

CMDB como columna vertebral de ITIL

En la Consejería de Salud de la Junta de Andalucía

Introducción

Desde finales del 2006 empezamos a cuestionar el modelo de trabajo que estábamos usando al detectar varios problemas:

1 - El tiempo que se requería para realizar un sistemas de información y posteriormente al tener que mantenerlo crecía exponencialmente y llegábamos a tener grandes insuficiencias debido fundamentalmente a:

- Los entregables no correspondían con lo realmente implantado.
- La documentación era escasa o no correspondía con lo implantado.

2 - No teníamos un inventario de nuestros sistemas de información, servidores, software, etc en definitiva sobre nuestra configuración, lo que nos provocaba duplicación de configuraciones y sobre todo no teníamos información para poder tomar decisiones estratégicas.

3 - No podíamos gestionar el tiempo de resolución de incidentes, este problema se detecto con anterioridad y ya se estaba trabajando en su resolución.

Con esta problemática nos planteamos cambiar el modelo de trabajo. Para ello se contrató una asistencia técnica con el fin de definir e implantar un sistema de aseguramiento de la calidad cuya primera misión sería plantear un modelo de negocio que pudiéramos seguir. Después de varios meses estudiando nuestro sistema de trabajo nos presentaron el modelo a seguir basado en ITIL como recomendaciones de buena practicas, explicando que era el modelo que estaban siguiendo las empresas TIC.

Primer contacto con ITIL

Desde la Oficina de Calidad se impartieron varios seminarios planteando ITIL ,llegando entre todos al convencimiento que:

- Se ajustaba a nuestra forma de trabajar.
- Se podía implantar de formar gradual, dado que quedo claro que era imposible abarcar toda la implantación a la vez.
- Se aprovechaba todo lo trabajado anteriormente con la gestión de incidentes de microinformática.
- Existían pocas implantaciones (accesibles por nosotros) de las cuales pudiéramos recopilar información antes de empezar.
- Las herramientas existentes eran caras y demasiado rígidas.

Igualmente cada uno de los responsables de área se vio reflejado en alguno de los módulos de ITIL:

- Sistemas con la gestión de la configuración.
- Sistemas de información con la gestión de cambios.
- Microinformática con la gestión de incidentes.
- hDirección con la toma de decisiones.

Tomando la decisión de empezar con estos módulos, a la oficina de calidad se le encomendó como segundo hito la búsqueda de herramientas que nos dieran la posibilidad de implantar estos módulos. Como resultado se nos planteo una serie de herramientas que permitían entre otras características la implantación gradual de ITIL y poder comprar los módulos según se fuera avanzando en la implantación, las herramientas de software libre que encontramos no abarcaban las gestión completamente y se especializaban sólo en determinados aspectos no consiguiendo relacionarlas entre si, ninguna de ellas conseguía ajustarse a nuestro presupuesto llegados a este punto tomamos la decisión de implantar la gestión de incidentes, cambios y problemas y buscar otras posibilidades para la gestión de la configuración y toma de

decisiones.

Primer contacto con la CMDB

En la búsqueda de herramientas que cubriera la gestión de la configuración descubrimos que el componente principal era la Base de Datos de la Configuración la CMDB y todo los demás módulos se encargaban de consultar o actualizar esta base de datos. Estudiamos todo la información que pudimos sobre la CMDB la cual era escasa y muy teórica. No encontrábamos información de implantaciones de CMDB. Ante este panorama decidimos empezar a implantar nuestra propia CMDB usando herramientas de software libre con la finalidad de evitar gastos innecesarios, nos evitábamos las licencias y aumentábamos el número de horas que le podíamos dedicar al proyecto.

