| Tema | Diseño detallado y vistas funcional, despliegue, información, integración y tecnología​: **Consideraciones para implementar la arquitectura de referencia** |
| --- | --- |
| Palabras clave | SOA, Arquitectura de referencia, características, implementación, hoja de ruta, |
| Autor |  |
| Fuente |  |
| Versión | **1.b5d67e5** del 29 Jul 2023 |
| Vínculos | [Ejecución Plan de Trabajo SOA](onenote:#N001d.sharepoint.com); [Procesos de Negocio FNA](onenote:#N003a.com) |

# Consideraciones de Diseño de las Arquitecturas de Referencia del FNA

En esta sección ampliaremos las consideraciones dadas al FNA para la creación y mantenimiento de arquitecturas de referencia. En este caso daremos información que advierta las incorreciones de diseño de las que debe alejarse la empresa. Estos diseños problemáticos que describiremos aquí deben ser usadas para vigilar cada cambio en las arquitecturas del FNA, así como en las evaluaciones, y comparar estos cambios contra estas advertencias para luego determinar si los diseños (y soluciones e implementaciones) requieren acciones de remediación.

## Diseños Centrados en el Proveedor

El principal diseño del que debe alejarse el FNA es todo aquel el que aumente la dependencia de proveedor. Es decir, *los diseños centrados en el proveedor son los problemáticos*. Ejemplo de esto, y así diagnosticado en Fase I de esta consultoría, es el ERP Cobis. Este reune más de nueve módulos de software, la mayoría relacionados con la misión del Fondo Nacional. Es por tanto un diseño problemático que el ERP esté en el centro de la empresa y tender relaciones directas a este desde otros puntos de la empresa.

Nota: en este contexto, nos referimos a una perspectica tecnológica de la empresa.

En el diagrama siguiente presentamos esta consideración de forma visual en la que queremos se aprecie el problema de tender relaciones directas desde los procesos de negocio a un proveedor tecnológico, COBIS.



Figure 1: Consideraciones de Diseño FNA. Acoplamiento y Dependencia al proveedor.

En la imagen: las relaciones directas que los procesos de negocio puedan tener con un proveedor son la causa de dos problemas: el acomplamiento al proveedor, lo que a su vez contribuye al problema de diseño que hemos llamado dependencia al proveedor.

Este mismo problema se presenta de forma indirecta cuando existe un intermediario entre las partes. Siguiendo con el ejemplo ilustrado arriba, un intermediario entre los procesos de negocio y el proveedor, sea un bus empresarial, un integrador, proxie, o algún intermediario cualquiera de sus formas, comporta el mismo problema que denunciamos en este diseño centrado en el proveedor. La presencia de un intermediario no elimina la complicación del acomplamiento, solo lo soslaya, y al contrario, lo acrecienta.

Nota: este mismo problema se presenta de forma indirecta cuando existe un intermediario entre las partes. La presencia de un intermediario no elimina este problema, solo lo esconde, y por tanto, lo acrecienta.

## Diseños de Capas Generales

El diseño de capas, si bien es uno con los que inicia una arquitectura, comporta los problema de la generalidad: no representa en realidad a ninguna arquitectura. Si consideramos a la arquitectura como la elección de una decisión de diseños sobre otra, las separación por capas genéricas no alcanza a realizar nada de eso. Este diseño no profundiza en ningún aspecto o problema a resolver, pero en cambio da la sensación de que sí, y ese es el problema.

En este diagrama presentamos esta consideración de forma visual. Debemos observar que el diseño de capas sí organiza y prepara a una posterior arquitectura ante un problema, sea de flexbilidad, despliegue, etc., pero no está tomando decisiones más allá de eso.



Figure 2: Consideraciones de Diseño FNA. Capas genéricas, sin arquitectura. Preparan (organizan) un sistema información, servicio o componente de software para la arquitectura.

El FNA debe alejarse de los diseños generales que funcionan solamente como una organización base que carece de arquitectura (decisiones). Estos diseños no aportan ni resuelven un problema específico y son apenas una forma de organización de las partes de un sistema de información, pero no una arquitectura como la conocemos en este contexto.