



Session 6 – Data Flow: From the Field, What Works!

Scott Arnett

Director, Data Flow Product Management

Heidi Geronimo

Senior Product Analyst



DFS Settings

- **DFS memory allocation**
- **Overview DFS memory settings**
- **Computing DFS cache and memory**
- **Spill To Disk**
- **Further settings**
- **Registry settings**



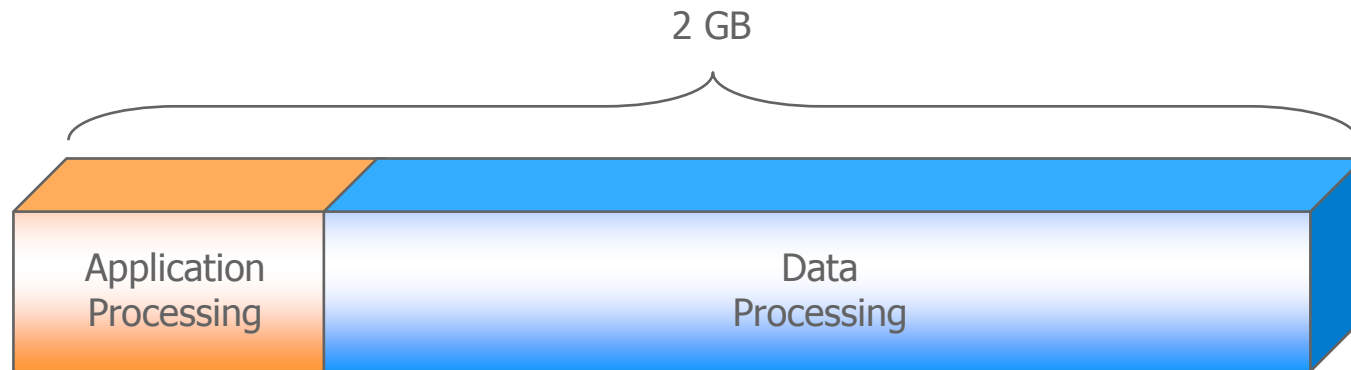
DFS Settings

- **DFS memory allocation**
 - Overview DFS memory settings
 - Computing DFS cache and memory
 - Spill To Disk
 - Further settings
 - Registry settings

DFS Settings

DFS Memory Allocation 1

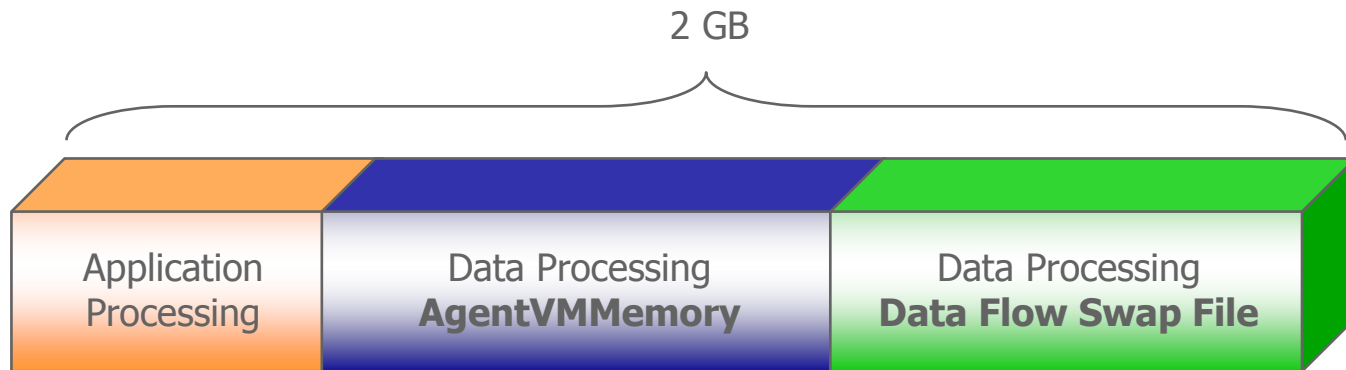
- Data Flow Server is a 32-bit application with a maximum application memory space of 2 GB
- Data Flow breaks this 2 GB into two parts
 - Application Processing Memory
 - Data Processing Memory



