



# **Adm03. Successful Domino Consolidation and Cost Reduction**



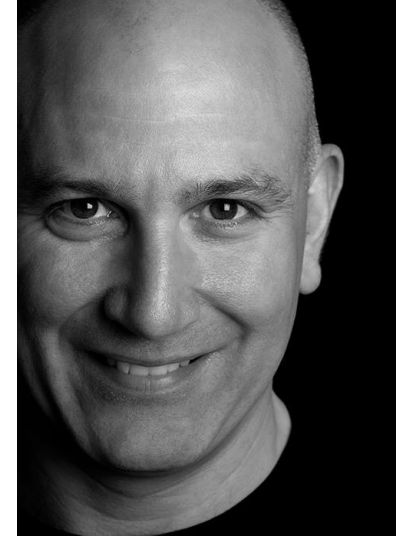
**Daniel Nashed, Nash!Com Germany**

**May 9th, 2017**

**Elisabeth Center Antwerp, Belgium**

# About “Daniel Nashed”

- Nash!Com – IBM® Business Partner/ISV
- Focus
  - Cross-Platform C-API, IBM® Domino® Infrastructure, Administration, Integration, Performance, Security, Troubleshooting and IBM® Traveler
- Author Domino on Linux®/UNIX® Start Scripts
- Contact:
  - Mail: [nsh@nashcom.de](mailto:nsh@nashcom.de)
  - Blog: <http://blog.nashcom.de>



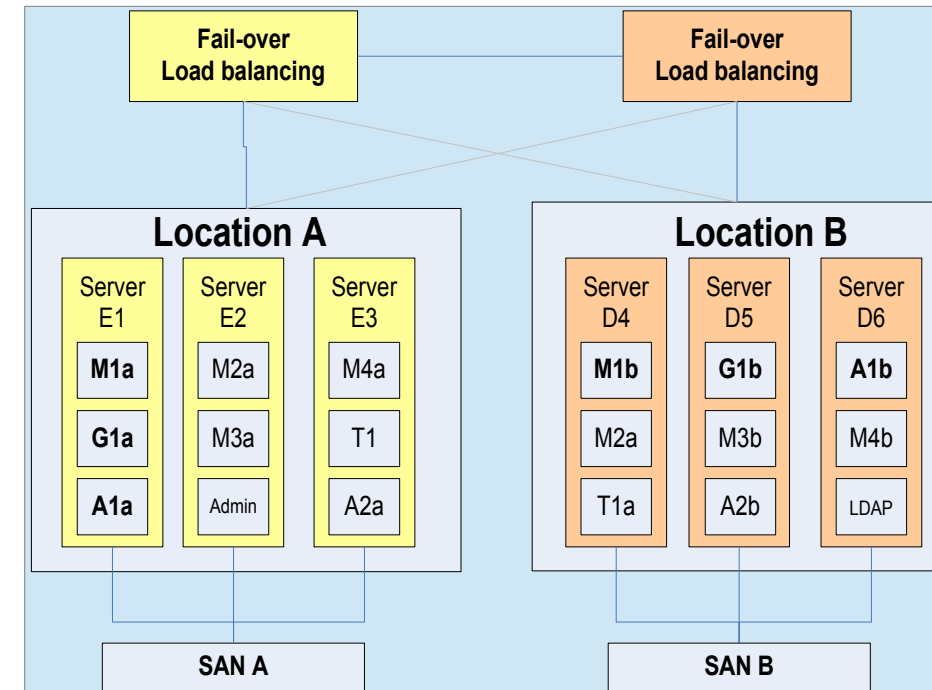
# Agenda / Key Concepts



- Right Technology
- Right Architecture
- Best Practices / Implementation
- Server Operations
- Q&A

# Classical Deployment

- Physical machines with single or partitioned servers used to be the standard deployment model
- Single Server usually cannot take full benefit of the full hardware resources
- Partitioned servers can better use the resources of a single machine with just one OS level instance
- Still often used with local disk or SAN storage
- Today most customers are moving to “virtual servers” for standalone and clustered servers



# Benefits of Virtual Server Deployment Model

- Faster OS deployment and virtual hardware deployment
- Better use of hardware resources and more granular control of resources assigned
  - **Specially for smaller servers → Less idle resources or server oversizing**
  - Load-balancing can move machines to other hosts, etc
  - Ease of adding resources like RAM and CPU
- Standardized machine layout and installation
  - Virtual hardware has the same drivers on OS level for every machine etc
- High availability by moving machines to different virtualization hosts
  - **That also means → Switching hardware without downtime**
- Storage allocation and migration without downtime and impact for Domino
  - You can even move from one storage pool to another without downtime with ESX servers
- New backup options depending on used storage and configuration

# Server Consolidation in the virtual World

- Server consolidation does not always mean to put as much users as you can on a single Domino server
  - **Consolidation often means to host your servers centrally in a standardized infrastructure with standardized configuration, optimized to make best use your infrastructure**
- In most cases you reach cost reduction by optimizing
  - Server-deployment
  - Server-update
  - Servers maintenance
  - Server operations
  - Resource usage, etc.
  - Standardized client deployment and management

# Server Sizing

- Too large servers might not have best resource usage
  - Serve size does not only depend on number of registered users
  - But can also depend on
    - **Storage used**
    - **Concurrent users**
    - **Number of heavy users!**
    - **Number of Mobile device users → Traveler**
  - Note: Large mail files, **document count** and specially docs in **inbox** has impact on scalability!
- Usually a good size of a single Domino mail server instance is 1500 users
  - In a cluster environment **1500** per home-server and **3000** mailfiles per server (split between servers)

# Consolidation Pro and Contra

- Server Consolidation to a single location or a “few” central data-centers is not always the best or only approach
- You need to carefully analyze and take into consideration multiple aspects
  - Network bandwidth
    - Other services that also use the network at the same time!
  - Network latency!
  - **Infrastructure** you have in those locations
  - Applications you are hosting
  - Access method → e.g. Notes Client, Web, Mobile



# Consolidation Pro and Contra

- If you have a server infrastructure (including backup) for other applications in a smaller location, having local servers is still a good solution!
  - Can and should be a virtual environment with a smaller Domino server
  - Improves performance for end-users and reduce network traffic!
    - Access to databases in a local network
    - Documents/Mails are just routed once for multiple recipients
    - **DAOS will not send the attachment if already stored on server for another user**

