IBM Software

Connect 2014

January 26–30 Orlando, Florida

ENERGIZING LIFE'S WORK

BP 102: IBM Domino 64bit - All You Need to Know

Daniel Nashed | CTO | Nash!Com

IBM

About the Speaker

- **Daniel Nashed**
  - Nash!Com – German IBM/Lotus Business Partner/ISV
  - Member of The Penumbra group -- an international consortium of selected Business Partners pooling their talent and resources
  - focused on Cross-Platform C-API, Domino® Infrastructure, Administration, Integration, Troubleshooting and IBM® Traveler
  - Platform Focus: Windows® 32/64, Linux®, AIX®
  - Author of the Domino on Linux/UNIX® Start Script
-

Agenda

- **Introduction and Basics about 32/64bit**
- **IBM Lotus® Domino® 32bit on a 64bit OS**
  - Best Practices and Experience from the Field
- **Native Domino 64bit**
  - Best Practices and Experience from the Field
- **Q&A**

Operating Systems Covered

- **Focus in this presentation is Windows and Linux 32bit / 64bit**
- **There are other 64bit implementations**
  - **zLinux** is only available in native 64bit mode
    - because previous versions have been technically a 31bit implementation with 2 GB memory address limit
  - **AIX** supports 32bit and 64bit Domino
    - It's strongly recommended to use native 64bit because of address limitations caused by the segmented memory model on AIX
  - **iSeries - i5/OS** has been a true 64bit – or technically 128bit implementation for a long time
  - **Solaris** is not supported any more in Domino 9

A Brief History

- **Domino native 64bit for Windows has been introduced in Domino 8.0.1**
  - But most customers did not switch to native 64bit immediately
  - Has to be seen as a separate platform which needed separate testing
  - With the late 8.5.x and 9.x releases more and more customers moved to native 64bit
  - Specially Traveler customers are moving to 64bit because of higher memory demand for “larger” Traveler installations
- **Domino native 64bit for Linux has been introduced with Domino 9.x**
  - Quite new platform but we already see customers migrating to native 64bit on Linux
    - Traveler Servers
    - Servers moved from Domino 8 with new hardware
    - Important: Domino 9 requires SLES 11.x and RHEL 6.x

32 Bit / 64 Bit Basics

- **Without “tricks” a 32 Bit OS can only address at most 4 GB Memory**
  - That's why 32 Bit Process can at most allocate 4 GB Memory

- **Address Space Limits**
  - 32 Bit = 2^32 = 4 GB
  - 64 Bit = 2^64 = 18,446,744,073,709,551,615 = 18.45 Exabytes
    - That's more than we will “ever” need …
    - But that's what we thought about 32bit (and long time ago 8bit) as well

- **Windows 32bit Limits**
  - Split memory into 2 GB for System and 2 GB for applications
  - Total memory available was only 2 GB for applications
  - /3GB switch to change to 3/1 Memory limit was not really a solution

- **So it makes a lot of sense to at least switch to a 64bit Operating System**

Domino 9.0.1 - System Requirements

- **Windows**
  - Windows Server 2008 Standard Edition R2 / Enterprise Edition R2 **x86-64** plain + SP1
  - Windows Server 2012 Standard Edition/Datacenter Edition **x86-64**
  - **Windows Server 2012 R2**

  - Starting with Windows 2008 R2 there is no supported 32bit Microsoft Server OS anymore
  - 32bit and 64bit Domino is supported

- **Linux**
  - Red Hat Enterprise Linux (RHEL) Server 6.x **x86-64**
  - SUSE Linux Enterprise Server (SLES) 11.x **x86-64**

