SECRETARÍA DE ESTADO DE DEFENSA

DIRECCIÓN GENERAL DE
INFRAESTRUCTURA

SUBDIRECCIÓN GENERAL TIC

INSTRUCCIÓN TÉCNICA

MODELO DE ARQUITECTURA ORIENTADA A SERVICIOS DEL MINISDEF

VERSIÓN 2.0

IT-01/SDGTIC/12/MSOADEF/v2.0

MARZO/2012

Documento

Título:	*MODELO DE ARQUITECTURA ORIENTADA A SERVICIOS DEL MINISDEF*				
Categoría:	Instrucción Técnica	Versión del Documento:	*2.03*	Fecha:	*5/3/2012*
Departamento:	*Subdirección General de Tecnologías de la Información y Comunicaciones*				
Área:	*Área de Tecnologías*				

Control de Firmas

REDACTADO	REVISADO
Jefe de la Unidad de Arquitectura del Área de Tecnologías de la Subdirección General TIC	Jefe del Área de Tecnologías de la SDGTIC
[redacted] Fecha: 2012.03.05 11:06:21 +01'00'	[redacted] Fecha: 2012.03.05 16:58:28 +01'00'

APROBADO
El Subdirector General de Tecnologías de la Información y Comunicaciones
[redacted] Fecha: 2012.03.06 10:15:51 +01'00'

Control de Cambios

VERSIÓN	REVISIÓN	FECHA	OBSERVACIONES
1	0	1DIC09	Primera versión aprobada por IGECIS
1	1	27ABR11	Adecuación a nueva organización de la SDGTIC. Añadidos nuevos casos de uso
1	1	11OCT11	Modificaciones originadas por CCOMSI y Área de Seguridad
1	1	21OCT11	Borrador consolidado de cambios consensuados CCOMSI, Seguridad y Tecnologías
2	0	16DIC11	Versión alineada con WAN 2.0
2	.03	5MAR12	Edición final con informe favorable de CPCMAE

1 INTRODUCCIÓN

1.1 *REFERENCIAS Y BIBLIOGRAFÍA*

A. Real Decreto 1287/2010, de 15 de octubre, por el que se desarrolla la estructura orgánica básica del Ministerio de Defensa.

B. LEY 11/2007, de 22 de junio, de acceso electrónico de los ciudadanos a los Servicios Públicos.

C. Instrucción Técnica de IGECIS "Modelo de Arquitectura I*Net" – Mayo 2006.

D. "Especificación de escenario objetivo", Grupo de Trabajo de WAN 2.0, IGECIS, Febrero 2010.

E. NNEC Feasibility Study v 2.0. NATO Consultation, Command and Control Agency October 2005.

F. Reglamento que desarrolla la ley 11/2007 y Esquema Nacional de Interoperabilidad.

G. "Modelo de Arquitectura Orientada a Servicios del MINISDEF" versión 1.0, IGECIS, Diciembre 2009.

H. Enterprise Service Bus, D. Chappell, O'Reilly, 2004.

I. Service Oriented Architectures, Concepts, Technologies and Design, T.Erl, 2005

J. Design an SOA solution using a reference architecture, IBM, Marzo 2007 http://www.ibm.com/developerworks/library/ar-archtemp/

K. Introduction to SOA governance, B. Woolf, IBM, Jul 2007 http://www.ibm.com/developerworks/library/ar-servgov/

1.2 *ANTECEDENTES*

Según el RD 1287/2010 art. 6 [Ref. A], corresponde a la Subdirección General de Tecnologías de la Información y Comunicaciones desarrollar, entre otras, las siguientes funciones:

- Definir las políticas y estrategias corporativas en el ámbito de las tecnologías de la información, comunicaciones y seguridad de la información del Ministerio de Defensa, así como la planificación y coordinación de las actuaciones en estas materias.
- Dirigir y gestionar, de forma completa e integrada, las infraestructuras, los servicios y el ciclo de vida de los sistemas de información y telecomunicaciones de ámbito corporativo para Propósito General, así como de los sistemas de información que sean de interés específico del Órgano Central.

- Operar un centro corporativo como plataforma única para la prestación de todos los servicios de información y telecomunicaciones de propósito general, asegurando su disponibilidad.

La ley 11/07 [Ref. B] establece el principio de "cooperación en la utilización de medios electrónicos por las Administraciones Públicas al objeto de garantizar tanto la interoperabilidad de los sistemas y soluciones adoptados por cada una de ellas como, en su caso, la prestación conjunta de servicios a los ciudadanos".

Asimismo, en su artículo 41 establece que "las Administraciones Públicas utilizarán las tecnologías de la información en sus relaciones con las demás administraciones y con los ciudadanos, aplicando medidas informáticas, tecnológicas, organizativas, y de seguridad, que garanticen un adecuado nivel de interoperabilidad técnica, semántica y organizativa y eviten discriminación a los ciudadanos por razón de su elección tecnológica".

Por último, en el artículo 43 establece que "la Administración General del Estado, […], adoptarán las medidas necesarias e incorporarán en sus respectivos ámbitos las tecnologías precisas para posibilitar la interconexión de sus redes con el fin de crear una red de comunicaciones que interconecte los sistemas de información de las Administraciones Públicas españolas y permita el intercambio de información y servicios entre las mismas, así como la interconexión con las redes de las Instituciones de la Unión Europea y de otros Estados Miembros."

La I.T. del Modelo I*Net [Ref. C] establece el modelo de arquitectura I*Net (Modelo I*Net) para la Red de Área Extensa de Propósito General (WAN PG) que normaliza los servicios que ofrece dicha red desde o hacia el exterior. El Ministerio de Defensa ha determinado, entre otros, como servicios con el exterior asociados a la WAN PG la publicación de información accesible por Internet, acceso desde el exterior a recursos de la red de Propósito General, acceso a la Intranet Administrativa de la AGE, y acceso mediante líneas dedicadas de otros organismos y entidades al Ministerio de Defensa. Dichas necesidades sumadas a condicionantes de seguridad e interoperabilidad establecidos por el Ministerio de Defensa y normativa vigente del Centro Criptológico Nacional del Centro Nacional de Inteligencia ponen de manifiesto la necesidad de la definición de un modelo de arquitectura que englobe y regule los servicios previamente expuestos.

En el contexto anterior del Modelo I*net y de los requisitos establecidos por la aplicación de la Ley 11/2007 es clara la necesidad de establecer una arquitectura de interoperabilidad uniforme y corporativa que permita al MINISDEF publicar servicios para el ciudadano y para otras administraciones públicas y, de modo simétrico, consumir los servicios expuestos por otras administraciones públicas, en ambos casos de modo seguro, monitorizado y gestionado de manera uniforme, y conforme a los estándares establecidos por el Ministerio de Defensa.

El documento "Especificación de escenario objetivo" de febrero de 2010 elaborado por el grupo de trabajo de la WAN 2.0 [Ref. D], establece una serie de condiciones iniciales que determinan el escenario objetivo al que ha de evolucionar la red corporativa de propósito general del Ministerio de Defensa. La nueva red, denominada WAN 2.0, tendrá en cuenta la situación actual de la red corporativa, las diferentes normas en vigor y un conjunto de requisitos puestos de manifiesto por organismos y usuarios.

La WAN 2.0 estará soportada por un nuevo Modelo de Interconexión del Ministerio de Defensa al que debe adaptarse el modelo SOA objeto del presente documento. Por tanto, esta edición del documento (versión 2.0) es coherente con el nuevo modelo de interconexión y la arquitectura de referencia de la WAN 2.0.

1.3 *OBJETO*

El propósito de este documento es establecer un modelo de arquitectura orientada a servicios de alto nivel que satisfaga los siguientes requisitos:

- Definir el concepto de Arquitectura Orientada a Servicios (SOA) del MINISDEF en el ámbito de la red de Propósito General.

- Marcar estrategias tecnológicas y principios de diseño aplicables a la infraestructura tecnológica que soporte la SOA del MINISDEF, sirviendo como documento maestro para la implantación de SOA en el Departamento.

- Establecer un modelo de arquitectura orientada a servicios que permita cumplir con la política de seguridad del MINISDEF, la normativa aplicable del Centro Criptológico Nacional del Centro Nacional de Inteligencia y la normativa legal vigente.

- Constituir un marco de referencia del que derivan documentos técnicos más detallados para llevar a cabo una implantación efectiva de SOA en el MINISDEF, y las necesarias guías, plantillas y procedimientos que sean requeridos.

- Definir los roles y responsabilidades ligados a la implantación, operación y mantenimiento de la SOA del MINISDEF en la red de Propósito General.

1.4 *ALCANCE*

El presente documento es de aplicación a todos los sistemas de información de la red de Propósito General del MINISDEF, tanto a sistemas funcionales como de plataforma, ya sean corporativos o específicos.

Es de aplicación en lo referente a la solución de requisitos de integración entre sistemas de información dentro del ámbito de la red de propósito general en cualquiera de los dominios previstos por la WAN 2.0.

Es de aplicación a la interacción de sistemas de la WAN 2.0 con agentes externos al MINISDEF vía la infraestructura desplegada en el Nodo de Interconexión (interacción con sistemas de información de terceros vía Internet, red SARA y líneas dedicadas).

Los requisitos de integración de sistemas que resuelve el presente modelo de arquitectura se restringen a la interacción automática entre sistemas de información para intercambiar datos y reutilizar funcionalidades.

Dado que en la fecha de edición de este documento la WAN PG se encuentra en un periodo transitorio de migración hacia WAN 2.0, el modelo SOA refleja esta realidad, y esta versión recoge por un lado la visión final del modelo sobre la WAN 2.0 y la situación actual durante este periodo transitorio.

1.5 *SITUACIÓN DE PARTIDA*

Actualmente, en lo que se refiere a la integración de sistemas de información en la WANPG, la situación de partida se resume en los aspectos siguientes:

- Tecnologías de desarrollo de aplicaciones y bases de datos diversas: .net, J2EE, NATURAL ADABAS, Oracle, SQL-Server, Sybase, etc.
- Carencia de estándares específicos de integración de aplicaciones corporativos en cuanto a guías de diseño, guías de despliegue y operación, protocolos de transporte, tecnologías, seguridad, etc.
- La mayoría de las integraciones entre sistemas son punto a punto con desarrollos a medida, poco flexibles ante cambios y con baja utilización de servicios de middleware.

- No hay definido un modelo de arquitectura de aplicaciones para resolver integraciones con sistemas externos al MINISDEF.
- La concepción de las aplicaciones suele ser vertical dando lugar a una baja reutilización de funcionalidades y sistemas poco flexibles ante los cambios.
- Carencia de un modelo común de arquitectura de aplicaciones de servicios web para exponer funcionalidades a usuarios externos al MINISDEF, lo cual dificulta la adopción de políticas de seguridad, administración y monitorización uniformes para dichos servicios.
- Actualmente hay varios sistemas de información que están utilizando tecnologías e infraestructura SOA para resolver problemas de integración, pero se hace de un modo localizado en cada sistema, sin compartir un modelo de arquitectura común y uniforme, dando lugar a implementaciones denominadas SOAs tácticas.

En línea con los principios de SOA comúnmente aceptados, este modelo de arquitectura no pretende sustituir la variedad de plataformas y tecnologías existentes (que en la mayoría de los casos responde a necesidades operativas concretas y/o a razones históricas), sino aceptar dicha situación de partida y evolucionar gradualmente a un modelo uniforme. Así, el objetivo de este modelo de arquitectura es proporcionar la infraestructura requerida para reutilizar los servicios y procesos ya existentes de modo uniforme en una arquitectura de integración corporativa, dando lugar de este modo a una SOA estratégica que complemente a las diversas SOA's tácticas ya existentes.

1.6 *PRINCIPIOS*

A continuación se enumeran una serie de principios, de aplicación para el diseño del modelo de arquitectura SOA del MINISDEF:

- **Solución corporativa**. Solución capaz de abarcar las necesidades de integración de los distintos sistemas de información de la WAN 2.0, independientemente de sus plataformas tecnológicas, tanto en interacciones internas como con el exterior.

- **Robustez y perdurabilidad** en el tiempo de la Arquitectura. Se pretende un modelo que sea lo más independiente posible de la evolución de la tecnología y los productos software con que se implemente.

- **Flexibilidad**. El modelo debe ser flexible y evolucionar de acuerdo a los cambios tecnológicos y las nuevas necesidades de integración de los sistemas de la WAN 2.0.

- **Neutralidad tecnológica** basada en estándares abiertos, independiente de la plataforma y del lenguaje de programación. De este modo se garantiza soportar las necesidades de integración de la mayor parte de los sistemas, independientemente de su tecnología. La utilización de estándares abiertos facilita la elección de distintos fabricantes y la interoperabilidad de las distintas plataformas de desarrollo de sistemas.

- **Escalabilidad**. El modelo debe permitir implantar soluciones técnicas fácilmente escalables de modo que soporte tanto necesidades de integración con requisitos de baja criticidad y volúmenes de datos discretos como necesidades más exigentes en volumen, rendimiento y fiabilidad.

1.7 *GLOSARIO DE TÉRMINOS*

Este apartado incluye la definición de los términos clave para comprender el modelo de arquitectura propuesto.

