# Revisión Arquitectura Dynamics AX

## Consultas iniciales

### Bloqueos de Dynamics AX

¿Cómo detectar si un bloqueo del aplicativo se debe a la plataforma de Dynamics AX o a la línea a través de la cual los usuarios acceden a su cliente mediante Terminal Server?

El cliente de Dynamics AX requiere conexión constante con el servidor de aplicación (AOS). Si ésta conexión sufre un corte por pequeño que sea el usuario será expulsado de la sesión.

Si el cliente muestra el mensaje “Processing” y después de hacerle un click en cualquier lugar queda gris en modo “No Responde” se deberá a que aunque no haya perdido la conexión con el servidor de aplicación está esperando una respuesta por su parte.

Si el cliente queda completamente bloqueado y el cursor del usuario inmóvil, se deberá a una interrupción en la comunicación del usuario con el servidor de Terminal Server.

Dado que la Es importante monitorizar los bloqueos de la BBDD de Dynamics AX para ver si

### Personalizaciones por Filial

Comentamos que la forma más apropiada de poder controlar desarrollos que deben aplicar o no a diversas filiales que compartan el mismo aplicativo pasa por utilizar las claves de configuración estándares de Dynamics AX. De éste modo, cuando se realice un desarrollo, deberemos discernir entre dos escenarios:

1. Se trata de un desarrollo que aplicará al Core (Entendiendo como Core de la plataforma actual una Filial Comercial Genérica y no los diversos tipos de filiales existentes en la actualidad.
   1. Todo desarrollo realizado deberá estar activo o no dependiendo de la clave de configuración CORE COMERCIAL.
2. Se trata de un desarrollo que aplicará a una única filial en concreto dentro del modelo de Filial Comercial Genérica:
   1. Todo desarrollo realizado deberá estar activo o no dependiendo de una clave de configuración creada específicamente para el proceso en concreto de la filial al que estará dedicado, y siguiendo una nomenclatura normalizada.

Siguiendo éstos conceptos los desarrollos ya existentes deberán ser adaptados para incluirse en la estructura especificada.

## Situación Actual

Se ha revisado la infraestructura de servidores actual de Dynamics AX y las versiones del aplicativo en uso para el entorno de producción.

### Entornos

#### Producción

* [**SV00082**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00082)
  + AOS
* [**SV00084**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00084)
  + SSRS
  + SSAS
* [**CL00008**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=CL00008)
  + *CL00008-SQL*
    - SHAREPOINT DB
  + *CL00008-SQL2*
    - DATA DB
    - MODEL DB
* [**SV00087**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00087)
  + SHAREPOINT
  + ENTERPRISE PORTAL

#### Test

* [**SV00052**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00052)
  + AOS
* [**SV00053**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00053)
  + SSRS
* [**SV00083**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=SV00083)
  + *SV00083*
    - DATA DB
    - SSAS DB
  + *SV00083-SQL2*
    - SHAREPOINT DB

#### Desarrollo

* ¿??

### Servidores TS

#### Producción

* [**TS00022**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=TS00022)

#### Test

* [**TS00025**](http://maintenance.corp.solerpalau.com/informe.asp?reportname=Sysinfo&servername=TS00025)

### Versiones de Software

#### *Kernel*

* 6.0.1108.4316 – 2012 CU 5

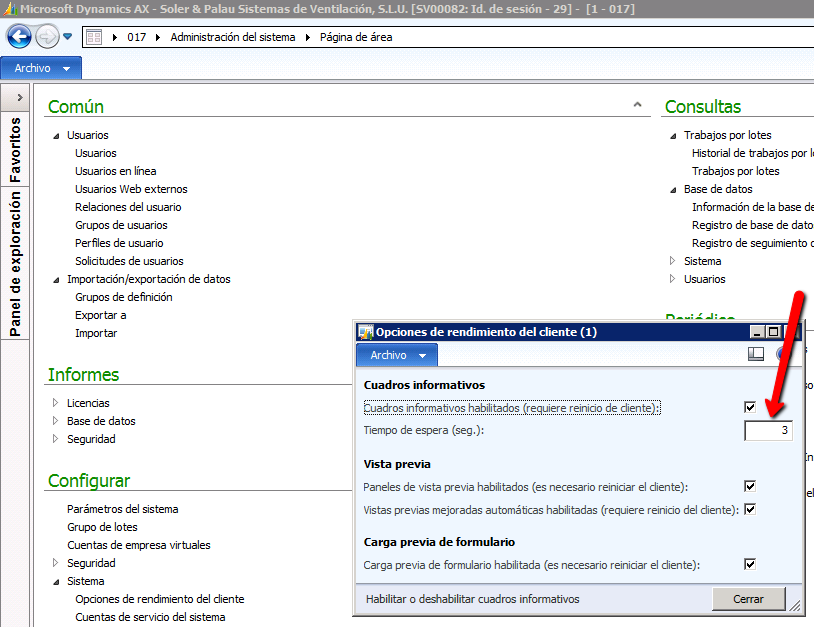
#### Application

* 6.0.1108.4316 – 2012 CU 5

## Estado General de la Plataforma

### Rendimiento de interfaz

Se ha establecido un límite de tiempo de espera a los cuadros informativos que aparecen dentro del aplicativo:



Queda pendiente estudiar la necesidad de mostrar los cuadros informativos y los paneles de vista previa. En principio son una ayuda y un resumen visual fácil de utilizar por los usuarios pero penalizan el rendimiento de la apertura de formularios en Dynamics AX, por lo que si no se están utilizando para ningún proceso en concreto o la información que contienen puede ser examinada de otra forma sin complicar el proceso en exceso, es aconsejable desactivarlos.

