



PERÚ

Ministerio  
de Economía y Finanzas

Organismo Supervisor  
de las Contrataciones  
del Estado

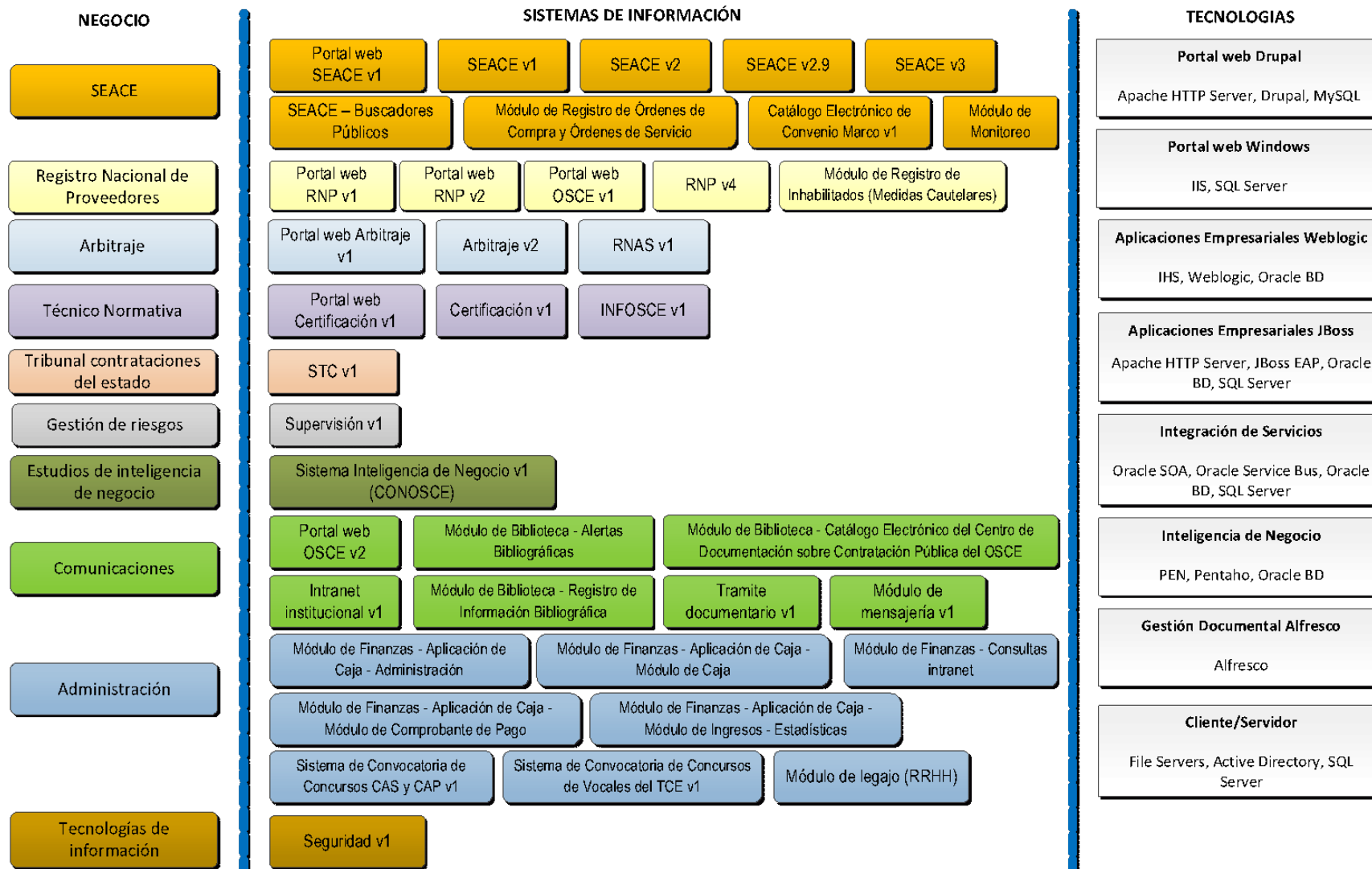


# DIAGNÓSTICO Y PROPUESTA DE VISIÓN DE ARQUITECTURA DE SOFTWARE DE LOS NUEVOS SISTEMAS DEL OSCE

**Unidad de Gestión de Desarrollo de Software**

# Diagnóstico de las arquitecturas de software

## Sistemas actuales del OSCE



## Sistemas actuales del OSCE

- **PORTALES WEB:** Portal web SEACE v1, Portal web Certificación v1, Portal web Arbitraje v1, Portal web RNP v1, Portal web RNP v2, Portal web OSCE v1, Portal web OSCE v2, Intranet institucional v1.
- **SEACE:** SEACE v1, SEACE v2, SEACE v2.9, SEACE v3, Módulo de Registro de Órdenes de Compra y Órdenes de Servicio, Catálogo Electrónico de Convenio Marco v1, Módulo de Monitoreo.
- **RNP:** RNP v4, Módulo de Registro de Inhabilitados (Medidas Cautelares).
- **INTELIGENCIA DE NEGOCIO:** Sistema Inteligencia de Negocio v1 (CONOSCE).
- **SISTEMAS DE SEGURIDAD:** Seguridad v1.
- **SISTEMAS PRIMARIOS:** Arbitraje v2, RNAS v1, STC v1, Supervisión v1, Certificación v1, INFOSCE v1, SICAN v1, SDA v1.
- **SISTEMAS DE APOYO:** Módulo de Finanzas - Aplicación de Caja – Administración, Módulo de Finanzas - Aplicación de Caja - Módulo de Caja, Módulo de Finanzas - Consultas intranet, Módulo de Finanzas - Aplicación de Caja - Módulo de Comprobante de Pago, Módulo de Finanzas - Aplicación de Caja - Módulo de Ingresos – Estadísticas, Sistema de Convocatoria de Concursos de Vocales del TCE v1, Módulo de legajo (RRHH), Sistema de Convocatoria de Concursos CAS y CAP v1, Módulo de mensajería v1, Módulo de Biblioteca - Alertas Bibliográficas, Módulo de Biblioteca - Catálogo Electrónico del Centro de Documentación sobre Contratación Pública del OSCE, Módulo de Biblioteca - Registro de Información Bibliográfica, Tramite documentario v1

## Análisis de la arquitectura de software actual

- No se ha identificado que los lineamientos de la arquitectura de software de la institución estén aplicados en su totalidad en todos los sistemas del OSCE.
- Los sistemas implementados siguen diferentes diseños arquitecturales según la plataforma en la que están desplegados.
- Existen documentos arquitecturales que deben ser actualizados para incluir el uso de nuevas tecnologías.
- Existen arquitecturas de referencia para implementaciones en el OSCE mas no se ha identificado arquitecturas concretas por cada tipo de implementación requerida.
- Las aplicaciones software son diseñadas bajo un enfoque de organización funcional no se ha identificado aplicaciones que faciliten el acceso y la navegación a la información que los usuarios (proveedores, operadores logísticos, etc.) realizan a múltiples áreas del OSCE.
- Se ha identificado que un usuario del OSCE (proveedor, operador logístico, etc.) puede tener diferentes credenciales de acceso según los sistemas del OSCE en el cuales realiza alguna actividad, esto requiere tener replicada la gestión de seguridad en diferentes aplicaciones software y aumenta las tareas que el usuario puede realizar en temas de seguridad cuando tiene más de una credencial.
- Existen múltiples plataformas tecnológicas para almacenar los documentos registrados a través de los sistemas del OSCE lo cual no permite homologar funcionalidades como las de realizar búsqueda en contenido indexado.

## Análisis de la arquitectura de software actual

- Se ha identificado software y/o tecnologías con soporte técnico de fabricante finalizado o próximo a finalizar, estos son:
- ✓ Active Server Page 6 (UA: 26/02/2012)
  - ✓ Microsoft Internet Information Services 6.0 (IIS 6) (UA: 26/02/2012)
  - ✓ Microsoft SQL Server 2000 (UA: 26/02/2012)
  - ✓ Oracle Database 11g R2 (11.2) (UA: 26/02/2012)
  - ✓ Oracle weblogic Server 11g R1 (10.3.6) (UA: 26/02/2012)
  - ✓ JBoss 3.2.5 (UA: 02/03/2006)
  - ✓ JBoss AS 6.1 (UA: 16/08/2011)
  - ✓ Apache Http Server 2.0.x (UA: 02/07/2013)

## Análisis de la arquitectura de software actual

- Se ha identificado la utilización de software/tecnologías /frameworks que dificultan incorporar estándares y/o técnicas y/o especificaciones:

PRODUCTO	DIFICULTAD
Primefaces 3.5	No permite implementar el diseño web responsivo. Soporta JSF 2.0 siendo la versión vigente JSF 2.2 que entre otras funcionalidades incorpora la gestión de flujo de formularios, soporte a multiplantillas, integración a HTML 5.
IceFaces 1.8	No permite implementar el diseño web responsivo. Soporta JSF 1.2 siendo la versión vigente JSF 2.1 entre otras funcionalidades incorpora la gestión de flujo de formularios, soporte a multiplantillas, integración a HTML 5.
Hibernate 3.6.6	Da soporte a JPA 1.0 siendo la versión vigente JPA 2.1 entre otras funcionalidades mejora las implementaciones de conversores, actualización en lote, utilización de procedimientos almacenados, contextos persistentes asíncronos
Servlet 2.4, 2.5	Se utilizan Servlets 2.4 y Servlet 2.5 siendo la versión vigente Servlet 3.1, el cual entre otros permite mejorar la gestión de comunicación de HTTP entre el cliente y el Servlet y con técnicas como Non-blocking IO mejora el desempeño en aplicaciones altamente concurrentes
Apache Http Server 2.0.x	Apache Http Server 2.0.x tiene última liberación 02/07/2013, la versión menor más actualizada es Apache Http Server 2.4.x que incorpora mejoras en lo relacionado SSL y desempeño a entorno de alta concurrencia.
JBoss 3.2.5	JBoss 3.2.x tiene última liberación 02/03/2006 y da soporte a j2ee 1.3 (JDK 1.3, JSP 1.2, EJB 2.0, Servlet 2.3, JMS 1.0, JDBC 2.0, etc.), tecnología descontinuada.
JBoss AS 6.1	JBoss 6.1.x tiene última liberación 16/08/2011 y da soporte a java EE 6 (JDK 1.6, JSF 2.0, JSP 2.2, EJB 3.1, Servlet 3.0, JMS 1.1, JPA 2.0, etc.), la versión actual de java EE es la 7.
Weblogic Server 11g	Weblogic Server 11g tiene última liberación 26/02/2012 y da soporte a java EE 5 (JDK 1.6, JSF 1.2, JSP 2.1, EJB 3.0, Servlet 2.5, JMS 1.1, JPA 1.0, etc.), la versión actual de java EE es la 7.

# PROPUESTA DE LA VISIÓN DE ARQUITECTURAS DE SOFTWARE PARA EL OSCE



## Descripción de problemas - Stakeholders y sus preocupaciones

- La información de los usuarios del OSCE está distribuida en múltiples sistemas esto obliga a los usuario a interactuar con diferentes softwares de aplicación sean estas aplicaciones de registro o de consulta de información.
- Los softwares de aplicación que incorporan funcionalidades de búsqueda de los sistemas del OSCE como el SEACE suelen presentar inconvenientes de tiempos de respuesta en temporadas donde se incrementa las consultas que realizan los usuarios.
- Se debe mejorar la interacción en funcionalidades de consultas y/o búsquedas las cuales están diseñadas para que el usuario ingrese valores para cada atributo con el que va buscar y/o consultar, este diseño debe simplificarse para que con un solo valor el usuario pueda realizar una búsqueda general en diferentes atributos.
- Existen usuarios que tienen diferentes credenciales en diferentes sistemas del OSCE y/o que con una credencial deben autenticarse en diferentes softwares de aplicación, es necesario que con un solo proceso de autenticación el usuario pueda ingresar a Los softwares de aplicación que conforman todos los sistemas del OSCE.
- Aplicaciones web del OSCE como el RNP v4 son dependientes del navegador internet explorer, esto perjudica a los usuarios que acceden a los sistemas del OSCE desde dispositivos donde no se pueden hacer uso de internet explorer como el caso de los dispositivos móviles o dispositivos de escritorio con versiones de Windows actualizados.
- Las aplicaciones web del OSCE no tienen un diseño web responsivo que permita al usuario interactuar a través de diferentes dispositivos de escritorio y móviles.

## Descripción de problemas - Stakeholders y sus preocupaciones

- Existen softwares de aplicación con diseños de formularios web densos en contenido de información lo cual repercute en una interacción poco amigable cuando el usuario accede a través de dispositivos móviles.
- No se han implementado aplicaciones móviles que muestren la información del OSCE y a través del cual se pueda notificar al usuario de la información de su interés, esto bajo el enfoque que sea la información actualizada la que vaya al usuario y no sea el usuario el que venga a buscar la información actualizada.
- Se usan versiones de software que están con soporte técnico de fabricante vencido, extendido o próximo a vencer.
- Existen sistemas cuya tecnología no permite utilizar nuevos diseños, estándares y/o especificaciones que mejorarían la experiencia de usuario en Los softwares de aplicación que conforman los sistemas del OSCE.
- No se cuenta con plataforma tecnológica que permita incorporar la cultura de devops en el ciclo de vida del desarrollo de software, teniendo que realizarse muchas actividades de manera semi-automática o manual.

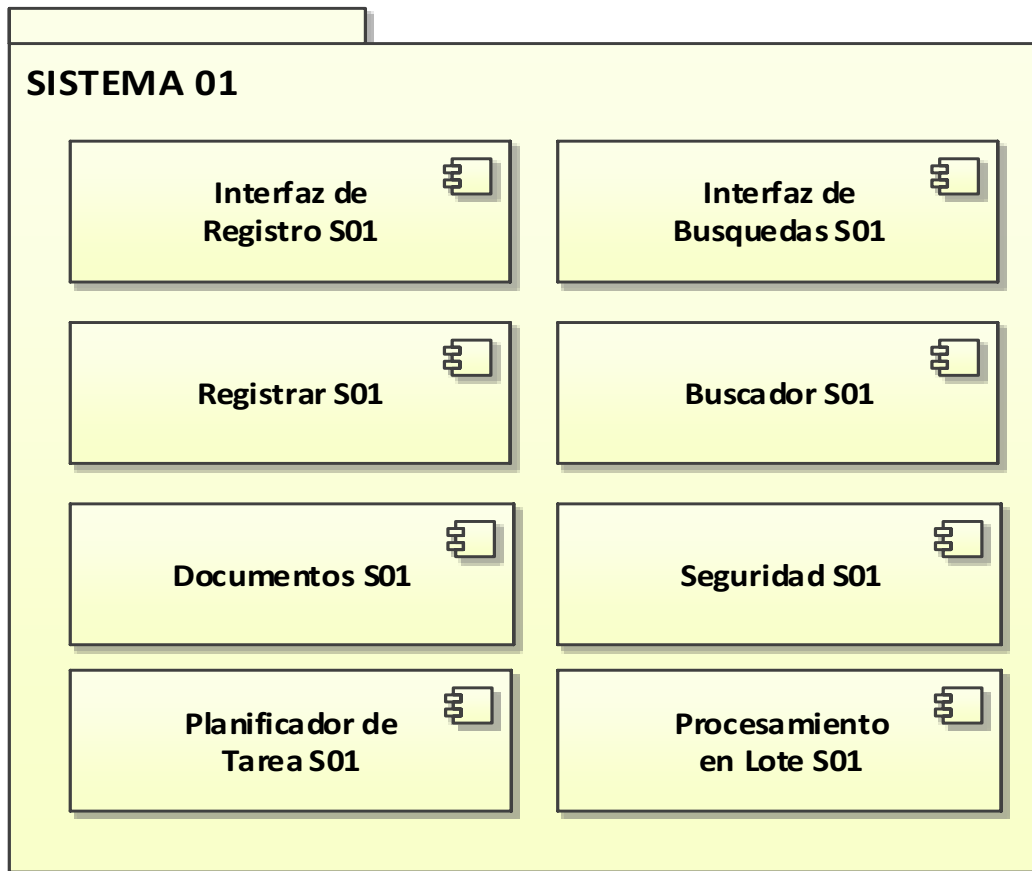
## Objetivos específicos

- Permitir que Los softwares de aplicación del OSCE sean compatibles con tecnologías de acceso común a los interesados del OSCE.
- Mejorar las funcionalidades de búsquedas y consultas de los diferentes sistemas del OSCE.
- Permitir desarrollar softwares de aplicación que consolide el acceso de la información de los interesados del OSCE.
- Facilitar el intercambio de información entre los sistemas de las diferentes áreas organizacionales OSCE.
- Proporcionar a los interesados del OSCE acceso único a las diferentes softwares de aplicación que conformen los sistemas del OSCE.
- Permitir que los sistemas del OSCE realicen notificaciones de información a los dispositivos de escritorios y móviles de los interesados del OSCE.

