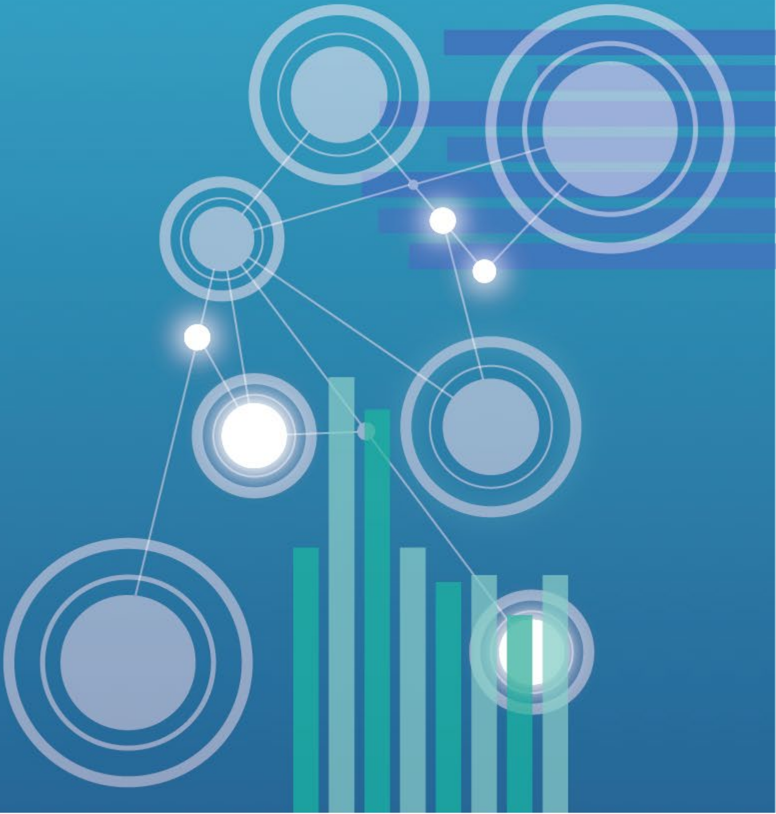
Operating IBM Planning Analytics:

Guidance and Recommendations

V2.7



Andreas Kugelmeier

Executive Consultant, FOPM

IBM Data and AI Expert Labs

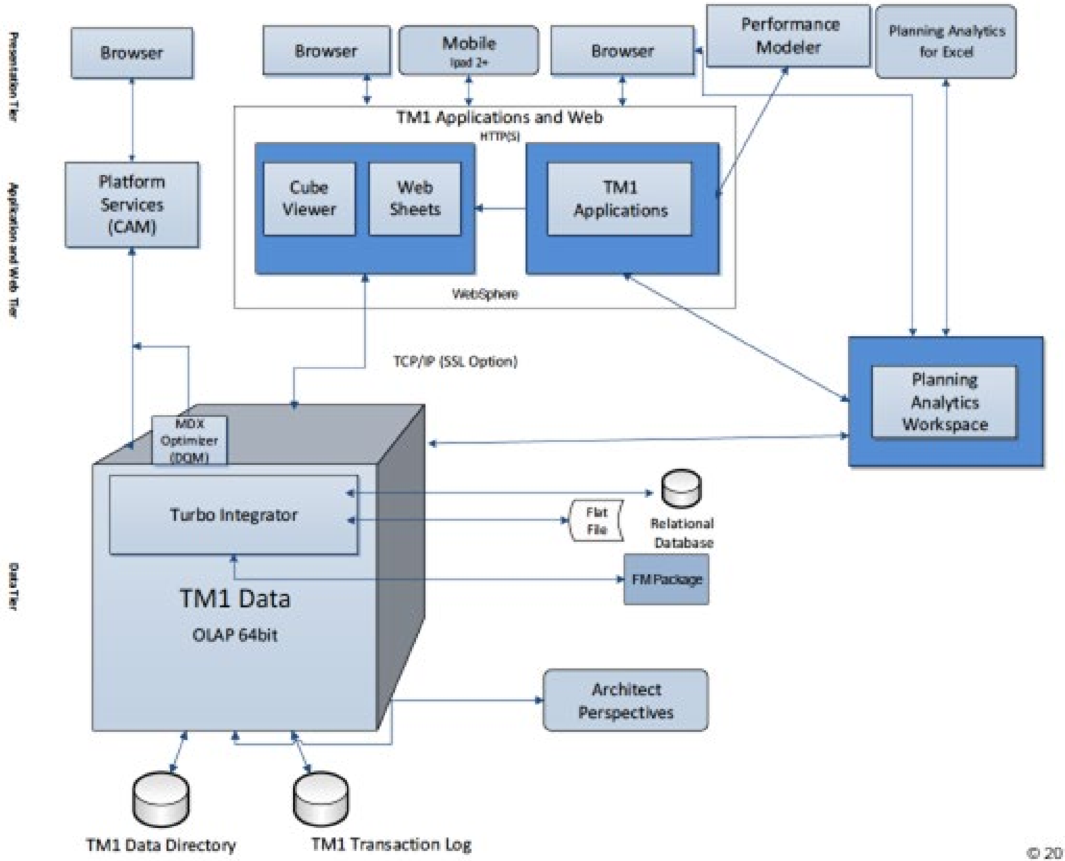
Kugelmeier@us.ibm.com

+1-215-384-7302

TM1 Server and Database Configuration

- Architecture
- TM1s.cfg
- MTQ
- Timeouts
- Query Cache (VMM & VMT)
- Logging (User Activity, Transaction Logging, Audit Logging, …)
- Dimension Sort Order Optimization
- Audit Logging
- }HierarchyProperties
- Performance Modeler settings
- Security
- Parallel Processing
- Minimizing Contention

Planning Analytics Architecture



TM1 Database Server Configuration: TM1s.cfg

- Click here for documentation on Planning Analytics tm1s.cfg parameters
- Apart from parameters needed to start and operate the database (such as security authentication settings, port numbers, paths etc.), the TM1s.cfg should ONLY contain parameters that are
  - Proven to be needed or beneficial for the installed PA release version
  - Used to achieve a certain user experience / enable a feature
- Any other parameter changes (such as changes prompted by IBM support to circumvent a bug or other issues) should ONLY be temporary, because:
  - Parameter changes are often of a type that will ‘throttle’ TM1
  - If temporary parameter changes are kept in the tm1s.cfg, the database will over time be configured to run in an ‘outdated’ operating mode.
- Temporary parameters (such as parameter changes prompted by support) should be commented as such and should include an explanation of why they were added (logical reason, not just ‘as told by IBM support’)

TM1 Database Server Configuration: TM1s.cfg

	Tm1s.cfg parameter	explanation

Recommended non-default parameters:	AllowSeparateNandCRules=T
	ViewConsolidationOptimizationMethod=Tree
	LoadPrivateSubsetsOnStartup=T
	ReduceCubeLockingOnDimensionUpdate=T
	PersistentFeeders=F vs. T
		• Default is F. It is the proven practice to separate N- and C-level rules
		• Array is default (legacy). Tree is typically faster
		• Default is F. T will improve performance for end users with private subsets
		• Default is F. T will significantly improve scalability and reduce contention
		• Default is T, consider/test setting to F and using MTFeeders (see below). For new TM1 databases, set to F and only turn on if/when needed.

Optional recommended non-default parameters	MTCubeLoad=T
	MTFeeders=T
	MTFeeders.AtStartup=T
	MTCubeLoad.UseBookmarkFiles=T
	IndexStoreDirectory=<IndexStoreDirectory>
		• Default is F. Allows TM1 to use multiple threads/cores for Cube Load on Startup. Will likely allow for faster database startup. Click here for more information
		• Default is F. Allows TM1 to use multiple threads/cores for feeder processing. Will likely result in significantly faster 'refresh' & startup of database instances. Click here and here for more information. Note: use of MTFeeders functions is typically OK with conditional feeders that are based on attributes or regular lookup cubes.
		• The use of Bookmarks/Bookmark files can speed up startup time significantly. It is recommended to compare startup time with or without use of Bookmarkfiles (after bookmarkfiles were created)

New default to be handled with Caution	**MTQQuery=T** (default) **vs.** **MTQQuery=F**	• Default=T. MTQQuery will use MTQ to query and 'pre-'cache ***C-level*** views used by TI-processes. While this typically leads to performance improvements, it can also significantly increase RAM consumption for larger views. That is because a TI with MTQQuery=F / without MTQQuery will not load a view into RAM, but instead 'cycle' through a view in smaller blocks, not really increasing RAM usage much. Yet with MTQQuery=T, TM1 will attempt to query and load the entire C-level view into RAM prior to processing it via TI ⇒ MTQQuery=T (default) can increase RAM use significantly ⇒ only use IF (i) size of views is managed (known to be small enough) and (ii) sufficient RAM is available, otherwise, use MTQQuery=F; with MTQQuery=T, use DisableMTQViewConstruct () for Tis for which you want to disable MTQQuery
Other	ForceReevaluationOfFeedersForFedCellsOnDataChange -> Leave at F (default), i.e. do <u>not</u> set to T	• Using ForceReevaluationOfFeedersForFedCellsOnDataChange=T can very significantly degrade TI processing and input contribution performance. • Only use ForceReevaluationOfFeedersForFedCellsOnDataChange=T if you have conditional feeders that change based on regular data inputs <u>and</u> if the continuous re-evaluation is needed. Otherwise, do not use this parameter. • Enabling this parameter is not necessary even for use of TM1 Applications/Contributor. The parameter is only needed if TM1 Performance Modeler is used for cube rule modeling (it is not recommended to use TM1 Performance Modeler for cube rule modeling or TI-process modeling).

TM1 Database Server Configuration: TM1s.cfg

- On MTQ:
  - What is it? -> MTQ stands for ‘Multi-Threaded-Queries’. It allows TM1 to split a query into multiple chunks that then are processed in parallel CPU threads. Without MTQ, every TM1 query runs in a single thread, with MTQ, multiple threads can be leveraged, hence significantly improving performance.
  - **MTQ significantly improves performance and should typically be set to leverage all available cores.** Caveat: While MTQ=ALL ensures that hardware is leveraged at its maximum, it is only a recommended setting provided that the HW-sizing is adequately matched with the demands of the TM1 Database(s) that are running on the machine. Operating TM1 databases with MTQ=ALL yet on machines with insufficient HW capabilities may over-tax the HW and lead to performance degradation.
  - TM1 10.2 Default: MTQ OFF => Recommended setting in TM1 10.2.2 is MTQ=ALL
  - **PA / TM1 11 Default: MTQ=-1 / MTQ=ALL (all available cores).**
  - If your HW supports Hyperthreading, ensure you enable it on the Bios and/or OS. A Hyperthreaded Core provides about 80% of the performance of a ‘real’ CPU core. Therefore, enabling Hyperthreading can improve performance by about 80%.