DFS Settings

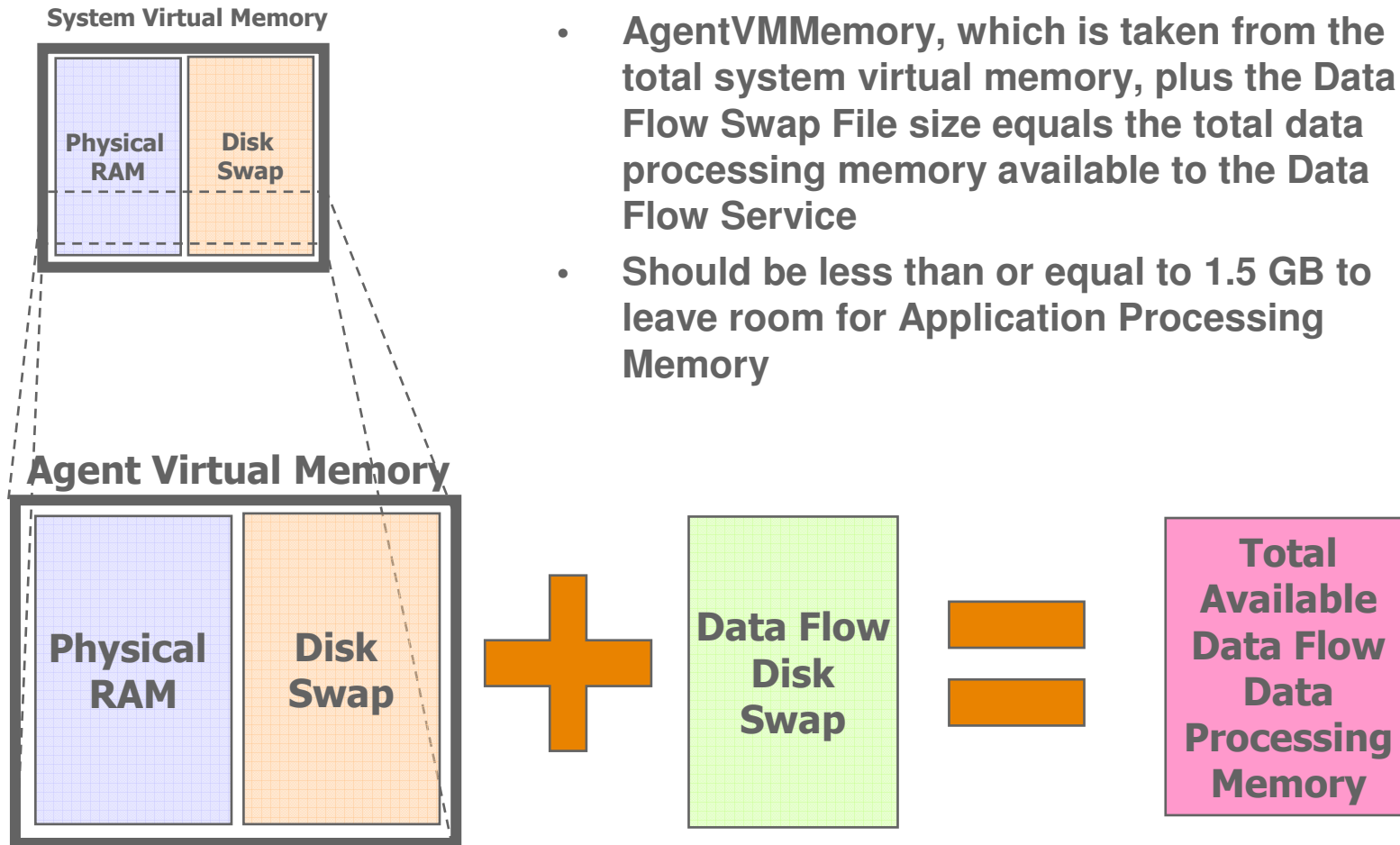
DFS Memory Allocation 2

- The Data Processing Memory can be drawn from two parts
 - AgentVMMemory
 - Data Flow Swap File