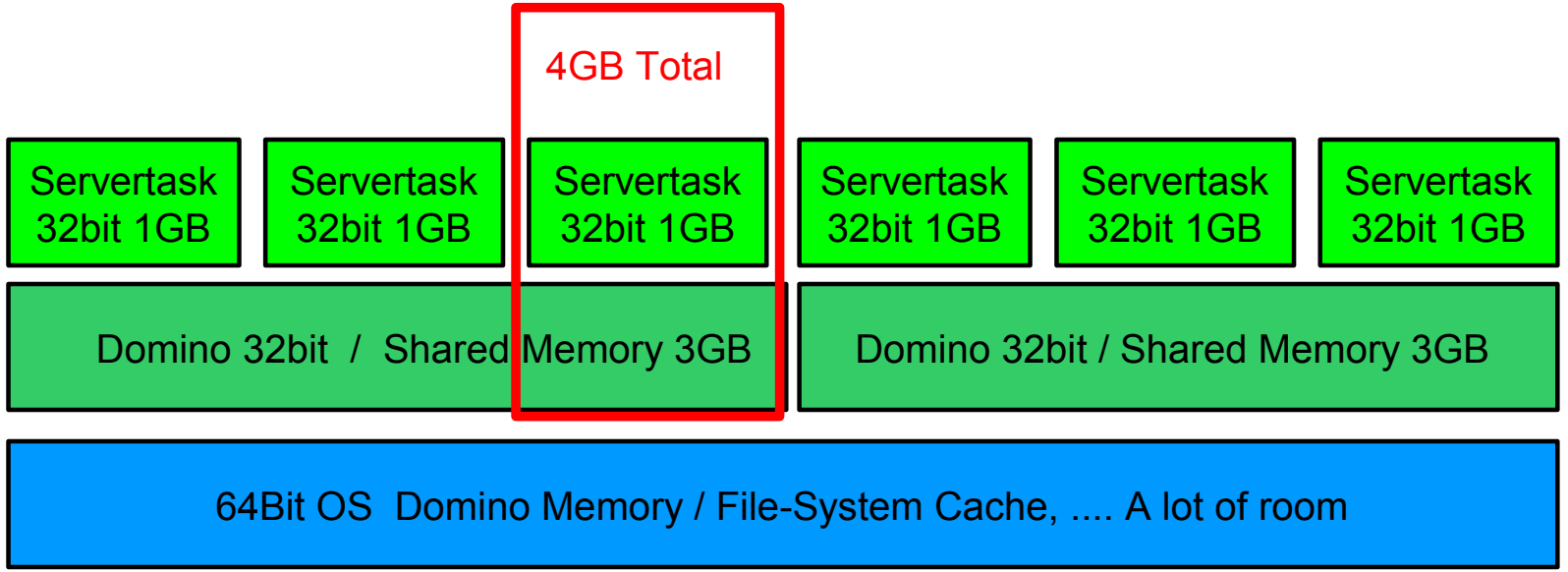
  - Domino only supports 64bit Linux
    - 32bit and 64bit Domino is supported

Domino 32bit Limits on 64bit OS

- **Domino uses Local Process Memory and Shared Memory**
  - **Local memory** = each process has local memory only used by this process
    - Http, Router, Amgr Traveler use a lot of local process memory
  - **Shared memory** = shared among all processes for different pools
    - All shared memory is mapped to all Domino processes
    - Most prominent pool: NSF Buffer Pool (internally known as UBM)
      - Default: 512 MB for 32bit Domino, 1 GB for 64bit Domino
- **On a 64bit OS all Domino 32bit processes have a separate 32bit address space within the 64bit OS address space**
- **<u>But</u> the combination of local and shared memory cannot exceed 4 GB for any process**

Domino 32bit on a 64Bit Operating System

- **Total Memory per Process is 32Bit = 4 GB**
  - Router / HTTP uses most local process memory
  - NSF Buffer Pool is the biggest Shared Memory block (512 MB)



Domino 32bit on a 64Bit OS

- **64Bit OS allows multiple partitions with dedicated 32Bit address space per process**
  - Very good for consolidation of multiple Domino servers
- **Memory Limits**
  - In normal cases shared + local process memory does not exceed 3-4 GB
- **The remaining memory is used by the 64Bit OS**
  - File caching, buffers, internal resources
- **Running 32Bit Domino on 64bit OS gives you already most of the performance and scalability benefits**
  - But there are some details you should take care about (see next slides)
  - And there are still good reasons to migrate to native 64bit

Memory Considerations

- **Add memory for file-system caching**
  - Domino will only use 3-4 Memory for each Domino Partition
    - This will not change dramatically with native Domino 64bit
  - But the OS will use the remaining memory for file-system caching
  - Specially on virtual servers this can dramatically improve performance and reduce read I/Os
- **We have seen dramatic reduction of I/Os when increasing RAM from 4 to 16 GB**
  - This is true for large mail and application servers on physical and virtual machines
  - RAM is relatively inexpensive and modern system boards offer sufficient slots for RAM
- **But you need to be aware some issues of file-system cache that can occur**
  - Details on next slides

Large System Cache for 64 Bit Windows

- **By default there is a very high physical memory limit for the file-system cache**
  - It will try to use all memory which can cause Domino Memory to be swapped out
    There is a Windows 64bit call “SetSystemFileCacheSize()” to limit the cache
    - Available since Win2003 R2 or SP2 64Bit

