los objetos y servicios que deben ser utilizados en el desarrollo de proyectos en ABAP Cloud. Esta información se encuentra organizada por categorías y se actualiza regularmente para garantizar que los desarrolladores tengan acceso a los recursos necesarios.



Exploración de APIs: Al seleccionar un producto específico, como S/4HANA Cloud Public Edition.



Los usuarios pueden explorar las APIs disponibles mediante una pestaña de "overview".



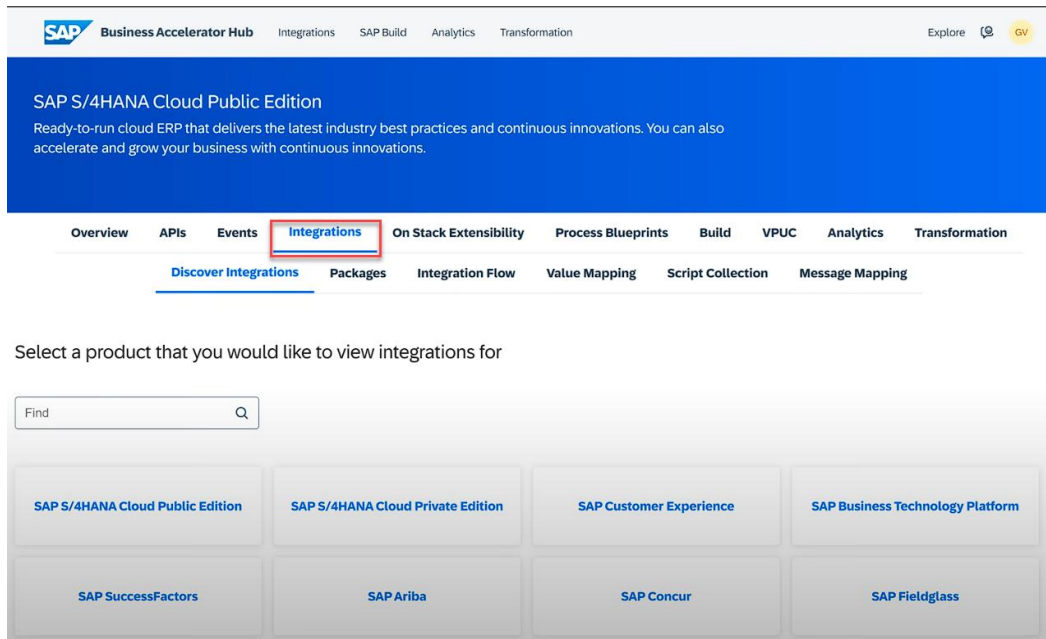
Exploración de APIs y Eventos: La plataforma permite a los usuarios explorar APIs y eventos utilizando diversas categorías y campos de búsqueda. Esto incluye la posibilidad de buscar APIs SOAP, OData (versión 2 y 4), REST, así como objetos de negocio específicos y eventos. La interfaz del Hub está diseñada para proporcionar una navegación intuitiva y eficiente, facilitando el acceso a la información necesaria para el desarrollo y la integración.