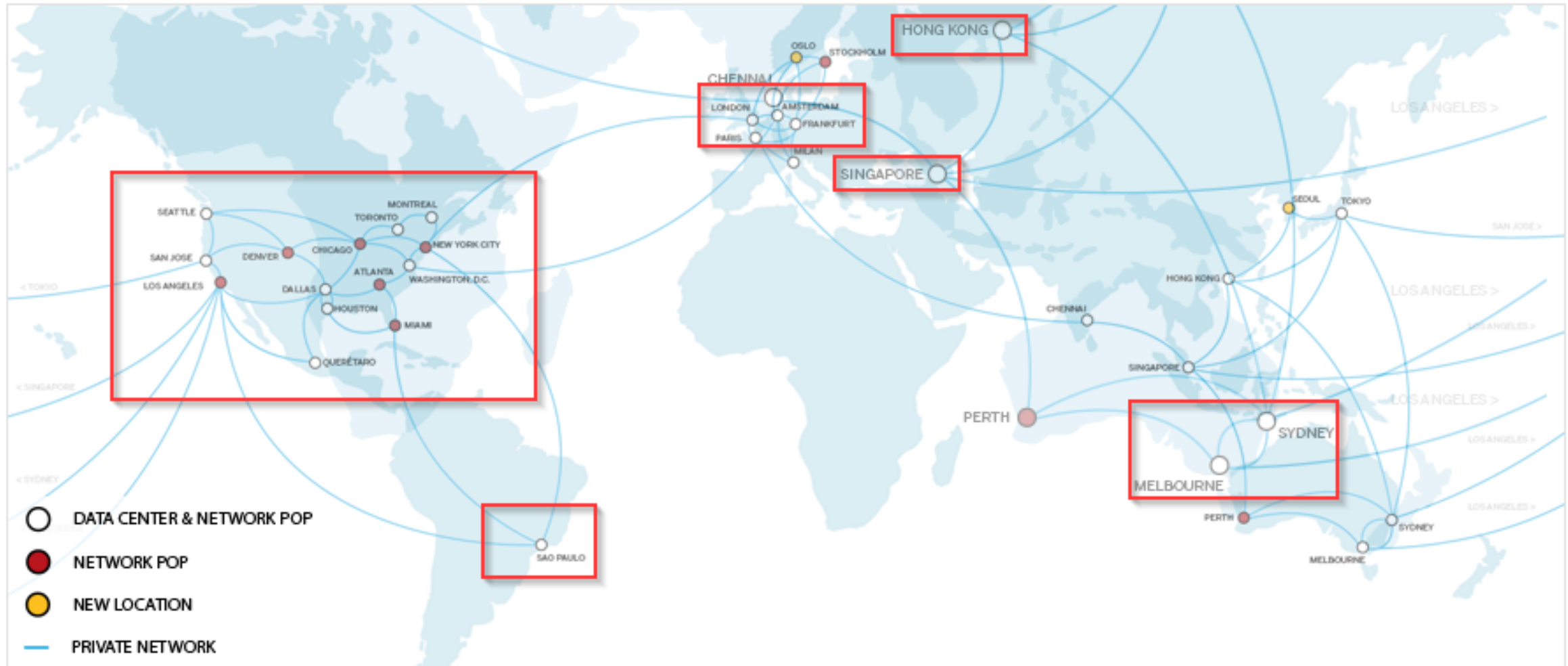
# Network Considerations for Consolidated Environments



- Different Networks have different latency
  - LAN = around **1 ms**
    - Take care: Wireless LAN connections can have different performance already!
  - WAN in same country depending on network around **50 ms**
  - WAN in Asia or other remote locations up to **400 ms**
- Speed alone does not matter if you have many small transactions
  - Network Accelerators can help to some extent specially for re-transmissions, data flow optimization, ..
    - But don't expect miracles!
  - Notes client **serializes** transactions
    - Web client can request multiple resources at the same time

# Data-Center Global Locations – Example: IBM Softlayer

- For an international company you should have least data centers in every continent
- Network connectivity and low latency is the most important aspect!



# Network Optimization Tips

- Network Compression
  - a.) Either Notes port compression
  - b.) Or compression on router level / Network appliance
    - Don't use both usually network level works best, if available
      - If you use **Notes port encryption**, enable **Notes port compression** as well because encrypted data cannot be compressed
- Take care about MTU size and auto MTU discovery
  - Specially when using VPN connections and mobile networks
  - MTU size needs to be aligned with smallest MTU of the connection
  - MTU Auto discovery might not always work, Notes/Domino uses the full MTU size
  - **You should check with your network administrator**

# Applications and Latency

- For complex applications already 50 ms make it difficult to work
  - Example: 100 NoteOpen Transactions
    - in LAN = 100 ms
    - in WAN is already 5 seconds
- Multiple ways to optimize performance
  - Optimize Network → not always possible
  - Optimize Application
  - User different access method like **Web** or **Citrix**
  - Offload operations to the server (e.g. agent run on server)

# Some Application Pitfalls for WAN Connections

- Avoid Private on First Use Views/Folders
  - If you have more than 40 design elements with the same name the internal design cache breaks.
  - See Technote for details → #1322578, SPR #RSTN7K2EM4
    - Email me for a tool to test if the design cache is broken for a database
  - When the design cache is broken fall-back code is used to find design elements
  - Instead of **FINDDESIGN\_NOTES** transactions the design collection is opened and searched sequentially until a match is found → This overhead breaks your performance with WAN connections!
- ComputeWithForm” does not use cache.ndk
  - It even does not leverage the newer finddesign transaction but does a lookup in the “**design collection**”
  - So every ComputeWithForm does read all forms, sub-forms, shared fields which are involved ...
  - A lot of overhead. Maybe OK for a single ComputeWithForm but refreshing multiple documents can be painful
- **Idea/Workaround:** Use Run on Server Agents off-loading the work to the server
  - This can be a good approach for other operations that need many transactions

# Optimizing “Mail”

- Local replica or managed replica can help and has other advantages
  - Work independent from server, have same configuration for mobile and desktop users, etc.
  - But usually mail application design is locally cached and performance is quite good in WAN environments
  - Good starting point for “Managed Replica”  
[https://www.lotus.com/ldd/dominowiki.nsf/dx/IBM\\_Lotus\\_Notes\\_and\\_Domino\\_8.x\\_local\\_mail\\_replicas\\_Advantages\\_considerations\\_and\\_best\\_practices](https://www.lotus.com/ldd/dominowiki.nsf/dx/IBM_Lotus_Notes_and_Domino_8.x_local_mail_replicas_Advantages_considerations_and_best_practices)
- iNotes or Verse on Prem (VOP)
  - Access via browser has different challenges but invokes multiple requests at the same time
  - VOP offers cache and offline capabilities
    - Easy setup but with local browser limitations
  - For high availability you need a load-balancer or reverse proxy

# REF - Trace Notes Client Transactions



- Client\_Clock=1
  - Can be used to track Notes Client/Server Transactions (NRPC)
  - Notes.ini debug\_outfile=c:\debug\_notes.log
- Client\_Clock logs
  - Transaction sequence
  - Transaction name
  - Transaction data (ReplicaID, NoteID)
  - Response time (ms)
  - Bytes send, received
- Example:
  - (15-78 [15]) OPEN\_NOTE(REPC1256B16:0072BCBE-NT00000E3E,00400020): 0 ms. [52+1454=1506]
- Free annotation tool OpenNTF Client\_Clock Parser
  - <http://www.openntf.org/Projects/pmt.nsf/ProjectLookup/Notes%20RPC%20Parser>



# Ref – Notes Transactions

START_SERVER	Start User Session
OPEN_DB/CLOSE_DB	Open/Close a database
OPEN_NOTE/NIF_OPEN_NOTE	Open a Note
UPDATE_NOTE	Update a Note – there is no close transaction
OPEN_COLLECTION/ CLOSE_COLLECTION	Open/Close a view/folder collection
READ_ENTRIES	Reads data from a view/folder
UPDATE_COLLECTION	Updates a view/folder collection
FIND_BY_KEY	Finds notes in a view/folder collection
FINDDESIGN_NOTES	Finds design notes
SEARCH	Search operation with formula
GET_MODIFIED_NOTES	Find table of modified notes
GET_ALLFOLDERCHANGES_RQST	Get changes in all folders Unread count in mail folders

# Ref – Notes Transactions

NAME_LOOKUP	Lookup information in Domino Directory
ALLOC_UPDATE_OBJECT	Create or update an object
READ_OBJECT	Read data from an object
WRITE_OBJECT	Write data to an object
READ REPLICATION HISTORY	Read Replication History
DB_INFO_GET	Get database info buffer
GET_NOTE_INFO	Get Note information
DB_MODIFIED_TIME	Get the modified date of DB
DB_REPLINFO_GET	Get Replication info
POLL_DEL_SEQNUM	Get Delivery Sequence Number
DB_GETSET_DEL_SEQNUM	Get or set Delivery Sequence Number
GET_SPECIAL_NOTE_ID	Get a special Note-ID by number

# REF - Server Clock – Trace Transactions on Server Side



- **"Show trans"** shows transactions categorized
  - Tip: **"Show trans reset"** resets transaction counters
  - Can be used to get an idea about transactions and transaction types
- **"Server\_Clock=2"** dumps all transactions to server console/ console.log
  - Nicely delimited output, that can be parsed :-)
  - Example: 01.01.2017 12:37:57 39789504 ms 'OPEN\_NOTE' 0 ms (0 ms NETIO) TCPIP 00010EDC Rcvd 0 Sent 3026 User 'Full Admin/NashComLab' Db 'mail\fadmin.nsf' Ip '192.168.100.101' Lm 1