TM1 Database Server Configuration: TM1s.cfg

- On MTQ (continued):
  - The MTQ value (MTQ=X, like MTQ=4) is the total thread pool for all users that the TM1 Server makes available for Multi-Threaded-Queries.
    - ⇒ Example: If you set MTQ=4 on an 8-CPU machine, only 4 threads will be used in Total for MTQ . If two users run a query that leverages 4 MTQ cores, one user will get 1 non-MTQ thread, and the other user will get 1 non-MTQ Thread plus 4 MTQ Worker Threads.
    - ⇒ MTQ is not the MTQ pool per user, but for the TM1 Database server in total.
  - An MTQ pool thread becomes ‘available’ to other users once it has finished.
    - ⇒ Example: If a query initially gets assigned 4 MTQ worker threads, this does not mean it will keep all those threads. One thread may finish and the query will continue with 3 worker threads. Or 2 threads will finish, but the TM1 MTQ engine will determine that the next ‘part’ of the query is best served with an additional 8 threads, resulting in 10 threads being used…
    - ⇒ As a particular MTQ worker threads is finished (right after ‘commit’ in TM1Top), the thread becomes available again, and it may be used by the same query (for a new worker thread), or by a different user query.

TM1 Database Server Configuration: TM1s.cfg

- On MTQ (continued):
  - MTQ is NOT dependent/reliant on the # of CPUs on the machine!
    - ⇒ MTQ will leverage multi-core processing capabilities by splitting and resolving queries via parallel worker ***threads***
    - ⇒ If you set MTQ to ALL or leave it at the default (ALL), TM1 will set an MTQ thread pool equal to the number of cores on the machine.
    - ⇒ You can set MTQ to a value higher than the number of cores on the machine. You will see more worker threads than CPU cores. This is acceptable, even though on most hardware, it is typically not going to result in performance gains, yet at lower concurrency typically will also not lead to performance degradation.
  - The CPU time and resources given to a MTQ worker thread are entirely handled by the Operating System.
  - An MTQ worker thread will ONLY occupy an entire CPU core if the core is not busy otherwise.
  - TM1 simply initiates a parallel thread. Where and how this thread is being processed is handled by the Operating System.

⇒ it is typically best to leave MTQ at its default and hence have it leverage as many worker threads as there are CPU cores.

Caveat: While MTQ=ALL ensures that hardware is leveraged at its maximum, it is only a recommended setting provided that the HW-sizing is adequately matched with the demands of the TM1 Database(s) that are running on the machine. Operating TM1 databases with MTQ=ALL yet on machines with insufficient HW capabilities may over-tax the HW and lead to performance degradation.

TM1 Database Server Configuration: TM1s.cfg

- On MTQ (continued):
  - If you have two or more TM1 databases on the same machine, it is still recommended to set MTQ to ALL, because: An MTQ worker thread does ***not*** occupy a CPU core. MTQ worker threads will share CPU cores where needed and where applicable. It is the Operating System that handles CPU time and resources.
    - ⇒ If you have two databases on an 8 core machine, each set to MTQ=8, and on each a user runs a query leveraging 8 MTQ threads, the 16 MTQ threads are balanced among the 8 cores on the machine, utilizing the HW in an optimal way.
    - ⇒ If only one user runs a query leveraging 8 MTQ threads, the HW is still used in an optimal way.
    - ⇒ But if you set MTQ=4 and just one user runs a query, the HW is utilized at only 50%.
    - ⇒ If we extend these simple examples to many users and dozens to hundreds of threads (like in a typical, large TM1 environment), we can argue that every TM1 database should be configured to run queries as efficiently as possible, leveraging the available HW as much as possible.
    - ⇒ Balancing HW utilization is the job of the Operating System

TM1 Database Server Configuration: TM1s.cfg

- On MTQ (continued):
  - On very powerful HW (high # of CPU cores) on smaller to mid-sized databases, it may not make much of a difference to set MTQ to a value that is lower than the total number of cores.
  - On very large Databases, a high MTQ # typically does make a big difference

⇒ It is a good practice to start with MTQ=ALL (= default).

⇒ To evaluate if different MTQ settings improve performance, adopt a holistic testing approach:

- ***MTQ Worker Thread ≠ CPU Core*** => An MTQ Worker Thread is 'just' a TM1 Thread.
- What is the peak load on each database?
- What is the Avg CPU utilization at Peak?
- Do we currently max-out our HW capabilities? (HW should be used as much as possible; HW idle time is not good, because it means 'things' could be running faster)
- How do the # of cores affect our end-user queries? This depends on the database/cubes: are they very large, do they leverage rules, …

⇒ Analyze!

⇒ Is there an MTQ setting where using additional worker threads do not matter much anymore? If Yes, then this should be your max MTQ setting (i.e. do not go higher than where you see a tangible difference)

TM1 Database Server Configuration: TM1s.cfg

- On MTQ (continued):

  Some additional considerations when evaluating MTQ Settings:

  - One could argue that setting MTQ to a slightly lower value than the # of cores will allow more CPU time to be available for TI-processing (for example). But: this is ***only*** true IF the total concurrency on the database(s) is sufficiently low (compared to the # of cores). Example: The Operating System is balancing 200 concurrent TM1 users (and their TM1 threads) among its 36 cores. In such a context, setting MTQ to 36 or 34 or 30 will not really make a difference, because the CPU will be operating at maximum capacity regardless of the MTQ setting.

  ⇒ When considering to lower MTQ, take into account the overall concurrency on the database at peak time and the data volume that you are querying.

  ⇒ If concurrency is high and the data volume high, a higher MTQ value typically results in better performance in that end-user queries will be faster.

  ⇒ When you operate multiple Databases on one server and each database has a high concurrency, lowering MTQ will not necessarily free up CPU time for processing etc., because the overall load on the environment is already high.

  ⇒ Lowering MTQ in environments where multiple TM1 Database Instances share the same hardware will only have a positive impact on performance if the concurrency on each database is sufficiently low such that HW resources are kept available for other TM1 databases (and if those other databases need the free CPU time.

  ⇒ When setting MTQ, do not optimize it for low concurrency times. Optimize it for when the database is used most.

TM1 Database Server Configuration: TM1s.cfg

- **Recommendations:**
  - Remove Duplicate TM1s.cfg entries and entries that are not applicable
  - Apply TM1s.cfg settings as per prior slides on TM1s.cfg
  - For as long as IT manages the HW & SW environment and infrastructure / unless the Business Owns and Manages the HW & SW environment and infrastructure:

    ⇒ Performance-related parameters should be configured identically for all databases (i.e. TM1 shall ‘operate’ identically in all environments)

    ⇒ Allow database-specific parameters only if they are related to enabling/disabling/modifying user experience & features, such as Time-Outs, EnableNewHierarchyCreation, EnableSandboxDimension, EnableTIDebugging, …
  - Always check what the default parameter values are

Planning Analytics: Configuring Time-Outs

- TimeOuts: Cognos Analytics timeouts should be set to be smaller than TM1 Web and TM1 Database Timeouts,

  ⇒ CA Session timeout

  < PAW Session Time-Out

  < TM1 Web & TM1 Web Applications timeout

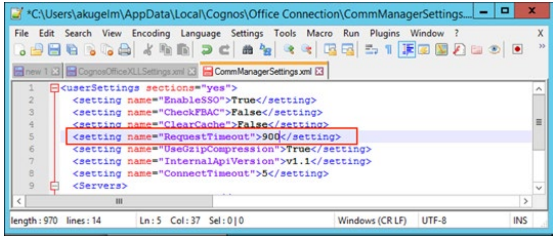
  < TM1 Database timeouts (for TM1 Database Servers, keep in mind that there is a 'legacy' client IdleConnectionTimeOutSeconds, and a Rest-API connection HttpSessionTimeout.

- Recommended TM1 Web Session Time-Out when using PAW: 60 minutes
- Planning Analytics Time-Outs Overview (TM1 DB Server, Web, Applications, PAW, …)
- New as of 2.0.6: TM1 Web HttpSessionTimeout

Planning Analytics for Excel: Time-Outs

- In **network environments with high latency (high ping rate and/or many network hops),** it can be beneficial to increase the Request Timeouts and Execution Timeout for PAX in the PAX xml configuration files:
- **Request Timeout in CommManagerSettings.xml:**

  <setting name="RequestTimeout">300</setting>


*C:\Users\akugelm\AppData\Local\Cognos\Office Connection\CommManagerSettings...
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
new 1
CognosOfficeXLLSettings.xml
CommManagerSettings.xml

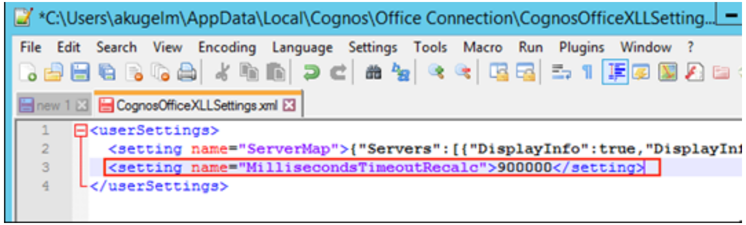
```
<userSettings sections="yes">
  <setting name="EnableSSO">True</setting>
  <setting name="CheckFBAC">False</setting>
  <setting name="ClearCache">False</setting>
  <setting name="RequestTimeout">900</setting>
  <setting name="UseGzipCompression">True</setting>
  <setting name="InternalApiVersion">v1.1</setting>
  <setting name="ConnectTimeout">5</setting>
  <Servers>
```

length : 970 lines : 14
Ln : 5 Col : 37 Sel : 0 | 0
Windows (CR LF)
UTF-8
INS

- **RestAPI Execution Timeout in CognosOfficeXLLSettings.xml**

  <setting name="MillisecondsTimeoutRecalc">300000</setting>

  The default is for the execution timeout is 300000, and by default, there is no entry for this setting. Note that this setting is in milliseconds, not seconds

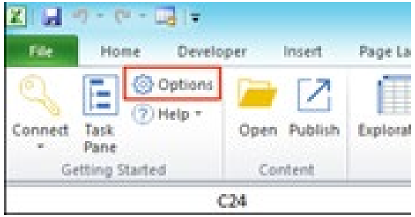

*C:\Users\akugelm\AppData\Local\Cognos\Office Connection\CognosOfficeXLLSetting...
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
new 1
CognosOfficeXLLSettings.xml

```
<userSettings>
  <setting name="ServerMap">{"Servers":[{"DisplayInfo":true,"DisplayInf
  <setting name="MillisecondsTimeoutRecalc">900000</setting>
</userSettings>
```

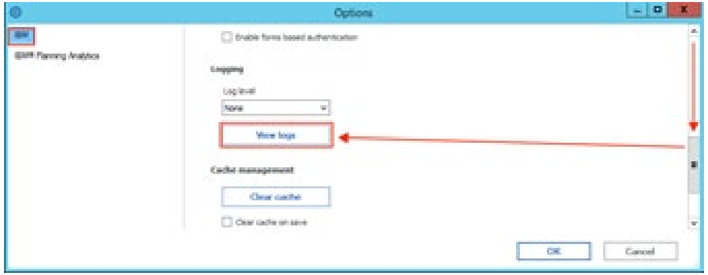
Planning Analytics for Excel: Time-Outs

How to find the location of the PAX xml config files:

i. In PAx, click on ‘Options’:



ii. In the top left corner, click on ‘IBM’, then scroll down and click on ‘View Logs’:


Options

IBM

IBM Planning Analytics

□ Enable forms based authentication

Logging

Log level: None

View logs

Cache management

Clear cache

□ Clear cache on save

OK Cancel

iii. In the window that opens, go up one folder

Query Caching: VMM & VMT – Where do I start?

