
Artículo

[Nancy Martínez](#) · 23 feb, 2022 Lectura de 5 min

Receta para migrar un servidor de Ensemble a IRIS

Os voy a contar cómo hacer una migración de Ensemble a IRIS para chuparse los dedos, es una receta antigua que me pasó mi abuela pero yo le he puesto algunos detallitos. Aquí os la dejo para que cuando os toque alguna podáis tenerlo paso a paso.

Preparar los Nuevos servidores

Lo primero de todo es instalar y configurar la nueva instancia de IRIS, aquí prestamos especial atención a dimensionar adecuadamente el servidor. Atención a la memoria y al espacio en disco para el sistema, para los journals, para las bases de datos y también para los backups. También tened en cuenta si vais a acceder desde puestos remotos en habilitar los puertos adecuados de los firewalls, normalmente usaremos el 22 si hacemos SSH, el 1972 si usamos el Studio y el puerto 52773 para acceder al portal. Aunque si no os gustan esos puertos podeis poner otros. Ya sabeis que la cocina hay que mantenerla siempre limpia y ordenada!

Instalamos IRIS

Pues nada, instalamos IRIS y luego configuramos Global buffers y Routine buffers. Ya sabeis que eso es más importante y casi lo único a configurar para que la cocción salga en su punto. Otra cosa a tocar es la configuración de journals, para evitar que el principal y el alternativo sean el mismo directorio. Así IRIS se queda más tranquilo y nos evitamos algunos warnings que luego salpica todo y aquí que usar el quitagrasas!. Alguna cosilla más que tocar, eliminar la opción de desactivar cuenta después de 90 días de inactividad, porque a veces se queda mucho tiempo sin acceder y te encuentras que no hay manera de entrar. Y a ver quien abre el horno a 200º !!

Tomar nota de todo

En la cocina hay que apuntar todos los ingredientes y ahora toca hacer el catalogo de Namespaces y producciones

del servidor que vamos a migrar, anotamos que producciones están en marcha y cuales no y le echamos un vistazo a cada una. No está de más algún pantallazo de algunos detalles para recordar y tomar notas. Que luego la foto del making-of queda muy chula!

Exportar, exportar

Exportamos las producciones y todas las clases que sean de usuario dentro de un Namespace. Eso lo podemos hacer desde el Studio o incluso desde el portal. Al final la idea es tener un paquete XML con todas las clases necesarias de cada Namespace que despues dejaremos en remojo al menos una noche.

Replicar, replicar

Mientras tenemos las clases en remojo, en el nuevo servidor creamos las bases de datos y atenderemos al recurso que se usa, al directorio donde se crea y al nombre de cada una. Espolvorear con sal marina y un poco de aceite.

Ya es el turno de la creación de los Namespaces, esto es fácil, le ponemos un nombre le asociamos las bases de datos creadas y que no se nos olvide que el check para habilitar la interoperabilidad del namespace esté activado

para que podamos importar las producciones sin problema. Lo metemos todo en una olla de tamaño grande. Más cositas, ¿qué tal unas aplicaciones Web? pican un poco pero a veces hay aplicaciones web adicionales, tenedlo en cuenta al revisar las producciones del servidor original.

Esto va a gusto del consumidor. Otros ingredientes que debemos tener en cuenta son:

SQL Gateways

Esto es algo muy común en producciones que acceden a bases de datos externas y usan JDBC. Para ello lo primero a tener en cuenta es que en IRIS la versión mínima soportada de JAVA es la 1.8. Podemos usar OpenJDK, por ejemplo para un sistema Linux RedHat podemos instalar OpenJDK11 con un solo comando:

```
sudo yum install java-11-openjdk-devel
```

Y tambien es importante obtener los driver JDBC necesarios y colocarlos en una carpeta adecuada. No olvideis descongelarlos con tiempo.

Ahora nos toca exportar todos los SQL Gateways del servidor original. Esto se puede hacer exportando la tabla %Library.sysSQLConnection como un fichero de texto tipo CSV desde el portal. Es recomendable usar un carácter separador de campos fácilmente reconocible como el |. Y tomad nota de las columnas porque entre versiones el número de columnas o campos de la clase puede cambiar. Normalmente si hay campos nuevos no son obligatorios o necesarios.

Configuraciones SSL

Otros ingredientes comunes son las configuraciones SSL, sobre todo si se usan comunicaciones seguras mediante HTTPS para servicios REST o SOAP. Para ello lo primero copiar los certificados existentes del servidor original y moverlos al nuevo servidor si es posible en las mismas carpetas, sino recordad que luego tendremos que tocarlos a mano.

Para exportar desde el antiguo servidor usaremos:

```
%SYS>w ##class(Security.SSLConfigs).Export("YYMMDD_SSLConfigsExport.xml")
```

Y para importar en el nuevo:

```
%SYS>w ##class(Security.SSLConfigs).Import("YYMMDD_SSLConfigsExport.xml",.NumImported
)
```

Podemos comprobar cuantos se han importado con la variable que hemos pasado por referencia NumImported.

Credenciales

Estas son un rollo, se pueden hacer al ajillo o salteadas pero no he encontrado como exportarlas y se deben hacer a mano. Así que nos toca preguntar las passwords de cada una de ellas. Si encontrais como automatizar esta tarea ponedlo en los comentarios ;-)

Default Settings

Otro ingrediente que aparece por ahí. Hay que revisar y tenerlos en cuenta porque son útiles y si no lo movemos tendremos un problema. Podemos exportarlos mediante:

```
MYNAMESPACE>w ##class(Ens.Config.DefaultSettings).%Export("YYMMDD_MYNAMESPACE_DefaultSettings.xml")
```

Ojo que esto debeis hacerlo por cada namespace!!. Podeis Importarlos mediante:

```
MYNAMESPACE>w ##class(Ens.Config.DefaultSettings).%Import("YYMMDD_MYNAMESPACE_DefaultSettings.xml",.NumImported)
```

Igualmente con NumImported podemos comprobar cuantos settings se han importado.

Importar las clases y compilar

Esto es fácil, vamos por cada namespace importando el fichero XML que exportamos y cruzamos los dedos para que todo vaya bien. De vez en cuando algo no compila bien y toca corregir algún detalle, lo probamos con una cucharita pequeña y con cuidado de no quemarnos. Luego es útil hacer un `$system.OBJ.CompileAll()` para que así se compile todas las clases dependientes o las que se quedaron sin compilar. Y quedan deliciosas!

Una vez que tenemos esto lo ideal es probar las producciones de manera individual, pero solo un poco que hay que dejar algo para los comensales. Comprobad que todo funciona como debiera siempre con entornos de prueba y no con la vajilla nueva. Esto es lo más pesado y además tendremos que molestar a más gente para validar que la integración hace lo mismo que antes.

Y para terminar postre

Por ejemplo reproducir búsquedas guardadas de mensajes si es que había y sobre todo reproducir las tareas programadas que existiesen en el servidor original. Podemos reducirlas con Pedro Ximenez y quedan bien dulces. Reproducir la configuración de Backups, borrado de journals y purgados de mensajerías. Y si seguís usando el Studio, que para mi gusto repite un montón, no olvidar actualizar instalaciones de las máquinas de desarrollo para que puedan acceder a la nueva instancia.

Pues nada, ya teneis una migración rica rica y con fundamento!!

[#Ensemble](#) [#InterSystems](#) [IRIS](#)

URL de fuente: <https://es.community.intersystems.com/post/receta-para-migrar-un-servidor-de-ensemble-iris>