

## Capítulo VI

# Administración de Versiones

## **Administración de versiones**

### **Tabla de contenido**

1.-	¿En qué consiste la administración de versiones?.....	83
1.1.-	Ventajas .....	85
1.2.-	Barreras .....	85
2.-	Consideraciones de tipo práctico.....	86
2.1.-	Tipos de versiones .....	86
2.2.-	Actualización y custodia de componentes .....	87
2.3.-	Versiones anteriores.....	88
2.4.-	Nomenclatura de versiones .....	88
2.5.-	Estatus de una versión.....	89
3.-	Actividades de administración de versiones .....	89
3.1.-	Planificación .....	90
3.2.-	Verificación de cumplimiento de estándares .....	91
3.3.-	Verificación de cumplimiento de pruebas .....	91
4.-	Implementación.....	93
5.-	Soporte inicial .....	94
6.-	Evaluación de la disciplina.....	94

# **Administración de versiones**

---

---

## ***1.- ¿En qué consiste la administración de versiones?***

La administración de versiones es la disciplina que incluye las tareas de coordinación y control sobre el proceso de implementación de todo el software instalado en el ambiente de producción.

Algunos autores prefieren incluir también el control del hardware instalado dentro de esta disciplina; sin embargo, dado que hardware y software son elementos tan diferentes no consideramos práctico que la administración de versiones se ocupe de los ítems de hardware. Por ejemplo, el software definitivo –instalado en producción- requiere que una copia de los programas ejecutables y sus correspondientes módulos fuente, sean custodiados bajo estrictos controles; sin embargo, no tiene ningún sentido plantear algo similar para el hardware instalado. Consideramos además, que la administración de cambios y de configuraciones ofrecen unos marcos procedimentales suficientemente amplios para el control y la administración de los ítems de hardware.

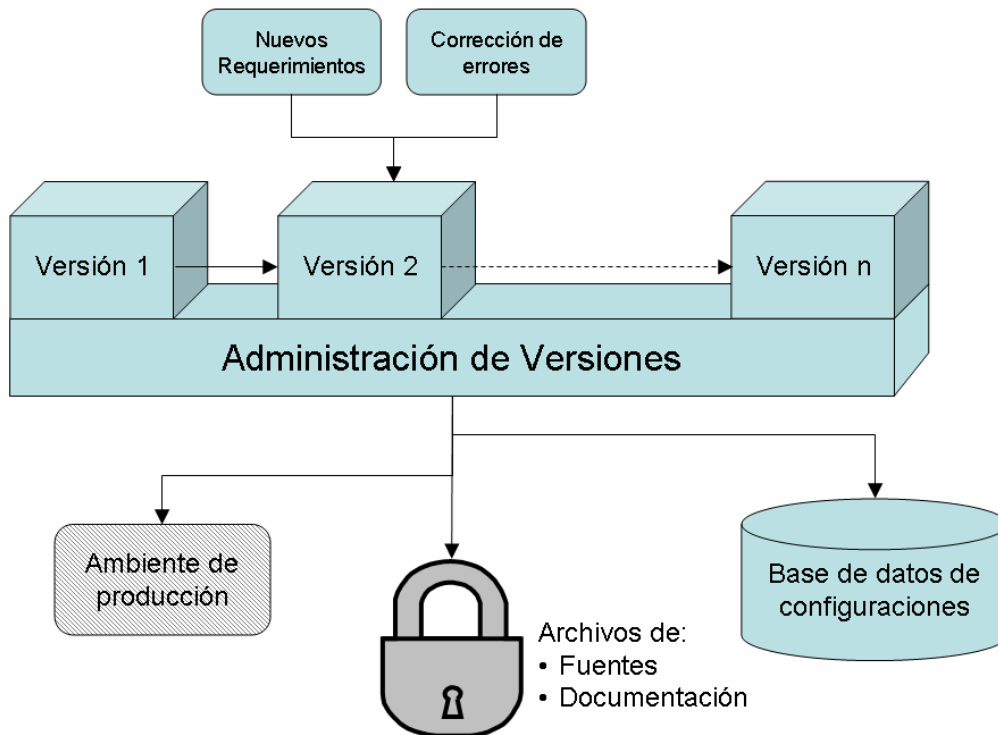
Una versión –o release- es un conjunto de ítems de configuración que han sido probados y están listos para ser migrados al ambiente de producción.