DFS Settings

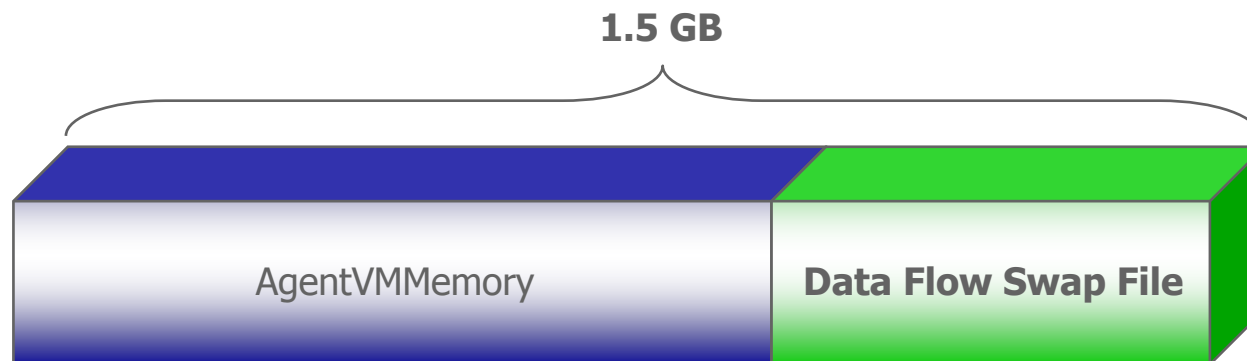
DFS Memory Allocation 3



DFS Settings

DFS Memory Allocation 4

- Example
 - Total Available Virtual Memory = 2 GB
 - OS accounts for 0,5 GB
 - Data Flow Server requires 0,5 GB
 - AgentVMMemory = 1 GB
 - Data Flow Swap = 0.5 GB





DFS Settings

- DFS memory allocation
- **Overview DFS memory settings**
- Computing DFS cache and memory
- Spill To Disk
- Further settings
- Registry settings

DFS Settings

Service Execution Memory

- AgentBlockSize:
 - Size of a memory block (default: 4,096 Bytes). Value should be a multiple of the OS system page file size (Windows = 4,096 bytes)
- MaxMemoryBlocks:
 - Maximum number of memory blocks for the whole system.
$$\text{MaxMemoryBlocks} = \text{AgentVMMemory} + \text{Sagent Swap File} - \text{MaxBlocks}$$
- MaxUsrMemoryBlocks:
 - Maximum number of memory blocks for each user
- MaxClients:
 - Maximum Number of concurrent clients using DFS





DFS Settings

Service Cache Memory

- MaxBlocks:
 - Maximum number of blocks that can be kept in cache. Value should not exceed total available memory (Max 1.5 GB). Depending on intended purpose (ETL – Frontend) this value ought to be adjusted
- MaxResultBlocks:
 - Maximum number of result blocks that can be kept in cache
- MaxSinkBlocks:
 - Maximum number of sink blocks that can be kept in cache with $\text{MaxSinkBlocks} \leq \text{MaxResultBlocks}$
- MaxResults:
 - Maximum number of plan results that can be kept in cache



DFS Settings

- DFS memory allocation
- Overview DFS memory settings
- Computing DFS cache and memory**
- Spill To Disk
- Further settings
- Registry settings