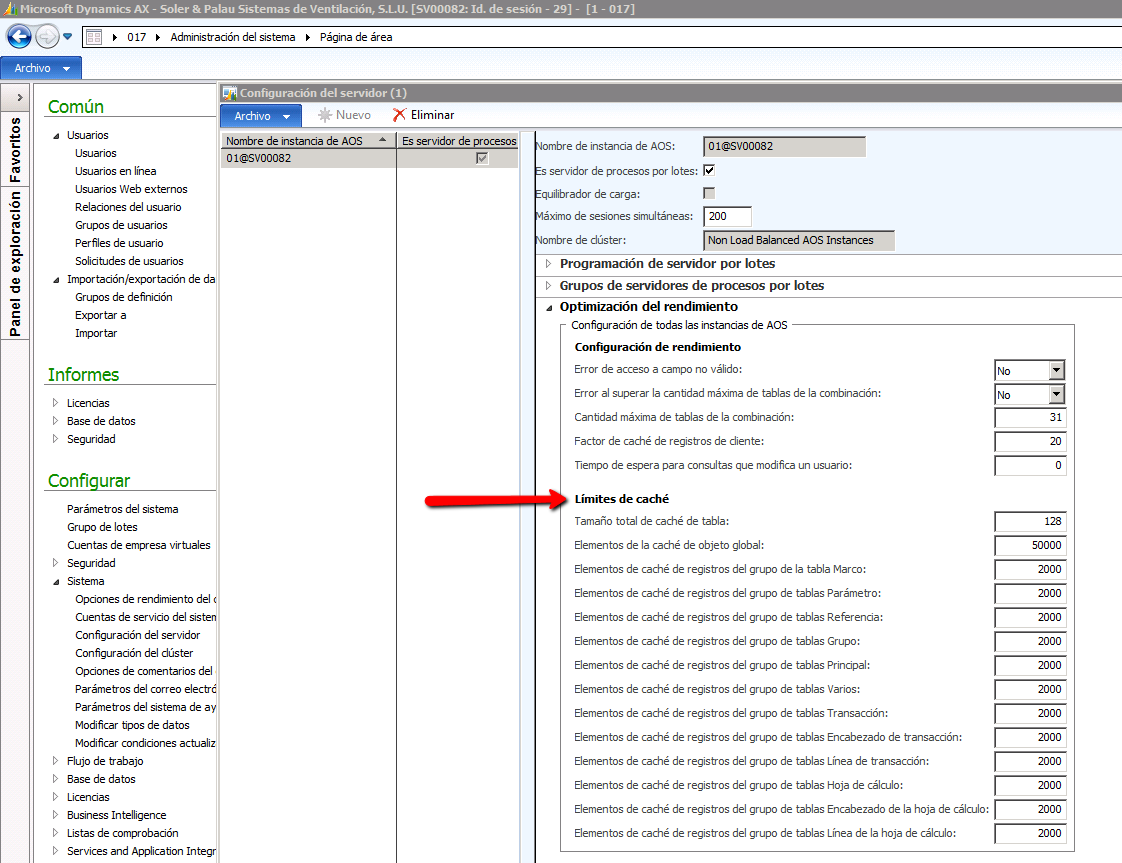
### Alta disponibilidad de AOS

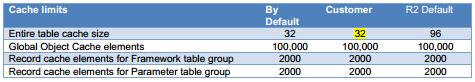
Actualmente no disponemos de alta disponibilidad del servidor AOS, deberemos corregir la situación mediante el plan de Disaster Recovery.

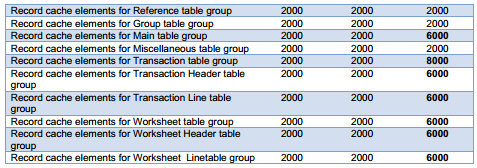
El servidor AOS tiene un total de 8 núcleos de CPU, un solo servidor se está encargando de la totalidad de los procesos. En el caso específico de los procesos programados por lotes aunque es recomendable tenerlos separados en un servidor AOS dedicado, no se observan problemas de rendimiento debido a la poca cantidad e intensidad de procesamiento que requieren los procesos existentes.

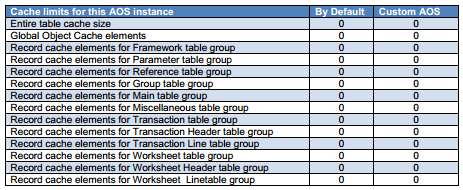
### Límites de caché por tipo de tabla

Se observa que los límites de cache establecidos por tipo de tabla no son los óptimos (establecidos por defecto al actualizar a Dynamics AX 2012 R2) Por lo que se recomienda actualizar los valores actuales a los valores especificados en el documento de auditoría de Marzo del 2014.