# Platform Focus / Architecture



- **Windows 64bit** and **RHEL Linux 64bit** are currently the two strategic IBM platforms for Domino and other collaboration products
  - All other currently supported platforms continue to be supported in their release life-cycle (current plan)
    - For example Domino 9.0.1 with all planned new Feature Packs will be supported!
    - **No Linux 32bit** and **AIX 32bit** for 9.0.1 FP8 → It's best practice to move to native 64bit anyway
    - New consolidation environments should be based on one of those two platforms!
- Widely used by customers with physical and virtualized servers
- Most of our customers using Domino 64bit on Windows or Linux on the ESX Server Platform
  - We see far less customers on other virtualization hardware
  - ESX was the first "Intel" platform supported by Domino and is a "Tier 1" virtualization platform
  - Works best with **64bit** applications

# REF - IBM Domino 9.0.1 Virtualization Platform Support



- **VMware ESX and ESXi 4.0, 5.0, 5.1, 5.5 & 6.0 (including future Fix Packs) wherever VMWare ESX and ESXi are supported**
  - Most common used hypervisor for the Intel platform
  - Very stable large enterprise deployments with Domino and other IBM products
- KVM: KVM support is based on KVM/Kernel level. Support for level 2.6.32 x86\_64 as baseline on either RHEL or SLES.
  - Automatically support patch releases beyond this level as well (but not feature releases).
- Microsoft Hyper-V Server 2008 R2
- Microsoft Hyper-V Server 2012 & 2012 R2

# Domino Clustering (& Traveler HA)



- Why use Clustering even if we have Virtual Servers?
  - Email and Domino Applications are mission critical for most customers (24x7 service)
  - It's not just about the TCO of your servers.  
It's more about the cost of **not having** a **service** (not a certain server) available when you need it!
- Total availability depends on
  - a.) Planned Down-Time and
  - b.) Unplanned Downtime for your Domino or Traveler Service
  - You cannot reduce both to zero
- Planned downtime for maintenance and updates helps to keep the server stable and at good performance
- Virtualization does not protect against all type of down-times and also not against Domino Server crashes, database corruptions, planned downtime, etc  
→ **Domino Clustering is still the best choice for 24x7 operations!**

# Tips for Clustering

- Notes Clients are “cluster aware” and can silently failover to another cluster mate!
  - Still not enabled by default!
  - Client silent failover: notes.ini **FailoverSilent=1** or via Desktop Policy
    - Ensures client failover without prompting → also works when saving open documents!
- Usually you don't need to mirror your SAN data to another data-center when clustering on Domino level
  - Domino “cluster-mates” should be located on other ESX infrastructure – e.g. different data center in the same country
- Port Stealth Mode on Windows should be disabled!

# REF - Port Stealth Mode Windows 2008 and higher

- Port-Stealth Mode is a firewall independent behavior to protect server ports
  - If an application does not respond, packets are dropped
  - Leads to long client wait times until the client fails over to another server
  - Can be disabled → Needs reboot after registry is changed!

## • References

- <https://www.ibm.com/support/docview.wss?uid=swg21498755>
- <http://msdn.microsoft.com/en-us/library/ff720058%28v=prot.10%29.aspx>

## • Registry Settings

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Microsoft\WindowsFirewall\DomainProfile]

"DisableStealthMode"=dword:00000001

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Microsoft\WindowsFirewall\PrivateProfile]

"DisableStealthMode"=dword:00000001

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Microsoft\WindowsFirewall\PublicProfile]

"DisableStealthMode"=dword:00000001

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Microsoft\WindowsFirewall\StandardProfile]

"DisableStealthMode"=dword:00000001



# Additional RAM helps to reduce Read I/O



- Domino itself usually does not need more than 4 GB RAM for all processes
  - Usually only large application Web servers and large Traveler servers need more RAM based in the application and load
- Additional RAM will be used as file-system cache on a 64bit OS!
- Dramatically reduces I/O read operations even with Domino random I/O pattern
- Additional 16 – 64 GB RAM can make sense
  - Really depends the type of server and how much RAM you have left in your ESX server
  - Usually 64 GB RAM improves performance already