Termino	Definición breve
SOA - Arquitectura Orientada a servicios	Modelo de arquitectura que soporta servicios débilmente acoplados para facilitar la flexibilidad del negocio de forma interoperable e independiente de la tecnología. Una arquitectura en que las funciones se definen como servicios con interfaces bien definidos que pueden invocarse separadamente o en secuencias predefinidas, creando procesos de negocio. El foco de la arquitectura está en el "interfaz de servicio" que se define en términos del los parámetros requeridos por el servicio y de la naturaleza del resultado de su invocación. Una SOA permite que los servicios sean publicados, descubiertos y utilizados. NNEC Feseability Study [Ref. E]
Servicio	Funcionalidad concreta, autocontenida y con significado de negocio que, proporcionada por una aplicación informática, está disponible on-line en una red para poder ser reutilizada por otras aplicaciones que la requieran, permitiendo además la posibilidad de crear nuevas aplicaciones a partir de la composición de *servicios* ya existentes.

Termino	Definición breve
Interoperabilidad de sistemas	Capacidad de los sistemas de compartir datos, combinada con la capacidad de utilizarlos o comprenderlos, permitiendo que dichos sistemas operen conjuntamente de forma eficiente (ISO TC 204). Capacidad de los sistemas de información, y por ende de los procedimientos a los que éstos dan soporte, de compartir datos y posibilitar el intercambio de información y conocimiento entre ellos. Ley 11/07 [Ref. B].
ESB - Bus de Servicios	Plataforma de integración basada en estándares que combina mensajería entre aplicaciones, webservices, transformación de datos y enrutado inteligente con el fin de conectar y coordinar de modo fiable la interacción de un número significativo de aplicaciones diversas a través de una o más organizaciones, garantizando seguridad e integridad transaccional.
WAN PG	Red de Área Extensa de Propósito General. Red corporativa del Ministerio de Defensa.
Gobierno *SOA*	*Gobierno SOA* establece los mecanismos y políticas necesarios para asegurar que los principios de orientación a servicios y la arquitectura son gestionados adecuadamente, enfocándose en la gestión del ciclo de vida de los servicios para que se cumplan los objetivos de la organización. El Gobierno SOA abarca los procesos de una organización para asegurar que la adopción, implementación y gestión de SOA se hace de acuerdo con las buenas prácticas, principios arquitectónicos, regulaciones de control y políticas de seguridad que mejor se adapten a la organización, garantizando además el cumplimiento de la normativa legal vigente.
Modelo Común de Datos (CDM) Canonical Data Model	Es un modelo de datos compuesto por una colección de esquemas XML que proporciona una visión y formato común de las entidades de negocio de una organización de modo independiente del modelo de datos de cada aplicación. Permite desacoplar e independizar la integración de aplicaciones que usan modelos de datos distintos, requiriendo que cada aplicación produzca y consuma mensajes en este formato común.

Termino	Definición breve
Centro de competencia SOA	La adopción de nuevos procedimientos y prácticas implica cambios organizativos, siendo necesario disponer de un grupo dentro de la organización que mantenga, optimice y supervise su cumplimiento. Este centro de competencia debe gestionar y supervisar a las personas, procedimientos y tecnologías involucradas durante todo el ciclo de vida SOA.

2 LA ARQUITECTURA ORIENTADA A SERVICIOS

2.1 *INTEROPERABILIDAD DE SISTEMAS*

El logro del necesario grado de interoperabilidad entre los sistemas es imprescindible para que los recursos CIS apoyen de modo eficiente al MINISDEF en el desarrollo de sus funciones y constituye un requisito fundamental tanto en los sistemas de propósito general como en los del área operativa.

Según ISO TC 204, la interoperabilidad puede definirse como la capacidad de los sistemas de compartir datos, combinada con la capacidad de utilizarlos o comprenderlos, permitiendo que dichos sistemas operen conjuntamente de forma eficiente.

En este contexto, interoperabilidad se refiere al intercambio de datos entre sistemas o aplicaciones informáticas de modo automático, sin intervención humana.

No se consideran los intercambios de datos producidos en los siguientes contextos:

- Intercambio de información destinado a personas, sin procesamiento automático. Por ejemplo, la difusión de información a través de servidores (o portales) web, o el empleo de anexos a mensajes de correo electrónico que se distribuyen.
- Internamente en las aplicaciones, entre componentes o módulos pertenecientes a una misma aplicación. Sin embargo, la infraestructura SOA puede ser utilizada por un sistema distribuido para intercambiar datos entre distintos componentes desplegados del sistema.

2.2 *DEFINICIÓN DE ARQUITECTURA ORIENTADA A SERVICIOS*

SOA es un modelo de arquitectura de software que proporciona un enfoque modular y flexible para la implementación de requisitos funcionales en los sistemas. En una SOA las funciones software de los sistemas se exponen como **servicios reutilizables** que pueden ser descubiertos e invocados a través de la red. El uso de SOA facilita la compartición de aplicaciones y datos, proporcionando un mecanismo flexible para reutilizar los servicios, habilitando el desarrollo de nuevas aplicaciones mediante la composición de servicios ya existentes.

Un objetivo fundamental de SOA es hacer que los recursos de información estén **disponibles en la red** para todos los potenciales sistemas consumidores, y soportar el descubrimiento y entrega eficiente de dicha información al sistema consumidor.

El desarrollo del concepto SOA está ligado a los **requisitos cambiantes** de los entornos corporativos, lo que provoca la necesidad de adaptarse más rápidamente a los cambios. SOA facilita el desarrollo de aplicaciones nuevas basadas en la reutilización de los servicios disponibles en la red.

SOA es también consecuencia de la búsqueda de **acoplamiento débil entre sistemas**, de modo que un sistema pueda utilizar las funciones y datos de otro mediante interfaces bien definidos. Este acoplamiento débil debe ser lo más neutral posible desde el punto de vista tecnológico, facilitando la evolución independiente de cada sistema.

SOA puede definirse como un modelo de arquitectura para la creación de sistemas modulares, que pueden interoperar mediante **interfaces** bien definidas e **independientes** de la tecnología en la que estén implementados, que utilizan **metadatos estándar** y cuya localización es posible gracias a un **registro de servicios** estandarizado. Estos servicios pueden invocarse de modo independiente o en secuencias definidas para formar **procesos de negocio**.

Los sistemas están débilmente acoplados y se ven los unos a los otros como servicios, accesibles vía interfaces estándar sin conocer la implementación subyacente del servicio. Un **bus de servicios** gestiona y coordina la interacción entre los servicios. Los datos se intercambian mediante un formato común usando protocolos estándar, lo cual asegura la interoperabilidad. Todo ello permite que se produzcan interacciones sistema-a-sistema no previstas en el momento del diseño de cada sistema.

2.3 *ROLES DE SISTEMA EN UNA SOA*

Cualquier SOA contiene tres roles básicos: *consumidor* de servicios, *proveedor* de servicios e *intermediario* de servicios.

1. Un *proveedor* de servicios es responsable de la creación y descripción de un servicio, publicarlo en un *intermediario de servicios*, y recibir los mensajes de invocación del servicio desde uno o más *consumidores* a través del *intermediario de servicios*.

2. Un *consumidor* de servicios es responsable de encontrar una descripción de un servicio en un *intermediario de servicios*, y utilizar la descripción para invocarlo.

3. Un *intermediario de servicios* es responsable de *publicar* las descripciones de servicios publicadas por los *proveedores* y de permitir a los *consumidores* descubrir e invocar servicios.

La Plataforma de Interoperabilidad y Gobierno SOA del MINISDEF debe proporcionar una serie de servicios comunes que permitan a cualquier sistema de información de la WANPG actuar como *proveedor* o *consumidor* de servicios, asumiendo esta Plataforma el rol de *intermediario de servicios*.

2.4 *DEFINICIÓN DE SERVICIO*

En SOA se pretende que los *servicios*, entendidos estos como funcionalidades concretas y autocontenidas en las que se descompone un sistema de información, estén disponibles para:

- Poder ser reutilizados por otros sistemas que los requieran, o

- Crear nuevas aplicaciones ad-hoc desarrolladas a partir de la composición de servicios ya existentes.

2.4.1 Características de los servicios

1. **Vista lógica**: Un servicio es una vista lógica y abstracta de un programa, bases de datos, o cualquier otro componente software que se define en términos de la operación de *negocio* que realiza.

2. **Orientación a mensajes**: Un servicio se define formalmente en términos de los mensajes intercambiados entre los componentes software proveedores y consumidores del servicio, y no de las propiedades de dichos componentes. La estructura interna de un componente, tal como su lenguaje de implementación, estructura de proceso, estructura de base de datos, etc. se abstrae de modo deliberado: no debe ser necesario conocer cómo un componente implementa un servicio.

3. **Orientación a la descripción**: Sólo los detalles que se exponen en la red y son importantes para usar el servicio deben incluirse en su *descripción*, que debe cumplir la capacidad de ser procesable de modo automático por un programa de software. Un servicio se describe mediante metadatos "interpretables por máquina" como contraposición a una descripción "destinada a personas". La semántica de un servicio debe incluirse en su descripción.

4. **Orientación a la red**: Los servicios deben tender a estar visibles y ser utilizados a través de la red.

5. **Neutralidad de formato**: Los mensajes para invocar servicios se envían en un formato estándar neutral respecto a los sistemas proveedores y consumidores. XML es el formato más adecuado para cumplir este requisito.

6. **Integración de sistemas legados mediante servicios**. La integración de sistemas legados se facilita mediante la adición de componentes software para "envolver" los servicios de dichos sistemas y permitirles adherirse a una definición de servicios estándar de la SOA.

7. **Visión de negocio**: Lo que diferencia un *servicio* de una simple llamada a una rutina remota de otro sistema es que el *servicio* tiene significado de *negocio*. Los *servicios* pueden ser simplemente para consultar información u ofrecer funcionalidades más complejas que implican proceso y actualización de datos en el sistema que ofrece el *servicio*.

8. **Contrato de interfaz de servicio**: Cuando un sistema ofrece un *servicio* establece un *contrato de interfaz*[1], de modo que se *compromete* a proporcionar ese *servicio* en las condiciones establecidas en dicho *contrato de interfaz*. Estos *contratos de*

[1] En este contexto el término *contrato de interfaz* es un concepto técnico que no tiene ninguna connotación administrativa o legal.

interfaz deben ser independientes de la tecnología subyacente que implementa el *servicio,* posibilitando simultáneamente la evolución tecnológica independiente de cada sistema y una gestión uniforme de servicios.

2.5 *GOBIERNO SOA*

El "Gobierno SOA" (*SOA Governance*) se centra en la gestión del ciclo de vida de los servicios con el objetivo de garantizar el valor de negocio de SOA para la organización. Se refiere a las actividades relacionadas con el ejercicio de control sobre los elementos que componen la arquitectura, afectando especialmente al ciclo de vida de los servicios.

2.5.1 Propósito del Gobierno SOA

La necesidad de gobierno SOA es consecuencia de la inherente granularidad y complejidad de la arquitectura, en la que diferentes elementos (sistemas proveedores y consumidores, componentes de servicio, bus de servicios, procesos de negocio, etc.) cooperan para resolver necesidades de negocio comunes y además, frecuentemente, bajo ámbitos de responsabilidad diferentes.

A lo largo del tiempo los servicios deben adaptarse a nuevas necesidades y cambiar su funcionalidad y comportamiento. La alineación entre requisitos de la organización y tecnología implica adaptarse a cambios constantes y evitar la creación de servicios con funcionalidades similares o la existencia de múltiples versiones de un mismo servicio de modo innecesario.

El propósito de establecer *Gobierno SOA* en una organización consiste en ejercer control mediante el establecimiento de políticas para la gestión y monitorización de los servicios tanto desde un punto de vista técnico (disponibilidad de los servicios) como de negocio (qué servicios son necesarios, cómo y quién los autoriza, quién los realiza, etc.).

2.5.2 Definición de Gobierno SOA

Gobierno SOA establece los mecanismos y políticas necesarios para asegurar que los principios de orientación a servicios y la arquitectura que los soporta son gestionados adecuadamente, enfocándose en la gestión del ciclo de vida de los servicios para que se cumplan los objetivos de la organización.

Gobierno SOA son los procesos que una organización establece para asegurar que los servicios son conformes a buenas prácticas, principios de diseño, normativa de seguridad y aspectos legales de aplicación en la organización. Estos procesos deben gobernar la adopción, implementación y evolución de la propia SOA, afectando a las personas, procesos y tecnologías involucradas en el ciclo de vida de los servicios, asegurando los niveles de calidad y seguridad requeridos.

Los aspectos en los que se centra el *Gobierno SOA* son:

- **Cumplimiento de estándares y normas**: El comportamiento y dependencias entre servicios debe ser conocido y estar controlado en el marco del cumplimiento de los requisitos normativos vigentes, especialmente los relacionados con la seguridad de la información.

- **Gestión del cambio**: Cuando se cambia un servicio es necesario prever qué impacto puede tener sobre otros sistemas. En una SOA avanzada los consumidores de un servicio pueden ser desconocidos por el proveedor en tiempo de diseño del sistema. Sin embargo las interacciones entre servicios en tiempo de ejecución deben ser controladas y auditadas.

