Patrón de la centralizacion del Inventario de servicios

Tabla de contenido

[1 Generalidades. 3](#_Toc22553925)

[2 Proceso de Centralización del Servicio 3](#_Toc22553926)

[2.1.1 Problema 3](#_Toc22553927)

[2.1.2 Solución 4](#_Toc22553928)

[2.1.3 Aplicacion 4](#_Toc22553929)

[2.1.4 Impactos 4](#_Toc22553930)

[2.1.5 Relaciones 4](#_Toc22553931)

[3 ESQUEMA de Centralizacion del servicio 5](#_Toc22553932)

[3.1.1 Problema 5](#_Toc22553933)

[3.1.2 Solución 5](#_Toc22553934)

[3.1.3 Aplicacion 5](#_Toc22553935)

[3.1.4 Impactos 6](#_Toc22553936)

[3.1.5 Relaciones 6](#_Toc22553937)

[4 politicas de Centralizacion del servicio 6](#_Toc22553938)

[4.1.1 Problema 6](#_Toc22553939)

[4.1.2 Solución 7](#_Toc22553940)

[4.1.3 Aplicacion 7](#_Toc22553941)

[4.1.4 Impactos 7](#_Toc22553942)

[4.1.5 Relaciones 8](#_Toc22553943)

[5 reglas de Centralizacion del servicio 8](#_Toc22553944)

[5.1.1 Problema 8](#_Toc22553945)

[5.1.2 Solución 9](#_Toc22553946)

[5.1.3 Aplicacion 9](#_Toc22553947)

[5.1.4 Impactos 9](#_Toc22553948)

[5.1.5 Relaciones 10](#_Toc22553949)

# Generalidades.

Los siguientes patrones abordan aspectos físicos de la arquitectura de inventario de servicios, de la siguiente manera:

1. Procesos de Centralización
   1. Aboga por que la lógica asociada con diferentes procesos de negocios se mantenga en la misma ubicación.
2. Esquemas de Centralización.
   1. Posiciona esquemas estandarizados como partes físicamente independientes de la arquitectura de inventario para que puedan compartirse entre los servicios y también usarse independientemente de los servicios.
3. Políticas de Centralización.
   1. Ayuda a establecer políticas globales y de nivel de dominio que están físicamente aisladas y, por lo tanto, también pueden ser compartidas y aplicadas a múltiples servicios.
4. Reglas de Centralización
   1. Se centra en separar la lógica de procesamiento y el almacenamiento de datos específicos para la gestión de datos de reglas de negocio.

Como se trata de patrones de centralización, cada uno introduce cierto grado de estandarización en todo el inventario.

Este es un requisito de aplicación (e impacto) importante y recurrente para todos los patrones de Servicios web y el SOA.

# Proceso de Centralización del Servicio

### Problema

Dentro de entornos que contienen inventarios de servicios muy grandes, es común que lleguen requerimientos para implementar Servicios de Tareas la idea es que este tipo de servicios puedan respaldar simultáneamente la automatización de múltiples procesos comerciales.

Los servicios de tareas se implementan comúnmente como servicios web individuales. Debido a que cada programa contiene lógica de proceso de negocio integrada, esto da como resultado en una arquitectura físicamente descentralizada.

Si bien estos servicios existen como miembros homologos a los servicios de un inventario de servicios, el hecho de que se implementen de forma independiente da como resultado que la lógica del proceso comercial de una empresa se distribuya físicamente en varias ubicaciones. Cuando se producen cambios, se inhibe la capacidad de ampliar, racionalizar o incluso combinar la lógica del proceso de negocio de manera eficiente porque la lógica subyacente de cada servicio de tarea afectado debe revisarse, abrirse y cambiarse, según sea necesario.

Además, debido a la naturaleza de la lógica de flujo de trabajo variable, algunos procesos comerciales no pueden llevarse a cabo en tiempo real. En cambio, pueden imponer actividades de servicio de larga duración que pueden abarcar minutos, horas e incluso días.

### Solución

La lógica del proceso comercial principal (que representa algunos o todos los procesos comerciales dentro de un dominio determinado) se centraliza en una ubicación. Una plataforma de orquestación aloja y ejecuta esta lógica mientras permite su mantenimiento continuo y centralizado.