La administración de versiones es una disciplina que debe coordinarse e integrarse perfectamente con las disciplinas de administración de cambios y de configuraciones, con el fin de asegurar que toda la información relacionada con las nuevas versiones se actualice adecuadamente en la base de datos de configuraciones.

La persona o el equipo responsable de la administración de versiones tiene como responsabilidad fundamentalísima la custodia de las librerías o bibliotecas de programas fuentes y ejecutables que han sido aceptados en el ambiente de producción, así como también, de las bibliotecas de documentación que deben acompañar dichos componentes.

La disciplina de administración de cambios se encarga de aprobar y supervisar todo el proceso de cambio y la administración de versiones se encarga del proceso de migración al ambiente de producción de los ítems

de configuración nuevos o modificados; obviamente, ambas disciplinas tienen áreas comunes que deben ser tomadas en cuenta al desarrollar los procedimientos, para evitar duplicidad de funciones o choque de responsabilidades. Por esta razón, en algunas organizaciones las actividades de administración de versiones se incluyen dentro de la administración de cambios.



Entre los principales objetivos de la administración de versiones podemos señalar los siguientes:

- Implementar las nuevas versiones de software en el ambiente de producción, una vez que se hayan realizado pruebas suficientemente rigurosas.
- Asegurar que el proceso de cambio cumpla las especificaciones de la solicitud de cambio que lo originó.
- Asegurar, en coordinación con la administración de cambios y de configuraciones, que todos los cambios se actualicen correctamente en la base de datos de configuraciones.
- Archivar y custodiar tanto los programas fuente, como copias de los ejecutables en producción, así como de toda su documentación asociada, en la biblioteca de *componentes en producción*.

- Mantener la custodia, dentro de los plazos que establezcan las políticas de la organización, de los ítems reemplazados –versiones anteriores-.

### ***1.1.- Ventajas***

Entre las ventajas que pueden derivarse de una disciplina de administración de versiones implementada correctamente pueden señalarse:

- Las nuevas versiones cumplen los objetivos que originaron su desarrollo.
- Se reduce el número de incidentes que puedan ocasionarse por incompatibilidades con otro software o hardware instalados.
- El proceso de pruebas que debe preceder la implantación, permite asegurar la calidad del software y hardware que se instala.
- El correcto mantenimiento de la bibliotecas de componentes originales -fuentes, ejecutables, documentación, etc.- evita que se pierdan valiosos elementos de la infraestructura.
- Se mantiene un control centralizado del software que pasa al ambiente de producción.

### ***1.2.- Barreras***

Entre las principales barreras que encuentra la implantación de la disciplina de administración de versiones pueden señalarse las siguientes:

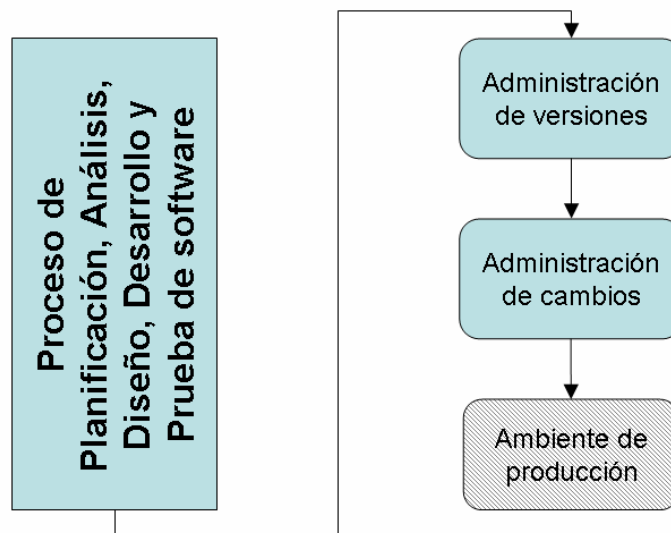
- Resistencia a la centralización del proceso de cambio.
- El personal y, en general, la organización TI no reconocen la autoridad del administrador de versiones, especialmente en lo relacionado a la custodia de los ítems instalados en el ambiente de producción.
- No se realizan pruebas lo suficientemente rigurosas de las nuevas versiones, lo cual genera emergencias en el ambiente de producción y la permanente necesidad de hacer excepciones en el cumplimiento de los procedimientos, con el fin de poder modificar los componentes directamente en ese ambiente.
- Hay resistencia a desarrollar planes de retorno o “backout”, para deshacer cualquier cambio que se haya implementado en producción, pero que, por cualquier razón, no esté funcionando adecuadamente.