- **Asegurar la calidad de los servicios**: La flexibilidad de una SOA para añadir nuevos servicios requiere una atención especial a la calidad del servicio, tanto en su diseño como en su explotación, dado que el mal funcionamiento de un servicio puede afectar a varias aplicaciones.

2.5.3 Instrumentos del Gobierno SOA

La Plataforma de Interoperabilidad y Gobierno SOA del MINISDFE debe incluir componentes y herramientas software para facilitar el Gobierno SOA, complementando los procedimientos y normativas que establezcan buenas prácticas para una gestión completa del ciclo de vida de los servicios.

Según el momento en que se produce, el gobierno SOA se presenta en dos ámbitos:

- **Gobierno SOA en tiempo de diseño**: Asegurar que la implementación de los servicios durante el diseño y desarrollo de las aplicaciones se hace de acuerdo con las normas establecidas. Afecta tanto a la producción como al consumo de los servicios.
- **Gobierno SOA en tiempo de ejecución**: Asegurar y validar que los servicios están actuando como se espera y que mantienen el nivel de calidad y seguridad

requerido en tiempo de producción/explotación de los sistemas.

Para llevar a cabo el Gobierno SOA en la organización se requiere establecer una serie de roles y responsabilidades. Esta cuestión es objeto del capítulo 4 del presente documento.

Desde un punto de vista tecnológico los elementos de la arquitectura que soportan el Gobierno SOA son:

- **Registro de Servicios**: Repositorio de información donde se publican y categorizan las descripciones de servicios disponibles junto con sus, interfaces, restricciones de uso, condicionantes de seguridad, versionado, etc. Se utiliza primordialmente en tiempo de diseño.

- **Registro de Metadatos**: Repositorio de información dónde se publican y categorizan los metadatos estructurales utilizados por los sistemas de información. En el contexto de SOA, en el Registro de Metadatos se publican y gestionan los esquemas de datos y otros artefactos (representados en XML, tales como XML-schemas, DTD's,…) utilizados para componer los mensajes que intercambian los servicios cuando se invocan. Sólo se utiliza en tiempo de diseño.

- **Herramientas de monitorización**: En tiempo de ejecución monitorizan el estado de los servicios en su aspecto técnico (disponibilidad, rendimiento, errores, log,..) soportando la supervisión del cumplimiento de los posibles SLA's que se establezcan. Adicionalmente monitorizan los propios elementos de la infraestructura SOA.

- **Elementos de seguridad**: En tiempo de ejecución supervisan el cumplimiento de las políticas de seguridad establecidas para cada servicio.

2.5.4 Gestión del Ciclo de Vida de los servicios

El principal aspecto del que se ocupa el Gobierno SOA es la supervisión de la creación de servicios. Los servicios deben ser identificados, su funcionalidad descrita, su comportamiento determinado y sus interfaces diseñados. La función de Gobierno SOA no realiza estas tareas pero se asegura de que son realizadas conforme a las normas de la organización, coordinando a los equipos implicados y evitando la duplicación de esfuerzos.

Los servicios son componentes software y como tales, deben ser planeados, diseñados, desarrollados, desplegados, mantenidos y, en última instancia, retirados. El ciclo de vida de un servicio tiene mayor impacto que otro tipo de componentes software puesto que de él suelen depender varias aplicaciones.

Se distinguen cinco fases o estados en el ciclo de vida de un servicio:

- **En desarrollo:** Está identificado y en desarrollo, pero aún no está implementado.
- **En pruebas:** Una vez está desarrollado, es necesario probarlo en los diversos entornos dónde se va a desplegar.
- **En Operación:** El servicio está plenamente desarrollado y cumple su funcionalidad. Además ha sido desplegado en el entorno de producción.
- **En abandono:** Está todavía en operación, pero está prevista su retirada en un plazo determinado. Es una advertencia a los consumidores para que dejen de usarlo.
- **Retirado:** Este estado final indica que el servicio ya no se provee. Antes de alcanzar este estado es necesario advertir formalmente y siguiendo un procedimiento adecuado a los consumidores de este servicio. Las fechas previstas de puesta "en abandono" y "retirada" deben estar previstas en el posible SLA del servicio.

Versionado de Servicios

Una vez que un servicio está disponible surge la necesidad de cambios: corrección de errores, nueva funcionalidad, retirada de funcionalidad no requerida, cambios de interfaz, etc.). Hay que balancear esa necesidad de cambio con la operación de las aplicaciones consumidoras del servicio, por tanto, es necesario mantener varias versiones de un mismo servicio. El versionado soluciona este problema, pero también introduce la necesidad de la migración de servicios.

Migración de Servicios

Cuando un consumidor empieza a usar un servicio se crea una dependencia, que debe ser gestionada. Debe planificarse adecuadamente la migración periódica a nuevas versiones del servicio, beneficiando tanto al proveedor como al consumidor del servicio que puede incorporar nueva funcionalidad. Sin embargo, debido a razones diversas (presupuesto, recursos, aplicaciones legadas, prioridades,…) las migraciones de servicios no son sencillas, por tanto hay que mantener un balance entre las distintas versiones soportadas por el proveedor del servicio y los planes de migración de los consumidores.

Registro de Servicios

El Registro de Servicios es el medio que utilizan los proveedores de servicios para publicarlos junto con la información necesaria para poder consumirlos.

El Registro de Servicios también facilita la gestión de versiones, incluyendo el seguimiento de la compatibilidad entre las distintas versiones de un servicio y su estado (en desarrollo, pruebas, activo, etc.).

Adicionalmente, el Registro debe mantener las dependencias de servicios llevando el seguimiento de qué servicios consume qué aplicación consumidora, permitiendo notificar de los cambios previstos a las partes interesadas.

Modelo de datos canónico (común)

En una invocación de servicios, el consumidor y el proveedor deben acordar el formato de mensajes intercambiados (como esquemas XML). A medida que el número de servicios crece, y puesto el formato de mensajes puede ser acordado por equipos de desarrollo diferentes, el número de formatos y personas involucradas se hace inmanejable.

Para paliar este problema se utiliza un **Modelo de Datos Canónico (**o común), compuesto por un conjunto de formatos (esquemas XML) independiente de cualquier aplicación y sin embargo compartido por todas las que lo requieran. De este modo, los equipos de desarrollo implicados acuerdan qué formatos de mensajes van a utilizar de los ya existentes en el **modelo de datos canónico**. Este modelo de datos se focaliza en la semántica y sintaxis de los datos con contenido funcional, adicionalmente es necesario acordar los detalles técnicos de los mensajes tales como cabeceras, carga útil, estructura, etc. pero la referencia del modelo de datos canónico proporciona previamente el acuerdo en el contenido fundamental del mensaje.

El **Modelo de Datos Canónico** estará publicado y gestionado en el Registro de Metadatos de Defensa, en forma de recursos XML (esquemas XML, DTD's, etc) junto con el resto de metadatos y documentación asociada para facilitar su uso a los desarrolladores de sistemas.

Monitorización de Servicios

Una aplicación compuesta, que combina la invocación de varios servicios, será tan fiable y robusta como lo sean los servicios de los que depende. Puesto que varias aplicaciones comparten un servicio, el fallo de un sólo servicio puede afectar a varias aplicaciones. Opcionalmente deben definirse SLA's que fijen la fiabilidad y rendimiento que deben esperar los consumidores de un servicio. Los proveedores de servicios deben ser monitorizados apoyándose para ello en los elementos de infraestructura, para garantizar el cumplimiento de los SLA's que se establezcan.

La monitorización de servicios debe ser proactiva, en el sentido de que debe detectar picos de carga, caída de servicios, etc. de modo que se advierta la situación de error antes de que sea crítica. Adicionalmente la monitorización de servicios debe proporcionar mediciones del uso de los servicios para detectar los más usados e incrementar la capacidad de los recursos de infraestructura que los proporcionan.

Responsabilidad (propiedad) de los servicios

En la responsabilidad o propiedad de cada servicio intervienen tres roles de la organización:

- El usuario funcional a cuya necesidad responde la existencia del servicio.
- El equipo de desarrollo del servicio
- El equipo de operaciones de TI que administra la explotación del servicio y lo monitoriza.

Estos tres roles son los habituales de cualquier sistema de información, sin embargo en este caso hay que compaginar los intereses no sólo de los responsables directos del servicio, sino también de los responsables de las aplicaciones consumidoras del servicio, y de los responsables de la administración de la infraestructura SOA (como por ejemplo, el registro de servicios).

Adicionalmente, a medida que la SOA evoluciona en la organización aparecen servicios más complejos, compuestos a partir de otros servicios más básicos. La responsabilidad sobre estos servicios corporativos es difícil de definir.

Por tanto, es necesaria la existencia de alguna unidad o grupo dentro de la organización que coordine estos diferentes intereses.

Pruebas de Servicios

El despliegue de un servicio debe realizarse una vez se ha comprobado que el servicio ha superado unas pruebas adecuadas. La responsabilidad de estas pruebas corresponde al sistema proveedor del servicio.

Para beneficiarse del principio de reutilización de SOA, los consumidores del servicio no deberían probar el servicio, sino confiar en que el servicio funciona adecuadamente según sus especificaciones, responsabilizándose únicamente de las pruebas de su propia aplicación y de las pruebas de interfaz o conectividad con el servicio.

Por tanto, es necesario establecer un nivel de exigencia de pruebas uniforme para todos los proveedores de servicios.

2.6 *SEGURIDAD EN SOA*

Obviamente, el objeto sobre el que se aplica los condicionantes de seguridad en una SOA es el *servicio*, respecto al que hay que cubrir las necesidades siguientes:

- Autenticación: Identificación del consumidor del servicio.
- Autorización para consumir el servicio.
- Protección (confidencialidad, integridad y no repudio) de los datos intercambiados en la invocación del servicio.

- Auditoría de las invocaciones a los servicios.

Estos aspectos de seguridad son los habituales de cualquier componente software, sin embargo en SOA se añaden los siguientes factores:

- Aumento de la vulnerabilidad de las aplicaciones, puesto que ofrecen más puntos de acceso a su funcionalidad en forma de servicios.
- Las invocaciones a servicios pueden producirse entre dominios de seguridad diferentes, como por ejemplo entre USO OFICIAL y una entidad externa vía Internet.
- Los consumidores de los servicios no son usuarios finales, sino aplicaciones. Sin embargo, la autorización para acceder al contenido del servicio puede depender de la identidad del usuario final de la aplicación.
- Las necesidades de protección de datos pueden no ser las mismas para todos los posibles consumidores de un servicio, dependiendo por ejemplo de en cuál entorno reside la aplicación consumidora.
- La necesidad de protección se aplica a todas las interacciones posibles en la SOA: publicación, descubrimiento e invocación de los servicios.

Por tanto, el modelo de arquitectura debe proporcionar soluciones tecnológicas para satisfacer estas necesidades:

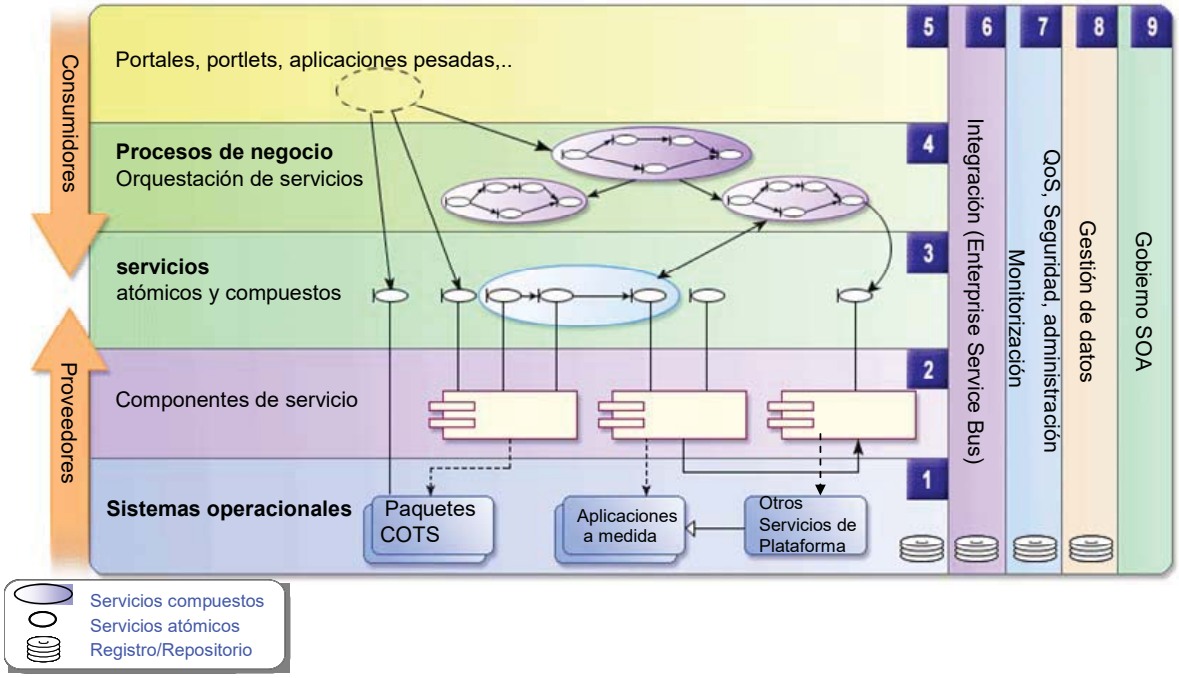
- Estándares y tecnologías de invocación de servicios que soporten las capacidades de seguridad requeridas.
- El acceso a los servicios (incluso a la descripción de los servicios disponibles) debe estar limitada a los usuarios y aplicaciones autorizadas.
- La identidad de usuario final debe poder propagarse desde la aplicación consumidora de un servicio hasta la aplicación proveedora cuando se requiera.
- El control final de autorización de acceso a los datos debe residir en la aplicación proveedora del servicio.
- Los diferentes niveles de protección de datos asignado a un servicio deben poder establecerse mediante *policies* (reglas) declarativas, uniformizando la gestión de la protección de datos e independizándola en lo posible de las aplicaciones consumidoras y proveedoras de los servicios. Adicionalmente, se pretende desacoplar en lo posible el desarrollo de servicios de la aplicación de políticas de seguridad sobre los mismos.
- La plataforma debe contar con elementos de infraestructura específicos que auditen las interacciones entre servicios y generen eventos producidos por incidentes de seguridad como por ejemplo: intentos de invocación no autorizados, fallos de autenticación, aumento de invocaciones a servicios fuera de lo habitual, etc.
- Capacidades de intermediación de servicios que garanticen restricciones sobre los servicios previamente establecidas (como por ejemplo número máximo de

invocaciones por periodo, máximo de invocaciones simultáneas, etc.)