DFS Settings

Computation of Cache 1

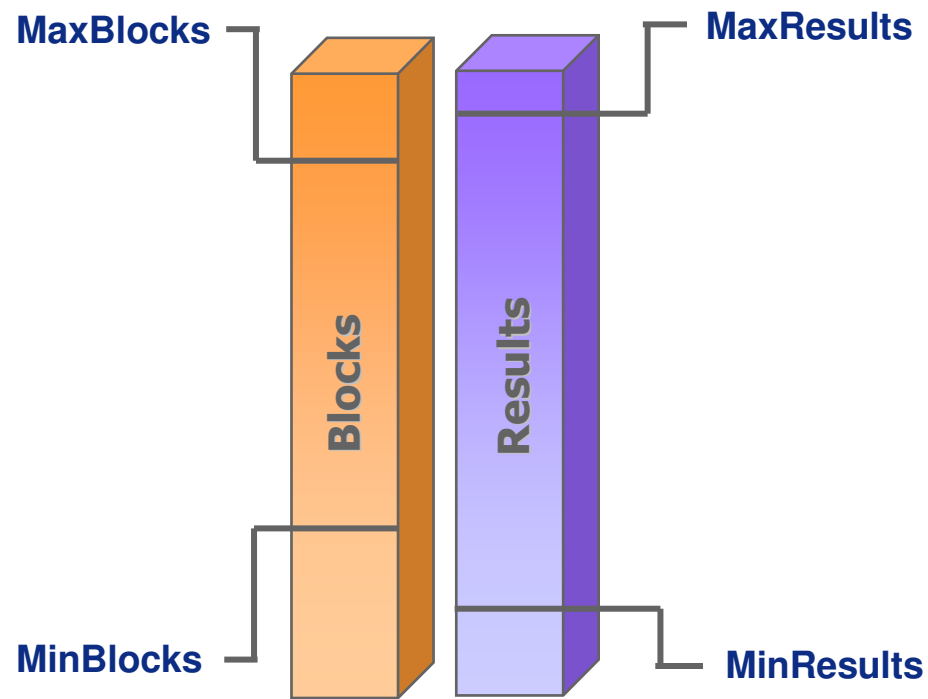
- $\text{MaxResultBlocks} = \text{Number of Sinks} * \text{MaxSinkBlocks}$
- $\text{MaxBlocks} = \text{MaxResultsBlocks} * \text{MaxResults}$

$\text{MaxBlocks} < \text{MaxMemoryBlocks}$!

- $\text{MinBlocks} \sim 0,75 * \text{MaxBlocks}$
- $\text{MinResults} \sim 0,75 * \text{MaxResults}$
- Turning off caching:
 - `HKLM\SOFTWARE\Sagent\DataMart\DC\ResultsCaching = 0`
 - Plan results are still cached, but in case a certain plan is run several times, DFS does not use them
 - Caching turned on = default

DFS Settings

Computation of Cache 2



DFS Settings

AgentVMMemory 1

General settings:

- „System Page File“ can be 1.5 times RAM-memory
- It is recommended to use a maximum of 2/3 of „System Page Files“ for „Block Allocation“
- **AgentVMMemory**
≤ 2/3 the size of System Page Files
≤ 1.5 GB
- **AgentVMMemory + Sagent Swap file**
≤ 1.5 GB
- **MaxMemoryBlocks + MaxBlocks**
≤ AgentVMMemory + Sagent Swap file(s)
≤ 1.5 GB



DFS Settings

AgentVMMemory 2

Scenarios

Scenario	RAM	System Paging file	AgentVMMemory GB/Blocks	Sagent Swap File GB/Blocks	MaxMemory blocks + MaxBlocks (*)
1	0.5 GB	0.75 GB	0.5 / 131,072	1 / 262,144	393,216
2	1 GB	1.5 GB	1 / 262,144	0.5 / 131,072	393,216
3	1.5 GB	2.25 GB	1.5 / 393,216	N/A	393,216
4	2 GB	3 GB	1.5 / 393,216	N/A	393,216
5	2.5 GB	3.75 GB	1.5 / 393,216	N/A	393,216
6 (**)	3 GB	4.5 GB	2.5 / 655,360	N/A	655,360

(*) If Sagent is mainly used for ETL (no client sinks) an optimization of run-time memory (MaxMemoryBlocks) ought to be used. If mainly used for Access plans (plans with client sinks) an optimization of cache memory (MaxBlocks) ought to be used

(**) This scenario is based on the usage of Product Update C2 for Data Flow v6.0



DFS Settings

AgentVMMemory 3

Scenario 2 – some rules of thumb

- **How to determine AgentVMMemory:**
 - AgentVMMemory = 2/3 System Page File = 2/3 1.5 GB = 1 GB = (1 x 1,024) MB = (1x 1,024 x 1,024) KB = 1,048,576 KB
 - 1 Sagent block = 4 KB, => 1,048,576 / 4 = 262,144 Sagent blocks
- **How to determine Sagent Swap file:**
 - Sagent Swap File = 1.5 GB – AgentVMMemory = 0.5 GB = (0.5 x 1,024) MB = (0.5 x 1,024 x 1,024) KB = 524,288 KB
 - 1 Sagent block = 4 KB, => 524,288 / 4 = 131,072 Sagent blocks
- **How to determine block size of memory reservation:**
 - Total memory block allocation = AgentVMMemory + Sagent Swap File = 262,144 + 131,072 = 393,216
- **How to determine MaxMemoryBlocks and MaxBlocks for ETL:**
 - AgentVMMemory= 262,144
 - MaxMemoryBlocks=391,216
 - MaxBlocks=2,000