- **Domino uses this system call on start-up to limit the Windows OS Level Cache**
  - Domino 8 ships a 64bit helper binary “cacheset.exe” to set the cache size for Domino 32bit
    - Domino 64bit has this call integrated into the core code
    - Will need the system privilege “SE_INCREASE_QUOTA_NAME”
    - See TN #1391477 for details

  - **Default is 30% of memory**
    - Can be tuned via notes.ini **MEM_FSCachePercentMem=n**
    - Set depending on the RAM in your machine – **also required for native Domino 64**
    - Example: 16 GB RAM, 6 GB reserved for Domino/OS = MEM_FSCachePercentMem=65
    - You can check the current settings with “cacheset.exe -g”

File-System Cache Issue with Windows 64bit

- **Windows 32bit only used around 300 MB of File-System cache**

- **With 64bit the file-system cache can grow to 1 TB virtual memory**
  - File-system cache is implemented in virtual memory in the same way applications use memory
    - Windows also keeps file cache in memory when it is not located in physical memory
  - You can end-up with 1 TB of virtual memory file-system space which is not in physical memory at all
  - At 1 TB of open file data the cache is reorganized which leads to very high CPU spikes
  - It turned out that Domino opens all files with "**random-optimized**" flag which lead to long caching of file-data in cache (FILE_FLAG_RANDOM_ACCESS)
  - Specially during backups on a large Domino server you can run into performance issues because even the database is already closed by the backup application it is still open in the DB cache

- **Solution**
  - Not really a Domino issue but IBM had to disable the RANDOM Access optimization flag
  - SPR# KBRN899NF6 (8.5.3/8.5.2 FP1) : **Disable_Random_RW_File_ATTR=1**
  - SPR# KBRN8AKKA9 (8.5.3/8.5.2 FP3): **Default enabled on Windows**

Platform Support for 32bit Sub-Systems on 64bit OS

- **Windows uses the WOW sub-system to allow 32bit applications to run unchanged**
  - Take care that registry parameters change
    - HKEY_LOCAL_MACHINE\SOFTWARE**Wow6432Node**\Lotus\Domino
  - See http://en.wikipedia.org/wiki/WoW64 for details
- **Linux needs 32bit sub-system libs to be installed**
  - On SLES you can install 32bit libs for all your libs with a single setting in YaST
  - On RHEL 6 either install all libs manually or set **multilib_policy=all** in **/etc/yum.conf**

Application Considerations for Domino 32bit on 64bit OS

- **There are no changes required for LotusScript®, Java and @Formula code**
- **You should take care that ODBC need 32bit versions of the ODBC drivers installed**
  - Separate configuration for native 64bit and 32bit ODBC resources
- **Usually no changes for system calls**
  - System finds the right 32bit implementation of the call
- **No change for C-API calls from LotusScript**
- **No code changes of C-API based code**
  - See one important detail for larger servers next slide

32 Bit Domino on Windows 64 over 2 GB

- **Add-On Applications need to be recompiled and linked with Visual Studio .Net 2003 with link flag /LARGEADDRESSAWARE**
  - Else if any process (servertask) exceeds 2 GB limit (local + shared memory) it would crash!
  - Check via dumpbin
  - Example: dumpbin /headers nnshdbcat.exe
- **Result of dumpbin /headers should contain:**
  - FILE HEADER VALUES
  - Application can handle large (>2GB) addresses
- **Default Max Shared Memory is set to 2GB**
  - So beyond 2 GB Shared Memory Domino would crash without increasing the limit
  - Increase the limit via notes.ini **ConstrainedSHMSizeMB=3072** gives you a maximum of 3GB
  - That means 1 GB left for local process memory per task

Tivoli® Data Protection Support for Domino 32bit

- **Mixed Mode = “Domino 32bit and OS is 64bit” is only supported in older versions**
  - Last Version with Mixed Mode Support is TDP 5.5.3
- **First Release with Domino 9.x support is TDP 7.1**
  - Which does not support any mixed mode implementations
- **TDP 6.x releases did not support Linux at all**
  - TDP 7.1 does support 64bit Linux only – Domino 9 only supports Linux 64
- **You can run TDP 5.5.3 for a while but once you are updating to Domino 9.x you need TDP 7.1 or higher**
  - TDP 7.1 on Linux is supported on 64bit Domino
  - because there is no supported 32bit OS for Domino 9 on Linux
- **We still have a PMR open with Tivoli but it looks like with Domino 9 TDP forces us to switch to native 64bit → which is in generally OK from strategy point of view**