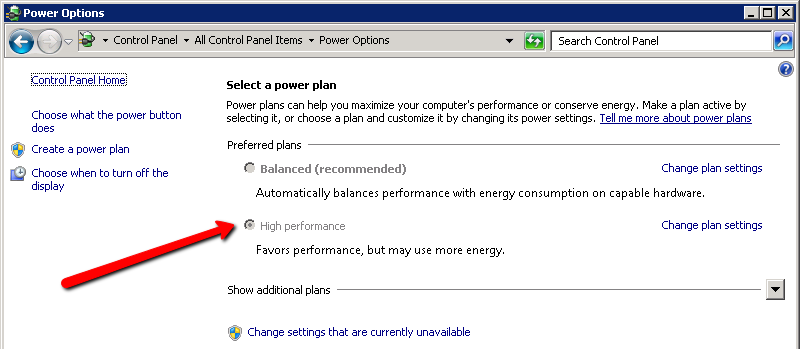






### Opciones de Energía

Debemos asegurar que la opción High Performance está activada en todos los servidores entre los cuales se reparten los servicios de la plataforma, ya sean físicos o virtuales. Esta opción la encontramos en el Panel de Control, Opciones de Energía:



### Flags activos en SQL

Los flags que tenemos activados en SQL actualmente son los siguientes:

**1117**

* Hace crecer todos los ficheros de la misma forma

**1118**

* Prevenir que en tempdb no se puedan utilizar extends mixtos
* Reservar memoria de SQL por sesión

**1224**

* Escala los bloqueos eliminando locks a nivel de tabla pero penaliza rendimiento
* Desactivar

**2371**

* Cambia cálculos de estadísticas de SQL
* Desactivar e integrar en plan de mantenimiento de SQL

**4136**

* Relacionado al anterior
* Trabajar siempre con estadísticas planas
* Desactivar e integrar en plan de mantenimiento de SQL

**4199**

* Mantener

### Recomendaciones

#### Instancia de SQL dedicada

La instancia de SQL donde se aloja la BBDD de Dynamics AX debería ser dedicada, actualmente está compartida con la BBDD de Sharepoint para Enterprise Portal. Si se activará la indización de Sharepoint el rendimiento de AX se vería muy afectado por la actividad de Sharepoint.

#### Desactivar opciones de SQL

Debemos consultar a Avanade si las integraciones de datos requieren las opciones de SQL Shared Memory y Named Pipes, en caso contrario debemos desactivarlas mediante el aplicativo SQL Configuration.

#### Eliminar a AXSQLServices de Administradores

Dando permisos específicos al usuario y eliminándolo del grupo de administradores aseguramos cubrir un agujero de seguridad que permitía acceso total al dominio mediante un ataque a SQL.

#### Revisar mínimo de CPU asignada

Debemos asegurar un mínimo de CPU para los servidores Terminal Server. Actualmente están alojados en un host físico sin núcleos suficientes para cubrir las asignaciones estándares de todas las máquinas virtuales que aloja. Si estos servidores pueden distribuir la carga sin un mínimo de procesamiento obligatorio por máquina se pueden producir cortes en el servicio.

#### Revisar alineación de discos

Actualmente los discos están alineados correctamente, es importante asegurar que si se añaden nuevas unidades de disco para los servidores de base de datos de la plataforma, éstos también estén alineados correctamente.

#### Revisar tamaño de bloque de discos

SQL Server instalado en un servidor físico debería disponer de discos con un tamaño de bloque de 64kb. Esto no es una norma fija si hablamos de máquinas virtuales sobre cabinas de discos. En estos casos particulares se debe revisar la documentación de los fabricantes y ver cuál es el tamaño óptimo de bloque para SQL Server, puede o no coincidir con el tamaño de bloque por defecto de unidades NTFS (4kb).

#### Utilidad para simular carga de disco

La herramienta gratuita **SQLIOSIM** nos permitirá simular carga sobre los discos que escojamos de forma que podamos medir su rendimiento en situaciones parecidas a un entorno de producción real, podremos encontrar la herramienta en el siguiente enlace:

<http://support.microsoft.com/kb/231619>

## Performance Monitor

**PENDIENTE SOLICITUD DE PERMISOS DE ADMINSITRACIÓN EN SERVIDORES**

## DynamicsPerf

### Instalación

Descargamos e instalamos la herramienta DynamicsPerf, que nos permitirá medir el rendimiento de la plataforma.

<https://dynamicsperf.codeplex.com/releases/view/122779>

Seguimos las instrucciones especificadas en el documento siguiente para realizar la correcta instalación y parametrización de la herramienta:



Una vez parametrizado el aplicativo tenemos una colección de scripts, tablas y vistas de SQL con los que vamos a poder analizar el uso que se hace de nuestra plataforma y el rendimiento que ofrece esta.