- Query caching behaviour is configured per cube via the VMM & VMT value in the }CubeProperties cube, where
  - The **VMM** value defines the maximum amount of memory to be used for caching of Stargate views per cube (i.e. the memory pool in RAM that is made available for caching). If no VMM value is specified the default value is 128KB. The valid range is 16 – 42,934,943,296 kb (16-2^32kb).
  - The **VMT value defines the** query time threshold (in seconds) as of which Stargate Views will be cached (default = 5 seconds).
- What’s a Stargate View? A Stargate view is a calculated and stored subsection of a TM1 cube that is created when querying a TM1 cube, with the purpose of allowing quicker access to the cube data. A Stargate view is different from a TM1 view object in that it contains only the data for a defined section of a cube (current title elements and row and column subsets), and does not contain the formatting information and browser settings that are in a view object.

Query Caching: VMM & VMT – Where do I start?

VMM:

- For applicable cubes, it is a good practice to increase the VMM value to a new 'default' value of between 1MB and 5MB (or higher). This will allow caching of relatively large as well as many views and improve performance accordingly.
- Running and monitoring representative sample queries via TM1 Operations Console will indicate when the Query Cache is leveraged (short query time and no MTQ activity) vs. if the cache is not being used (long query times & MTQ activity). One can also log stargate creation with

  ```
  log4j.logger.TM1.Cube.Stargate=DEBUG
  log4j.logger.TM1.Cube.Stargate.ViewStorage=DEBUG
  ```

  ONLY use such loggers in DEV or Test environments & for testing/evaluation purposes
- If the VMM value(s) are too low, A query may not be cached. In such a scenario, running the same query repeatedly will cause repeated query execution & calculation vs. retrieval from the cache. If such scenarios are common, i.e. if certain common queries will not be cached due to low cache size, increase the cache until the query result is retrieved from cache.
- Check the logs for stargate size(s) and increase VMM if needed. For larger cubes, a cache size in the high MB to GB range can be advantageous!

Query Caching: VMM & VMT – Where do I start?

VMT:

- For smaller environments, keeping VMT at its default of 5 seconds is typically a good idea.
- For larger environments, where MTQ is leveraged heavily and where many processors may get engaged, a decrease of the VMT threshold may be beneficial in that CPU intensive queries, even if short, will be cached and hence will employ only one CPUs once cached rather than many.

Query Caching: VMM With MTQ

The use of MTQ typically will require an increase in VMM size. If VMM cache is set too low, even queries that were cached without MTQ use may not be cached anymore once MTQ is enabled. In such cases – to avoid unnecessary re-execution of MTQs - increase the VMM value until repeated query execution will not trigger MTQ activity anymore (indicating the cache is used).

⇒ Running and monitoring representative sample queries via TM1 Operations Console will indicate when the Query Cache is leveraged (short query time and no MTQ activity) vs. if the cache is not being used (long query times & MTQ activity).

⇒ If the VMM value(s) are too low, a query may not be cached. In such a scenario, running the same query repeatedly will cause repeated query execution & calculation vs. retrieval from the cache.

⇒ If such scenarios are common, i.e. if certain common queries will not be cached due to low cache size, increase the cache until the query result is retrieved from cache.

Logging: Overview

- System and Performance Monitoring is an essential tools for the Analysis and Remediation of problems (actively and retroactively)
- Click here for an overview of available tools, methods, etc.

Logging: TM1 Server Logging

- Location: TM1s-log.properties file, in the same directory as the tm1s.cfg
- Only use DEBUG loggers if
  - prompted by support or
  - for temporary troubleshooting/testing or
  - if the particular debug logger has a low or insignificant performance impact (such as the Lock.Exception debugger below)
- Recommended loggers:
  - log4j.logger.TM1.Lock.Exception=DEBUG

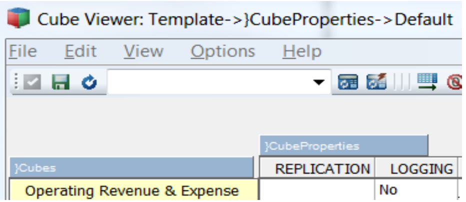
    (useful for identifying locking issues and analyzing means to remediate)
- [Available loggers]()

Logging: TM1 Database Activity Logging

- Log Activity of Users to disk!
- TM1Top and/or Operations Console can both be configured to write/log TM1 database / user activity to disk
- Those activity logs are essential in telling a story of ‘what was going on / what happened’, and are the best tool in determining how to approach and remediate a problem.
- View the document on Operations Console for details or – if using TM1Top – consult the TM1top.ini documentation (see parameters LogFile and LogPeriod)
- TM1Top logging on PA Cloud

Cube Transaction Logging

- TM1 Data that is loaded or entered into a cube is not automatically committed to disk (the *.cub file) but retained in-memory only.
- For Cubes with Cube Logging enabled (default) data changes will be logged in the TM1s.log file.
- Cube Logging may be disabled for a particular cube by setting the cube's 'Logging' value in }CubeProperties to 'No':



	}CubeProperties	
}Cubes	REPLICATION	LOGGING
Operating Revenue & Expense		No

- Use the functions
  - ➢ CubeSaveData() to commit one cube's data and/or
  - ➢ SaveDataAll() to commit all data to disk.

  When CubeSaveData() or SaveDataAll() are called, all corresponding log entries will be removed from the TM1s.log file and the logs will be 'archived' in new log files named tm1s<DateTimeStamp>.log such as tm1s20140210172014.log.

Cube Transaction Logging

- SaveDataAll() and CubeSaveData() will acquire a write lock on the cube/database. It is hence not recommended to use SaveDataAll() and CubeSaveData at the end of each TI load process (because that will cause lock-contention with other load processes that load data to any cube (SaveDataAll()) or also load data to the same cube (CubeSaveData() in a parallel load scenario (see below). There should be only one TI process that calls the SaveDataAll() function. Use a stand-alone, single, distinct chore to execute the SaveDataAll operation.
- Commit to disk prior to backups and/or at the end of a business day / at night. Do not commit to disk during regular working hours, as this may affect the user experience. There is no need to commit to disk more than once a day
- Undo operations will evaluate all applicable TM1 logs in the TM1 Log directory. If a large number of log files can be found in the Log directory, the UNDO operation may take a longer time (and may lead to a lock on the corresponding cube).
- For **larger read-only cubes it is a good practice to disable cube logging** as this will speed up data load performance. Because a read-only cube cannot be accidentally overwritten by users, cube logging is not needed for audit purposes.
- For Plan/Budget and other **input cubes, cube logging should however be enabled** to allow for proper auditing to occur and to allow for data-recovery in case of outages.

Optimization of internal Cube Dimension Order (Cube Optimization)

The order of dimensions significantly affects cube memory requirements and query performance

- **The affect on query performance tends to be more significant than the affect on RAM utilization, i.e. the relationship between memory requirements and query performance is not linear.**

  For example, a 20% reduction in memory requirements *may* be indicative of a >1000% improvement in query speed.

⇒ **Look at Cube Optimization as an essential tool and process**

Optimization of internal Cube Dimension Order (Cube Optimization)

So how do I optimize the cube dimension order to maximize performance?

- ➢ The TM1 ‘Cube Optimizer’ lets you re-arrange & evaluate a different, internal order of dimensions in a cube and apply the new sort order to the cube if desired.
- ➢ When you optimize the order of dimensions in a cube, TM1 does not change the actual order of dimensions in the cube structure. TM1 does change the way dimensions are ordered internally on the server, and because the cube structure is not changed, any rules, functions, or applications referencing the cube remain valid.
- ➢ You can have TM1 itself evaluate the cube and dimension structures and determine a system-generated optimal dimension sort order
- ➢ Or you can manually optimize dimension sort order, allowing for further optimization beyond the system-generated optimal sort order:

Optimization of internal Cube Dimension Order (Cube Optimization)

➢ To manually optimize dimension sort order:

1. Divide the dimensions into two groups: sparse and dense dimensions.
2. Order the dimensions as follows: smallest sparse to largest sparse, followed by smallest dense to largest dense.
3. Exceptions: It typically is better to put a very small, dense dimension before a very large but sparse dimension. For example, a dimension such as Version/Scenario (Act, FCST, Plan) that has only two or three elements is better positioned before a very large but sparse dimension, such as Product, which might have thousands of elements. => be flexible and experiment with different configurations.
4. The TM1 manual states that one “should optimize the order of dimensions in a cube only in a development environment while you are trying to determine optimal cube configuration”. This should be interpreted as follows: Determine the optimal cube order via tests in a production-like development environment. Then – at a time of zero-concurrency – implement the tested new dimension order in your production environment.
5. If a dimension (such as a measures dimension) contains string elements, it needs to be the last dimension.
6. Very often, moving the measures (or last) dimension (if it does not include string elements) upwards (from the last position) can result in significant memory savings and performance gains. The TM1 system-generated cube optimization will not consider moving the measures/last dimension. The impact of moving the measures/last will have to be evaluated manually.
7. If you find that moving the measures dimension upwards will render significant performance gains but you would like to use string elements too (for comments etc.), you may consider building a separate cube just for string measures.

TM1 DB Configuration: }HierarchyProperties

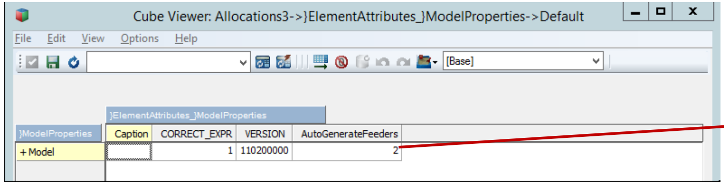
- The defaultMember for }hierarchy0 in }HierarchyProperties.cub is accessed and used if an MDX query against a cube does not include a dimension member specification.
- If the defautMember is empty, the member with index 1 is used (this member is NOT necessarily the APEX node of a hierarchy, it can be any element because it is the element that is/was created first when the dimension was updated/loaded)

- => It is recommended to define defaultMembers for all Dimensions

Performance Modeler: recommended default settings

(to prevent changes to TM1 Database if/when an admin logs into TM1 via Performance Modeler)

- Performance Modeler, upon login by an admin will attempt to modify (and recompile) rules right after login unless the following settings are set such that Performance Modeler is prevented from preventing the behavior:
- In }ElementAttributes_}ModelProperties, ensure that the value for 'AutoGenerateFeeders' is 2. This setting will set the Performance Modeler setting 'Generate Feeders Automatically' to No

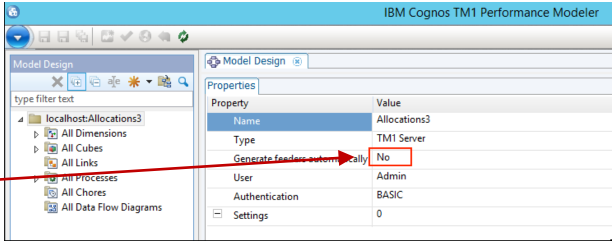


Cube Viewer: Allocations3->}ElementAttributes_}ModelProperties->Default

[Base]

}ElementAttributes_}ModelProperties

}ModelProperties	Caption	CORRECT_EXPR	VERSION	AutoGenerateFeeders
+ Model		1	110200000	2