- Carencia de herramientas de distribución de software, para la instalación de nuevas versiones que puedan requerir la instalación de uno o más componentes en diferentes servidores o en las estaciones de trabajo de los diferentes usuarios de una nueva versión.

## 2.- Consideraciones de tipo práctico

Señalábamos al inicio, que una versión es un conjunto de ítems de configuración nuevos o modificados, que han sido probados y están listos para ser puestos en producción, indicábamos también, que las especificaciones funcionales y técnicas de una versión deben estar establecidas en la solicitud de cambio que la originó.

En cierta forma, se puede afirmar que la administración de versiones es el último paso de los diferentes procesos de adquisición, desarrollo, mantenimiento e implantación de software, tanto software de aplicaciones para el apoyo del negocio, como software base, entendiendo como software base: los sistemas operativos, manejadores de bases de datos, compiladores, programas de utilidad, programas de automatización de oficinas, etc. Por ello, será importante establecer el conjunto de normas que deberán gobernar la implantación de nuevas versiones, con el fin de establecer unas sólidas metodologías de trabajo.



### 2.1.- Tipos de versiones

De acuerdo con su magnitud o impacto en la infraestructura de TI, las versiones pueden clasificarse en:

- Versiones mayores

Una versión mayor corresponde a la instalación de grandes piezas de software. Normalmente corresponde a un nuevo sistema operativo, aplicación o al reemplazo del software de apoyo a un número importante de operaciones del negocio.

- Versiones menores

Una versión menor corresponde a un grupo no muy grande de ítems de configuración, que complementan la capacidad de un elemento de software mayor o que cambian la funcionalidad de algún módulo de un sistema o corrigen varios errores conocidos.

- Versiones de emergencia

Una versión de emergencia corresponde a uno o varios ítems de configuración que reemplazan ítems en producción, para reparar un error conocido.

Será muy importante que los procedimientos de administración de versiones tomen en cuenta los diferentes tipos de versiones, con el fin de asegurar su agilidad, Evitando que para una versión de emergencia priven las mismas condiciones que para una versión mayor, sin perder de vista los controles fundamentales.

## ***2.2.- Actualización y custodia de componentes***

La administración de versiones debe establecer la forma cómo se mantendrán las diferentes librerías o bibliotecas de programas fuentes y ejecutables que quedan bajo la custodia del administrador de versiones. En el caso de las aplicaciones los componentes se almacenarán en bibliotecas propias de la instalación, cuyo acceso quedará restringido a los administradores de versiones únicamente; sin embargo, en el caso del software base y algunas aplicaciones adquiridas es preferible almacenar ordenadamente, en archivos de seguridad los ítems tal como fueron recibidos del proveedor.

Hoy día, es usual recibir vía internet nuevas versiones y correcciones de errores conocidos o, simplemente, actualizaciones; por lo que es muy importante establecer procedimientos que aprovechen la gran flexibilidad que esta forma de trabajo ofrece, pero que, además, establezcan suficientes controles sobre su aplicación.

Existen casos, como los antivirus, que se actualizan automáticamente, de forma continua. Si para estos componentes se puede tener la confianza de que el proveedor realiza una buena administración de versiones, no valdrá la pena invertir tiempo y energía para ponerlos bajo nuestra

administración de versiones. En estos casos, bastará con registrar el software en la base de datos de configuraciones y auditar periódicamente su proceso de actualización, para verificar que se está cumpliendo satisfactoriamente.

También es frecuente que se autorice a un proveedor a que tenga acceso a nuestra infraestructura, con el fin de que directamente, desde sus oficinas, pueda instalar ítems nuevos o modificados. En estos casos tampoco valdrá la pena invertir tiempo y energía para ponerlos bajo nuestra administración de versiones, bastará con llevar un registro detallado de estos eventos de actualización y auditar periódicamente el proceso de actualización, para verificar que se está cumpliendo satisfactoriamente.