3 MODELO DE ARQUITECTURA SOA DEL MINISDEF

3.1 *MODELO DE ARQUITECTURA SOA POR CAPAS*

La figura siguiente representa el modelo de arquitectura orientada a servicios para el MINISDEF.



El modelo de arquitectura se descompone en nueve capas lógicas. Las capas horizontales (1 a 5) engloban los componentes de arquitectura que satisfacen los requisitos funcionales. Las capas verticales (6-9) son componentes comunes de naturaleza no funcional que, además de proporcionar capacidades de integración, seguridad, monitorización, etc. permiten el gobierno adecuado de la arquitectura, tanto en tiempo de diseño como en ejecución.

Respecto a este modelo se asumen los siguientes condicionantes:

- No existe una división estricta en las dependencias entre las distintas capas. Ejemplo: un consumidor puede acceder a la capa de procesos de negocio o directamente a un servicio prestado por un sistema operacional en el nivel más inferior.
- El modelo es flexible, no es necesario que un sistema implemente todas las capas. Ejemplo: inicialmente un sistema puede excluir la capa de procesos de negocio, esta solución no se beneficiará de esa capa, pero podrá añadirse en una evolución del sistema posterior.
- El modelo es evolutivo: el grado en que la SOA del MINISDEF implementará todas

las capas del modelo evolucionará de acuerdo al nivel de madurez en los servicios de integración impuestos por los requisitos funcionales de los sistemas.

Capa 1. Sistemas Operacionales

Incluye todas los sistemas o aplicaciones (COTS o desarrollos a medida) existentes o a desplegar en la WAN PG, que dan soporte a las necesidades funcionales del MINISDEF. Son sistemas nuevos o legados basados en plataformas tecnológicas variadas tales como J2EE, .net, NATURAL/ADABAS, etc.

Capa 2. Componentes de Servicio

Esta capa está constituida por componentes software que proporcionan la implementación para la ejecución de un servicio. Refleja la definición de un servicio, tanto funcionalmente como la calidad de servicio que proporciona.

Debe proporcionar un punto de refuerzo para una ejecución fiable del servicio, asegurando la calidad del servicio y el cumplimiento de posibles SLA's. Permite la composición y estratificación de los servicios disponibles en la WANPG.

Capa 3. Servicios

Esta capa consta de todos los servicios definidos en la SOA, considerando el servicio como una especificación abstracta de una serie de capacidades de sistemas de información que responde a requisitos funcionales específicos. La especificación proporciona a los consumidores el suficiente detalle para invocar el servicio de modo independiente de la plataforma tecnológica del proveedor.

Las especificaciones de interfaz de servicio estarán descritas en WSDL, añadiendo documentación adicional sobre las políticas de uso, gestión, restricciones de uso, políticas de seguridad aplicables, etc. Estas descripciones permiten el acceso autorizado a los servicios a través de la red.

Los servicios deben ser accesibles independientemente de su implementación y protocolo de transporte de nivel de aplicación.

Capa 4. Procesos de Negocio

Las composiciones y orquestaciones de los servicios expuestos en la capa 3 se definen en esta capa. La composición y orquestación de servicios se usa para combinar grupos de servicios en flujos, permitiendo construir aplicaciones de usuario final a partir de los servicios. Estas aplicaciones son las que soportan procesos de negocio y casos de uso. Para diseñar estos procesos de negocio se usan herramientas software que permiten componer visualmente los flujos de servicios.

Capa 5. Presentación

Esta capa provee las capacidades requeridas para proporcionar el interfaz de usuario final. Se deben proveer capacidades para la creación rápida de interfaces de usuario para procesos de negocio y aplicaciones compuestas. En función de los tipos de usuarios y entornos disponibles se requiere el uso de distintas tecnologías de interfaz de usuario tales como portales web, clientes pesados, interfaces adaptadas para PDA, etc. Sin embargo, si el diseño de los servicios es correcto, todas esas diversas implementaciones de interfaz de usuario podrán invocar a los mismos servicios que proveen la funcionalidad de la aplicación.

Capa 6. Integración – ESB (Enterprise Service Bus)

Provee la capacidad de mediar, enrutar y transportar los mensajes de requerimiento de servicio desde el consumidor hasta el proveedor de servicios correcto. Las capacidades requeridas en esta capa son proporcionadas por un ESB. Aunque el proveedor de un servicio sea capaz de proporcionar descripciones válidas de servicio en forma de WSDL, el ESB también provee independencia de localización y proporciona a los servicios funcionales una capa en la que se integra con el resto de capacidades no funcionales de la arquitectura como son seguridad, monitorización, gestión de errores, etc.

Capa 7. Calidad de Servicio, Seguridad, Monitorización y Gestión

Proporciona a la SOA las capacidades requeridas para satisfacer los requisitos no funcionales. Se ocupa de capturar, monitorizar, registrar y señalizar el incumplimiento de las calidades de servicio requeridas. Esta capa sirve como un "observador" de las otras capas y puede generar eventos o notificaciones cuando se detecta una situación de incumplimiento, o preferiblemente se anticipa dicha situación.

Asegura que se cumplen los requisitos con respecto a fiabilidad, disponibilidad, rendimiento, gestionabilidad, escalabilidad y seguridad.

Capa 8. Gestión de Datos

Incluye todo lo relacionado con la gestión del modelo de datos canónico (común) de la SOA y cualesquiera otras especificaciones de datos (preferentemente en XML) usadas en todos los componentes de la SOA.

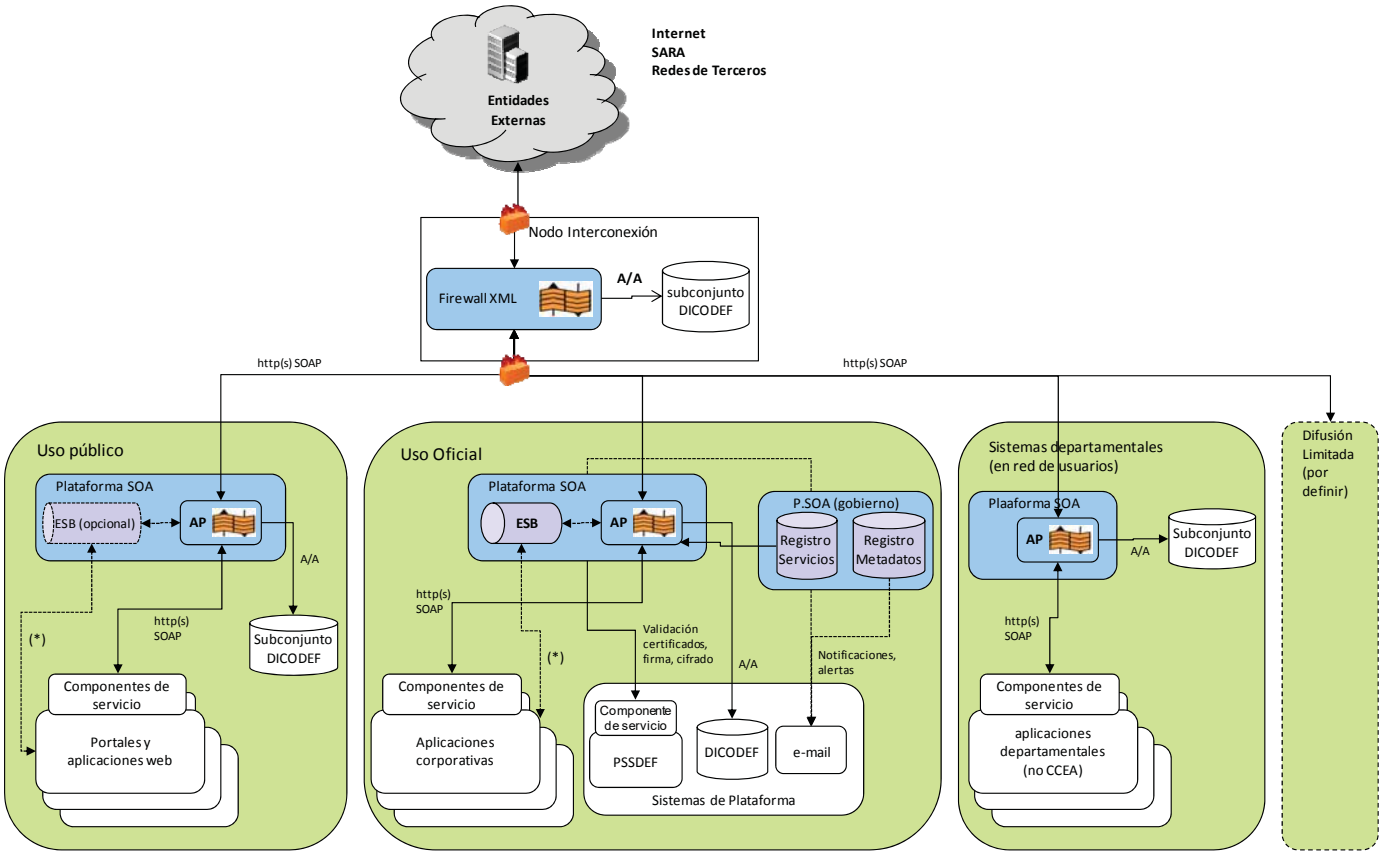
Esta capacidad está cubierta por el Registro de Metadatos de Defensa disponible actualmente en la WAN PG.

Capa 9. Gobierno SOA

Esta capa cubre todo los aspectos relacionados con la gestión del ciclo de vida de los servicios SOA. Debe proporcionar planeamiento, dirección y control para tomar decisiones sobre la SOA y dirigir la gestión todos los aspectos globales de la SOA, incluyendo capacidad, rendimiento, seguridad y monitorización.

3.2 *PLATAFORMA DE INTEROPERABILIDAD Y GOBIERNO SOA*

En el diagrama siguiente se resume la arquitectura lógica de alto nivel definida para soportar la SOA corporativa del MINISDEF, reflejando la relación con los dominios previstos en el escenario objetivo de la WAN 2.0.



En color azul claro (gris oscuro en caso de impresión en b/n) se muestran los elementos que conforman la plataforma SOA, en qué dominio están desplegados y su interacción con otros sistemas.

En los dos apartados siguientes se detallan características generales de esta arquitectura y la descripción detallada de cada elemento.

3.2.1 Características generales

- La plataforma permite la interacción de servicios entre sistemas/aplicaciones de los distintos dominios, considerando los sistemas de organizaciones externas al MINISDEF como un dominio más.
- En cada dominio existe un elemento que asegura que las interacciones de servicios se producen bajo las medidas de seguridad y monitorización adecuadas. Estas capacidades están cubiertas por los agentes de políticas de servicios web que actúan como "reguladores de servicios entre sistemas" aplicando las políticas de autenticación y autorización requeridas en cada caso. Cuando el servicio invocado está en otro dominio, el agente dirige la petición al agente de políticas del dominio correspondiente, de este modo el "tráfico de servicios" entre dominios está siempre controlado por políticas uniformes.
- El firewall XML es el componente que se utiliza para controlar las interacciones de servicios con sistemas externos al MINISDEF (vía internet, SARA o cualquier otra red de terceros).
- Los agentes de políticas y el firewall XML se utilizan para controlar las interacciones de servicios en la WAN 2.0 de dos formas: dentro de su propio dominio (un "salto"), o de un dominio a otro (dos "saltos"). En caso de relacionarse con el exterior, lo harán siempre vía firewall XML (dos "saltos"). Este comportamiento se detalla más abajo en los casos de uso genérico que permite la Plataforma SOA.
- La tecnología de servicios mediados por la plataforma será preferentemente de webservices basados en http(s)/SOAP. Como se deduce de la figura, las interacciones de servicios entre dominios de seguridad diferentes deben basarse en http(s)/SOAP de modo obligatorio.
- Las peticiones/respuesta (http(s)/SOAP) entre cada elemento se resuelven a nivel de protocolo de aplicación, de modo que en cada segmento de un flujo es posible personalizar el protocolo de transporte utilizado (http, https,…) o añadir características de seguridad a nivel de mensaje (por ejemplo: cifrado total o parcial, firma digital, tokens,…).
- En los dominios que se requiera, se desplegará un bus de servicios (ESB) que proporciona capacidades de integración entre aplicaciones. El ESB provee una capa de componentes de servicio que permite a la aplicación ofrecer webservices http(s) SOAP susceptibles de ser gestionados por los agentes de políticas.
- Tanto los agentes de políticas como el firewall XML pueden utilizar DICODEF para autenticar/autorizar a una aplicación/sistema para consumir un servicio.