Los servicios de tareas pueden continuar siendo implementados como servicios web separados, pero como parte de una plataforma de orquestación, su lógica de proceso comercial colectiva se ubica y gobierna centralmente (lo que resulta en servicios de tareas "orquestados").

### Aplicacion

Las variaciones modernas de las plataformas de orquestación que surgieron durante la era EA proporcionan un medio fundamental para centralizar la lógica del proceso. Cuando se combinan con el soporte para lenguajes de definición de procesos de negocio abiertos (como WS-BPEL), estas plataformas se vuelven adecuadas para establecer una capa de composición primaria dentro de SOA.

Para realizar este patrón de diseño, se requiere una plataforma de orquestación moderna. Dicho entorno generalmente se compone de lo siguiente:

* + - una herramienta gráfica frontal que permite a los usuarios expresar y mantener la lógica del proceso comercial
    - un entorno de tiempo de ejecución de middleware back-end capaz de alojar servicios de tareas orquestadas y la colección correspondiente de definiciones de procesos empresariales creadas con la herramienta front-end
    - características que cumplen con los estándares de la industria relacionados con la expresión y ejecución de la lógica de procesos de negocios, como WS-BPEL

En pocas palabras, la lógica de composición para un proceso empresarial específico se define utilizando la herramienta de front-end y luego se encapsula mediante un servicio de tareas orquestado específico. La plataforma de back-end aloja el servicio en el mismo entorno que otros, lo que permite que estos servicios lleven a cabo su lógica de composición con una gama de características de soporte, incluida la administración del estado y varios agentes de servicio.

### Impactos

Introducir la tecnología de orquestación en el BP puede ser costoso y disruptivo. Los requisitos de infraestructura para alojar y ejecutar el middleware necesario pueden aumentar el tamaño y los costos operativos generales del entorno de TI en su conjuntos, su introducción afectará la infraestructura circundante y la complejidad de la arquitectura tecnológica general, incluidos los inventarios de servicios existentes.

Además, los productos de front-end y back-end necesarios para soportar la orquestación rara vez se implementan de forma aislada. Al crear un entorno de orquestación, la plataforma de middleware generalmente se expande para abarcar una gama de funciones centralizadas de gobierno de servicios.

### Relaciones

Este patrón plantea una serie de consideraciones arquitectónicas debido a que el proceso de centralizacion representa una parte central de la orquestación.

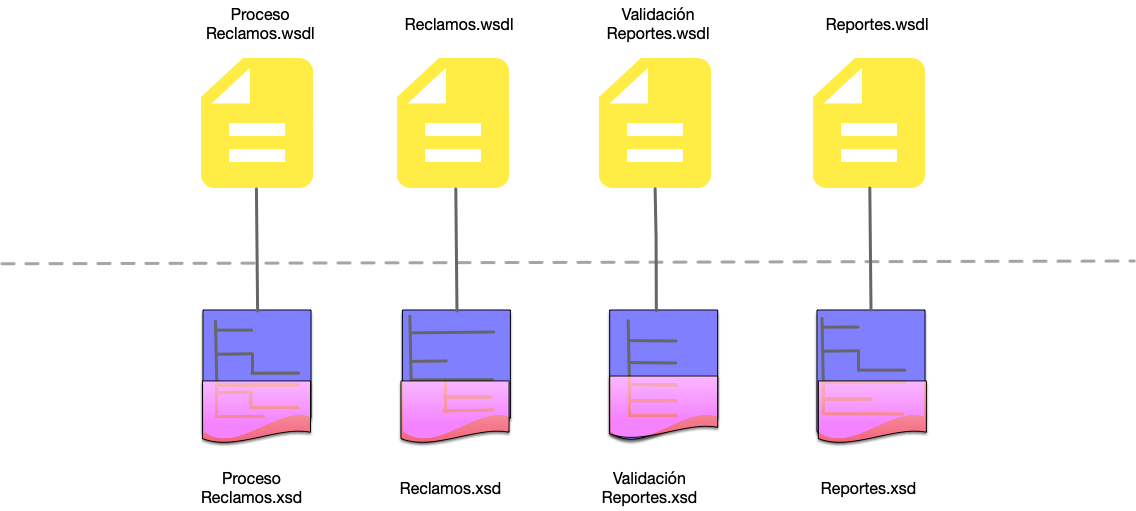
La centralización de procesos establece un centro de procesos físicos dentro de una arquitectura y, por lo tanto, puede afectar la aplicación de varios otros patrones.