DFS Settings

- DFS memory allocation
- Overview DFS memory settings
- Computing DFS cache and memory
- **Spill To Disk**
- Further settings
- Registry settings



DFS Settings

Spill to Disk 1

- Some Transforms require that the entire dataset be brought into memory in order to perform it's function
- With only 1.5 GB of Data Processing Memory to work with, you may run out of memory
- Spill To Disk addresses this problem



DFS Settings

Spill to Disk 2

- **Spill To Disk** monitors memory usage, and begins writing any excess blocks to several disk swap files
- This is separate from the Data Flow Swap File that is part of the Data Processing Memory
- Spill To Disk does not map it's data space to memory
- The amount of data is only limited by available disk space
- The files carry the extension VLM (very large memory) and are erased after plan execution



DFS Settings

Spill to Disk 3

- **Spill To Disk** monitors memory to see if available memory has dropped below a specified threshold
- **LowBlksMinThreshold** – Threshold of available memory blocks where Spill To Disk is activated. Can be specified as absolute value of available memory blocks or available percentage of total memory blocks (AgentVMMemory)
- **LowBlksMaxThreshold** – Threshold of available memory blocks where Spill To Disk is de-activated. Can be specified as absolute value of available memory blocks or available percentage of total memory blocks
- **LowBlksValsAbsolute** – This setting determines if the other two LowBlks settings are a percentage of available memory blocks or an absolute value of available memory blocks



DFS Settings

Spill to Disk 4

- **Spill To Disk** also monitors cache memory to see if available memory has dropped below a specified threshold
- **MaxSpilledBlocks** – The value is the maximum number of blocks to allow to be spilled to disk. If the setting does not exist or is explicitly set to 0 then there is no limit on the total number of spilled blocks
- **SpillCacheMaxPoll** – Frequency in seconds with which to test for low memory condition and write blocks to disk when needed



DFS Settings

Spill to Disk 5

- **SpillCachePercent** – Percent of Total Needed blocks to attempt to write to disk for each test interval. Total Needed blocks is defined as the difference between the current number of blocks and the LowBlksMinThreshold setting
- **SpillCacheResults** – Allow cached plan results to be written (spilled) to temporary disk files as needed. 0 = Disallow cache spilling; 1 = Allow cache spilling

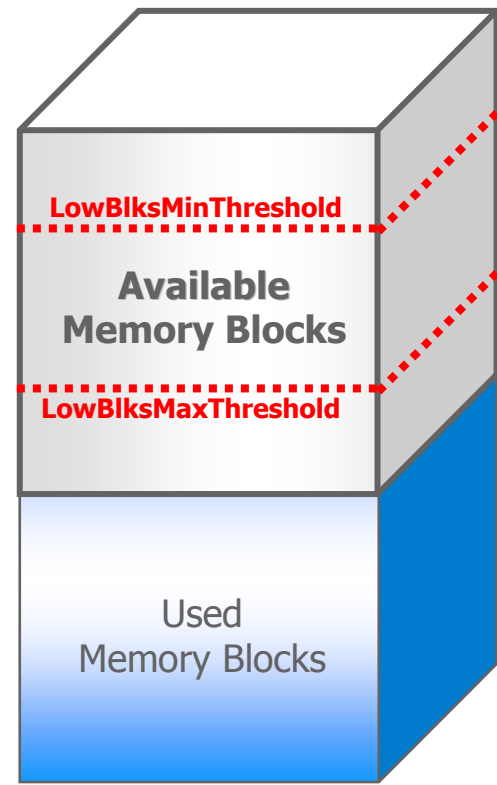


DFS Settings

Spill to Disk 6 - Example

- A user creates a plan that has a Sort Transform in it
- At the beginning of plan execution, the Data Processing Memory looks like this