## Principios TI

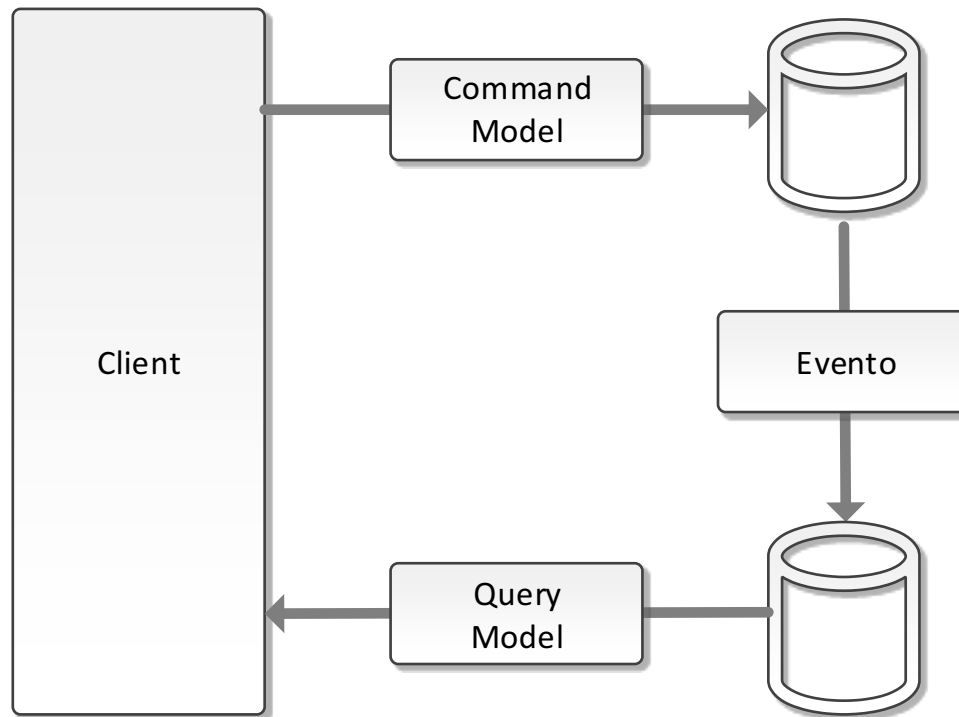
- El uso de los recursos debe ser optimo, teniendo en cuenta criterios de pertinencia y reutilización.
- Se debe fortalecer los esquemas que estandaricen y faciliten el intercambio de información, el manejo de fuentes únicas y la habilitación de servicios entre áreas organizacionales y entidades externas del OSCE.
- Se deben componer nuevas soluciones y servicios sobre lo ya construido y definido, con la participación de todas aquellas áreas organizacionales del OSCE.
- Se debe permitir la evolución continua y el ajuste de todos los componentes y dominios que integran la arquitectura del software, sin perder calidad ni articulación.
- La definición, implementación y verificación de controles de seguridad de la información deben ser incluidos en los diseños de los sistemas del OSCE.
- Para el planteamiento e incorporación de soluciones tecnológicas se debe tener en cuenta las recomendaciones, conceptos y normativas de los organismos internacionales competentes en la materia.
- Es deseable que las herramientas utilizadas dispongan de licencia open source y/o software libre, y además puedan ser ofrecidas como Software Público.
- Todas las interfaces expuestas entre sistemas del OSCE deben cumplir con las disposiciones de accesibilidad y usabilidad.
- Deben generarse servicios que proporcionen funciones generales a las necesidades de negocio del OSCE.
- El intercambio de archivos a través de la plataforma de servicios deberá realizarse usando el estándar MTOM.
- Todos los sistemas deben tener documentada su arquitectura.

## Generalización de sistemas actuales del OSCE



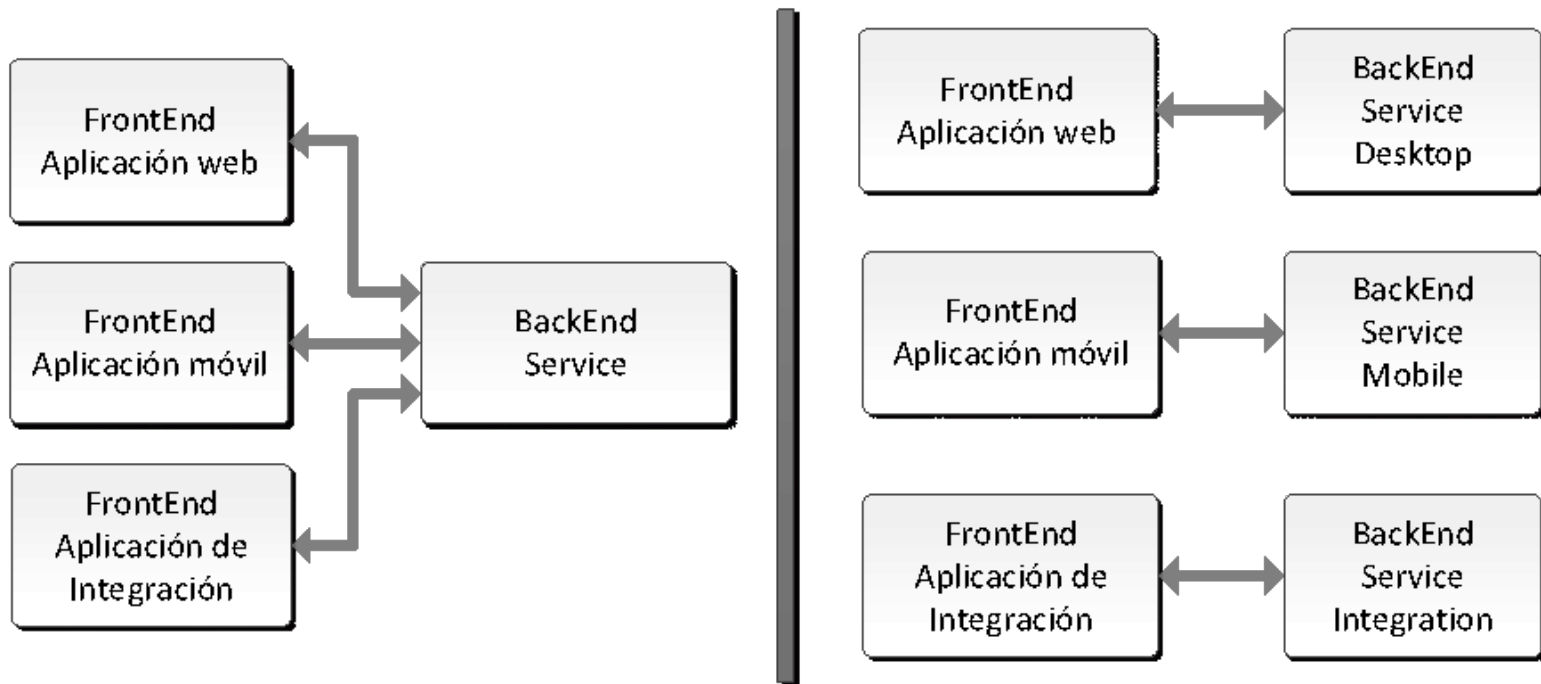
## Patrones - CQRS

La segregación de responsabilidad de consultas y comandos (CQRS) separa el código que modifica el estado del código que lee el estado.



## Patrones - Backends for Frontends (BFF)

Cree servicios back-end separados para ser consumidos por aplicaciones frontend o interfaces específicas. Este patrón es útil cuando desea evitar la personalización de un único backend para múltiples interfaces. Esta relacionado a API Gateway



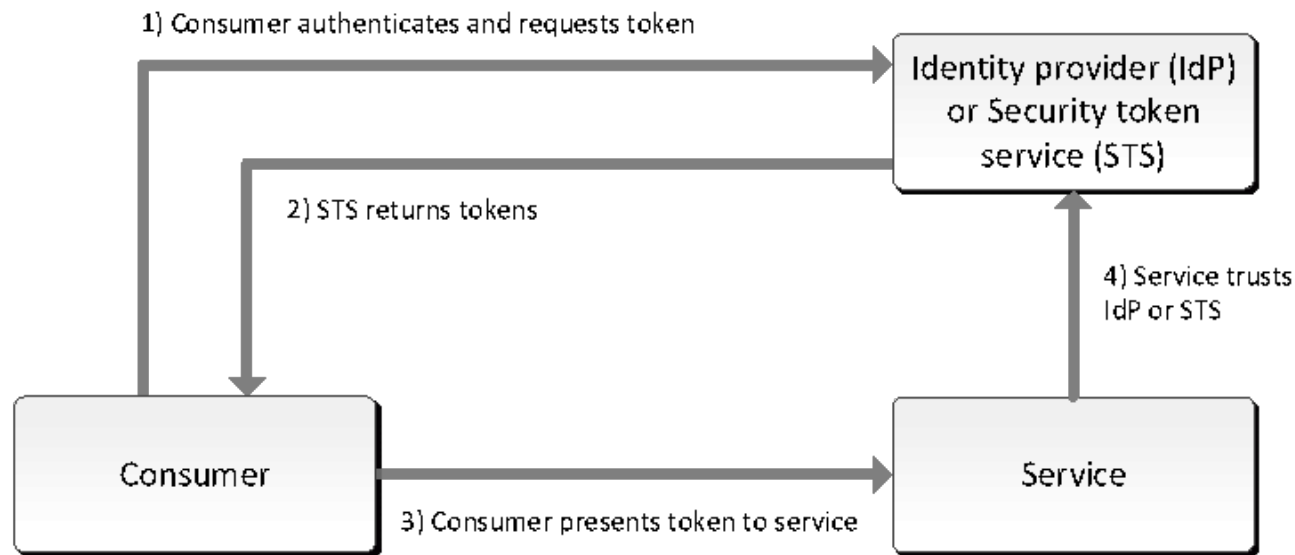
<https://docs.microsoft.com/en-us/azure/architecture/patterns/backends-for-frontends>

<http://samnewman.io/patterns/architectural/bff/>

<http://microservices.io/patterns/apigateway.html>

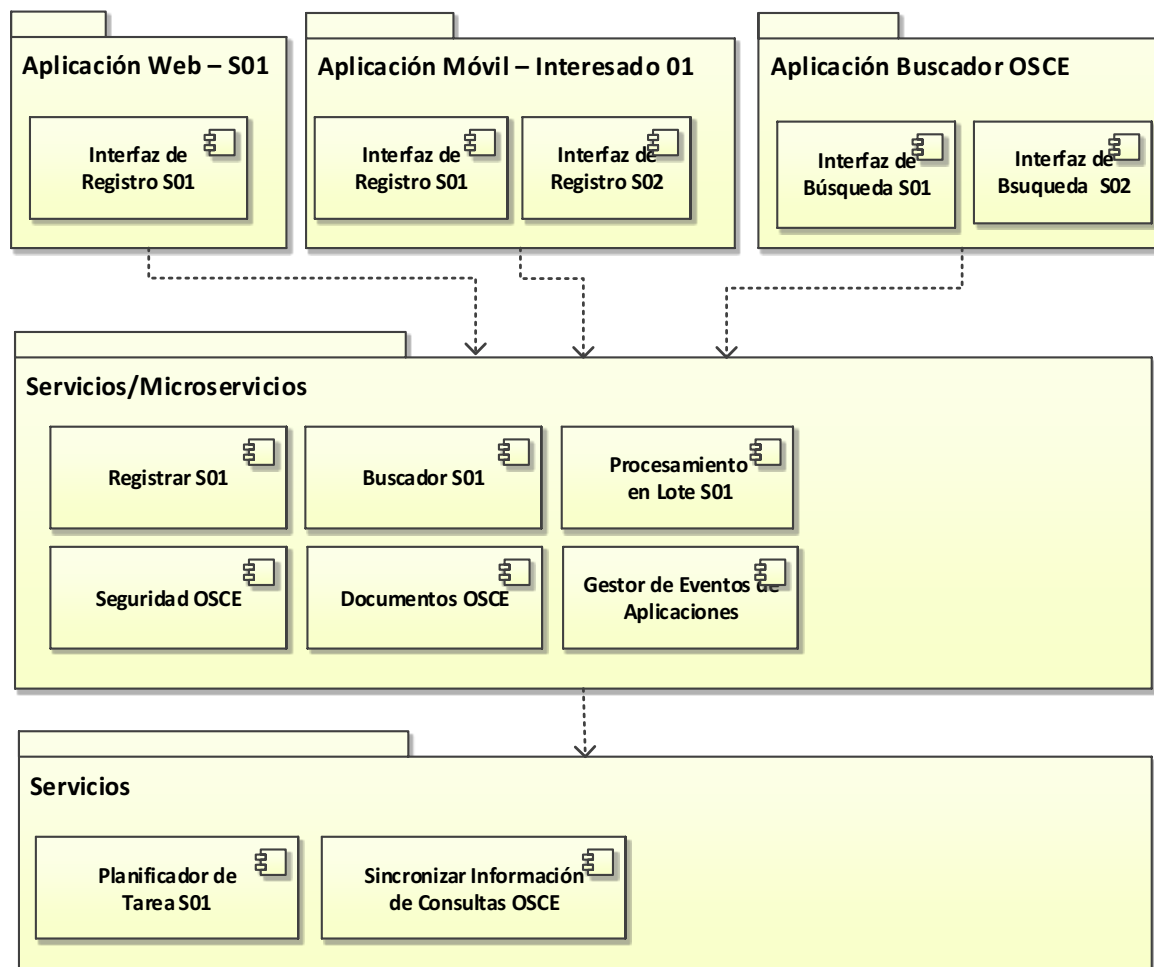
## Patrones - Federated Identity

Delegue la autenticación a un proveedor de identidad externo. Esto puede simplificar el desarrollo, minimizar el requisito de administración del usuario y mejorar la experiencia del usuario de la aplicación