# ESQUEMA de Centralizacion del servicio

### Problema

Al crear servicios para entornos empresariales más grandes, el contexto establecido por cada límite de servicio generalmente no será exclusivo de un cuerpo de datos.

Como resultado, surge la necesidad de modelos de datos de esquema duplicados, lo que lleva a la definición de contratos de servicio con contenido redundante, Incluso si los modelos de datos en estos contratos están estandarizados, la implementación redundante y descentralizada de esquemas de contratos presenta desafíos constantes de gobernanza asociados principalmente a mantener sincronizados los modelos de datos de esquemas.

Figura 1 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 en realidad puede introducir una redundancia de contenido de esquema significativa como lo indican las áreas sombreadas en rojo.

### Solución

Los esquemas pueden diseñarse e implementarse independientemente de las capacidades de servicio que los utilizan para representar la estructura y la tipificación del contenido del mensaje. Como resultado, se puede establecer y estandarizar una arquitectura de esquema de forma algo separada de la capa de servicio principal. Por ejemplo, si se define un esquema que representa datos de reclamos, cualquier servicio con una capacidad que necesite procesar datos de reclamos usaría el mismo esquema.



Figura 2 Las definiciones WSDL que comparten esquemas XML comunes terminan compartiendo los mismos modelos de datos de mensajes. Observe cómo la reducción de contenido redundante también da como resultado esquemas de menor tamaño.

### Aplicacion

Lo que defiende principalmente este patrón es la creación de una arquitectura de esquema (o representación de datos) independiente. Los esquemas deben definirse como parte de la iniciativa SOA, y se recomienda que se creen antes de la finalización de los contratos de servicios individuales.

Idealmente, la incorporación de la capa de esquema separada se tiene en cuenta después de la finalización del plan de inventario de servicio, en preparación para la entrega del inventario de servicio físico.

Se sugiere la siguiente secuencia:

1. Complete un plan de inventario de servicios para establecer una representación conceptual de los servicios planificados dentro de un inventario.
2. Determine las definiciones de esquema centralizado requeridas para representar las entidades comerciales comunes y los conjuntos de información que probablemente sean procesados ​​por los servicios en este inventario.
3. Cree las definiciones de esquema aplicando estándares de diseño para garantizar la coherencia y la normalización.
4. Cree la definiciones WSDL utilizando los esquemas estándar donde sea apropiado y complementando el contrato con cualquier esquema específico de servicio requerido.

### Impactos

Debido a las dependencias formadas en las definiciones de esquema compartido, su diseño inicial es crucial. Después de que varios contratos de servicio forman enlaces a un esquema, la evolución de la definición del esquema se convierte en una parte clave de la gobernanza general del inventario de servicios. Cualquier cambio en un esquema centralizado puede afectar numerosos contratos de servicio.

Para organizaciones grandes, este nivel de estandarización de datos puede plantear desafíos desalentadores, muchos de los cuales giran en torno al mantenimiento de los esquemas compartidos y la aplicación de los estándares de diseño asociados.

### Relaciones

La capa de esquema establecida por los esquemas centralizados se puede construir e incorporar como parte de un esfuerzo de Centralización de contrato y, debido a su énfasis en reducir la redundancia del modelo de datos, también lleva adelante los objetivos de Normalización de servicio.

Puede existir una relación interesante entre este patrón y la Abstracción de validación en que la creación de esquemas centralizados oficiales puede traer más detalles de los que todos los servicios pueden requerir.