IBM Cognos TM1 Performance Modeler

Model Design

type filter text

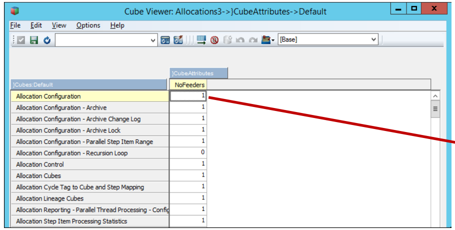
- localhost:Allocations3
  - All Dimensions
  - All Cubes
  - All Links
  - All Processes
  - All Chores
  - All Data Flow Diagrams

Model Design

Properties

Property	Value
Name	Allocations3
Type	TM1 Server
Generate feeders automatically	No
User	Admin
Authentication	BASIC
Settings	0

- To prevent Performance Modeler from generating feeders when modeling/creating new rules, Disable feeder generation explicitly per cube in }CubeAttributes by ensuring the setting for NoFeeders is 1 for every cube:

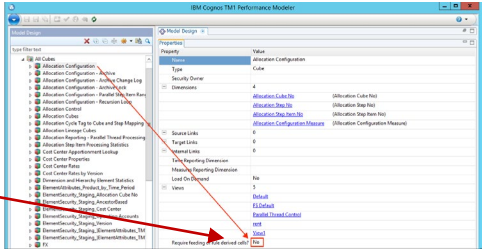


Cube Viewer: Allocations3->}CubeAttributes->Default

[Base]

}CubeAttributes

}Cubes:Default	NoFeeders
Allocation Configuration	1
Allocation Configuration - Archive	1
Allocation Configuration - Archive Change Log	1
Allocation Configuration - Archive Lock	1
Allocation Configuration - Parallel Step Item Range	1
Allocation Configuration - Recursion Loop	0
Allocation Control	1
Allocation Cubes	1
Allocation Cycle Tag to Cube and Step Mapping	1
Allocation Lineage Cubes	1
Allocation Reporting - Parallel Thread Processing - Config	1
Allocation Step Item Processing Statistics	1


IBM Cognos TM1 Performance Modeler

Model Design

type filter text

- All Cubes
  - Allocation Configuration
  - Allocation Configuration - Archive
  - Allocation Configuration - Archive Change Log
  - Allocation Configuration - Archive Lock
  - Allocation Configuration - Parallel Step Item Rang
  - Allocation Configuration - Recursion Loop
  - Allocation Control
  - Allocation Cubes
  - Allocation Cycle Tag to Cube and Step Mapping
  - Allocation Lineage Cubes
  - Allocation Reporting - Parallel Thread Processing
  - Allocation Step Item Processing Statistics
  - Cost Center Apportionment Lookup
  - Cost Center Properties
  - Cost Center Rates
  - Cost Center Rates by Version
  - Dimension and Hierarchy Element Statistics
  - ElementAttributes_Product_by_Time_Period
  - ElementSecurity_Staging_Allocation Cube No
  - ElementSecurity_Staging_AncestorBased
  - ElementSecurity_Staging_Cost Center
  - ElementSecurity_Staging_Operating Accounts
  - ElementSecurity_Staging_Version
  - ElementSecurity_Staging_}ElementAttributes_TM
  - ElementSecurity_Staging_}ElementAttributes_TM
  - FX

Model Design

Properties

Property	Value	
Name	Allocation Configuration	
Type	Cube	
Security Owner		
Dimensions	4	
	Allocation Cube No	(Allocation Cube No)
	Allocation Step No	(Allocation Step No)
	Allocation Step Item No	(Allocation Step Item No)
	Allocation Configuration Measure	(Allocation Configuration Measure)
Source Links	0	
Target Links	0	
Internal Links	0	
Time Reporting Dimension		
Measures Reporting Dimension		
Load On Demand	No	
Views	5	
	Default	
	FS Default	
	Parallel Thread Control	
	rent	
	View1	
Require feeding of rule derived cells?	No	

General Architecture & Design: Security

- Do not use rules in }CubeSecurity.cub, }DimensionSecurity.cub, }ProcessSecurity.cub, }ElementSecurity_<Dimension>.cub
- Cube Security (‘}CubeSecurity.cub’), Dimension Security (‘}DimensionSecurity.cub’), Process Security (‘}ProcessSecurity.cub’), and Element Security Data (‘}ElementSecurity_<Dimension>.cub’) should be processed via TI instead of cube rules.
- If rules used, a security metadata change - for example due to a hierarchy change (with corresponding/resulting security changes for parent and/or child nodes) or due to a new element being added to a hierarchy (like a new archived version for which READ access now to be granted to all applicable groups) – will always require running the ‘SecurityRefresh()’ command in TM1, effectively rendering all cached security settings invalid and hence renewing/refreshing all security credentials.
- A security refresh on large models will typically lead to a multi- to many minute lock of all user activity due to TM1 refreshing security access credentials for all active users and groups.
- If security is manually entered or processed via TI (and hence directly stored in the corresponding security cube), a security refresh is not necessary for such security changes. The security changes will propagate automatically and with only very short locks.
- Optimize CellSecurity cubes by only using dimensions that are needed for determining cell security access credentials (see next slide for a utility to optimize cell security cubes via TM1 Architect)
- Be aware that CellSecurity by default may overrule ElementSecurity by allowing data access to cells without ElementSecurity access credentials. I.e. if a cell in CellSecurity is READ, but the Element is set to access NONE (empty), a user can still read the value in Excel by typing in the Element Name. Use }CubeSecurityProperties.cub and set CellSecurityMostRestrictive to Yes to only allow Cell Security to restrict access further (and to prevent cell security from being able to ‘open’ access.
- For details, go to:

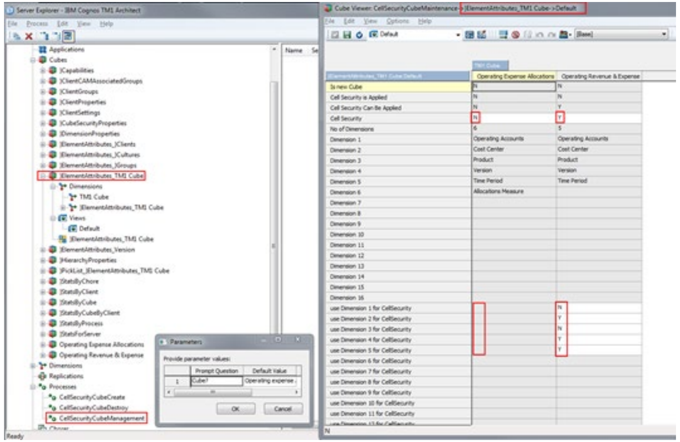
  [Proven Practices for Security Management and Auditability]

  [Workflow with TM1 and Planning Analytics: Workflow Types, Design Options & Guidelines]

General Architecture & Design: Security

Click here for a utility for the creation of optimized CellSecurity Cubes. Note: the utility does not save and re-attach cell security rules. Ensure that a backup of the existing cell security rules is available and attach the rules after the new cell security cube was created. How to use the utility:

1) Add the cube name to dimension ‘TM1 Cube.dim’
2) To delete an existing cell security cube,
   a) go to }ElementAttributes_TM1 cube.cub and set value for attribute ‘Cell Security’ to N
   b) then run process ‘CellSecurityCubeManagement.pro’
3) To create a new cell security cube,
   a) go to }ElementAttributes_TM1 cube.cub and set value for attribute ‘Cell Security’ to Y
   b) then, in }ElementAttributes_TM1 cube.cub, configure the cell security cube dimensions like in the following screenshot:


Server Explorer - IBM Cognos TM1 Architect
File Process Edit View Help
Applications
Cubes
}Capabilities
}ClientCAMAssociatedGroups
}ClientGroups
}ClientProperties
}ClientSettings
}CubeSecurityProperties
}DimensionProperties
}ElementAttributes_}Clients
}ElementAttributes_}Cultures
}ElementAttributes_}Groups
}ElementAttributes_TM1 Cube
Dimensions
TM1 Cube
}ElementAttributes_TM1 Cube
Views
Default
}ElementAttributes_TM1 Cube
}ElementAttributes_Version
}HierarchyProperties
}PickList_}ElementAttributes_TM1 Cube
}StatsByChore
}StatsByClient
}StatsByCube
}StatsByCubeByClient
}StatsByProcess
}StatsForServer
Operating Expense Allocations
Operating Revenue & Expense
Dimensions
Replications
Processes
CellSecurityCubeCreate
CellSecurityCubeDestroy
CellSecurityCubeManagement
Chores
Ready
Name
Cube Viewer: CellSecurityCubeMaintenance->}ElementAttributes_TM1 Cube->Default
File Edit View Options Help
Default
[Base]
OK
Cancel

Parameters

Provide parameter values:

	Prompt Question	Default Value
1	Cube?	Operating expense

}ElementAttributes_TM1 Cube Default	Operating Expense Allocations	Operating Revenue & Expense
Is new Cube	N	N
Cell Security is Applied	N	N
Cell Security Can Be Applied	N	Y
Cell Security	N	Y
No of Dimensions	6	5
Dimension 1	Operating Accounts	Operating Accounts
Dimension 2	Cost Center	Cost Center
Dimension 3	Product	Product
Dimension 4	Version	Version
Dimension 5	Time Period	Time Period
Dimension 6	Allocations Measure	
Dimension 7		
Dimension 8		
Dimension 9		
Dimension 10		
Dimension 11		
Dimension 12		
Dimension 13		
Dimension 14		
Dimension 15		
Dimension 16		
use Dimension 1 for CellSecurity		N
use Dimension 2 for CellSecurity		Y
use Dimension 3 for CellSecurity		N
use Dimension 4 for CellSecurity		Y
use Dimension 5 for CellSecurity		Y
use Dimension 6 for CellSecurity		
use Dimension 7 for CellSecurity		
use Dimension 8 for CellSecurity		
use Dimension 9 for CellSecurity		
use Dimension 10 for CellSecurity		
use Dimension 11 for CellSecurity		

4) then run process ‘CellSecurityCubeManagement.pro’
5) to re-create an existing cell security cube with different dimensions, first execute step 2 and then step 3

General Architecture & Design: Optimizing Processing Speed

1. **Use CubeClearData() prior to a re-load of all data in a cube**
2. **Use ViewZeroOut() to zero-out sections of a cube**
3. **Separate TI-processing for**
   a) **Dimension Load/Update &**
   b) **Data (Fact) Load/Update**

   Doing so will – via procedural separation of metadata and data-update procedures - allow the load and update of cube data using parallel load threads, hence significantly improving cube load time.

4. **Where applicable** (i.e. for adding not yet assigned masterdata elements), **use the new TM1 10 functions DimensionElementInsertDirect and DimensionElementComponentAddDirect** directly in the TM1 data tab along with loading new data (unless the dimension is very large),

5. **Utilize a Parallel Data Load Regime for loading very large amounts of data very rapidly:**

General Architecture & Design: Parallel Data Load & Processing

- **What is it?**

  In a parallel data-load regime, data is loaded and processed
  into **one** or multiple cubes via separate & **<u>parallel</u> processing threads**.

