Principios de Estandarización y Diseño del contrato de servicios del BP.

Tabla de contenido

[1 Introduccion 3](#_Toc18940207)

[2 Perfil del contrato de Servicio. 3](#_Toc18940208)

[3 Tipos de Estandarizacion de Contratos de Servicio. 3](#_Toc18940209)

[3.1 Estandarizacion de las expresiones funcionales del Servicio 4](#_Toc18940210)

[3.2 Estandarizacion de la representacion de Datos del Servicio. 4](#_Toc18940211)

[3.3 Estandarizacion de las politicas del Servicio. 7](#_Toc18940212)

[3.3.1 Vocabularios de afirmación de propiedad 7](#_Toc18940213)

[3.3.2 Parámetros y Políticas Anidadas 8](#_Toc18940214)

[3.3.3 Modularidad y Centralización de POliticas 8](#_Toc18940215)

[3.3.4 Estándar Estructural 8](#_Toc18940216)

[4 Diseño del contrato de Servicios. 9](#_Toc18940217)

[4.1 Estandarización de representación de datos y Evitar La transformación 9](#_Toc18940218)

[4.2 Estandarización de la Granuralidad 10](#_Toc18940219)

[4.3 Consideraciones entre el uso de contratos Estandarización y contratos de tareas. 11](#_Toc18940220)

# Introduccion

Los contratos de servicios son un punto focal del diseño del servicio porque son fundamentales para casi todo lo que hacen los servicios.

Debido a que los contratos también son componentes arquitectónicos centrales de las soluciones orientadas a servicios (SOA), es fundamental que los principios de Diseño de Contratos sean aplicados cuidadosamente ya que esto influyen directamente en cómo se posicionan, diseñan y finalmente se utilizan los Servicios.

El papel fundamental del principio de Diseño de contratos es garantizar la expresión coherente de las capacidades del servicio y el propósito general para el cual fue construido

# Perfil del contrato de Servicio.

En el establecimiento de la gobernanza del proyecto de SOA del BP se están implementado la construcción de perfiles de los contratos de servicios para garantizar la estandarización y las buenas practicas.

El perfil de contrato de servicios debe de expresar el propósito y las capacidades del servicio de una manera consistente además debe de establecer los términos de uso, proporcionando restricciones técnicas y requisitos.

|  |  |
| --- | --- |
| Principios de Diseño del Perfil de Contrato del WS | |
| Definición Corta | Texto Explicativo |
| Definición Larga | Texto Explicativo |
| Metas | Texto Explicativo |
| Características de Diseño | Texto Explicativo |
| Requerimientos de Implementación | Texto Explicativo |
| Regionalización o Influencia del WS | Texto Explicativo |

# Tipos de Estandarizacion de Contratos de Servicio.

Como parte de una transición hacia una arquitectura orientada a servicios, necesitamos armar un inventario de servicios, muchos de los cuales se habrán diseñado deliberadamente como recursos reutilizables.

En última instancia, nuestro meta es alcanzar un estado en el que podamos cumplir con los nuevos requisitos de automatización de negocios al minimizar la cantidad de esfuerzo de desarrollo personalizado y, en su lugar, reutilizar cada vez más los servicios disponibles.

Esto suena como un plan lógico en teoría, pero llevarlo a cabo en la vida real requiere una atención seria al diseño de cada contrato de servicio, ya que la estandarización se aplica en varios niveles.

## Estandarizacion de las expresiones funcionales del Servicio

Cuando los servicios se vuelven comunes dentro de una empresa, es beneficioso que cada servicio exprese los detalles de sus respectivos dominios funcionales utilizando las mismas convenciones.

Específicamente, la aplicación de convenciones de expresión funcional a los servicios incrementa la facilidad con la que eventualmente los humanos (y las herramientas) interpretan el diseño y, en última instancia, da como resultado una empresa orientada al servicio que es intuitiva y fácil de navegar.

Debido a que se hace un esfuerzo para aclarar constantemente el significado de cada servicio, las oportunidades de reutilización para aquellos con un contexto agnóstico se identifican más fácilmente.