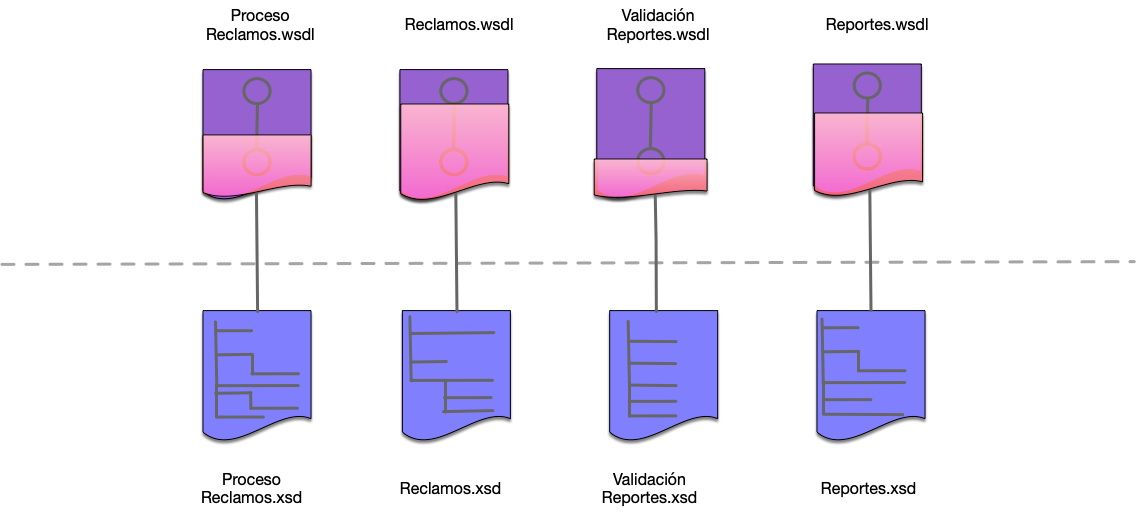
# politicas de Centralizacion del servicio

### Problema

Es posible que se requieran servicios para procesar una variedad de políticas individuales (también llamadas expresiones de políticas). Las áreas comúnmente abordadas por las políticas incluyen requisitos de seguridad y transacción, así como una variedad de propiedades de calidad de servicio (QoS).

Las políticas regulatorias pueden afectar una variedad de servicios, mientras que otras políticas pueden ser específicas del servicio. Un servicio creado como un servicio web puede establecer requisitos de política como parte de su contrato mediante el uso de expresiones WS-Policy, o puede aplicar políticas dentro de su lógica de servicio subyacente.

Cuando las políticas comunes se repiten en múltiples contratos de servicio, introducen redundancia en el inventario de servicios. Esto conduce a un contenido de políticas inflado y aumenta la carga de gobernanza requerida para garantizar que las políticas comunes se mantengan sincronizadas a lo largo del tiempo.

Figura 3 Cada uno de los cuatro documentos WSDL mostrados se amplía con definiciones individuales de WS-Policy. El sombreado rojo indica el grado de redundancia entre las políticas.

### Solución

Las políticas que se aplican a múltiples servicios se pueden resumir en documentos de definición de políticas o agentes de servicios que forman parte de un marco de aplicación de políticas de todo el inventario. Las políticas resumidas pueden posicionarse para aplicarse a múltiples servicios, reduciendo así la redundancia y proporcionando una gobernanza centralizada de las políticas .

