

Diagnóstico SOA Actual FNA

Documentación del proyecto 181-2020, E-Service, ([permalink](#)) versión indicada a continuación, se encuentra en [6ddfe81](#) on January 31, 2023.

Grupo E-Service

- **Harry Wong, ing.**
· GitHub icon [e_hwong](#)
Arquitecto SOA, Stefanini
- **Eddie Hernandez, ing.**
· GitHub icon [e_ehernandez](#)
Datos, Stefanini
- **Federico Suárez, ing.**
· GitHub icon [e_fsuarez](#)
Infraestructura, Stefanini
- **Darío Correal, ing.**
· GitHub icon [e_dcorreal](#)
Arquitecto TI, Stefanini

Fase 2: SOA Objetivo FNA

Contenido de los Productos Contractuales, 181-2020

Producto 5: PR5. Comparativa de la industria y el FNA

Describir y comparar con el Fondo la manera cómo las empresas del segmento y de otros sectores de la industria enfrentan los retos de flexibilidad de negocio, independencia de proveedor tecnológico, y fortaleza SOA.

Nota: los análisis de este producto provienen únicamente de los objetivos del proyecto SOA: dependencia de proveedor (OBJ1), fortaleza SOA de las aplicaciones (OBJ2), y el tiempo de mercado (OBJ3).

Justificación

La comparativa SOA de la industria proveerá al Fondo de hallazgos aplicables a sus procesos y productos. A la vez, esta información de guía puede ser traducida en futuras iniciativas pertenecientes a la hoja de ruta de mejora, tanto del gobierno como de los procesos SOA del Fondo.

Contenidos

1. Arquitectura SOA
2. Portafolio de servicios
3. Gobierno SOA
4. Versionamiento de servicios
5. Monitoreo y operación SOA
6. Mantenimiento y evolución de servicios
7. Nuevas tecnologías

Criterios de Aceptación

- Lista de oportunidades e iniciativas ralizables por el FNA
- Mejoras visibles para las capacidades de la arquitectura SOA del Fondo

Tema	Comparativa de la industria y el FNA: Comparación SOA FNA - Sector
Palabras clave	SOA, Situación actual, Comparativa
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	N003a Vista Segmento SOA FNA

Comparación SOA FNA - Sector

Para presentar este análisis comparativo hemos seleccionado algunas dimensiones que facilitan la presentación de la comparación en categorías.

Los aspectos a tener en cuenta en la comparativa serán:

1. Arquitectura SOA
2. Portafolio de Servicios
3. Gobierno SOA
4. Estrategia de Versionamiento
5. Monitoreo y Operación
6. Mantenimiento y Evolución
7. Nuevas tecnologías

Presentaremos cada uno de estos aspectos desde el sector financiero en general y; posteriormente serán aplicados al FNA.

Arquitectura SOA

Descripción

En esta categoría comparativa establecemos como criterios las prácticas asociadas al manejo de los modelos de arquitectura SOA. En este se encuentran consideraciones como si se cuenta con un depósito de arquitectura formal y un grupo de gobierno y desarrollo de la Arquitectura SOA dentro de la organización. Esto último incluye a los cargos y el nivel de formalismo en realizar y mantener las arquitectura SOA en el Fondo.

Prácticas del Sector Financiero

Depósito de arquitectura Análisis de modelos de arquitectura Herramientas de modelado de arquitecturas de servicios

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Depósito de arquitectura	Mega y Archimate		- Herramientas de AE como habilitadores, sin embargo documentando lo necesario bajo enfoques de tipo: MVA (Minimum Viable Architecture) y JIT (Just In Time Architecture) -GitHub
Análisis de modelos de arquitectura		- Optimización de recursos - Buscar funcionalidades comunes - Elasticidad - Gestión de recursos y seguridad - Prescripción de modelos con base en casuística	Enfoque interno donde hay un enfoque donde se promueve el reuso por capacidades comunes: sistemas compartidos, componentes compartidos, plataformas compartidas, código compartido
Herramientas de modelado de arquitecturas de servicios	Integration Designer IBM, Archimate		Utilización de lenguajes estandarizados, sin atarse a una herramienta específica

Portafolio de Servicios

Descripción

Esta categoría busca establecer un comparativo entre las prácticas del FNA y de la industria relacionadas con el manejo de los servicios y el manejo de su portafolio. Herramientas relacionadas para su gestión, registro, búsqueda y documentación.

Prácticas del Sector Financiero

Herramientas para el manejo del portafolio de servicios Gobierno del portafolio de servicios

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Herramientas para el manejo del portafolio de servicios	Archimate	- API Market - APIGateway	API Gateway

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Gobierno del portafolio de servicios	Arquitectura de aplicaciones oficina de informática fondo nacional del ahorro		Definición de prácticas internas basadas en microservicios: - Principio de una sola responsabilidad - Tener un repositorio de datos para cada microservicio - Utilizar la comunicación asíncrona para lograr un acoplamiento flexible - Fallos rápidos mediante un disyuntor para lograr tolerancia a fallos - Proxy de las solicitudes de microservicios a través de una pasarela API - Asegurar que los cambios en la API son compatibles con versiones anteriores - Versionar microservicios para cambios disruptivos - Disponer de una infraestructura dedicada para alojar su microservicio - Crear un "tren de liberación" independiente - Crear eficacia organizativa

Gobierno SOA

Descripción

Esta categoría busca comparar las mejores prácticas de gobierno SOA en el sector y en el FNA. Este ítem se relaciona de forma directa con el punto de portafolio de servicios pero va más allá. En esta categoría se establecen criterios como políticas, guías y procedimientos asociados a la dirección de la arquitectura de servicios de la organización.

Mejores prácticas en el sector

Comité de arquitectura Políticas y lineamientos Arquitectura de referencia Toma de decisión Priorización de servicios

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Comité de arquitectura	No existe formalmente		Comité conformado donde se involucran todos los especialistas, con un enfoque de la "solución" es lo primordial.
Políticas y lineamientos	Arquitectura de aplicaciones oficina de informática fondo nacional del ahorro	Orientación a procesos y automatización	Claramente establecidos, de conocimiento público y en constante evolución. - Innersource: "Desarrollo de aplicaciones empresariales con prácticas de código abierto"
Arquitectura de referencia	- Bus de servicios de Entidad (ESB). - Arquitectura de referencia SOA	- Service mesh - Domain driven Design - REST - GraphQL - Cloud computing	Uso de arquitecturas de referencia con un enfoque proactivo más que reactivo. Donde se busca que la mejor documentación se encuentre en el futuro propuesto. - Cloud Computing - Microservicios - Arquitectura orientada a eventos - Event-Driven Architectures (EDA) - Serverless - Service mesh
Toma de decisión	Fragmentados en silos	metodología SGMM (SOA Governance and Management Method)	Decisiones de arquitectura basadas al tener en cuenta diferentes perspectivas pasando desde el negocio a lo técnico.
Priorización de servicios	Orientado por necesidades de área de negocio		Enfoque desde el punto de vista de minimizar los "Riesgos para el negocio"

Estrategia de Versionamiento

Descripción

Esta categoría busca establecer los mecanismos, herramientas y mejores prácticas relacionadas con el manejo de las versiones de los servicios.

Prácticas del Sector Financiero

Manejo de la gestión del cambio Herramientas para documentación y gestión de las versiones de servicios Documentación de los servicios y sus versiones

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Manejo de la gestión del cambio	Definida al interior de la organización		API Gateway, con versiones para cada cambio
Herramientas para documentación y gestión de las versiones de servicios			Swagger, MadCap Flare
Documentación de los servicios y sus versiones	Definida al interior de la organización. No centralizada, ni alineada a un gobierno		OpenAPI, Swagger

Monitoreo y Operación

Descripción

Esta categoría analiza las diferentes herramientas, mecanismos y mejores prácticas relacionadas con el monitoreo y operación de los servicios en operación.

Prácticas del Sector Financiero

Herramientas utilizadas para gestionar el despliegue y operación de los servicios Herramientas para monitoreo de la operación, gestión de alarmas y notificaciones

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Herramientas utilizadas para gestionar el despliegue y operación de los servicios	IBM Jazz	GitLAB CI	Jenkins, herramientas del proveedor de nube, kubernetes
Herramientas para monitoreo de la operación, gestión de alarmas y notificaciones	Herramientas para monitoreo a nivel de Infraestructura	New Relic	herramientas del proveedor de nube

Mantenimiento y Evolución

Descripción

Esta categoría analiza las estrategias recomendadas para favorecer el mantenimiento y evolución de los servicios. Es decir la capacidad de la organización de poder mantener, mejorar, corregir y evolucionar los servicios de la organización.

Prácticas del Sector Financiero

Gestión del cambio Manejo de solicitudes de cambio Gestión de prioridades para los cambios Gestión de inconformidades y defectos

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria líder mundial (Capital One)
Gestión del cambio	Comité de cambios	PI Planning , Oficina Agile	Enfoque en "reducción del riesgo"
Manejo de solicitudes de cambio	Procedimiento interno		Implementación de microservicios con separación de lógica y responsabilidades
Gestión de prioridades para los cambios	Comité de cambios		Service Design: diseñar la experiencia orquestada de todos los puntos de contacto del servicio.
Gestión de inconformidades y defectos	Procedimiento interno		El principio de "sala blanca"(clean room) se centra en la prevención de defectos

Nuevas tecnologías

Descripción

En esta categoría se analizan las prácticas del FNA comparadas con la industria en temas relacionados con la vigilancia de nuevas tecnologías asociadas a servicios, así como las políticas de renovación y adopción de dichas tecnologías.

Prácticas del Sector Financiero

Comparativa FNA

Práctica	Estado FNA	Estado Entidad Bancaria Líder en Hispanoamericana (BBVA)	Estado Entidad Bancaria Líder mundial (Capital One)
Políticas de vigilancia	No existe formalmente	'Reinventar la empresa en la era digital'	Observación del mercado, revisión interna y constante experimentación
Laboratorios para experimentación de nuevas tecnologías		BBVA Next Technologies	Equipo interno con orientación centrada al cliente, API abierta a integración con terceros, hackatons, iteraciones y prototipados agiles
Capacitación y entrenamiento		Política de certificación con aliados claves de tecnología	Presupuesto interno para estimular el crecimiento académico, adicional hay enfoque en proyectos personales
Plan programado de adopción y migraciones			Política de salida a producción completamente automatizada, bajo el principio de "Sala blanca", donde el proceso garantiza que no hay dudas sobre lo que sale a producción

Fase 2: SOA Objetivo FNA

Contenido de los Productos Contractuales, 181-2020

Producto 6: PR6. Gobierno SOA

El gobierno SOA es el encargado de vigilar las relaciones entre las áreas de negocio incluidas en el segmento FNA del presente diagnóstico (la vicepresidencia de operaciones y la vicepresidencia de crédito del FNA) y la implementación y diseño de soluciones SOA. El modelo de gobierno SOA para el Fondo debe asistir en la aplicación y ejecución de un régimen (estándar) de implementación, observación y puesta en marcha de soluciones SOA.

Nota: los análisis de este producto están dirigidos a cumplir los objetivos del proyecto SOA: dependencia de proveedor (OBJ1), fortaleza SOA de las aplicaciones (OBJ2), y tiempo de mercado (OBJ3).

Justificación

El cierre de brechas de alineación negocio FNA-SOA (Ver PR9. Portafolio de iniciativas y brechas) es una de las principales razones por las que implementar el modelo de gobierno SOA. Pero no es la única. El gobierno también debe demostrar los beneficios de aumentar el nivel de eficacia (madurez) SOA, tanto en el segmento FNA seleccionado por este diagnóstico como a todo el Fondo. Por último, el gobierno SOA funciona como una disciplina para hacer frente a los retos futuros con una mirada en la inversión de tecnología, los riesgos operativos y en la alineación con el negocio del FNA.

Contenidos

1. Definición de objetivos y capacidades del gobierno SOA
2. Equilibrio arquitectura y procesos SOA
3. Supervisión de efectividad y factibilidad SOA
4. Mejora de los índices de efectividad (madurez) SOA
5. Consideraciones para la puesta en marcha del gobierno SOA en el FNA

Criterios de Aceptación

- Objetivos y capacidades del gobierno SOA para el Fondo
- Índices de efectividad (madurez) SOA 1

Tema	Gobierno SOA: Definición de objetivos del gobierno SOA
Palabras clave	SOA, Situación actual, Capacidades de negocio
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	Fase 2 PR6 Gobierno SOA

