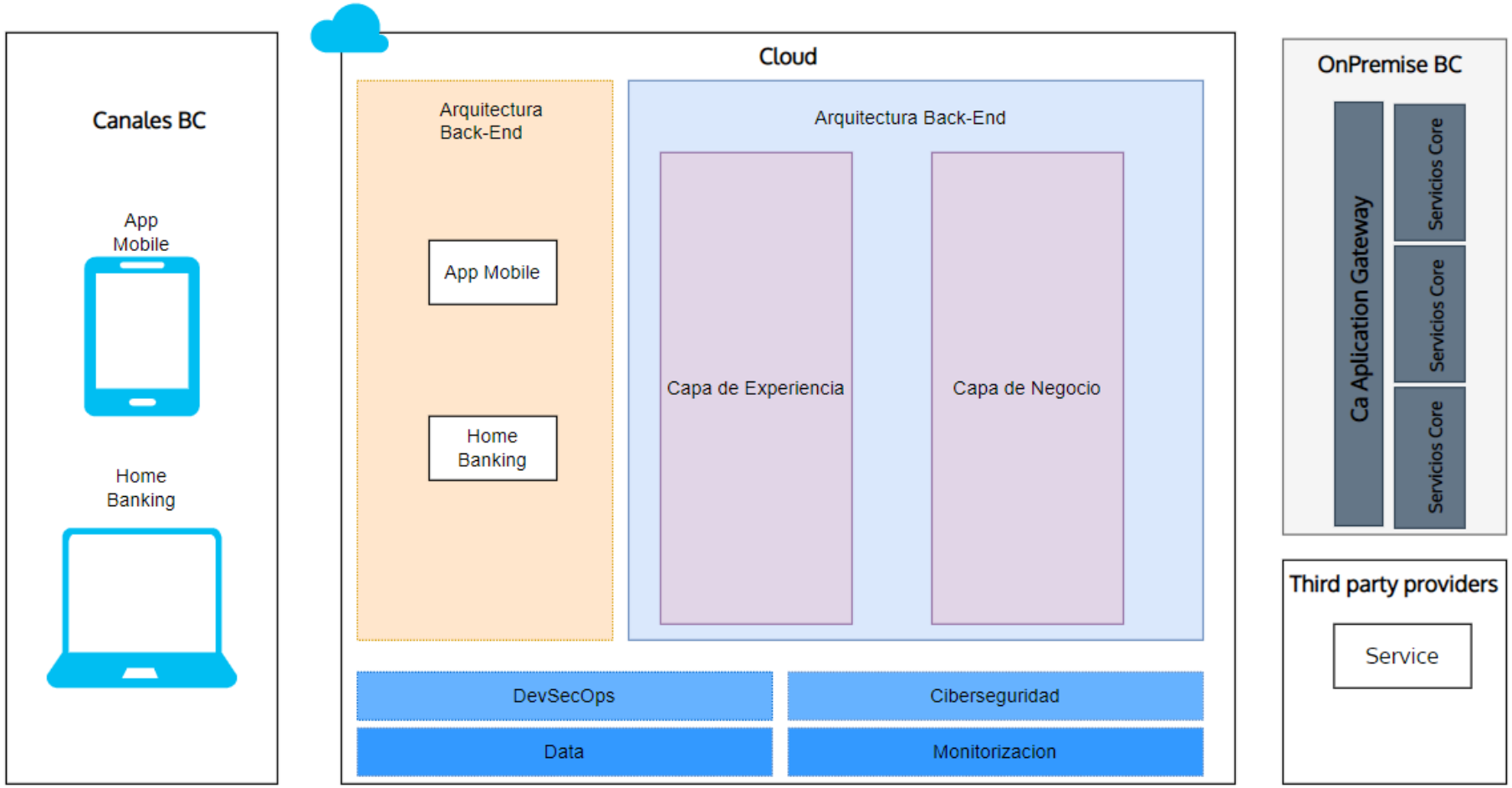
Reto técnico BP

Propuesta de Arquitectura.