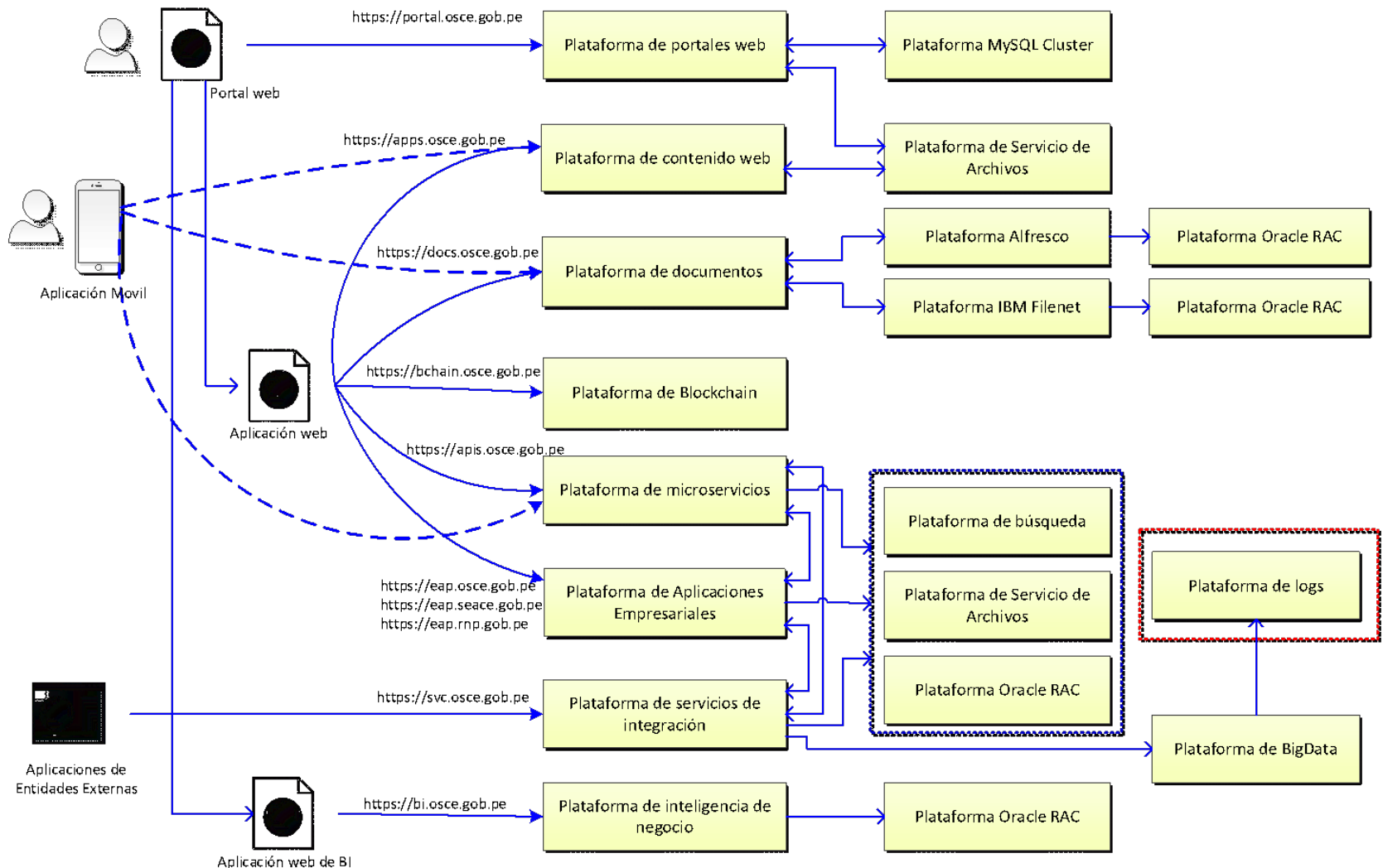


## Generalización de propuesta de sistemas del OSCE



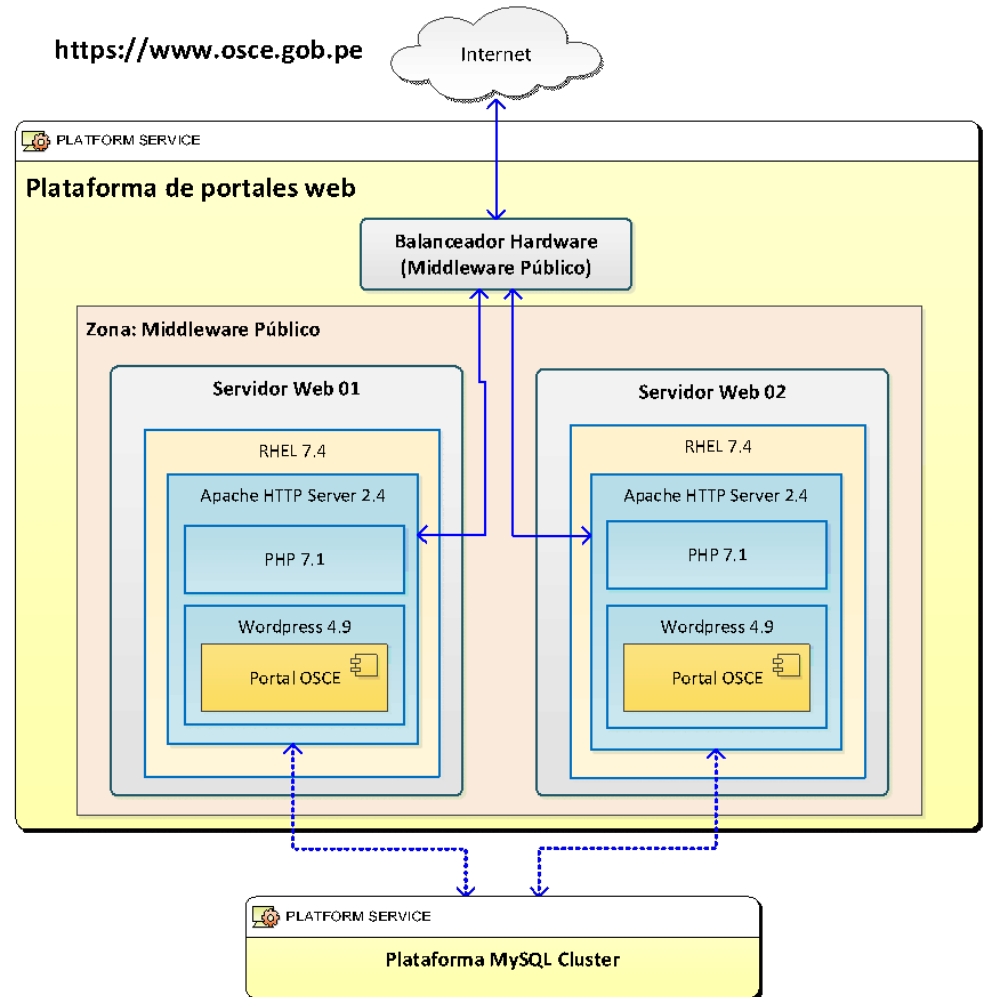
## Propuesta de plataformas tecnológicas

### Plataformas para sistemas OSCE



## Plataforma de Portales

- RHEL 7.4
- Apache HTTP Server 2.4
- PHP 7.1
- Wordpress 4.9



## Plataforma de Portales – RHEL 7.4

Red Hat Enterprise Linux también conocido por sus siglas RHEL es una distribución comercial de GNU/Linux desarrollada por Red Hat. Es la versión comercial basada en Fedora.

### ➤ Características:

- ✓ Las variantes son versiones personalizadas de Red Hat Enterprise Linux Server para casos prácticos, aplicaciones y arquitecturas de hardware específicos.
- ✓ Red Hat Enterprise Linux Atomic Host; es un sistema operativo (SO) seguro con impacto mínimo en la memoria y optimizado para ejecutar contenedores Linux.

### ➤ Escenarios de uso:

- ✓ Sistema operativo para servidores que proporcionen servicios TI en arquitecturas x86\_64, power, System Z.

### ➤ URL's de referencia:

- ✓ <https://www.redhat.com/es/technologies/linux-platforms/enterprise-linux/options>
- ✓ <https://www.forbes.com/sites/adrianbridgwater/2017/08/03/red-hat-pushes-linux-towards-the-four-footprints-of-technology/#67102c9f44d5>

## Plataforma de Portales – Apache HTTP Server 2.4

Apache ha sido ampliamente adoptado porque es completamente de código abierto, y su arquitectura modular se puede personalizar para adaptarse a diferentes necesidades.

### ➤ Características:

- ✓ Debido a que Apache aísla su funcionalidad principal (cómo maneja las solicitudes, los enchufes, etc.) del resto de sus módulos, los desarrolladores han podido crear y contribuir con módulos sin afectar la funcionalidad central del servidor web.

### ➤ Escenarios de uso:

- ✓ Servidor web con soporte a lenguajes como PHP, Python, Perl
- ✓ Balanceador de carga.
- ✓ Proxy reverso.

### ➤ URL's de referencia:

- ✓ <https://httpd.apache.org/>
- ✓ [https://w3techs.com/technologies/history\\_overview/web\\_server](https://w3techs.com/technologies/history_overview/web_server)

## Plataforma de Portales – PHP 7.1

Acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

### ➤ Características:

- ✓ Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- ✓ Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).

### ➤ Escenarios de uso:

- ✓ Scripts del lado del servidor. Este es el campo más tradicional y el foco principal.
- ✓ Scripts desde la línea de comandos. Se puede crear un script de PHP y ejecutarlo sin necesidad de un servidor o navegador.

### ➤ URL's de referencia:

- ✓ <http://php.net>
- ✓ <http://php.net/manual/es/index.php>
- ✓ [https://w3techs.com/technologies/history\\_overview/programming\\_language](https://w3techs.com/technologies/history_overview/programming_language)

## Plataforma de Portales – Wordpress 4.9

WordPress es un sistema de gestión de contenidos o CMS (por sus siglas en inglés, Content Management System) enfocado a la creación de cualquier tipo de página web

### ➤ Características:

- ✓ La estructura y diseño visual del sitio depende de un sistema de plantillas independiente del contenido.
- ✓ Bloques con funciones específicas por medio de complementos cuya publicación se realiza por medio de widgets.
- ✓ La gestión y ejecución corre a cargo del sistema de administración con los complementos y widgets que usan las plantillas.

### ➤ Escenarios de uso:

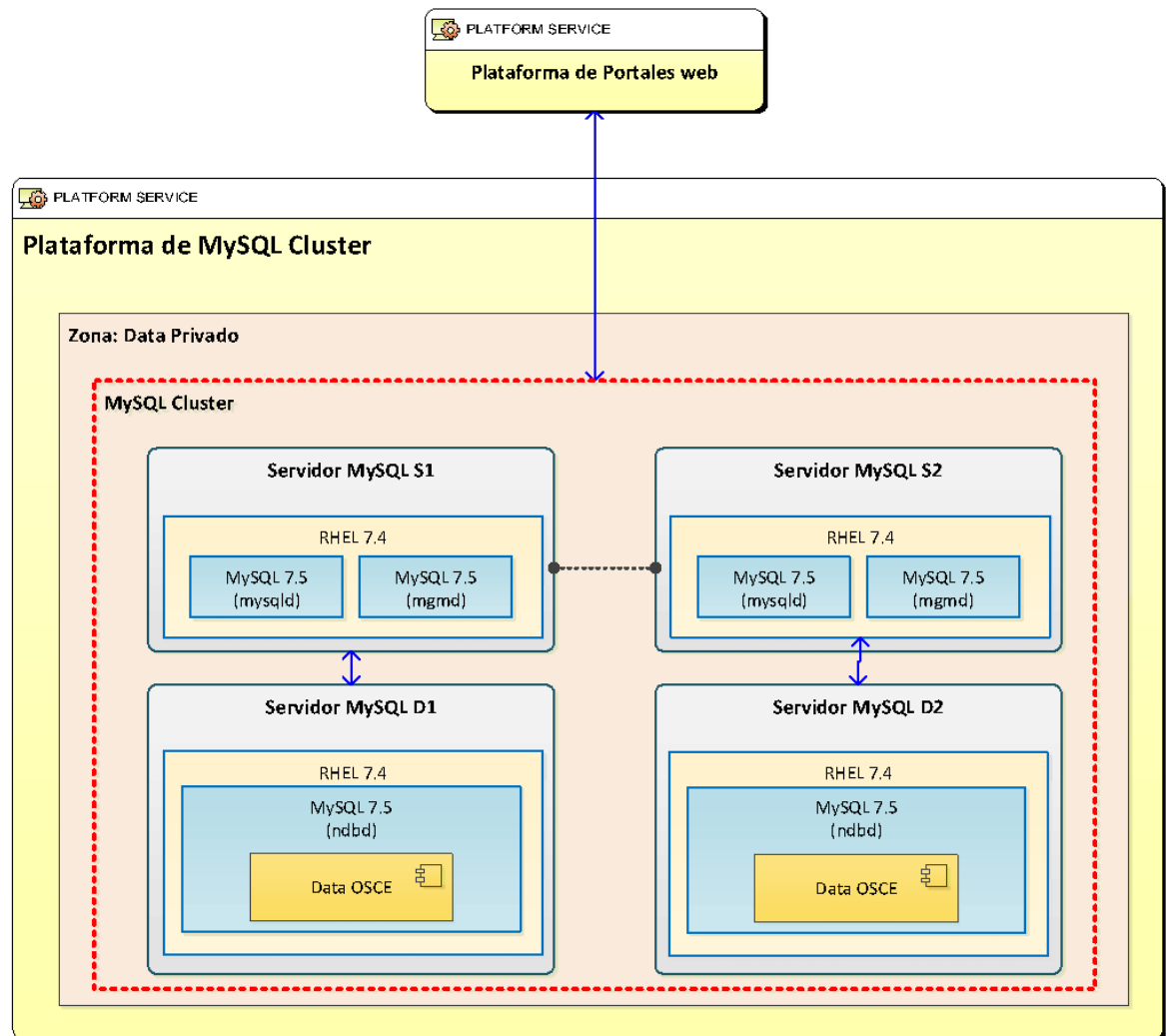
- ✓ Portales web de instituciones, tiendas virtuales (e-commerce), páginas de captura.

### ➤ URL's de referencia:

- ✓ <https://wordpress.org/>
- ✓ <https://easywpguide.com/wordpress-manual/>
- ✓ [https://w3techs.com/technologies/history\\_overview/content\\_management](https://w3techs.com/technologies/history_overview/content_management)

## Plataforma MySQL Cluster

- RHEL 7.4
- MySQL Cluster 7.5





## Plataforma MySQL Cluster – MySQL Cluster 7.5

MySQL Cluster es una base de datos transaccional, escalable en tiempo real y compatible con ACID, diseñada para ofrecer una disponibilidad del 99.999%. Con una arquitectura distribuida, multi-master, MySQL Cluster escala horizontalmente en hardware básico con auto-sharding (partición) para servir cargas de trabajo de lectura y escritura, a las que se accede a través de interfaces SQL y NoSQL. Licenciamiento GPL.

### ➤ Características:

- ✓ Rendimiento optimizado para acceso a claves primarias y escaneos de tablas. Beneficia el acceso a datos SQL y NoSQL
- ✓ Replicación geográfica activa-activa completa con detección y resolución automáticas de conflictos para lecturas, escrituras y eliminaciones
- ✓ Operaciones de mantenimiento y recuperación en línea con el nodo de datos
- ✓ Optimización en supervisión distribuida, gestión del uso de memoria y operaciones internas de la base de datos
- ✓ Permite leer tablas optimizadas como lectura desde la copia de seguridad y opciones de tabla completamente replicadas
- ✓ Compatibilidad con SQL, incluido el soporte para el tipo de datos JSON y columnas generadas

### ➤ Escenarios de uso:

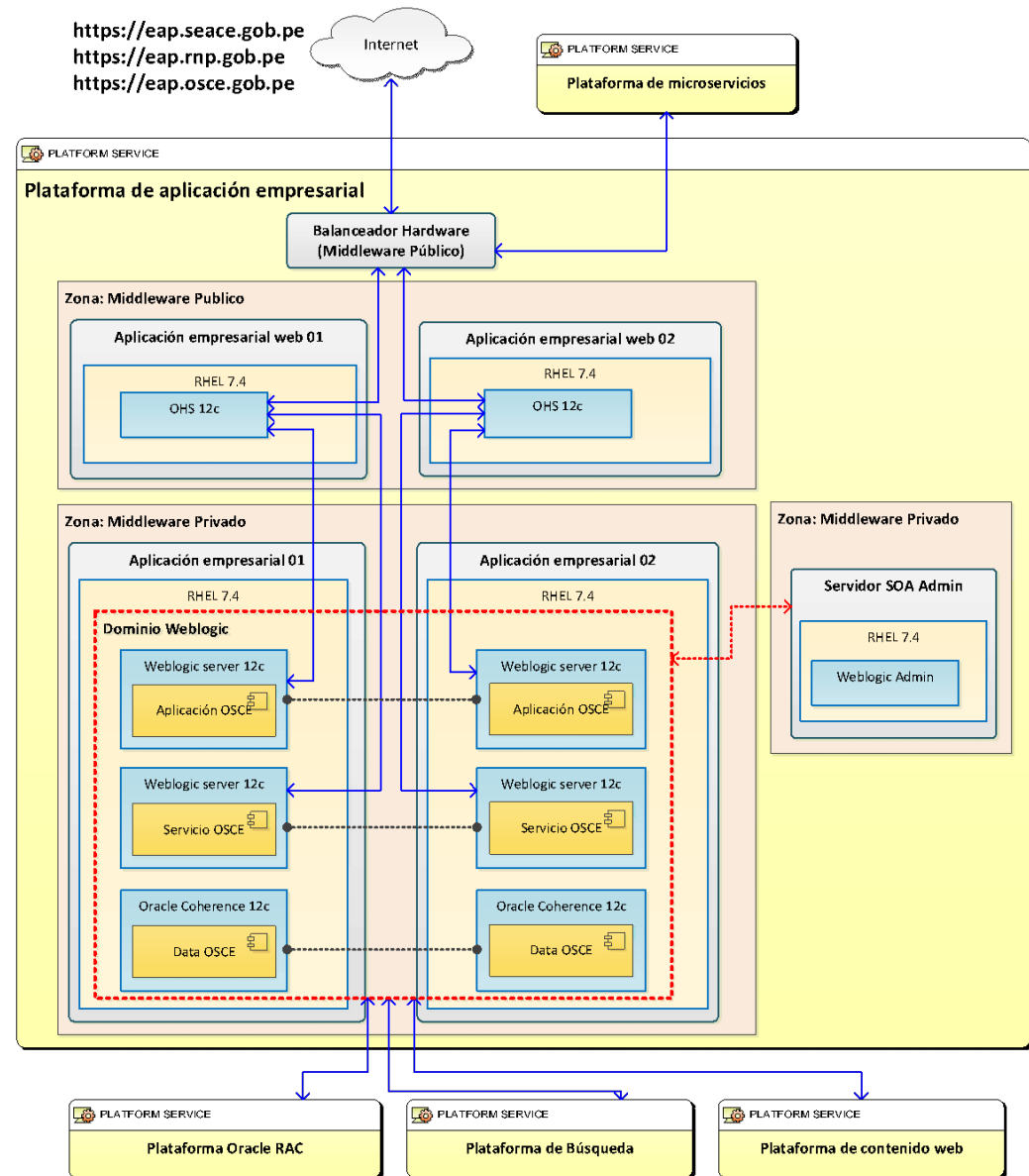
- ✓ Portales web basados en sistemas de gestión de contenido
- ✓ Aplicaciones web

### ➤ URL's de referencia:

- ✓ <https://dev.mysql.com/downloads/cluster/>
- ✓ [https://db-engines.com/en/ranking\\_trend/relational+dbms](https://db-engines.com/en/ranking_trend/relational+dbms)

## Plataforma de Aplicaciones empresariales

- RHEL 7.4
- OHS 12c
- Weblogic Server 12c
- Oracle Coherence 12c
- Java EE 7



## Plataforma de Aplicaciones empresariales – OHS 12c

Oracle HTTP Server (OHS) es un servidor web basado en Apache HTTP Server creado por Oracle Technology Network.