Definición de objetivos del gobierno SOA

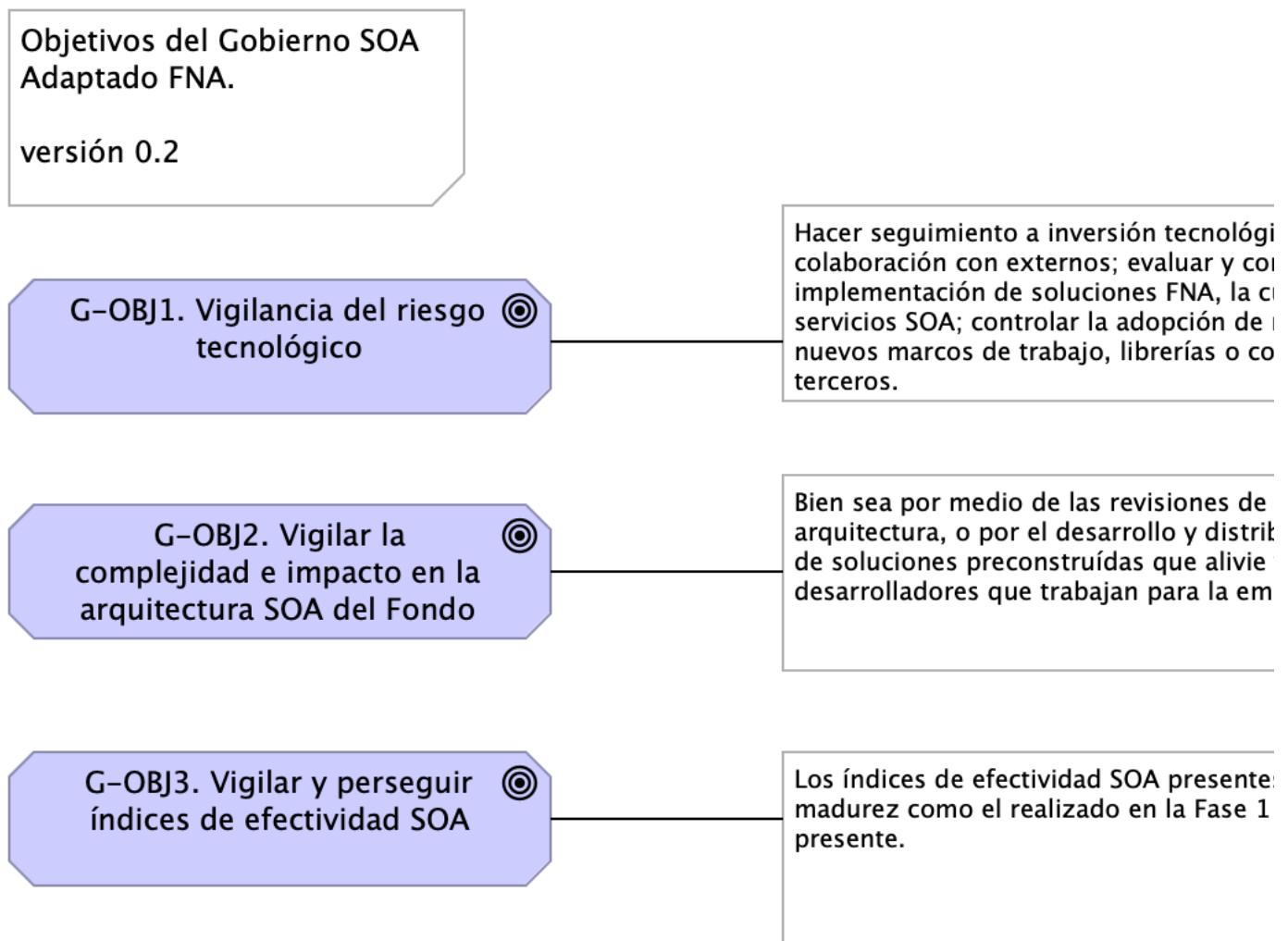
Dado los niveles de complejidad que el FNA ha alcanzado y por los riesgos tecnológico que contraen las partes analizadas en la Fase 1 de este diagnóstico (ver [04b.Resumen Fase 1](#)), esto es, aplicaciones, servicios, procesos, infraestructura, entre otros, la figura de gobierno SOA es de carácter obligatorio en el Fondo. Esta sirve además para complementar los procedimientos de TI del Fondo (ver anexo 1, [06n. Anexos](#)).

De las problemáticas encontradas en la Fase 1 del presente diagnóstico, nos estamos refiriendo explícitamente a las que incrementan la complejidad de las herramientas de software y soluciones tecnológicas del FNA, *como lo es el manejo de dependencias de los servicios SOA del Fondo*. Es muy conocido, y demostrado, que solo esta condición es la causa de la mayoría de los sobreesfuerzos en los cambios y de la dificultades a la hora de predecir su impacto.

Junto a esta complicación, que de por sí hace obligatoria la introducción de la figura del gobierno SOA en el Fondo, y que organizaciones como el OpenGroup señalan que se dan por la falta de Arquitectura Empresarial, y por consiguiente, por un débil gobierno, se encuentran otras que fueron levantadas en la Fase 1 del presente diagnóstico: agilidad limitada, complejidad e imprecisión en la trazabilidad, ocultamiento de funcionalidades (ver [04b.Resumen Fase 1](#)).

Objetivos Principales del Gobierno SOA del FNA

Una vez identificadas estas problemáticas de orden mayor existentes en la empresa y que podemos resumir en tres: riesgo tecnológico, complejidad y nivel de adopción SOA presente en los desarrollos e implementaciones del FNA, el gobierno SOA propuesto para el Fondo debe incluir entre sus objetivos, como mínimo, los siguientes



[Imagen](#). Objetivos principales del gobierno SOA del FNA.

Fuente: elaboración propia.

1. Vigilancia del riesgo tecnológico en tres vías. Primero, hacer seguimiento a la inversión tecnológica, el cual involucra a los proveedores del FNA y colaboración con externos; segundo, evaluar y constatar la implementación de herramientas de software, la cual podría limitarse a servicios SOA; y finalmente, controlar la adopción de nuevos diseños, nuevos marcos de trabajo, librerías o componentes de terceros.
2. Del objetivo anterior (Vigilancia del riesgo tecnológico) se desprende este para enfatizar el control sobre la inversión de TI: vigilar la efectividad y factibilidad de los proyectos SOA de la organización en términos del área de efectividad del costo y factibilidad SOA. Ambos conceptos explicados en [Supervisión de efectividad y factibilidad SOA](#).

3. Vigilar el crecimiento de la complejidad y el impacto de los nuevos cambios en la arquitectura de referencia SOA del Fondo, bien sea por medio de las revisiones de diseño, el comité de arquitectura, o por el desarrollo y distribución de un catálogo de soluciones preconstruídas que alivie y guíe a las fábricas y desarrolladores que trabajan para la empresa.
4. Vigilar y perseguir el aumento de los índices de adopción, adaptación y efectividad SOA presentes en los análisis de madurez como el realizado en la Fase 1 del diagnóstico presente (ver [02.Fase 2 PR2 Estudio Madurez SOA FNA](#)).
5. Poner en marcha el proceso de gobierno SOA del FNA descrito más adelante en este ejercicio (181-2020). Adaptar y monitorear los índices de rendimiento (KPI) del proceso.

Otros Objetivos del Gobierno SOA

Para complementar la lista de objetivos del gobierno SOA recomendados por este diagnóstico al Fondo, la lista siguiente expone objetivos que pueden ser conseguidos de forma indirecta, o con la mediación de proyectos transformadores, como la Arquitectura Empresarial, transformación digital, arquitectura de negocio, entre otros.

1. Desde el área, o rol, de gobierno SOA del FNA, servir de guía en la entrega de soluciones de software conforme a la arquitectura de referencia estregada por esta consultoría.
2. Informar de desviaciones en la relación de efectividad de costos de los proyectos SOA del FNA (en términos del área de inefectividad de costo e infactibilidad SOA).
3. Hacer el seguimiento de las implementaciones de los cambios en la arquitectura de referencia: phase G, Implementation Governance, TOGAF ADM.

Objetivos del Gobierno SOA Adaptado FNA.

versión 0.4

G-OBJ1. Vigilancia del riesgo tecnológico

Hacer seguimiento a inversión tecnológica; colaboración con externos; evaluar y controlar implementación de soluciones FNA, la creación de servicios SOA; controlar la adopción de nuevos marcos de trabajo, bibliotecas o componentes de terceros.

G-OBJ2. Vigilar la complejidad e impacto en la arquitectura SOA del Fondo

Bien sea por medio de las revisiones de arquitectura, o por el desarrollo y distribución de soluciones preconstruidas que alivie a los desarrolladores que trabajan para la empresa.

G-OBJ3. Vigilar y perseguir índices de efectividad SOA

Los índices de efectividad SOA presentes en la fase 1 indican la madurez como el realizado en la Fase 1 presente.

G-OBJ Secundario 1. Guiar soluciones conforme a la arquitectura de referencia

G-OBJ Secundario 2. Informar de desviaciones efectividad-costos de los proyectos SOA

[Imagen](#). Objetivos secundarios del gobierno SOA del FNA.

Fuente: elaboración propia.

Tema	Gobierno SOA: Riesgos tecnológicos del FNA
Palabras clave	SOA, Gobierno, Riesgo, Objetivos de negocio
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	Fase 2 PR6 Gobierno SOA

Riesgos Tecnológicos del FNA

Los riesgos tecnológicos encontrados en el diagnóstico SOA de Fase 1 (ver [03.Fase 1 PR3 Resultado Diagnóstico Situación Actual](#)), y que describimos abajo, están incorporados a las vigilancias del gobierno SOA propuesto. A la vez, estos mismos riesgos descritos a continuación deben ser agregados a la

matriz de gestión actual del Fondo, y gestionados por los métodos propios con los que ya cuenta el FNA.

Gobierno SOA y Riesgos Tecnológicos del FNA

El gobierno SOA del FNA debe buscar la capacidad para medir y gestionar (agregar a la lista de riesgos de la empresa) los riesgos que lo están afectando.

- R1. Riesgo de agilidad limitada (ver imagen 1)
 - R2. Riesgo de baja orquestación SOA (ver imagen 2a y 2b)
 - R3. Riesgo de crecimiento de dependencias entre servicios SOA
 - R4. Riesgo de crecimiento de adaptadores particulares –opuesto al estándar– (ver imagen 3)
 - R5. Riesgo de baja reutilización de servicios SOA
 - R6. Riesgo de permanencia de aplicaciones silos
 - R7. Riesgo de falta de trazabilidad para la evolución e implementación de los servicios SOA

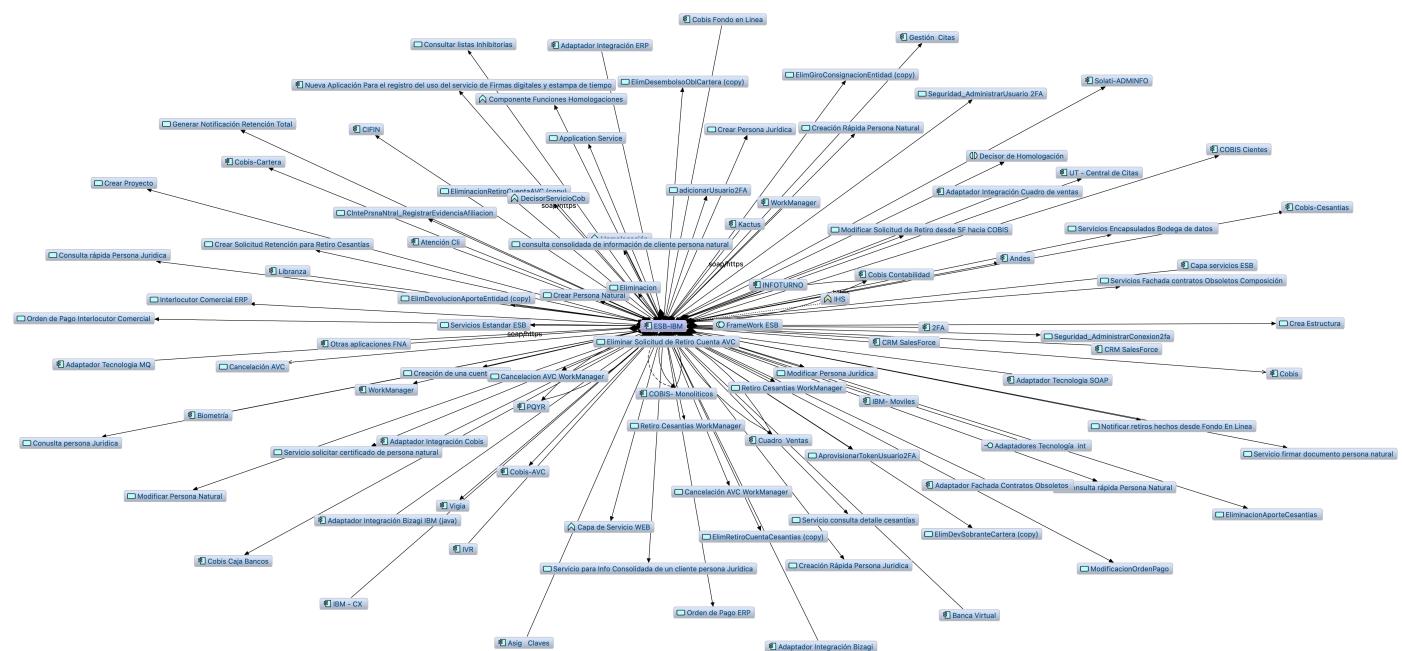
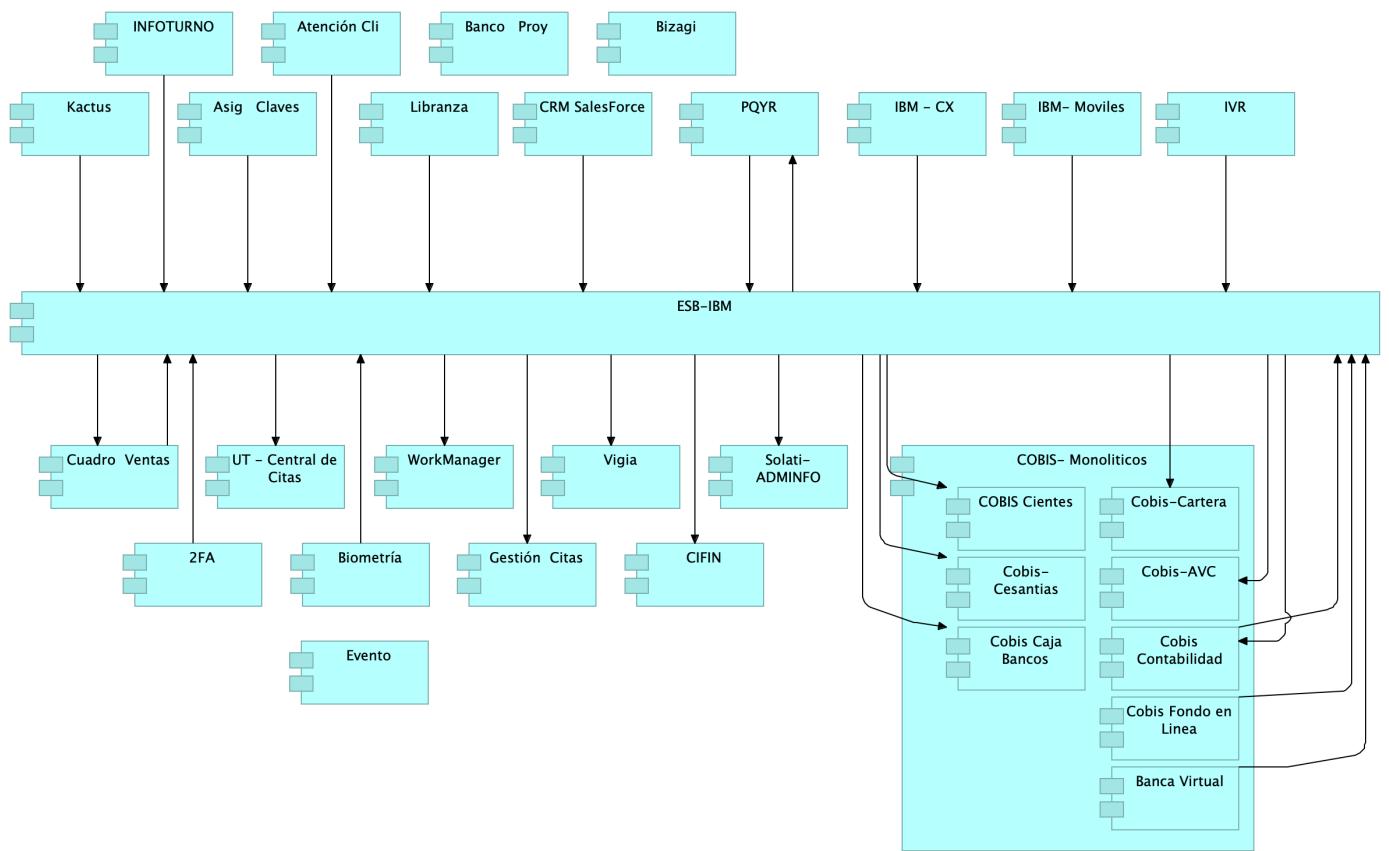