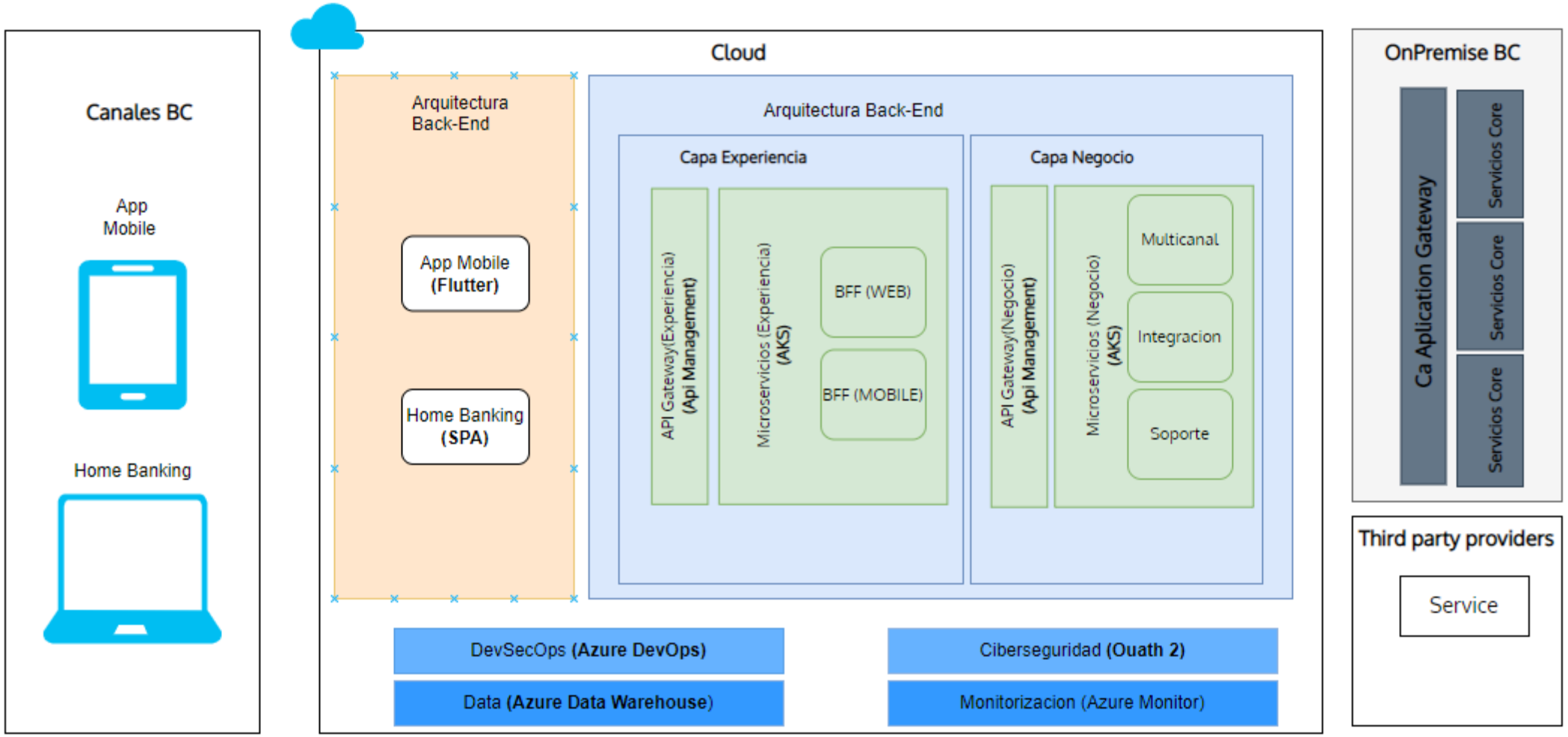
Para la implementación de los canales de App Mobile y Home Banking