Además, debido a que se reducen las posibilidades de malas interpretaciones, se mitiga el riesgo de que los equipos del proyecto creen inadvertidamente nuevos servicios con lógica conflictiva o redundante.



Ejemplo 1 Las definiciones de mensaje en el Listado # 1 son incómodas y hacen referencia a nombres de elementos de esquema XML que parecen muy específicos para esta operación. El Listado # 2 representa una versión de las mismas definiciones de mensaje a las que se aplicaron los expresiones funcionales estándares, lo que resulta en el uso de convenciones de nomenclatura más genéricas y referencias de tipos de datos genéricos.

## Estandarizacion de la representacion de Datos del Servicio.

La descripción de la interfaz técnica que forma la base de cualquier contrato de servicio casi siempre incluirá una definición formal de los datos de entrada y / o salida requeridos por cada capacidad de servicio.

El detalle de esta definición generalmente incluye el tipo de datos.

Cuando se trabaja con servicios web y esquemas XML asociados, los modelos de datos generalmente se componen de tipos de datos complejos que organizan piezas de información relacionadas en una estructura formal.

Esta forma de estandarización aboga por mantener esquemas y tipos de datos asociados para conjuntos específicos de datos en alineación entre servicios en la medida de lo posible.

Es fácil crear esquemas para servicios específicos que se adaptan a la forma en que estos servicios necesitan representar datos (siguiente Figura). Puede conducir a diseños de esquemas muy eficientes que se optimizan para representar solo datos que son relevantes para la funcionalidad encapsulada por el servicio.

Sin embargo, este enfoque puede causar muchos problemas.



Ejemplo 2 Se debe de evitar caer en la practica de crear un conjunto de definiciones WSDL para el que se ha personalizado un conjunto correspondiente de esquemas XML.

Esto tiene la apariencia de una arquitectura de contrato muy limpia, pero puede socavar seriamente el potencial de interoperabilidad de una SOA.

Uno de los objetivos clave de la informática orientada a servicios es permitir el ensamblaje ágil e incluso ad-hoc de composiciones de servicios. Es a través de las composiciones de servicios que ejercemos la mayoría de las oportunidades de reutilización que se nos presentan.

Si dos capacidades de servicio dentro de una composición representan el mismo tipo de datos usando diferentes representaciones (modelos de datos, esquemas), entonces su relación se basa en una representación de datos no estandarizada.

Este escenario lleva a la necesidad de transformación de datos.

Aunque la tecnología de transformación adecuada está disponible para superar la disparidad del modelo de datos, no es deseable usarla.

De hecho, gran parte del esfuerzo de estandarización en torno al diseño de contratos de servicios se centra en la "evitación de la transformación".

En el mundo de los servicios web, los esquemas se deben de diseñar e implementar por separado de las capacidades del servicio (operaciones) que los utilizan para representar la estructura y la tipificación del contenido del mensaje.

Como resultado, se puede establecer y estandarizar una arquitectura de representación de datos de forma algo independiente de la capa de servicio principal.

Esto permite la aplicación de un patrón de diseño conocido como Centralización de Esquemas, que aboga por la definición de un esquema "oficial" para cada conjunto de información.

Los contratos de servicios web pueden compartir estos esquemas centralizados.

El resultado es una representación de datos de servicio estandarizada.

La aplicación de este patrón también fomenta la creación de esquemas de entidad, esquemas que representan datos asociados con una entidad de negocio en particular, como se ilustra en la siguiente figura



Ejemplo 3 Las definiciones WSDL que comparten esquemas XML comunes terminan compartiendo los mismos modelos de datos para los mensajes. Si los servicios web representados por estas definiciones WSDL alguna vez necesitan interactuar como parte de una composición, ya habrán establecido un grado significativo de interoperabilidad intrínseca.

Vale la pena señalar que en el mundo real de los servicios web, este nivel de estandarización de representación de datos plantear desafíos en torno al gobierno de los estándares y los esquemas mismos.

Un patrón de diseño establecido llamado patrón de inventario de dominios admite la partición de una empresa en dominios separados que pueden ser estandarizados y gobernados de forma independiente.

