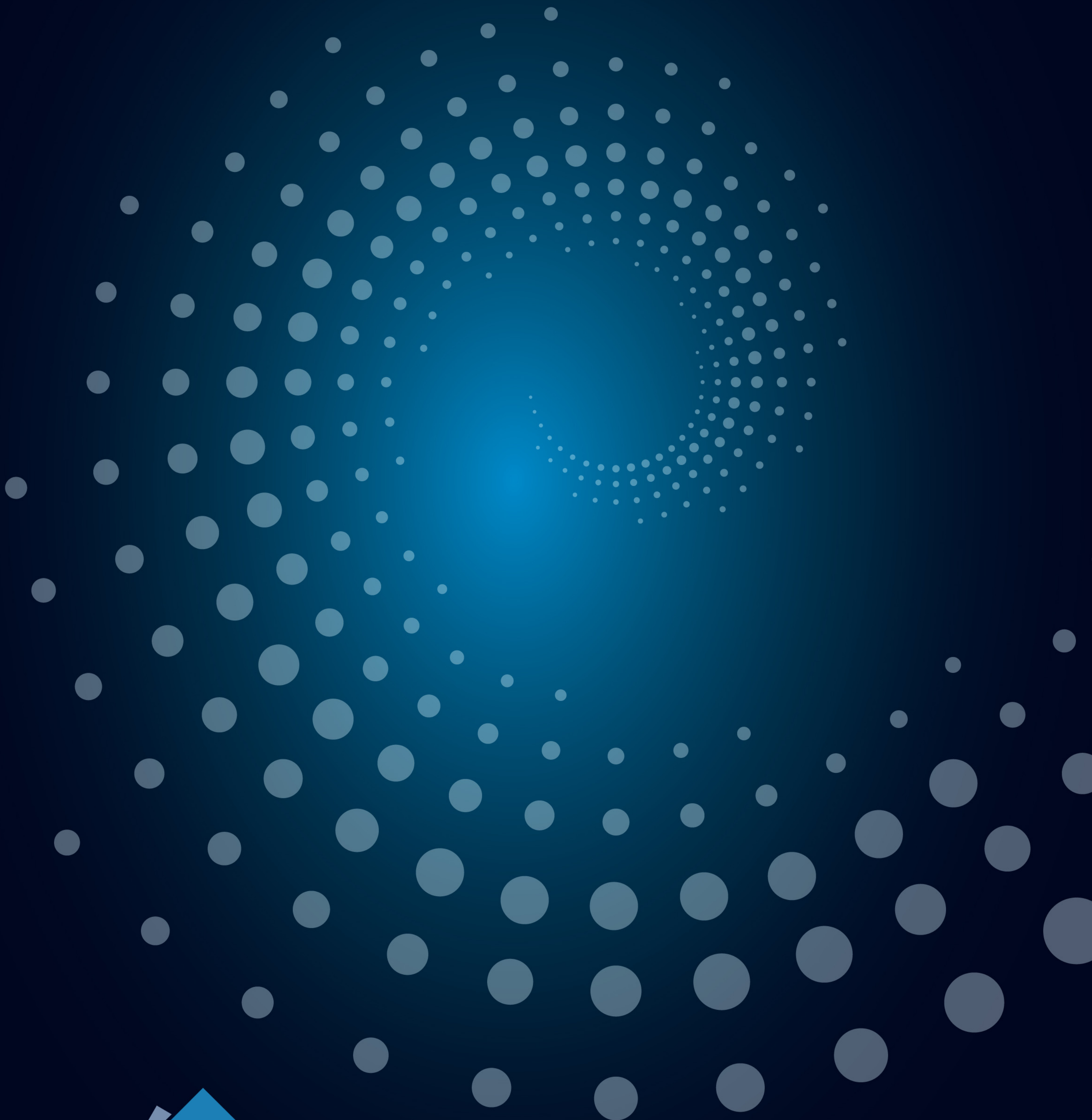
MarvelClient™ Administrator's Guide



panagenda

ADMINISTRATOR'S GUIDE

MarvelClient 3.x

Contact

panagenda Austria
(Headquarters)
panagenda GmbH
Schreyvogelgasse 3/10
AT 1010 Vienna (Austria)
Phone: +43 1 89 012 89
Fax: +43 1 89 012 89 – 15