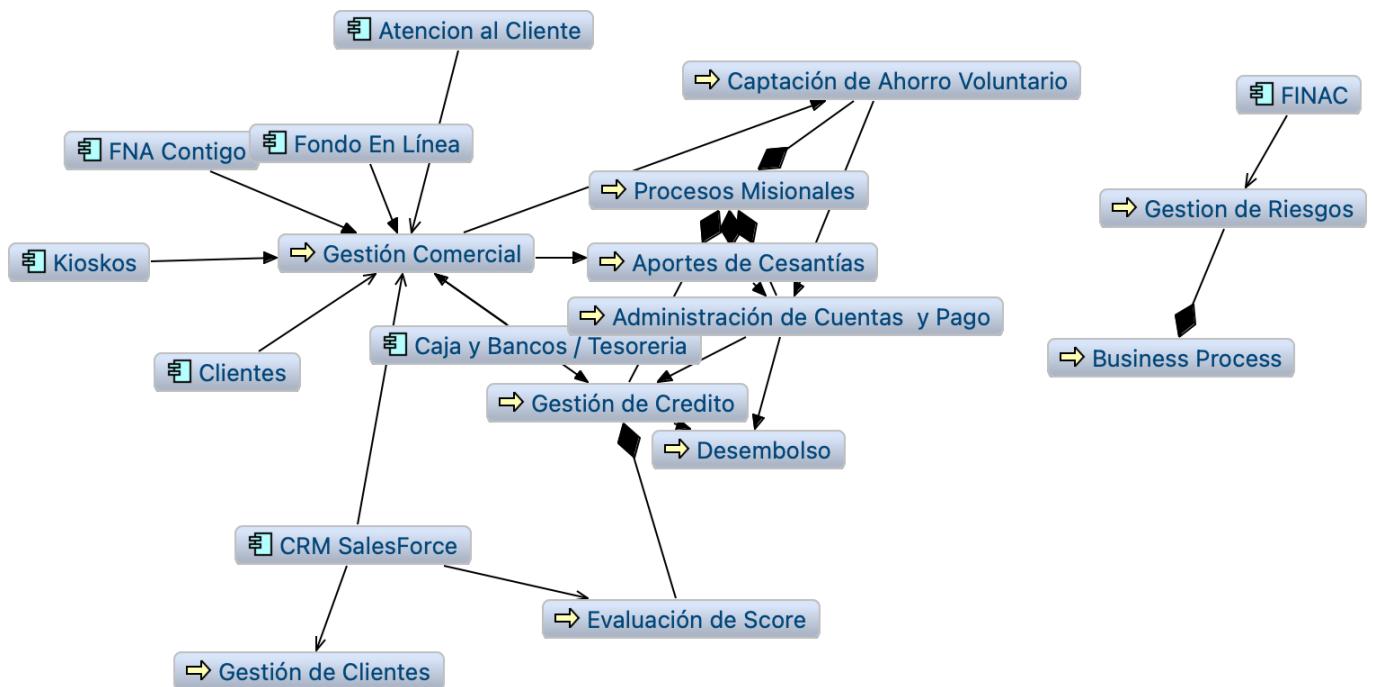
Imagen 1. Riesgo no. 1 y 2. Agilidad limitada por el proliferación de la complejidad en los servicios SOA del Fondo.

Fuente: elaboración propia, ae_fna_as_is.archimate.



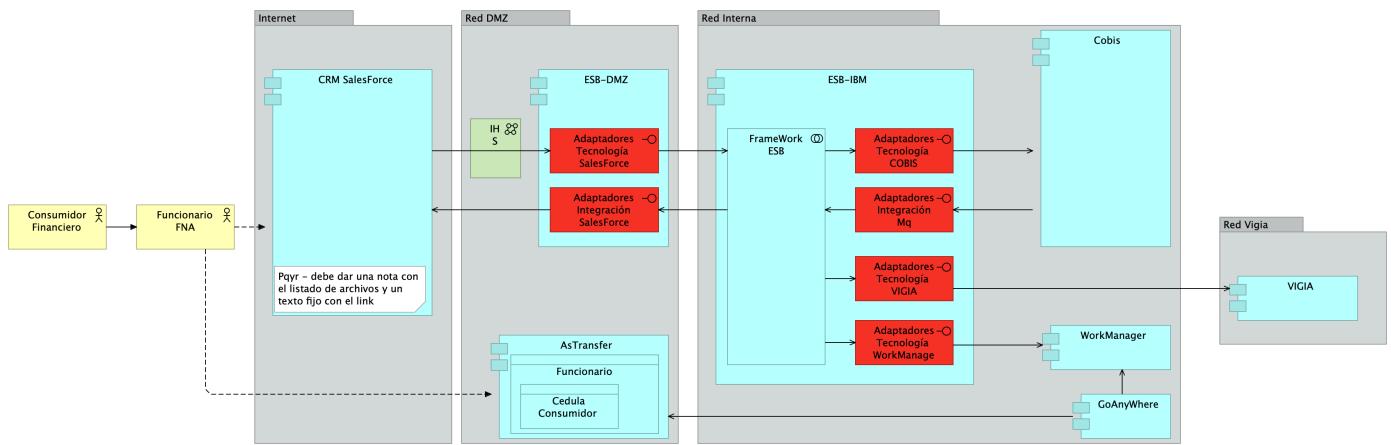
[Imagen 2a.](#) Riesgo no. 2. En el modelo “ae_fna_as_is.archimate” de situación actual del FNA no aparecen relaciones de este con los procesos de negocio. La principal utilización del bus (ESB-IBM, en la imagen) está destinada a la integración de aplicaciones.

Fuente: ae_fna_as_is.archimate.



[Imagen 2b.](#) En el modelo “ae_fna_as_is.archimate” no es posible encontrar relaciones del bus con los procesos.

Fuente: ae_fna_as_is.archimate.



[Imagen 3.](#) Riesgo no. 4 y 5. Caso de CRM del FNA. Le afecta el riesgo de estándarización en la interoperabilidad, aumenta la cantidad de adaptadores particulares.

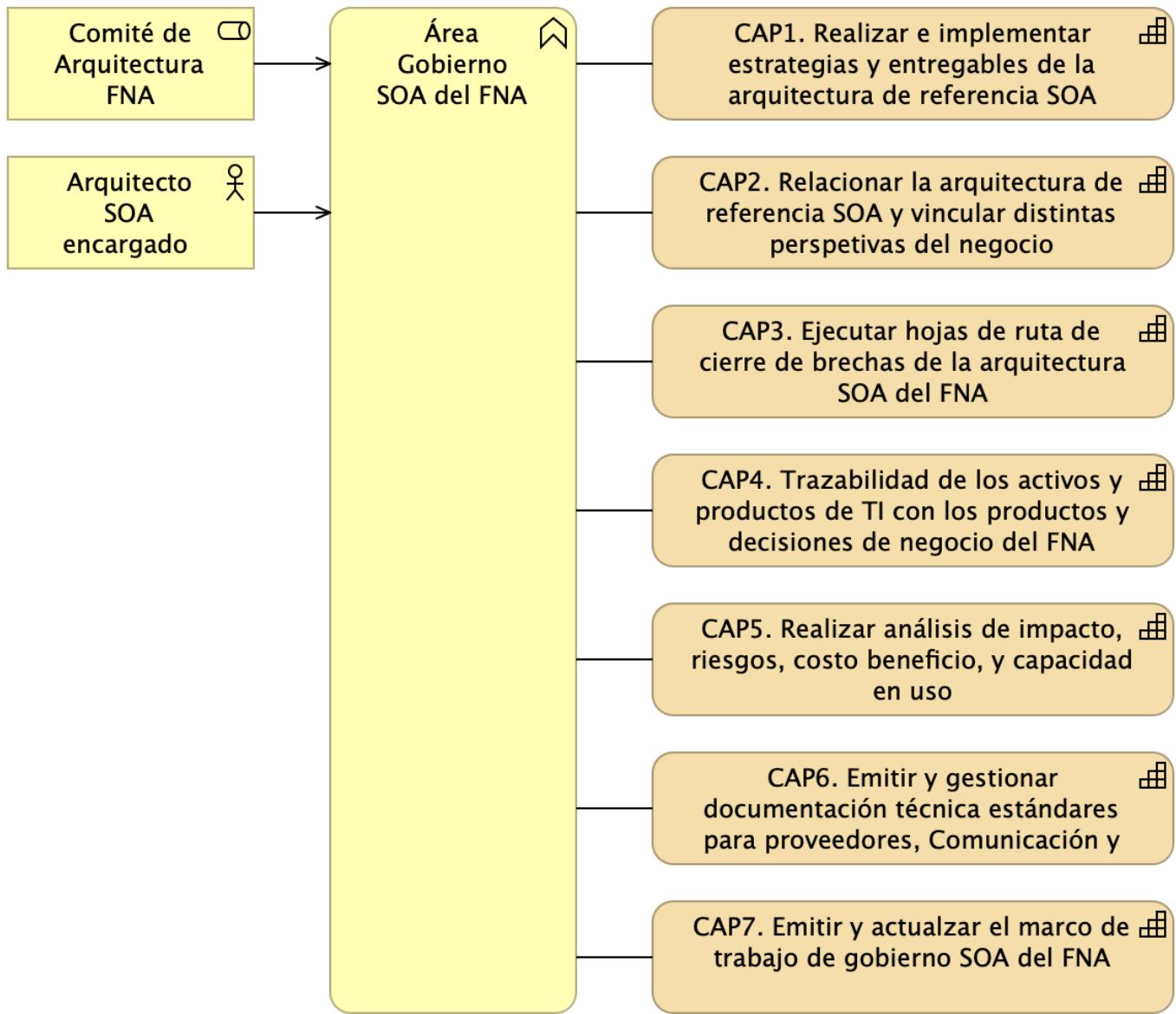
Fuente: ae_fna_as_is.archimate.

Tema	Gobierno SOA: Capacidades FNA por Desarrollar para el Gobierno SOA
Palabras clave	SOA, Situación actual, Capacidades de negocio
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	Fase 2 PR6 Gobierno SOA

Capacidades FNA por Desarrollar para el Gobierno SOA

Las capacidades requeridas por el gobierno SOA del Fondo, que a la vez soportan a la matriz de roles y responsabilidades (RACI) futuras del equipo de gobierno SOA o del Comité de arquitectura del Fondo están relacionados directamente con los objetivos del gobierno SOA y con los riesgos tecnológicos encontrados en la Fase 1 (ver [06a.Objetivos gobierno](#), y [06c.Riesgos tecnológicos SOA](#)).

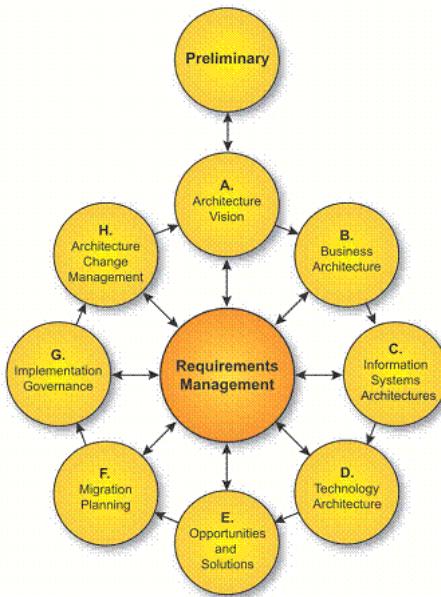
Nota: no es del alcance de esta consultoría, E-Service (contrato 181-2020), desarrollar estas capacidades. Esto requeriría de la ejecución de otro(s) proyecto. El Fondo podría desarrollar estas capacidades indicadas por este diagnóstico de forma interna, o contratar proyectos para la creación y gestión de estas capacidades.



Fuente: elaboración propia.

1. CAP1. Realizar e implementar estrategias y entregables para planear y analizar la arquitectura de referencia SOA del Fondo (resultado Fase 2 de este diagnóstico).
2. CAP2. Relacionar elementos de la arquitectura de referencia SOA (181-2020), esto es, procesos de negocio, servicios, datos, aplicaciones, componentes, tecnologías, etc., y vincular distintas perspectivas a una vertical de negocio del Fondo, como, Gestión Comercial, Cesantías, Ahorro Voluntario para desarrollar modelos consistentes y desarrollar la oportunidad de dirigir cambios y atacar los problemas de integralidad de las pruebas.
3. CAP3. Realizar y gestionar la puesta en marcha de hojas de ruta para cerrar brechas y alcanzar estados futuros de la arquitectura (TOGAF 9, Fase G, Implementation governance). Ver anexo 1.
4. CAP4. Ejercitarse la trazabilidad de los activos y productos de TI con los productos y decisiones de negocio de las áreas del FNA, como la vicepresidencia de Crédito y la de Operaciones ante la llegada de impactos, cambios, reformas y bajas tecnológicas.
5. CAP5. Realizar análisis de impacto, riesgos, costo beneficio, capacidad usada, y gestión de demanda ante la evaluación de cambios a la arquitectura de referencia SOA y activos de TI, cuando lo amerite.
6. CAP6. Emitir y gestionar los contenidos de documentación técnica, como principios, restricciones tecnológicas, marcos de trabajo, normas, diseños, y estándares para el uso de los proveedores, servicios de fábricas del FNA, planta interna, Comunicación y Planeación.
7. CAP7. Emitir y actualizar el marco de trabajo de gobierno SOA que complementemente el de TI del Fondo y que enuncie los procesos, procedimientos y responsabilidades de este con el fin de generar la autoridad y asistir en la toma de decisiones internas (áreas de TI y Planeación del Fondo) y externas (proveedores del Fondo)

Anexo 1. Proceso de Arquitectura Empresarial



[Imagen 2.](#) TOGAF 9, Fase G, Implementation governance para la implementación de las capacidades de ejecución de hojas de ruta del FNA.