### Tareas periódicas

Es necesario realizar determinadas tareas para asegurar una recolección de datos analizables periódica, se detallan a continuación:

#### Captura de datos de AOT

Es necesario programar la ejecución periódica por lotes de la clase de Dynamics AX **AOTExport**, la cual rellenará tablas en el mismo aplicativo con datos sobre los objetos y datos modificados. También puede lanzarse bajo demanda si no se van a analizar éstos datos muy a menudo. La captura de datos mediante esta clase suele tardar entre 1 y 2 horas en las cuales Dynamics AX se realentiza ligeramente pero no impide su uso corriente por parte de otros usuarios.

#### Captura de datos de AOS

Es necesario lanzar de forma periódica un script de captura de datos de los diferentes servidores de aplicación de la plataforma, para realizar éste proceso ejecutaremos el script **AOSAnalysis.cmd** en los diferentes AOS pasando como parámetros el servidor y la instancia de SQL donde alojamos la BBDD de Dynamics AX y el nombre de la BBDD en cuestión. Dentro de Dynamics Perf lanzaremos posteriormente la siguiente consulta para examinar los datos obtenidos de nuestros servidores de aplicación:

SELECT \* FROM DynamicsPerf.dbo.AOS\_EVENTLOG

SELECT \* FROM DynamicsPerf.dbo.AOS\_REGISTRY

### Análisis

#### Ejecuciones de Captura

Antes de analizar datos con DynamicsPerf es necesario ver en qué momento se han realizado las ejecuciones de capturas de datos y si estamos analizando datos recientes o por algún motivo los procesos de captura han quedado bloqueados. Lo revisaremos con la consulta siguiente:

SELECT \*

FROM DynamicsPerf.dbo.STATS\_COLLECTION\_SUMMARY

ORDER BY STATS\_TIME DESC

#### Log de Errores de SQL

Debemos revisar periódicamente el log de errores de SQL para detectar errores recurrentes y fallos graves puntuales obteniendo un punto de partida para su análisis. También utilizaremos este análisis para comprobar la periodicidad con la que se están realizando los Backups de la BBDD. Podremos ver este log de forma cómoda mediante la siguiente consulta:

SELECT \*

FROM SQLERRORLOG

ORDER BY LOGDATE DESC

#### Configuración de SQL

Para revisar la configuración actual de las diferentes BBDD que compartan instancia con la BBDD de DynamicsPerf, utilizaremos la consulta siguiente:

SELECT \*

FROM DynamicsPerf.dbo.SQL\_DATABASES\_CURR\_VW

#### Archivos de SQL

Para revisar los archivos utilizados por las BBDD que comparten instancia con la BBDD de DynamicsPerf utilizaremos la consulta siguiente:

SELECT \*

FROM DynamicsPerf.dbo.SQL\_DATABASEFILES\_CURR\_VW

#### Secuencias Numéricas

Debemos analizar periódicamente las secuencias numéricas configuradas en el sistema y el número de secuencias consumidas por los usuarios para poder tomar decisiones de actuaciones al respecto.

##### Secuencias configuradas

Mediante la siguiente consulta lanzada contra DynamicsPerf, veremos un resumen de las secuencias configuradas en el sistema:

-- VER SECUENCIAS NUMÉRICAS ACTUALES

SELECT

N.COMPANYID as [CompanyID] ,

isnull(N.[Number\_sequence\_total], 0) AS [Number\_sequence\_total] ,

isnull(N.[Number\_sequence\_continuous], 0) AS [Number\_sequence\_continuous] ,

isnull(N.[Number\_sequence\_non\_continuous], 0) AS [Number\_sequence\_non\_continuous] ,

isnull(N.[Number\_sequence\_non\_continuous\_with\_preallocation], 0) AS [Number\_sequence\_non\_continuous\_with\_preallocation]

FROM

(

SELECT DISTINCT

COMPANYID,

(SELECT COUNT(\*) FROM AX\_NUM\_SEQUENCES\_CURR\_VW NT1 WHERE NT.COMPANYID = NT1.COMPANYID AND NT1.NEXTREC > 0 GROUP BY NT1.COMPANYID) AS [Number\_sequence\_total] ,

(SELECT COUNT(\*) FROM AX\_NUM\_SEQUENCES\_CURR\_VW NT2 WHERE NT.COMPANYID = NT2.COMPANYID AND NT2.CONTINUOUS = 'Yes' AND NT2.NEXTREC > 0 GROUP BY NT2.COMPANYID) AS [Number\_sequence\_continuous] ,

(SELECT COUNT(\*) FROM AX\_NUM\_SEQUENCES\_CURR\_VW NT3 WHERE NT.COMPANYID = NT3.COMPANYID AND NT3.CONTINUOUS = 'No' AND NT3.NEXTREC > 0 GROUP BY NT3.COMPANYID) AS [Number\_sequence\_non\_continuous] ,