El diagrama de contexto del sistema

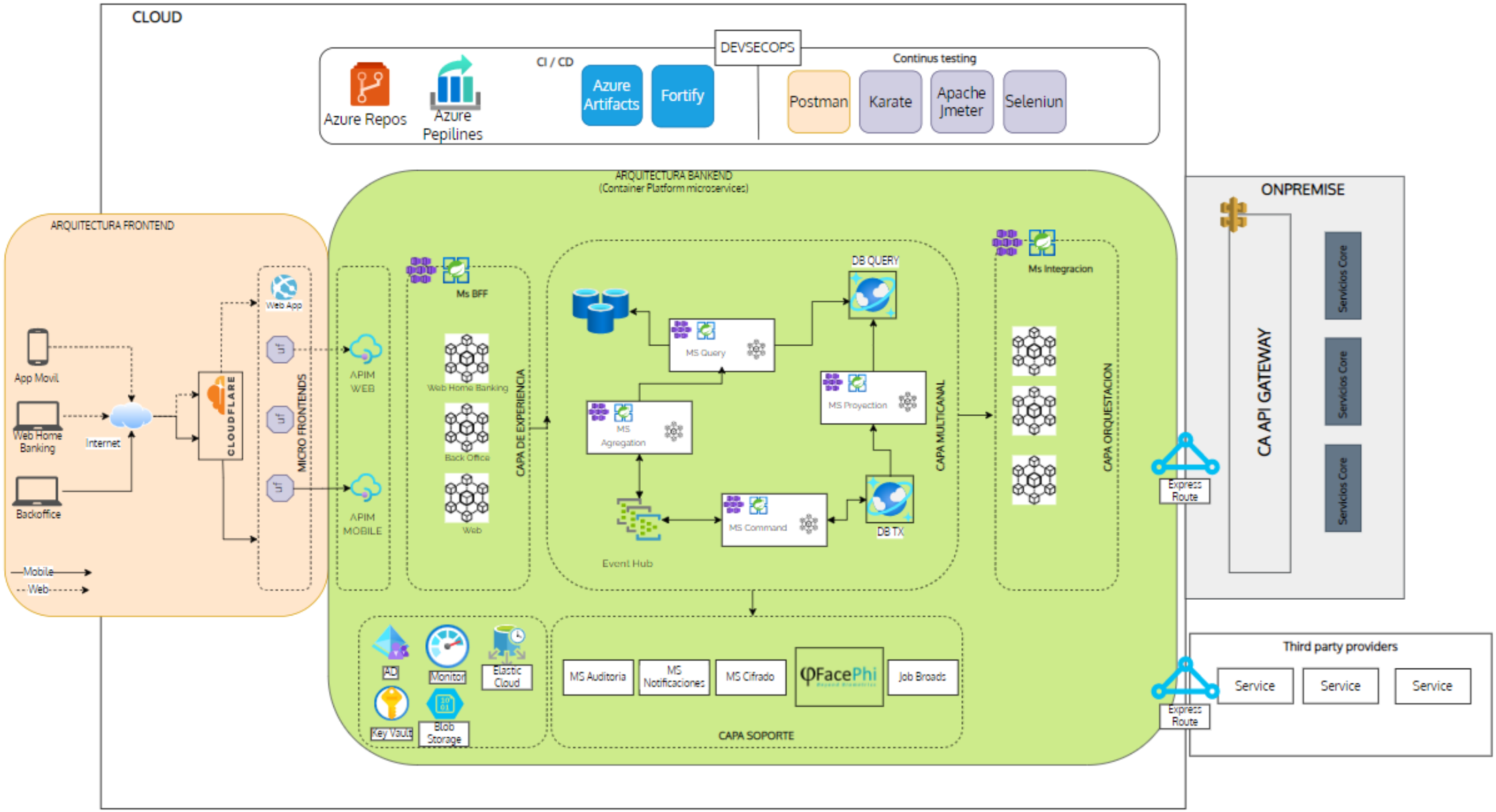
La solución contempla una arquitectura mobile, frontend y arquitectura backend bajo los estándares cloud de Azure permiten la omnicanalidad y la reutilización de componentes entre los diferentes canales. Esta solución plantea un desacoplamiento frontend web en microfronts y desacoplamiento backend en capa de experiencia, negocio.