Fuente: www.opengroup.org/soa/source-book/togaf.

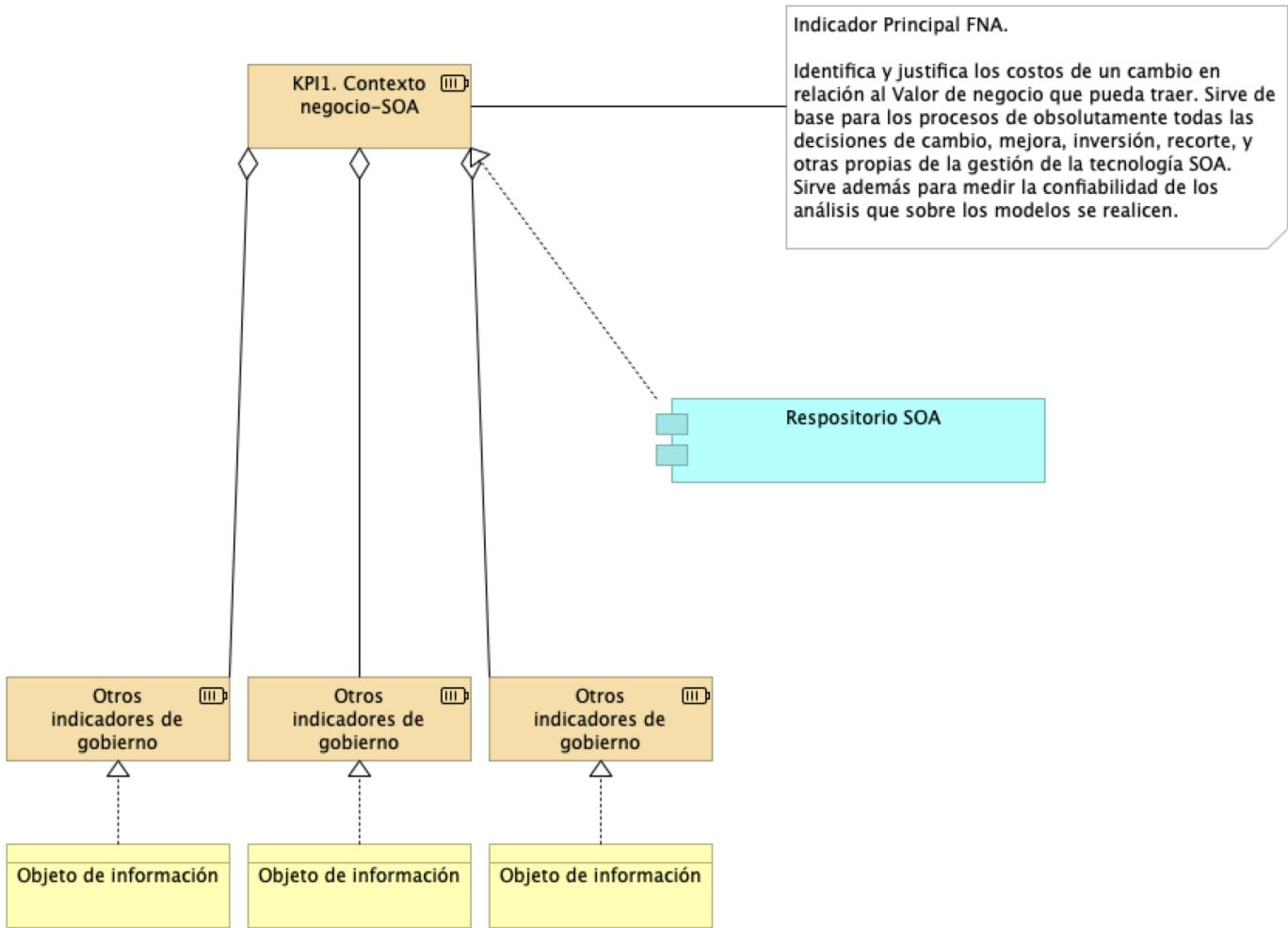
Tema	Gobierno SOA: Mejora de los indicadores de efectividad (madurez) SOA
Palabras clave	SOA, Inversión TI, Eficacia, KPI, Proceso SOA
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	Fase 2 PR6 Gobierno SOA

Indicadores de Efectividad del Gobierno SOA

Más allá de los índices propuestos por el análisis de madurez SOA desarrollado en la Fase 1 de este diagnóstico, para el FNA es clave mantener activo el vínculo de sus activos tanto de infraestructura como los activos SOA (y de todos, aunque se escapen del alcance de este ejercicio) con el contexto de negocio de las vicepresidencias de Operaciones y de Crédito. Esto es, el principal indicador del gobierno SOA es la existencia y actualidad de los vínculos entre los contextos de negocio y la arquitectura de referencia SOA FNA, y su tecnología.

Este solo indicador del gobierno SOA, el del vínculo de los contextos negocio-tecnología SOA, que es útil también para otras disciplinas de gestión TI, identifica y justifica los costos de un cambio en relación al Valor de negocio que pueda traer. Sirve de base para los procesos de absolutamente todas las decisiones de cambio, mejora, inversión, recorte, y otras propias de la gestión de la tecnología SOA. Sirve además para medir la confiabilidad de los análisis que sobre los modelos se realicen. Es por estas razones que para el Fondo este es el principal indicador de gobierno SOA a desarrollar.

De todos, el indicador de Vínculo Contexto Negocio-SOA es el más importante para el FNA. Para mantener el puntaje de este indicador alto debe recurrir a la actualización y mantenimiento del repositorio de arquitectura. En la medida en que este repositorio capture la mayor cantidad de información de los contextos referidos, el indicador aumentará, a la vez que el repositorio será un activo clave para todas las operaciones de gestión de TI del FNA.



[Imagen 1.](#) Indicador principal del gobierno SOA: KPI de Vínculo Contexto Negocio-SOA. Identifica y justifica los cambios SOA en relación al Valor de negocio.

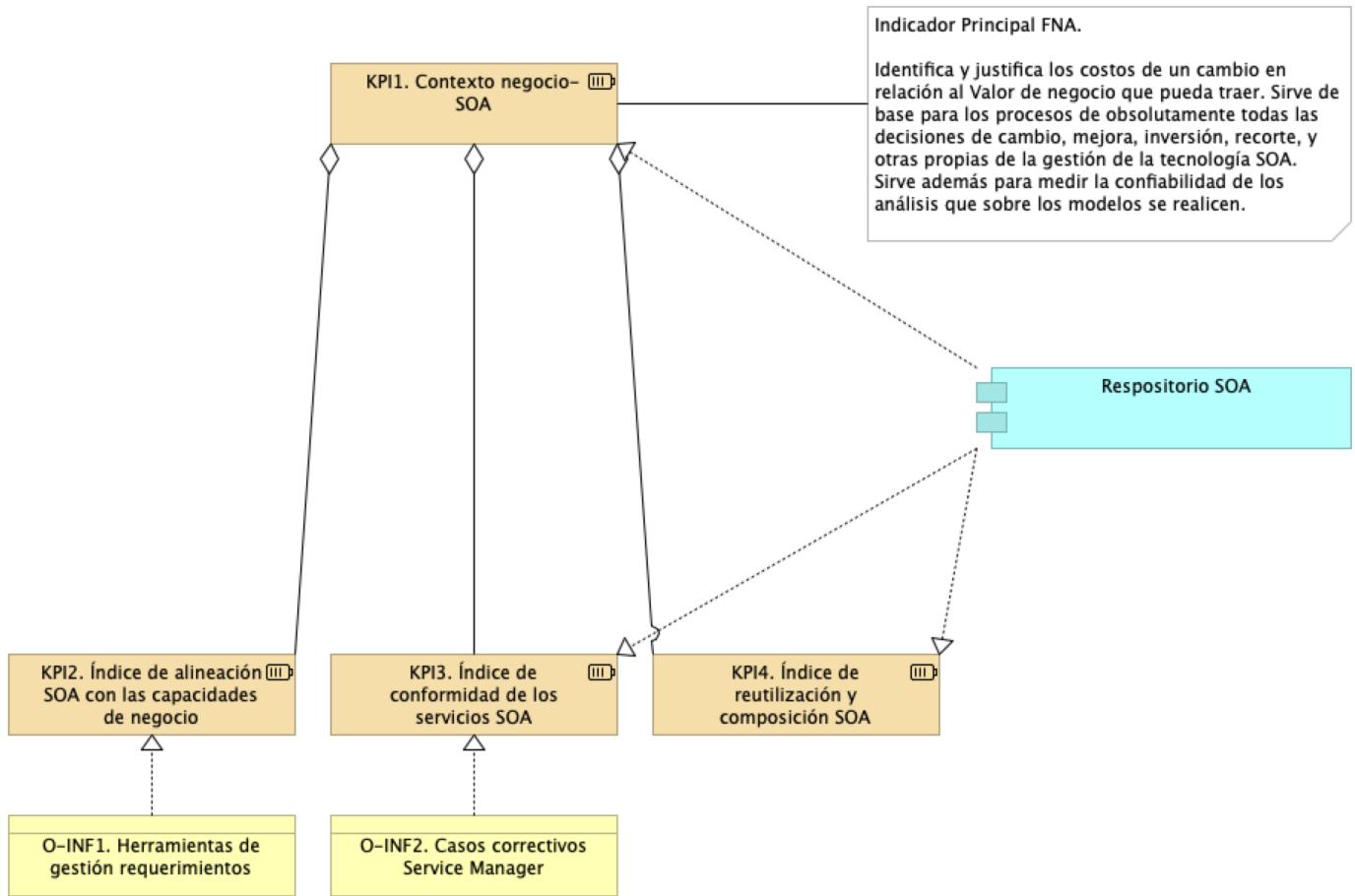
Fuente: elaboración propia.

Indicadores Complementarios del Gobierno SOA

Los siguientes indicadores han sido escogidos para atacar los problemas en la gestión de SOA encontrados en la Fase 1 del presente diagnóstico (ver [04b.Resumen Fase 1](#)). sirven de base de pirámide para el indicador principal.

1. Índice de alineación de la arquitectura de referencia SOA con las capacidades de negocio
2. Índice de conformidad de los servicios SOA a los estándares de diseño orientados a servicio
3. Índice de reutilización y composición de los servicios SOA del portafolio FNA

Cada uno de estos indicadores debe ser soportados con fuentes de información existentes o por desarrollar, que aparecen en la imagen como *Objeto de información*.



[Imagen 1.](#) Indicador principal del gobierno SOA: KPI de Vínculo Negocio-SOA. Identifica y justifica los cambios SOA en relación al Valor de negocio.

Fuente: elaboración propia.

Algunas fuentes u objetos de información sugeridas y ya existentes en el FNA son, por ejemplo, el número y tipo de requerimientos registrados en las herramienta de gestión, como Clear Case (IBM); los resultados de las evaluaciones de calidad de las implementaciones de los proveedores del Fondo; o el número de casos correctivos en un período contenidos en el Service Manager, entre otros.

Indicadores de Eficiencia SOA

Además de los indicadores del proceso de gobierno SOA enunciados arriba, el FNA debe ejercer el gobierno SOA sobre los siguientes indicadores de eficiencia de la arquitectura de referencia.

Del Análisis de Madurez SOA del FNA (Fase 1)

La dimensión que tuvo puntajes meritorios fue la de Negocio que alcanzó el segundo percentil de eficacia: 37 / 100. Este índice sintetiza el hecho de que operativamente SOA responde a las exigencias de negocio del FNA, pero con esfuerzos altos que seguramente afectan el retorno de inversión SOA.

Según los resultados internos del diagnóstico, los problemas del alto esfuerzo en la eficacia de la dimensión de negocio es causado por las dificultades siguientes: 1. El bajo grado de independencia de proveedor: (ver imagen abajo) 38 puntos / 100 puntos 1. Muy baja flexibilidad y tiempos de entrega (time-to-market): 20 / 100 puntos.

