

# AGESIC

## Área de Tecnología

### PDI - Versionado de servicios

#### Historial de Revisiones

Fecha	Versión	Descripción	Autor	Aprobado Por
28/04/2011	1.0	Versión inicial	Esteban Hourcade	

Nombre actual del archivo: VersionadoWS-v03-00.odt

## Índice de contenido

Introducción.....	3
Tipos de versiones.....	3
Cambio compatible .....	3
Cambio incompatible.....	3
Documentación de versión en WSDL.....	4
Namespace.....	4
Esquemas.....	5
Gestión del cambio.....	5
Versión menor.....	5
Versión mayor.....	5
Recomendaciones en cambios.....	6

## Introducción

El presente documento describe las principales consideraciones que se deben tomar en cuenta al momento de actualizar un servicio que es publicado a través de la Plataforma de Interoperabilidad. El objetivo primario es evitar que al momento de realizar un cambio de un servicio, esto no afecte la continuidad del negocio de aquellos organismos que consumen estos servicios.

## Tipos de versiones

Distinguimos dos tipos de cambio cuando hablamos de nuevas versiones:

### ***Cambio compatible***

Son aquellos cambios sobre el contrato de servicio que no afectan negativamente a los consumidores del servicio. Significan un cambio de versión Menor.

#### Nomenclatura

Para aquellas versiones que signifiquen un cambio compatible se utilizará la siguiente lógica: vieja versión: N.m => nueva versión: N.(m+1)

Utilizando una v minúscula antes del número para redundar en un cambio de versión menor:

*Ej: nombreDeServicio\_v.1.3*

#### Son cambios compatibles

- *Agregado de nuevas operaciones WSDL*
- *Agregado de nuevos esquemas XML que no estén contenidos por tipos ya existentes*
- *Agregado de parámetros opcionales a operaciones ya existentes*

*Cuando se realiza un cambio compatible no es necesario mantener la versión anterior del servicio.*

### ***Cambio incompatible***

Impactan negativamente a los consumidores. Significan un cambio de versión Mayor.

#### Nomenclatura

Para aquellas versiones que signifiquen un cambio incompatible se se utilizará la siguiente lógica: vieja versión: N.m => nueva versión: (N+1).0

Utilizando una V mayúscula antes del número para redundar en un cambio de versión mayor:

*Ej: nombreDeServicio\_V.2.0*

### Son cambios incompatibles

- Remover una operación
- Renombrar una operación
- Cambiar orden o tipo de los parámetros de una operación
- Cambiar la estructura de un tipo complejo de datos (complexType)
- Cambiar el *namespace*
- Cambiar el soapAction
- Cambiar orden o tipo de datos de la respuesta de un método

*Cuando se realiza un cambio incompatible se deberá mantener la versión anterior del servicio hasta que no existan consumidores utilizando esa versión. El tiempo de vida máximo de una determinada versión puede ser acordado al momento de realizar una actualización entre los distintos actores involucrados en el cambio.*

## Documentación de versión en WSDL

Tanto para versiones compatibles o incompatibles debe documentar en el WSDL con la etiqueta “wsdl:documentation” la descripción, autor, la versión actual y el historial de cambio, tal como se muestra en el siguiente ejemplo:

```
<wsdl:service name="cabezalService">
  <wsdl:documentation>
    Servicio para creación de cabezales de traza

    Autor: Juan Perez
    Version: 1.2
    History: 1.2 2016-01-07 Agregada operación 'persist'
            1.1 2015-12-15 Agregado atributo enum
            1.0 2015-10-19 Versión inicial
  </wsdl:documentation>
  <wsdl:port binding...>...</wsdl:port>
</wsdl:service>
```

Justo debajo de la definición del servicio(tag:wsdl:service), siendo “documentation” el primer nodo hijo.

## Namespace

Para los *cambios compatibles* debe mantenerse *el mismo namespace*, pero para los *cambios incompatibles*, y como se mantienen dos versiones simultáneamente, *se debe cambiar*.

Se propone mantener el id de versión mayor luego del namespace elegido, por ejemplo: “namespace/v1”.

En este escenario el cambio de namespace se haría solamente en el cambio de versión mayor: “namespace/v2”.

## **Esquemas**

Cambios compatibles:

- No cambiar el namespace ni en el WSDL ni en el esquema XML

Cambios incompatibles:

- Para cambios mayores, incompatibles, cambiar el namespace del WSDL y el namespace de los documentos de esquema XML a los cuales el WSDL refiere. El nuevo namespace debe contener el número de versión mayor.

## **Gestión del cambio**

### ***Versión menor***

Se propone sobrescribir el WSDL en la Plataforma de Interoperabilidad y ejecutar un set de pruebas previamente definido. El set de pruebas debe contener pruebas de todas las versiones de consumo de los wsdl's anteriores (de versiones menores anteriores).

*En el ambiente de testing se debe probar que efectivamente se trata de un cambio compatible y que los consumidores de versiones anteriores pueden consumir sin problemas.*

### ***Versión mayor***

Se debe dar de alta el nuevo wsdl, sin sobrescribir el anterior, de modo que ambas versiones coexistan durante un período de tiempo determinado.

**Es importante que el proveedor del servicio se comprometa a mantener ambas versiones en línea por un tiempo acordado.**

Se le comunicará a los consumidores de la versión anterior que en un período de tiempo dado se retirará la versión anterior. En este período deberán desarrollar su implementación de consumo de la nueva versión. Pasado ese tiempo se efectúa el retiro del WS.

El proceso de retiro debe ser:

*Publicado - > Desuso -> Retirado*

## **Recomendaciones en cambios**

Se pueden tomar ciertas acciones para evitar un cambio de versión mayor en determinadas ocasiones:

- Si se desea agregar un parámetro a una operación ya existente:

Puede mantener una versión menor al agregar una nueva operación en el wsdl.

Puede mantener su código de la siguiente manera:

```
método1(){  
método1_4(){  
método1_6(){
```

Realizar un cambio mayor(v1 -> v2) es una buena oportunidad para unificar las versiones. Podría ser tan fácil como tomar la mayor versión del método y borrar las demás:

```
//método1(){  
//método1_4(){  
//método1_6(){  
método2(){  
    //método1_6(){  
}
```