Definición del primer modelo a seguir

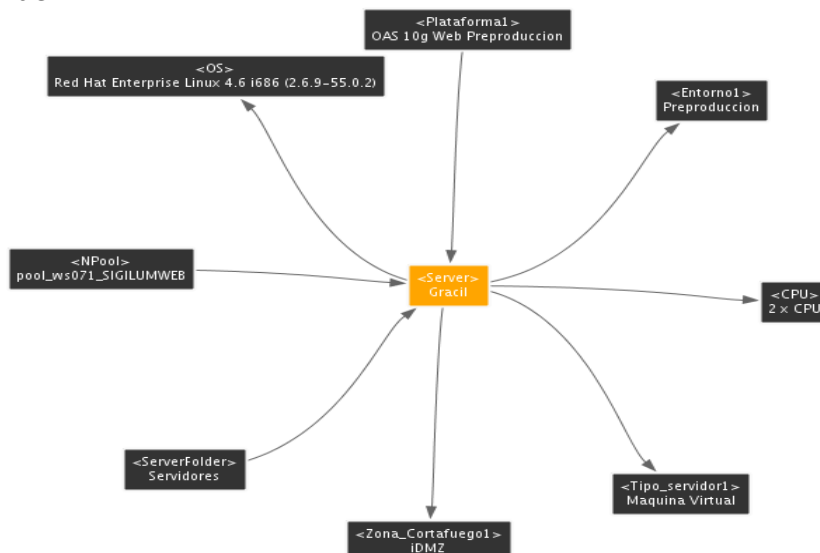
Con la información que habíamos recopilado abordamos la definición de un modelo teniendo en cuenta los siguientes aspectos:

- Variabilidad de la información: entendiendo el tiempo medio que tardaba la información en variar.
- Retorno del conocimiento: entendiendo el beneficio de tener esta información lo mas organizada posible.
- Origen o dueño de la información: quien mantiene dicha información y como se propaga o se anuncia.

Al realizar el proyecto desde el área de sistemas lo dirigimos al mantenimiento de la configuración de nuestro CPD. Los tres aspectos descritos se definían de la siguiente forma:

- Variabilidad; poca sólo en la adquisición de nuevo material.
- Retorno del conocimiento; esta información sólo es utilizables para averías técnicas o en caso de caída de la electricidad.
- Origen; el propio área de sistema mantiene esta información y la mantienen actualizada.

El modelo definido fue:



Ci's definidos:

- Servidor
- Tipo CPU
- Sistema Operativo
- Entorno
- Zona Cortafuego
- Tipo Servidores
- Armario (Rack)

- Circuitos eléctricos

Toda esta información la teníamos dispersa en hojas de cálculo y fue relativamente fácil recopilarla, con este modelo conseguimos:

- Inventario de servidores con número de serie, fecha de compra y fin garantía.
- Distribución de servidores por sistema operativo (licencias)
- Distribución de los servidores en el CPD
- Relación de armarios con circuitos eléctricos
- Distribución de servidores por entorno
- Distribución de servidores por zona cortafuego

Igualmente conseguimos responder rápidamente a las siguientes preguntas:

- ¿Servidores que terminaban la garantía?
- ¿Distribución de licencias por sistema operativo?
- ¿que servidores se veían afectados por caídas de red eléctrica?
- ¿que servidores son los más críticos (entorno producción)?
- ¿que servidores son vulnerables ante ataques de red (estar en la misma zona)?

Conclusiones del primer modelo

Las conclusiones que sacamos de este primer modelo fueron muy positivas, vimos que la herramienta elegida nos permitía una flexibilidad que, sabiéndola administrar, nos facilitaba modificar el modelo si detectábamos que podíamos abordar más información o incluso no introducir información en esta primera fase para introducirla a posteriori, las copias de seguridad se realizaban con la base de datos general, el rendimiento era bastante bueno.

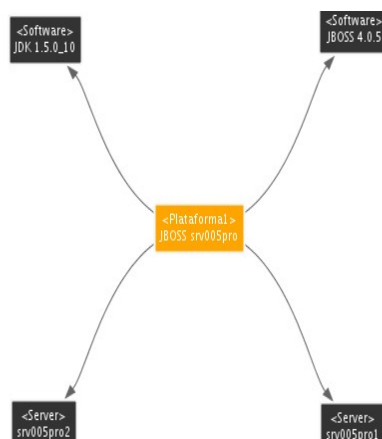
No disponíamos de una herramienta de informes pero la suplimos creando agrupaciones de ci's y jugando con los nombre de dichos ci's.