- Los registros de servicios y de metadatos son aplicaciones web en el dominio de Uso Oficial pudiendo ser accedidas desde fuera de la WAN 2.0 (internet y/o SARA). Los desarrolladores del MINISDEF los utilizarán para publicar y consultar descripciones de webservices y esquemas XML utilizados en los servicios disponibles en la plataforma.

3.2.2 Descripción de los elementos de la plataforma

Se detalla a continuación cada uno de los elementos de la arquitectura:

- Las **aplicaciones corporativas** en el dominio de uso oficial pueden ser sistemas legados o nuevas aplicaciones, estando basados en diferentes plataformas tecnológicas (J2EE, .net, Natural/ADABAS, etc.). Mediante componentes de servicio se conectan al bus de servicios usando protocolos de transporte de nivel de aplicación tales como http, ftp, jdbc, JMS, SMTP, etc. Esta conexión se ha indicado en el dibujo con la llamada (*).

  De modo similar, existen **portales y aplicaciones web** en el dominio de uso público susceptibles de interaccionar con aplicaciones de otros dominios mediante webservices. A priori se prevé que estas aplicaciones sólo utilicen webservices http(s)/SOAP por lo que no requieren de la mediación proporcionada por un bus de servicios, siendo este elemento opcional en este dominio.

  En la red de usuarios pueden existir **aplicaciones departamentales** con necesidad de interacción de servicios. Sólo se permite la implementación de webservices http(s)/SOAP y por tanto se descarta el despliegue de un ESB en este dominio.

  Siempre que sea posible se utilizarán esquemas XML para especificar los datos intercambiados por los servicios. Dichos esquemas XML estarán publicados en el Registro de Metadatos de Defensa.

- Los **sistemas de plataforma**, proporcionan diversos servicios comunes, pero no soportan funcionalmente el "negocio" de la organización. Inicialmente se requiere la integración con los sistemas siguientes:

  - PSSDEF. Provee servicios de seguridad tales como validación de certificados electrónicos, firma digital y cifrado. Se persigue centralizar en PSSDEF estas capacidades y cuando una integración las requiera a nivel de mensaje de servicio (por ejemplo en cabeceras WS-Security) deberán solicitarse a

PSSDEF. Por tanto, la Plataforma SOA debe estar totalmente integrada con PSSDEF mediante la invocación de sus servicios.

Actualmente, por motivos de rendimiento y dificultades técnicas se están realizando algunos de estas capacidades (cifrado a nivel de campo y firma digital en mensajes SOAP) sin invocar los servicios de PSSDEF.

- DICODEF. Directorio corporativo del MINISDEF que gestiona los usuarios, roles y autorizaciones. Diferentes componentes de la plataforma SOA utilizan DICODEF (y subconjuntos existentes en cada dominio) para autenticar y autorizar sistemas/aplicaciones para consumir servicios.

- E-mail. Cuando la operativa lo requiera proporciona envío de notificaciones mediante correo interpersonal. Ejemplo: alertas sobre mensajes entre aplicaciones mediados por el bus que no han podido entregarse por las razones que sean.

- **Bus de servicios en el dominio de uso oficial**. Permite la ejecución de los servicios de la plataforma proporcionando las siguientes características:
  1. Mediación de datos: Permite la interacción entre el consumidor y proveedor del servicio. Contiene capacidades para transformación y transporte de datos a nivel de aplicación. Adicionalmente, permite la orquestación de servicios.
  2. Servicios *core* no funcionales. Proporciona gestión de errores y de trazas comunes para todos los servicios, además de una gestión de logs uniforme.
  3. Capacidades de monitorización y vigilancia de cumplimiento de SLA's.

- **Agentes de políticas**. Aplican las políticas de seguridad que se requieran para permitir la invocación de un servicio. Se utilizan para validar, autorizar y enrutar los mensajes SOAP entre los distintos dominios. Permiten uniformizar las políticas aplicadas a los servicios de modo desacoplado de los sistemas proveedores de servicios y sirven como punto de ruptura de protocolo a nivel de mensaje de aplicación (http(s)/SOAP). Adicionalmente se limita el número de conexiones abiertas en los firewalls que separan los distintos dominios y monitoriza las llamadas entre servicios.

- **Entidades externas**. Representan aplicaciones externas al MINISDEF que, vía red SARA, Internet u otras redes de terceros, consumen servicios de aplicaciones del MINISDEF o exponen servicios que son consumidos por aplicaciones del MINISDEF. En los casos en que la aplicación externa actúa como proveedor del servicio, la

aplicación consumidora del MINISDEF debe ajustarse a los requisitos y mecanismos de seguridad de webservices requeridos por el servicio. Siempre que sea posible, esta adaptación y ajuste se debe implementar en el último elemento de conexión: el firewall XML. De modo similar, cuando la aplicación del MINISDEF actúa como proveedor, el firewall XML es el elemento que implementa la capa de seguridad de webservices en beneficio de la aplicación del MINISDEF proveedora del servicio.

- **Firewall XML en el nodo de interconexión**. Su función primordial es mediar los servicios publicados por el MINISDEF susceptibles de ser consumidos por organizaciones externas, garantizando la seguridad. Ofrece las siguientes capacidades básicas:

    1. Al igual que el Agente de Políticas implementa políticas de seguridad de webservices. En este caso concreto autentica y autoriza las aplicaciones ajenas al MINISDEF e implementa a su vez las capas de seguridad requeridas por los servicios externos que se invocan desde el MINISDEF.
    2. Protección frente a amenazas XML y ataques específicos a webservices XML.
    3. Capacidades básicas de bus: en caso necesario transforma y enruta los mensajes de la aplicación consumidora hacia el servicio final.
    4. En caso requerido implementa capacidades WAF (firewall de aplicaciones web).

- **Registro de Servicios.** Repositorio de información donde se publicarán los servicios disponibles en la red para poder ser reutilizados por otras aplicaciones.

    El Registro de Servicios publica tres tipos de servicios:

    1. Servicios internos expuestos por una aplicación del MINISDEF para ser consumido por otras aplicaciones del MINISDEF.
    2. Servicios públicos expuestos por el MINISDEF que son consumidos por aplicaciones de organizaciones externas al MINISDEF.
    3. Servicios proporcionados por una organización externa al MINISDEF consumidos por aplicaciones internas del MINISDEF.

    En todos los casos, los servicios (publicados en forma de WSDL) son mediados por la plataforma SOA para garantizar su seguridad y monitorización

    Existirá una única instancia del Registro de Servicios en el dominio de uso oficial de modo que sea accesible desde la red de usuarios y desde Internet o red SARA cumpliendo con los requisitos de seguridad aplicable a cualquier aplicación web desplegada en el dominio de uso oficial.

- **Registro de Metadatos**. Se publican y gestionan los esquemas de datos, metadatos y documentación asociada relacionada con la Plataforma, así como los esquemas pertenecientes al Modelo de Datos Común. Al igual que el registro de servicios será una aplicación del dominio de uso oficial de modo que se accesible desde la red de usuarios y desde Internet o red SARA cumpliendo con los requisitos de seguridad aplicables en este dominio.

3.3 *CASOS DE USO*

A continuación se detallan una serie de casos de uso genéricos para los que se ha diseñado este modelo de arquitectura y que deben ser soportados por la Plataforma SOA.

Estos casos de uso sirven como marco de referencia para el desarrollo de integraciones concretas. En cada integración en particular se deberán definir los mecanismos de seguridad y las necesidades de monitorización y mediación específicas de las aplicaciones a integrar. La Instrucción Técnica "Guía de requisitos de integración en la Plataforma SOA" proporcionará una guía para especificar los requisitos de una integración de aplicaciones en la Plataforma SOA tomando como referencia los casos de uso descritos a continuación.

1. Una aplicación en cualquier dominio de seguridad de la WAN 2.0 interacciona con una aplicación externa al MINISDEF.

2. Una aplicación en un dominio de seguridad de la WAN 2.0 interacciona con una aplicación de otro dominio.

3. Una aplicación interacciona con otra aplicación en su mismo dominio de seguridad.

Estos tres casos de uso se han generalizado, de modo que el origen o destino de la petición de servicio se refiere un dominio de seguridad de la WAN 2.0 (Uso público, Uso Oficial o Red de Usuarios)[2].

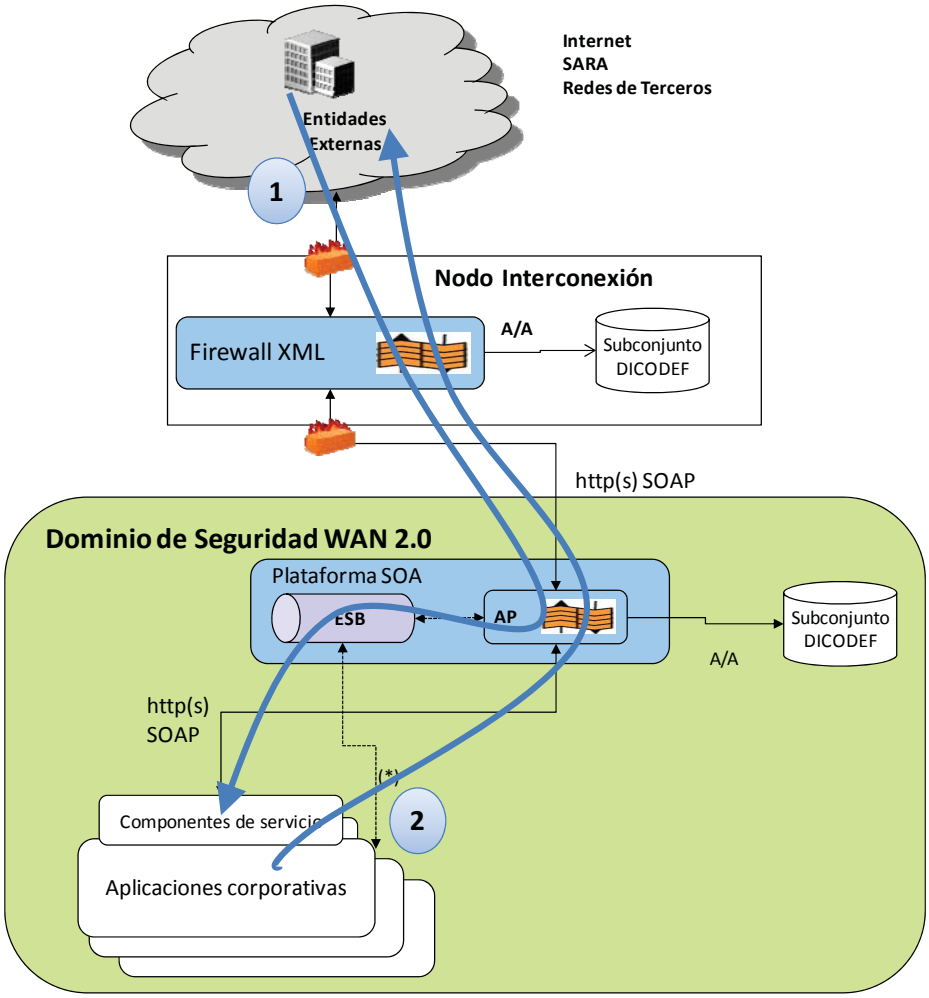
En los siguientes apartados se detalla cada uno de los casos mediante un diagrama, indicando los flujos de información y describiendo los posibles mecanismos de seguridad establecidos que deberán personalizarse en cada integración concreta.

[2] Por ahora no está definido el dominio de difusión limitada

3.3.1 *Aplicación en un dominio de seguridad de la WAN 2.0 interacciona con aplicación externa al MINISDEF*

Deben cumplirse requisitos de seguridad (autenticación, autorización, confidencialidad, evitar ataques DoS, etc.), y de monitorización (disponibilidad, auditoría de uso, cumplimiento de SLA's, etc.) de modo coordinado con el organismo proveedor o consumidor del servicio. Adicionalmente puede ser necesario proveer servicios de mediación de datos.

El siguiente diagrama indica los flujos de información a través de los elementos de la arquitectura:



En el diagrama, el término "Dominio de Seguridad WAN 2.0" representa cualquiera de los tres dominios actualmente definidos: Uso Oficial, Uso Público y Red de Usuarios. A continuación se detallan los pasos en cada flujo.