panagenda Germany
panagenda GmbH
Lahnstraße 17
DE 64646 Heppenheim
(Germany)
Phone: +49 6252 305 28 41
Fax: +49 6252 305 284 – 2

panagenda USA
panagenda Inc.
60 State Street
Suite 700
Boston, MA 02109 (USA)
Phone: +1 850 226 9393
Fax: +1 415 449 5940

E-Mail Sales: sales@panagenda.com
E-Mail Support: support@panagenda.com
Web: www.panagenda.com

MarvelClient Administrator's Guide: 8/9/13

Table of Contents

GENERAL INFORMATION

System Requirements

- Server:
  - IBM Domino version 5.x-9.0
  - on all IBM supported platforms
  - 50 MB disk space + approx. 100 KB/end-user in Analyze Database

- Administrators workstations:
  - IBM Notes version 8.x-9.0 including the IBM Notes Browser Plugin
    (Notes 8.5 or higher is required for the decryption of centrally collected ID file copies)
  - Windows NT4SP2, 2000, XP, Vista, 7, 2003, 2008, Citrix, Intel Mac OS X and Linux
  - 5MB disk space

- End-user workstations:
  - IBM Notes version 5.x-9.0 beta including the IBM Notes Browser Plugin
  - Windows NT4SP2, 2000, XP, Vista, 7, 2003, 2008, Citrix, Intel Mac OS X and Linux
  - 5 MB disk space

Support for **beta releases**, **EOL releases** (= End Of Life = releases that are no longer supported by IBM) and operating systems no longer supported by the respective manufacturer is limited.

Architecture/Design

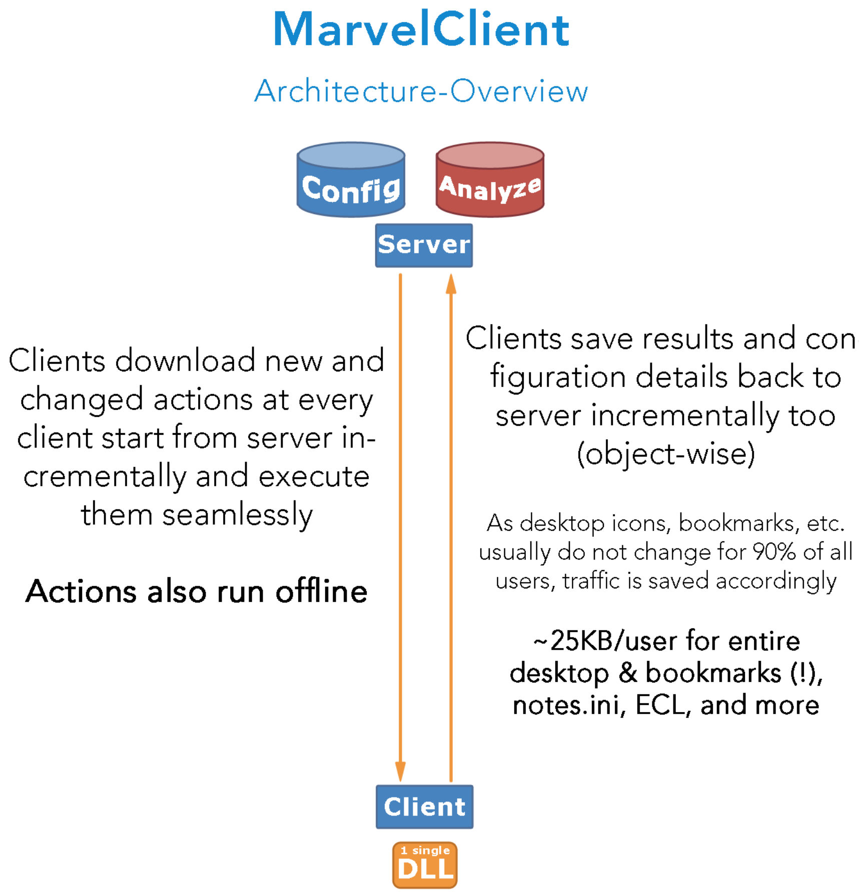
MarvelClient consists of the following three components:

1. A **Configuration database** (server) – contains instructions for clients
2. An **Analyze database** (server) – stores detailed information about clients and their respective configuration

   Typically, both of the above databases are replicated across all mail servers, which are assumed to be the servers end-users can reach most efficiently. Both databases scale along easily, even in big environments with several 100,000 users.

   No server tasks are required!

3. A **local MarvelClient file** (*mc.dll/libmc.so/libmc.dylib*) on the client-side



Clients download new and changed actions at every client start from server incrementally and execute them seamlessly

Actions also run offline

Clients save results and con figuration details back to server incrementally too (object-wise)

As desktop icons, bookmarks, etc. usually do not change for 90% of all users, traffic is saved accordingly

~25KB/user for entire desktop & bookmarks (!), notes.ini, ECL, and more

Figure 1: Architecture Overview

About the Local MarvelClient File

This (**.dll/*.so/*.dylib*) file is deployed into either the IBM Notes program or data directory and activated on clients with one additional entry in the *notes.ini* (*EXTMGR_ADDINS=*, see below). Note that for architectural reasons, on Citrix it is highly recommended to deploy the local MarvelClient file into the Notes program directory. Also, Citrix customers are advised to use the non-UPX packed version of MarvelClient, which is available from the MarvelClient support (support@panagenda.com).

The MarvelClient file auto-adapts to the respective end-user's Notes release, install type and language – from Notes 5 to including 8 Standard (=Eclipse) and Basic Edition, through local and network-, single- and multi-user-, roaming- and non-roaming-installations – meaning that one single file covers all possible IBM Notes client types and configurations.

Integration with IBM Notes

MarvelClient is entirely based on the security architecture of IBM Notes.

The integration with IBM Notes is based on the so called extension manager service, a specific form of integration into the IBM Notes/Domino platform (see: http://tinyurl.com/ax98ylc).

Extension managers are widely used for a great many different topics in the IBM Notes/ Domino world, such as: (on both Domino servers, as well as Notes clients):

- Virus scanners,
- Fax extensions, or
- Automatic attachment (de)compression

In principle an extension manager is a (*.dll/.so/.dylib*) file written in C (++), which makes use of an integration mechanism into Notes/Domino provided by IBM.

Upon starting the Notes client, each extension manager add-in is loaded, which again registers itself for a variety of subsequent events in IBM Notes (such for when attachments are being opened or saved, or the successful login of an end-user) as to be called again by IBM Notes upon each such subsequent event.

panagenda MarvelClient integrates itself into IBM Notes/Domino at several different Run Times – depending on licensed and used feature set (for a detailed description refer to "Run Types/Times/Sections" on page 25).

MarvelClient and Your Infrastructure

Client management and control with MarvelClient is exercised by creating so called Actions in the server-side MarvelClient Configuration database. For example, an Action can add, change or remove one or multiple *notes.ini* variables, add and/or position a database icon, create a replica and configure it on the replicator page, add or modify a workspace page, or even apply mass changes (such as changing all links across desktops, bookmarks and replicator pages from one server to another) – to name just a few examples.

On the client-side, MarvelClient synchronizes new and changed Actions with a local file in the so called MarvelClient Working Directory. At certain Run Times (usually during the client startup and shutdown; see: "Run Types/Times/Sections" on page 25) MarvelClient compares this information with a view in the Configuration database and downloads new and changed Actions **incrementally**. Hence, if there are no new or changed Actions, client-server communication is reduced to the absolute minimum – usually less then one (1) kilobyte of traffic. New or changed Actions are between 2 and 5 kilobytes in uncompressed size – thereby even 50 or 100 Actions lead to only very little traffic, which also only occurs just once due to the incremental update architecture.