64bit Support for Add-On Products

- **IBM Sametime® (Community Server)**
  - Only available on 32bit but supports 64bit OS
  - Not much data to backup. Take a manual off-line backup or replicate the few important databases
- **Lotus Quickr®**
  - Only available on 32bit but supports 64bit OS
  - No new version. Domino 8.5.x is still supported with TDP 5.5
- **IBM Enterprise Integrator (LEI)**
  - Native 64bit Version for Windows for 8.5.x and 9.0
  - New with 9.0 native support for 64bit on Linux
- **Traveler**
  - Recommended to install 64bit native
  - Only LotusTraveler.nsf needs backup, state DB is either local Derby DB or DB2/SQL Server

Domino Native 64bit

IBM Software

Connect 2014

January 26–30 Orlando, Florida

ENERGIZING LIFE'S WORK

What should you expect from Native 64bit

- **Should run a bit faster**
  - But the performance you gain moving from Domino 8.x to Domino 9.x is bigger
  - Also switch from 32bit OS and 64bit gives bigger boost, specially on Windows

- **It not double as scalable**
  - Sizing the “number of users” on 64bit should be similar to 32bit

- **Will better perform in high load scenarios**
  - Helps when you already have high memory utilization
  - That's does not mean that memory leaks with not hurt you!

- **Will need a bit more RAM (e.g. 30% more)**
  - Pointers are 64bit instead of 32biz

New Server Platform Native 64bit Intel Linux

- **Domino 9.0 introduces 64bit on xLinux**

- **True 64bit implementation**
  - Specially Traveler will benefit from it (Traveler 9.0 also supports 64bit on Linux)
    - Traveler needs a lot of local process memory for the traveler servertask
    - Side Note: Traveler ships both sets of binary in the same installer for Windows and Linux

- **For developers this requires a new development environment**
  - SLES 11 x86_64 (64 bit)
  - GNU Compiler Collection (gcc/g++) version 4.3.x
  - Each new compiler + environment uses a newer set of LIBs
    - By the way this is why SLES 10 and RHEL 5.x is not supported any more
  - Not a big change for application developers if already ported to other 64bit platforms (e.g. Win64)
  - You can run 32bit applications compiled for Domino 8 but you might need compat libs
    More details about C-API background later

Native Domino 64bit

- **Supported Platforms**
  - since Domino 8.0.1 on AIX64
  - since Domino 8.0.1 on Win2003 R2 64Bit
  - since Domino 9.0 on Linux SLES 11/RHEL 6
  - (iSeries and zLinux available for a longer time)

- **Domino as a 64Bit Application allows much more memory**
  - In theory a Win64 application could use up to 8 TB of data
  - But Domino 8.0.x / 8.5 64bit does allow more memory but is not specially optimized for larger memory
    - There are some performance enhancements in Domino 9 for native 64bit

Native 64bit Resources

- **A 64bit application – if compiled correctly – will run faster on a 64bit OS**
  - No WOW-Sub-System on Windows
  - No separate 32bit LIBs on Linux
  - Native 64bit Pointers
  - More registers available for 64bit applications
  - Better Process/Thread scheduling
  - Optimized direct memory access

- **Increased memory requirements**
  - Because of 64bit addresses, alignment changes and larger size for some data-types Domino 64bit will need more memory.
  - It's hard to say how much and it also depends on your environment.
    - I would assume 30%
  - But more memory usually is not big cost issue
  - You should have more memory for file-system caching anyway