Spill To Disk - Off



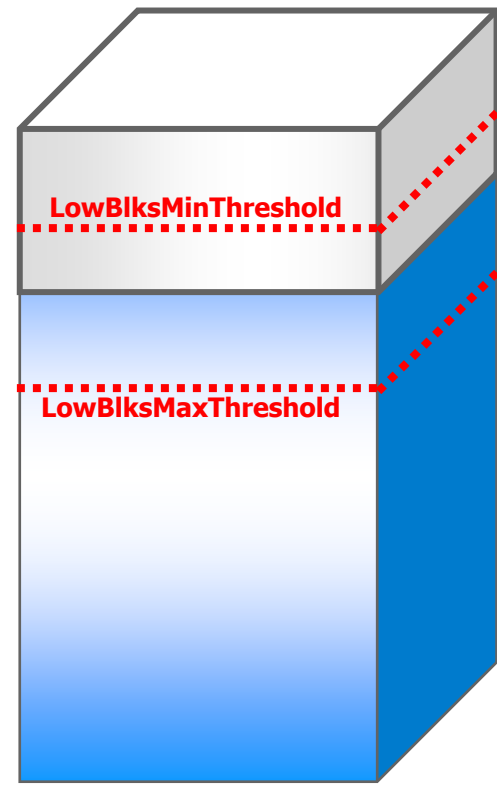


DFS Settings

Spill to Disk 7 - Example

- As the plan executes, the Sort Transform begins to buffer records in memory. Crossing the **LowBlksMaxThreshold**, but not activating Spill To Disk

Spill To Disk - Off



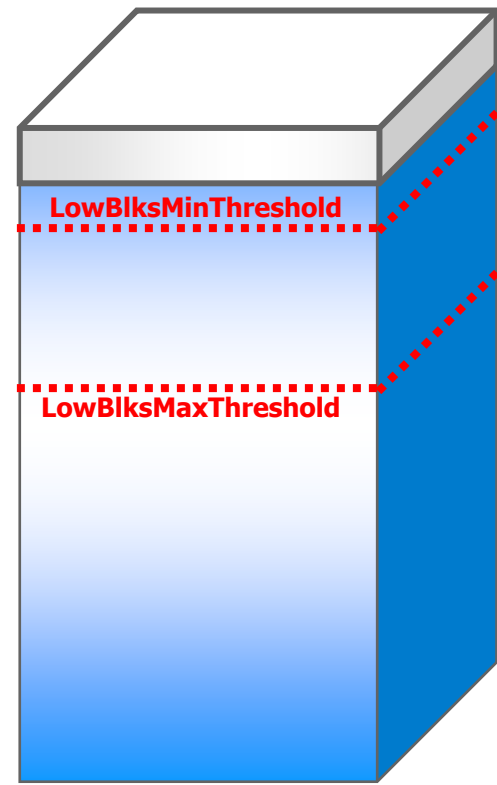


DFS Settings

Spill to Disk 8 - Example

- As it continues to execute, the Sort Transform continues to buffer blocks in memory, passing the **LowBlksMinThreshold**, and activating Spill To Disk. At this point the system will begin spilling excess blocks to the disk swap file

Spill To Disk - On



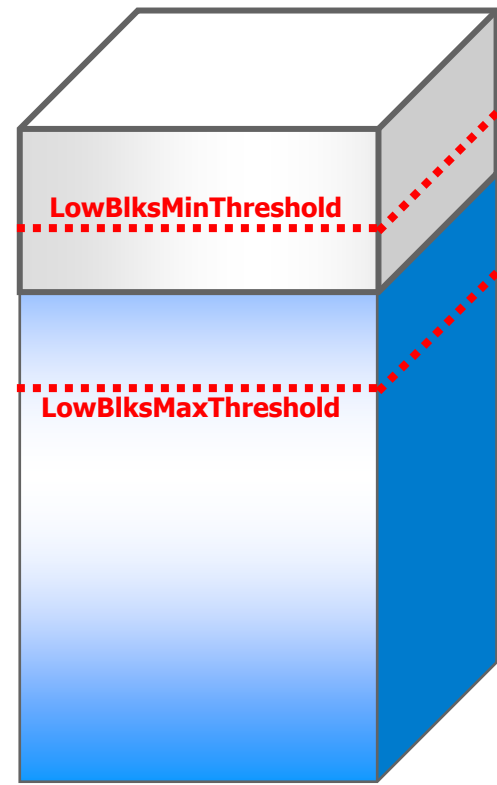


DFS Settings

Spill to Disk 9 - Example

- As the Sort Transform completes its operation, it begins to clear memory, dropping it below the **LowBlksMinThreshold**. However, this alone is not enough to deactivate the Spill To Disk functionality