Flujo 1: Entidad externa consume un servicio de una aplicación en dominio de la WAN 2.0

Paso	Descripción
Aplicación Externa → XML firewall	Aplicación de entidad externa envía petición de servicio al XML firewall adjuntando credenciales.
XML firewall → Agente de políticas del dominio de WAN 2.0	XML firewall autentica la petición y autoriza consumir el servicio consultando subconjunto de DICODEF. En caso necesario progresa y/o mapea las credenciales para el agente de políticas del dominio de WAN 2.0.
Agente de políticas del dominio de WAN 2.0 → Bus de servicios (a)/ Sistema de dominio de WAN 2.0 (b)	El agente de políticas del dominio de WAN 2.0 valida credenciales y autoriza a consumir el servicio. El servicio final puede residir en el bus de servicios (a) (cuando se requieran servicios de mediación) o en el sistema proveedor (b).

Flujo 2: Aplicación en dominio WAN 2.0 consume un servicio de una entidad externa

Paso	Descripción
Aplicación corporativa → Agente de políticas de su dominio de WAN 2.0	Aplicación corporativa envía petición de servicio al agente de políticas de su dominio adjuntando credenciales.
Agente de políticas del dominio de WAN 2.0 → XML firewall	Agente de políticas valida las credenciales y autoriza consumir el servicio consultando DICODEF (o subconjunto en su dominio). En caso necesario progresa y/o mapea las credenciales para el firewall XML.
XML firewall → Servicio externo	El firewall XML valida la petición y, en caso necesario, añade los mecanismos de seguridad requeridos por la Entidad Externa para consumir el servicio destino. (*)

(*) En estos casos de integración en los que la entidad externa es el proveedor del servicio, lógicamente, en el último segmento prevalecerán los requisitos de integración y seguridad fijados por la entidad externa.

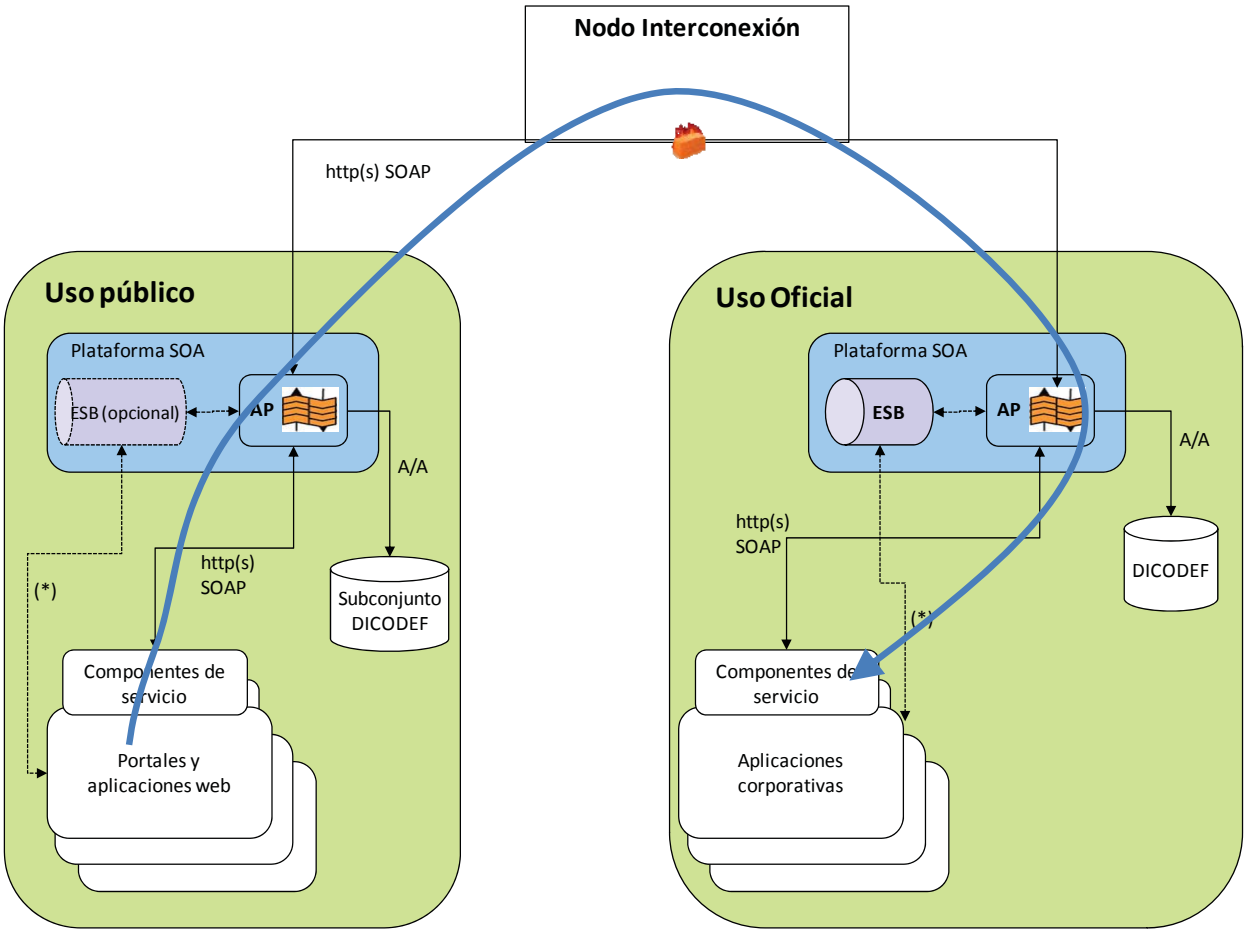
3.3.2 Aplicación en dominio de seguridad de la WAN 2.0 interacciona con aplicación en otro dominio de seguridad de la WAN 2.0

Este caso ocurre cuando un sistema de la WAN 2.0 expone un servicio para ser consumido por un sistema que reside en un dominio diferente.

Deben cumplirse requisitos de seguridad en la invocación (autenticación, autorización, confidencialidad,…), monitorización (disponibilidad, auditoría de uso, vigilancia de SLA's) como en el caso anterior. Sin embargo, las políticas de seguridad y monitorización podrán ser diferentes, habitualmente menos restrictivas.

El siguiente diagrama indica los flujos de información a través de los elementos de la arquitectura, tomando como ejemplo el consumo por parte de una aplicación/portal web de uso público de un servicio provisto por una aplicación corporativa que reside en el dominio oficial.

En este tipo de integraciones, dependiendo de los dominios de seguridad implicados y el sentido del flujo de información, las medidas de seguridad aplicadas en cada segmento deberán personalizarse de entre las soportadas por la plataforma.



En el dibujo se observa que el flujo pasa por el nodo de interconexión. Desde el punto de vista de la plataforma SOA el paso por el nodo de interconexión es transparente en cuanto que no hay ningún elemento de la plataforma SOA implicado. Sin embargo, siguiendo los principios de la WAN 2.0, el flujo de protocolo de transporte a través del nodo de interconexión estará asegurado mediante los dispositivos de seguridad de red adecuados (tales como firewalls, IDS, etc.).

La tabla siguiente contiene los pasos o interacciones entre los elementos de la arquitectura implicados:

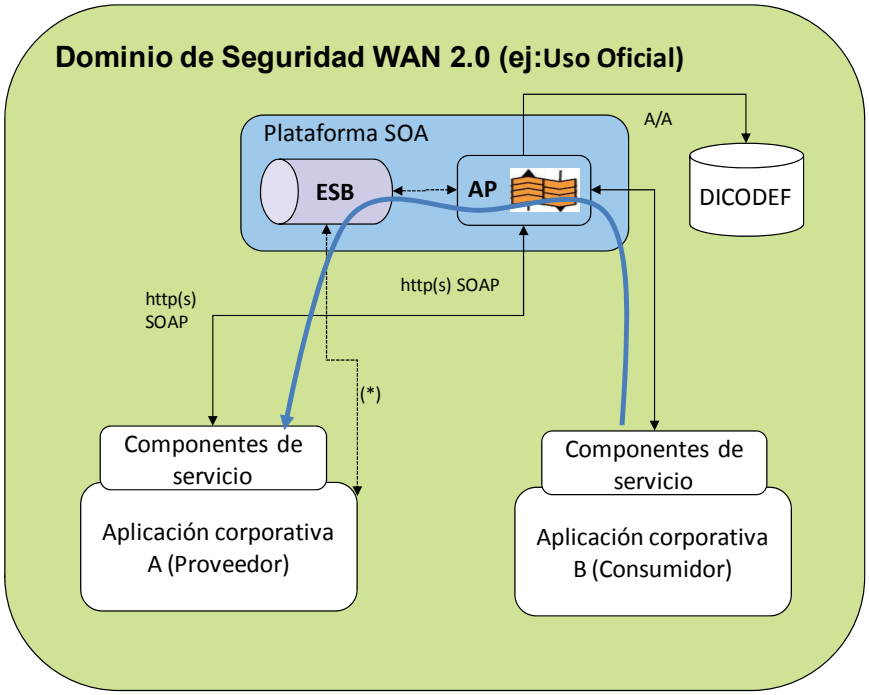
Paso	Descripción
Portal/Aplicación web de Uso Público → Agente de políticas de su dominio	Portal/Aplicación web envía petición de servicio al agente de políticas de su dominio adjuntando credenciales.
Agente de políticas del dominio Uso Público → Agente de Políticas de Uso Oficial	Agente de políticas de Uso Público valida las credenciales y autoriza consumir el servicio consultando subconjunto DICODEF. En caso necesario progresa y/o mapea las credenciales para el Agente de Políticas de Uso Oficial.
Agente de Políticas de Uso Oficial → Bus de servicios (1) / Aplicación corporativa en Uso Oficial (2)	El agente de políticas de Uso Oficial valida la petición para consumir el servicio. El servicio final puede residir en el bus de servicios (1) (cuando se requieran servicios de mediación) o en la aplicación corporativa mediante su componente de servicio (2).

3.3.3 Aplicación interacciona con otra aplicación en su mismo dominio de seguridad

Este caso ocurre cuando un sistema de la WAN 2.0 expone un servicio para ser consumido por un sistema que reside en su mismo dominio. Sin embargo, puede ocurrir que las tecnologías en las que están implementados los sistemas sean diferentes o que los sistemas estén bajo ámbitos de responsabilidad distintos.

Los requisitos de seguridad y monitorización dependerán de la criticidad y protección de información requerida por el servicio. Pueden existir servicios que no devuelvan datos confidenciales (por ejemplo: realizan algún tipo de cálculo especializado o devuelven datos de una tabla maestra que no tienen clasificación de seguridad), estos servicios requerirán una seguridad y monitorización mínima. En el otro extremo existirán servicios críticos y/o que devuelvan datos que requieran medidas de protección de datos (por ejemplo: servicios que devuelven o actualizan datos de carácter personal o información clasificada).

El siguiente diagrama indica el flujo de información a través de los elementos de la arquitectura:

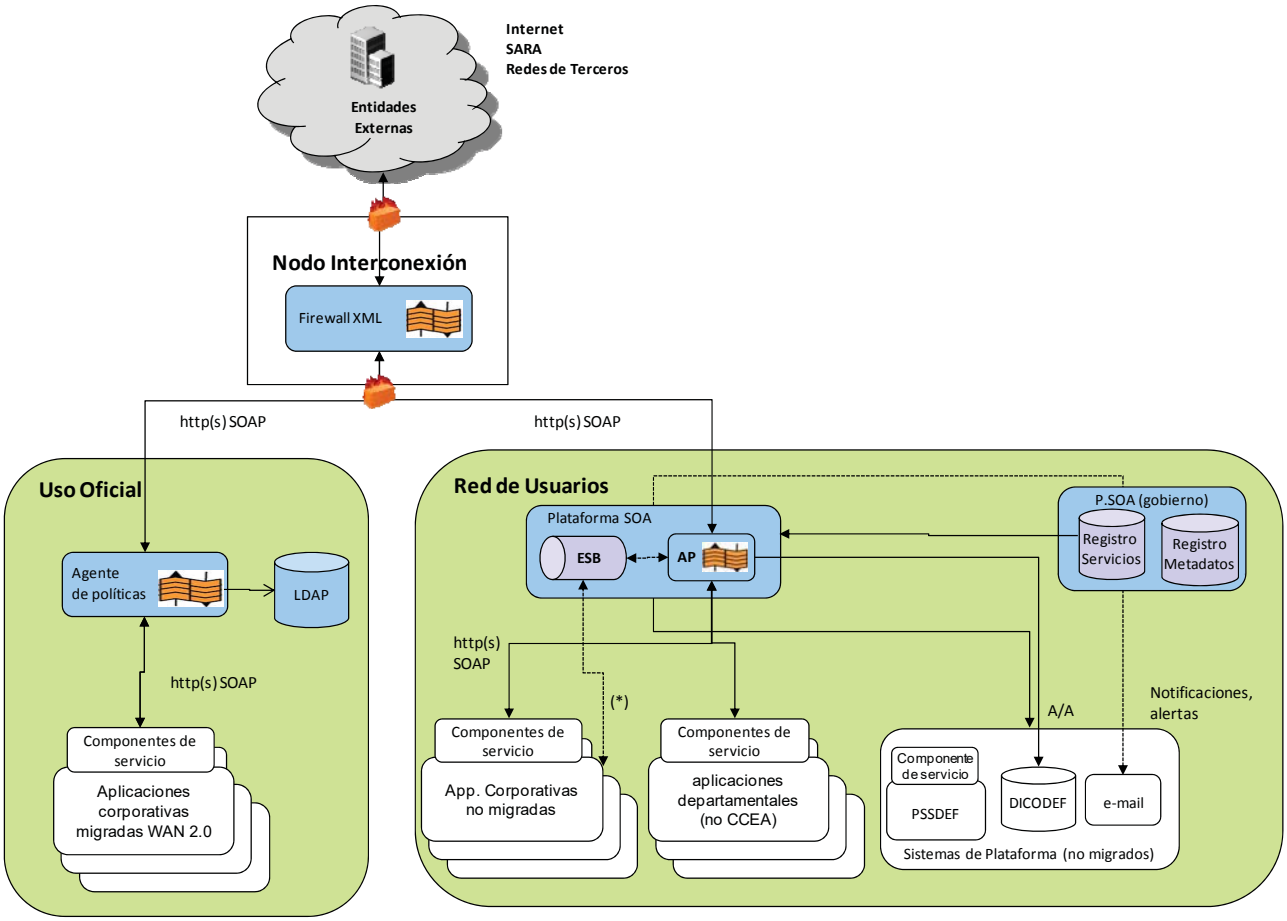


La tabla siguiente contiene los pasos o interacciones entre los elementos de la arquitectura implicados:

Paso	Descripción
Aplicación corporativa B en Uso Oficial → Agente de políticas del dominio	Aplicación corporativa B envía petición de servicio al agente de políticas de su dominio adjuntando credenciales.
Agente de Políticas de Uso Oficial → Bus de servicios (1) / Aplicación corporativa A en Uso Oficial (2)	El agente de políticas valida las credenciales y autoriza a consumir el servicio consultando DICODEF. El servicio final puede residir en el bus de servicios (1) (cuando se requieran servicios de mediación) o en la aplicación corporativa A mediante su componente de servicio (2).