How 64bit Native looks like - Linux

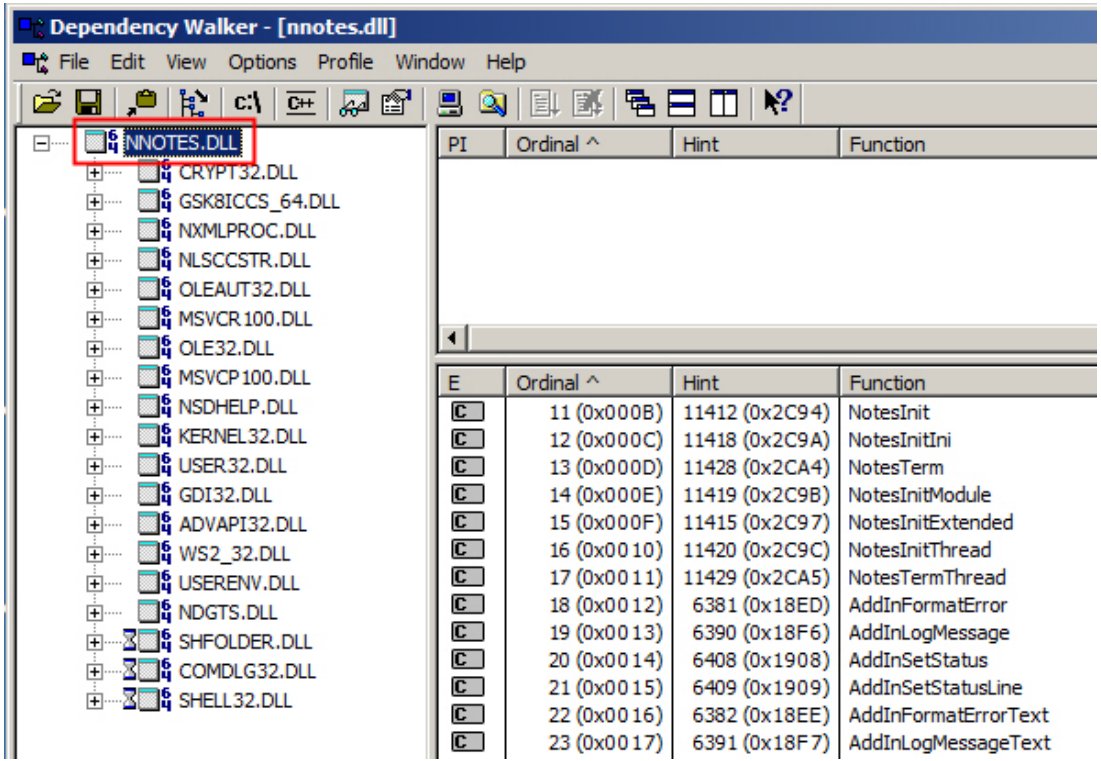
- **Use the “file” command to figure out if the binary is 64bit**
- **Use “ldd” command to check dependencies of binary**
  - Note: resolves only libs from the path

```
file libnotes.so
libnotes.so: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linked, not stripped

 ldd libnotes.so
        linux-vdso.so.1 =>  (0x00007fff589fd000)
        libndgts.so => /opt/ibm/domino/notes/latest/linux/libndgts.so (0x00002b14a6568000)
        libdl.so.2 => /lib64/libdl.so.2 (0x00002b14a6770000)
        librt.so.1 => /lib64/librt.so.1 (0x00002b14a6974000)
        libstdc++.so.6 => /usr/lib64/libstdc++.so.6 (0x00002b14a6b7c000)
        libpthread.so.0 => /lib64/libpthread.so.0 (0x00002b14a6e83000)
        libresolv.so.2 => /lib64/libresolv.so.2 (0x00002b14a70a0000)
        libc.so.6 => /lib64/libc.so.6 (0x00002b14a72ba000)
        libxmlproc.so => /opt/ibm/domino/notes/latest/linux/libxmlproc.so (0x00002b14a764f000)
        libgsk8iccs_64.so => /opt/ibm/domino/notes/latest/linux/libgsk8iccs_64.so (0x00002b14a7c79000)
        libm.so.6 => /lib64/libm.so.6 (0x00002b14a7dd3000)
        libgcc_s.so.1 => /lib64/libgcc_s.so.1 (0x00002b14a8058000)
        /lib64/ld-linux-x86-64.so.2 (0x00002b14a377a000)
```

How 64bit Native looks like - Windows

- **Use “Depends” application to check dependencies and platform**



How to figure out if the Server is 64bit

- **Show Server**

```
sh server

IBM Domino (r) Server (64 Bit) (Release 9.0.1 for Linux/64) 31.12.2013 12:30:42
IBM Domino (r) Server (64 Bit) (Release 9.0.1 for Windows/64) 31.12.2013 12:30:42
```

- **show stat Server.Version.Architecture**

```
show stat Server.Version.Architecture

Server.Version.Architecture = 64 Bit
```

Check Platform via LotusScript

- **Domino 9.0.1**
  - Check via
    - @Formula Language → @Platform([Specific])
    - LotusScript → session.Platform
  - Not completely consistent on Linux

	Domino 32bit Linux	Domino 64bit Linux	Domino 32bit Windows	Domino 64bit Windows
@Platform([Specific])	Linux	Linux/64	Windows/NT 6.1	Windows/NT 6.1
session.Platform	UNIX	Linux/64	Windows/32	Windows/64