Análisis de Servicios SOA -- FNA

Objetivos	Evidencia (Caracterización, Procedimiento, Manual , Guía, Formato, Documento Soporte, Indicador)	Negocio
Independencia Proveedor	<p>Existe una Arquitectura de procesos de negocio definida formalmente, documentada y mantenida?</p> <p>Existen varias líneas de negocio diferentes? ¿Cada una necesita o define sus propios procesos?</p> <p>Hay un dueño de los procesos, aplicaciones y servicios?</p> <p>Qué tan ágiles son los procesos de negocio actuales?</p> <p>¿Los procesos del Gobierno SOA existen? ¿Están documentados?</p> <p>¿Cada uno de las áreas involucradas dentro del proceso SOA tiene roles (Arq. Señal, Infraestructura, Negocio, etc.)?</p> <p>Existen cambios en la organización que afecten en: regulaciones y políticas, creación de nuevas unidades de negocio?</p> <p>¿Existe alguna autoridad SOA dentro de la organización (Comité Arquitectura)? ¿Es independiente?</p> <p>¿Existe alguna autoridad de Gobierno TI?</p> <p>¿Está organizada la administración de proyectos IT, existe una PMO?</p> <p>¿Qué nivel de uso de herramientas de diseño (Enterprise Architecture) y colaboración entre las unidades de negocio?</p> <p>¿Existe un repositorio para almacenar los activos y buenas prácticas definidas dentro de la organización?</p> <p>¿Existe un modelo de integración de las aplicaciones o sistemas dentro de la organización, (Servicios Web, API, ESB, etc.)?</p> <p>¿Es alta la tasa de fallos por cambios y el tiempo de mercado (time-to-market) de las aplicaciones?</p> <p>¿Qué tan común es el uso de frameworks de desarrollo en la arquitectura de la organización?</p> <p>Qué nivel de control tiene el FNA sobre la participación de los proveedores en la ejecución de los procesos?</p> <p>¿Qué nivel de independencia tiene la infraestructura con los proveedores actuales?</p>	40 60 20 40 20 40 40 60 40 40 20 20 20 40 40 40 38
Fortaleza SOA	<p>¿Qué nivel de administración tienen los requerimientos en los sistemas de software?</p> <p>¿Qué tan fuerte es la administración de proyectos IT?</p> <p>¿Existen y usan herramientas colaborativas para diseño y validación de arquitectura?</p> <p>¿Existe un repositorio para almacenar los activos y buenas prácticas definidas dentro de la organización?</p> <p>¿Existe un estilo de desarrollo para las aplicaciones dentro de la organización?</p> <p>¿Cómo se integran las aplicaciones o sistemas dentro de la organización? Es demostrable?</p> <p>¿Se utiliza alguna tecnología asociada con SOA como ESB (IBM, WSO2, Oracle Service Bus), datos compartidos y servicios web?</p> <p>¿Cuán confiables son las aplicaciones de negocio críticas para la organización, tiene respaldo?</p> <p>¿Cuán modernos son los repositorios de datos se utilizan en la organización? ¿Los proveedores son propietarios?</p> <p>¿Considera que la integración funcional actual está bien lograda?</p> <p>¿Considera que la integración de datos actual está bien lograda?</p> <p>¿Existen principios arquitectónicos o lineamientos están definidos? ¿son evidenciables?</p> <p>Existe un comité de cambios de arquitectura, o una bitácora de decisiones de arquitectura. ¿cómo se evalúan las transformaciones?</p> <p>Considera que existe un modelo lógico gestionado, y que este es el que gobierna al modelo físico, o es lo contrario?</p> <p>Qué nivel de esfuerzo requieren para atender solicitudes eventuales (sin planificación)</p> <p>Las transformaciones de datos se realizan mediante esquemas comunes, como DTD, JSON. Quién gobierna las transformaciones?</p> <p>¿Existe un modelo de datos común a todas las aplicaciones? ¿quién gobierna los modelos?</p> <p>¿Los traslados de información de una aplicación a otra suceden sin contratiempos? ¿Para todas las aplicaciones?</p> <p>¿Los modelos de datos representados como entidades corporativas o objetos de negocio son activos corporativos?</p> <p>¿Cómo se ejecutan las transformaciones de datos entre aplicaciones? ¿Existe un intermediario que realiza esas transformaciones?</p> <p>¿Se está trabajando en la unificación de las representaciones de datos internas de la organización?</p>	40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40
Flexibilidad y Tiempo de Mercado	<p>Cuáles son los tiempos de respuesta de los proveedores para implementar nuevas aplicaciones?</p> <p>¿Existe área de Devops que permita interactuar a los equipos de desarrollo e infraestructura dentro de la organización?</p> <p>Las aplicaciones legadas (antiguas) interactúan sin problemas con la plataforma base de Servidores de aplicaciones.</p> <p>Existe aplicaciones centradas en el negocio (accesibilidad, seguridad, confiabilidad, etc.)</p> <p>Se aplica en el diseño de servicios los diseños SOA, como estandarización contrato, abstracción, autonomía.</p> <p>Si existen reglas de transformación de datos entre distintos modelos, ¿estas reglas son administradas y mantenidas?</p> <p>¿El comité de arquitectura, o el que haga las veces, discute y aprueba los requerimientos no funcionales (altas disponibilidad, calidad de servicio, etc.)?</p>	20 20 20 20 20 20 20 20

[Imagen](#). Resultados Según los resultados internos del diagnóstico. Dimensión de Negocio.

Fuente: herramienta de diagnóstico de madurez SOA. Elaboración propia.

Tema	Gobierno SOA: Consideraciones para la puesta en marcha del gobierno SOA en el FNA
------	---

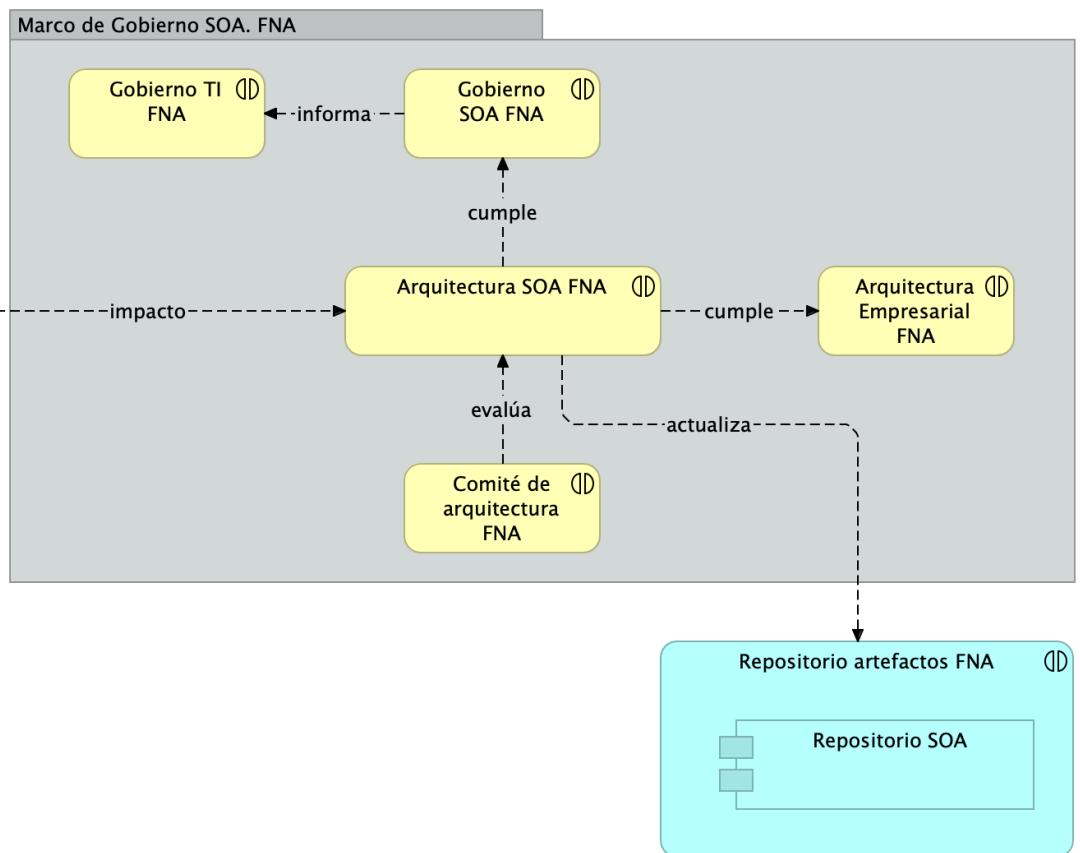
Tema	Gobierno SOA: Consideraciones para la puesta en marcha del gobierno SOA en el FNA
Palabras clave	SOA, Inversión TI, Eficacia, KPI, Proceso SOA
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	Fase 2 PR6 Gobierno SOA

Consideraciones para Implementación del Gobierno SOA en FNA

Marco de Trabajo para Implementar Gobierno SOA en FNA

Un Marco de trabajo adaptado para la organización que responde y gestiona los cambios evolutivos de la arquitectura de referencia SOA instalada en el Fondo. Esta versión del marco de gobierno, que necesariamente es una versión inicial para el FNA dado el resultado del análisis de madurez de la Fase 1, procura únicamente la gestión de cambios de la arquitectura. Versiones posteriores de este modelo puede y deberá abordar problemáticas señaladas por nuevos análisis.

Modelo de Gobierno SOA
Adaptado FNA.
versión 0.2



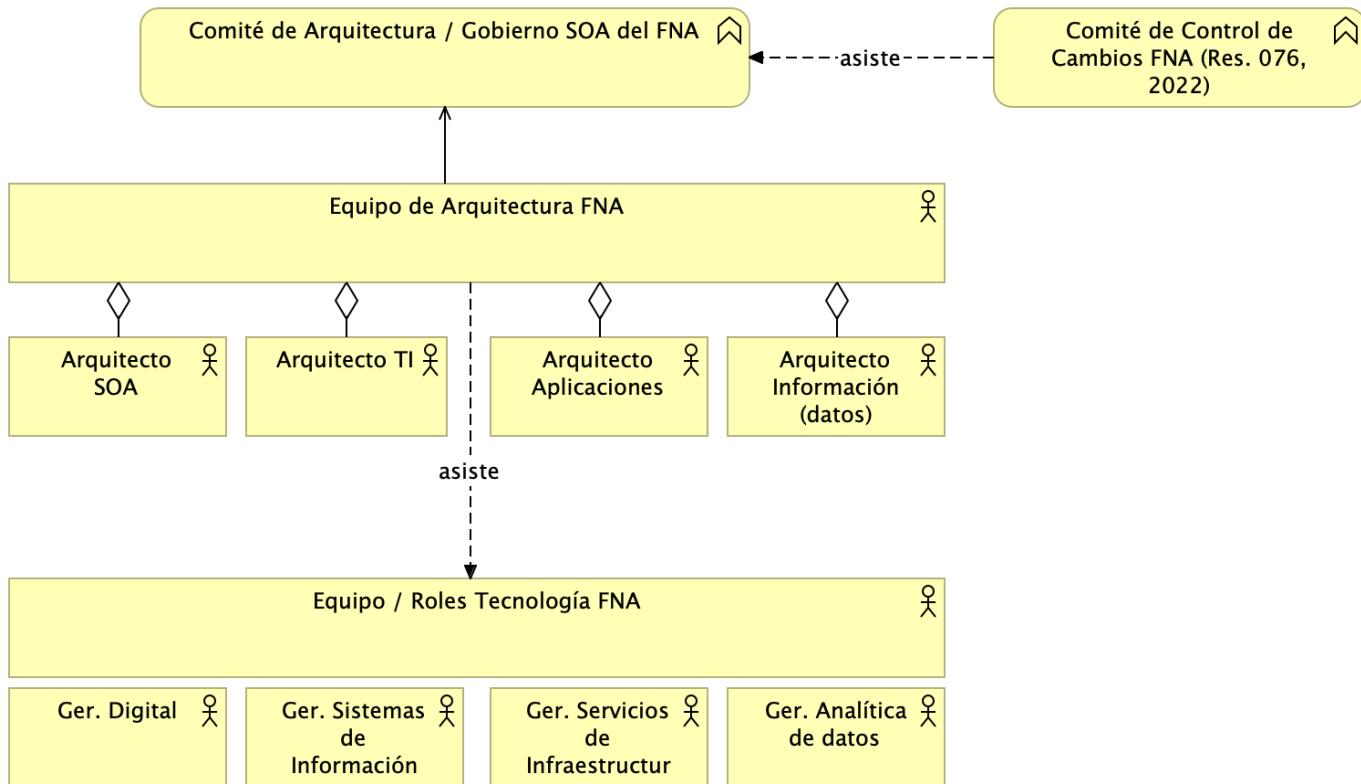
[Imagen 1.](#) Actores del FNA necesarios para el Gobierno SOA.

Fuente: elaboración propia.

Equipo de Trabajo para el Gobierno SOA del FNA

El equipo de trabajo requerido conforme el modelo de gobierno SOA presentado aquí se compone del capital humano en los roles de gestión de la tecnología y arquitectura en los ámbitos de diseño de servicios y componentes, infraestructura tecnológica y redes, aplicaciones de solución y herramientas de software y del especialista de las estructuras de negocio e información. Estos recursos deben ser preferiblemente internos a la

organización debido a la gestión de conocimiento del activo más importante del gobierno SOA, la arquitectura de referencia y los vínculos de esta con los contextos de negocio y TI.

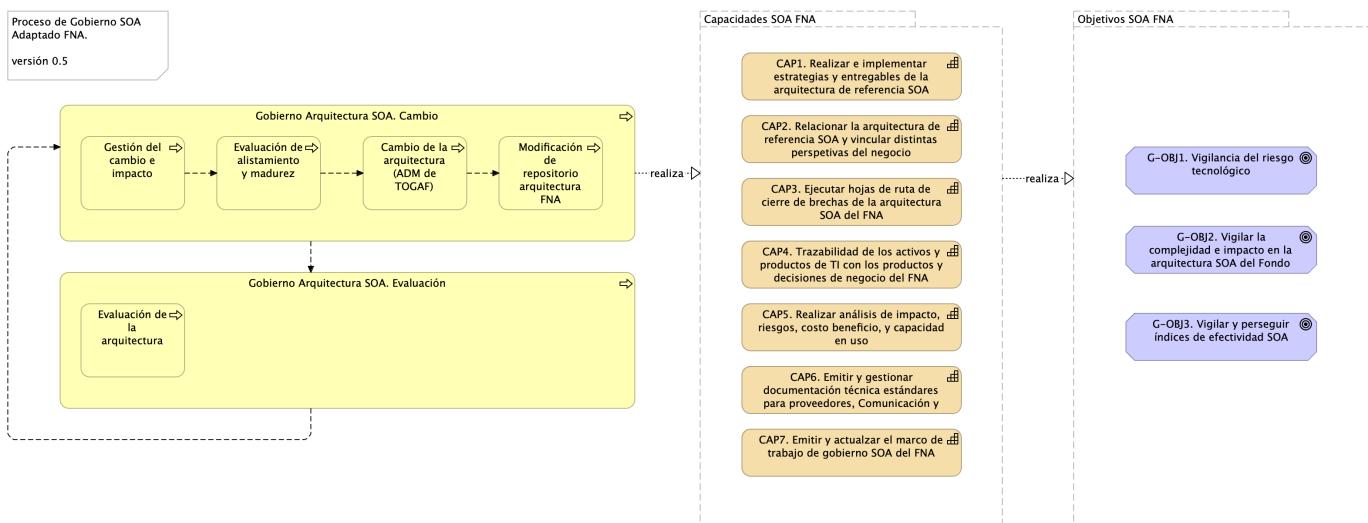


[Imagen 2.](#) Roles y Grupos de trabajo del gobierno SOA del FNA.

Fuente: elaboración propia.

Proceso de Gobierno SOA para el FNA

Por último, el proceso de gobierno SOA es la conjugación de las dos condiciones aquí presentadas, el marco de trabajo (acciones y herramientas) y el equipo de trabajo, que son los recursos de capital humano y sus interacciones. Este proceso adaptado al FNA busca alcanzar los objetivos SOA determinados por este diagnóstico. (ver imagen abajo)



[Imagen 3.](#) Actividades y relaciones del proceso principal de gobierno SOA para el FNA. Relación con capacidades y objetivos SOA necesarios para el FNA.

Fuente: elaboración propia.