El diagrama de contenedores del sistema



El diagrama de arquitectura de tecnología



Arquitectura Mobile (Framework crossplatform)

Para el presente proyecto, **se** propone utilizar un *framework crossplatform* de desarrollo de aplicaciones móviles que permita asegurar el ***time to market,*** **eficiencia en los desarrollos** evolutivos y a su vez contar con capacidades nativas de los dispositivos manteniendo una **experiencia diferenciada** para los usuarios.
Se propone dos framework para esta necesidad, **React Native** y ***Flutter*** sin embargo, finalmente recomiendo el uso de ***Flutter*** por los siguientes motivos:

DESARROLLOS ACELERADOS

Uso de una misma línea de código para el desarrollo de aplicaciones utilizando mismo set de librerías para las distintas plataformas (iOS, Android y Huawei).
Flutter permite el reuso del 75% del código hecho en Dart entre iOS y Android en las capas de UI, dominio, comunicación y acceso a datos. Esto a diferencia de otras plataformas crossplatform donde la reutilización promedio es del 65%.

ALTO DESEMPEÑO

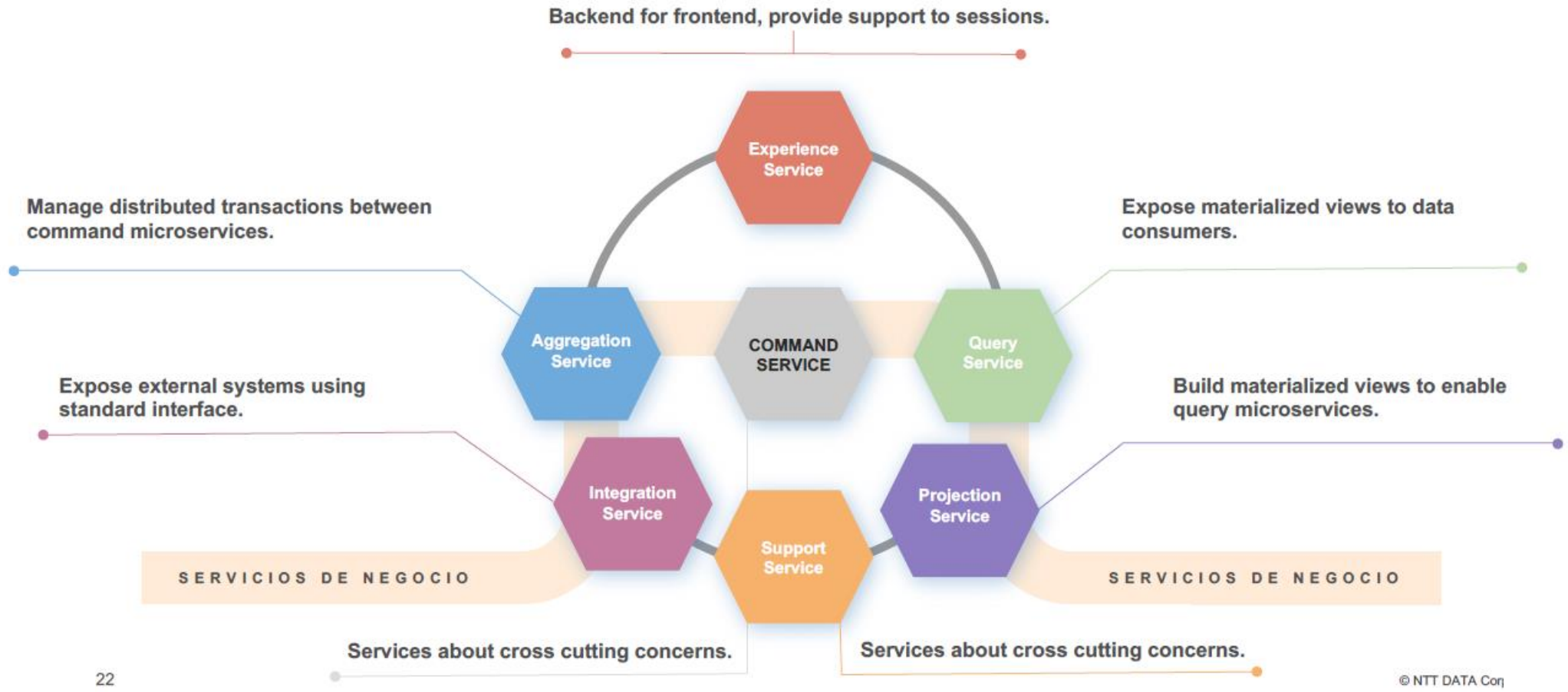
Dart es un lenguaje de programación orientado a objetos que usa técnicas de compilación *Ahead of Time* (AOT), el Código dart es compilado en plataforma nativa. Este enfoque acelera drásticamente el inicio de la aplicación y maximiza la performance al punto de compararse con aplicaciones nativas.