# Customer Example Linux 64bit on ESX with a lot of RAM



- Customer configured 1 TB RAM per Domino instance and had almost zero read I/O during the day
  - Customer business case: High speed for Adminp Processing for a full organization rename
  - Most Operations have been improved 20 times! → Usually great benefit already with 64 GB RAM

```
top - 16:49:40 up 11 days,  2:36,  1 user,  load average: 0.20, 0.27, 0.33
Tasks: 415 total,  1 running, 414 sleeping,  0 stopped,  0 zombie
Cpu(s):  0.7%us,  0.2%sy,  0.0%ni, 99.1%id,  0.0%wa,  0.0%hi,  0.1%si,  0.0%st
Mem:  1032574064k total, 1030662104k used, 1911960k free, 5423572k buffers
Swap: 2097144k total,  628k used, 2096516k free, 1015683544k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
15777	notesp2	18	0	3025m	1.3g	1.2g	S	21.9	0.1	984:23.84	server
18050	notesp2	18	0	2906m	1.0g	1.0g	S	4.0	0.1	217:10.31	update
18049	notesp2	15	0	2952m	1.1g	1.0g	S	2.0	0.1	62:09.21	router

# Large File System Cache for 64 Bit Windows



- By default on Win64 there is a very high physical memory limit for the file-system cache
  - It will try to use all memory for file-system cache which can cause Domino Memory to be swapped out
  - Therefore Domino uses Windows 64bit call "**SetSystemFileCacheSize()**" to limit the cache
    - See TN #1391477 for details
    - cacheset.exe -g will show the settings
- 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"
- Recommendation: 4-6 GB for OS & Domino itself, configure remaining memory as cache

# Linux File-System Tuning

- Disable write of meta information via mount option `-noatime`
  - Meta information like last accessed time
- A real `Runfaster=1` Parameter
  - Change the default scheduler from CFQ (complete fair queuing) to NOOP
  - CFQ tries to optimize disk access by reordering requests
  - But it would be better to send it to a SAN, RAID controller directly
  - Tests have shown that this works better for almost all SAN or local disk configurations
- Disable per device
  - `echo noop > /sys/block/hda/queue/scheduler`
- Disable globally via kernel boot parameter
  - Edit `/boot/grub/grub.conf` and enter in kernel line **elevator=noop.**

# View Rebuild and Notes\_TEMPDIR

- View Rebuilt will optimize view update
  - Notes.ini: **view\_rebuild\_dir=/dev/shm/view\_rebuild**
    - Should have at least 20 GB “free” space
- You can move the Notes tempdir to a separate disk or tmpfs on Linux
  - Notes.ini: **Notes\_tempdir=/dev/shm/notes\_temp**
- On Windows best location is either separate file-system/virtual disk
  - Or system disk if you have sufficient space free!
- On Linux leverage “tempfs”
  - Works “similar” to a RAM disk but only needs RAM for used space
  - Enabled by default with 50% of your RAM, mounted by default in **/dev/shm**
  - Does swap to disk if memory can be better used by processes or file-system cache → very optimized
    - Ensure you have sufficient swap space → At least 50% of your physical memory
  - Tip: Nash!Com Start Script has options to create sub-directories automatically at server-startup

# Disk Requirements in the virtual World

- In virtual environments we usually don't have much influence on disk configuration
  - But you can explain Domino requirements and usage patterns to help optimize
- In earlier days virtual servers with large storage used direct SAN storage via iSCSI etc.
  - Today most customers use VMDK with larger RAID10 NAS/SAN Volumes
- It's still a good idea to split data into virtual disks similar to physical machine deployments
- They might be on the same RAID group but too much write concurrency in the same VMDK causes contention!
- Recommendation
  - Translog should be on separate VMDK or on system VMDK
  - NSF Data should be on separated VMDK
  - DAOS should be on separate VMDK
  - FT and NIF (new in FP8) should be on separate VMDK

# REF - File System Recommendation “TRANSLOG”

- Size: 4.1 GB for Circular Translog
  - For archive-style translog size highly depends on your data
  - For linear style translog size should be in the range of 8-20 GB
- Recommended: RAID1
- Allocation Size: 4K
  - Use larger disk block size and matching Stripe size
- Access Pattern
  - TX logging writes fixed sequential 4k blocks
  - Transaction log files are opened in a synchronous mode
  - OS file system cache is not used

# REF - File System Recommendation "NOTESDATA"



- Recommended: Fastest Disk – Usually RAID 10 instead of RAID 5
- Allocation Size: 16K
- Access pattern:
  - Random Access I/O – 16K blocks
  - Files Opened Fully Buffered
  - Domino does not do simple predictable sequential reads
  - Read Ahead Caching should be set to a low value
    - Unrequired read-aheads can increase I/O traffic, as well as filling OS/SAN etc cache buffer with unwanted blocks
    - You need read ahead caching for backup and some read ahead also improves Domino performance
- Faster drives perform disk I/O in less time
  - Choose drives with a combination of low seek time & high RPM



# REF - File System Recommendation “DAOS”

- Recommended: RAID5 to save costs (if available)
- Allocation Size: 256K up to the size of an attachment – several MB
- Access pattern
  - Sequential I/O Read – 16-32K blocks
  - Sequential I/O Write – 16-32K blocks
  - Only occurs with new object – static afterwards
- Read Ahead Caching depends on the performance you need
  - You need read ahead caching for backup and normal operations
  - In most cases more read than write cache makes sense
- Entry Level Storage is sufficient from performance point of view
  - Lower I/O rate and specially lower IOPs than what we see for NSF

# REF - File System Recommendation: "FT" and "NIF"

- Same recommendations as "NSF"
- Similar access pattern than "NSF"
- Should be a separate file-system / disk
  - Depending on size you can have one file-system with different directories
    - For larger servers separate virtual disks can make sense

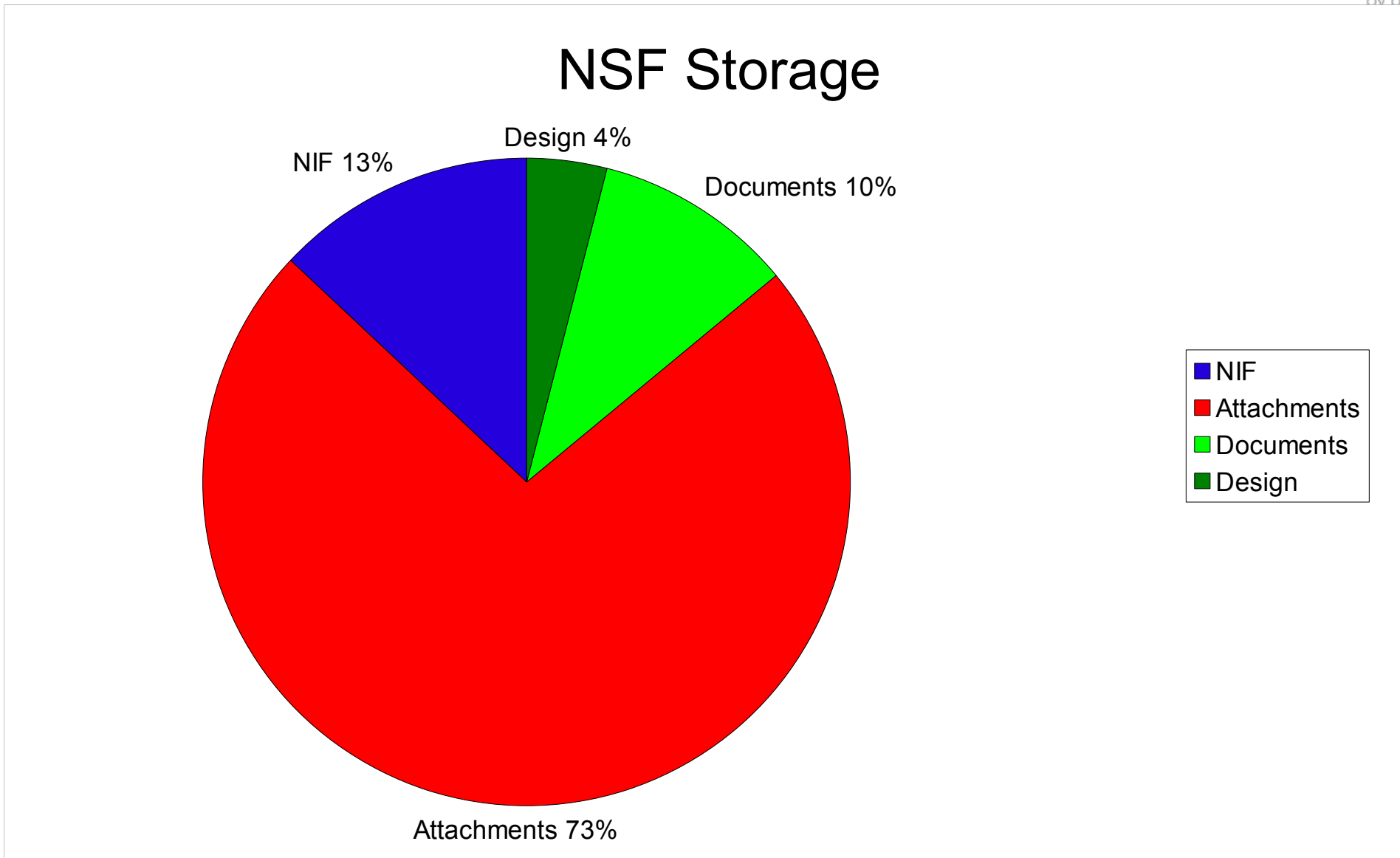
# REF - Best Practices: FT Index

- UPDATE\_FULLTEXT\_THREAD=1
  - Separate Thread for Full-Text indexing
- 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
- FTG\_USE\_SYS\_MEMORY=1