Lograr un nivel significativo de estandarización puede resultar muy beneficioso, ya que respalda la creación de composiciones de servicio efectivas y mejora tanto la eficiencia con la que se pueden prestar estos servicios como la eficiencia con la que llevan a cabo sus funciones en tiempo de ejecución.

Sin embargo, el beneficio notable de incorporar un conjunto centralizado de esquemas es que este conjunto de esquemas se puede mantener en apoyo de múltiples servicios (a diferencia de múltiples conjuntos de esquemas dispares que requieren mantenimiento separado y continuo).

Ejemplo del mundo real:

Una compañía internacional de servicios financieros dividió su empresa en dominios comerciales, cada uno de los cuales representa un límite en el que se estandarizaron los esquemas XML se refiere; (una implementación de los patrones de diseño mencionados anteriormente).

Esto estableció una arquitectura de representación de datos autónoma en cada dominio.

Los servicios se están diseñando actualmente para dos de estos dominios con el objetivo de establecer inventarios de servicios gobernados por separado.

Dentro de cada inventario, se espera que los servicios compartan repetidamente esquemas XML centralizados que representan documentos comerciales comunes (también denominados "esquemas centrados en la entidad").

Un objetivo principal de este esfuerzo es una reducción dramática en los requisitos de transformación de datos.

## Estandarizacion de las politicas del Servicio.

Las definiciones de Políticas del Servicio (WS-Policy) agregan una capa separada de abstracción potencial a los contratos de servicio, permitiendo que la lógica de la política se exprese a través de afirmaciones de políticas individuales que pueden estar contenidas en documentos de definición de políticas físicamente separados.

La estandarización entra en juego en varios niveles, dependiendo de la naturaleza de las políticas y la medida en que las políticas se utilizan en general.

### Vocabularios de afirmación de propiedad

Al estandarizar los servicios con definiciones de políticas separadas, el punto focal generalmente está en la sintaxis utilizada para expresar afirmaciones de políticas.

Algunas afirmaciones de políticas están predefinidas otras pueden personalizarse utilizando vocabularios propietarios del negocio que expresan reglas comerciales o políticas comerciales corporativas internas.

De la misma manera que el esquema XML admite la creación de una capa de representación de datos abstractos las definiciones de políticas introducen la oportunidad de establecer vocabularios abstractos utilizados específicamente para El contrato de servicio web con lógica de validación relacionada con la política.

Sin embargo, esto solo se puede lograr cuando dichos vocabularios están estandarizados en un inventario de servicios.

Ezfuerzo que el BP debera de realizar como parte de su implementacion de Gobierno SOA

De lo contrario, los desarrolladores y consumidores de servicios corren el peligro constante de formar sus propios bocabularios, una practica poco saludable.

### Parámetros y Políticas Anidadas

Las afirmaciones de política se pueden anidar o expresar mediante el uso de parámetros.

La integridad de la anidación de la política siempre es verificadas y validadas por el procesador de políticas del sistema. La mayoría de los procesadores ignoran las afirmaciones expresadas con parámetros y, por lo tanto, debe de ser verificadas por la lógica de servicio de manera subyacente o previa a la implementación del Ws.

Una excepción a esto es cuando el servicio está alojado en un entorno propietario del proveedor que proporciona un procesador de políticas capaz de validar parámetros que expresan afirmaciones que fueron predefinidas y proporcionadas por la plataforma del proveedor. (Ej: La mayoría de los WS de soluciones de caja, tales como sistemas ERP, Soluciones de Retail o Core Banking etc.)

En general, se considera una mejor práctica utilizar siempre confirmaciones de políticas anidadas.

Sin embargo, también plantea la necesidad de una convención de diseño relacionada asociada con el uso de parámetros y políticas anidadas. Cómo y cuándo se deben aplicar estas partes del lenguaje WS-Policy debe establecerse explícitamente dentro de los estándares de diseño para evitar la creación de definiciones de políticas inconsistentes.

### Modularidad y Centralización de Politicas

Si bien es relativamente común que se creen definiciones de políticas personalizadas para servicios web individuales, debido a que las confirmaciones de políticas se pueden separar en documentos de definición de políticas separados, las políticas se pueden modularizar permitiendo incluso que algunas se centralicen.