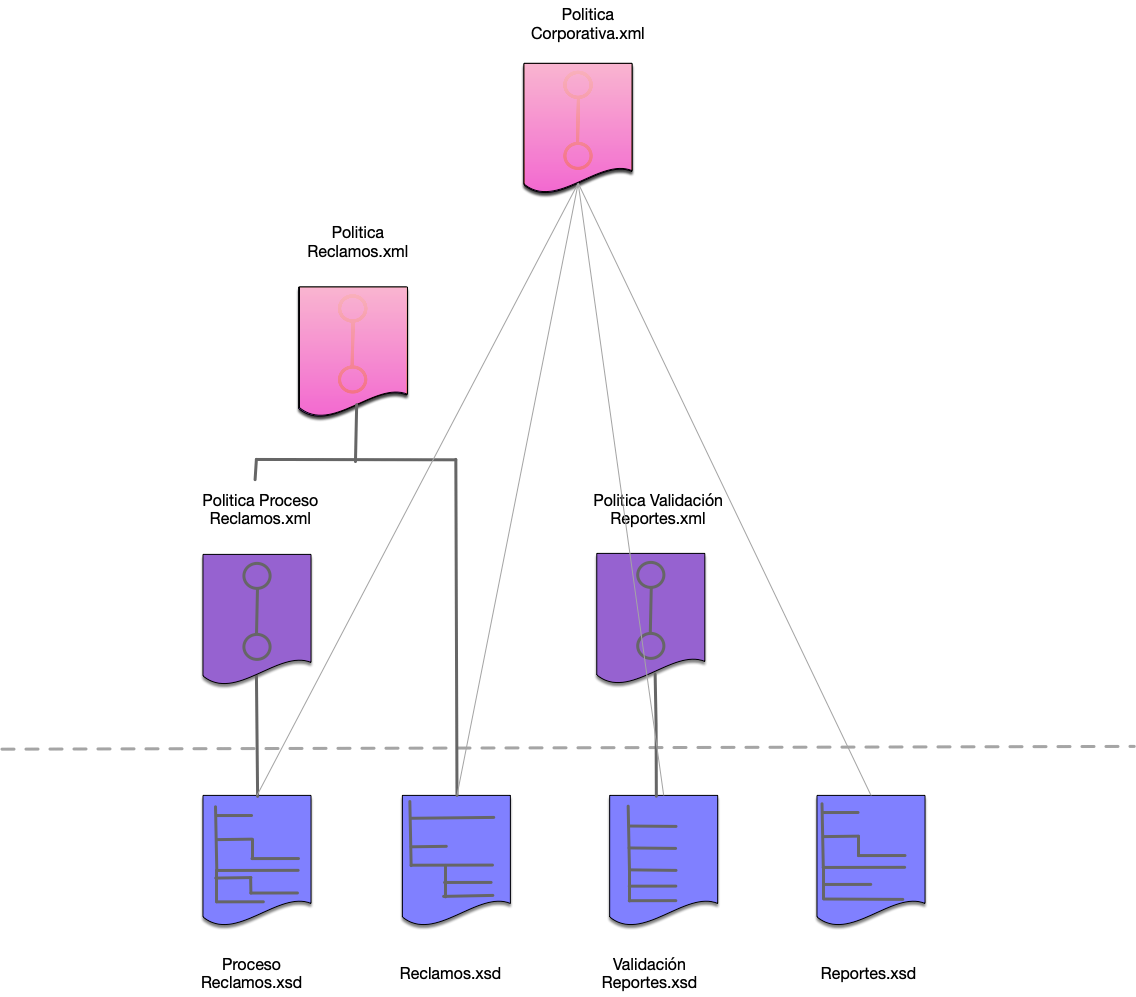


Figura 4 Se establece una definición de política global (Corporate Policy.xml) y se aplica a todos los documentos WSDL, y se crea una política de dominio adicional (Claims Policy.xml) y se vincula a dos definiciones WSDL. Esta nueva estructura de políticas elimina la redundancia del contenido de las políticas y garantiza una aplicación coherente de las políticas.

### Aplicacion

Es necesario agregar un marco de políticas a la arquitectura de inventario para que las políticas se puedan definir y asociar por separado con los servicios y luego validar, aplicar e incluso auditar en tiempo de ejecución.

El marco WS-Policy incluye una especificación WS-PolicyAttachments separada que explica los mecanismos vinculantes para las políticas. Las definiciones de políticas se pueden incrustar o vincular a documentos WSDL. Para aplicar este patrón, una o más políticas generalmente deben agruparse en una definición de política que esté disponible para que los contratos de servicios web a los que se aplican las políticas puedan agregar las referencias apropiadas.

Las plataformas de clase mundial (como las que proporcionan los productos ESB) pueden proporcionar características de políticas compatibles con los agentes de tiempo de ejecución que llevan a cabo la verificación de cumplimiento de políticas. En estos entornos, las políticas globales y de nivel de dominio también se pueden establecer a través de agentes de servicio que actúan como puntos de aplicación de políticas.

### Impactos

La definición de políticas es un paso adicional en el ciclo de vida de prestación de servicios que debe tenerse en cuenta desde el principio. Parte del análisis involucrado en la definición de políticas es determinar qué tipos de políticas no deberían formar parte del contrato de servicio técnico.