- **What is it good for?**

  - With parallel data load / processing / export, a cube can be processed using as many CPU threads as available and applicable, hence providing significant performance gains.
  - Using a parallel data load regime/framework, TM1 can process upwards of 50.000 records per second per CPU core*
  - => Using 16 CPUs for parallel data load, this can result in an overall data-load/update speed of roughly 48Mio records per second
  - => Using 16 CPUs for parallel data load, you can load/update roughly 2.88 Billion records (2.880.000.000) in one hour, or 1.000.000.000 in just 21 minutes!

*: on high-performance CPUs at over 3Ghz

General Architecture & Design: Parallel Data Load & Processing

- In cases where parallel data load is to be used frequently/extensively, the use of TM1RunTI.exe or RunProcess to trigger multiple data load processes built to allow parallel execution and data load is the preferred method to initiate parallel data load.
- Proven Practices on how to initiate and manage data load & processing threads using RunTI.exe and Unix, Linux & Windows scripting language are available per request.
- See Proven Practices for Parallel Data Load and Processing in TM1 for details. Templates/Assets can for parallel data load and processing can be downloaded here.
- Solution Assets such as the IBM Planning Analytics Solution for Allocations and Profitability Modeling already contain utilities for parallel data processing.
- For information on how to manage TI processes using Cognos Command Center, please refer to the Cognos Command Center Online Documentation or to http://www-01.ibm.com/support/knowledgecenter/SSPLNP_10.2.1/com.ibm.swg.ba.cognos.ag_ccc.10.2.1.doc/c_ug_ccc_pi_tm1_overview.html for more information on TM1 Integration

General Architecture & Design: Minimizing Contention

1. **Use the new PA TM1s.cfg parameters to reduce contention**
2. **Use SaveDataAll() & CubeSaveData() only when and where needed**: SaveDataAll() will acquire a write-lock on the database, CubeSaveData() will acquire a write lock on the cube. It is hence not recommended to use SaveDataAll()/CubeSaveData() at the end of each TI load process (because that will cause lock-contention with other load processes that load data to any cube (SaveDataAll()) or also load data to the same cube (CubeSaveData() in a parallel load scenario (see below).
3. **Use Unique Views and Subsets for TI processing** used in the context of TI-processing. Any subset and view (and the subsets used by that view) that is used by a TI process should have a name that is unique to the user executing the process and the time the process is executed. Using subsets and views with unique names ensures that
   - The Subsets and Views can - if needed – be isolated and debugged without affecting other processing (be it by the same process or by different processes)
   - No contention will occur between different TI processes creating and/or destroying subsets and views at the same time
   - No contention will occur between the same TI processes creating and/or destroying subsets and views at the same time
4. **Implement a synchronization & serialization framework for applicable TI processes** (NOT for all processes – you want to be able to run in parallel what shall run in parallel)

General Architecture & Design: Minimizing Contention

4. Implement a synchronization & serialization framework for <u>applicable</u> TI processes! Why?

- The concurrent execution of Turbo Integrator processes that lock the same objects and block each other can lead to contention.
- TI process contention typically expresses itself successive rollbacks and retries, also known as thrashing. Examples:
  - a) Two or more TI processes may perform an update of one and the same dimension
  - b) The first process to acquire the dimension 'lock' will block the other process from continuing
  - c) The 2nd process will hence do a rollback of its actions before encountering the lock and then will attempt to start anew
  - d) The 2nd process may possibly hit the same lock later should process 1 not have finished or have released the lock (again resulting in a roll-back, i.e. back to the beginning...).
  - e) Or the 2nd process will find that the lock was released, but: the 2nd TI process ,may obtain a lock on a different object that is needed by the first process (but later on, i.e. towards the end of the 1st process running
  - f) Now the 2nd process is locking the 1st process & the 1st process will do a roll-back...
  - ⇒ Not only may there be roll-backs and retries, but the rollbacks and retries – depending on the TI procedures in place, may affect one another, resulting in two or more processes locking each other out.

General Architecture & Design:
Minimizing Contention - a Lock Contention Example

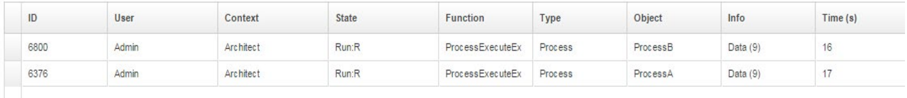
4. Implement a synchronization & serialization framework for <u>applicable</u> TI processes! Why?

The concurrent execution of Turbo Integrator processes that lock the same objects and block each other can lead to contention. TI process contention typically expresses itself successive rollbacks and retries, also known as thrashing. Examples:

a) Two or more TI processes may perform an update of one and the same dimension
b) The first process to acquire the dimension 'lock' will block the other process from continuing
c) The 2nd process will hence do a rollback of its actions before encountering the lock and then will attempt to start anew
d) The 2nd process may possibly hit the same lock later should process 1 not have finished or have released the lock (again resulting in a roll-back, i.e. back to the beginning...).
e) Or the 2nd process will find that the lock was released, but: the 2nd TI process ,may obtain a lock on a different object that is needed by the first process (but later on, i.e. towards the end of the 1st process running
f) Now the 2nd process is locking the 1st process & the 1st process will do a roll-back...

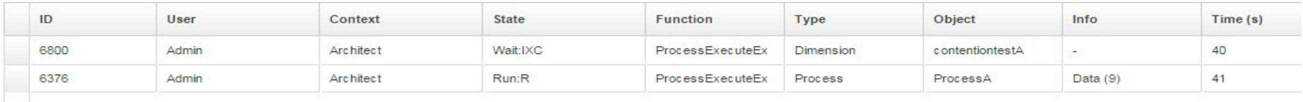
⇒ Not only may there be roll-backs and retries, but the rollbacks and retries – depending on the TI procedures in place, may affect one another, resulting in two or more processes locking each other out.

Two Processes start at the same time & both process complete the metadata procedures and start the data tab:



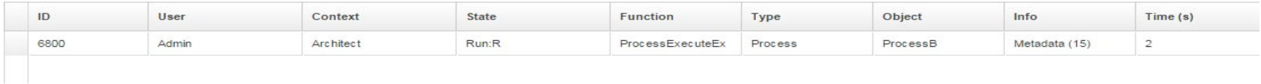
ID	User	Context	State	Function	Type	Object	Info	Time (s)
6800	Admin	Architect	Run:R	ProcessExecuteEx	Process	ProcessB	Data (9)	16
6376	Admin	Architect	Run:R	ProcessExecuteEx	Process	ProcessA	Data (9)	17

When Process B encounters a lock (in the data tab, of dimension 'ContentiontestA') & consequently enters a wait state:



ID	User	Context	State	Function	Type	Object	Info	Time (s)
6800	Admin	Architect	Wait:IXC	ProcessExecuteEx	Dimension	contentiontestA	-	40
6376	Admin	Architect	Run:R	ProcessExecuteEx	Process	ProcessA	Data (9)	41

When Process A has finished and the lock on dimension 'ContentiontestA' has been released, Process B starts anew:



ID	User	Context	State	Function	Type	Object	Info	Time (s)
6800	Admin	Architect	Run:R	ProcessExecuteEx	Process	ProcessB	Metadata (15)	2

General Architecture & Design: Minimizing Contention

So what do we need to do to prevent thrashing?

⇒ Applicable processes should be kept from being executed in parallel and instead should only be allowed to be run in serial to avoid thrashing and excessive (or even endless) rollback actions.

⇒ The same applies to simultaneous fact updates and dimension updates: if you are running a process that updates fact data you should not attempt to update the dimension master data using a different process as this will cause locking and rollbacks.

So how do we do this?

⇒ A semaphore logic needs to be employed to synchronize certain (applicable) TI processes to run in serial execution mode only. The synchronization logic will ensure that a process is told to ‘wait’ at the very beginning of execution (before it would require a roll-back or data updates etc.) until the ‘lock’ that would prevent the process from continuing is released.

⇒ As of TM1 10, the **SYNCHRONIZED() function** allows for easy serialization of TI processes if needed:

- For TI processes that are built specifically to updated a certain dimension, add the following code to the beginning of the TI process (TI prolog): **SYNCHRONIZED (<DimensionName>);**
- Proven Practice: Use **Dimension Names as the lock names** for SYNCHRONIZED()
- For TI processes that are parameterized such that multiple or varying cubes, dimensions or objects could be accessed and locked, implementation of a configurable TM1 **TI-Synchronization utility** is recommended. The utility is to leverage the SYNCHRONIZED() function in combination with a configurable Lookup model to allow setting, maintaining and testing different serialization configurations in an ad-hoc/dynamic fashion. Click here for Synchronization Utility Objects

TM1 Web
&
Planning Analytics Workspace
(PAW)

Planning Analytics Local: TM1Web on Planning Analytics (from Apache to WebSphere Liberty)

- TM1 Web installation
- TM1 Web Integrated Login (configuration)
- TM1 Web Configuration Parameters
- TM1 Web & WebSphere Liberty Application Server configuration and performance tuning
- Advanced TM1 Web Logging
- TM1 Web-Sheet load optimization
- Increasing Heap Size in TM1. Important Note on Heap Size / Maximum Memory for WebSphere Liberty:

  ⇒ especially in environments where use of TM1 Web is not managed by IT but the TM1 Web environment is managed by IT, it is a **good practice to set the Maximum WebSphere Liberty Memory to a significantly higher value that the default of 1.5 GB**.

  ⇒ When upgrading from TM1 Web 10.2.2 to TM1 Web PA, set the Maximum memory in WebSphere Liberty to the same value as Apache Tomcat was set to in TM1 Web 10.2.2. Note the recommended limits for PA 2.0.5 or lower and upgrade to PA 2.0.6 or higher if needed (it is NOT recommended to have a Java Heap set to larger than 24GB for Planning Analytics 2.0.5 and lower (which uses JRE 7). As of Planning Analytics 2.0.6 and higher (JRE 8), the limit may be increased but not recommend to be set larger than 56GB. The max Java Heap allowed while also maintaining -Xcompressedrefs is 24GB (2.0.5 and lower) or 56GB (2.0.6 and higher))

Planning Analytics Local:
Planning Analytics Workspace (PAW)

- Installing Planning Analytics Workspace (PAW): Overview
- **PAW Architecture and advanced Administration**

Organization, DevOps Optimization and Cloud-Readiness:

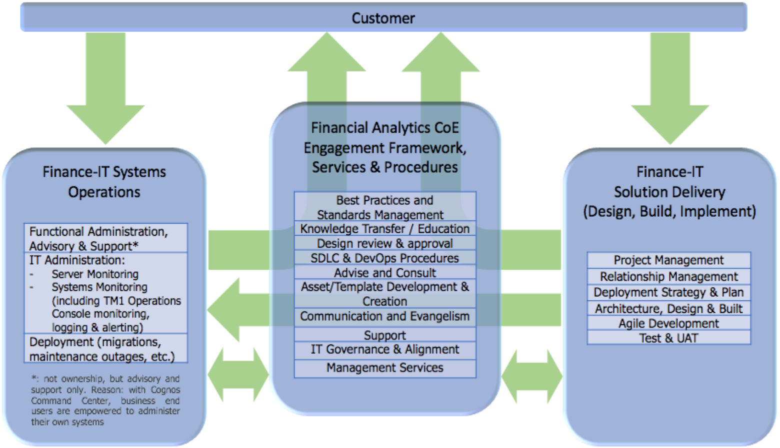
Considerations for

- Organization
- Operations Management & Orchestration
  -> Cognos Command Center
- Data Integration
  -> Cognos Integration Server
- Performance and Regression Testing
- Migration

Organization, DevOps Optimization, and Cloud Readiness

- Organization
- DevOps Optimization
- Assessment tools

- Strategy & Organizational Advisory
  - Analytics Center of Excellency (ACE) Workshop
  - The Analytics Center Of Excellency (ACE) in an IBM Planning Analytics context: Focus Areas, Responsibilities & Roles
  - Transitioning to Modern Self-Service Analytics: Workshop & Advisory
  - Financial Analytics Infrastructure & Solutions Strategy Assessment



Customer

Finance-IT Systems Operations

Functional Administration, Advisory & Support*

IT Administration:
- Server Monitoring
- Systems Monitoring (including TM1 Operations Console monitoring, logging & alerting)

Deployment (migrations, maintenance outages, etc.)