### **2.3.- *Versiones anteriores***

En la administración de versiones es muy importante mantener una práctica ordenada para almacenar y custodiar las versiones que preceden a los componentes en producción –versiones anteriores-, con el fin de poder recuperar cualquier ítem, aunque haya transcurrido algún tiempo desde su reemplazo. Esta práctica resulta especialmente valiosa para los ítems de software, que a veces tienen alguna falla que ha pasado inadvertida en los procesos de prueba y que, tiempo después de haber sido instalados, causan una interrupción en el servicio o presentan un funcionamiento inadecuado.

### **2.4.- *Nomenclatura de versiones***

Una práctica común es la de establecer una codificación que identifique unívocamente cada versión. Normalmente, se utiliza un sistema de aceptación muy generalizada, que consiste en denominar:

- Versiones mayores: 1.0, 2.0, etc.
- Versiones menores: 1.1, 1.2, 1.3, etc.
- Versiones de emergencia: 1.1.1, 1.1.2, etc.

Igualmente, los ítems de configuración que integran una versión, deberán asociar su nombre o código de identificación con la versión a la cual pertenecen.

Cabe señalar que este esquema de denominación de versiones es apropiado para empresas productoras de software, en el caso de cualquier otra empresa, sólo será necesario identificar los ítems que conforman un sistema o una pieza grande de software e identificar las diferentes versiones a nivel componente.



### 2.5.- Estatus de una versión

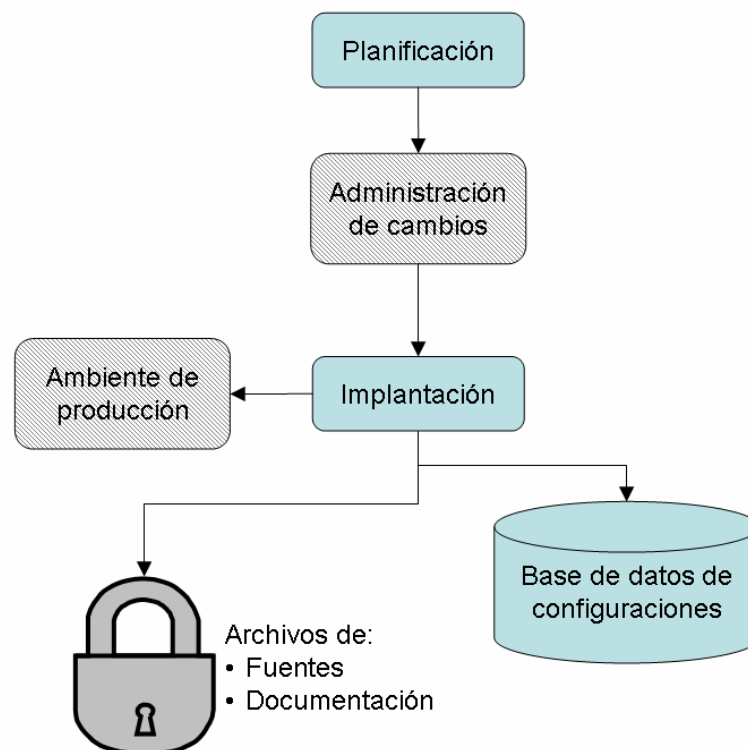
Las diferentes versiones cumplen un ciclo de vida, desde que se formula la solicitud de cambio que la origina, hasta que es reemplazada del ambiente de producción, por lo que hablamos de que una versión puede pasar por diferentes estatus:

- Solicitada
- En desarrollo
- En prueba
- En producción
- Descontinuada

Junto con la información pertinente de cada ítem de configuración, en la base de datos de configuraciones, será importante que se registre su estatus.

### 3.- Actividades de administración de versiones

Para dar inicio a la disciplina de administración de versiones, será importante establecer el conjunto de normas y políticas que deberán gobernar los procedimientos de implantación de nuevas versiones, con el fin de establecer unas sólidas metodologías de trabajo.