El proceso de gobierno SOA está en línea y apoya a las capacidades SOA que el FNA debe implementar. De esa manera, las capacidades SOA relacionadas con este proceso buscan lograr los objetivos SOA establecidos por esta consultoría y que el FNA debe perseguir.

Anexos

Procedimiento del Área TI del FNA

Fase 2: SOA Objetivo FNA

Contenido de los Productos Contractuales, 181-2020

Producto 7: PR7. Vigilancia Tecnológica SOA

Describir y justificar la selección de tecnologías y las técnicas de diseño, implementación y gestión de servicios SOA aplicables al Fondo Nacional del Ahorro (FNA).

Nota: los análisis de este producto están dirigidos a cumplir los objetivos del proyecto SOA: dependencia de proveedor (OBJ1), fortaleza SOA de las aplicaciones (OBJ2), y tiempo de mercado (OBJ3).

Justificación

Identificar los avances en el desarrollo SOA que impulsen a las iniciativas y propuestas de solución SOA para el FNA.

Contenidos

1. Lista tecnologías SOA afines al Fondo
2. Lista de beneficios al FNA por las tecnologías seleccionadas
3. Implicaciones para la adquisición y adopción tecnológica

Criterios de Aceptación

- Evaluación e Identificación de la tecnología SOA en alineación con los objetivos del proyecto
- Presentación de una arquitectura SOA candidata versión 1.1

Tema	Vigilancia Avances de Industria: Estrategia de modelado e identificación de servicios
Palabras clave	SOA, Tecnologías, Vigilancia, Avances
Autor	
Fuente	
Version	6ddfe81 del 31 Jan 2023
Vínculos	N003a Vista Segmento SOA FNA

Arquitecturas Orientadas a Servicios (SOA)

SOA como estilo de arquitectura, representó un cambio fundamental en los procesos de desarrollo de soluciones informáticas, así como de procesos de negocio, en los últimos años. Como estilo de arquitectura, el uso de SOA pretendía favorecer la reutilización de código con la promesa de construir piezas de software que pudieran ser localizadas y consumidas por demanda.

En la actualidad, vemos que varias de las premisas se cumplieron, pero otras no han podido realmente implementarse. En primer lugar, la construcción de servicios ha presentado un reto en cuanto al diseño y granularidad de los servicios, los mecanismos de despliegue y comunicación de servicios y su relación con las arquitecturas monolíticas y la evolución y mantenimiento de los servicios. Otro aspecto que hace complejo el uso de los servicios, es el costeo de los servicios y la demostración del retorno de inversión de los mismos.

Diseño y granularidad de los servicios

En buena medida la justificación de reutilización de los servicios ha ido implicando que se implementen modificaciones y extensiones al software para cubrir diferentes funcionalidades de negocio dentro de la organización. Lo anterior implica que para poder reutilizar un servicio, usualmente se adicionan a su implementación funcionalidades nuevas específicas para una unidad de negocio. Lo anterior genera un aumento en el tamaño del código y la complejidad de los servicios. Finalmente, esto redunda en un decremento de la disponibilidad de los servicios, aumento de las fallas en producción y aumento en los costos de mantenimiento y modificación de cada servicio.

Mecanismos de Comunicación

SOA como toda tecnología ha venido evolucionando. En temas de comunicación, los inicios de SOA impusieron SOAP y WS-* como mecanismos de intercambio de información y flujo de lógica de negocio. Sin embargo, la complejidad de SOAP sigue siendo un limitante en la utilización de SOA. La utilización reciente de mecanismos más ligeros como REST y JSON han facilitado el uso de los servicios, principalmente intermediados por APIs.

El esquema de trabajo actual ha permitido el uso de servicios aunque la realidad es que las aplicaciones actuales tienen una mezcla de servicios legados utilizando SOAP con nuevos servicios utilizando REST. Esto lleva a introducir nuevos elementos en la arquitectura, como conectores y adaptadores, complicando un poco más el ecosistema de servicios de una solución.

Arquitecturas Monolíticas y Servicios

Uno de los principales retos desde el punto de vista de arquitectura y el uso de los servicios es poder mantener una consistencia de la oferta de servicios con las aplicaciones legadas de la organización. Muchas de las implementaciones de servicios, proponen un nivel intermedio que oculta o encapsula la funcionalidad de aplicaciones legadas. En este sentido los servicios son ofrecidos como puntos de contacto con los consumidores de la lógica de negocio. Estos servicios realizan labores de transformación y adaptación para poder llamar lógica de negocio de sistema legados.

Esta estrategia ha funcionado en buena medida, pues aplicaciones que no habían sido pensadas para interoperar en esquemas heterogéneos de soluciones, ahora pueden ser incluidas en este tipo de orquestaciones mediante los servicios que las representan. Sin embargo, aun cuando se tiene un nuevo nivel de servicios, las aplicaciones siguen siendo monolíticas y por lo tanto, su migración, reutilización, modificación y adaptación, conlleva grandes tiempos y esfuerzos. SOA en este sentido se ha visto como una opción válida para la exposición de capacidades expuestas por aplicativos legados, sin embargo, también implica tiempos de entrada en producción altos y costos considerables.

El uso de SOA no facilita la incorporación de prácticas como la integración continua y el despliegue continuo, requeridas actualmente en los procesos de desarrollo modernos y en organizaciones más ágiles.

Localización y Orquestación

Otro aspecto a considerar es el uso de tecnologías para la localización y orquestación de servicios. Hasta hace unos años, el bus de servicios (ESB) era una parte fundamental de los ecosistemas de tecnología de las organizaciones, fundamentalmente en esquemas de despliegue on-premises. Los ESB se convirtieron en pieza fundamental para la orquestación de servicios y para la construcción de soluciones basadas en llamados sincrónicos entre servicios. Este esquema funciona, cuando en general todas las aplicaciones involucradas se encuentran desplegadas en un mismo ambiente (por ejemplo: on-premises). Sin embargo, con la llegada de la nube y el despliegue de soluciones en este ambiente, los ESB no encajan perfectamente, ni tampoco ofrecen soluciones claras para arquitecturas no monolíticas y heterogéneas.

Reflexiones

Si bien SOA constituye una alternativa viable y vigente para la construcción de soluciones basadas en servicios, su utilización y evolución a futuro deja muchas inquietudes e interrogantes. Las arquitecturas actuales, se han convertido en algo más híbrido, con el uso de soluciones on-premises mediadas por buses de servicios, y soluciones desplegadas en la nube, orientadas a eventos y coreografía de servicios en lugar de los esquemas tradicionales de orquestación.

Si bien, los activos desarrollados en esquemas tradicionales de SOA y ESBs siguen siendo utilizados y la inversión realizada en estas soluciones debe seguir siendo recuperada, es importante gradualmente ir considerando una migración a soluciones basadas en eventos, colas de mensajes y esquemas de comunicación más orientados a la nube, somo los service-mesh.

Web Oriented Architectures (WOA)

El esquema SOA se ha venido combinando con esquemas más orientados al uso de las tecnologías Web. A este cambio se le denomina el Web Oriented Architectures (WOA), o arquitecturas de solución orientadas al uso de tecnologías como Microservicios, APIs(API Economy), Arquitecturas Orientadas a Eventos, y mecanismos alternativos para la orquestación y comunicación sincrónica de microservicios (Service-Mesh).

API / API Economy

En la actualidad las APIs han pasado de ser un mecanismo de integración, a un elemento crítico tanto a nivel interno dentro de las organizaciones, como mecanismo de interoperabilidad entre socios de negocio y clientes finales.

Portafolios de APIs

Uno de los riesgos, sin embargo, que presenta esta estrategia de integración, es la proliferación de APIs sin control, un bajo esquema de gobierno que no controle la creación de dichas APIs, así como su evolución y mantenimiento.

Un mecanismo para solucionar este riesgo, es diseñar un portafolio de APIs, donde se mantenga una estructura clara y controlada de los diferentes tipos de APIs con los que cuenta la organización. Este portafolio debe ser diseñado con cuidado, buscando satisfacer atributos de calidad, niveles de servicios, monitoreo en ejecución, documentación y entrenamiento, así como control de costos y propiedad.

Actualmente las organizaciones hablan de un término denominado API Economy, que tiene dos visiones. De una parte el negocio va hora en las APIs de la organización, una nueva fuente de ingresos, como consecuencia de la implementación de procesos con clientes y socios de negocio. De otra parte, las áreas de tecnología de las organizaciones, ven en el uso de las APIs, un mecanismo de comunicación y tercerización de sus tareas de desarrollo de software con fábricas de software y desarrolladores.

Proceso de diseño de APIs

Usualmente la organización tiene muchas APIs que van surgiendo con el paso del tiempo. Un portafolio de APIs es más que una colección desordenada de APIs. El portafolio debe ser diseñado con cuidado, de forma que todas las APIs del portafolio sean consistentes unas con otras, reutilizables, descubribles y personalizables.

El Portafolio debe cumplir con requisitos funcionales y no funcionales como : consistencia, reuso, personalización, descubrimiento y longevidad

Consistencia: Un cliente o una solución seguramente usarán varias APIs del portafolio. Es deseable que la salida de una API pueda ser usada como entrada en otra API. Si las entradas esperadas por una API son muy difíciles de lograr a partir de las mismas salidas de otra API del portafolio, se tendrá falta de consistencia. Una estrategia para favorecer la consistencia es abstraer aspectos comunes y llevarlos a otras APIs.

La URI de las APIs debe seguir una estructura común e intuitiva: Nombres de campos y formatos deben ser consistentes a lo largo del portafolio. Las validaciones a las entradas deben hacerse de forma similar en todo el portafolio. Los mecanismos de seguridad deben ser los mismos en todo el portafolio.

Reuso: De ser posible las APIs no debe ser construidas para un cliente específico. Se espera que su diseño e implementación sirvan a múltiples consumidores, al menos parcialmente.

- Reuso de algunas características de las APIs
- Reuso de APIs completas
- Reuso de APIs propias
- Reuso de APIs de terceros

Personalizable: Si bien las APIs no deberían ser diseñadas para un cliente específicamente, si deben poder ser configurables y personalizables para la necesidad particular un cliente o grupo de clientes. Aspectos a personalizar: Formateo de datos, entrega de datos, agrupamiento de datos

Descubrible: Debe ser fácil para un consumidor descubrir las APIs ofrecidas en el portafolio. El portafolio debe ofrecer mecanismos de consulta y descubrimiento de las APIs, de forma que para los consumidores se fácil encontrar el API que buscan.

Longevidad: Se debe buscar que las APIs sean funcionales por la mayor cantidad de tiempo posible. Esto usualmente ocurre cuando una API refleja la propuesta de valor del cliente, dado que estas se mantienen en el tiempo. Desde el punto de vista técnico, se espera que las APIs evolucionen constantemente, pero las firmas de las operaciones deben permanecer estables tanto como sea posible.

Gobernable: Se debe buscar un equilibrio entre un proceso de desarrollo de APIs ágil que permita probar fácilmente nuevas iniciativas de negocio. Reglas y políticas firmes para evitar el desorden en el portafolio de APIs.

El proceso de desarrollo de APIs dentro de las organizaciones debe ser institucionalizado y formalizado. Se debe buscar formalizar el diseño, desarrollo y puesta en producción de las APIs de la organización. Como mínimo los proyectos de diseño y desarrollo de APIs deberían contar con actividades como análisis, desarrollo, operación y retiro.

Análisis

La fase de análisis debe buscar responder preguntas como:

- ¿La organización requiere una API?
- ¿Cuáles funciones cumpliría?
- ¿Cuál es la propuesta de valor de esta API?
- ¿La propuesta será algo fundamental para el negocio?
- ¿Qué requiere el negocio para operar esta API?
- ¿Cuáles métricas se deben cumplir?

El objetivo fundamental de la fase de análisis deberá ser la alineación con las necesidades de negocio. Entendiendo que unidades de negocio serán usuarias de la API y cuáles serán propietarias y financiadores de las APIs. En este sentido es fundamental entender qué tipo de API se desea desarrollar:

APIs Privadas: Usualmente buscan modernizar tecnología y procesos dentro de la organización, optimizar la cadena de valor e integrar sistemas de información legados al interior de la compañía.

APIs tipo consorcio: Buscan usualmente mejorar procesos operativos con socios de negocio y proveedores de servicios. Buscan mejorar la gestión administrativa entre socios y mejorar el relacionamiento y comunicación externa con empresas del ecosistema de la organización. Responder a regulaciones de industria y exponer información parcialmente pública.

APIs Pùblicas: Buscan automatizar procesos con clientes y potenciales clientes, mejorar la experiencia de usuarios y público en general, reducir tiempos de ingreso al mercado y proveer nuevas opciones de integración e interoperabilidad.

Desarrollo

Se deben definir estándares de desarrollo, tecnologías aprobadas por la organización para la implementación de APIs, tipos de APIs a desarrollar, pero sobre todo, tener en cuenta los atributos de calidad que debe exponer el API a desarrollar. Por ejemplo, a nivel de desempeño determinar las cargas de trabajo, niveles de escalabilidad, tiempos de respuesta y volúmenes de datos por unidad de tiempo que podrán ser respondidos ante una consulta. Determinar la utilización de REST como esquema de comunicación o la adopción de nuevas tecnologías como GraphQL.

En este punto la organización debe tomar decisiones relacionadas con:

- Librerías de bloques construcción reutilizables
- Lenguajes de implementación de APIs
- IDEs y depuradores
- Lenguajes de diseño para APIs
- Herramientas para diseño de interfaces para APIs

- Herramientas para la generación de código y documentación

Operación / Plataforma de ejecución

La plataforma de ejecución se encarga de la puesta en producción, despliegue y consumo de las APIs en operación. Esta plataforma se encarga de responder a consultas hechas a las APIs, favoreciendo atributos de calidad como la baja latencia y la escalabilidad.

La plataforma de ejecución se encarga de la ejecución de las APIs. Se encarga de la recepción de peticiones desde los consumidores y de la entrega de las respuestas. Esta plataforma también responsable del cumplimiento de los atributos de calidad, como el desempeño, la disponibilidad y la seguridad. Algunas tácticas de arquitectura que ejecuta la plataforma de ejecución son: balanceo de carga, cache y pool de conexiones. La plataforma ofrece también servicios de monitoreo, log, análisis de cumplimientos de niveles de servicio.

La plataforma de ejecución es responsable igualmente por proveer capacidades para facilitar el despliegue de nuevas APIs y el mantenimiento de APIs existentes. Igualmente, esta plataforma se encarga del manejo de credenciales y configuraciones de múltiples ambientes de ejecución.