### ➤ Características:

- ✓ Permite inicio de sesión único (SSO) Encriptación con Secure Sockets Layer
- ✓ Compatibilidad con creación y control de versiones distribuidas (DAV)
- ✓ Reescritura de URL y capacidades del servidor proxy Páginas del servidor con soporte a lenguajes Perl, PHP, PL/SQL, C / C ++ (CGI y FastCGI)

### ➤ Escenarios de uso:

- ✓ Servidor web
- ✓ Servidor de balanceo de carga
- ✓ Servidor proxy

### ➤ URL's de referencia:

- ✓ <https://docs.oracle.com/middleware/1213/webtier/>

## Plataforma de Aplicaciones empresariales – Weblogic Server 12c

Oracle WebLogic es un servidor de aplicaciones Java EE(J2EE) y también un servidor web HTTP, desarrollado por BEA Systems y Oracle Corporation. Se ejecuta en Unix, Linux, Microsoft Windows, y otras plataformas.

### ➤ Características:

- ✓ Servidor de aplicaciones java EE
- ✓ Despliegue y gestión con entornos virtualizados.
- ✓ Gestión integrada del trafico http.
- ✓ Alta disponibilidad y recuperación de desastres.

### ➤ Escenarios de uso:

- ✓ Desplegar aplicaciones web java EE
- ✓ Desplegar servicios web
- ✓ Desplegar servicios rest

### ➤ URL's de referencia:

- ✓ <http://www.oracle.com/technetwork/middleware/weblogic/overview/>

## Plataforma de Aplicaciones empresariales – Oracle Coherence 12c

Oracle Coherence es una red de datos en memoria basada en Java, diseñada para tener una mejor confiabilidad, escalabilidad y rendimiento que los sistemas de administración de bases de datos relacionales tradicionales.

### ➤ Características:

- ✓ Mecanismo de persistencia permite la recuperación automática de una interrupción parcial o completa del clúster.
- ✓ Puede usar con almacenamiento local para velocidad o almacenamiento conectado a la red para la máxima disponibilidad.
- ✓ Se implementa sobre Weblogic Server
- ✓ Almacenamiento en caché federado, se puede elegir entre topologías multiusos activa-activa, activa-pasiva y hub-and-spoke, etc.
- ✓ Proporciona soporte distribuido único para Java 8 lambdas y streams.

### ➤ Escenarios de uso:

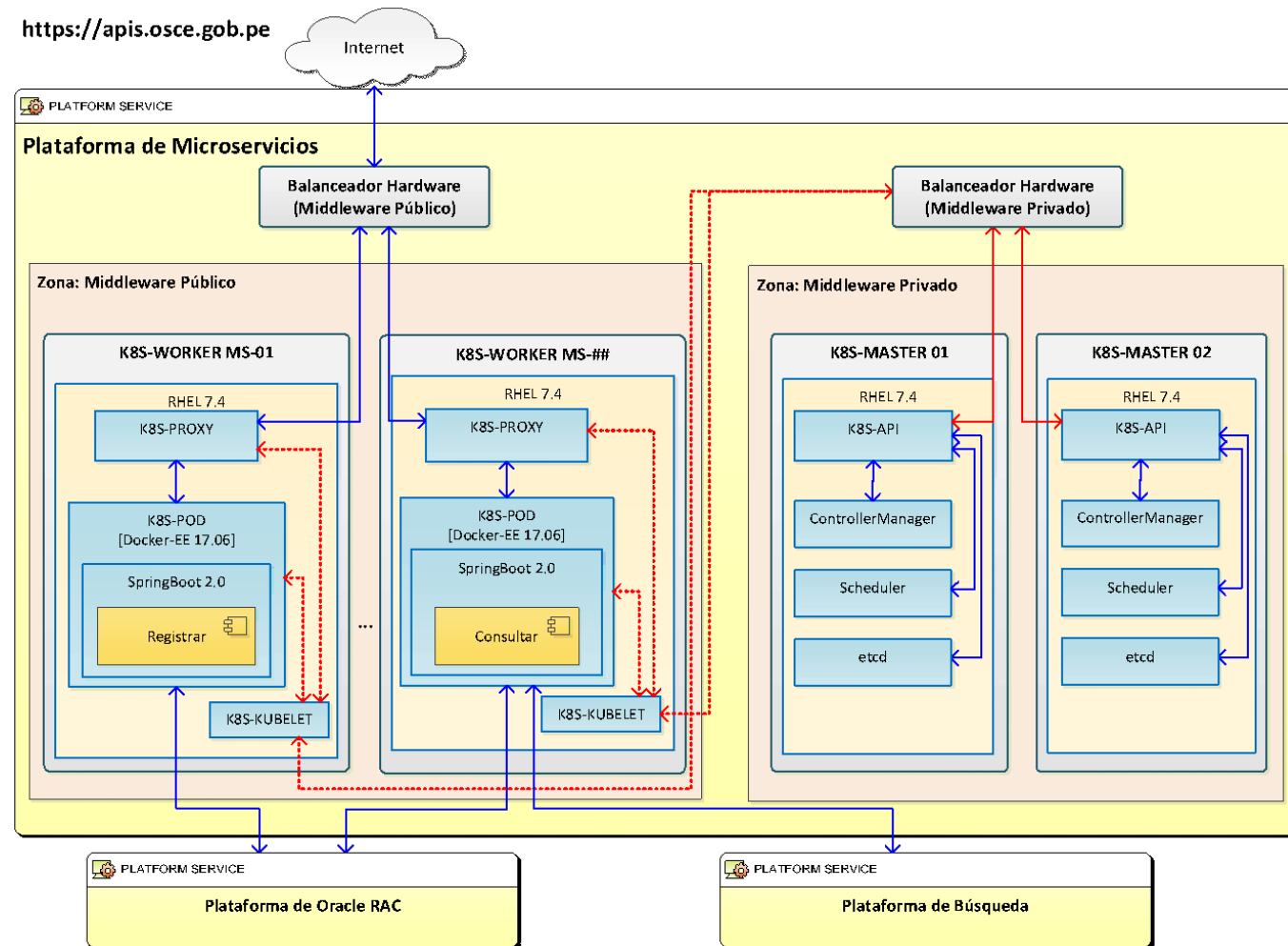
- ✓ Crear redes de datos en memoria para agilizar las lecturas/escrituras de objetos

### ➤ URL's de referencia:

- ✓ <https://docs.oracle.com/middleware/1221/coherence/>

## Plataforma de Microservicios

- RHEL 7.4
- Kubernetes 1.8
- Spring Boot 2.0
- Spring Security 5.0
- Spring Batch 4.0



## Plataforma de Microservicios – Kubernetes 1.8

Kubernetes (referido comúnmente como “k8s”) es un sistema de código libre para la automatización del despliegue, ajuste de escala y manejo de aplicaciones en contenedores. Soporta un rango de aplicaciones de contenedores, incluido Docker.

### ➤ Características:

- ✓ Portátil: en nubes públicas, privadas, híbridas
- ✓ Extensible: modular, enchufable, enganchable, composable
- ✓ Autorreparación: colocación automática, reinicio automático, replicación automática, escalado automático

### ➤ Escenarios de uso:

- ✓ Gestionar múltiples host remotos para el despliegue y mantenimiento de contenedores

### ➤ URL's de referencia:

- ✓ <https://www.thoughtworks.com/radar>
- ✓ <https://www.udacity.com/course/scalable-microservices-with-kubernetes--ud615>

## Plataforma de Microservicios – Docker EE 17.06

Docker automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de Virtualización a nivel de sistema operativo en Linux.

### ➤ Características:

- ✓ Permite crear contenedores con la portabilidad, ligereza y la autosuficiencia
- ✓ Permite gestionar repositorios, imágenes y contenedores

### ➤ Escenarios de uso:

- ✓ Podremos establecer una base desde la que comenzar nuestros proyectos, lo que nos ahorrará el tiempo de preparar el entorno para cada uno de ellos.
- ✓ Despliegue de aplicaciones libres de las dependencias instaladas en el sistema anfitrión.

### ➤ URL's de referencia:

- ✓ <https://apiumhub.com/tech-blog-barcelona/top-benefits-using-docker>
- ✓ <https://rubygarage.org/blog/advantages-of-using-docker-for-microservices>
- ✓ <https://medium.com/flow-ci/introduction-to-containers-concept-pros-and-cons-orchestration-docker-and-other-alternatives-9a2f1b61132c>



## Plataforma de Microservicios – Spring Boot 2.0

Spring Boot facilita la creación de aplicaciones basadas en java con calidad de producción, para "simplemente ejecutar" con el mínimo de configuración posible.

### ➤ Características:

- ✓ Permite crear aplicaciones Spring independientes
- ✓ Embebe Tomcat, Jetty o Undertow directamente (no es necesario implementar archivos WAR)
- ✓ Proporciona POM's de 'inicio' para simplificar su configuración de Maven
- ✓ Utiliza configuraciones Spring automáticas siempre que sea posible
- ✓ Proporciona información de los ambientes de producción, como métricas, controles de estado y configuración externalizada
- ✓ No requiere generar código y sin requisitos de configuración XML

### ➤ Escenarios de uso:

- ✓ Implementar aplicaciones web
- ✓ Implementar servicios web
- ✓ Implementar servicios rest
- ✓ Implementar aplicaciones batch

### ➤ URL's de referencia:

- ✓ <https://projects.spring.io/spring-boot/>
- ✓ <https://projects.spring.io/spring-boot/#quick-start>

## Plataforma de Microservicios – Spring Security 5.0

Spring Security es un marco que se enfoca en proporcionar autenticación y autorización para aplicaciones Java.

### ➤ Características:

- ✓ Soporte completo y extensible para Autenticación y Autorización
- ✓ Protección contra ataques como la fijación de sesiones, clickjacking, falsificación de solicitudes entre sitios, etc.
- ✓ Integración API Servlet

### ➤ Escenarios de uso:

- ✓ Implementar seguridad en aplicaciones web
- ✓ Implementar seguridad en servicios web
- ✓ Implementar seguridad en servicios rest

### ➤ URL's de referencia:

- ✓ <https://projects.spring.io/spring-security/>
- ✓ <https://projects.spring.io/spring-security/#quick-start>

## Plataforma de Microservicios – Spring Batch 4.0

Spring Batch proporciona funciones reutilizables que son esenciales en el procesamiento de grandes volúmenes de registros, incluidos el registro / seguimiento, la gestión de transacciones, las estadísticas de procesamiento de trabajos, el reinicio del trabajo, la omisión y la administración de recursos

### ➤ Características:

- ✓ Gestión de transacciones
- ✓ Procesamiento basado en fragmentos
- ✓ E / S declarativa
- ✓ Funcionalidad de Start / Stop / Restart, Reintentar / saltar
- ✓ Interfaz de administración basada en web (Spring Batch Admin)

### ➤ Escenarios de uso:

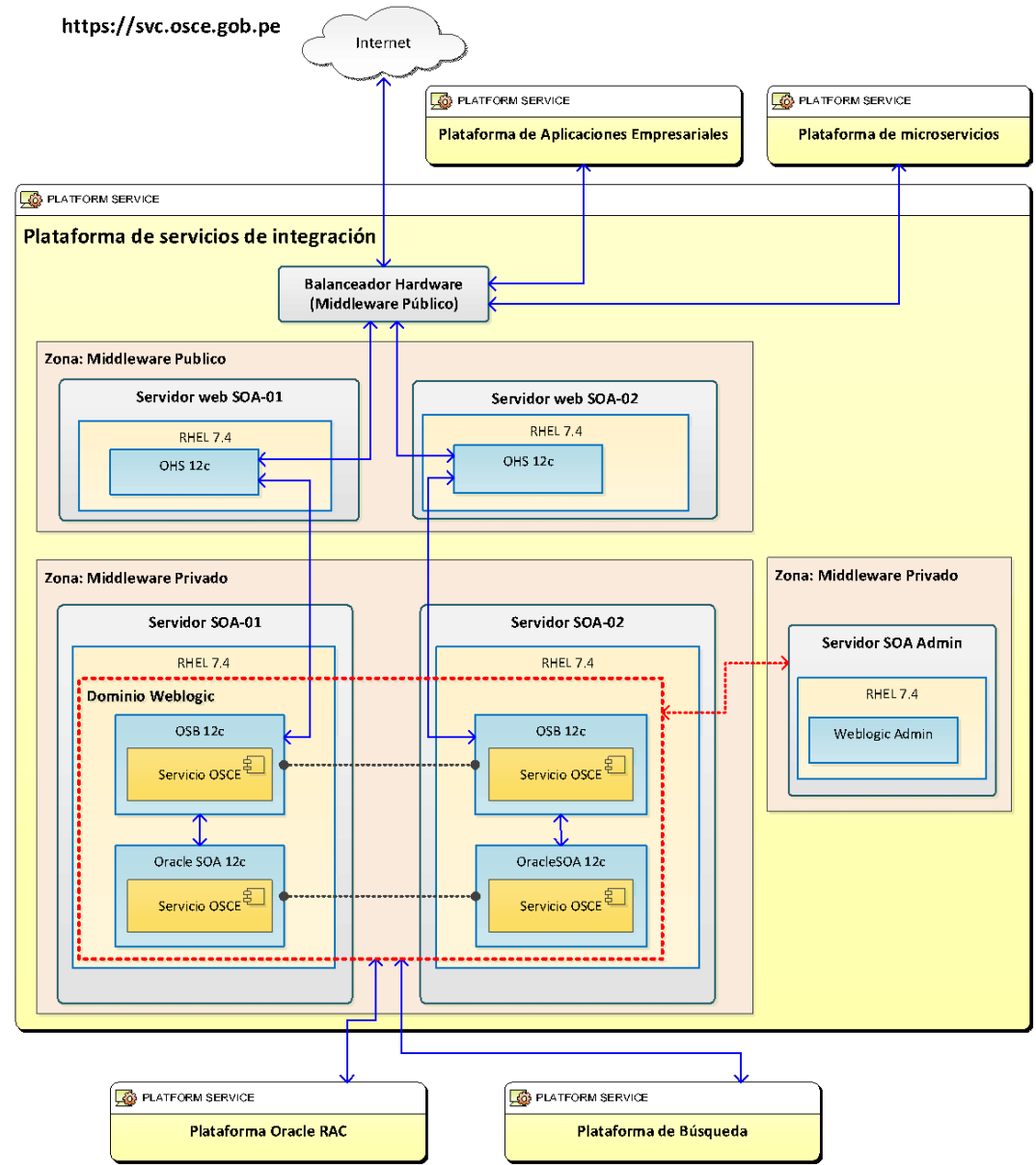
- ✓ Implementar procesamiento en lote con alto desempeño.