3.4 *Arquitectura actual de la Plataforma SOA*

En la fecha de edición de este documento, la WANPG se encuentra en una situación transitoria de migración desde el modelo anterior a la nueva WAN 2.0. Además, la plataforma SOA fue desplegada en producción conforme al modelo SOA 1.0. El siguiente diagrama refleja la arquitectura actual de la Plataforma SOA en este escenario transitorio.



Las diferencias respecto al escenario de plataforma SOA para WAN 2.0 se concretan en:

- El dominio de Uso Público no tiene desplegado ningún elemento de la plataforma SOA. Cuando haya alguna integración que lo requiera se desplegará un agente de políticas.
- El firewall XML valida credenciales contra un repositorio en el propio dispositivo, no contra un directorio ldap (subconjunto DICODEF).
- El agente de políticas de Uso Oficial valida las credenciales contra un directorio ldap ad-hoc que deberá sustituirse por un subconjunto DICODEF.

- El bus de servicios está desplegado en la red de usuarios donde residen todavía la mayoría de las aplicaciones corporativas. Se moverá a Uso Oficial a medida que los sistemas que media se desplacen a Uso Oficial.

Esta arquitectura da respuesta a las integraciones en producción y actualmente en desarrollo, que pueden generalizarse en los siguientes casos de uso:

1. Aplicaciones corporativas en el dominio de Uso Oficial interaccionan con servicios web de aplicaciones en la red de usuarios (intranet en el modelo SOA 1.0) o con servicios de entidades externas vía nodo de interconexión.

2. Aplicaciones de entidades externas interaccionan con servicios en la red de usuarios vía nodo de interconexión.

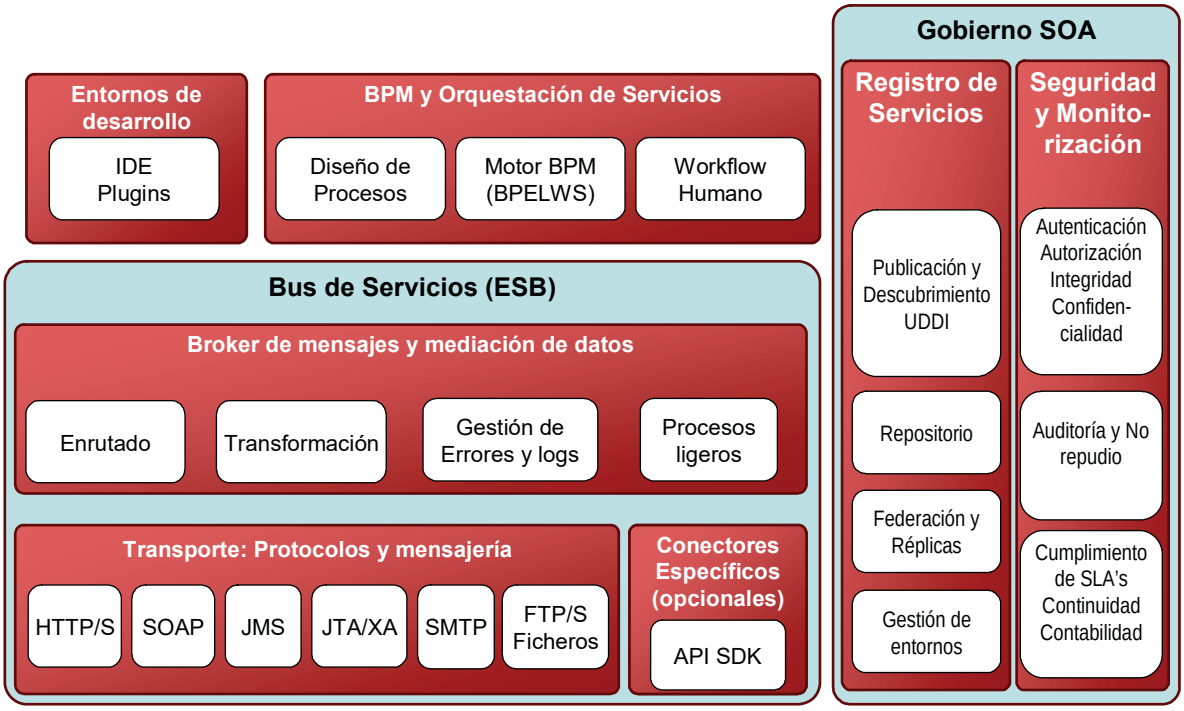
3. Una aplicación interacciona con servicios de otra aplicación dentro de su mismo dominio.

Los flujos de información en estos casos de uso son prácticamente idénticos a los expuestos anteriormente para WAN 2.0 y los elementos de la plataforma SOA desplegados son los mismos, con la salvedad de estar desplegados en dominios diferentes. Por tanto, los principios de diseño y mecanismos de seguridad establecidos para WAN 2.0 son de aplicación en la situación actual.

Actualmente hay varias integraciones en producción y en desarrollo que se corresponden con los tres casos de uso anteriores.

3.5 *ARQUITECTURA LÓGICA DE COMPONENTES*

En el diagrama de bloques siguiente se resume la arquitectura lógica de componentes definida para soportar la SOA corporativa del MINISDEF.



A continuación se describen los componentes de la arquitectura.

3.5.1 Entornos de desarrollo

Son necesarios entornos de desarrollo específicos para desarrollar servicios y procesos de integración sobre la plataforma. Las capacidades que debe cumplir el IDE (entorno de desarrollo integrado) son:

- Una interfaz de usuario uniforme para todas las actividades de desarrollo de la plataforma SOA como por ejemplo: desarrollo de servicios, procesos de integración, procesos de negocio, especificaciones de transformaciones, conversiones y mapeo de datos, gestión de esquemas y otros recursos XML, etc.

- Interfaz gráfico para gestionar objetos de programación permitiendo una programación visual y declarativa.

- Capacidades de depuración de código integradas con la plataforma

- Capacidades de integración con herramientas de control de versiones

- Integración con el registro de servicios de modo que un desarrollador pueda consultar los servicios disponibles en el registro de modo integrado desde el propio entorno de desarrollo.

- Ampliación de capacidades mediante plugins, conservando el mismo interfaz de usuario independientemente del componente a desarrollar.

- Capacidad de integración con el entorno de despliegue.

3.5.2 Bus de servicio (ESB)

El ESB (Bus de Servicios Corporativo) es dónde se ejecutan los servicios de integración, en lugar de estar codificados en la lógica de la aplicación, proporcionando una clara separación entre la lógica de negocio y la lógica de integración. La conectividad, la transformación de datos y el enrutado de los mensajes se considera parte de la lógica de integración.

Un ESB es un conjunto integrado de componentes software que ofrecen servicios de integración entre aplicaciones basados en estándares y tecnologías de integración comúnmente aceptadas tales como webservices (SOAP, WSDL, UDDI, WS-*) XML (XSLT, XPath, XQuery), Java (JMS, JCA, JTA, JBI), etc.

Las capacidades básicas que ofrece el ESB son:

- MOM (Middleware orientado a mensajes): Transporte fiable y robusto de mensajes XML o, si fuera necesario, cualquier otro contenido. Fiabilidad de extremo a extremo (de aplicación a aplicación) en cuanto a entrega, transaccionalidad, prioridad, etc. Las aplicaciones pueden acceder a estos servicios directamente mediante las API's disponibles o a través de servicios de mediación del ESB que adapta los protocolos de transporte que "hable" la aplicación.

El estándar seleccionado para MOM es JMS

- Servicios web: Soporte para el desarrollo rápido de servicios web estándar.

Recubrimiento de servicios no estándar ofrecidos por las aplicaciones. Invocación de servicios web disponibles en la red incluyéndolos dentro de una orquestación de servicios para componer un proceso de negocio o simplemente de integración entre sistemas heterogéneos.

Los estándares seleccionados para servicios web son SOAP sobre http(s) para invocar a servicios y WSDL para describir los servicios y permitir su publicación en el registro de servicios.

Las guías técnicas que posteriormente desarrollen esta instrucción técnica establecerán los criterios detallados para que un servicio web se considere estándar.

- Enrutado de mensajes basado en el contenido del propio mensaje o en el contexto y/o reglas de negocio del proceso de integración.

- Transformación de datos basada en tecnología XML: Los formatos de mensajes consumidos por los distintos sistemas pueden no ser compatibles, pues se adaptan a las necesidades específicas de cada sistema. El ESB ofrece servicios de transformación de formatos, conversión y mapeo de datos basados en XML.

Los estándares seleccionados para transformación de datos son XSLT, XPath y XQuery.

- Invocación de servicios dirigida por eventos (soporte para EDA). Determinados servicios sólo deben ser invocados cuando se produce un determinado evento. En estos casos el ESB debe ser capaz de gestionar la recepción del evento en forma de mensaje y enrutarlo a los servicios adecuados. Estas capacidades del ESB son las que soportan el mayor grado de desacoplamiento de sistemas en esquemas de tipo Publish/Subscribe: un sistema genera eventos en un *topic* y los sistemas interesados se suscriben a dichos eventos. Cuando se produce un evento el ESB es capaz de invocar a los servicios suscritos y pasarles la información asociada al evento en forma de mensaje.

El estándar seleccionado para EDA es JMS

- Soporte de múltiples protocolos para conectar aplicaciones. El ESB debe soportar nativamente los siguientes protocolos para interactuar con sistemas proveedores y consumidores de servicios:

    - XML sobre http(s) y JMS
    - SOAP sobre http(s) y JMS
    - JDBC para acceso a bases de datos relacionales y procedimientos almacenados de base de datos
    - JTA/XA para soporte de transacciones distribuidas
    - SMTP (integración con e-mail)
    - FTP y acceso directo a ficheros para sistemas que no soporten otras alternativas de integración.
    - Adicionalmente podrán desarrollarse (o adquirirse) conectores específicos mediante un SDK disponible en el ESB para ampliar el soporte de protocolos de conectividad de aplicaciones.

- Facilidades para auditoría, trazabilidad y gestión de errores. Las invocaciones de servicios a través del ESB quedarán registradas de modo independiente a los sistemas proveedores y consumidores, descargando de esta capacidad a los sistemas específicos y permitiendo de este modo la auditoría y trazabilidad de las dependencias entre servicios disponibles en la red. Un modelo de gestión de errores unificado en los procesos de integración permite recuperarse más rápida y fácilmente de los errores que se produzcan.

Servicios básicos comunes del ESB

Para estandarizar las capacidades ofrecidas por el bus de servicios, la plataforma de interoperabilidad debe ofrecer unos servicios básicos comunes que se describen a continuación.

- Framework de desarrollo sobre el bus de servicios corporativo que permite uniformizar el entorno de desarrollo, reutilizar componentes y facilitar el despliegue de los servicios.
- Gestión central de errores: unifica el tramiento de los errores durante el desarrollo de procesos de integración en el bus de modo que el tratamiento de errores en producción sea más eficiente.
- Gestión uniforme de trazas y logs de los procesos de integración durante el desarrollo de modo que facilite la contabilidad y auditoría de los servicios en el entorno de producción.

3.5.3 Gobierno SOA

Gobierno SOA engloba los componentes de la arquitectura relacionados con la gestión del ciclo de vida de los servicios, y la seguridad y monitorización de los servicios. Estos componentes son:

- Registro de servicios
- Registro de Metadatos
- Seguridad y monitorización

3.5.3.1 *Registro de servicios*

Los distintos servicios disponibles, tanto servicios web SOAP descritos con un WSDL como cualquier otro servicio que proporcione información al que se pueda subscribir un sistema consumidor (p.ej. un *topic* JMS en el que se publican un determinado tipo de mensajes), deben ser publicados en el Registro de Servicios, para que los sistemas consumidores autorizados puedan descubrirlos e invocarlos.