*: not ownership, but advisory and support only. Reason: with Cognos Command Center, business end users are empowered to administer their own systems

Financial Analytics CoE Engagement Framework, Services & Procedures

Best Practices and Standards Management

Knowledge Transfer / Education

Design review & approval

SDLC & DevOps Procedures

Advise and Consult

Asset/Template Development & Creation

Communication and Evangelism

Support

IT Governance & Alignment

Management Services

Finance-IT Solution Delivery (Design, Build, Implement)

Project Management

Relationship Management

Deployment Strategy & Plan

Architecture, Design & Built

Agile Development

Test & UAT

- DevOps Optimization and Guidance:
  - IBM Planning Analytics Developer Handbook: Sample Outline and Topics
  - IBM Planning Analytics Performance Maintenance & Tuning Team: Mandate, Benefit, Responsibilities & Enablement
  - Scenarios, Options, and Proven Practices for TM1 Code Migrations, Administrative Roles, Duties, and Procedures
  - TM1 Operations Console,
  - Interpreting TM1Top and Operations Console Output
  - Regression- & Performance Testing of TM1 & Planning Analytics Solutions: Recommendations & Guidelines
  - PAW Local Operation and Administration
  - Misc. PA Local installation and configuration

- Assessment tools:
  - Financial Analytics Assessment Pre-Engagement Questionnaire
  - FP&A Process Assessment Questionnaire
  - IBM Planning Analytics - Project Scoping Questionnaire

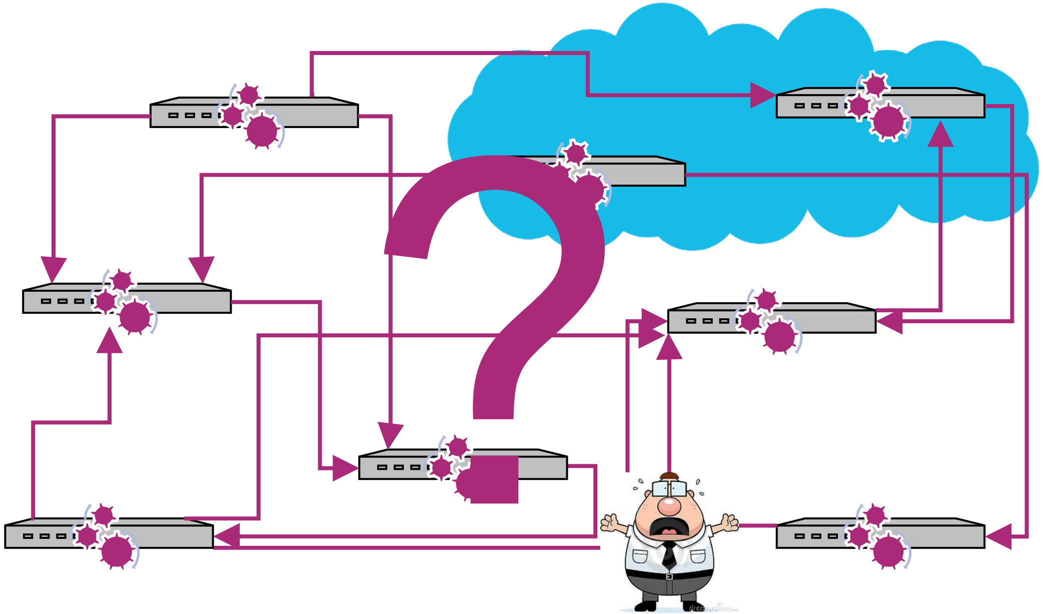
Cognos Integration Server

In the context of MDM, Master- & Meta-data Governance, Data-Integration & Transformation, IBM Cognos Integration Server provides out-of-the-box hybrid-cloud integration capabilities, hence minimizing compliance risk and improving IT-process efficiency

Feature	Advantage	Out-of-the-Box Benefit
Integrated and Automated Extraction	• Automates data extraction • Extract TM1, Oracle Hyperion Essbase, HFM, SAP BW Fact-, Meta-, & Master-Data • Dynamically and Automatically generates • RDBMS Target Schemas • Files • Target Dimensions and Cubes • TM1 TI-scripts	• Establish and maintain a Dimension Meta- & Master-data backup & repository • Automatically Integrate different (Meta)-Data formats • Automate Meta-, Master- & Fact-Data export/import • Integration and export/import from • Cube Model to RDBMS (CIS Star Schema), • Cube Model to Text File, • Cube Model to Cube Model, • CIS Star Schema to Star Schema • CIS Star Schema to Text File • CIS Star Schema to Cube Model
Hybrid-Cloud Support	Native, RestAPI-based support of TM1, Planning Analytics Local, Planning Analytics Cloud	• Out-of-the-box integration of Hybrid-Cloud TM1/PA Environments • (on premise to on premise, on premise to cloud, cloud to on premise, cloud to cloud) • Transfer/Replicate/Copy data between TM1/Planning Analytics instances
Performance Optimized	supports continuous, near real-time extraction & import	Support for continuous planning & analysis in a Hybrid-Cloud environment

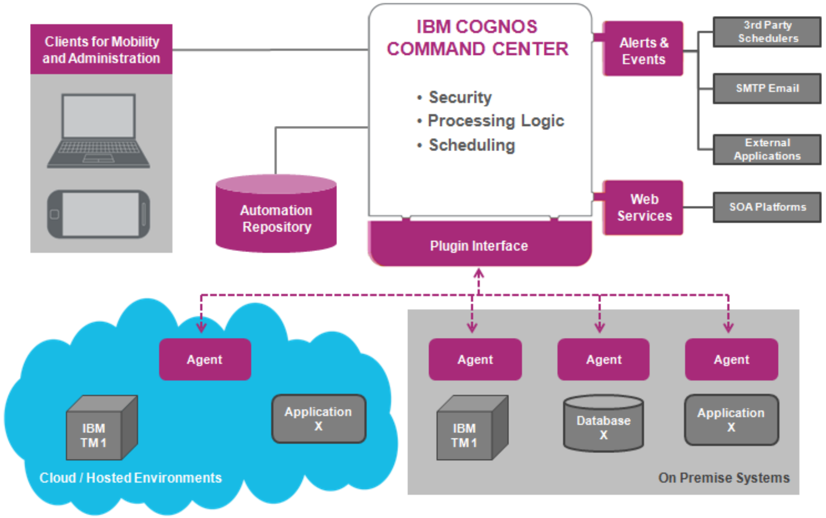
Cognos Command Center

Cognos Command Center provides a solution for orchestrating recurring maintenance & administration tasks around Master-/Meta- & Fact-Data Management for TM1, hence unlocking significant DevOps productivity gains while establishing full procedural auditability



Cognos Command Center

- Orchestrates applications, either on-premise or in the cloud
- Libraries of business functions eliminating custom code
- Business friendly interface that supports mobility
- Central repository for compliance and audit
- Handshake with enterprise schedulers and web Services
- Each application with a command-line interface



Features:

❿ Email automation
❿ File automation (copy, move, delete,…)
❿ FTP, SFTP, FTPS automation
❿ Systems & Monitoring utilities
❿ IBM Cognos Integration Server integration (run CIS tasks)
❿ RDBMS:
  ❿ SQL Server,
  ❿ DB2,
  ❿ Oracle DB

❿ IBM Cognos TM1:

- Check TM1 Server
- Delete Server Object
- View & Filter Server Log, Audit Log, Transaction Log
- List Server Objects
- List Users
- List/Delete Views
- Manage BLOBs
- Read/Write Cell
- Run TI Chore, TI Process
- Start/Stop TM1 Server (Windows)

Cognos Integration Server & Cognos Command Center Guides, Training, & Tutorial

- Click here to access a Cognos Command Center (CCC) deck which includes a thorough introduction and tutorial
- Click here to access a folder with CCC & CIS training courses (ppt decks and recordings)
- Click here for a document on how to use the CIS command line utility (highly recommended)

Performance and Regression Testing, TM1 Rest API

FOPM Performance and Regression Testing: Proven Practices

Tools and methodologies for testing via TM1 Rest API:

- Cucumber (click here for some guidance)
- IBM DeveloperWorks API Testing Framework Using BDD Approach Through Cucumber and Rest- Assured (involving Jenkins)

TM1 Rest API Materials:

- Introduction and Tutorial on the TM1 Rest API
- TM1 Rest API Developer Guide
- TM1 Rest API wrapped into Python
- Mastering the Rest API with Postman

Migrations
Separation of Roles, Duties & Procedures

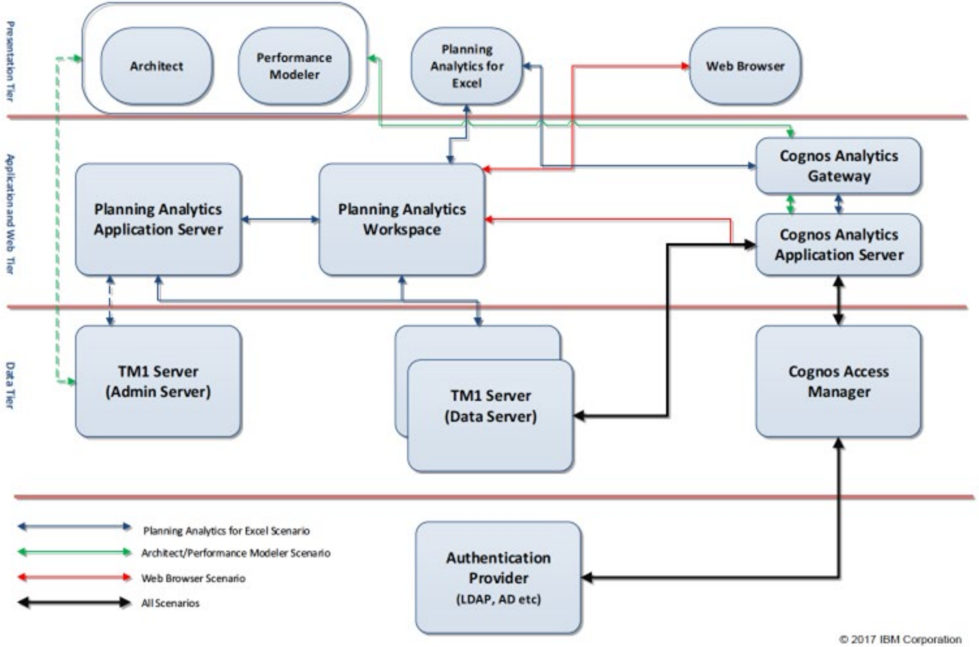
- **Migrations**: As of Planning Analytics 2.0.7, the recommended migration method is via Git source control: To learn more, see Managing TM1 database assets with Git and an introductory guide to Planning Analytics Git integration.
- IBM Cognos TM1 & Planning Analytics Code Migrations, **Separation of Roles, Duties & Procedures**: Scenarios, Options & Proven Practices