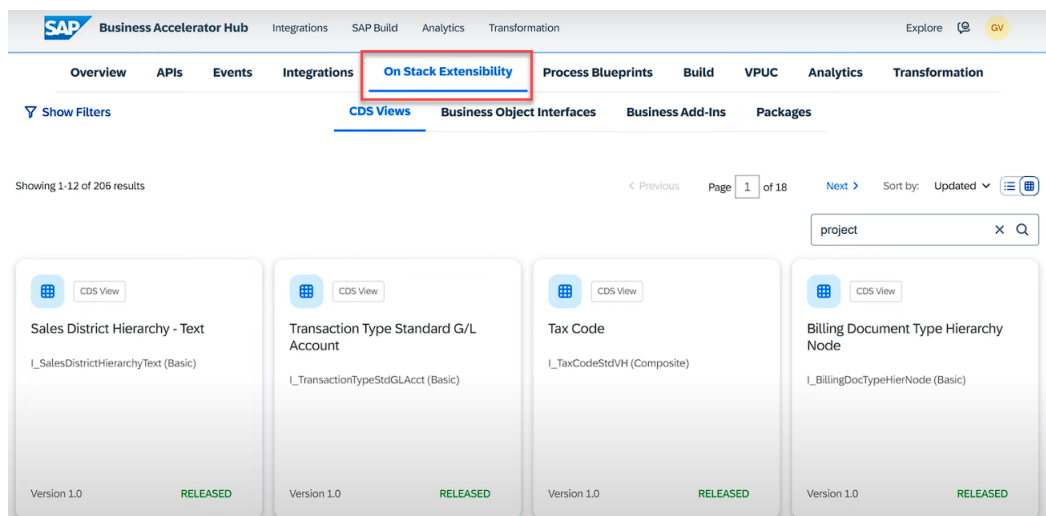
The screenshot shows the SAP Business Accelerator Hub interface. The top navigation bar includes 'Integrations', 'SAP Build', 'Analytics', and 'Transformation'. The main header area displays 'SAP S/4HANA Cloud Public Edition' with a description. Below this, a horizontal menu has 'APIs' highlighted with a red box. Under 'APIs', there are sub-tabs: 'Packages', 'All', 'SOAP', 'ODATA V2', 'ODATA V4', and 'REST'. The 'Packages' sub-tab is active. The main content area shows a list of API packages, including 'SAP Ariba Central Invoice Management', 'SAP Watch List Screening', 'SAP S/4HANA Cloud Public Edition', and 'SAP S/4HANA Cloud Public Edition for ABAP Platform'. A search bar is visible on the right.

The screenshot shows the SAP Business Accelerator Hub interface with the 'Events' section highlighted in the main menu with a red box. Under 'Events', there are sub-tabs: 'Event Objects' and 'Packages'. The 'Event Objects' sub-tab is active. The main content area displays a list of event objects, including 'Service Contract Events', 'Field Logistics Received Handling Unit Events', 'Chemical Compliance Info Events', and 'Freight Booking Events'. A search bar is visible on the right.

Capa de Integración: La plataforma permite descubrir integraciones disponibles por tipo de producto, paquetes, flujos de integración, mapping value, scripting y mapeo de mensajes. Esta capa es esencial para gestionar la integración de diferentes sistemas y servicios en el entorno SAP.

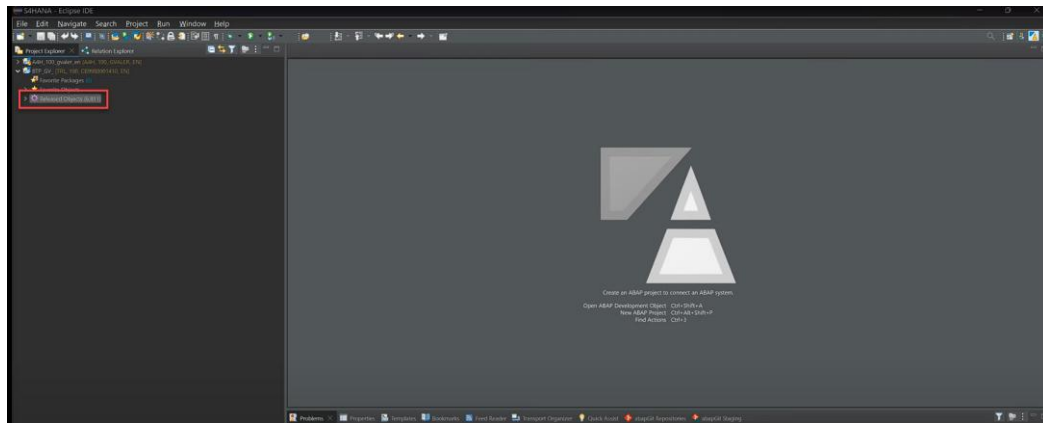


On-Stack Extensibility: Esta característica permite a los usuarios explorar los puntos de extensión disponibles en la pila de S/4HANA. Se pueden utilizar objetos de negocio e interfaces liberadas por SAP para implementar extensiones y mejoras en el sistema.