Spill To Disk - On



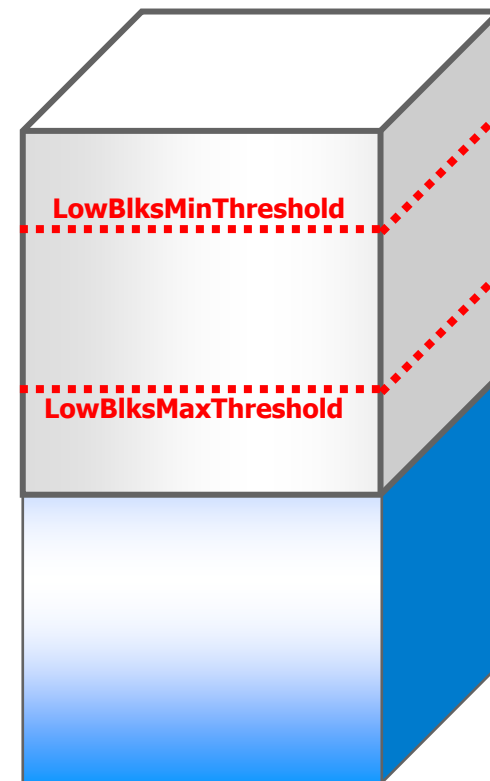


DFS Settings

Spill to Disk 10 - Example

- It is not until the memory level drops below the LowBlksMaxThreshold, that the Spill To Disk functionality is disabled

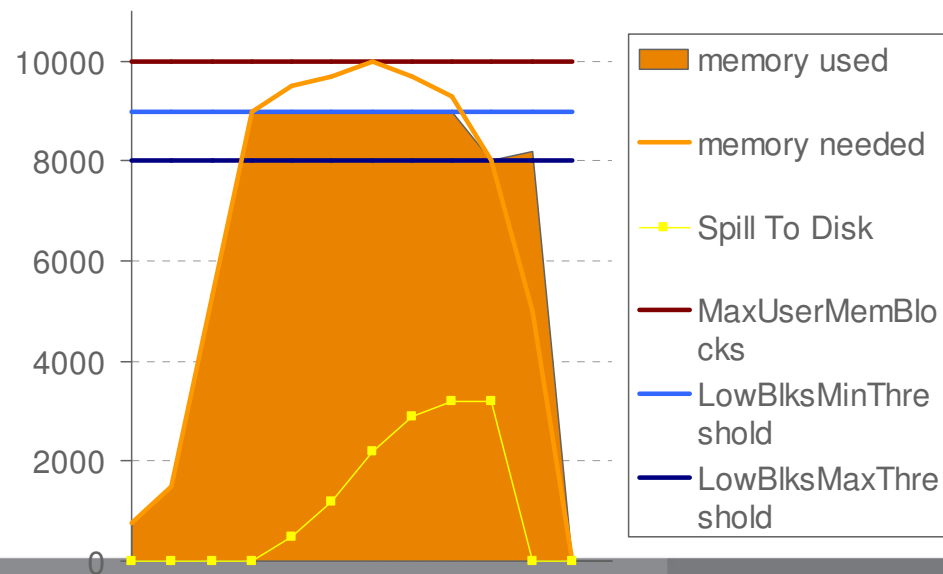
Spill To Disk - Off



DFS Settings

Spill to Disk 11 – Wrap Up

- Spill To Disk will allow transforms that buffer data to handle data sets greater than can be held in memory
- Theoretical maximum is the amount of disk space available in the swap file directory
- The swap file directory is specified by the SpillToDiskTempPath environmental setting, or the Temp directory of the DFS machine





DFS Settings

- DFS memory allocation
- Overview DFS memory settings
- Computing DFS cache and memory
- Spill To Disk
- **Further settings in Sagent Admin**
- Registry settings



DFS Settings

Database Connections 1

- MaxConnections:
 - Maximum number of user connections (sessions) available for the whole system
- MaxUsrConnections:
 - Maximum number of user connections (sessions) available for each user. Value must be less than or equal to the value for MaxConnections
- MaxDBConnections:
 - Maximum number of database connections available for the whole system