Adicionalmente, deberán establecerse los procedimientos que deberán seguirse para cumplir las siguientes actividades, que conforman la disciplina:

- Planificación
- Verificación de cumplimiento de estándares
- Verificación de cumplimiento de pruebas
- Implementación
- Soporte inicial

### **3.1.- Planificación**

Antes de instalar una nueva versión en producción se requiere formular varios planes, dependiendo del tipo de versión -mayor, menor o de emergencia-. Estos planes formarán parte del material que será revisará por la administración de cambios y deberán tomar en cuenta los siguientes aspectos:

- Alcance, contenido, riesgos, responsabilidades y usuarios involucrados en la versión.
- Estrategia para obtener la participación de todas las partes involucradas –unidades de soporte, analistas de sistemas, técnicos de bases de datos, usuarios, etc.-
- Planes de retorno –backout- para las situaciones de falla. Verificar que los planes de retorno o respaldo y de retorno o backout han sido convenientemente preparados y que son totalmente viables.
- Establecer quién, cómo y cuándo tomará la decisión de aplicar los planes de retorno –en caso de ser necesario-, para minimizar el impacto negativo sobre el servicio.
- ¿Qué se va a instalar? Establecer qué ítems de configuración están involucrados en la nueva versión.
- ¿Quiénes son los usuarios?
- ¿Dónde están de los usuarios? En el caso de distribuciones internacionales, ¿cuales consideraciones deben ser hechas?
- ¿Cuáles son las principales dificultades que se prevén?
- ¿Cómo serán entregados los componentes de la nueva versión?
- ¿Cuál es el impacto que puede tener la instalación de la nueva versión en la continuidad y calidad de los servicios?

- Definir cuáles son los recursos humanos y técnicos necesarios para llevar a cabo la implementación de la nueva versión con éxito.
- Establecer quiénes serán los responsables directos en las diferentes etapas del proceso de migración.
- Verificar que se hayan desarrollado planes de comunicación y adiestramiento para que los usuarios participen y estén debidamente informados, con el fin de evitar que una nueva versión llegue como una sorpresa.

### ***3.2.- Verificación de cumplimiento de estándares***

La administración de versiones constituye la “recta final” de los procesos de desarrollo y mantenimiento. En algunas oportunidades estos desarrollos se realizan "en casa" y en otras se realizan con la participación de proveedores externos. En ambos casos, la primera tarea de la administración de versiones será la de asegurar que el paquete o paquetes de software cumplan con las definiciones de línea base y estándares. En algunas empresas existe una unidad que cumple las funciones de aseguramiento de calidad, encargada de vigilar que durante todo el proceso de desarrollo se cumplan los estándares. Si ese fuera el caso, estas funciones estarán fuera del proceso de administración de versiones y únicamente se mantendrá la tarea de verificar que las sesiones de revisión de calidad se hayan cumplido satisfactoriamente.

La disciplina de administración de versiones también incluirá las tareas correspondientes a la actualización de la base de datos de configuraciones.

### ***3.3.- Verificación de cumplimiento de pruebas***

Una nueva versión, antes de pasar a producción, debe cumplir una serie de procesos de prueba, como los que se describen en los siguientes párrafos

Normalmente, los procesos de prueba corresponden a las metodologías de desarrollo y mantenimiento que utiliza la empresa, por lo que no corresponden a las actividades de administración de versiones; sin embargo, dentro de esta disciplina debe incluirse la tarea de verificar que dichas pruebas se hayan cumplido satisfactoriamente.

#### ***3.3.1.- Prueba unitaria***

Una prueba unitaria es una prueba que se hace de un solo programa o de un módulo. El programador del módulo es responsable de realizar la prueba unitaria en el ambiente de desarrollo.

### **3.3.2.- Prueba de integración**

Una prueba de integración es la prueba que se hace de las interfaces que existen entre diversos componentes dentro de un módulo, con el fin de detectar cualquier problema de intercambio de datos, archivos o parámetros y asegurar que pueden ser ejecutados en el orden o secuencia requeridos.

El analista del proyecto con ayuda de los restantes miembros del equipo de desarrollo o mantenimiento, realiza estas pruebas en el ambiente de prueba.

### **3.3.3.- Prueba Funcional**

El propósito de una Prueba Funcional es identificar las discrepancias que puedan existir entre un componente o sistema con sus especificaciones funcionales.

Estas pruebas las realizan el representante funcional en el equipo de trabajo, junto con el analista del proyecto y la ayuda de los restantes miembros del equipo de desarrollo o mantenimiento. Una prueba funcional se lleva a cabo en el ambiente de prueba.

### **3.3.4.- Prueba de Sistema**

La Prueba de Sistema es el complemento de la prueba funcional, está dirigida a probar los aspectos técnicos del sistema para detectar discrepancias con respecto a sus objetivos de desempeño como tiempo de respuesta, cumplimiento de estándares. Estas pruebas las ejecuta el analista responsable del proyecto con la ayuda de los restantes miembros del equipo y la supervisión de personal de soporte técnico.