Configuring
Planning Analytics Local
Security Authentication

Security Authentication

Configuring CAM Security for

- TM1, TM1 Web, PMHUB, PMPSVC
- Panning Analytics Workspace (PAx via PAW)
- Detailed Walkthrough and Guidance on CAM Security with PA



Presentation Tier
Architect
Performance Modeler
Planning Analytics for Excel
Web Browser
Application and Web Tier
Cognos Analytics Gateway
Planning Analytics Application Server
Planning Analytics Workspace
Cognos Analytics Application Server
Data Tier
TM1 Server (Admin Server)
TM1 Server (Data Server)
Cognos Access Manager
Planning Analytics for Excel Scenario
Architect/Performance Modeler Scenario
Web Browser Scenario
All Scenarios
Authentication Provider (LDAP, AD etc)

Encryption

- Data Transmission Encryption
- Encryption at Rest (on Premises only)

SSL/TLS encryption of TM1/PA Data Transmissions

- SSL encryption via TLS 1.2 is enabled with TM1s.cfg parameter UseSSL=T
- Overview of TLS Data Transmission Security
- To configure custom (WBA-issued) SSL certificates, please refer to the following links:
  - Using Custom TLS
  - Planning Analytics 2.0 Local - Configure Custom SSL Certificates

New with Planning Analytics: Encryption at Rest

- Why ‘Encryption at Rest’?

  In its default state (and as per a common default practices for database objects), the data objects in a TM1 database data directory and the TM1 database log files are not encrypted. In the unencrypted state, the restoration of a TM1 database is possible just based on data directory files and - where applicable - transaction logs. Additionally, unencrypted TM1 objects can be read using a text editor.

  => Ideally, we would want to have the TM1 Database Objects be encrypted. That is what Encryption at Rest does:

- What is ‘Encryption at Rest’?

  Encryption at Rest provides the capability to encrypt TM1 objects and log files such that their content cannot be read and such that the objects themselves are insufficient to restore a TM1 database without having access to the encryption keys used by the TM1 server.

  Click here for a comprehensive guide on how to configure, enable and manage TM1 Database Encryption at Rest

IBM Planning Analytics Proven Practices, Assets, Utilities, and Templates

IBM Planning Analytics Proven Practices, Assets, Utilities, and Templates

Proven Practices, Assets, Utilities, Configuration & Design Guidelines for TM1 Modeling and Configuration

- **Introductory Guidelines and Proven practices for TM1 and Planning Analytics Configuration, Architecture, and Design (or "*An introductory architecture and design guide for building very large yet fast performing TM1/PA models*")**
- Proven Practices for Version/Scenario Management,
- TM1 Multi-Threaded-Queries: Configuration Guidelines
- Guidelines & Proven Practices for IBM Planning Analytics and IBM Cognos TM1 Integration and Performance Optimization with Cognos BI and Cognos Analytics
- Use Case, Benefits & Functionalities of a Cube, Dimension, Master-, Meta-, and Fact-Data Management & Maintenance Framework for IBM Planning Analytics
- TM1 Turbo Integrator Utilities for View- & Subset-Generation (download objects here)
- TM1 Turbo-Integrator Utilities for Dimension and Hierarchy Maintenance (utilities include TI processes to build PA V2.0 Hierarchies, using new PA V2.0 TI functions)
- Parallel Data Processing with Planning Analytics: Methods, Options, Restrictions, and Templates (download objects here)
- Planning Analytics FX conversion: Transaction/Input Currency and Reporting Currency
- Planning Analytics V2.0 Hierarchies - Guidelines and Proven Practices
- and more

Security Management and Automation

- Proven Practices for Security Management and Auditability
- Management of TM1 Security: An Introduction
- IBM Planning Analytics Security Management Framework
- Utility for CellSecurity Cube Optimization

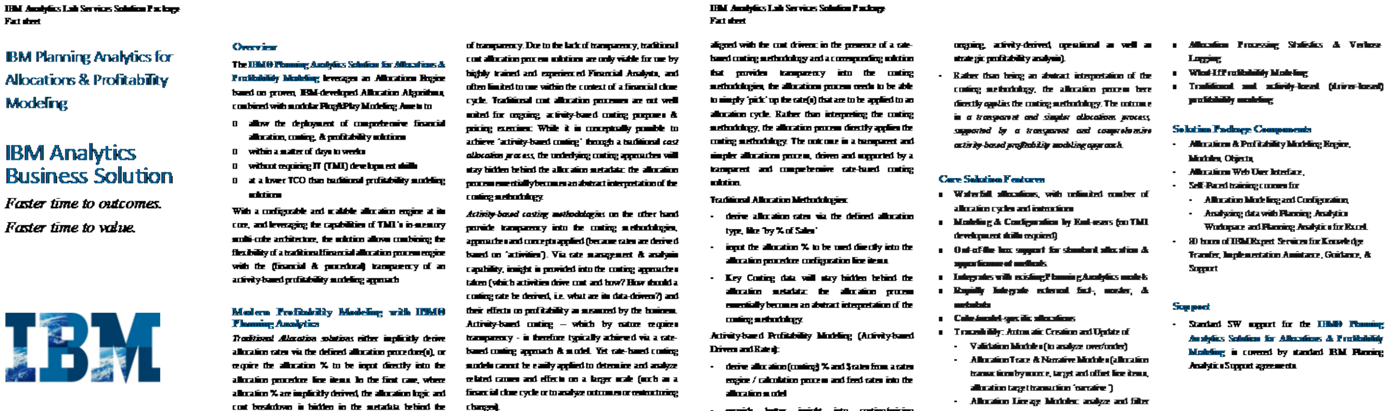
Workflow

- Workflow with TM1 and Planning Analytics: Workflow Types, Design Options & Guidelines

Authentication

- IBM Planning Analytics Local: Security Authentication Modes

IBM Planning Analytics Solution for Allocations and Profitability Modeling
Solution Fact Sheet



IBM Analytics Lab Services Solution Package
Fact sheet

IBM Planning Analytics for Allocations & Profitability Modeling

IBM Analytics Business Solution

Faster time to outcomes.

Faster time to value.

Overview

The **IBM® Planning Analytics Solution for Allocations & Profitability Modeling** leverages an Allocations Engine based on proven, IBM-developed Allocation Algorithms, combined with modular Plug&Play Modeling Assets to

- allow the deployment of comprehensive financial allocation, costing, & profitability solutions
- within a matter of days to weeks
- without requiring IT (TM1) development skills
- at a lower TCO than traditional profitability modeling solutions

With a configurable and scalable allocation engine at its core, and leveraging the capabilities of TM1's in-memory multi-cube architecture, the solution allows combining the flexibility of a traditional financial allocation process engine with the (financial & procedural) transparency of an activity-based profitability modeling approach.

Modern Profitability Modeling with IBM® Planning Analytics

Traditional Allocation solutions either implicitly derive allocation rates via the defined allocation procedure(s), or require the allocation % to be input directly into the allocation procedure line items. In the first case, where allocation % are implicitly derived, the allocation logic and cost breakdown is hidden in the metadata behind the allocation instruction. The allocation process and underlying business logic hence suffer from a lack of transparency and often become 'unwieldy'. In the second case, where allocation % are directly entered, the cost breakdown becomes more transparent, yet the possibly high number of required allocation instructions still cause a lack of transparency. Due to the lack of transparency, traditional cost allocation process solutions are only viable for use by highly trained and experienced Financial Analysts, and often limited to use within the context of a financial close cycle. Traditional cost allocation processes are not well suited for ongoing, activity-based costing purposes & pricing exercises. While it is conceptually possible to achieve 'activity-based costing' through a traditional *cost allocation process*, the underlying costing approaches will stay hidden behind the allocation metadata: the allocation process essentially becomes an abstract interpretation of the costing methodology.

Activity-based costing methodologies on the other hand provide transparency into the costing methodologies, approaches and concepts applied (because rates are derived based on 'activities'). Via rate management & analysis capability, insight is provided into the costing approaches taken (which activities drive cost and how? How should a costing rate be derived, i.e. what are its data-drivers?) and their effects on profitability as measured by the business. Activity-based costing – which by nature requires transparency - is therefore typically achieved via a rate-based costing approach & model. Yet rate-based costing models cannot be easily applied to determine and analyze related causes and effects on a larger scale (such as a financial close cycle or to analyze outcomes or restructuring changes).

An actionable, activity-based costing approach hence should be based on cost rates and an underlying data model that manages the process from setting the rate logic to the rate calculation and analysis. Such a Rate-based costing model can then very effectively be leveraged by a cost allocations process for scenarios where cost is not directly

IBM Analytics Lab Services Solution Package
Fact sheet

aligned with the cost drivers: in the presence of a rate-based costing methodology and a corresponding solution that provides transparency into the costing methodologies, the allocations process needs to be able to simply 'pick' up the rate(s) that are to be applied to an allocation cycle. Rather than interpreting the costing methodology, the allocation process directly applies the costing methodology. The outcome is a transparent and simpler allocations process, driven and supported by a transparent and comprehensive rate-based costing solution.