Porting Applications to Native Domino 64bit

- **Client based application code is not affected**
  - You only need to take care of code invoked on the server (HTTP, Agents etc)
- **Standard LotusScript/Java/@Formula Code is safe**
  - Usually no change needed
- **ODBC needs to be reviewed**
- **All add-on C-API based applications need to be ported to 64Bit**
- **All C-API from LotusScript calls need to be ported to 64Bit**
- **All native Operating System calls, calls to different libs need to be ported**

Porting C-API Applications to 64bit

- **You need the right 64bit compiler**

- **Compiler has changed for Domino 9 on Windows**
  - Domino 8 used Intel Compiler because at first porting the Microsoft compiler was not “ready”
  - Now IBM switched back to the Microsoft Visual Studio Compiler also for Win64
  - That's good news because it makes porting easier
  - You need: “**Visual Studio 2010 SP1 Standard for x86 and x64**”
  - Free download:
    - Microsoft Windows SDK for Windows 7 and .NET Framework 4
      - http://www.microsoft.com/en-us/download/details.aspx?id=8279
    - Microsoft Visual C++ 2010 Service Pack 1 Compiler Update for the Windows SDK 7.1
      - http://www.microsoft.com/en-us/download/details.aspx?id=4422

- **For Linux the official supported Build Platform is SLES 11 with gcc/++ 4.3.x**

Porting to Native 64bit for ISVs

- **You can expect more and more software that is available for Domino 9 64bit**
  - Some ISVs might now look into 64bit for Domino 9 because of the new compiler support
  - Porting of stand-alone applications should not be too complicated

  - Most of the changes are straight forward and C-API has own defines for most important data-types
  - If the developer did use the right C-API data types like LONG, WORD, NUMBER. DBHANDLE, applications can be compiled on 32bit and 64bit with just some changes

  - One difference: HANDLE on Windows is now 64bit and remains 32bit in Domino
    - Domino switched to DHANDLE for all generic C-API handles → code needs to be changed
    - That also means that Domino handles (DHANDLES) are still limited to 32bit
      - And there are even some 16bit handles → So 64bit does not remove all 32bit “limitations”

  - Be aware that instead of the 1-byte alignment you need the natural platform alignment
    - Another reason why 64bit needs a bit more memory

Porting to Native 64bit for ISVs

- **Not all applications can be ported straight forward**
  - More complexity in ISV applications often depends on external libs from other vendors
  - All external resources like RSA Libs, ZIP Libs, Antiv-Virus Engine Code, program-libs like the “boost lib” need to be available for 64bit
  - Sometimes interfaces between the LIBs and C-API code needs to be changed

  - **Not as straight forward than porting native C-API code**
    - The Domino team has done a great job porting the back-end infrastructure to 64bit
    - And the C-API is a sub-set of what IBM uses to build Domino

- **This is not a C-API porting session but for details about porting check the following resources**

  - C-API reference guide section “Porting 32bit Domino applications to 64bit Domino”
  - Check the current example code and make-files

SYM/PDB File Support for Add-On Products

- **Domino uses a special SYM file format integrated into one large SYM file**
  - Since D6.5.1 Domino is able to read SYM files for individual binaries
- **Lotus Development (Iris) Tool Map2iSym to create matching SYM files is part of the Lotus C-API Toolkit since Domino 6.5.1**
  - Has not been shipped with all C-API toolkits since then
- **Domino 9.0.x**
  - Domino 9 64bit still uses SYM files but uses a new SYM file format
  - But add-on/ISV applications have to use PDB files instead
  - NSD is enabled to use PDB files
  - ISVs have to ship PDB files instead of SYM files for Domino 9 64bit

Porting LotusScript to C-API Calls

- **First of all re-think if you still need the code**
  - Often you are using legacy code that could be meanwhile coded in LotusScript
- **Depending on the complexity of the C-API calls it can be quite complicated to be ported**
  - Structures passed between C-API calls
  - Callback functions
  - In many cases a native C-API solution (servertask) which interfaces with LS code is the better solution
    - More stable and easier to maintain
- **Data-Types stayed the same but pointers are now 64bit instead of 32bit**
  - The only data-type that can hold a pointer in LotusScript is a "**double**"
  - So depending on the code you will end up with separate implementations for 32bit and 64bit

Porting LotusScript to Native Platform Calls

- **Mostly used on Windows**
- **Again - First of all “re-think” if you still need the code**
- **Take care of changes data-types**
  - For 64bit Values you need “double”
  - Addresses are 64bit (you need “double” like for C-API calls)
- **There are some issues you should be aware of**
  - See details on next slides