  - Use system memory instead of Domino pooled memory
- Avoid using “Immediate FT index” if possible  
Don't index attachments if possible!
  - If you have to use exclude lists for attachment types (Example below)
    - Notes.ini FT\_INDEX\_IGNORE=\*.asf,\*.avi,\*.bin,\*.bmp,\*.CSS,\*.dat,\*.DXL,\*.EOT,\*.EPS,\*.FLV,\*.EPS,\*.HTC,\*.ICS,\*.iso,\*.JAR,\*.JS,\*.mpeg,\*.MP4,\*.PNG,\*.rar,\*.SVG,\*.SWF,\*.TIF,\*.TTF,\*.WMV,XML,\*.XSL,\*.zip

# Virus Scanners & NSF Files

- You should disable on access virus scanning for all Notes related files!
  - Huge impact on performance and stability!
  - Use Domino aware virus scanners instead!
- Either exclude directories if possible or extensions!
  - Important extensions: **\*.NSF, \*.NTF \*.BOX, \*.DTF, \*.ID, \*.FT**
  - Translog: **\*.TXN**
  - DAOS: **\*.NLO**
  - New extension: **\*.NDX** for view index!
  - Extension used for copy style compacts: **\*.ORIG, \*.TMP**

# Database Optimization – NSF Storage



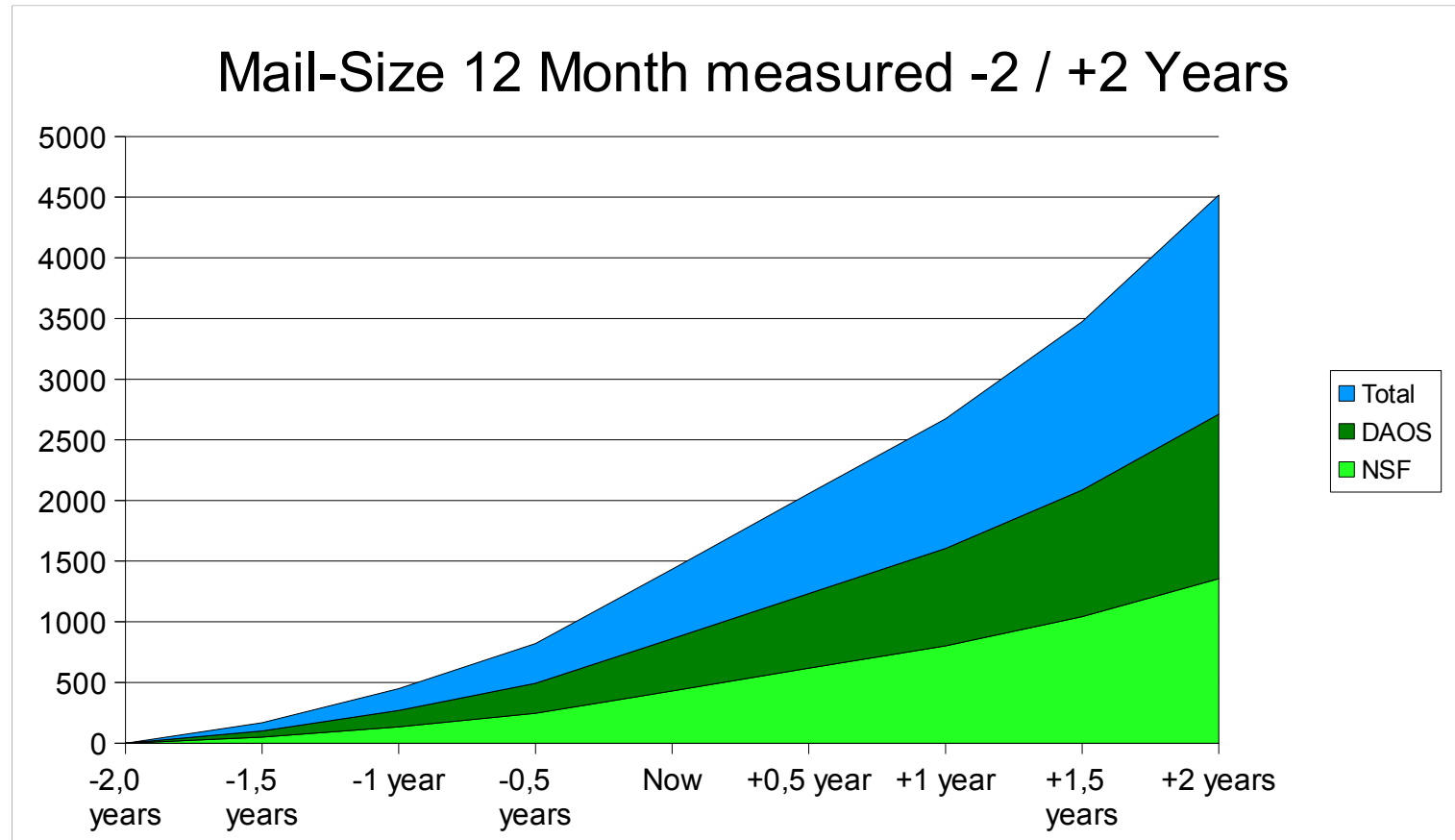
# DAOS Storage & Deduplication



- DAOS is supported on NAS storage even leveraging mounts (NFS and Windows Shares)
- Some NAS storage like NetApp supports storage **deduplication** in the same storage container
  - Deduplication usually works on block level (e.g. NetApp uses 4 KB blocks)
  - If you have multiple mail-servers or archiver servers deduplication can reduce storage dramatically!
  - **Only works if disabling NLO encryption** → notes.ini **DAOS\_ENCRYPT\_NLO=0**
  - Also reduce the number of NLOs per sub-folders (default : 40000)
    - Notes.ini **DAOS\_MAX\_FILES\_PER\_SUBCONTAINER=20000** (or less in smaller environments)
- Tip: Some backup solutions like Tivoli support deduplication if configured
  - Speak with your backup team how to backup DAOS in the most efficient way!

# DAOS Example Results

- 70% reduction of NSF
- 25-35% total reduction by deduplication



# Database Optimization

- Database Design Compression
  - **50%** of Design → **2%** total reduction only, but it reduces the I/O!
- Document Compression
  - **50%** of Summary data → **5%** total and reduction of I/O
- Tip: Enable via compact, recover space with DBMT
  - Compact -c -n -v mail
  - In case you are using DBMT use without -c to set the options and have DBMT compact and compress data/design
- DAOS
  - **70%** NSF File size reduction, **25-35%** data reduction by deduplication



# Fulltext Index on separate disk

- Since 8.5.3 full text index can be move to separate directory / disk
  - Specially useful to separate storage in larger environments and to distribute I/O
  - Also helpful when using snapshot backup
- Fulltext index can cause disk fragmentation
  - FT Index should be rebuild regularly → **See DBMT slides**
  - Separating FT index reduces disk fragmentation on “NSF” disk
- Notes.ini **FTBasePath=d:\full\_text**
  - Restart Server after moving FT Index
  - Rebuild of FT index will move the FT index data

# “NIFNSF” – Move NIF outside NSF

- Available since 9.0.1 Feature Pack 8
  - But you should run at least 9.0.1 FP8 IF1 → Performance issue introduced with NIFNSF
    - Fix → SPR #SWASAKELQ8 Perf issue w/ nifnsf enabled
- Moving NIF (View Index data) outside the NSF
  - Reduces backup data and optimizes disk I/O and database locking
  - Allows further split of data into different file-systems
  - **Database locking optimization is planned for FP 9**
    - More parallel NSF and NIF operation possible
- NIF is stored in one separate “.NDX” file per database
  - Example Results: Mail File NIF size: ~ 10% of the database size
  - With DAOS ~ 30% of the remaining physical file!

# “NIFNSF” – Move NIF outside NSF

- Requires Translog
- Requires at least ODS 51 better ODS 52
  - Notes.ini CREATE\_R9\_DATABASES=1
- Notes.ini **NIFNSFEnable=1**
  - Enables NIFNSF → restart your server
- Notes.ini **NIFBasePath=path**
  - Optionally move index files to separate directory / disk
  - **Multiple Options**
    - a.) relative path in Data-Directory
    - b.) absolut path
      - Depends on your strategy and storage requirements
- Notes.ini **CREATE\_NIFNSF\_DATABASES=1**
  - Creates new databases with NIFNSF enabled

# NIFNSF – Commands

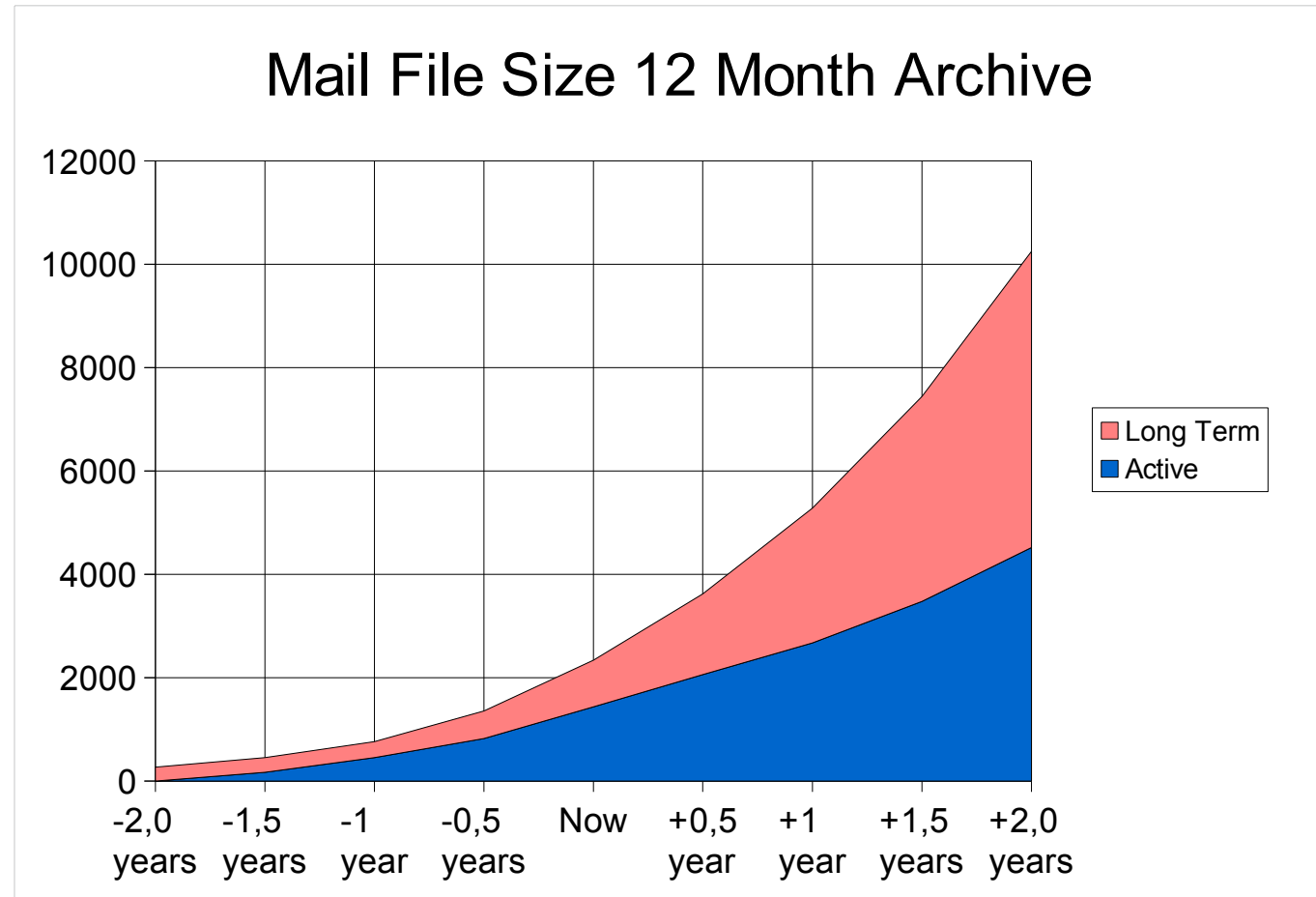
- Enable / Disable via compact
  - load compact -c -nifnsf on db.nsf
  - load compact -c -nifnsf off db.nsf
- Server Console Commands
  - **show dir -nifnsfonly**
    - Shows all NIFNSF enabled databases
  - **show dir -nifnsf**
    - Shows additional NIFNSF information

# NIFNSF - Tuning

- NIFNSF needs separate File-Handle for NDX file!
- NDX file is “sort of a database” and needs DbCache Entry!
- Default Max Cache-Entry Entries = ~ NSF Buffer Pool in MB \* 3 ~ 1024 \* 3 ~ 3072
- Can be increased if needed via Notes.ini **NSF\_DbCache\_Maxentries=n**
- Check you statistics before and after enabling NIFNSF
- Statistics
  - Database.DbCache.**CurrentEntries** = 1173
  - Database.DbCache.**HighWaterMark** = 1189
  - Database.DbCache.**MaxEntries** = 3072
  - Database.DbCache.**OvercrowdingRejections** = 0 → should be always zero

# Server based Mail-Archiving

- Without quotas and archiving mail data grows exponentially
  - Doubles every 9 to 18 month depending on customer environment
- DAOS can help but in most cases archiving data to an Notes based Archive Database makes sense!
- Data size and also **number** of documents will be reduced
- Archive is integrated in Mail-DB UI
  - One click to switch to Archive
  - Global Search in Standard Client UI
- Server based archive can be enabled by archive policy



# Recommendation Server based Mail-Archiving



- Archive Database can and should be DAOS enabled
- Depending on environment archive is on same mail-server or dedicated archive server
  - Two Different approaches:
    - a.) Leverage DAOS to deduplicate attachments between mail and archiv → Same server
    - b.) Consolidated Archive Server without clustering → Different server, with weekly/monthly backup
      - Customer example: 8 mail-servers, 2 archive servers. Home server group used to assign archive policy with target archive server
- Compact -a / -A is used to invoke archiving