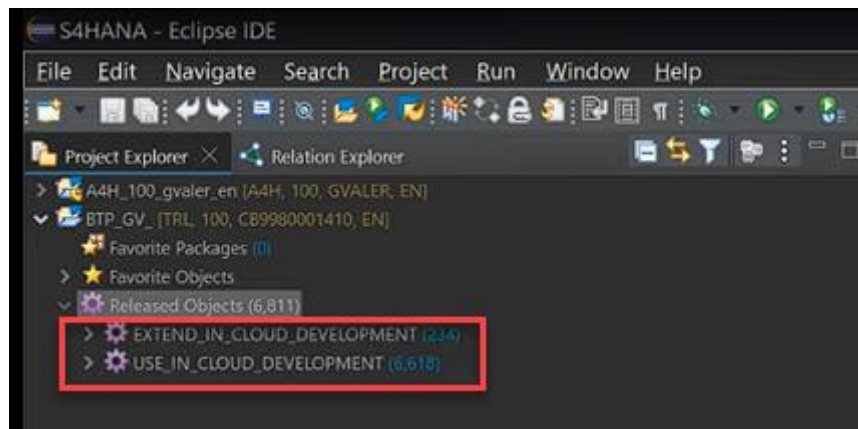


Exploración de APIS por medio de Eclipse:

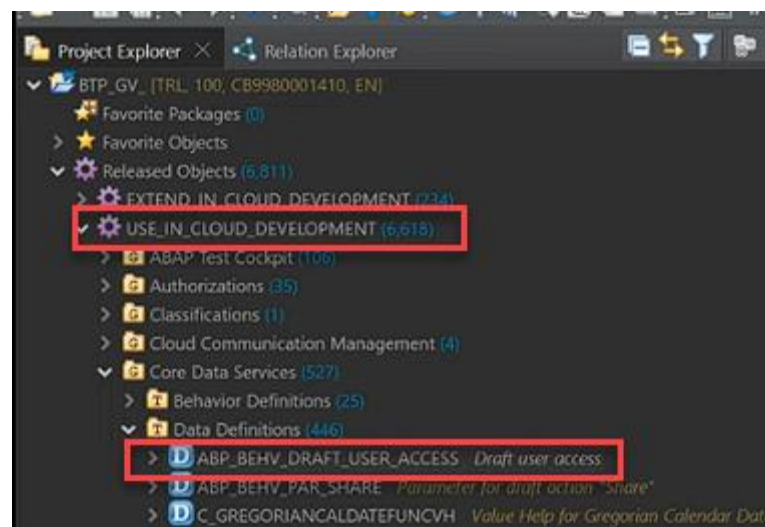
Cuando hablamos de la exploración dentro del sistema S/4HANA, o en un sistema BTP del tipo ABAP, se realiza por medio de la sección objeto liberado (Release Object), en el Editor de código Eclipse.



Al seleccionar la sección “Release Object”, disponemos de categorías para los objetos liberados. Es crucial entender qué objetos se deben utilizar en el desarrollo en la nube, ya sean Core Data Services, definiciones de datos (Data Definitions) u otras carpetas.