Known Issues when Calling native APIs from LotusScript

- **Each of those issues causes the Domino Server to hang, crash or causes abnormal process terminations**
  - If you are calling any API from LS you should update to the latest fixpacks!
  - Current Status: Domino 8.5.3 FP6, Domino 9.0.1

- **8.5.3 FP3 / 9.0**
  - **SPR# EFEN8MJJCY** - Fixed an issue that would occur on 64-bit Domino platforms running an agent that integrated with C code. Prior to this fix, extra padding was added to the return value of the various data types which causes data issues going back and forth from 32-bit to 64-bit Domino versions.
  - In other words: Re-aligns the data properly for C callouts using Types.
  - **SPR# RDJS8W6QYE** - This is a companion fix to EFEN8MJJCY also fixed in this release.

Known Issues when Calling native APIs from LotusScript

- **8.5.3 FP3 / 9.0**
  - **SPR# PCHE8QLKPT** - Companion fix to SPR EFEN8MJJCY also fixed in this release.
  - Additional Info: Fix for functions returning pointers. Compiler when optimized uses a float register when LotusScript sets Double as a return value. So the return value is always NULL
  - Important: For the fix to work you need to set notes.ini **LS64BITCCALLOUTPointerSupport=1**
    - Tells LotusScript to do Callouts as if a pointer is returned, not Double.
    - If you not a C-API from LotusScript developer ignore this
      - The developer should tell you when to use this parameter

Known Issues when Calling native APIs from LotusScript

- **9.0.1 / 8.5.3 FP5**
  - **SPR# RDJS94GTVD** - Fixes Domino crash on 64 bit platforms, caused by having a LotusScript agent that makes C API calls to Operating System APIs iteratively, such as in a loop. This is a regression in 8.5.3 FP2.
  - **SPR# TTSU94HQZJ** - Fixes issue where LotusScript: Lotus C API REGNewUser registers Alternate Name of users without converting to LMBCS
    - This is a general issue with international character sets when calling C-API functions
  - **SPR# KJKJ9468AY** - Fixed a Windows Domino 64-bit Server Lotusscript OLE crash. Domino in this environment was not handling the size of parameters being passed in.
  - **SPR# JFRA8EAJGE** - Fixes Domino 64-bit Server crash on fpoplong and SlowFloatToLong.
    - Fixed in 9.0.1 only. No additional information available

LotusScript 64bit Limitation

- **Stack size for variables was not increased in Domino 64bit**
  - TN #1451119 “32k limit for string arrays applies to both 32-bit and 64-bit Notes/Domino”
  - Example: Dim myArray(1 to 8000) as String
    - 32K limit for a static string array at entry 4049.
    - On 32-bit platforms, there can be 8190 elements declared

    Work-Around: Use dynamic arrays
    - Dim myArray() as String
      ReDim myArray(1 to 8000) as String

ODBC Connections

- **You need native 64bit ODBC drivers**
  - With Domino 32bit you needed the ODBC 32Bit drivers on a 64bit OS
  - In general this will be easier than in mixed mode specially with database drivers like Oracle
- **Lotus Connector (LC) is the recommended technology**
  - LSXODBC (LS:DO) is not supported in Domino 64bit
  - Lotus Connector is the more current technology. LSXODBC is legacy code
- **Native JDBC connections work unchanged and connect directly to the remote DB**
- **“IBM ODBC Driver for Notes/Domino 9.x (for Windows 64-Bit English)”**
  - Can be used to access Domino data
  - Available for 32bit and 64bit

Migrating to Domino 64bit

- **1. Uninstall Domino 32bit**
  2. Install Domino 64bit
  - You cannot just install 64bit, Domino 32bit and Domino 64bit are two different platforms
  - Data Should remain untouched

- **Delete and re-create all physical FT indexes when the server is still down**
  - You could also switch FT Index to a different disk via notes.ini **FTBasePath=**f:\ftdir
    - New setting since 8.5.3. Makes sense for larger servers with large FT Indexes

- **All view/folder indexes need to be rebuild**
  - There are multiply ways depending on your migration scenario

Rebuilding View/Folder Index

- **a.) Via Updall**
  - Simple way: Load updall -r

- **b.) Compact / DBMT**
  - 1.) Discard all view indexes via Compact -D (can run multi-threaded in Domino 9 via -# switch)
    Example: load compact -D -# 4 mail/

    Note1: if you want to ensure all Dbs use design/data compression add **-n -v** flags
    Note2: In case databases have the wrong DB class upgrade them to current ODS via -C **-upgrade**