(SELECT COUNT(\*) FROM AX\_NUM\_SEQUENCES\_CURR\_VW NT4 WHERE NT.COMPANYID = NT4.COMPANYID AND NT4.CONTINUOUS = 'No' AND NT4.NEXTREC > 0 AND NT4.FETCHAHEAD <> 0 GROUP BY NT4.COMPANYID) AS [Number\_sequence\_non\_continuous\_with\_preallocation]

FROM AX\_NUM\_SEQUENCES\_CURR\_VW NT

) AS N

##### Secuencias consumidas

Mediante la siguiente consulta lanzada contra DynamicsPerf, veremos un resumen de las secuencias consumidas por el sistema:

-- VER SECUENCIAS NUMÉRICAS CONSUMIDAS EN LAS ÚLTIMAS 24 HORAS

WITH OrderedStatsCollections AS

(

SELECT run\_name,

ROW\_NUMBER() OVER (ORDER BY stats\_time) AS 'RowNumber'

FROM stats\_collection\_summary

)

-- Select max created rows, in all stats collections, for each continuous numbersequence/dataareaid.

SELECT [COMPANYID] , [NUMBERSEQUENCE] , [TXT] , MAX ([TOTAL NUMBERS CONSUMED] ) AS [TOTAL NUMBERS CONSUMED] , duration, FORMAT

from

(SELECT RUN1. COMPANYID, RUN1. NUMBERSEQUENCE as NUMBERSEQUENCE , RUN1. TEXT AS TXT,

RUN2. NEXTREC - RUN1. NEXTREC AS [TOTAL NUMBERS CONSUMED] ,

duration = 'In ' + STR((DATEDIFF(hh, RUN1. STATS\_TIME,

RUN2. STATS\_TIME)), 2) + 'hour periods' ,

RUN1. [FORMAT]

FROM [AX\_NUM\_SEQUENCES\_VW] RUN1

INNER JOIN [AX\_NUM\_SEQUENCES\_VW] RUN2

ON RUN1. NUMBERSEQUENCE = RUN2. NUMBERSEQUENCE

AND RUN1. COMPANYID = RUN2. COMPANYID

INNER JOIN OrderedStatsCollections S1

ON S1. Run\_Name = RUN1. Run\_Name

INNER JOIN OrderedStatsCollections S2

ON S2. Run\_Name = RUN2. Run\_Name

WHERE

RUN1. [CONTINUOUS] = 'Yes'

AND S2. RowNumber = S1. RowNumber + 1

AND RUN2. NEXTREC > RUN1. NEXTREC

AND DATEDIFF(hh, RUN1. STATS\_TIME, RUN2. STATS\_TIME) = 3

) RES

GROUP BY [COMPANYID] , [NUMBERSEQUENCE] , [TXT] , duration, FORMAT

ORDER BY [TOTAL NUMBERS CONSUMED] DESC;

##### Secuencias no-continuas sin pre-allocation

Mediante la siguiente consulta veremos las secuencias no continuas del sistema que no utilizan pre-allocation, es decir que no reservan una serie de números para un usuario en concreto cada vez que éste solicita un número de nuevo. Si encontramos entre ellas secuencias que se consumen de forma masiva ya sea por un proceso automático o por alguna acción de usuario que se realice muy a menudo será interesante ahorrar consultas a la bbdd utilizando un valor de pre-allocation que reduzca las peticiones:

WITH OrderedStatsCollections AS

(

SELECT run\_name,

ROW\_NUMBER() OVER (ORDER BY stats\_time) AS [RowNumber]

FROM stats\_collection\_summary

)

-- Select max created rows, in all stats collections, for each non-continuous + no prefetch numbersequence/dataareaid.

SELECT [NUMBERSEQUENCE] , [TXT] ,

MAX ([TOTAL NUMBERS CONSUMED] ) AS [TOTAL NUMBERS CONSUMED] ,

MAX ([AVG HOURLY CONSUMPTION RATE] ) AS [AVG CONSUMPTION in 1 hour periods] ,

FORMAT

from

(

SELECT RUN1. NUMBERSEQUENCE as NUMBERSEQUENCE , RUN1. TEXT AS TXT,

RUN2. NEXTREC - RUN1. NEXTREC AS [TOTAL NUMBERS CONSUMED] ,

(RUN2. NEXTREC - RUN1. NEXTREC)/(DATEDIFF(hh, RUN1. STATS\_TIME,

RUN2. STATS\_TIME)) AS [AVG HOURLY CONSUMPTION RATE] ,

RUN1. [FORMAT]

FROM [AX\_NUM\_SEQUENCES\_VW] RUN1

INNER JOIN [AX\_NUM\_SEQUENCES\_VW] RUN2

ON RUN1. NUMBERSEQUENCE = RUN2. NUMBERSEQUENCE

INNER JOIN OrderedStatsCollections S1

ON S1. Run\_Name = RUN1. Run\_Name

INNER JOIN OrderedStatsCollections S2

ON S2. Run\_Name = RUN2. Run\_Name

WHERE

((RUN1. [CONTINUOUS] = 'No') AND (RUN1. FETCHAHEADQTY = 0))

AND S2. RowNumber = S1. RowNumber + 1

AND RUN2. NEXTREC - RUN1. NEXTREC > 100

) RES

GROUP BY [NUMBERSEQUENCE] , [TXT] , FORMAT

ORDER BY [TOTAL NUMBERS CONSUMED] DESC;