La aplicación del concepto de centralización a las políticas esencialmente permite la creación de una definición de política básica que contiene confirmaciones amplias y generalizadas.

Se pueden colocar confirmaciones más especializadas en definiciones de políticas separadas que luego se pueden adjuntar a la misma definición WSDL cuando se invoca el servicio web durante una rutina de tiempo de ejecución, como set trata de ilustrar en la siguiente figura.



Ejemplo Se define una política de seguridad que contiene aserciones (confirmaciones) que se aplican a todas las definiciones WSDL. Sin embargo, solo uno de estos contratos de servicios web debe ampliarse aún más con una afirmación especializada asociada solo para el (procesamiento de reclamos).

### Estándar Estructural

Las políticas pueden afectar la estructura del contrato de servicio.

Las confirmaciones de políticas individuales pueden aplicarse a diferentes partes de una definición WSDL y pueden adjuntarse a esas partes.

Además, la manera en que algunas confirmaciones se expresan estructuralmente también puede variar.

Es necesario tener en cuenta este tipo de consideraciones de diseño estructural al estandarizar los contratos de servicio.

La disparidad estructural, especialmente dentro de las definiciones de políticas altamente reutilizadas y centralizadas, van a provocar desafíos de gobernanza.

# Diseño del contrato de Servicios.

Los contratos forman la base para la comunicación entre servicios y, por lo tanto, representan el elemento arquitectónico más fundamental de una SOA.

El diseño orientado al servicio es un proceso dedicado a garantizar que los factores y problemas necesarios se tengan en cuenta al configurar un contrato de servicio a través de la orientación al servicio.

Lo siguiente es una colección de consideraciones de diseño específicas para la aplicación de este principio.

## Estandarización de representación de datos y Evitar La transformación

Una clave para comprender lo que se necesita para lograr que la estandarización de los modelos de datos suceda dentro de una organización es saber más acerca de por qué históricamente ha sido un objetivo tan difícil de lograr.

Con un enfoque en los servicios web, aquí hay algunas razones comunes por las que los contratos no están estandarizados:

* Fueron autogenerados por herramientas de desarrollo
* Formaban parte de los adaptadores de servicio adquiridos
* No existían estándares de diseño cuando se crearon
* Los estándares de diseño fueron ignorados

Una vez implementados y parte del entorno de producción, los contratos de servicios web no estandarizados dan como resultado la creación e implementación de diferentes modelos de datos. Para superar estas diferencias se requiere el uso de una tecnología de transformación de datos y la definición de lógica de mapeo entre un esquema y otro.

Este mapa se implementa en un componente de software actual, que posteriormente ejecuta la lógica de transformación en tiempo de ejecución cada vez que los servicios necesitan intercambiar información.

Las tecnologías de transformación de datos proporcionan características importantes esenciales para permitir la conectividad dentro de las arquitecturas empresariales integradas.

Sin embargo, al estandarizar el diseño de servicios como parte de un inventario de servicios bien definido, uno de nuestros objetivos principales es evitar tener que recurrir a la transformación de datos siempre que sea posible.

La transformación de datos presenta una serie de problemas, que incluyen:

* Se requiere un mayor esfuerzo de desarrollo de integración para crear la lógica de mapeo
* Aumento de la sobrecarga de rendimiento resultante de la necesidad de ejecutar la lógica de mapeo en tiempo de ejecución con cada intercambio de datos
* Mayor complejidad arquitectónica debido a la incorporación de capas de transformación
* aumento de la carga de gobernanza debido a la necesidad de mantener y evolucionar las capas de transformación junto con el resto de la arquitectura

La aplicación coherente de los estándares de representación de datos evita estos problemas al mantener alineados los modelos de datos del contrato de servicio como tratamos de ilústralo en la siguiente figura.



Ejemplo Al aumentar la estandarización de los contratos de servicio y la representación de datos de mensajes resultante, se reduce la cantidad de capas de transformación requeridas. Esto da como resultado una interoperabilidad más eficiente y simplificada, como se muestra en este diagrama donde se evita la transformación de mensajes en tiempo de ejecución cuando dos servicios comparten datos basados ​​en un esquema XML común.