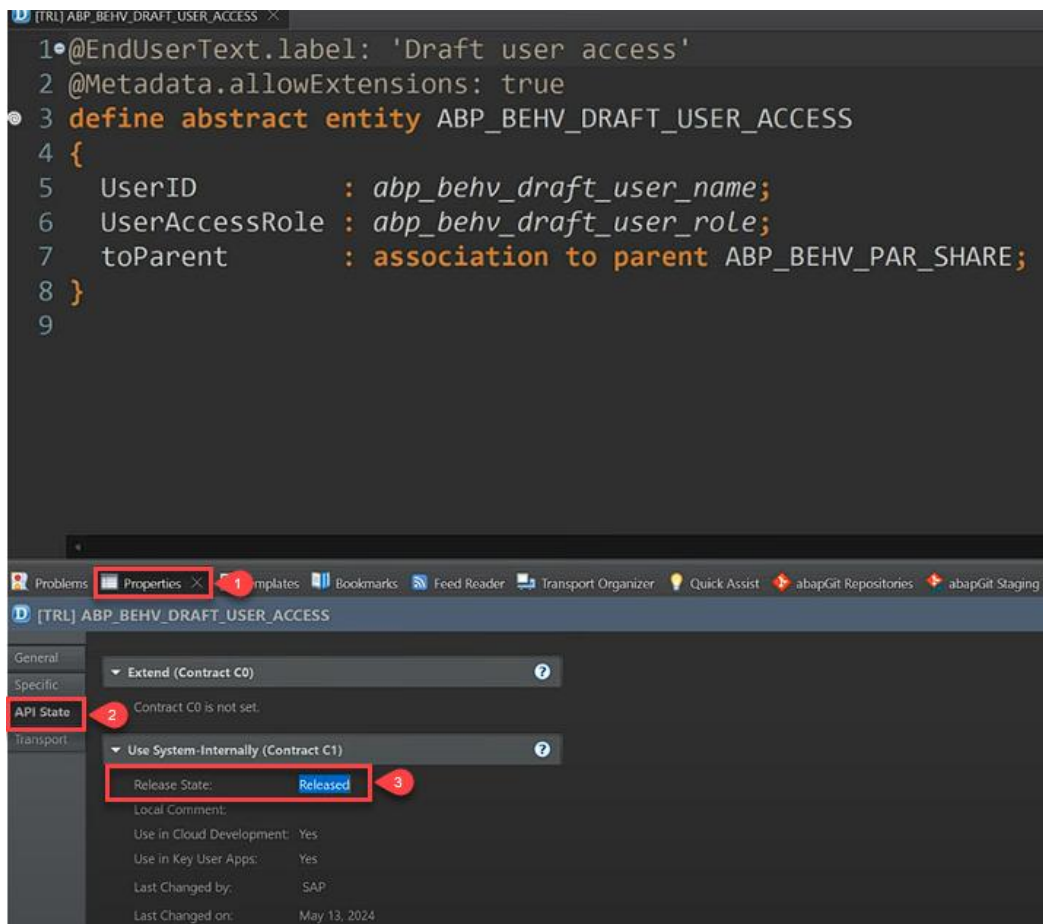


Por ejemplo, podemos seleccionar un artefacto que sea una CDS de tipo abstracto (Abstract Entity). Dentro de la sección “USE_IN_CLOUD_DEVELOPMENT”.

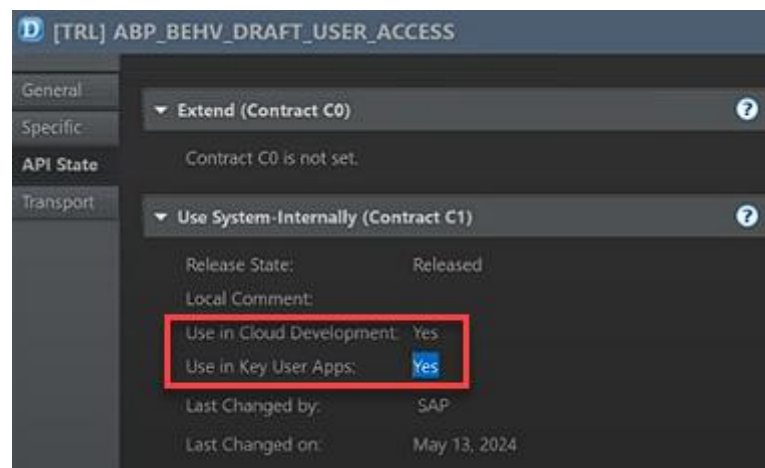




La parte importante a observar que en la vista de propiedades, específicamente en la pestaña "API State" o "Estado de la API", donde se muestra si el estado es "Released" o liberado.



Este objeto puede utilizarse tanto en el desarrollo en la nube como en extensiones de tipo Key User. Los objetos liberados por SAP son esenciales para mantener el Clean Core, asegurando un sistema limpio y optimizado en los desarrollos.





SAP se compromete a liberar las APIs que deben ser utilizadas, mientras que el cliente se compromete a utilizar estas APIs liberadas para mantener un sistema limpio. Este enfoque facilita las actualizaciones y upgrades, y es aplicable tanto en las ediciones Private Edition de S/4HANA como en las ediciones on-premise, promoviendo una transformación tecnológica eficiente y ordenada.