  - 2.) Leverage DBMT to only build most important views only
    Example l**oad dbmt -updallThreads 4 mail/**

  - Takes a while but would be more clean than just rebuilding views

Performance Tuning

- **NSF_BUFFER_POOL_SIZE_MB**
  - By default 512 MB for 32bit Domino
  - By default 1 GB for 64bit Domino
  - Some other default values depend on the buffer pool size
- **Notes.ini NSF_DbCache_Maxentries=3000**
  - default value around 3 times the buffer pool size in MB
    if you have more than n users or more than n open files you can increase the value
- **EVENT_POOL_SIZE=41943040**
  - Needed on all servers if higher number of log messages
- **Increase internal pools – also required by larger 32bit Servers**
  - CATALOG_POOL_SIZE_MB=100
  - dirman_poolsize_mb=100
  - nsf_monitor_pool_size_mb=200

General Performance Tuning for 32bit and 64bit

- **Server_Pool_Tasks=80**
  - Number of IOCP pooltasks per Notes Port
- **Server_Max_Concurrent_Trans=160**
  - maximum concurrent transactions. should be Server_Pool_Tasks multiplied by number of ports.
- **UPDATE_FULLTEXT_THREAD=1**
  - Separate Thread for Full-Text indexing
- **Disable_BCC_group_expansion=1**
  - Disables Router BBC Group expansion for performance reasons
- **FT_FLY_INDEX_OFF=1**
  - Disables on the fly FT indexing when agents use search queries on a not FT indexed DB
    - Avoids “extremely inefficient” temporary FT index
    - Agent will not run and code should be changed

Domino 9.0 - Performance Improvements

- **Source: IBM presentation at IBM Connect 2013**

- **Large UBM support for 64 Bit Exploitation**
  - e.g. NSF_BUFFER_POOL_SIZE_MB=8192 settable in notes.ini
  - Improved View Access with more Views remaining cached in memory

  Increased MAX Pool Sizes for 64 Bit Exploitation
  - MAX_NETPOOLSIZE to 1Gig (from 512 MB)
  - MAX_TASK_POOL _SIZE to 16 MB (from 4 MB)
  - NETSESSIONPOOL_SIZEMAXMB to 20000 MB (from 2000 MB)
  - MAX_GROUPCACHE_POOLSIZE to 32 MB (from 15)

- **It's hard to tell how much the extra memory will help because usually customers switch to new hardware and Domino release in the same step**

Session Summary

- **Domino 64bit Native is stable and not new anymore and you should consider it**
  - Already used for larger Traveler deployments
  - Many customers already migrated to Domino 64bit Windows
  - Domino on Intel Linux is quite new
    - But is the platform used for IBM SmartCloud® for Social Business
    - And Domino 64bit on zLinux has been also around for a while

- **It's not that complicated but needs some details to consider**
  - Specially with “custom”/add-on code
  - Your IBM Business Partners should help you with their add-on applications and custom code
    - Now you are aware what to ask for :-)

- **Native 64bit is a strategic decision**
  - You don't have to move servers now
  - But you can consider it already for your Domino 9 upgrade
  - Specially when you switch to new hardware – or start deploying virtual servers

Thank You!

Your feedback is important!

- **Access Connect Online to complete your session surveys using any:**
  - Web or mobile browser
  - Connect Online kiosk on-site

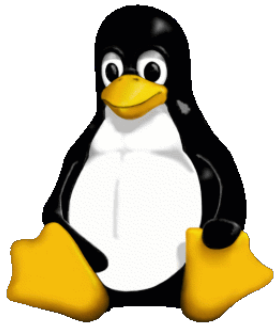
- **Questions?**
  - Now, after the session or via email

Daniel Nashed
nsh@nashcom.de
http://blog.nashcom.de
http://www.nashcom.de

Get the Line on Linuxfest V!

Back for another informative all-inclusive Linux session in 2014
Join Bill Malchisky, Wes Morgan, and guest Daniel Nashed!

When: **Thursday**, 30 January
Where: **Dolphin Hotel – Oceanic 2 (End of hallway between Asia 1 & Australia 3)**
Time: **12:30 - 1:30 pm**
Other: Bring your box lunch!
Audience: Admins, Developers, Architects
We're not in the program guide, so mark your calendar, or see our listing in the ConnectOsphere agenda Notes app



Red Hat is providing our session swag—third consecutive year

Acknowledgements and Disclaimers

Availability. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.