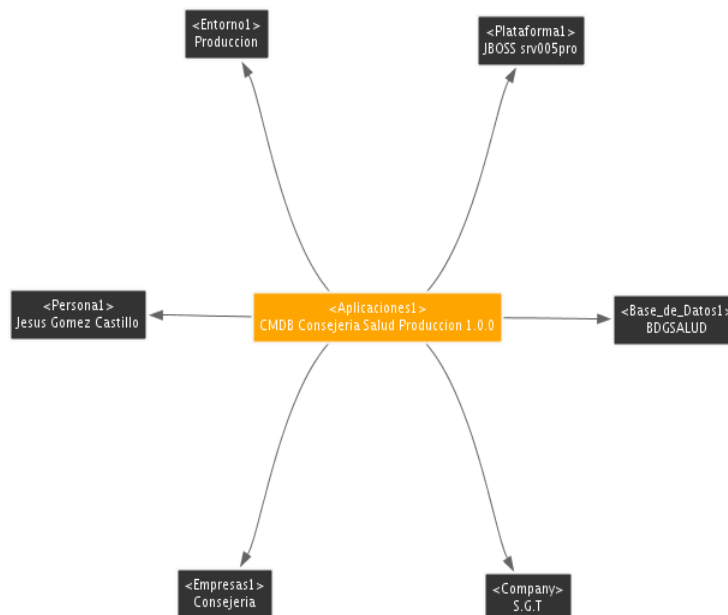
Lo más importante es que debíamos abordar la gestión de cambios si queríamos tener actualizada la CMDB y permitir consultar a los responsables toda la información.

La implantación de este modelo fue muy rápida en una semana lo teníamos finalizado, toda la información la teníamos previamente recopilada y el número de equipos a inventariar no era numeroso, se realizo por un técnico y en modo local

Definición del segundo modelo

Con todo lo aprendido, abordamos el reto de controlar las aplicaciones, que software base y servidores usaban y construimos los siguientes ci's para gestionarlos. Tuvimos que pensar la forma de guardar información más heterogénea y surgieron ci's compuesto sólo de relaciones. En este punto se nos presentó el mayor problema: la herramienta elegida sólo permitía un tipo de relación 'usada en' lo que simplificaba los modelos pero perdía claridad en su definición. Ante esto empezamos a trabajar con información no inventariada y que no dependía de nosotros, surgiendo los siguientes ci's:





Ci's definidos:

- Aplicación
- Responsable técnico
- Dirección General
- Plataforma
- Base de datos
- Empresas
- Software Base
- Servicios

Toda esta información se fue recopilando de los servidores directamente y fue verificada por los responsables técnicos. Al terminar esta fase teníamos un inventario de todas las aplicaciones y servicios, donde se encontraban y que características tenían. Nos encontramos varias aplicaciones que estaban en el entorno de producción pero conectadas a la base de datos de pre-producción (alguna prueba de rendimiento), aplicaciones que no se usaban y por último y más importante detectamos que había demasiadas aplicaciones sin finalizar.

Conseguimos dar un verdadero salto de calidad al tener todos los sistemas de información inventariados y controlados por versiones. De esta forma podíamos saber que servidores soportaban mayor carga de trabajo y por tanto eran más críticos, que aplicaciones se veían afectadas ante caída de un servidor, a quien se le asignaban los incidentes de las aplicaciones. Así mismo permitía a los responsables técnicos de las aplicaciones conocer exactamente las características de los servidores, donde estaban ejecutándose sus aplicaciones, indispensables para autorizar cambios teniendo en cuenta que para actualizar una aplicación es necesario parar todas las aplicaciones que estén ejecutándose en el mismo servidor. Por otra parte ha sido indispensable para aprovisionar nuevas plataformas: sabemos cuales tenemos y si es necesario crear una nueva.

Conclusiones del segundo modelo

Este modelo sólo es sostenible con la definición de unos procedimientos rigurosos de subidas de versiones donde se contemplen la actualizaciones de la CMDB y una gestión de cambios que los gestione, todo fue asumido y conseguimos una CMDB con un modelo estable consiguiendo la confianza de los responsables y personal técnico lo que nos permitió dar otro gran paso.

El mantenimiento de este modelo era asumible por el departamento de sistemas, era necesario seguir explicando el modelo al resto de departamentos e ir pidiendo que aportaran toda la información que quisieran. Llegando a una madurez en la CMDB tal que era consultada para cualquier duda antes de investigar más afondo.