Si bien el evitar la transformación a través de la estandarización de la representación de datos es factible es más desafiante cuando se comparten datos entre organizaciones. La falta de voluntad de las organizaciones para ponerse de acuerdo sobre el mismo esquema de manera predecible conduce a la necesidad de la transformación de datos.

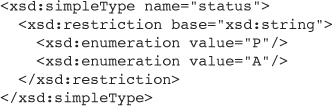
Un conocimiento previo de estos problemas pone aún más énfasis en realizar una adecuada gobernanza SOA de la organización para que se pueda mitigar el impacto general y la carga de las capas de transformación.

## Estandarización de la Granuralidad

El uso de estándares de diseño puede afectar los cuatro tipos de granularidad de diseño relacionada con el servicio.

* La granularidad de nivel de servicio se ve afectada y, a menudo, determinada por la elección del modelo de servicio.
* Si bien la granularidad de la capacidad a menudo se define inicialmente cuando se lleva a cabo el proceso de modelado del servicio, los estándares de diseño derivados o basados ​​en patrones de diseño concentrados en el contrato de servicio a menudo darán forma a la granularidad de una capacidad o agregarán capacidades con diferentes niveles de granularidad.
* La granularidad de los datos también puede definirse directamente por los estándares de representación de datos y verse más afectada por los estándares de diseño arquitectónico relacionados con la regulación de los tamaños de los mensajes y los viajes de ida y vuelta por razones de escalabilidad y rendimiento.
* Finalmente, la granularidad de restricción a menudo está dictada directamente por una serie de estándares de diseño detallados. Por lo general, se trata de convenciones centradas en la validación que especifican cuán flexibles pueden ser las restricciones (en términos de permitir rangos de valores de datos de entrada o salida) y cómo, específicamente, las restricciones mismas deben expresarse.

La incorporación de valores de código predeterminados en los tipos de datos, por ejemplo, puede aumentar drásticamente la granularidad de restricción (como se muestra en el Ejemplo Siguiente).



Ejemplo Un ejemplo de una restricción muy fina que utiliza códigos estandarizados de un solo carácter para representar valores de estado pendientes ("P") y aprobados ("A").

Los estándares de diseño también pueden determinar la cantidad y el tipo de lógica de validación que debe ubicarse en el contrato de servicio.

Por ejemplo, puede haber un estándar de diseño que requiera que la lógica de validación asociada con las reglas comerciales se lleve a cabo dentro de la lógica de servicio subyacente, no dentro del contrato de servicio. Si bien la cantidad general de lógica de validación sigue siendo la misma, la aplicación de este estándar de diseño particular daría como resultado una granularidad de restricción más gruesa porque esta forma de granularidad solo se refiere al contrato.

## Consideraciones entre el uso de contratos Estandarización y contratos de tareas.

Los modelos de servicio nos proporcionan plantillas para tipos comunes de servicios. Los dos modelos que cuentan con un alto potencial de reutilización son los servicios de entidades y servicios públicos. Los contratos para estos tipos de servicios reciben una gran atención, ya que están altamente optimizados y ajustados para facilitar numerosos escenarios de uso.

Los servicios de tareas a menudo se entregan para representar una sola tarea. Esto reduce las posibilidades de que se compartan, lo que también reduce la necesidad de optimizar el contrato más allá de los requisitos inmediatos para los que se presta el servicio. Del mismo modo, los servicios de tareas orquestadas también requieren contratos que sean más específicos de la lógica de procesos que representan.

Otra característica que distingue es que los servicios agnósticos necesitan proporcionar una variedad de funcionalidades genéricas, generalmente están equipados con varias capacidades finamente ajustadas. Los servicios específicos del proceso, por otro lado, a veces requieren solo una capacidad única que permite a un consumidor de servicios externos "poner en marcha" la lógica del proceso que encapsulan.

Independientemente de la naturaleza de la lógica representada por estos modelos de servicio, todos se benefician de estar sujetos al mismo conjunto de estándares de diseño y convenciones de denominación en la medida de lo posible.