OTRAS VENTAJAS

- ✓ Capacidad de insertar código nativo sin tener restricción del user experience.
- ✓ Capacidad de implementar funcionalidades «hot-reload» que se puedan habilitar sin necesidad de un nuevo despliegue en el Store (App Bundle).
- ✓ Permite implementar interfaces con animaciones y transiciones detalladas sin comprometer el desempeño de la aplicación.
- ✓ Posibilidad de integrar con cualquier SDK nativo.

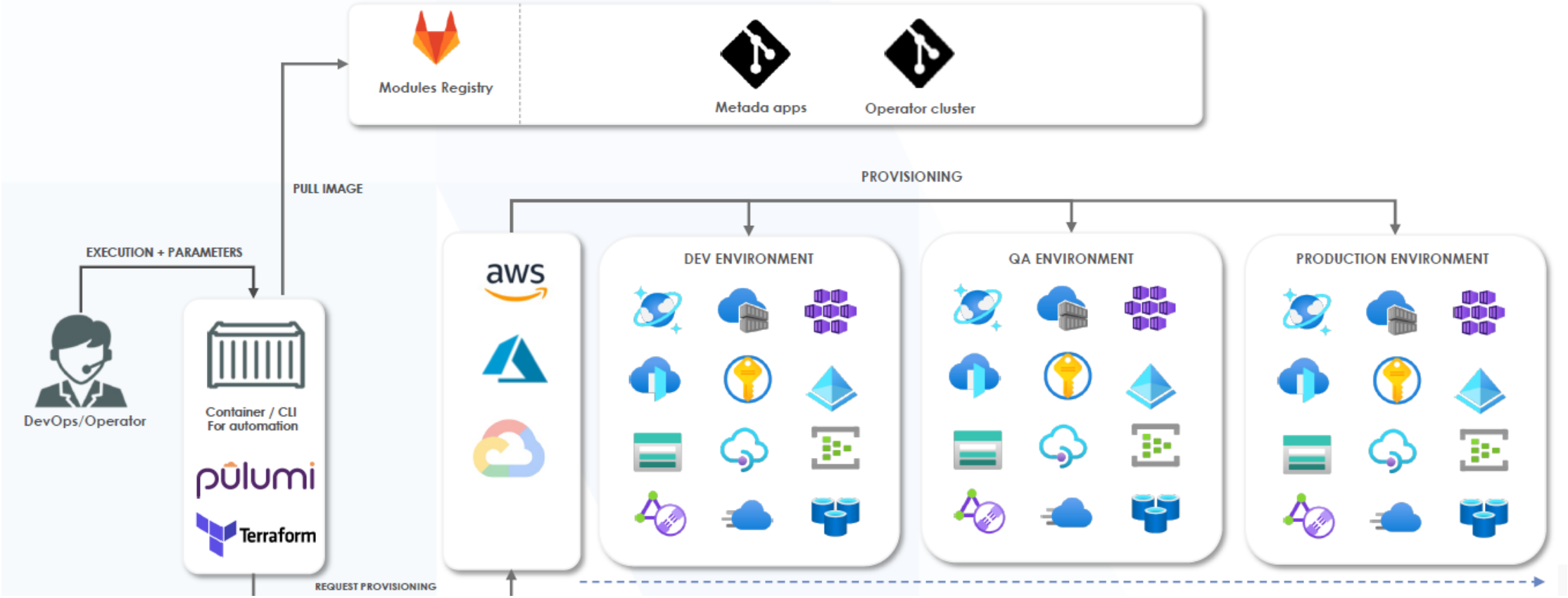
Arquitectura Backend Event-Driven Architecture