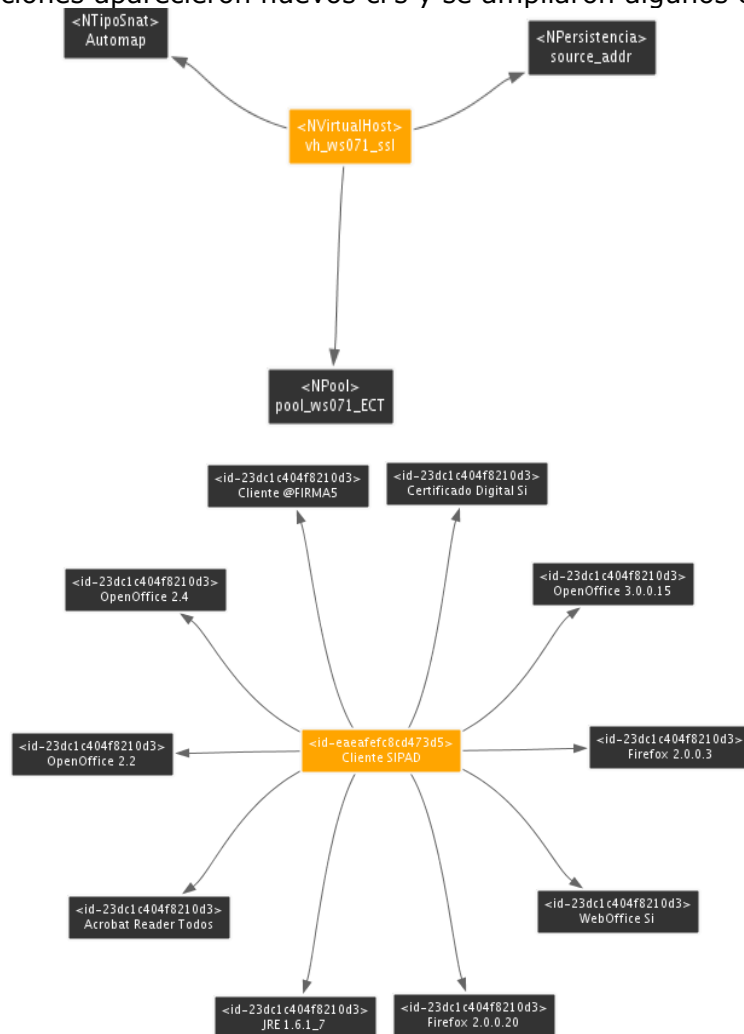
El tiempo necesario para implantar este modelo fue muy superior casi 2 meses debido

fundamentalmente a las dispersión de la información, no estaba recopilada previamente y había que mirarla sobre la marcha, se aprovecho para poner la CMDB centralizada se modificó para que usará nuestro modelo de arquitectura, el número de técnicos para este modelo fue uno para la definición y estudio (1 semana) y posteriormente se gestionó como incidencias y se resolvía por el departamento de sistemas (3 técnicos por turnos), en esta fase también se mantenía el modelo anterior.

Definición del tercer modelo

Toda la aportación que fueron haciendo el resto de áreas se estudiaron e incorporaron a la CMDB de esta manera se aporó la configuración de los balanceadores, se amplió la información de la base de datos con la ubicación física de los datos, se conectaron los sistemas de información entre ellos y con el exterior en definitiva de verdad se consiguió relacionar toda la información entre si y dar utilidad práctica a dicha información.

Con todas estas aportaciones aparecieron nuevos ci's y se ampliaron algunos existentes:



Ci's definidos:

- VirtualHost
- Tipo Persistencia
- Tipo Ámbito Acceso
- pool
- Tipo Snat
- Aplicaciones (ampliación del existente)
- Servidores (ampliación del existente)
- Base de Datos (ampliación del existente)
- Características especiales
- Definición cliente
- Incompatibilidades de clientes

Con todos estos ci's se amplia significativamente la información existente en la CMDB e implicamos a otras áreas en su mantenimiento.
Conseguimos tener un mapa exacto de los elementos que intervienen en los sistemas de información.

Este modelo se alargó en el tiempo al modificarlo según lo demandaban las otras áreas, si bien los técnicos de sistemas que conocen como administrar la CMDB ha subido a tres y se ha montado una CMDB de prueba donde se verifican estas nuevas aportaciones.