### ➤ URL's de referencia:

- ✓ <https://projects.spring.io/spring-batch/>
- ✓ <https://projects.spring.io/spring-batch/#quick-start>

## Plataforma de Servicios de integración

- RHEL 7.4
- OHS 12c
- OSB 12c
- Oracle SOA 12c



## Plataforma de Servicios de integración – OSB 12c

Es el Enterprise Service Bus de Oracle y es un componente de Oracle SOA 12c, permite transformar las arquitecturas conectando, mediando y administrando interacciones entre servicios y aplicaciones.

### ➤ Características:

- ✓ Proporciona componentes de conectividad, enrutamiento, transformación
- ✓ Proporciona virtualización de servicios
- ✓ Gestiona el versionamiento de servicios

### ➤ Escenarios de uso:

- ✓ Exponer servicios en un ESB

### ➤ URL's de referencia:

- ✓ <http://www.oracle.com/technetwork/middleware/service-bus/>
- ✓ <http://www.oracle.com/technetwork/middleware/soasuite/overview/wp-soa-suite-whats-new-12c-2217186.pdf>

## Plataforma de Servicios de integración – Oracle SOA 12c

Oracle SOA es parte de la familia de Oracle Fusion Middleware y permite integraciones dentro de una sola plataforma.

### ➤ Características:

Proporciona los siguientes componentes:

- ✓ Oracle BPEL Process Manager
- ✓ Oracle Service Bus (OSB)
- ✓ Oracle Event Processing
- ✓ Oracle Web Services Manager, a security and monitoring product for web services
- ✓ Oracle Business Rules
- ✓ Oracle Business Activity Monitoring
- ✓ Oracle Mediator
- ✓ Oracle JDeveloper

### ➤ Escenarios de uso:

- ✓ Implementar aplicaciones SCA para servicios web y servicios REST

### ➤ URL's de referencia:

- ✓ <http://www.oracle.com/technetwork/middleware/soasuite/>
- ✓ <http://www.oracle.com/technetwork/middleware/soasuite/overview/wp-soa-suite-whats-new-12c-2217186.pdf>

## Plataforma Servicio de Archivos

### ➤ RHEL 7.4

Esquema activo/pasivo

#### ✓ PaceMaker (\*)

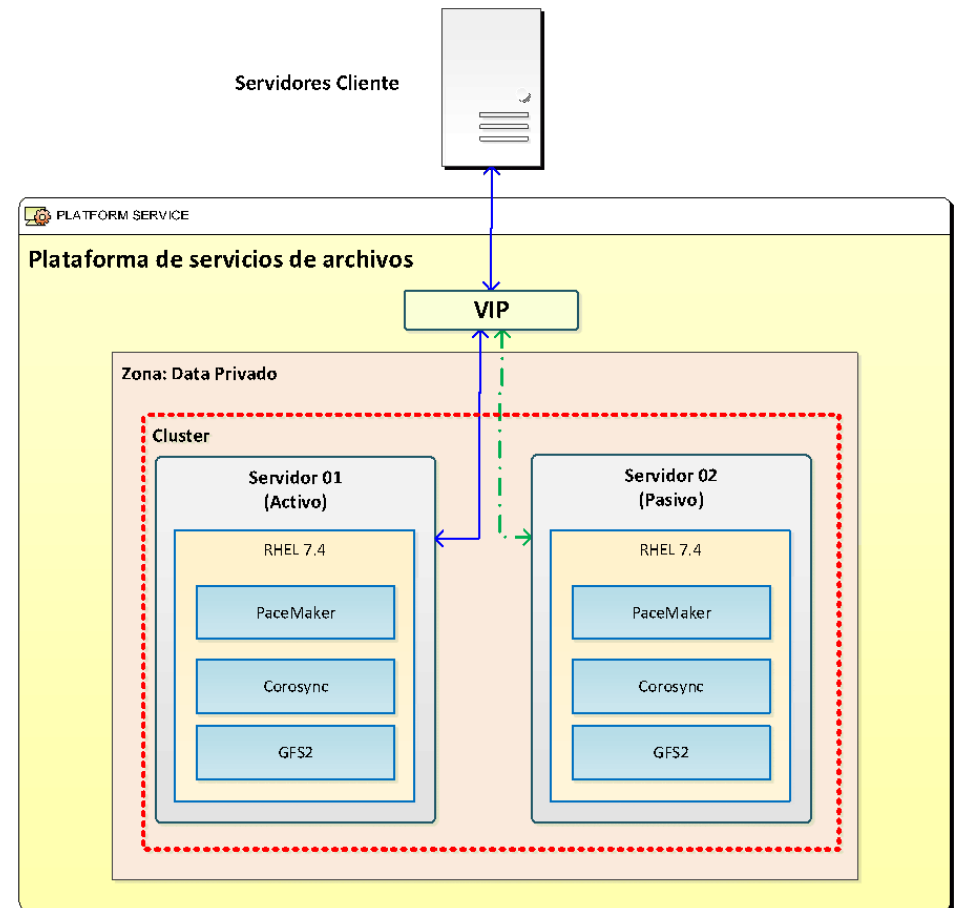
- Permite gestionar la disponibilidad de servidores linux, anteriormente usado como Heartbeat

#### ✓ Corosync (\*)

- Sistema de grupo de comunicación en clúster de servidores linux.

#### ✓ GFS2 (\*)

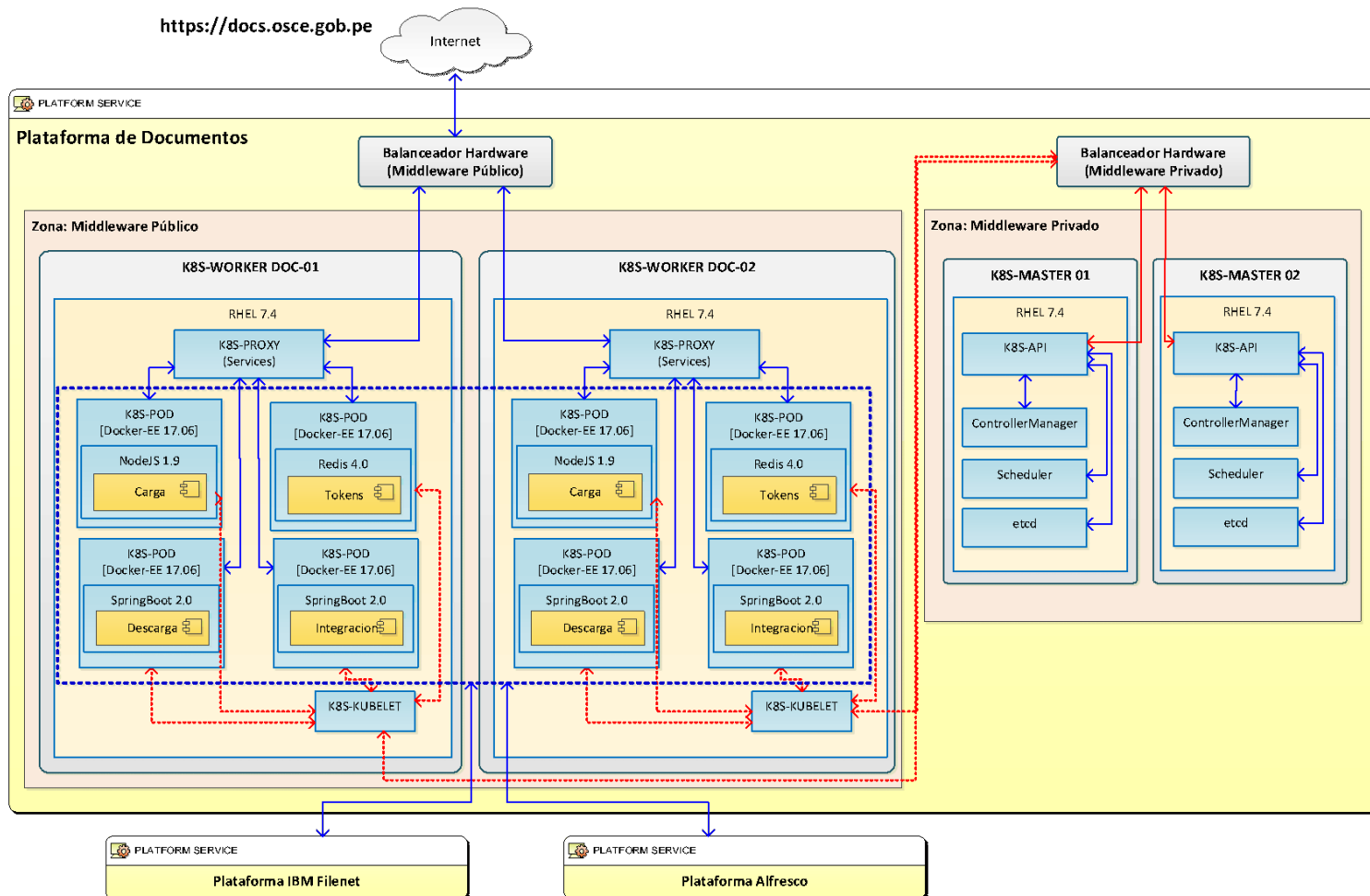
- El Global File System (GFS, también GFS2), es un sistema de archivos de discos compartidos o sistema de archivos distribuido para clústeres de servidores de sistemas GNU/Linux (principalmente).



## Plataforma de documentos

➤ NodeJS 1.9, SpringBoot 2.0, Redis 4.0

<https://docs.osce.gob.pe>





## Plataforma de documentos – Redis 4.0

**Redis** es un motor de base de datos en memoria, basado en el almacenamiento en tablas de hashes (clave/valor) que opcionalmente puede persistir los datos en disco.

### ➤ Características:

- ✓ Proporciona almacenamiento en memoria, lo cual proporciona rapidez al acceder a la información.
- ✓ Proporciona patron observador - PubSub, es decir que puedes crear emisión de datos desde un *publisher*, y tener *subscribes*, de modo que toda la información que el *publisher* emita a un canal, todos los *subscribes* se enteran en tiempo real.

### ➤ Escenarios de uso:

- ✓ Session Cache - para compartir sesiones en distintas plataformas.
- ✓ WebSocket – Podemos suscribir nuestras paginas web a eventos de nuestros sistemas a través de socket. Permite tener paginas en tiempo real evitando el polling.

### ➤ URL's de referencia:

- ✓ <https://redis.io/documentation>
- ✓ <https://redislabs.com/solutions/use-cases/redis-for-caching/>

## Plataforma de documentos – NodeJS 1.9

Node.js® es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente

### ➤ Características:

- ✓ Proporciona Un Solo hilo de OS, en lugar de generar un nuevo hilo de OS para cada conexión (y de asignarle la memoria acompañante), cada conexión dispara una ejecución de evento dentro del proceso del motor de Node. Esto permite tener multiples conexiones concurrentes a bajo coste de recursos.
- ✓ No realiza bloqueo de E/S, nunca se quedará en punto muerto porque no se bloquea directamente para llamados E/S. Lo cual es muy util en operaciones lectura y escritura de Archivos.

### ➤ Escenarios de uso:

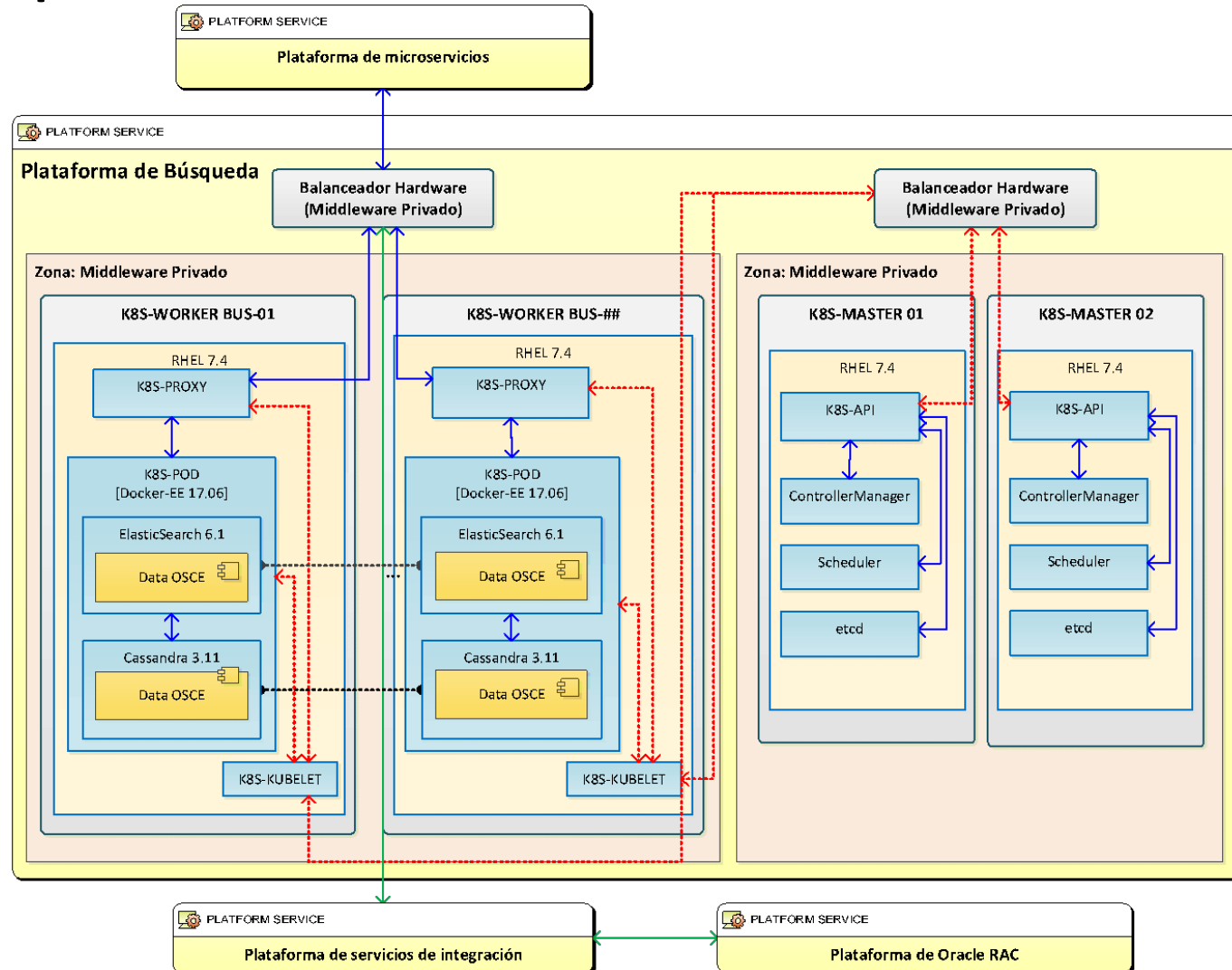
- ✓ Concurrencia –implementar aplicaciones altamente concurrentes.
- ✓ E/S – Implementar aplicaciones de procesamientos de archivos no bloqueantes.

### ➤ URL's de referencia:

- ✓ <https://nodejs.org/es/docs/>
- ✓ <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>

## Plataforma de búsqueda

- RHEL 7.4
- Kubernetes 1.8
- Docker-EE 17.06
- ElasticSearch 6.1
- Cassandra 3.11



## Plataforma de búsqueda – Elasticsearch 6.1

Elasticsearch es un servidor de búsqueda basado en Lucene. Provee un motor de búsqueda de texto completo, distribuido y con capacidad de multi-tenencia con una interfaz web REST y con documentos JSON. Elasticsearch está desarrollado en Java y está publicado como open source bajo las condiciones de la Licencia de Apache.

