## Dominio de Infraestructura FNA

Para la arquitectura SOA objetivo se propone un enfoque de computación en la nube donde se hace uso de componentes IaaS y PaaS, de infraestructura y plataforma en la nube. Dentro de la oferta actual de componentes se pueden encontrar soluciones que permiten garantizar escalabilidad, disponibilidad, cumplimiento regulatorio y menores tiempos de salida al mercado.

### Arquitectura Objetivo

En respuesta a las necesidades determinadas en los diagnósticos de Fase I, proponemos una base de infraestructura futura, pero compatible con el contexto actual del FNA. Por tanto, esta propuesta se centra en el estilo de arquitectura SOA mejorado con la malla de servicios y microservicios elásticos (auto escalado).



Infraestructura de arquitectura mallas de servicios SOA. Característica principal: los servicios elásticos (auto escalado).

A continuación resaltamos algunas características de interés respecto de los problemas que esta arquitectura soluciona.

### Características de la Arquitectura Tecnológica 2.0 del FNA

**Arquitectura sin servidor para la interfaz web**: Distribución de contenidos con baja latencia y altas velocidades de transferencia usando CDN hospedando el contenido estático de la aplicación en un servicio de almacenamiento de objetos, eliminando la necesidad de utilizar servidores en esta capa.

**Enfoque de microservicios**: Permite obtener capacidad de cómputo particular según sea requerido para soportar la carga.

**Exposición de la funcionalidad de negocio a través de API Gateway**: Entrega las solicitudes a los microservicios, habilitando la gestión de API para establecer modelos de Calidad en el Servicio (QoS) y controlando la autenticación y autorización.

**Orquestación de contenedores**: Los administradores de Kubernetes permiten automatizar tareas clave como los parches, el aprovisionamiento de nodos y las actualizaciones. El gestor de Kubernetes por lo general permite gestionar dinámicamente la creación o terminación de nodos y contenedores en base a la demanda.

**Service mesh**: Permite distribuir la carga proveniente de API Gateway, obteniendo mayor visibilidad y controles uniformes del tráfico de red.

**Servicios administrados**: Se aprovechan para elementos tales como la gestión de cache y bases de datos (relaciones y no relacionales). Con esto se evitan tareas administrativas, como el aprovisionamiento de hardware, parches de software, configuración, ajustes, recuperación de fallos y backups.

**Integración de sistemas**: A través de un enlace Virtual Private Network (VPN) La funcionalidad se expone a través del Bus de Servicios, habilitando un modelo híbrido de aplicaciones nativas de nube y al mismo tiempo de arquitectura orientada a servicios (SOA).