Una prueba de sistema se lleva a cabo en el ambiente de prueba.

### **3.3.5.- Prueba de Aceptación Técnica**

Una Prueba de Aceptación Técnica es un proceso de prueba llevado a cabo por el personal técnico distinto del personal que desarrolló el sistema; en algunas empresas se cuenta con grupos dedicados realizar esta clase de pruebas para todos los sistemas que se desarrollan o modifican, antes de que los mismos pasen a producción. Sin embargo, para cumplir con esas tareas, la mayoría de las empresas prefiere organizar grupos de prueba ad-hoc con personal de operaciones y de soporte técnico. Normalmente, en una prueba de aceptación técnica se someten los sistemas a condiciones extremas, con el fin de asegurar que el sistema funcionará satisfactoriamente en cualquier caso.

## ***4.- Implementación***

Una vez que se ha establecido la coordinación con el proceso de administración de cambios, se procede con la implantación en producción. Este proceso puede ser:

- Completa
- Parcial

A su vez, una vez cumplidos los procesos de implantación, la administración de versiones debe asegurarse de que:

- Se incluyan los programas fuente, la documentación y las copias de los programas ejecutables, en las bibliotecas correspondientes-
- La base de datos de configuración quede correctamente actualizada.
- Se informe debidamente a la unidad de atención a los usuarios.

### ***4.1.1.- Implementación completa***

Cuando se realiza instala una implantación completa, se instalan todos los ítems que componen la versión simultáneamente para todos los usuarios, en todas las localidades.

Normalmente, para este tipo de implementaciones se establece un período de prueba final, denominado de aceptación funcional. Este período de prueba lo cumplen los usuarios, con el personal de soporte y el personal de operaciones, para evaluar que el sistema se desempeña adecuadamente trabajando bajo condiciones reales.

Algunas veces estas pruebas pueden incluir el cumplimiento de paralelos. Probar un sistema nuevo en paralelo con el sistema vigente, implica la instalación total del sistema en el ambiente de producción, su ejecución con datos reales y la comparación de sus resultados con los que produce el sistema vigente.

Una vez que estos procesos de aceptación se cumplen, el nuevo sistema pasa a ser el sistema oficial y se desinstalan todos los componentes que correspondan al sistema que se reemplaza.

### ***4.1.2.- Implementación parcial***

Muchas veces se realiza lo que se denomina instalación piloto, para asegurar que el sistema se comporta adecuadamente en el ambiente de producción. También, si los usuarios se encuentran dispersos en localidades muy distantes y existen diferencias importantes de horario, se prefiere ir cubriendo grupos de usuarios separadamente. En estos casos,

los usuarios finales deben estar informados del calendario de instalación y la forma cómo podrían afectarse sus actividades diarias.

### ***5.- Soporte inicial***

Normalmente, en el inicio de un nuevo servicio pueden haber errores que hayan pasado desapercibidos por los diferentes procesos de prueba, por lo que debe considerarse que las versiones que están cumpliendo el período de aceptación pudieran requerir la intervención de algún miembro del personal de desarrollo o del personal de soporte, para ello los procedimientos deben contemplar la posibilidad de otorgar autorizaciones temporales, para que estos técnicos puedan cumplir su cometido, con agilidad.

Será muy importante para evaluar el desempeño de los equipos de desarrollo que, aunque estas tareas de soporte inicial se cumplan flexiblemente, queden bien registradas bajo las disciplinas de manejo de incidentes y de problemas.

### ***6.- Evaluación de la disciplina***

Para evaluar el desempeño de la disciplina de administración de versiones será importante generar informes periódicos que ofrezcan una información precisa de las acciones cumplidas y presenten una métrica cubriendo aspectos como:

- Número de nuevas versiones implantadas, por categoría
- Número de procesos de retorno que se han tenido que realizar y razones que privaron para realizarlos.
- Incidentes asociadas a nuevas versiones.
- Uso de recursos en cada caso.
- Existencia de versiones ilegales de software.
- Adecuado registro de las nuevas versiones en la base de datos de configuración.
- Incidentes provocados por un uso incorrecto de la nueva versión por parte de los usuarios.
- Disponibilidad del servicio durante y después del proceso de implantación de la nueva versión.