### ➤ Características:

- ✓ Actualización de datos en tiempo real.
- ✓ Permite realizar búsquedas full-text.
- ✓ Preparado para escalar horizontalmente.
- ✓ Capaz de detectar y eliminar nodos que estén fallando y reorganizarse a sí mismos para asegurar que los datos estén a salvo y permanezcan accesibles
- ✓ Puede alojar múltiples índices, que podrán ser consultados de manera independiente o en grupo.
- ✓ Proporciona un API Restful, con JSON y sobre HTTP, que permite realizar casi acciones administrativas y ACID de los datos.

### ➤ Escenarios de uso:

- ✓ Motor de búsqueda distribuido en tiempo real.

### ➤ URL's de referencia:

- ✓ <https://www.elastic.co/products/elasticsearch>
- ✓ <http://www.tnrglobal.com/blog/2012/03/elasticsearch-evaluation-white-paper-released-promising-for-big-data/>
- ✓ <https://www.oreilly.com/ideas/10-elasticsearch-metrics-to-watch>

## Plataforma de documentos – Cassandra 3.11

Apache Cassandra es una base de datos NoSQL distribuida y basada en un modelo de almacenamiento de «clave-valor», de código abierto que está escrita en Java. Permite almacenar grandes volúmenes de datos en forma distribuida. Su objetivo principal es la escalabilidad lineal y la disponibilidad. La arquitectura distribuida de Cassandra está basada en una serie de nodos iguales que se comunican con un protocolo P2P con lo que la redundancia es máxima. Está desarrollada por Apache Software Foundation.

### ➤ Características:

- ✓ La información está repartida a lo largo de los nodos del cluster.
- ✓ Rendimiento de forma lineal respecto al número de nodos que añadamos.
- ✓ Escala de forma horizontal,
- ✓ Cassandra Query Language (CQL) es el lenguaje de acceso a datos en Cassandra, es un derivado reducido de SQL

### ➤ Escenarios de uso:

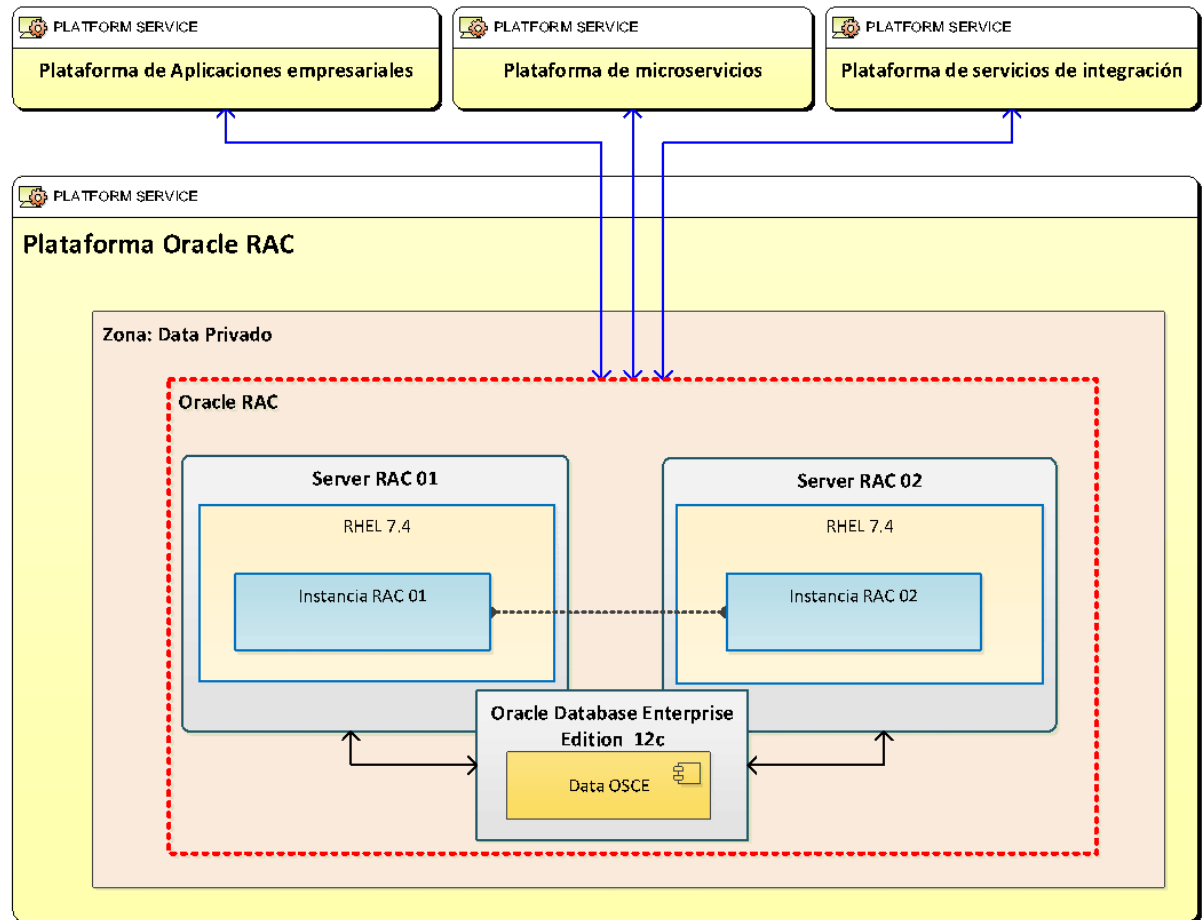
- ✓ Almacenamiento de datos no estructurados
- ✓ Búsqueda en datos no estructurados

### ➤ URL's de referencia:

- ✓ <http://cassandra.apache.org/>
- ✓ <https://www.panel.es/blog/por-que-cassandra-no-es-una-base-de-datos-cualquiera/>
- ✓ <https://www.paradigmigital.com/dev/cassandra-la-dama-de-las-bases-de-datos-nosql/>
- ✓ <https://www.datastax.com/products/datastax-enterprise-support>

## Plataforma Oracle RAC

- RHEL 7.4
- Oracle RAC 12c



## Plataforma Oracle RAC – Oracle RAC 12c

Es producto Oracle que permite que múltiples computadoras accedan a un Oracle RDBMS de manera simultánea mientras acceden a una base de datos, esto proporciona clustering. Es un componente adicional para la Oracle Database Enterprise Edition

### ➤ Características:

- ✓ Permite que una o más instancias a acceden a una misma base de datos Oracle

### ➤ Escenarios de uso:

- ✓ Implementar clúster de base de datos Oracle

### ➤ URL's de referencia:

- ✓ <http://www.oracle.com/technetwork/database/options/clustering>
- ✓ <https://docs.oracle.com/database/121/DBLIC/editions.htm>

## Plataforma Oracle RAC – Oracle Database Enterprise Edition 12c

Oracle Database es un sistema de gestión de base de datos de tipo objeto-relacional (ORDBMS) propiedad de Oracle Corporation.

### ➤ Características:

- ✓ Bases de datos 'conectables'
- ✓ Oracle database 12c monitoriza la actividad de cada columna de las tablas, determina qué tipo de compresión se adapta mejor a cada una
- ✓ Permite crear índices duplicados sobre el mismo conjunto de columnas, útil en entornos de Data Warehouse
- ✓ Devolver resultsets por referencia el mismo cursor desde las mismas funciones y procedures
- ✓ Incorpora columnas de Identidad (auto incrementador)
- ✓ Sentencias que permiten filas, offset y fetch, permiten crear consultas de tipo Top N además permitir la paginación de los datos para seleccionar ciertos subconjuntos de datos.
- ✓ Tamaños máximos de Varchar2 y NVarchar2 se han incrementado de 4K a 32K

### ➤ Escenarios de uso:

- ✓ Implementar base de datos relacionar para sistemas de negocio

### ➤ URL's de referencia:

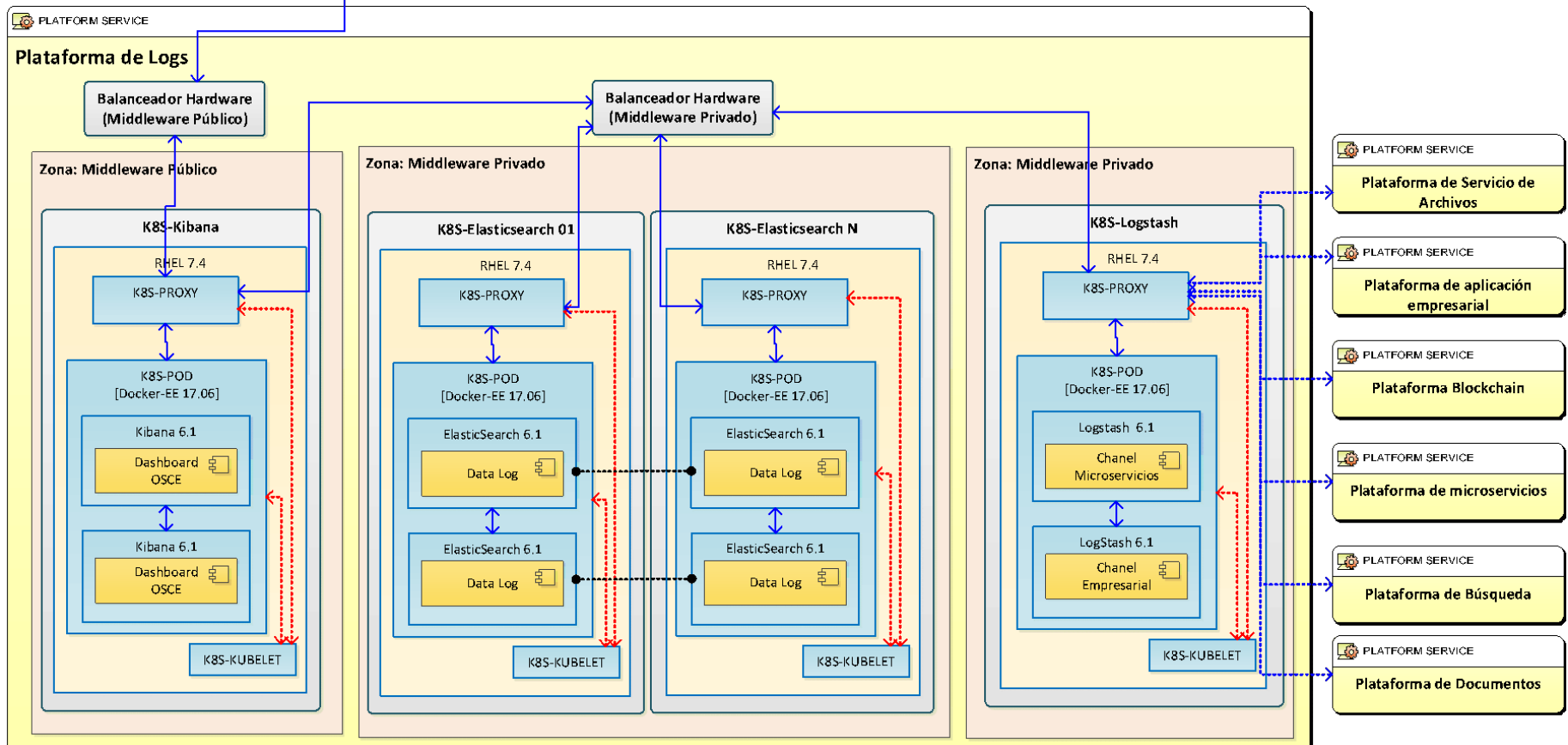
- ✓ <http://www.oracle.com/technetwork/database/enterprise-edition>
- ✓ <https://docs.oracle.com/database/121/DBLIC/editions.htm>