Algunas políticas pueden estar sujetas a cambios inesperados y, por lo tanto, es más probable que exijan nuevas versiones de contrato. Otras políticas pueden ser más adecuadas para un acuerdo de nivel de servicio (SLA) que existe como un documento utilizado por humanos (generalmente los propietarios de posibles programas de consumo). Por lo tanto, el principio de abstracción del servicio es un factor clave para garantizar que la granularidad de restricción de las capacidades del contrato siga siendo razonable.

Una vez que se establecen las políticas globales o de nivel de dominio, deben mantenerse con gran cuidado. Un cambio o adición a una política compartida afectará a todos los servicios que dependen de ella. Esto, a su vez, afecta a todos los consumidores de servicios que han formado dependencias de los contratos de servicios correspondientes. Por lo tanto, se requiere una estructura de gobierno, compuesta de roles asignados (como custodios de políticas) y procesos que garanticen que las políticas comunes se desarrollen adecuadamente.

Además, se requiere un mayor análisis por adelantado antes de la entrega de las definiciones de WS-Policy para que las políticas se diseñen con el equilibrio correcto de restricciones y flexibilidad para acomodar el rango de contratos que pueden requerirse para usarlas. Un problema común cuando se trabaja con políticas centralizadas es que pueden surgir conflictos entre políticas a diferentes niveles. Por ejemplo, una nueva política global puede contradecir inadvertidamente una política de nivel de servicio para un servicio en particular. El análisis formal y los procesos de gobernanza pueden ayudar a evitar estas situaciones.

Además, los agentes de servicio y los servidores proxy que establecen los puntos de aplicación de políticas dentro de la arquitectura de inventario pueden agregar sobrecarga de rendimiento y modos de falla independientes, que la infraestructura circundante necesita para adaptarse. Cada política centralizada agrega efectivamente una capa de procesamiento en tiempo de ejecución y dependencia del servicio.

Finalmente, cuando se implementa un marco de políticas basado en el uso de agentes de servicio (como parte de un producto ESB, por ejemplo), es relativamente común que el estándar WS-Policy no sea totalmente compatible. En cambio, el marco puede requerir que las políticas se definan a través de herramientas de front-end que generan un formato de política patentado. Una vez implementado, esto puede conducir a escenarios de bloqueo de proveedores indeseables que contrarrestan los objetivos de la característica de arquitectura neutral del proveedor. El uso de formatos de políticas patentados puede prevenir el intercambio de datos entre organizaciones a menos que ambas organizaciones usen los mismos productos.

### Relaciones

Debido a que la centralización de políticas establece esencialmente una capa de políticas independiente que extiende los contratos de servicios, se relaciona directamente con la centralización de contratos y se beneficia de ellos.

Este patrón continúa los conceptos establecidos por la Normalización del servicio en el sentido de que evita la redundancia entre las políticas a través de la centralización, y debido a que la Centralización de políticas puede afectar el contenido de un contrato de servicio, existe una relación adicional con la Abstracción de validación.

Figura 5 La centralización de políticas posiciona las políticas dentro de una arquitectura de inventario de servicios y, por lo tanto, afecta otros patrones que se relacionan con la capa de contrato de servicio o con la naturaleza de la lógica de la política.

# reglas de Centralizacion del servicio

### Problema

La lógica del flujo de trabajo dentro de cualquier proceso comercial determinado se basa en reglas específicas de cómo se debe llevar a cabo la lógica, de acuerdo con las políticas, regulaciones y preferencias de la organización. Las capacidades de servicio comercial individual con frecuencia deben llevar a cabo su lógica encapsulada de acuerdo con estas reglas.

No es raro que la misma regla se aplique a diferentes escenarios que involucran a diferentes entidades comerciales. Esto da como resultado la necesidad de incorporar una regla dentro de múltiples cuerpos de lógica de servicio. A medida que una organización cambia con el tiempo, también cambian ciertas reglas comerciales. Esto puede conducir a modificaciones dentro de los servicios empresariales de la entidad individual, así como a la lógica del proceso empresarial encapsulada por los servicios de tareas o de otro tipo (incluido el servicio de utilidad ocasional). Tener que volver a visitar múltiples servicios cada vez que cambia una regla de negocio puede ser contraproducente.