In addition to downloading new and changed Actions from the MarvelClient Configuration database, information about each end-user's client configuration is also uploaded into the MarvelClient Analyze database, again incrementally: All relevant files (desktop and bookmark information, database details and more) are checked for changes and only transferred if a generated checksum has changed. Furthermore all files are zipped (a raw XML of ~1 MB is compressed into ~25 KB).

Also, MarvelClient does not increase the load on servers on which the MarvelClient databases reside – with the exception of using ONE (huge) Analyze database on ONE server that collects all audit data from the whole enterprise.

Usual Traffic per User per Session:

- 0.5 KB download from the Configuration database
- 25 KB upload (75 KB upon first client start) to the Analyze database

MarvelClient and the User

MarvelClient is 100% transparent to the end-user – including the initial installation (when using the *Mailbox Postopen Script* option or corporate software deployment). The whole management – and in many cases also the initial installation – is done from the Configuration database, your **central control unit**. In general, MarvelClient is entirely invisible for end users, exept for the following cases:

- You choose for a **Mailbased Install** where the user needs to open the MarvelClient Install Mail (instead of a *Mailbox Postopen Script*, where the user just needs to open his or her mailbox – see "Client Setup" on page 15)
- You define Actions that access the server before the end-user has effectively logged into the Notes client (this would display a small password dialog during the client startup prior to the usual login dialog)
- You have created an Action that needs input from the end-user
- You display optional progress information during potentially lengthy Actions
- You have licensed MarvelClient Zip/Unzip or Attachment Blocking

SETUP

Note: The panagenda MarvelClient Quick Start Guide provides you with step by step instructions for the most important setup details

Installation Types

Local Installation:

Only recommended for testing purposes, since you can then only manage the one local user.

- Use the two database templates (refer to: "Server Installation" on page 9) to create a local Configuration and Analyze database

Dedicated Server:

Whilst we don't recommend to just one dedicated server because of the lack of load-balancing, you can "hard-code" MarvelClient tu use one particular server as opposed to each user's mailserver:

All users and/or test servers need to have a good connection to this server. The server also needs to be able to process the maximum number of concurrent sessions at peak times – this number depends on the number of users that potentially log on during peak working hours and the session duration on the IBM Domino server.

- Use the two database templates (refer to: "Server Installation" on page 9) to create a Configuration and an Analyze database on the appropriate server. Ensure that the server has the role [Admin] in the Analyze database and the role [Server] in the Configuration database (for further details on roles please refer to: "Role descriptions" on page 11)

Load-balanced/Production:

The Configuration and the Analyze database are replicated onto every mail server, where each Analyze database contains all data from all users across all mail servers.

The advantage of this "non-hardcoded" installation type is, that any change at the server level (such as moving an end-user from one mailserver to another, or adding new servers) doesn't affect MarvelClient: In *notes.ini*, the corresponding (MC_DB=) entry refers to

%notes_homeserver%!!YOURPATH\YOURCONFIGDATABASENAME.nsf (for more details see "References" on page 32), which means, that there is no hard-coded servername.

Note: %notes_homeserver% is dynamically resolved from the users then current Location Document.

Also note that you must make sure that de Audit All Agent in the Analyze database (refer to: "(Audit All) Agent" on page 13) only runs on ONE server, which then replicates audited documents back to all other server (if each mailserver were to run the Audit All Agent, this would generate a large number of replication conflicts).

- Use the two database templates (refer to: "Server Installation" on page 9) to create both a Configuration and Analyze database on one server – then, create replicas of both databases on all other mail servers. For the Analyze database, ensure that all serves have the role [Admin], for the Configuration database ensure that all servers have the role [Server] (for further details on roles please refer to "Role descriptions" on page 11).

Note: You should only schedule the Audit All Agent on ONE server!