## Plataforma de logs

➤ RHEL 7.4, Docker EE 17.06, Kibana, Elasticsearch 6.1, logstash 6.1

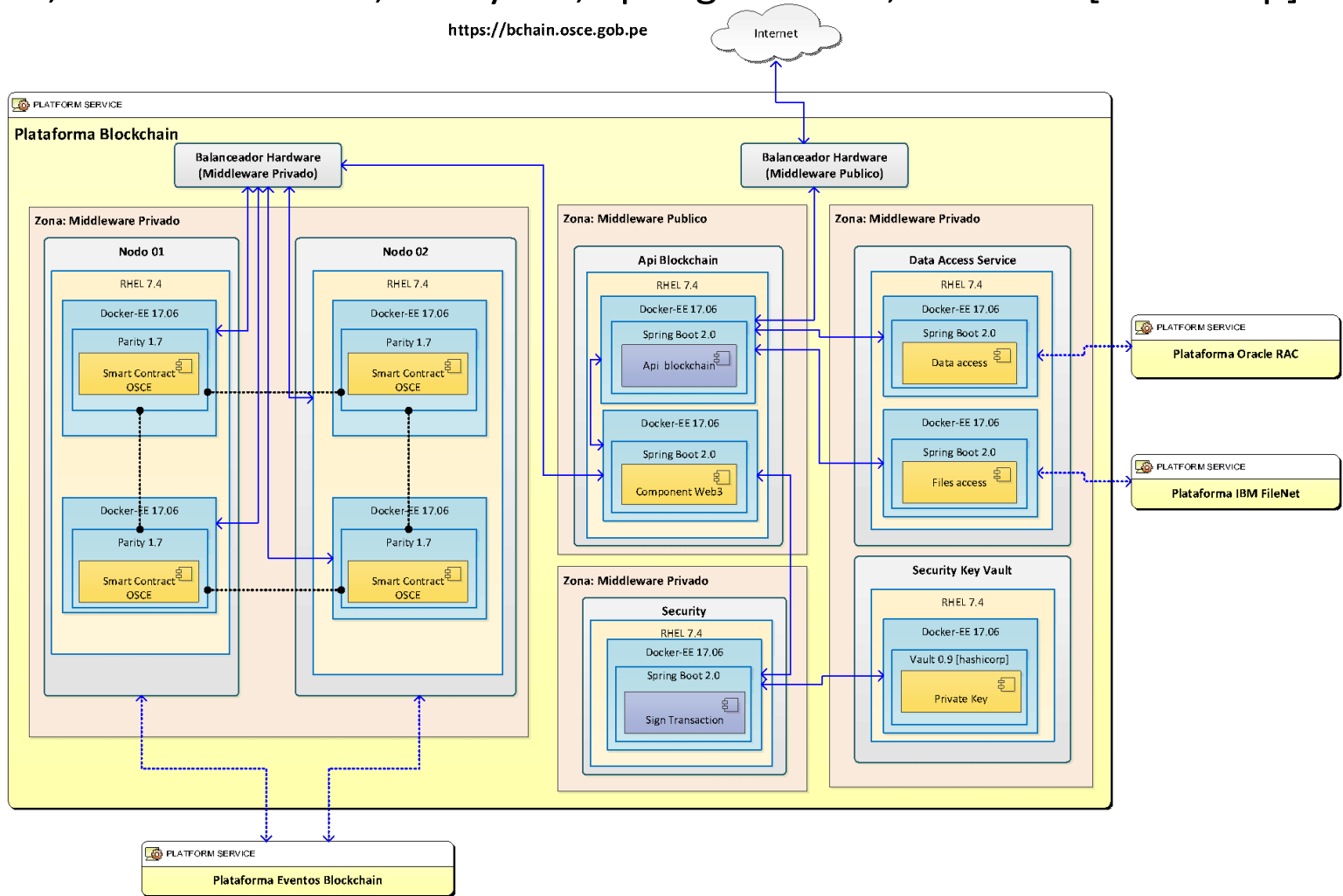
<https://monitor.seace.gob.pe>



## Plataforma Blockchain

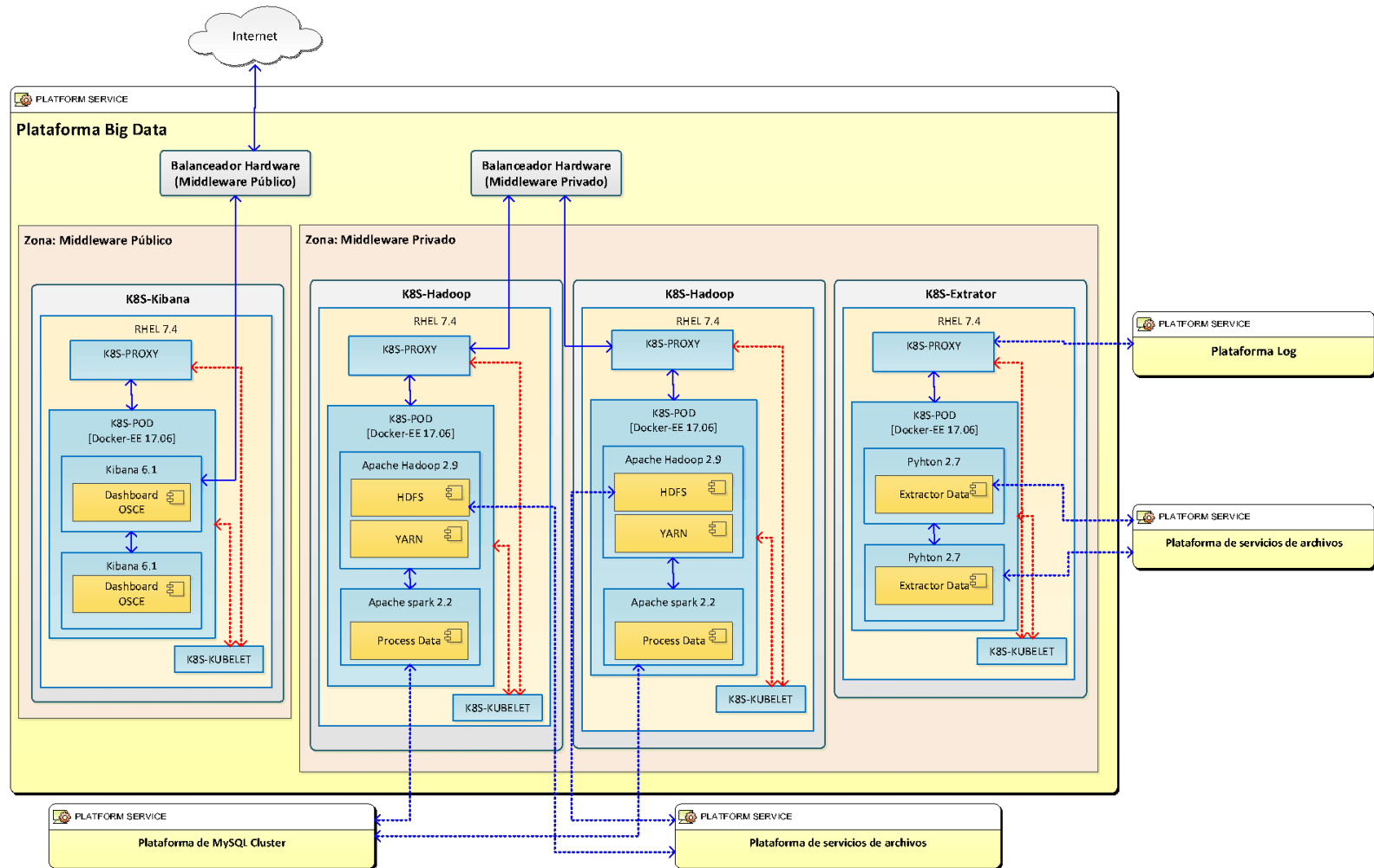
➤ RHEL 7.4, Docker EE 17.06, Parity 1.7, Spring Boot 2.0, Vault 0.9 [hashicorp]

<https://bchain.osce.gob.pe>



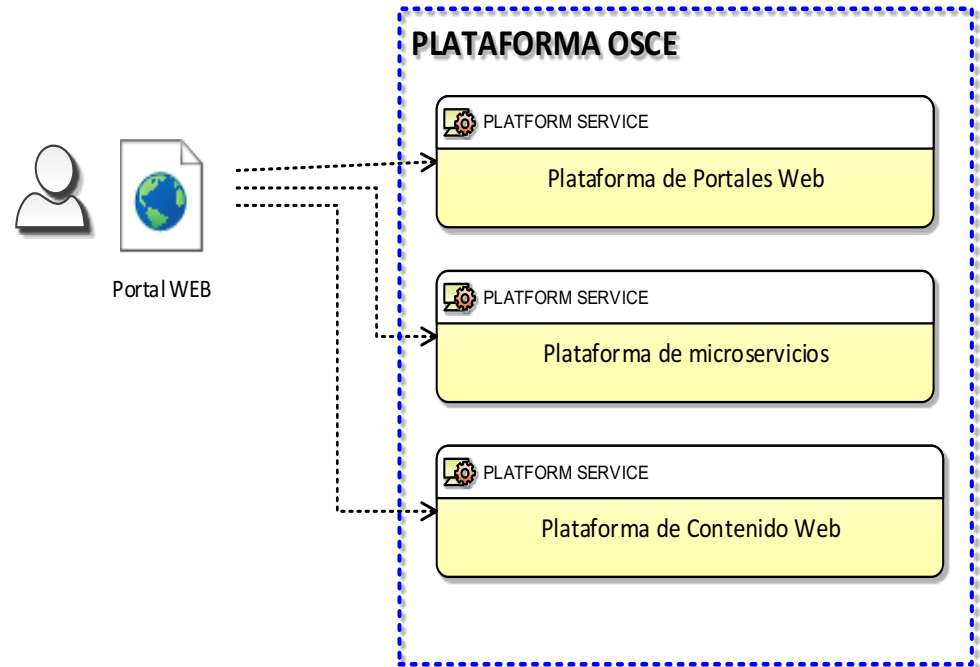
## Plataforma Bigdata

- RHEL 7.4, Kubernetes 1.8, Apache Hadoop 2.9, Apache spark 2.2, Kibana 6.1  
<https://monitor.seace.gob.pe>



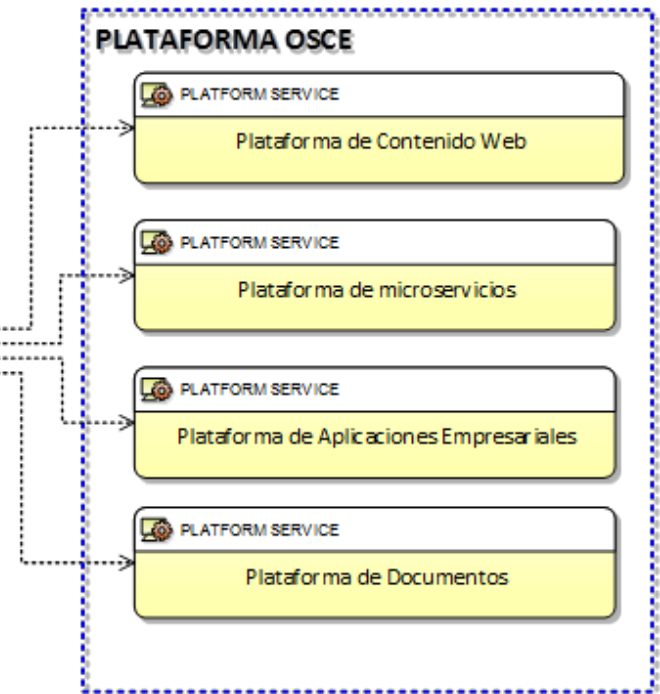
## Arquitectura software para portales web

- La información de portales web permiten proporcionar información de alcance general a los usuarios del OSCE.
- Proporciona soporte tecnológico a los portales web de las diferentes áreas organizacionales del OSCE.
- Permite la inclusión de páginas web con información dinámica consumida desde la plataforma de microservicios.
- Permite la inclusión de contenido web de alcance general que este expuesto en la plataforma de contenido web del OSCE.



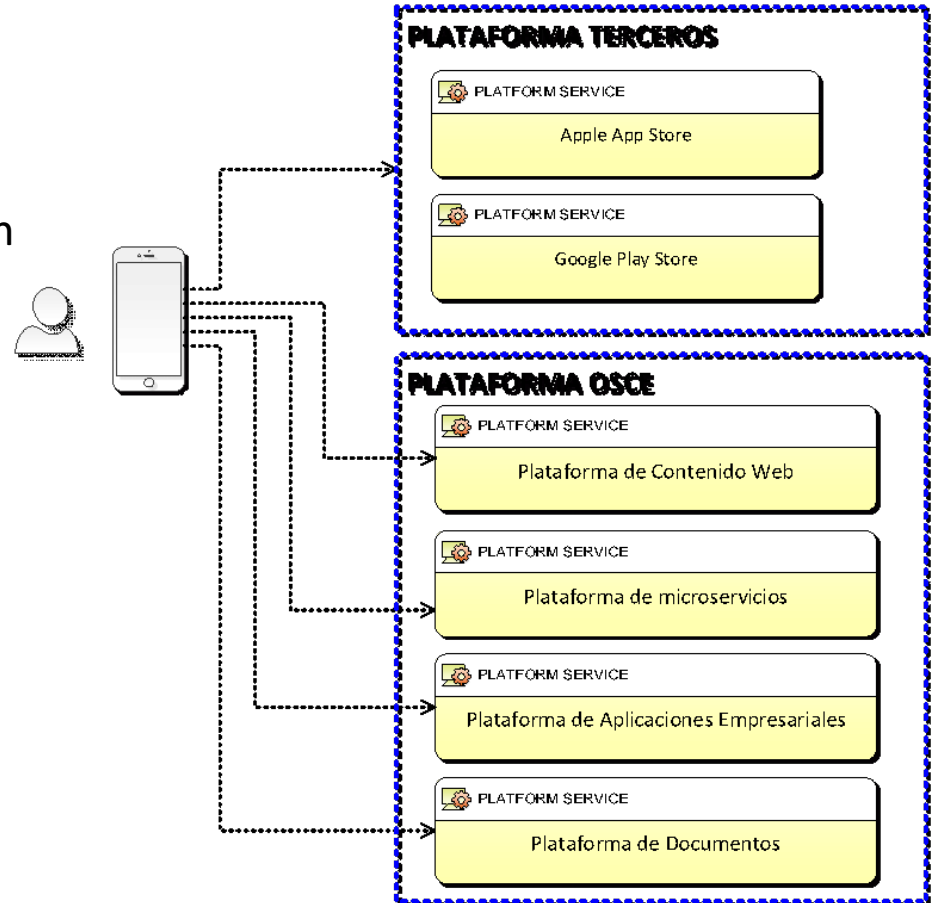
## Arquitectura software para aplicaciones web

- Las aplicaciones web permiten procesar información transaccional y de consultas de los procesos de negocio del OSCE.
- Las aplicaciones web pueden reutilizar el contenido web general al OSCE o el contenido web de otras aplicaciones web que estén publicadas en la plataforma de contenido web.
- Las aplicaciones web procesan la información a través de los servicios expuestos en la plataforma de microservicios y plataforma de aplicaciones empresariales.
- Las aplicaciones web acceden a los documentos del OSCE a través de la plataforma de documentos.
- Las aplicaciones web del tipo SPA se publican en la plataforma de contenido web
- Las aplicaciones web que forman parte de un sistema monolítico son desplegadas en la plataforma de aplicaciones empresariales y el contenido web que utiliza es desplegado en la plataforma de contenido web.



## Arquitectura software para Aplicaciones Móviles

- Las aplicaciones móviles permiten realizar operaciones transaccionales y de consultas de los procesos de negocio del OSCE.
- Las aplicaciones móviles son publicadas en Apple App Store y Google Play Store para que los usuarios la descarguen
- Las aplicaciones móviles hacen uso de la plataforma de contenido web para reutilizar contenido de otras aplicaciones web del OSCE.
- Las aplicaciones móviles procesan la información a través de los servicios expuestos en la plataforma de microservicios y plataforma de aplicaciones empresariales.
- Las aplicaciones móviles acceden a los documentos del OSCE a través de la plataforma de documentos.

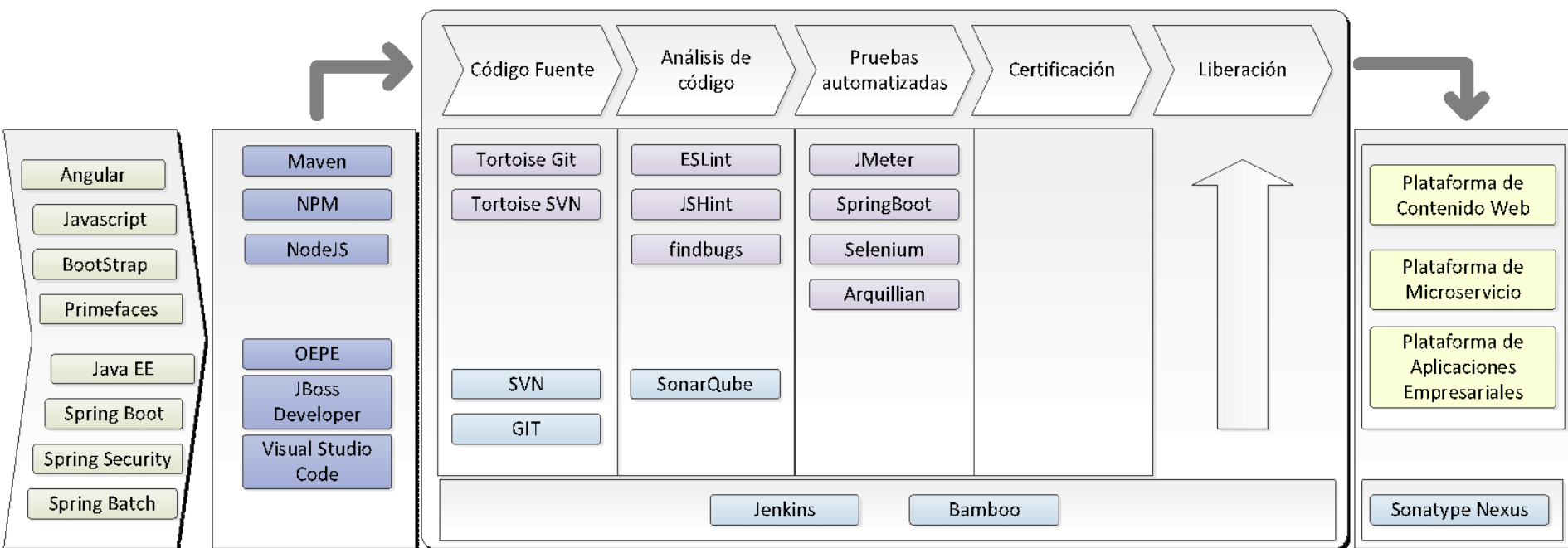


## Arquitectura software para Servicios de Integración

- Los servicios de integración permite orquestar el procesamiento entre diferentes sistemas del OSCE y/o sistemas externos al OSCE.
- Las aplicaciones del OSCE deben acceder a los servicios de entidades externas a través de la plataforma de integración de servicios.
- Las aplicaciones del OSCE deben acceder a los servicios internos a través de la plataforma de integración de servicios.
- Permite gestionar los eventos que comuniquen los sistemas del OSCE para invocar los servicios de negocio relacionados.
- Permite gestionar las tareas programadas requeridas por los diferentes sistemas del OSCE.