#### Índices de la BBDD

Es interesante optimizar el uso de índices que hace la BBDD de forma que optimicemos el rendimiento de ésta llevando a cabo las siguientes revisiones periódicas:

##### Índices no utilizados

Los índices que puedan existir y a los que no se haya dado uso en un periodo prudencial (un ejercicio completo, por ejemplo) se pueden eliminar, para revisar el uso de los índices haremos la consulta siguiente:

SELECT TABLE\_NAME,

INDEX\_NAME,

INDEX\_KEYS,

SUM(USER\_UPDATES) AS USER\_UPDATES

FROM INDEX\_STATS\_VW

WHERE INDEX\_DESCRIPTION NOT LIKE '%UNIQUE%'

AND INDEX\_DESCRIPTION != 'HEAP'

GROUP BY TABLE\_NAME, INDEX\_NAME, INDEX\_DESCRIPTION, INDEX\_KEYS, INCLUDED\_COLUMNS

HAVING SUM(USER\_SEEKS) = 0

AND SUM(USER\_SCANS) = 0

AND SUM(USER\_LOOKUPS) = 0

ORDER BY USER\_UPDATES DESC

Hay que tener muy en cuenta que activar nuevas funcionalidades en el ERP o empezar a realizar procesos que anteriormente no se usaban puede requerir de algún índice que hemos podido eliminar previamente.

##### Índices faltantes

Indexando las tablas a través de los campos por los que se suelen aplicar filtros de consulta optimizaremos el rendimiento de la BBDD siempre y cuando no creemos una cantidad de índices no controlada que haga crecer el tamaño de la tabla de forma notable.

Localizaremos los índices faltantes utilizando la consulta siguiente:

SELECT TOP 100 INDEX\_IMPACT AS [Index Impact] ,

SQL\_TEXT AS [Query Text] ,

QUERY\_PLAN AS [Query Plan] ,

TABLE\_NAME AS [Table Name] ,

EQUALITY\_COLUMNS AS [Equality Columns] ,

INEQUALITY\_COLUMNS AS [Inequality Columns] ,

EXECUTION\_COUNT AS [Execution Count] ,

STR(ROUND(TOTAL\_ELAPSED\_TIME, 0), datalength(cast(TOTAL\_ELAPSED\_TIME AS VARCHAR(32))) - 3, 0) AS [Total Time] ,

ROUND(AVG\_ELAPSED\_TIME, 2) AS [Average Time]

FROM MISSING\_INDEXES\_CURR\_VW

--WHERE INCLUDED\_COLUMNS = ''

ORDER BY TOTAL\_LOGICAL\_READS DESC

##### Índices actuales

Revisaremos los índices actualmente existentes en el sistema mediante la consulta siguiente:

SELECT TOP \* FROM [DynamicsPerf].[dbo].[AX\_INDEX\_DETAIL\_CURR\_VW]

#### Tamaño de Virtual Log Files de SQL

Debemos comprobar periódicamente el tamaño de los Virtual Log Files (VLF) de SQL de forma que tengamos su crecimiento controlado. Podremos revisar su tamaño mediante la consulta siguiente:

WITH VLFUSE AS

(

SELECT DATABASE\_NAME, FILEID,

COUNT(\*) AS VLF\_COUNT,

SUM(CASE WHEN STATUS = 0 THEN 1 ELSE 0 END) AS FREE,

SUM(CASE WHEN STATUS != 0 THEN 1 ELSE 0 END) AS INUSE

FROM LOGINFO

GROUP BY DATABASE\_NAME, FILEID

)

SELECT DISTINCT FILE\_NAME, VLF\_COUNT, FREE,INUSE,[DB\_SIZE(MB)] AS [FILE\_SIZE(MB)],[GROW\_MAX\_SIZE(MB)],GROWTH\_UNITS FROM VLFUSE V

INNER JOIN SQL\_DATABASEFILES F ON V.DATABASE\_NAME= F.DATABASE\_NAME and V.FILEID = F.FILE\_ID

WHERE F.DATABASE\_NAME = 'FMNAX09LIVE'

#### Procesos por Lotes

Los procesos por lotes deben programarse en franjas horarias concretas acorde con su funcionalidad e impacto a la plataforma. Deberemos disponer de un calendario de franjas horarias donde poder visualizar de forma rápida cuánto duran nuestros procesos. Podremos ver los procesos por lotes programados en Dynamics AX y su histórico de ejecuciones mediante la siguiente consulta:

SELECT \* FROM MicrosoftDynamicsAX.dbo.BATCH

SELECT \*

FROM MicrosoftDynamicsAX.dbo.BATCH B

LEFT JOIN MicrosoftDynamicsAX.dbo.BATCHHISTORY H ON

B.RECID = H.BATCHID

#### Consultas largas

Es recomendable activar la traza de consultas largas de AX para todos los usuarios con un margen de log de 2 segundos o superior de forma que podamos ver aquellos procesos que se ejecutan de forma lenta, si tenemos la traza de consultas largas activada podremos revisar los procesos que se han capturado mediante la siguiente consulta:

DECLARE @DATE AS DATETIME; SET @DATE = '01/29/2015'

SELECT TOP 1000 [TRACECONTAINER],[TRACETIME],[CATEGORY],[STATEMENT],[CALLSTACK],[CODE],[TEXT],[TEXTDETAILS],[CONNECTIONTYPE],[CONNECTIONSPID],[CONNECTIONID],[ISLOBSINCLUDED],[ISMOREDATAPENDING],[ROWSAFFECTED],[ROWSIZE],[ROWSPERFETCH],[ISSELECTEDFORUPDATE],[ISSTARTEDWITHINTRANSACTION],[STATEMENTTYPE],[STATEMENTID],[STATEMENTREUSECOUNT],[DETAILTYPE],[CREATEDDATETIME],[DATAAREAID],[RECVERSION],[RECID]

FROM [MicrosoftDynamicsAX].[dbo].[SYSTRACETABLESQL]

WHERE CREATEDDATETIME >= @DATE

#### Bloqueos de SQL

Podemos programar un registro de los bloqueos de SQL que han durado más de un tiempo prudencial, para posteriormente examinarlos desde DynamicsPerf. Con la siguiente instrucción solicitaremos que se guarde un registro de todos aquellos bloqueos de registro que hayan durado más de 5 segundos:

sp\_configure 'blocked process threshold' , 5  
reconfigure

Posteriormente podremos revisar los bloqueos mediante la consulta siguiente:

SELECT \* FROM BLOCKED\_PROCESS\_VW

ORDER BY BLOCKING\_LAST\_BATCH\_STARTED, [WAIT\_TIME(MS)] DESC

### Jobs SQL

##### CaptureStats

Ejecuta la recolección de datos de DynamicsPerf.

##### CaptureStatsBaseline

Ejecuta una recolección de datos como punto de referencia para futuras capturas.

##### CaptureStatsPurge

Elimina las capturas de datos con más de dos semanas de antigüedad.

##### CaptureStatsDefaultTraceStart

Activa la captura de traza SQL para todos los usuarios de Dynamics AX.

Podemos configurar la periodicidad de la captura y la ruta de destino del archivo resultante en el paso del script. Una vez generado el fichero lo podremos analizar con el mismo cliente SQL Server Management Studio.

##### CaptureStatsDefaultTraceStop

Desactiva la captura de traza SQL para todos los usuarios de Dynamics AX.

##### LongDurationTrace

Activa la traza de consultas de larga duración.

Es más óptimo utilizar la captura propia de Dynamics AX dado que este script produce una caída de rendimiento considerable.

##### OptionalPollingForBlocking

Traza para obtener información detallada de bloqueos de SQL. Produce una caída de rendimiento considerable.

##### PerfStatsHourly

Comprueba las variaciones de la BBDD cada hora.

##### PurgeBlocks

Limpia la información de OptionalPollingForBlocking.

##### PurgeSysTraceSQLAX

Limpia la información de las trazas de SQL de Dynamics AX.

##### SetAXUsersTraceOff

Deshabilita traza de consultas largas para todos los usuarios de AX.

##### SetAXUsersTraceOn

Habilita la traza de consultas largas para todos los usuarios de AX.

## Dynamics AX

### Análisis

#### Crecimiento de Tablas

Debemos revisar periódicamente el crecimiento de las tablas de log y traza de usuarios de forma que no se descontrole su crecimiento, podremos ver el tamaño de las tablas implicadas en éstos procesos de forma estándar con la consulta siguiente:

SELECT TABLE\_NAME, ROW\_COUNT

FROM INDEX\_STATS\_CURR\_VW

WHERE

(INDEX\_DESCRIPTION LIKE 'CLUSTERED%' OR

INDEX\_DESCRIPTION LIKE 'HEAP%' ) AND

TABLE\_NAME IN

( 'EVENTCUD' --alerts

, 'SYSTRACETABLESQL' --tracing

, 'INVENTSUMLOGTTS'

, 'INVENTSETTLEMENT'

, 'BATCH'

, 'BATCHHISTORY'

, 'SYSTRACETABLESQL'

, 'SYSDATABASELOG'

, 'SYSEXCEPTIONTABLE'

)

ORDER BY TABLE\_NAME

#### Procesos de Limpieza

Los procesos que debemos realizar de forma periódica para asegurar que el crecimiento de las tablas de log no se incrementa fuera de nuestro control se detallan a continuación:

##### Alertas

Las alertas de Dynamics AX se limpian mediante el proceso que encontramos en la ruta siguiente dentro del cliente de Dynamics AX. El proceso especificado lo utilizaremos programándolo como un proceso por lotes periódico.

**Administración del sistema – Periódico – Limpieza de notificaciones**

##### Traza de SQL

La traza de cambios realizados sobre registros configurada en Dynamics AX se limpia mediante el proceso que encontramos en la ruta siguiente dentro del cliente de Dynamics AX. El proceso especificado lo utilizaremos programándolo como un proceso por lotes periódico pero deberemos tener en cuenta durante cuánto tiempo queremos mantener el registro especificado por motivos funcionales y de seguridad.