El Registro de Servicios debe cumplir las siguientes características:

- Neutralidad respecto a las plataformas (Ej. .NET, J2EE) en las que estén implementados los sistemas proveedores y consumidores de servicios.

- Creación de taxonomías que permitan la clasificación flexible de los servicios publicados facilitando su gestión y descubrimiento.

- Servicios de suscripción y notificación sobre los servicios publicados.

- Capacidades de gestión de diferentes entornos de ejecución tales como desarrollo, pruebas y explotación, soportando el versionado de los servicios. El registro debe proporcionar mecanismos para soportar procedimientos de aprobación para gestionar el ciclo de vida de los servicios.

- Gestión de diferentes perfiles de usuarios y control de acceso al Registro de Servicios.

El estándar seleccionado para el Registro de Servicios es UDDI

3.5.3.2 *Registro de Metadatos de Defensa (RMD)*

El RMD mantiene y publica los esquemas, definiciones de estructuras de datos y documentación asociada utilizadas en las interacciones de servicios gestionadas por la Plataforma SOA, permitiendo su reutilización por los equipos de desarrollo.

El Registro de Metadatos soportará todos los estándares de definición de estructuras XML que se requieran como XML-schema (.xsd), WSDL, OWL, etc.

Adicionalmente contendrá el catálogo de trazas y errores del ESB, y los documentos técnicos asociados necesarios para el desarrollo de servicios (Guías de diseño, de desarrollo, de pruebas, etc).

Los estándares de referencia seleccionados para el Registro de Metadatos son ISO-11179 y ebXML RS/RIM.

3.5.3.3 *Seguridad y monitorización*

Los servicios de seguridad y monitorización están englobados dentro del Gobierno SOA, aunque serán implementados en tiempo de ejecución por elementos de la plataforma SOA. Deben garantizar que las interacciones de servicios tanto síncronas (basadas en http y SOAP) como asíncronas (basadas en JMS) sean seguras y monitorizadas.

Los servicios de seguridad se proveerán de modo uniforme e independiente de las aplicaciones, minimizando en lo posible el impacto de la implementación de la seguridad en el desarrollo de los servicios.

Las capacidades de gestión y monitorización de la "red de servicios SOA" deberán integrarse con el sistema de gestión de redes y sistemas del CCEA.

Los elementos de plataforma SOA que requieran el uso de certificados digitales utilizarán los proporcionados por la PKI de Defensa.

Los servicios de validación de certificados y firma electrónica deberán estar integrados con PSSDEF.

Características y capacidades de los servicios de seguridad y monitorización:

- Aplicación de políticas de seguridad a cada servicio, garantizando los siguientes aspectos cuando sea necesario en función de los requisitos de seguridad específicos de cada servicio:
    - Acceso al consumo de un servicio sólo por parte de los consumidores autorizados.
    - Autenticación extremo a extremo del sistema proveedor y consumidor del servicio.
    - Integridad y Confidencialidad extremo a extremo de la información transmitida en la invocación de un servicio.
    - Opcionalmente "No repudio", cuando se solicite este servicio. Permiten a consumidor y proveedor del servicio tener garantía de que la invocación del servicio se ha producido.
    - Propagación de la identidad del usuario final cuando se requiera.
- Generación de eventos producidos por incidentes de seguridad en los servicios (ej: intentos de invocación no autorizados, fallos de autenticación, etc.) integrados con COSDEF.
- Aplicación de políticas de seguridad a nivel de protocolos de transporte (Ej.: https y SSL).
- Monitorización técnica y operativa integrada con COSDEF: que garantice el tener conocimiento en todo momento de la disponibilidad de los servicios.
- Monitorización funcional: que garantice que se está cumpliendo con los SLA requeridos y no se ve afectada ninguna funcionalidad crítica.

- Contabilidad y estadísticas sobre la disponibilidad de los servicios en explotación.

- Auditoría del consumo de servicios, facilitando las consultas por diversos criterios como por ejemplo: sistema consumidor o proveedor, tipo de servicio, dominio, etc.

- Intermediación de servicios para garantizar restricciones de servicio previamente establecidas (como por ejemplo número máximo de invocaciones por periodo, máximo de invocaciones simultáneas, etc.)

- Generación de eventos producidos por incidentes en servicios (ej: no disponibilidad de un servicio crítico, exceder límites establecidos sobre un servicio, etc.).

3.5.4 BPM y orquestación de servicios

La capacidad de BPM (Gestión de procesos de negocio) y orquestación de servicios se considera un componente fundamental en una arquitectura SOA moderna. Estas son las capacidades que ofrece este componente:

- Diseño de procesos de negocio mediante un lenguaje de modelización estándar en una herramienta software con interfaz de usuario gráfico. Esta capacidad deberá estar integrada mediante un plugin en el IDE estándar, y acceder a las especificaciones de los servicios publicadas en el Registro de Servicios de modo que se puedan reutilizar en el diseño de los procesos de negocio.

El estándar de referencia para diseño BPM es el lenguaje UML

- Motor BPM. Es el componente que ejecuta los procesos de negocio. Debe estar integrado con el ESB y estar basado en un lenguaje estándar basado en reglas tal como BPEL4WS (Business Process Execution Language for web services).

- Workflow humano. Este componente permite que los procesos de negocio incorporen durante su ejecución la interacción de personas (por ejemplo: un determinado proceso de negocio implica que una persona del visto bueno a un informe). Este componente debe estar perfectamente integrado con el motor BPM además de con otros servicios de plataforma como los servicios de portal web y de

herramientas colaborativas.

El BAM (Business Activity Monitoring) consiste en monitorizar los *procesos de negocio* contenidos en la SOA con respecto a unos ciertos KPI *de negocio* (indicadores clave de rendimiento) previamente establecidos.

4 ROLES Y RESPONSABILIDADES

La definición del modelo de arquitectura orientada a servicios del MINISDEF y la implantación de la Plataforma de Interoperabilidad y Gobierno SOA requiere establecer una serie de roles y responsabilidades dentro de la organización CIS del Ministerio de Defensa. Los roles y responsabilidades que se describen a continuación corresponden únicamente al ámbito de la plataforma de Gobierno SOA.

A continuación figura la relación de roles y responsabilidades ligadas al modelo de arquitectura SOA, una vez desplegada la Plataforma en los diferentes entornos. También se identifican aquellos organismos CIS del MINISDEF que deberán asumir tales roles:

Rol	Organismo	Responsabilidades
Autoridad Operacional	**DIGENIN/SDGTIC**	• Coordinación con organismos que provean o consuman servicios del MINISDEF y establecimiento de los SLA's que se precisen.

Rol	Organismo	Responsabilidades
Centro de competencia SOA	**SDGTIC – Área de Tecnología**	• Establecer el marco metodológico de SOA. • Diseñar y mantener el Modelo de Arquitectura SOA que permita la evolución progresiva de la arquitectura de sistemas de propósito general hacia SOA. • Establecer un marco de coordinación y cooperación con los diferentes organismos CIS de Defensa para la mejora continua y adecuada evolución del Modelo de Arquitectura SOA. • Definición y diseño de la Plataforma SOA y de los servicios básicos comunes que residen en el bus corporativo. • Estudiar y aprobar modificaciones sobre los elementos de arquitectura de la plataforma SOA teniendo en cuenta los requisitos de los sistemas afectados • Elaboración, difusión y control de guías, normas y procedimientos referidos a buenas prácticas SOA, estándares recomendados u obligatorios para el diseño, desarrollo y despliegue de servicios en la plataforma SOA. Estos documentos serán consensuados con CCOMSI • Catalogación mediante taxonomías u otros mecanismos de los servicios publicados en el Registro de Servicios. • Gestión del procedimiento de aprobación de contenidos del Registro de Metadatos. • Definir la arquitectura en relación con los elementos de la Plataforma SOA en cada caso concreto de integración de sistemas. • Competencias de administración del entorno de desarrollo para el desempeño de sus funciones.

Rol	Organismo	Responsabilidades
Autoridad de Explotación	**SDGTIC - CCEA**	• Administrar y gestionar la infraestructura asociada a la Plataforma SOA en los entornos de preproducción y producción. Esta responsabilidad estará bajo una autoridad única designada por CCEA, independientemente del despliegue requerido en los distintos dominios de red en los que se precise (nodo de interconexión, uso oficial, etc.). • Monitorización técnica de los servicios y vigilancia de los SLA's. • Soporte de incidencias. • Gestión de la Seguridad. • Validar la arquitectura de despliegue de la plataforma sobre los recursos hardware y de software de base disponibles en los entornos de red requeridos. • Administrar y gestionar los entornos de preproducción y producción que se requieran. • Definición, difusión y control del conjunto de guías y normativas de administración y explotación de obligado cumplimiento. • Contribuir a la mejora de las normas, guías y procedimientos propios de la Plataforma SOA bajo responsabilidad del centro de competencia SOA. • Alojamiento y administración del hardware y software de base del entorno de desarrollo.

Rol	Organismo	Responsabilidades
Diseño, desarrollo y mantenimiento de servicios y procesos funcionales	**SDGTIC – CCOMSI** **y en general** **Cualquier UCO que desarrolle y/o mantenga sistemas de información en la WANPG**	• Identificación de servicios a desarrollar y su priorización. • Diseño, desarrollo y mantenimiento de los servicios dentro de su área funcional. • Diseño, desarrollo y mantenimiento de los servicios/procesos compuestos que afectan a distintas áreas funcionales. • Publicar y mantener contenido en el Registro de Metadatos referente a: esquemas XML, definiciones de estructuras de datos y documentación asociada utilizadas en los interfaces de servicios. • Publicación de las descripciones de servicios en el Registro de Servicios • Control de versiones de los servicios sobre el Registro de Servicios • Promoción de servicios desde desarrollo a producción. • Definición y ejecución de pruebas de los servicios y procesos desarrollados. • En cada caso concreto de integración de aplicaciones, definir y diseñar la integración desde un punto de vista funcional. **Responsabilidades exclusivas de CCOMSI** • Diseño, desarrollo, mantenimiento y uso de los servicios básicos comunes en el bus de servicios corporativo • Contribuir a la mejora de las normas, guías y procedimientos propios de la Plataforma SOA bajo responsabilidad del centro de competencia SOA. • Competencias de administración del entorno de desarrollo para el desempeño de sus funciones.
Seguridad de los servicios	**SDGTIC – Área de Seguridad**	• Validar los aspectos de de seguridad de la arquitectura de referencia y técnica de sistemas de propósito general hacia SOA. • Validar que la arquitectura de cada caso de integración concreto cumple con la normativa de seguridad y recomendar alternativas encaso necesario. • Gestión, generación y monitorización de los eventos de seguridad integrados con el Centro de Operaciones de Seguridad (COSDEF) • Supervisar que se cumplen los requisitos de seguridad establecidos

El desarrollo y explotación de los distintos elementos que componen este modelo supone la necesidad de asignar un número suficiente de recursos técnicos y humanos, para que la gradual evolución a SOA de la WANPG se realice de modo eficaz y con la calidad requerida.

API	Application Programming Interface
AGE	Administración General del Estado
ATU	Arquitectura Técnica Unificada del MINISDEF.
BAM	Business Activity Monitoring.
BPEL4WS	Business Process Execution Language for Web Services
BPM	Business Process Management
CCOMSI	Centro Corporativo de Obtención y Mantenimiento de Sistemas de Información
CORBA	Common Object Request Broker Architecture
COSDEF	Centro de Operaciones de Seguridad de Defensa
COTS	Commercial Off the shell
CIS	Communications and Information Systems
DICODEF	Directorio Corporativo de Defensa
DTD	Document Type Definition
ebXML	Electronic Business XML
ESB	Enterprise Service Bus (Bus de Servicios Corporativo)
FTP	File Transfer Protocol
HTTP	Hypertext Transmisión Protocol
IDE	Entorno de Desarrollo Integrado
IGECIS	Inspección General CIS

INVIFAS	Instituto para la Vivienda de las Fuerzas Armadas
ISFAS	Instituto Social de las Fuerzas Armadas
ISO	International Organization for Standardization
J2EE	Java 2 Enterprise Edition
JBI	Java Business Integration
JCA	J2EE Connector Architecture
JDBC	Java Database Connectivity
JMS	Java Messaging Services
JTA	Java Transaction API
MINISDEF	Ministerio de Defensa
MOM	Message Oriented Middleware
NNEC	NATO Network Enabled Capability
PG	Propósito General
PKI	Public Key Infrastructure
PSSDEF	Plataforma de Seguridad de Defensa
RD	Real Decreto
RMD	Registro de Metadatos de Defensa
SARA	Sistema de Aplicaciones y Redes para la Administraciones
SDK	Software Development Kit
SDGTIC	Subdirección General de Tecnologías de Información y Comunicaciones
SLA	Service Level Agreement
SMTP	Simple Mail Transfer Protocol
SQL	Structured Query Language

SOA	Arquitectura Orientada a Servicios
SOAP	Simple Object Access Protocol (XML protocol)
UML	Unified Modeling Language
WAN	Wide Area Network
WS	Web Service
WSDL	Web Service Description Lenguage
XML	eXtensible Markup Language
XSLT	Extensible Style Language Transformation