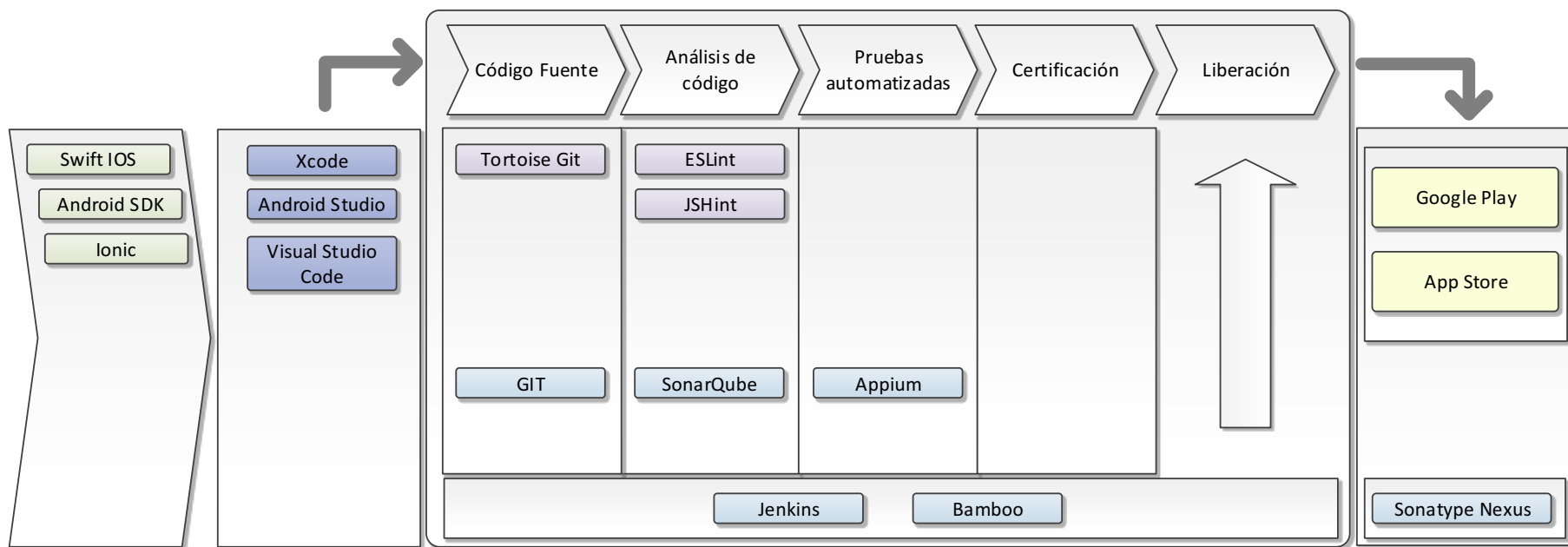


## Arquitectura software para Entorno de desarrollo de aplicaciones web

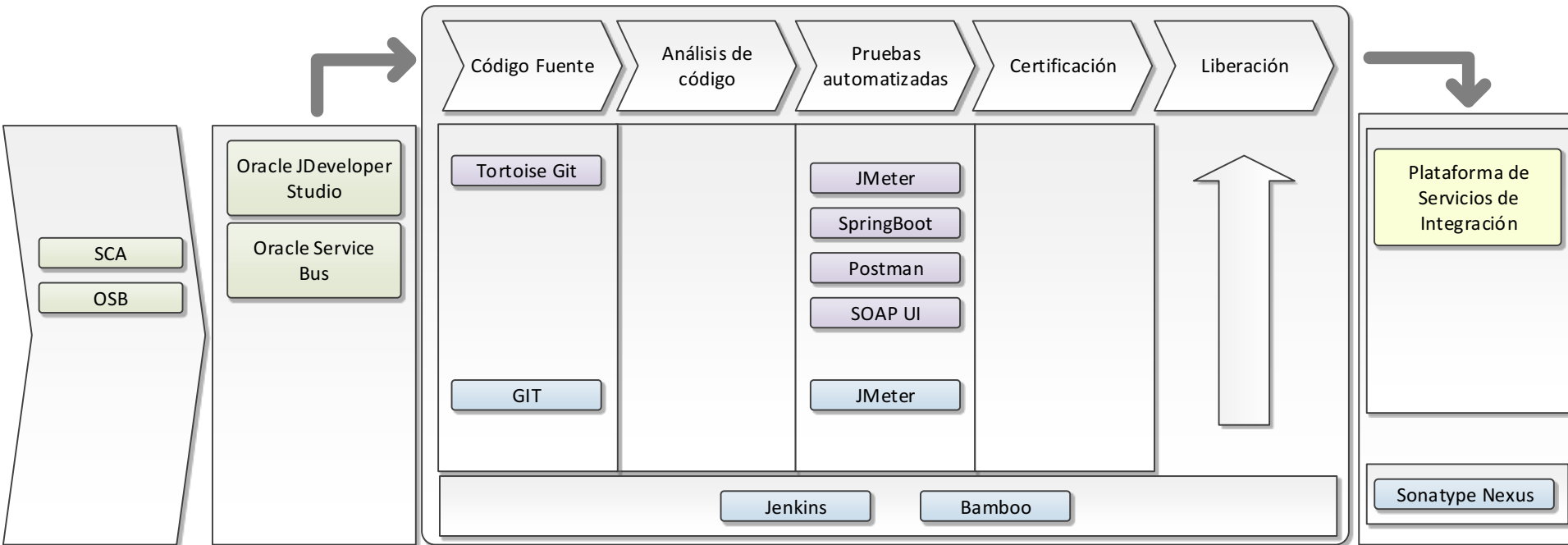




## Arquitectura software para de aplicaciones móviles



## Arquitectura software para de servicios de integración



## **CRITERIOS PARA DETERMINAR LA VIABILIDAD DE APLICACIÓN DE LA ARQUITECTURA DE SOFTWARE PROPUESTA**

### **CRITERIOS PARA APLICACIONES WEB - SPA**

Para el desarrollo de aplicaciones web del tipo SPA cuando se cumplan algunos de los siguientes criterios:

- Se requiera utilizar un diseño web responsivo (Responsive Web Design)
- Se pueda usar en navegadores web con soporte a HTML 5 y CSS3.
- Se tenga previsto que los sistemas del OSCE producirán servicios REST para el consumo de clientes.

### **CRITERIOS PARA APLICACIONES EMPRESARIALES**

La arquitectura de software para aplicaciones empresariales debe aplicarse cuando se cumplan alguno de los siguientes criterios:

- Se requieran implementar servicios que mantiene fuerte dependencia con otros componentes de la misma aplicación empresarial.
- Se implemente servicios web SOAP basados en tecnología java.
- Se extiendan funcionalidades de aplicaciones web existentes basadas en tecnología JSF.

## **CRITERIOS PARA DETERMINAR LA VIABILIDAD DE APLICACIÓN DE LA ARQUITECTURA DE SOFTWARE PROPUESTA**

### **CRITERIOS PARA SERVICIOS DE INTEGRACIÓN**

La arquitectura de software para servicios de integración debe aplicarse a servicios que cumplan con alguno de los siguientes criterios:

- Los servicios a implementar son producidos por sistemas del OSCE y serán consumidos por otros sistemas del OSCE.
- Los servicios a implementar son producidos por sistemas del OSCE y serán consumidos por sistemas externos al OSCE
- Los servicios a implementar son producidos por sistemas externos al OSCE y serán consumidos por sistemas del OSCE.
- Los servicios a implementar gestionan los eventos que comunican los diferentes sistemas del OSCE.
- Los servicios a implementar orquestan la sincronización de información de diferentes sistemas del OSCE.
- Los servicios a través del cual se planifica la ejecución de tareas programadas de los diferentes sistemas del OSCE.

## **CRITERIOS PARA DETERMINAR LA VIABILIDAD DE APLICACIÓN DE LA ARQUITECTURA DE SOFTWARE PROPUESTA**

### **CRITERIOS PARA MICROSERVICIOS**

La arquitectura de software para microservicios debe aplicarse cuando se cumplan algunos de los siguientes criterios:

- Se realizan procesos de migración de sistemas que requieran de actualización tecnológica.
- Las microservicios a implementar son independientes para despliegue de otras softwares de aplicación existentes en el OSCE.
- Se tiene previsto que los microservicios a implementar tendrán un crecimiento y decrecimiento horizontal por la carga operacional a la que serán expuestas.
- Los microservicios a implementar proporcionarán funcionalidad back-end para operaciones transaccionales y de búsqueda de los sistemas del OSCE.

## **CRITERIOS PARA DETERMINAR LA VIABILIDAD DE APLICACIÓN DE LA ARQUITECTURA DE SOFTWARE PROPUESTA**

### **CRITERIOS PARA APLICACIONES MÓVILES - NATIVAS**

Para el desarrollo de aplicaciones web con la propuesta de arquitectura de software se debe evaluar si como mínimo se cumple con los siguientes criterios:

- Aceptación que los costos de desarrollo son mayores.
- Se requiere aprovechar los recursos del dispositivo para obtener un alto rendimiento de la aplicación en el dispositivo móvil del usuario
- Requerimientos de sistemas que involucren el aprovechamiento óptimo del hardware del dispositivo.

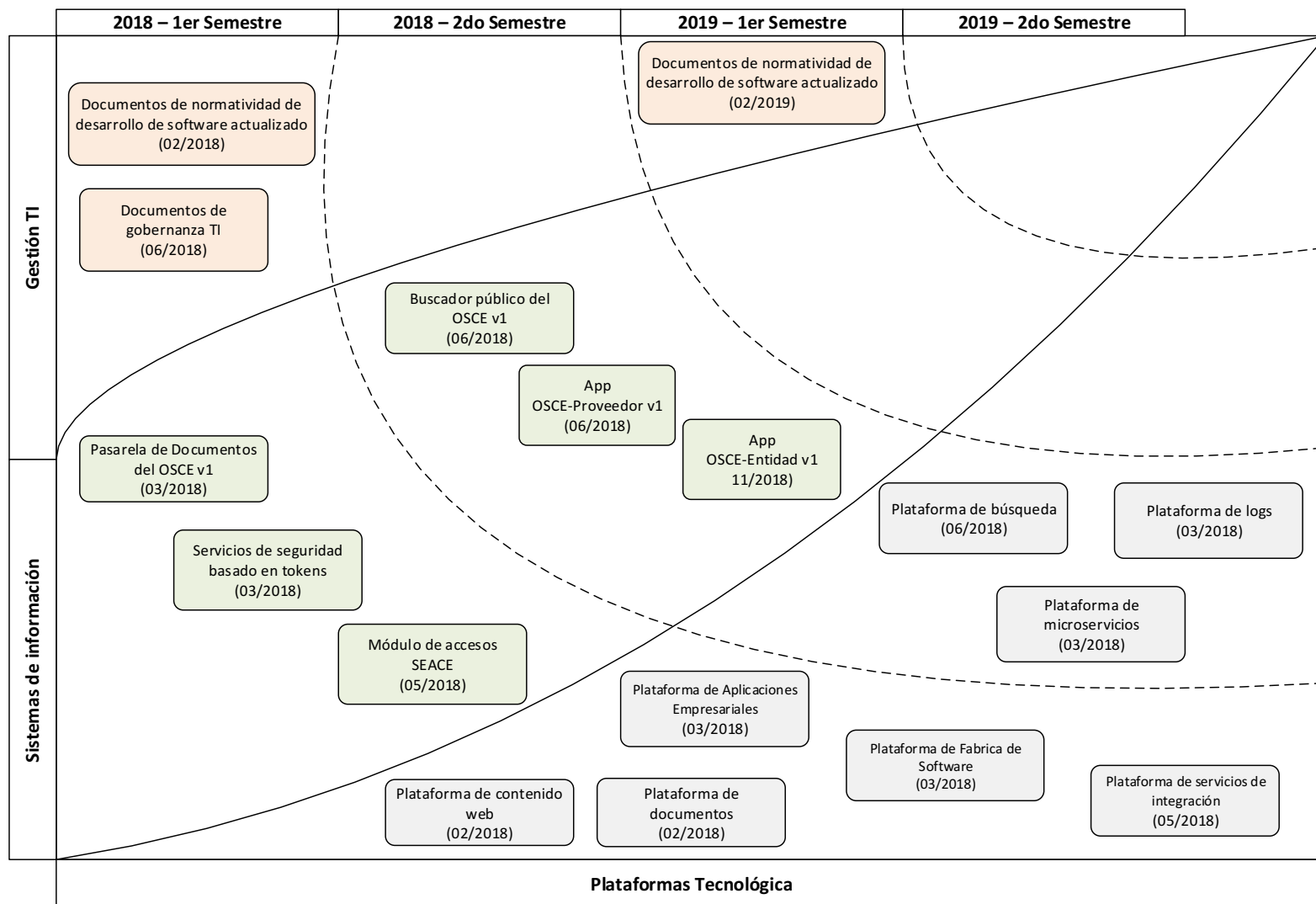
### **CRITERIOS PARA APLICACIONES MÓVILES - HÍBRIDAS**

Para el desarrollo de aplicaciones web con la propuesta de arquitectura de software se debe evaluar si como mínimo se cumple con los siguientes criterios:

- Aprovechar que una sola implementación sea desplegable en más de una plataforma.
- Se requiere rendimiento de gama media
- Funcionalidades principales basadas en intercambio de información con servicios back-end
- Se requiere que la estandarizar las capacidades de la aplicación en las diferentes plataformas destino
- Se cuenta con personal multidisciplinario para dar soporte a las aplicaciones

# Propuesta de la visión de arquitecturas de software para el OSCE

## Estrategias para la transición de arquitecturas de software en el OSCE



## Estrategias para la transición de arquitecturas de software en el OSCE

### ESTRATEGIA PARA INCORPORACIÓN DE MICROSERVICIOS

- Los nuevos sistemas deben ser diseñados usando el estilo de arquitectura de microservicios.
- Se debe identificar las funcionalidades que soportan alta carga operacional para realizar la migración al estilo de arquitectura de microservicios, teniendo en cuenta que:
  - ✓ Los microservicios back-end que realizan operaciones transaccionales deben ser desarrollados con tecnologías Spring.
  - ✓ Los microservicios back-end que realizan operaciones de búsquedas deben ser desarrollados con tecnologías NodeJS y hacer uso de la plataforma de búsqueda.
  - ✓ Implementar aplicaciones web del tipo single-page application (SPA) que utilicen los microservicios back-end.
- Se debe implementar servicios que gestionen la autenticación y autorización de usuarios basada en tokens para que sean utilizados por los microservicios que incorporen requerimientos de seguridad.
- Habilitar una plataforma tecnológica que permita incorporar la cultura devops al proceso de desarrollo de software, se debe tener en cuenta que con el aumento de microservicios será mayor la necesidad de automatizar el proceso de entrega y despliegue de software.
- Habilitar una plataforma tecnológica que permita gestionar el ciclo de vida de los microservicios en sus diferentes tecnologías Spring, NodeJS, en el presente documento se ha propuesto que la gestión de microservicios se realice a través de contenedores orquestados por Kubernetes.



## Estrategias para la transición de arquitecturas de software en el OSCE

### ESTRATEGIA PARA ESTANDARIZAR APLICACIONES EMPRESARIALES

- Identificar las aplicaciones empresariales existentes basadas en java EE que requieran de actualización tecnológica para sean modificadas y se les incorpore la tecnología java EE 7.
- Identificar las aplicaciones empresariales existentes basadas en java EE que se ejecuten en servidores de aplicaciones diferentes a Weblogic Server para que sean modificadas para desplegar en la plataforma Weblogic Server.
- Optimizar las funcionalidades de las aplicaciones empresariales que tiene fuerte dependencia entre componentes de la misma aplicación y exponerlos a través de servicios web SOAP para que sean reutilizados por otros sistemas del OSCE.

## Estrategias para la transición de arquitecturas de software en el OSCE

### ESTRATEGIA PARA ESTANDARIZAR SERVICIOS DE INTEGRACIÓN

- Identificar los servicios que son utilizados por más de un sistema del OSCE y que no estén siendo consumidos desde el bus de servicios empresarial para planificar su implementación y exposición en la plataforma de servicios de integración propuesta en el presente documento.
- Para las tareas programadas utilizados por los sistemas del OSCE se debe proceder de la siguiente manera:
  - ✓ Identificar las tareas programadas de base de datos, las tareas programadas de aplicaciones java que serán migradas a la plataforma de servicios de integración
  - ✓ Implementar los servicios de orquestación que invocan las tareas programadas y registrar como nuevas tarea programada en la plataforma de servicios de integración.
  - ✓ Realizar la documentación de las tareas programadas migradas y publicarlo en el repositorio de arquitectura.

## Estrategias para la transición de arquitecturas de software en el OSCE

### ESTRATEGIA PARA ESTANDARIZAR SERVICIOS DE INTEGRACIÓN

- Para los funcionalidades de búsquedas sobre información gestionada por diferentes sistemas del OSCE se debe proceder de la siguiente manera:
  - ✓ Identificar las búsquedas con información cruzada de diferentes sistemas del OSCE.
  - ✓ Implementar los servicios de sincronización de información para que consoliden, transformen y transfieran la información desde la fuente de información como base de datos relacionales hacia la plataforma de búsqueda propuesta en el presente documento.
  - ✓ Realizar la documentación de los servicios de sincronización de información y publicarlo en el repositorio de arquitectura.
- Para las funcionalidades que comunican eventos entre diferentes sistemas del OSCE se debe proceder de la siguiente manera:
  - ✓ Identificar las funcionalidades de un sistema origen que comunican de algún evento a otros sistemas del OSCE.
  - ✓ Implementar los servicios de configuración que gestionen el ciclo de vida del evento
  - ✓ Implementar los servicios que orqueste el procesamiento de información en los sistemas destinos.
  - ✓ Realizar la documentación de los servicios de sincronización de información y publicarlo en el repositorio de arquitectura.