**Administración del sistema – Consultas – Base de datos – Registro de la base de datos – Borrado de registro**

##### Lista de Transacciones de Inventario

La lista de transacciones de inventario se limpiará periódicamente mediante los procesos de cierre de inventario que realiza la empresa por motivos funcionales. En caso de no utilizar estos procesos en alguna de las filiales dónde se incorpore el uso de Dynamics AX, deberemos realizar un procedimiento programado propio de forma que no dejemos que la tabla crezca desmesuradamente.

##### Diarios Contables

Dependiendo de requerimientos funcionales podremos o no, realizar la limpieza de ésta tabla. En caso de estar permitido, la forma de limpiarla será mediante el proceso que encontramos en la ruta siguiente dentro del cliente de Dynamics AX. Deberemos tener en cuenta durante cuánto tiempo queremos mantener el registro especificado por motivos funcionales y de seguridad dependiendo de cada tipo de diario.

**Contabilidad general – Periódico – Limpiar – Limpiar diarios de contabilidad**

##### Procesos por Lotes e Historial

Los procesos por lotes deberemos limpiarlos una vez estén completados y haya transcurrido un tiempo prudencial en el que ya no los necesitemos más para su consulta. Este tiempo puede ser tan reducido como una semana dado que guardaremos el histórico de los procesos ejecutados en DynamicsPerf. Para realizar la limpieza de estos registros deberemos ejecutar el proceso que encontramos en la ruta siguiente dentro del cliente de Dynamics AX.

**Administración del sistema – Consultas – Trabajos por lotes – Trabajos por lotes – Funciones – Eliminar**

##### Histórico de Sesiones de Usuario

La tabla de sesiones de usuario debería vaciarse periódicamente teniendo en cuenta el tiempo requerido por necesidades funcionales para la permanencia de los registros. Para ejecutar la limpieza de los registros ejecutaremos el proceso que encontramos en la ruta especificada a continuación. Es adecuado programar este proceso como un trabajo por lotes que se ejecute de forma periódica.

**Administración del sistema – Consultas – Usuarios – Registro de usuario – Limpiar**

#### Cuenta de Usuario para Procesos por Lotes

Debemos asignar permisos para programar procesos por lotes únicamente a una cuenta de usuario de forma que podamos discernir entre procesos lanzados por un usuario bajo demanda o automáticamente al realizar cualquier acción que lo requiera y procesos programados en una franja horaria concreta por los administradores de la plataforma.

### Actualización a R2/R3

#### Requerimientos R2/R3

Los requerimientos para ambas actualizaciones son prácticamente los mismos, a continuación se detallan los requerimientos para la versión R3:

*Windows Server 2012*

*SQL Server 2012 SP2 - CU2 / 2014*

*Sharepoint 2013 SP1 Foundation / Enterprise*

*Analysis Management Objects para SQL Server*

*Enterprise Search para Sharepoint 2013 SP1*

*SSRS + SSAS: mismas versiones de SQL Server*

#### Destino de Actualización

Tal como podremos observar en los documentos de novedades de las versiones R2 y R3 de Dynamics AX, las diferencias entre ambos no son tanto a nivel de sistema como en la inclusión de nuevas funcionalidades respecto a la versión R2. Por lo tanto el esfuerzo de actualización es prácticamente el mismo para pasar a una u otra versión. Dado que la versión R3 ya lleva un tiempo prudencial en el mercado sin problemas mayores detectados, la recomendación es actualizar a esta.



## Dynamics Lifecycle

La herramienta Dynamics Lyfecicle nos permitirá analizar varios aspectos de nuestra plataforma mediante las herramientas especificadas a continuación.

<https://lcs.dynamics.com>

### Propuesta de Infraestructura

Mediante un proyecto de **Implementación** de Dynamics Lyfecicle podremos utilizar la utilidad **Infraestructure Estimator** para definir determinados parámetros de nuestra plataforma y obtener una estimación de la infraestructura de servidores y servicios óptimos.

### Coste de Actualización

Mediante un proyecto de **Actualización** de Dynamics Lyfecicle podremos utilizar la herramienta **Upgrade Analysis** para importar una exportación del Model Store de Dynamics AX y analizar cómo puede impactar la actualización de versión en nuestros desarrollos previos y el estándar de la aplicación.

### Exportación de ModelStore

A diferencia de versiones anteriores de Dynamics AX, el código de la aplicación al completo se almacena en la BBDD como datos, en el esquema Model de SQL. Para analizar nuestro código con Dynamics Lyfecicle será necesario realizar una exportación de éste en un archivo y enviarlo a Lyfecicle para obtener resultados.

La exportación del Model Store de Dynamics AX se llevará a cabo siguiendo las instrucciones especificadas en el siguiente enlace:

<https://technet.microsoft.com/en-us/library/hh352314.aspx>

**PENDIENTE RESPUESTA DE CARLOS A LA EXPORTACIÓN DEL MODEL STORE**