Conclusiones finales

La madurez de la CMDB ha convencido al resto de áreas llegando a hablar entre nosotros de elementos de la CMDB y abstrayendo los aspectos técnicos necesarios para realizar determinadas tareas, por ejemplo:

- Anteriormente se solicitaba acceso de la ip1 a la ip2 por el puerto P1, para saber la ip1, ip2 y P1 debían llamar a los técnicos o mirar directamente en los servidores esta información.
 - Actualmente; se solicita acceso de la aplicación AP1 (buscamos en la CMDB la aplicación y sacamos las ip's de los servidores que la alojan) al servicio SERV1 (buscamos en la CMDB el servicio1 y sacamos las ip's de los servidores y el puerto que esta usando)
- En definitiva hacemos más transparente toda esta información a la organización y evitamos incidentes para averiguar información.

Para la toma de decisiones es fundamental, aprobación de cambios:

- Anteriormente; para saber que implicaciones tenía el parar una máquina teníamos que entrar en ella y mirar exactamente que servicios o aplicaciones se veían afectados, no éramos capaces de saber si este servicio afectaba a otros.
- Actualmente; fácilmente identificamos los servicios y aplicaciones que se ven afectadas y sobre todo el resto de aplicaciones que se verían afectadas ante esta parada.

Para la gestión de la capacidad igualmente es indispensable sabemos exactamente que configuración tenemos y si es necesario adquirir nuevos elementos.

- Anteriormente; solicitaban un servidor con unos elementos y se tenía que crear nuevo o revisar cada uno de los existentes para saber si era igual al solicitado.
- Actualmente; con unos pocos pasos sabemos si tenemos una configuración igual a la solicitada.

La implantación de la CMDB coincidió con el cambio estratégico que se produjo hacia la virtualización lo que nos permitió un control exhaustivo de todos los activos de la organización imposible de abordar sin este control, en dos años hemos pasado de poco más de 40 servidores a más de 200.

Beneficios para la Organización

La instauración de la CMDB nos ha dirigido hacia una forma de trabajar distinta algunos de los cambios que hemos detectado son:

- La gestión de la monitorización con la CMDB es más intuitiva, al saltar una alerta en el monitor rápidamente sabemos todos los elementos que se ven afectados.
- En la gestión de incidentes los técnicos tienen disponible toda la información de la organización, actualmente saben cada elemento de la configuración de quien depende y se le asocia la incidente.

- Aseguramiento: Permite conocer los activos a tratar puntualmente, incluso su nivel de evolución y hasta se podría indicar si se encuentra asegurado o no.
- .- Sostenibilidad: Es importante conocer todo nuestro parque para optimizar su existencia y generar ahorros. Obviamente permite gestionar mucho mejor nuestros recursos de cualquier tipo.
- .- Conocimiento del Sistema: es un valor intrínseco (en sí mismo ha de tenerse en cuenta). Todo lo que hagamos debe partir de aquí para que llegue a fructificar.
- .- Proyectiva: permite conocer hacia donde evolucionar y servir información a entidades mayores (Junta, Innovación) sobre nuestra situación, que sea fiable y veraz.
- .- Calidad de Servicio: Indiscutiblemente se ha convertido en el pilar para mantener niveles de servicio de alta calidad pues nos permite en todo momento "saber como estamos".

Por último toda la organización asume que lo de verdad implantado es lo que aparece en la CMDB y ve reflejado su trabajo en continuas modificaciones sobre ella, subidas de versiones de sistemas de información, adquisición de equipos, aumento del rendimiento de las aplicaciones, movilidad de técnicos, conexión entre sistemas de información... la columna vertebral de la organización por si sola no puede hacer nada pero sin ella no se conectarían el resto de áreas.

Algunas referencias

Las herramientas utilizadas han sido:

- Para la gestión de la configuración: OneCMDB versión 1.2 (software libre)
- Para la gestión de incidentes, cambios, problemas: BMC ServiceDesk Express (software propietario)
- Para la gestión de la monitorización: Nagios personalizado a nuestro entorno (software libre)
- Bases de Datos: Oracle 10G
- Para el control medio ambiental: equipos de BlackBox (equipo propietario)

Los plazos de la implantación han sido:

- Primera fase: 1 semana y mantenimiento desde entonces
- Segunda fase: 2 meses y mantenimiento desde entonces
- Tercera fase: gradual a lo largo del proyecto

Algunos indicadores de referencia:

•nº de Ci en la primera fase	500
•nº de Ci en la segunda fase	1500
•nº de Ci en la tercera fase (actual):	2000
•nº de incidentes en el último año (2.009):	10.850
•nº de cambios en el último año (2.009):	657