Dentro de las decisiones a tomar con la plataforma de ejecución, se encuentra la configuración de múltiples ambientes de ejecución:

- Desarrollo
- Sandbox o simulación
- Testing
- Preproducción
- Producción

Gestión

La plataforma de gestión es utilizada por los proveedores para interactuar con su comunidad de consumidores. La plataforma le debe ofrecer a los proveedores las capacidades de manejo y configuración de APIs.

La plataforma de gestión debe ofrecer:

- Manejo de API: Configuración y reconfiguración de APIs existentes sin necesidad de redespelugues
- Descubrimiento de APIs: Mecanismos para que los clientes obtengan información de las APIs
- Identificación de clientes (Manejo de llaves)
- Información para la comunidad (foros, blogs, redes sociales, bases de conocimiento, etc.)

La plataforma de gestión ofrece adicionalmente documentación interactiva generada a partir de la descripción del API, manejo de versiones, manejo de los niveles de servicios y los cobros asociados

La plataforma de gestión les ofrece a los consumidores del API, mecanismos para explorar y obtener información del Portafolio de APIs, documentación del portafolio, manejo de credenciales y manejo de planes de pago.

La plataforma de gestión deber centrarse en ofrecer servicios de:

- Documentación del portafolio de APIs
- Soluciones basadas en el portafolio de APIs
- Posibilidad de usar el API interactivamente
- Ejemplos de soluciones / código
- Herramientas de apoyo a los usuarios de las APIs

Microservicios

Una segunda tecnología que se muestra como el reemplazo del SOA tradicional son los Microservicios, como componentes de software independientes, que tienen como característica principal su capacidad de despliegue autónomo, orientados a la nube y con mayor adaptación a procesos de desarrollo de entrega continua y despliegue continuo.

La utilización de Microservicios en lugar de las aproximaciones basadas en Servicios de alta granularidad, tipo SOA, sumado al uso de protocolos de comunicación basados en la exposición de APIs tanto REST como GraphQL, hacen que los tiempos de desarrollo y puesta en producción sean menores. Principalmente, la característica de ser independientes en el despliegue, facilita la modificación, cambio y evolución del software, inclusive desarrollado en diferentes tecnologías o lenguajes de programación.

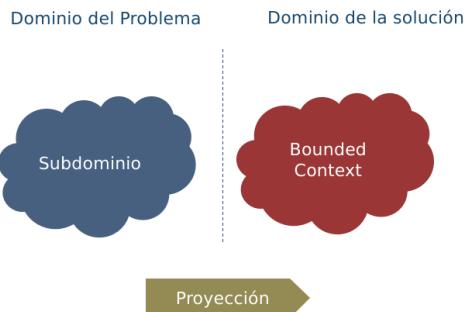
Proceso de desarrollo

Uno de los elementos más críticos en las arquitecturas orientadas a servicios y en los Microservicios como nuevo esquema de trabajo SOA, es la identificación y diseño de los Microservicios. Las empresas actualmente aplican diferentes estrategias para identificar los servicios a desarrollar e implementar. Sin embargo, uno de los riesgos más críticos tiene que ver con el crecimiento descontrolado de Microservicios, que no tienen una alineación clara con el negocio.

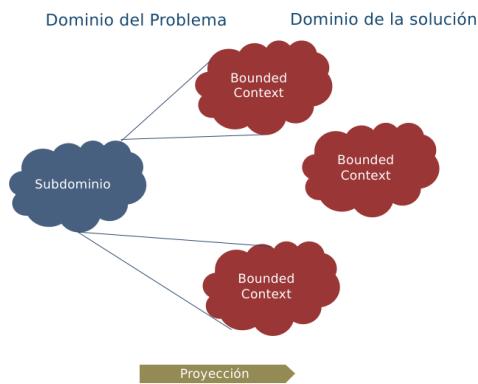
En la actualidad el proceso de identificación y diseño de Microservicios que se utiliza mayoritariamente, tiene que ver con el uso de Domain-Driven Design (DDD) para la identificación de los Microservicios y para la definición de los ecosistemas de servicios a desarrollar. A nivel de diseño detallado de los Microservicios, se utiliza principalmente la arquitectura hexagonal.

Domain Driven Design

El diseño orientado por dominio, busca que los servicios y microservicios no necesariamente busquen maximizar su reutilización, como se ha venido manteniendo en los últimos años. Por el contrario, en el DDD es normal que los servicios no sean reutilizables entre unidades de negocio, que manejan términos, conceptos y vocabularios independientes. En DDD, se busca tener servicios alineados con el negocio, en lugar de reutilizar un mismo servicio en todas las unidades de negocio, de esta forma se logra el desacoplamiento, un menor tiempo de entrega y puesta en producción, así como el uso de técnicas de integración y despliegue continuo.



En DDD la alineación con el negocio es lo fundamental, por eso, los subdominios de negocio o procesos de negocio se relacionan con las unidades de diseño, llamadas contextos acotados o *bounded contexts*.

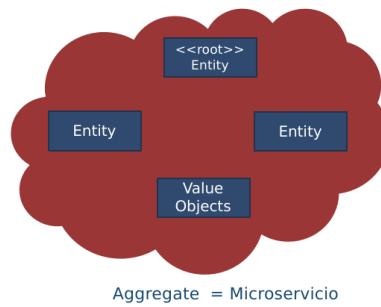


Una unidad de negocio puede tener varios contextos acotados, de forma que varios servicios son identificados para poblar el portafolio de servicios.



Finalmente, un contexto acotado, es a su vez, subdividido en agregados. Los agregados tienen una estructura basada en entidades y objetos de valor. Las entidades representan una operación de negocio indivisible.

Diseño Estratégico
(Bounded Context)



En este esquema de trabajo, usualmente un agregado es un microservicio. Esta técnica de identificación es mucho más precisa, que simplemente el proponer microservicios vistos como pequeñas unidades de software, que técnicamente cumplen con tener un tamaño de líneas de código pequeño, pero que no tienen en cuenta la alineación con el negocio.

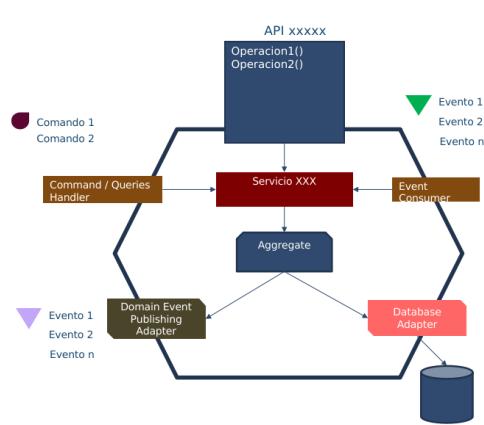
Al interior de un agregado, se identifican los siguientes elementos:

Diseño Estratégico
(Bounded Context)

Entity	Modela un elemento de forma individual. Tiene un identificador único.
Value Objects	Modela un concepto inmutable, el cual no tiene identificador único. Usualmente se usa para describir, cuantificar o medir un Entity
<<root>> Entity	Da el nombre a la agregación y contiene todos los otros elementos (entidades y valores). Es el único punto de entrada a la agregación

Arquitectura Hexagonal

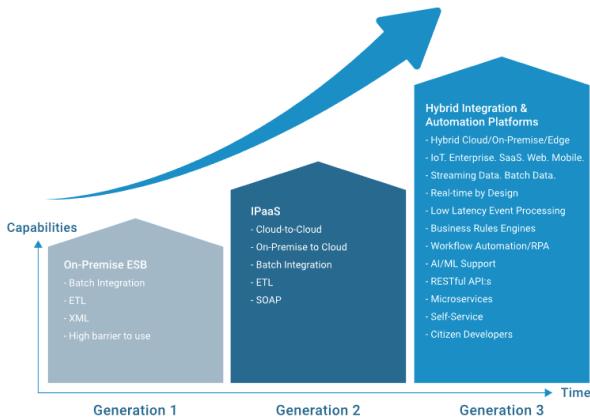
Una vez se tienen identificados los microservicios, y definida su estructura interna, el siguiente paso es identificar, los puntos de interacción con el ecosistemas de servicios. La arquitectura hexagonal sirve para definir estos puntos y para diseñar los mecanismos de comunicación de los microservicios, por ejemplo APIs o colas de comandos, así como para determinar los eventos que escuchará el microservicio, los mensajes a procesar, los eventos que emite el microservicio y la estrategia de almacenamiento de datos del microservicio.



Buses de servicios (ESB)

Sin duda alguno los buses de servicio han sido la piedra angular en las arquitecturas orientadas a servicios en los últimos años. La decisión de arquitectura de orientarse a los servicios implica también que los servicios deben poder ser localizados, que se debe poder intermediar la comunicación e intercambio de datos entre ellos y que deben contar con una infraestructura que les permita orquestar el llamado requerido entre ellos para implementar lógica de negocio.

Los ESBs han sido fundamentales en la evolución de las arquitecturas orientadas a servicios. Sin embargo, un reto crítico actualmente es la decisión de cómo integrar los ESBs a las arquitecturas modernas, no monolíticas y más basadas en tecnologías Web y microservicios.



Tomado de: <https://www.crosser.io/blog/posts/why-the-next-generation-enterprise-service-bus-esb-needs-to-be-intelligent/>

En la anterior figura, se muestra la evolución que han tenido los ESBs. La primera generación, caracterizada por ser infraestructura instalada normalmente en el datacenter de las organizaciones. En esta generación el apoyo a procesos ágiles de desarrollo y prácticas de DevOps era prácticamente impensable.

La segunda generación se basa en infraestructura como servicio, buses desplegados tanto en los datacenters locales como en la nube, con capacidades de intercambio de datos en formato SOAP e integración batch.

La tercera generación de buses, se basa en el despliegue en nubes híbridas, manejo de diferentes tipos de comunicación, ya sea punto a punto, mensajes, o corrientes de datos (streams). Quizás lo más importante a resaltar es el cambio de comunicaciones bloqueantes sincrónicas a comunicaciones basadas en eventos, pero sin los limitantes de latencia de las plataformas de mensajería antiguas. En este caso, hablamos de mecanismos de comunicación orientados a eventos, con capacidades de procesamiento muy altas y con muy baja latencia.

Arquitecturas Orientadas a Eventos

Sin duda alguna el mecanismo de comunicación más utilizado en arquitectura orientadas a componentes y servicios es el mecanismo conocido como request-reply. Es decir, comunicaciones basadas en transferencia sincrónica de información. Este mecanismo implica en general, la comunicación entre un servicio consumidor y un servicio proveedor. El servicio consumidor, debe inicialmente localizar al servicio consumidor (posiblemente con ayuda de un ESB) y ejecutar una petición al servicio proveedor. A través de protocolos como SOAP la comunicación se establece y el intercambio de información comienza (posiblemente en datos usando formatos en XML).

Este mecanismo de comunicación tiene ventajas, pero también presenta desventajas. La principal ventaja es que se trata de una comunicación sincrónica por lo que no es tan complicado seguir el flujo de control entre los diferentes servicios. A este proceso se le conoce como orquestación de servicios, es decir, el flujo de control iniciado por un servicio cliente, que consume en un orden específico los servicios proveedores. La desventaja de este esquema de comunicación es el manejo del error y la disponibilidad. En caso de que un servicio proveedor no esté disponible o falle en la mitad de una operación iniciada por un consumidor, toda la transacción falla. En situaciones de negocio críticas, este comportamiento es deseable, pero en otras ocasiones, implica una falla en la disponibilidad de la solución.

Como medida de mitigación a la comunicación sincrónica basada en comunicaciones tipo request-reply, surgen los mecanismos de intermediación y comunicación basados en mensajes. Por su naturaleza, estos mecanismos son esencialmente asíncronos.

En el esquema de comunicación basado en mensajes, tres conceptos son de suma importancia: Eventos, Queries y Comandos.

Los eventos, en una Arquitectura orientada por eventos (EDA) representan hechos cumplidos, que desean ser notificados de manera asíncrona a los demás servicios participantes de una solución.



Las consultas, representan peticiones a un servicio, que no tienen efecto de borde.

Comunicación basada en eventos



Queries

Consultas que se caracterizan por estar libres de efectos de borde, no alteran el estado del sistema de ningún modo

Debe tenerse en cuenta el volumen de datos esperado como respuesta

Comunicación entre servicios

Los comandos, representan una intención de cambio en el estado de la solución. Los comandos son solicitudes que pueden o no ejecutarse, no se tiene certeza de su cumplimiento. En cualquier caso, el resultado de ejecución de un comando, puede tener como resultado un evento notificando a los interesados el resultado obtenido.

Comunicación RPC



Commands

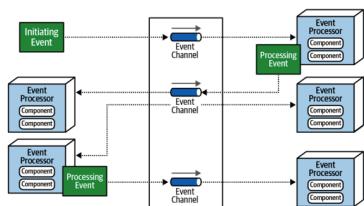
Acciones que pueden cambiar el estado del sistema

Tipicamente se ejecutan de forma síncrona

Operaciones que requieren se ejecutadas y retornar una respuesta

Se sugiere ejecutar dentro de un bounded context

Topologías EDA - Broker



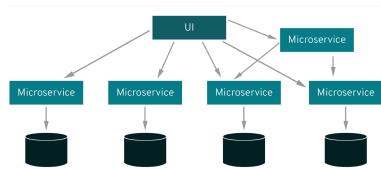
Tomado de: *Fundamentals of software Architecture*. Neil Ford y Mark Richards. O'Reilly, 2020

A partir de estos tres conceptos, se construyen soluciones desacopladas, que en su mayoría se comunican en forma asíncrona. El concepto fundamental, es que no se tienen servicios que realizan comunicaciones síncronas. Toda solicitud de realización de un comando es transmitida como un mensaje mediante una cola de mensajes. Los servicios proveedores reciben las peticiones, ejecutan los comandos o consultas y emiten eventos como resultado. No se tiene ninguna orquestación de servicios, solo coreografía de servicios desacoplado

La figura anterior, muestra el esquema de comunicación fundamental en las arquitecturas orientadas a eventos. Para mejorar los tiempos de respuesta, las aplicaciones suelen replicar parcialmente datos de otros sistemas. Estos datos son utilizados solo para lectura. Debido al retardo en la actualización de los datos, se habla de que existe consistencia eventual entre los sistemas de información.