DFS Settings

Threads

- MaxThreads:
 - Maximum number of concurrent threads available for the whole system
- MaxUsrThreads:
 - Maximum number of concurrent threads available for each user. Value must be less than or equal to the value for MaxThreads



DFS Settings

- DFS memory allocation
- Overview DFS memory settings
- Computing DFS cache and memory
- Spill To Disk
- Further settings
- **Registry settings**

DFS Settings

Global Registry Settings 1 – Plan Information

Settings can be found under

HKEY_LOCAL_MACHINE/Software/Sagent/Datamart/DC

- IncludeStepStatus = 1 (DWORD)
 - Include Step Status enables to list every single step of a running plan within Sagent Admin and the actual number of records per step is shown, too!



DFS Settings

Global Registry Settings 2 – Sortings

Settings can be found under
HKEY_LOCAL_MACHINE

**This parameter is
just necessary
for Versions
up to 5.0**

...art/DC

- AllowPushBack = 0
 - When compiled depending on plan structure, it is possible that ... point of occurrence, but is handed back to SQL-Query (and database as a consequence). As a result it is possible that Sagent and database are using different sorting methods



DFS Settings

Optional Registry Settings 1 – DB SQL-Performance

Settings can be found under

HKEY_LOCAL_MACHINE/Software/Sagent/Datamart

- /TF/FastXSQLCommandOFF = 0 (DWORD, default = 1) DFS Server
 - This entry accelerates SQL Command (Sink) around factor 5-10. Effects only with Oracle and DB2. This entry is by default = 1 in V5.0 and default = 0 in V4.5i



DFS Settings

Optional Registry Settings 2 – Plan Tracking

Settings can be found under

HKEY_LOCAL_MACHINE/Software/Sagent/Datamart

- /DC/PlanTracking = 1 (DWORD, Default = 0)
 - PlanTracking turns on PlanTracking for the entire DFS. This applies for all plans, no matter what settings were carried out. Using this setting adhoc plans, immediately discarded after running, are logged, too



DFS Settings

Optional Registry Settings 3 – Sagent Admin

Settings can be found under

HKEY_LOCAL_MACHINE/Software/Sagent/Datamart

- /AD/AllowNonSALogins=1 (DWORD, Default=0)
 - With this entry (from V5.0 on) other users besides „SA“user can get access. User has to have ‘administrator’-rights
- /AD/CollapseRegistries=1 (DWORD, Default=0)
 - With this entry all repository entries in Sagent Admin are collapsed. Serves as a clearer representation of repositories in Sagent Admin. (Necessary only in case of a large number of repositories.)



DFS Settings

Extended Registry Settings 1 – Sorting Sequence

Settings can be found under
HKEY_LOCAL_MACHINE/Software

- /ORACLE/Home0/NLS_SORT = ,BINARY'
 - If sorting problems arise despite previous settings this Windows registry entry can be set (applies only for Oracle)
- /Sagent/Datamart/DC/SortStyle = ,binary' or ,lexical'
 - If Oracle is set on binary Sagent ought to be set on binary, too



DFS Settings

Extended Registry Settings 2 – Sorting Paths

Settings can be found under

HKEY_LOCAL_MACHINE/Software

- /TF/DiskSortPath
 - Default directory for any Disk Sorts. This entry can be overwritten within Disk Sorts
- /DC/AutomaticDiskSortTempPath
 - Default directory for Automatic Disk Sort. Is used, for example, if AllowPushBack = 0 is set
- /DC/SpillToDiskTempPath
 - Default directory for SpillToDisk. Used if SpillToDisk is allowed.



DFS Settings

Extended Registry Settings 3 – Sequence of display sinks

Settings can be found under

HKEY_LOCAL_MACHINE/Software/Sagent/DataMart/

Sets sequence of display components. Can be set for DesignStudio and Weblink separately. The following entries can be made:

- 0 = Repository (database) sorting. Mostly, but not always, sequence of transform creation. (old behaviour.)
- 1 = Alphabetically ascending by name.
- 2 = Alphabetically descending by name.

In case of no registry value default value = 1.

- Design Studio (DWORD): /IS/DisplaySinkOrder
- WebLink (DWORD): /saidc/DisplaySinkOrder