  - Can be scheduled once per week → Backup is only needed once per week!
  - Important: **Compact -a -DAOS ON** .. should be used to enable DAOS on archive database
  - You can either recover space in combination with the archive compact or leverage DBMT

# Domino Backup Requirements

- Either use Domino supported solution with Domino backup agent (based on C-API)
- Or shutdown Domino Server for backup
  - Does only make sense if you are using snapshot backup → very short downtime
  - Tip: NetApp has a fully supported backup solution for Domino
    - Brings all databases into backup mode, creates a “snap-shot”
    - Ref -> <http://www.netapp.com/us/solutions/applications/lotus-domino/>



# Domino Backup

- With “storage optimization” Archive Style Translog and Incremental NSF Backup might not be the only good choice
  - Makes still sense if you need **point in time recovery**
  - Else a nightly full backup could make sense as well
- You should backup both cluster mates!
  - Tip: User alternate nightly backups → Backup every second night per server to reduce the risk without increasing the data
  - Important: Take into account your backup software license model

# Optimize Daily / Weekly Operations

- Newer Tools like DBMT allow optimized operations
- You should review your current strategy and operations
- Often maintenance, backup and archiving are not aligned
  - Depending on your configuration this can make a big difference
- New tools have shorter nightly runtime and more flexible options
  - Some ideas, tips, .. on the next slides with reference slides for more information

# Database Management Tool - DBMT

- Multi-threaded Domino Server Add-In Task
- Performs automated coordinated administrative tasks on databases
- DBMT replaces the need for running Compact on Non-System databases
- Standard Maintenance
  - Copy-style Compacting (compact) → Only copy-style compact!!
  - Fixup of Corruption (fixup)
  - Full Text Indexing (updall)
  - Incremental View Indexing (update)
- Additional Maintenance
  - Folder Reorganizing
  - Deletion Stub Purging
  - Expiring of Soft Deleted Entries
  - Updating of Unread Lists

# DBMT Example

- `dbmt -compactThreads 4 -updallThreads 4 -ftiThreads 4 -ftiNdays 30 -range 1:00AM 6:00AM -compactNdays 7 -force 1`
  - Compact all databases not compacted for 7 days with 4 threads
  - Update all views with 4 threads
  - Update FT Index with 4 threads, Rebuild if corrupt or older than 30 days
  - Run between 1 AM and 6 AM every night and stay in memory during the day
  - Fixup database which did not compact successfully 5 times on Sunday

## Additional Information

- See additional slides at the end of the presentation for more details about DBMT

## D9.0.1 - DBMT – File Space Pre-Allocation

- A copy style compact generates a temp file and copies the database note by note into the database
  - Allocation of file-space is also done step by step
  - Depending on the file-system, free space database is already fragmented when the compact finishes
    - It is very likely that database has many fragments
- DBMT will **pre-allocate** the calculated space that the physical database needs
  - It's DAOS aware
  - Enabled by default in 9.0.1
  - Notes.ini **DBMT\_PREFORMAT\_PERCENT=n** allows you to increase or decrease the size
  - For example DBMT\_PREFORMAT\_PERCENT=120 will have 20% contiguous free space in the database after compact (if there is sufficient unfragmented space in the file-system)

# REF - Best Practices: Performance Tuning for larger Servers



- **NSF\_BUFFER\_POOL\_SIZE\_MB**
  - By default 1 GB for 64bit Domino
- **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**
  - **CATALOG\_POOL\_SIZE\_MB=100**
  - **dirman\_poolsize\_mb=100**
  - **nsf\_monitor\_pool\_size\_mb=200**
  - **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)

# Tips for Server Operations and Configuration



- User groups for different server types → use them in config docs and program docs etc
- Have standard notes.ini parameters in default config doc
  - Put Notes.ini for groups of servers into config doc for your server group  
Set Notes.ini parameters per server locally
  - Priority of config docs: default → assigned by group → per server config doc
- Use Client Policies where possible
  - Desktop, Security, Mail, Server based archiving
- Leverage DDM for pro-active server monitoring
- Implement current fixes and patches regularly with a maintenance plan
  - In Q1/2017 IBM started to ship "Feature Packs" 3-4 times per year

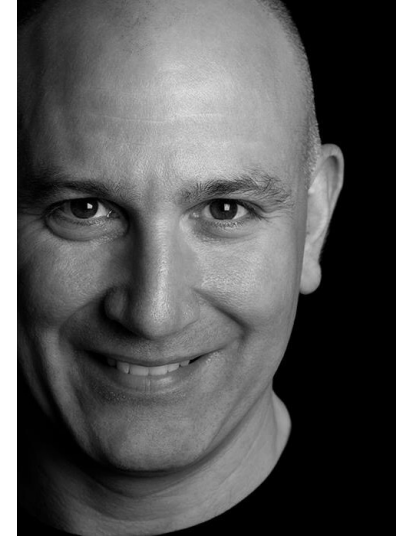
# Automated Server Install

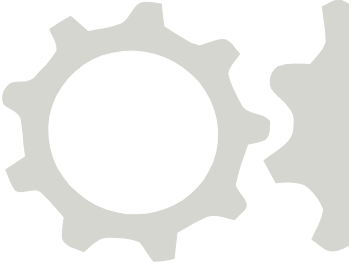
- Domino and Traveler installers have silent and response file installs
- Some customers build their own install and update scripts which can automatically update servers from an software install share
- I build a shell script for automatically checking Domino version and installing updates accordingly.
  - Including language pack, JVM patches, uninstalling Interims-Fixes and Hotfixes :-)
  - Ensures quicker and consistent server updates specially in larger environments
- Example command line for Linux Silent Installs
  - Domino 9.0.1: `./install -options domino_response.txt -silent > "domino.log"`
  - Fixpack/IF: `./install -script script.dat > "fp.log"`
  - JVM Patch: `./install -i silent > "jvm.log"`



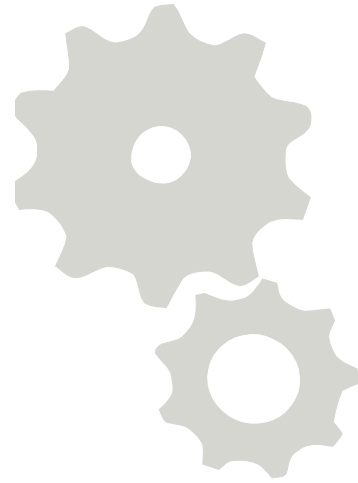
# Thank you

- Thanks for your attention!
- Fill out your evaluations to give feedback!
- Find me during the conference or contact me later
  - Mail: [nsh@nashcom.de](mailto:nsh@nashcom.de)
  - Blog: <http://blog.nashcom.de>
- Check Additional Slides appended to this presentation!





# Additional Material



# DBMT Limitations & Tips

- For Compact Options you cannot specify how much Space should be free for compact to run (-S parameter)
  - But you can configure how often a database should be compacted ()
- Compact only runs in copy-style compact
  - Works fine in a cluster
  - Else you need to make sure compact only runs on weekends/ middle of the night
- You should only have one DMBT with -range Option
  - No second DMBT for the weekend with a bigger time window