### Solución

Las reglas de negocio pueden abstraerse físicamente en una parte dedicada de la arquitectura bajo la administración de motores y plataformas de reglas especializadas. Esto centraliza el acceso a la lógica de las reglas de negocio y evita la redundancia. Además, centraliza la gobernanza de las reglas comerciales para que puedan modificarse y evolucionar desde una única ubicación.

### Aplicacion

Existen diferentes sistemas de gestión de reglas de negocio, cada uno de los cuales presenta una plataforma de administración y tiempo de ejecución relativamente propietaria. Se puede establecer un servicio central para proporcionar un punto de acceso oficial para la creación, modificación, recuperación y aplicación de reglas comerciales.

Las plataformas de tiempo de ejecución modernas también ofrecen repositorios de reglas nativas y lógica de procesamiento a la que se puede acceder a través de un conjunto de agentes de servicio del sistema y API. Esto permite que cualquier servicio interactúe con la lógica relacionada con las reglas comerciales sin tener que componer un servicio separado.

### Impactos

Debido a que este patrón se aplica en todo un inventario de servicios, puede afectar a una arquitectura de varias maneras:

* + - Si bien logra la centralización de los datos de las reglas comerciales dentro de un inventario, la centralización de reglas también termina descentralizando la lógica comercial asociada con los servicios comerciales.
    - Los requisitos de rendimiento de los servicios afectados aumentan debido a la necesidad de recuperar o aplicar las reglas de negocio en tiempo de ejecución. Los mecanismos de almacenamiento en caché pueden aliviar este impacto hasta cierto punto (generalmente cuando las reglas se almacenan temporalmente como información de estado para una composición de servicio particular).
    - Si las características existentes de la plataforma de tiempo de ejecución no pueden aprovecharse para establecer una administración centralizada de reglas, este patrón generalmente da como resultado la introducción de un producto de administración de reglas comerciales separado. Esta extensión puede aumentar el tamaño, la complejidad y el costo operativo general de una arquitectura tecnológica y, además, debe ser lo suficientemente confiable como para acomodar consistentemente los patrones de uso del servicio. Un sistema de administración de reglas propenso a fallas en el tiempo de ejecución puede paralizar un inventario de servicio completo.
    - El acceso a reglas comerciales centralizadas a través de agentes de sistema nativos y API impondrá dependencias arquitectónicas estrictas sobre los servicios. Si muchos servicios empresariales usan estas características de tiempo de ejecución, el inventario de servicio general podría quedar "bloqueado" en una plataforma de proveedor en particular.
    - Debido a que la lógica de la regla comercial real está físicamente separada, el alcance de la lógica encapsulada por varias capacidades de servicio comercial es incompleto (según sus contextos principales), y su autonomía general disminuye.

Otra cuestión que vale la pena señalar es la gestión real de las reglas comerciales centralizadas. A menudo, un administrador técnico está a cargo del sistema de reglas, pero es posible que se necesiten varios expertos en dominios comerciales para mantener las reglas comerciales. Esto puede llevar a desafíos de propiedad en el sentido de que el custodio de un servicio comercial también debe estar involucrado en el mantenimiento de un subconjunto de las reglas comerciales dentro del repositorio central de reglas, además de la gobernanza del servicio comercial en sí.

### Relaciones

Las reglas comerciales se pueden encontrar en casi cualquier lugar dentro de una solución típica orientada al servicio, por lo que la abstracción y centralización de los datos de las reglas pueden afectar el contenido de un contrato de servicio.

Las reglas comerciales centralizadas se aprovechan comúnmente para aumentar la sofisticación con la que los productos ESB llevan a cabo funciones de mensajería, enrutamiento y acarreo. En entornos ESB, la variación de este patrón que resulta en agentes nativos y API es más común que la creación de servicios de reglas dedicados. Del mismo modo, este patrón puede ser aprovechado por Orquestacion para que la lógica de reglas de negocio pueda incorporarse en el flujo de trabajo y la lógica de composición.