Traditional Allocation Methodologies:

- derive allocation rates via the defined allocation type, like 'by % of Sales'
- input the allocation % to be used directly into the allocation procedure configuration line items
- Key Costing data will stay hidden behind the allocation metadata: the allocation process essentially becomes an abstract interpretation of the costing methodology

Activity-based Profitability Modeling (Activity-based Drivers and Rates):

- derive allocation (costing) % and $ rates from a rates engine / calculation process and feed rates into the allocation model
- provide better insight into costing/pricing approaches taken (which activities drive cost and how? how should a corresponding costing rate be derived?) and their effects on profitability as measured by the business.
- Higher financial and procedural transparency ⇒ better suited in a Business Economics context (for ongoing, activity-derived, operational as well as strategic profitability analysis).
- Rather than being an abstract interpretation of the costing methodology, the allocation process here directly applies the costing methodology. The outcome is *a transparent and simpler allocations process, supported by a transparent and comprehensive activity-based profitability modeling approach.*

Core Solution Features

- Waterfall allocations, with unlimited number of allocation cycles and instructions
- Modeling & Configuration by End-users (no TM1 development skills required)
- Out-of-the box support for standard allocation & apportionment methods
- Integrates with existing Planning Analytics models
- Rapidly Integrate external fact-, master, & metadata
- Cube/model-specific allocations
- Traceability: Automatic Creation and Update of
    - Validation Modules (to analyze over/under)
    - Allocation Trace & Narrative Modules (allocation transactions by source, target and offset line items, allocation target transaction 'narrative')
    - Allocation Lineage Modules: analyze and filter allocations by allocation type, driver, sources, targets, offsets, etc.
- Fast performance & high scalability: leverage parallel allocation processing algorithms & multi-threaded query engine for the speedy processing and analysis of millions to billions of records
- Allocation Processing Statistics & Verbose Logging
- What-If Profitability Modeling
- Traditional and activity-based (driver-based) profitability modeling

Solution Package Components

- Allocations & Profitability Modeling Engine, Modules, Objects,
- Allocations Web User Interface,
- Self-Paced training courses for
    - Allocation Modeling and Configuration,
    - Analyzing data with Planning Analytics Workspace and Planning Analytics for Excel
- 80 hours of IBM Expert Services for Knowledge Transfer, Implementation Assistance, Guidance, & Support

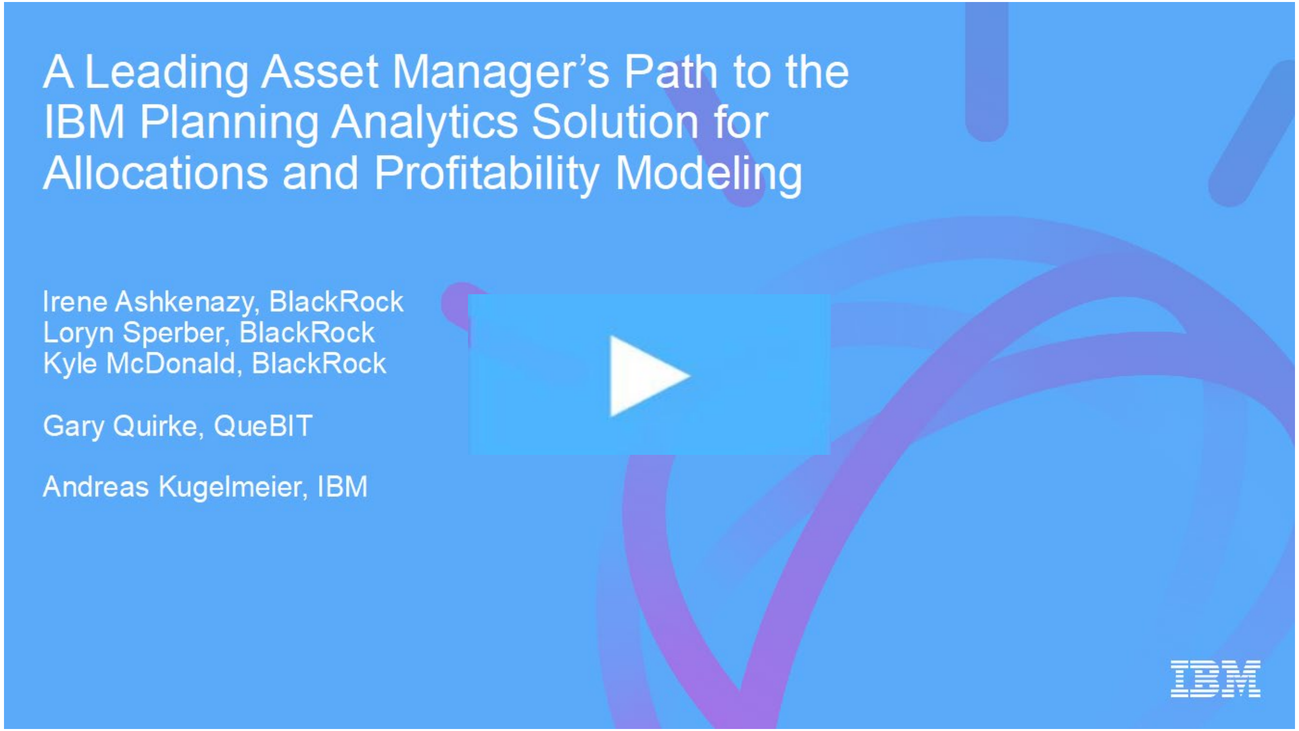
Support

- Standard SW support for the **IBM® Planning Analytics Solution for Allocations & Profitability Modeling** is covered by standard IBM Planning Analytics Support agreements.

IBM Planning Analytics for
Allocations & Profitability Modeling

Solution Overview

IBM Planning Analytics Solution for Allocations and Profitability Modeling: A Customer Story



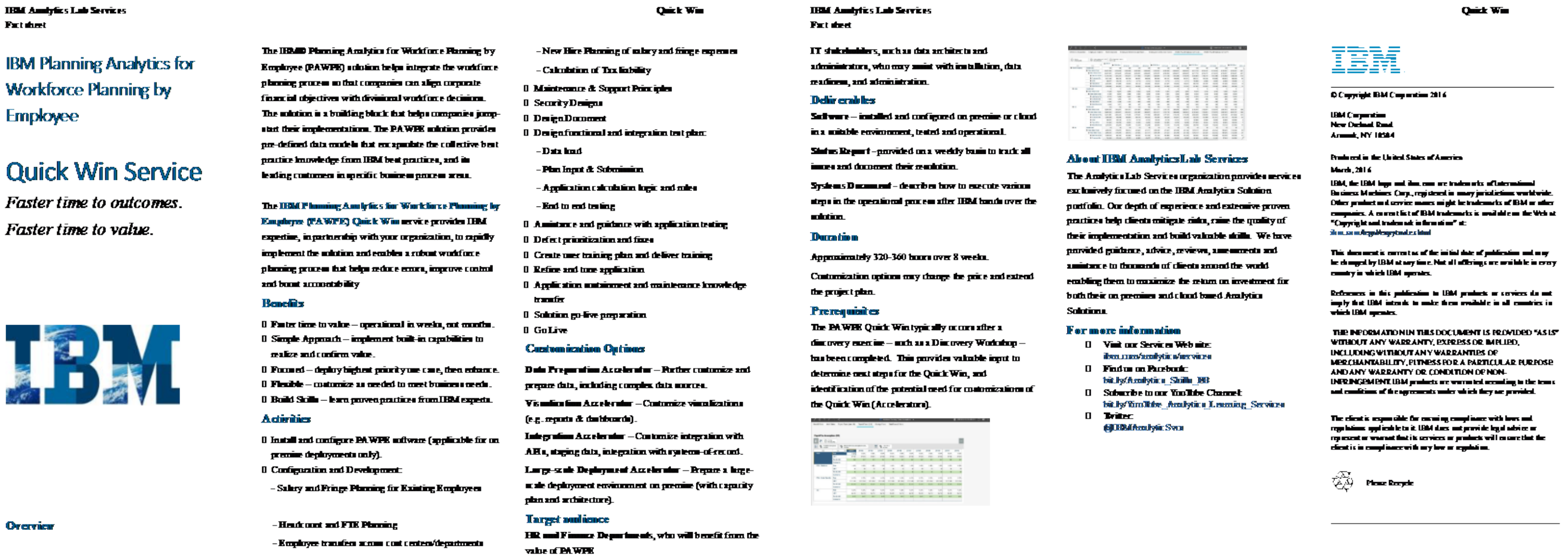
A Leading Asset Manager's Path to the IBM Planning Analytics Solution for Allocations and Profitability Modeling

Irene Ashkenazy, BlackRock
Loryn Sperber, BlackRock
Kyle McDonald, BlackRock

Gary Quirke, QueBIT

Andreas Kugelmeier, IBM

IBM Planning Analytics Solution for Workforce Planning by Employee
Solution Fact Sheet



IBM Analytics Lab Services
Fact sheet

Quick Win

IBM Planning Analytics for Workforce Planning by Employee

Quick Win Service

Faster time to outcomes.

Faster time to value.

Overview

The IBM® Planning Analytics for Workforce Planning by Employee (PAWPE) solution helps integrate the workforce planning process so that companies can align corporate financial objectives with divisional workforce decisions. The solution is a building block that helps companies jump-start their implementations. The PAWPE solution provides pre-defined data models that encapsulate the collective best practice knowledge from IBM best practices, and its leading customers in specific business process areas.

The **IBM Planning Analytics for Workforce Planning by Employee (PAWPE) Quick Win** service provides IBM expertise, in partnership with your organization, to rapidly implement the solution and enables a robust workforce planning process that helps reduce errors, improve control and boost accountability

Benefits

- Faster time to value – operational in weeks, not months.
- Simple Approach – implement built-in capabilities to realize and confirm value.
- Focused – deploy highest priority use case, then enhance.
- Flexible – customize as needed to meet business needs.
- Build Skills – learn proven practices from IBM experts.

Activities

- Install and configure PAWPE software (applicable for on premise deployments only).
- Configuration and Development:
  - – Salary and Fringe Planning for Existing Employees
  - – Headcount and FTE Planning
  - – Employee transfers across cost centers/departments
  - – New Hire Planning of salary and fringe expenses
  - – Calculation of Tax liability
- Maintenance & Support Principles
- Security Designs
- Design Document
- Design functional and integration test plan:
  - – Data load
  - – Plan Input & Submission
  - – Application calculation logic and rules
  - – End to end testing
- Assistance and guidance with application testing
- Defect prioritization and fixes
- Create user training plan and deliver training
- Refine and tune application
- Application sustainment and maintenance knowledge transfer
- Solution go-live preparation
- Go Live

Customization Options

Data Preparation Accelerator – Further customize and prepare data, including complex data sources.

Visualization Accelerator – Customize visualizations (e.g. reports & dashboards).

Integration Accelerator – Customize integration with APIs, staging data, integration with systems-of-record.

Large-scale Deployment Accelerator – Prepare a large-scale deployment environment on premise (with capacity plan and architecture).

Target audience

HR and Finance Departments, who will benefit from the value of PAWPE

IBM Analytics Lab Services
Fact sheet

Quick Win

IT stakeholders, such as data architects and administrators, who may assist with installation, data readiness, and administration.

Deliverables

Software – installed and configured on premise or cloud in a suitable environment, tested and operational.

Status Report – provided on a weekly basis to track all issues and document their resolution.

Systems Document – describes how to execute various steps in the operational process after IBM hands over the solution.

Duration

Approximately 320-360 hours over 8 weeks.

Customization options may change the price and extend the project plan.

Prerequisites

The PAWPE Quick Win typically occurs after a discovery exercise – such as a Discovery Workshop – has been completed. This provides valuable input to determine next steps for the Quick Win, and identification of the potential need for customizations of the Quick Win (Accelerators).

About IBM Analytics Lab Services

The Analytics Lab Services organization provides services exclusively focused on the IBM Analytics Solution portfolio. Our depth of experience and extensive proven practices help clients mitigate risks, raise the quality of their implementation and build valuable skills. We have provided guidance, advice, reviews, assessments and assistance to thousands of clients around the world enabling them to maximize the return on investment for both their on premises and cloud based Analytics Solutions.

For more information

- Visit our Services Web site:
  ibm.com/analytics/services
- Find us on Facebook:
  bit.ly/Analytics_Skills_FB
- Subscribe to our YouTube Channel:
  bit.ly/YouTube_Analytics_Learning_Services
- Twitter:
  @IBMAnalyticSvcs

IBM

IBM Corporation
New Orchard Road
Armonk, NY 10504

Produced in the United States of America
March, 2016

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

The client is responsible for ensuring compliance with laws and regulations applicable to it. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the client is in compliance with any law or regulation.

Please Recycle

Thank You