  - Means you should keep compact operations separate and run them during the day on weekends
- In case you have archive style transaction logging you need to schedule compacts for the weekend anyway
- Ensure you have the right quota setting in the Translog tab:
  - "Check space used in file when adding a note"
- Command Line Options in Program Docs is limited to 128 chars!

# Mail File Disable Compact Abort

- Since Domino 8.5.3FP3
- By default, compact operation will be aborted when mail is delivered
  - To make DBMT more efficient set notes.ini **MailFileDisableCompactAbort=1**
  - New mail in mail.box will sit undelivered in mail.box on the Domino Server
  - Mail router will periodically retry delivering new until the current compact completes
- Especially for large mail files, this may be an unacceptable extended period of time to go without receiving new email
  - But there is another new option in Domino 9.0 (see next slide)
- Best practice is to set
  - MailFileDisableCompactAbort=1

# Mail File Enable Delivery Failover



- Since Domino 9.0, mail Router has ability to failover mail delivery to an Available mail file replica on a cluster mate when the local replica is unavailable because:
  - Copy-style compact of mail file is in progress
  - Fixup of mail file is in progress (Performing consistency check on...)
  - Mail file is missing (File does not exist)
  - To enable delivery failover, set **MailFileEnableDeliveryFailover=1** in notes.ini on Domino Server
  - **DBMT** uses the replica index in clbdir.nsf to determine if it should compact the mail file on this Cluster Mate on this day to make sure all Cluster Mates are not compacting same replica on same day
    - **Only works in combination with -compactNdays**
  - **\$MailClusterFailover** item will be appended to the note with a value of the Domino Server name of the server where the note was actually delivered on
- Note: For Mail-Rules you need to allow your servers to use "monitors" (Security tab)

# DBMT command line options

- **-compactThreads <n>**
  - Use <n> threads for compact where **n** is between **0 & 100** inclusive
  - Number of threads DB copy style compact operations. **Default is 1 thread.** If **0** is specified, no compact operations are performed.
- **-updallThreads <m>**
  - Use <m> threads for updall, where m is between **0 & 100** inclusive
  - Number of threads doing updall operations. Default is **1**.
- **-ftiThreads <g>**
  - Use <g> threads for the rebuild of the full text indices, where g is between **1 & 100** inclusive
  - Configures number of threads for rebuild of full text indices if corrupt or **-ftiNDays** specified. Default is **1**; 0 is not allowed.
- **-ftiNdays <f>**
  - **Rebuild** full text indices every <f> days from initial creation
  - Default is to rebuild full text indexes only when found corrupt.

# DBMT Schedule Options

- **-range <s> <e>**
  - only run between times <s> and <e> daily
  - Minimum of 10 minutes between time <s> and time <e>
  - Times must be specified in 12hour format with a required AM/PM (e.g. 11:50PM)
  - DBMT process will remain loaded but dormant until -range comes around again the next day
  
- **-timeLimit <q>**
  - only run for <q> minutes
  - DBMT process will exit once **-timeLimit** expires and all active DBMT processing completes
  - -stoptime <e>
  - only run between now and time <e> once
  - Time must be specified in 12hour format with a required AM/PM (e.g. 6:00AM)
  - DBMT process will exit once -stoptime is reached and all active DBMT processing completes

# DBMT Compact Options

- **-compactNdays <p>**
  - Compact databases that have not been compacted in last <p> days
- **-noCompactLimit**
  - Allow compact to run past the end time to finish, but no new compacts will be started
- **-force <n>**
  - Take DBs offline for fixup & compact if flagged for force
    - Fixup is run only when 5 or more consecutive compact operations fail (and the failure is not due to database in use).
  - <n> between 0 and 7 where 1= Sunday, 2 = Monday, and so on.
    - If n is 0 (zero), the fixup operation will run any day.



# DBMT for System Databases



- DBMT does not compact System Dbs by default
  - but will perform -updall and -fti operations
  - You can explicitly specify databases on command-line or "ind" file
- System-Database List
  - names.nsf
  - events4.nsf , admin4.nsf, catalog.nsf, ddm.nsf
  - log.nsf, statrep.nsf, domlog.nsf, Indfr.nsf
  - busytime.nsf, clbdir.nsf, clubusy.nsf
  - daoscat.nsf, dbdirman.nsf, dircat.nsf
  - mtdata\mtstore.nsf

# DBMT Compact Filter indirect file

- dbmt\_compact\_filter.ind
  - Exclude list for Databases
  - DBMT will not compact database(s) < **ODS41** – Database(s) automatically added to dbmt\_compact\_filter.ind by DBMT
  - If -range is specified and **-noCompactLimit** is not specified
    - If DBMT compact of database takes longer than ~full -range specified and terminated because end of range reached , database automatically added to dbmt\_compact\_filter.ind by DBMT
    - Prevents DBMT from churning on same database every night
- Administrators may manually add additional database(s) to **dbmt\_compact\_filter.ind** if e.g. they have an Application/database that needs to be Highly Available all the time

# DBMT -updallThreads

- By default, specific views in mail databases are
  - brought up-to-date
  - **Created**
  - **marked non-discardable**
  - dramatically improving end user's fail over experience in a cluster
- Need Inheritance Enabled and Template Name one of StdR7Mail,StdR8Mail,StdR85Mail or StdR9Mail
  - (\$Inbox), (\$Drafts), (\$Sent), (\$All)
  - (\$RepeatLookup), (\$ToDo), (\$Calendar), (Haiku\_TOC)
  - (\$Alarms), (iNotes), (\$Users), (iNotes\_Contacts), (\$ThreadsEmbed)
- If you use customized mail template(s) use the following notes.ini
  - DBMT\_MailTemplate=templatename1,templatename2,templatename3

# DBMT -updallThreads

- Custom Database View Configuration
  - Required Inheritance Enabled and Template Name set on database(s)
  - DBMT\_TemplateName=ViewNameOrAlias1;ViewNameOrAlias2;...ViewNameOrAliasN