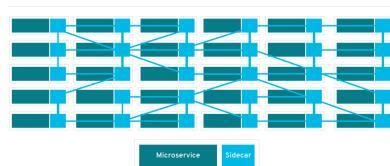
Service Mesh

En los esquemas de comunicación síncrona, donde los servicios localizan otros servicios y envían peticiones síncronas (conteniendo comandos o consultas). Uno de los problemas que pueden surgir, tienen que ver con la necesidad de satisfacer diferentes requisitos de calidad, como la seguridad, la disponibilidad y la escalabilidad. Varias de esos requisitos se resuelven con diferentes componentes, como, por ejemplo, los balanceadores de carga, herramientas de replicación y de autorización y autenticación. Adicionalmente, cada servicio implementa en su lógica, ciertas funcionalidades para manejar este tipo de requisitos, haciendo la complejidad y el mantenimiento de las aplicaciones más complejo. Este escenario se ilustra en la siguiente figura.



Tomado de: <https://www.redhat.com/en/topics/microservices/what-is-a-service-mesh>

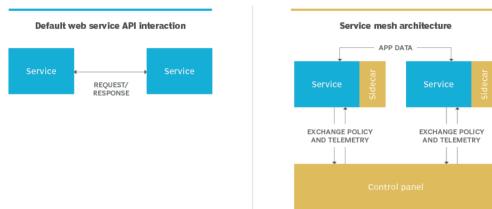
En un esquema service mesh, cada servicio es acompañado de un proxy (llamado sidecar), este proxy se encarga de gestionar toda la seguridad, disponibilidad y escalabilidad, de forma que los servicios no tengan que preocuparse por estos elementos. La siguiente figura ilustra esta idea.



Tomado de: <https://www.redhat.com/en/topics/microservices/what-is-a-service-mesh>

La siguiente figura resume lo antes expuesto, donde además se ilustran los dos elementos fundamentales de este esquema: data plane y control plane.

El término data plane, se refiere a los proxies (sidecars) asociados a cada microservicio. El control plane se encarga de gobernar los proxies asociados a los microservicios.



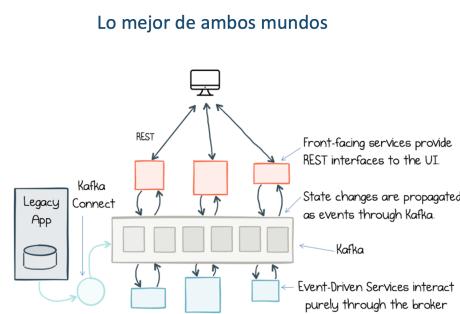
Tomado de: <https://www.techtarget.com/searchitoperations/definition/service-mesh>

Plataforma de Integración Híbrida (HIP)

Sin duda alguna tanto las tecnologías basadas en APIs y consumos de servicios síncronos han sido y seguirán siendo pieza clave en las arquitecturas de integración y orquestación de servicios. Las organizaciones han realizado inversiones altas en la instalación y mantenimiento de infraestructura como ESBs sin contar las múltiples soluciones que hoy día se manejan basadas en este esquema de comunicación.

De otra parte, los mecanismos de sincronización y comunicación asíncronos, basados primordialmente en colas de mensajes, se vienen posicionando como una alternativa cada vez más seria a los buses de servicios tradicionales.

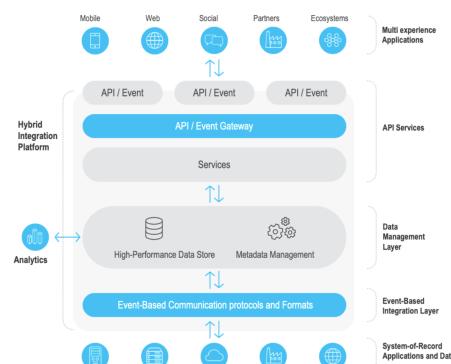
Las plataformas de integración híbridas (HIP) se posicionan cada vez más como una forma de conciliar las estrategias de integración tradicionales y las nuevas tendencias orientadas a eventos. Estas plataformas propendan por la transferencia de estado entre sistema de información utilizando las plataformas de mensajería y la consulta de información de forma síncrona sobre copias con ventanas de consistencia (consistencia eventual)



Tomado de: "Designing Event-Driven Systems". Ben Stopford. 2018. O'Reilly Media Inc.

Hub de Integración Digital (DIH)

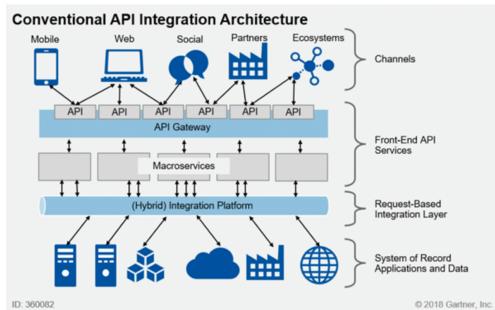
Un DIH es un estilo de arquitectura que busca desacoplar sistemas de información en un ecosistema de TI. No solo a nivel de flujo de control sino también a nivel de datos, adicionando una capa de datos de muy baja latencia. Adicionalmente se propone una mezcla de los tipos de conectores expuestos anteriormente. En primer lugar, se cuenta con un API de desacoplamiento. Los DIH permiten trabajar en esquemas híbridos, con despliegues on-premises y despliegues en cloud.



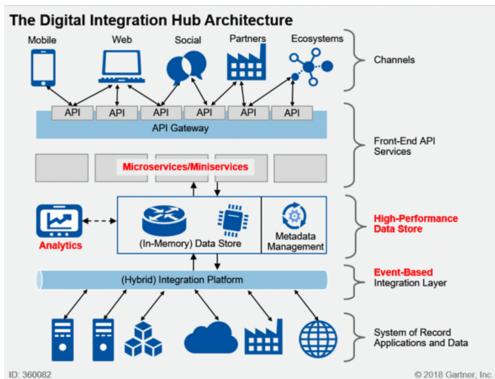
Tomado de: <https://www.gigaspaces.com/blog/digital-integration-hub-dih/>

Esta arquitectura permite la práctica de técnicas DevOps, como la integración y despliegue continuo. Adicionalmente, esta arquitectura se enfoca en la recolección de datos desde múltiples fuentes de datos.

El término DIH fue propuesto por Gartner unos años atrás. En las siguientes figuras, podemos apreciar la comparación realizada en el informe original de Gartner, entre una arquitectura tradicional basada en APIs en la primera figura y una arquitectura basada en DIH en la segunda figura.



Tomado de: <https://mindovermessaging.com/2022/03/29/maximising-your-apis-with-a-digital-integration-hub/>



Tomado de: <https://mindovermessaging.com/2022/03/29/maximising-your-apis-with-a-digital-integration-hub/>

En el esquema basado en DIH la diferencia fundamental radica en una nueva capa ubicada entre el API Gateway y los microservicios y la plataforma híbrida de integración. Esta capa denominada High Performance Data Store (HPDS). Esta capa se alimenta a partir de la plataforma de integración, usualmente utilizando un esquema basado en eventos.

Fase 2: SOA Objetivo

Contenido de los Productos Contractuales, 181-2020

Producto 8: PR8. Arquitectura SOA Candidata para FNA

La arquitectura SOA candidata para el FNA es la continuación de la referencia SOA desarrollada en la Fase 1 del presente diagnóstico. Esta nueva edición de la arquitectura es la base para planear el fortalecimiento SOA y de las capacidades de negocio del segmento de la empresa FNA. La arquitectura candidata recoge principalmente los análisis previos de situación actual SOA del Fondo, análisis de madurez SOA, el análisis de vigilancia tecnológica y estilos SOA, y el de la comparativa con el sector de la industria.

Nota: los análisis de este producto están dirigidos a cumplir los objetivos del proyecto SOA: dependencia de proveedor (OBJ1), fortaleza SOA de las aplicaciones (OBJ2), y tiempo de mercado (OBJ3).

Justificación

La arquitectura SOA candidata del FNA sirve de base para la planeación de cambios de cierre de brecha de la situación actual SOA del Fondo (ver Fase 1).

Contenidos

1. Arquitectura de referencia SOA FNA, versión 2.0
2. Consideraciones para gobierno y adaptabilidad de la arquitectura 2.0
3. Administración de requerimientos de arquitectura para evolución y mantenimiento
4. Procedimientos de evaluación de la arquitectura candidata FNA (req. no funcionales)

Criterios de Aceptación

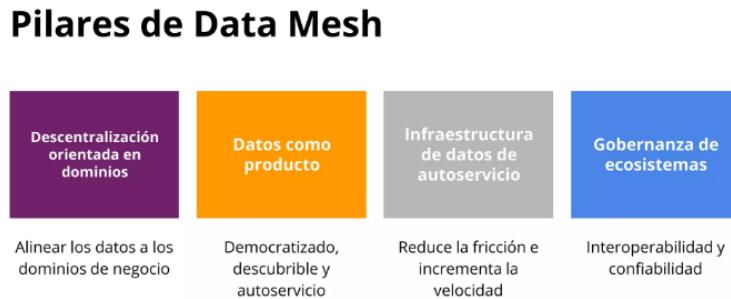
- Arquitectura de referencia SOA FNA, versión 2.0
- Articulación con gobierno SOA del FNA

Arquitectura de Datos Candidata FNA

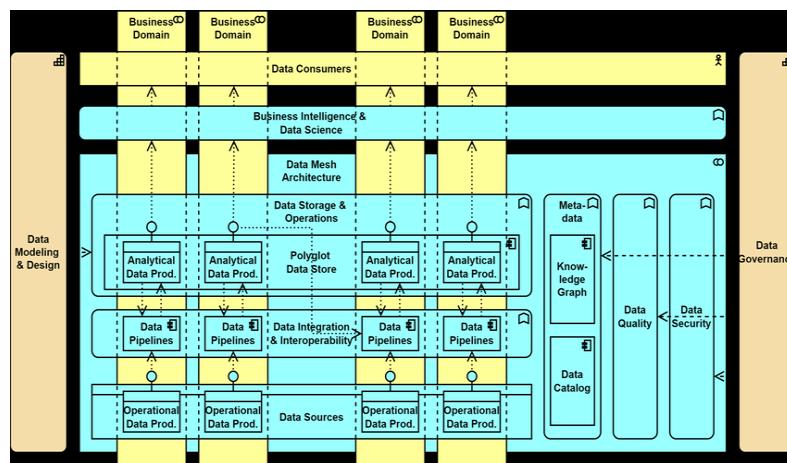
Para la arquitectura To be se propone un enfoque conocido como malla de datos, este es un cambio de paradigma que permite pensar en los datos como un producto. La malla de datos introduce cambios organizativos y de procesos que las empresas necesitarán para gestionar los datos como un activo de capital tangible del negocio.

Una malla de datos tiene como objetivo que los datos sean más accesibles y estén disponibles para los usuarios, conectando directamente a propietarios, los productores y los consumidores de datos. La malla de datos tiene como objetivo mejorar los resultados empresariales de las soluciones centradas en los datos, así como impulsar la adopción de arquitecturas de datos modernas.

Pilares de la malla de datos (Data Mesh)



Arquitectura To Be



Principales Características

Propiedad impulsada por el dominio: el primer principio de un data mesh es transferir el poder de los datos y la propiedad a las manos de los equipos de dominio (es decir departamentos, unidades de negocio, etc.). Ellos serán los dueños de los datos de un extremo a otro, y podrán asegurarse de tener desde las fuentes correctas hasta los procesamientos necesarios y la entrega de los datos para que otros equipos de dominio los aprovechen como productos

Datos como producto: los equipos de dominio son responsables de los datos y también de los productos de datos resultantes. El objetivo es lograr que cada "producto de datos" sea descubierto y utilizable por los consumidores y otros equipos de dominio, y el propietario del dominio es responsable de mantener y actualizar (o desaprobar) estos productos para garantizar la calidad y la precisión.

Infraestructura de autoservicio: que todo sea "autoservicio" hace que nos olvidemos para siempre de las tecnologías complejas y las habilidades de nicho. El Data mesh se basa, por principio, en una gestión de datos mediante una plataforma común y un conjunto de herramientas que cualquier equipo de dominio pueda aprovechar.

Gobernanza federada: si de verdad queremos eliminar controles de acceso necesitamos un equilibrio entre las políticas y controles de gobernanza global y la posibilidad de dominio y creación de productos. Esto es lo que llamamos gobernanza federada y en la práctica es fundamental para garantizar la privacidad, el cumplimiento y la posibilidad de escalar.

Fase 2: SOA Objetivo

Contenido de los Productos Contractuales, 181-2020

Producto 8: PR8a. Portafolio de Iniciativas y Brechas

La técnica del portafolio de brechas es utilizada en el desarrollo de este ejercicio de diagnóstico SOA del FNA para analizar la situación actual SOA (Fase 1 del proyecto) desde la perspectiva de proyectos ejecutables. La idea principal del análisis de brecha es resaltar los proyectos o ítems omitidos, o por definir, entre la situación actual SOA del FNA y la situación objetivo, incluso cuando esta última está todavía en evolución. Lo anterior da origen al portafolio de iniciativas priorizadas que ser ejecutadas cierra las brechas analizadas.

Nota: los análisis de este producto están dirigidos a cumplir los objetivos del proyecto SOA: dependencia de proveedor (OBJ1), fortaleza SOA de las aplicaciones (OBJ2), y tiempo de mercado (OBJ3).

Justificación

Asegurar que la arquitectura SOA del Fondo apegada a la [Vista de Segmento FNA](#) soporte al procesamiento de información, a los sistemas de información, a las capacidades de negocio, y tecnologías requeridas para cumplir los objetivos de este ejercicio de diagnóstico SOA. Dar continuidad y orden al impacto y realización de los cambios entre evoluciones de la arquitectura SOA.

Contenidos

1. Arquitectura SOA candidata para el segmento FNA (vista de segmento)
2. Matriz de brechas de las partes del segmento FNA
3. Bloques de cambios para el segmento FNA: ítems por retener, rediseñar, actualizar, retirar
4. Análisis de impacto y dependencia entre los cambios en el segmento de la empresa
5. Ficha descriptiva de iniciativas de mejora para el segmento FNA (justificación)
6. Hoja de ruta preliminar hacia un objetivo de mejora

Criterios de Aceptación

- Lista de cambios para el segmento FNA: ítems incluidos, por mejorar, nuevos y eliminados
- Hoja de ruta preliminar hacia un objetivo de mejora