En este nuevo enfoque aparecen servicio más desacoplados con responsabilidades atómicas. Patrón CQRS.



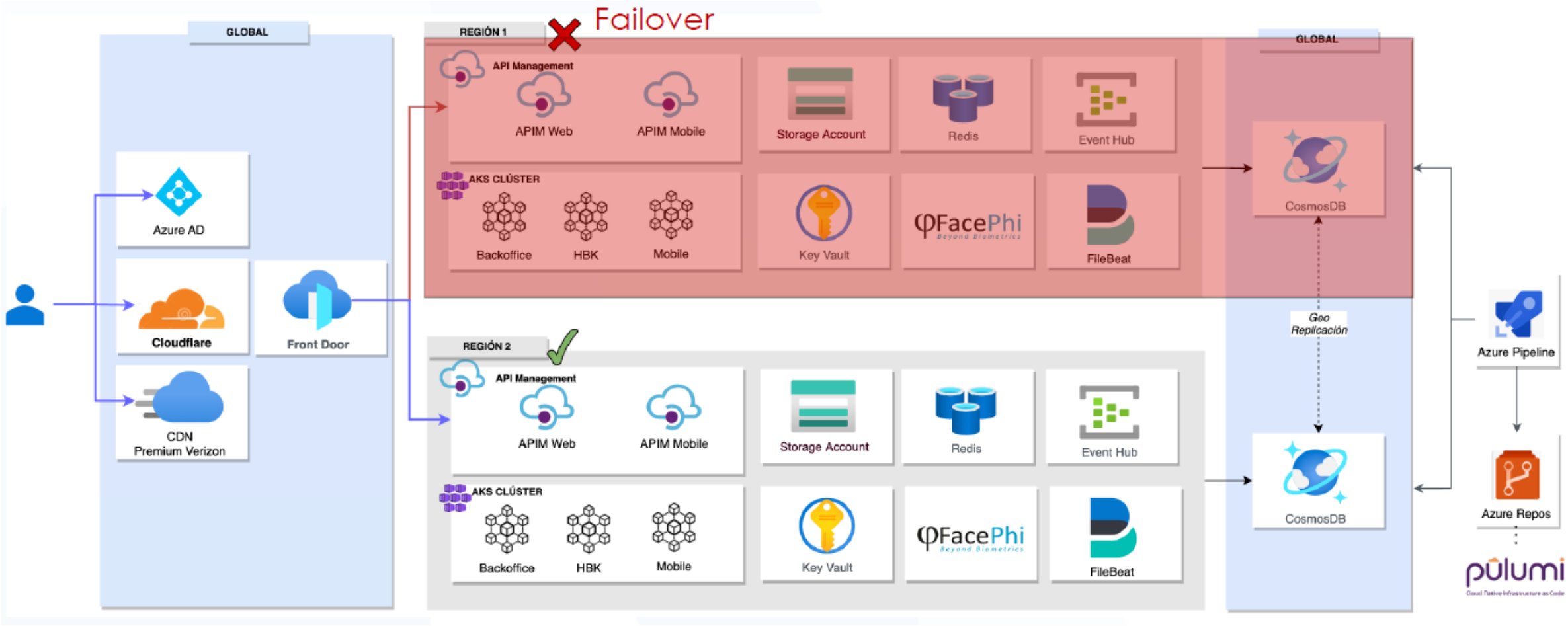
Plan de Continuidad de Negocio (BCP) & Infraestructura como código