  - Substitute TemplateName after the DBMT\_ with inherited database Template Name
  - Substitute the ViewNameOrAlias1-N with database View Name or Alias and separate the ViewNameOrAlias1-N with either semicolons or commas
- Examples in notes.ini
  - DBMT\_StdNotesLog=SecurityEvents, ReplicationEvents, MailRoutingEvents
  - DBMT\_StdR4AdminRequests=All Requests by Time Initiated;All Request by Server
- Remember notes.ini setting(s) are restricted to 128 characters!

# DBMT Statistics

- Statistics for Compact and Updall when DBMT terminates
  - DBMT.Compact.Began 04.06.2016 21:27:44
  - DBMT.Compact.Finished 04.06.2016 21:27:59
  - DBMT.Compact.Successful 26
  - DBMT.Compact.Unsuccessful 1
  - DBMT.Compact.Unsuccessful.InUse 1
  - DBMT.Compact.Unsuccessful.TimeLimit 0
  - DBMT.Compact.BackLog 0
  - DBMT.Compact.Fixup 0
  - DBMT.Updall.Began 04.06.2016 21:27:44
  - DBMT.Updall.Finished 04.06.2016 21:28:03
  - DBMT.Updall.Processed 36
  - 04.06.2013 21:28:05 Database compactor process shutdown

# Multiple DBMT Processes

- DBMT takes as input for database(s) to act on:
  - A single database (.nsf)
  - An indirect file (.ind)
  - A directory name
  - No specification - Runs against all databases
- Restricted in order to prevent overlap
  - Only one DBMT process running against all databases is allowed to be loaded at any given time.
    - Any additional DBMT process launched to run against all databases will not be allowed and will simply exit
  - Administrators are allowed to run multiple DBMT processes by specifying a database, indirect file or directory name, but consideration should be given before doing so to avoid overlap
  - DBMT by default, does not compact system databases, unless the system database (.nsf) is specified explicitly on the command line or explicitly in an indirect file (.ind)

# Domino 9 Updall

- Updall performs the following tasks by default.  
These are also tasks that the database maintenance tools performs:
  - purges deletion stubs
  - expires soft deleted entries
  - updates unread lists
- New -nodbmt option
  - Skips those operations for updall if you have DBMT configured
  - Faster if you run updall for other reasons

# Domino 9 Compact – New Options

- **-# nn**
  - Execute up to nn compactions in parallel using different threads.
  - Up to a maximum of 20 Threads with a default of 1.
- **-W nn**
  - Only compact databases which haven't been compacted successfully in last nn days. Ordered by last compact time.
- **-w**
  - Exclude system databases (e.g. log.nsf, names.nsf).
- **-X nn**
  - Limit each database compaction to no more than nn minutes. (applies to in-place compaction only)
- **-x nn**
  - Limit total compaction time to no more than nn minutes.
- **\_\***
  - Include \*.ntf and \*.box when processing a directory



# Upgrading ODS for DB with old DB Class



- If you used explicit file versions for a database like test.ns7 you cannot use compact to bring the database to a newer ODS level
  - Even when you change the extension of the database, the internal database header has a fixed version which will prevent the ODS change
- "-upgrade Upgrade databases created with older DB classes to the most recent class."
  - For example: **compact -C -upgrade help/** will change the ODS for all help databases
  - Tip: **-ODS** Perform a copy-style compact only if the current ODS is less than desired default ODS.
- Extract from C-API header file
  - **#define DBCLASS\_NOTEFILE 0xff01**
  - #define DBCLASS\_V6NOTEFILE 0xff10
  - #define DBCLASS\_V8NOTEFILE 0xff11
  - #define DBCLASS\_V85NOTEFILE 0xff12

# Top Statistics – NSF Buffer Pool

- Used for Buffering Database I/O
- Check Server Stats
  - Database.Database.BufferPool.Maximum.Megabytes
  - Database.Database.BufferPool. PercentReadsInBuffer
- Interpretation
  - Bad < 90% < PercentReadsInBuffer < 98% < Perfect
- Tune: notes.ini NSF\_Buffer\_Pool\_Size\_MB=n (in MB)
  - Default: 512 MB for 32bit and 1024 MB for 64bit Domino

# Top Statistics – NSF Cache

- Used for Caching Open Databases
- Check Server Stats
  - Database.DbCache.HighWaterMark
  - Database.DbCache.CurrentEntries
  - Database.DbCache.MaxEntries
  - Database.DbCache.OvercrowdingRejections
- Interpretation
  - Good = HighWaterMark < MaxEntries
  - Good = 0 OvercrowdingRejections
- Tune: notes.ini NSF\_DbCache\_MaxEntries = n
  - Default: NSF\_BUFFER Pool size multiplied by 3

# Top Statistics – (Cluster) Replication

- Use to check Cluster Replicator Performance
- Check Server Stats
  - Replica.Cluster.Failed
  - Replica.Cluster.SecondsOnQueue
  - Replica.Cluster.WorkQueueDepth
- Interpretation
  - Perfect < 10 < SecondsOnQueue > 15 > Bad
  - Perfect < 10 < WorkQueueDepth > 15 > Bad
- Tune:
  - Add more cluster replicators
  - Optimize cluster server usage (e.g. Split active users between cluster mates)

# Top Statistics – Transactions

- Use for Indication of Server Load
- Check Server Stats
  - `Server.Trans.PerMinute`
- Interpretation:
  - Heavy < 30 < `Trans.PerMinute` (per User) > 10 > Light
- Tune: Analyze Heavy users and try to avoid load

# Top Statistics – Concurrent Tasks

- Use to check Simultaneous Active Database Connections
- Check Server Stats
  - Server.ConcurrentTasks
  - Server.ConcurrentTasks.Waiting
- Interpretation
  - Waiting should be ZERO
- Tune:
  - Server\_Pool\_Tasks = n ( e.g. 80)
  - Server\_Max\_Concurrent\_Trans = m (e. g. Server\_Pool\_Tasks \* Number of Ports)

# Top Statistics – Platform CPU

- Used to check CPU Utilization on Server
- Check Server Stats
  - Platform.System.PctCombinedCpuUtil
  - Platform.System.PctTotalPrivilegedCpuUtil
  - Platform.System.PctTotalUserCpuUtil
- Interpretation:
  - OK < 90% CombinedCpuUtil > 90% > TOO HIGH
- Tune
  - Many Root Causes Possible

# Top Statistics – Platform Disk

- Used to check Disk Performance
- Check Server Stats
  - Platform.LogicalDisk.1.AvgQueueLen
  - Platform.LogicalDisk.1.PctUtil
- Interpretation
  - Good < 1 < AvgQueueLen > 3 > BAD
  - Good = PctUtil < 80%
- Tune
  - By several parameters (bufferpool, cache, namelookup), and OS / Disk Tuning
  - Note Platform.LogicalDisk.1.AssignedName=C points to the disk