Recomendamos un **enfoque end to end** para el gobierno de aprovisionamiento y del ciclo de vida de desarrollo de software
y de las aplicaciones.



Disaster Recovery

Con el objetivo de **maximizar la disponibilidad de las aplicaciones** se adoptan servicios de **replicación al nivel Global** y así mantener la **continuidad del negocio** del Banco. En materialice una caída (Failover) de un componente o la pérdida total de la región se contempla un nuevo aprovisionamiento usando IaC.

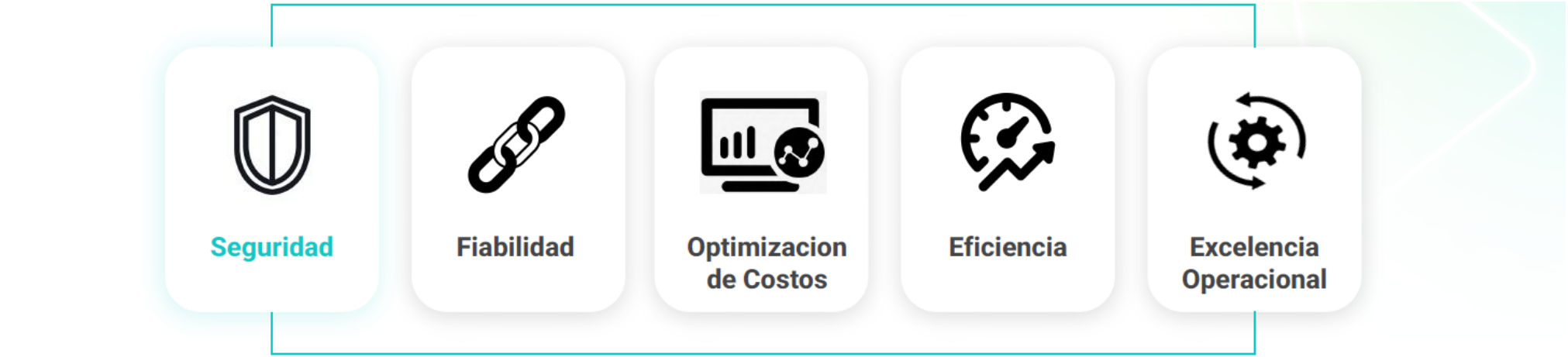


Monitoreo & Logs

ACTIVIDADES PRINCIPALES

- Discovery de monitoreo actual on-prem.
- Definición de métricas, indicadores, alertas y notificaciones.
- Configuración e integración con Azure Monitor para todos los ambientes.
- Implementación de SLA/SLI en concordancia con lo ya definido.
- Implementación de umbrales.
- Desarrollo de esquema de alertas y acciones (basado en políticas de escalamiento)
- Desarrollo de marco de trabajo de monitoreo.

Seguridad de Información y Ciberseguridad - Entornos Cloud



Identidad
- IAM
- User Policy
- PIM
- Compliance

Redes
- Firewall
- Security Groups
- VPNs
- CASB

Aplicaciones
- DOS/ WAF
- DLP
- Web Scanner
- Containers
- DBs
- Encryption

Deteccion
- Threat Detection
- Event Correlation
- EDR/XDR
- Forensics

Continuidad
- Multi Region / Zonas
- Balanceo Global

Soluciones de Cacheo


Gracias
T S B
login
T S B
login
T S B
login