Hub/Spoke:

Again, the Configuration and Analyze database are replicated onto every mail server, but in oder to reduce the size of the Analyze database on each mail server to just the documents of each mailserver's respective user population, you choose for a "Hub and Spoke" setup: In this setup scenario, all Analyze databases are replicated together into one large Analyze database on a hub. In the Analyze database, the hubserver is granted the role [Admin], whereas the mail serves are *not*. Thereby the Analyze database on mail servers only store the analyze data of "their own users" and replicate it to the hub, where all data is gathered and audited (for further details on roles please refer to: "(Audit All) Agent" on page 13).

Note: You should only schedule the Audit All Agent only on the hub.

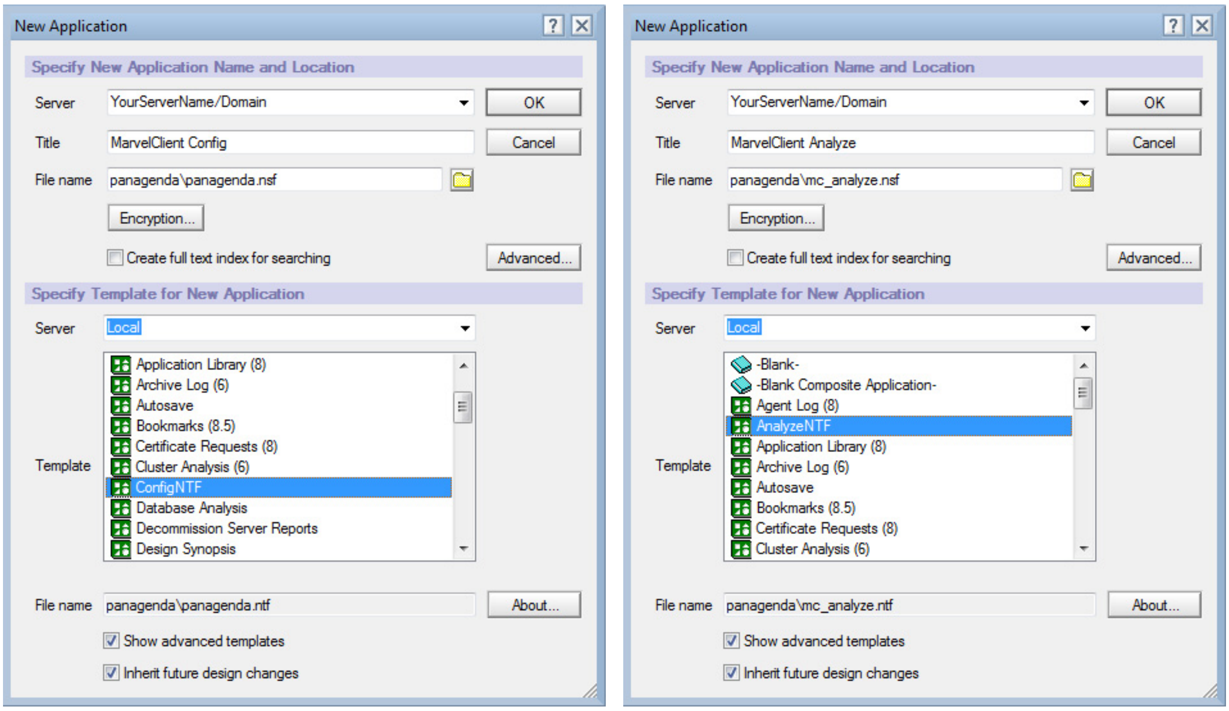
- Use the two database templates (refer to: "Server Installation" on page 9) to create both a Configuration and Analyze database on one server – then, create replicas of both databases on all other mailservers. For the Analyze database, ensure that the hubserver has the role [Admin] and that all other servers do *not*. For the Configuration database, ensure that all servers have the role [Server] (refer to "Role descriptions" on page 11).

Server Installation

Download the templates for both databases (Configuration and Analyze) into your IBM Notes data directory:

- www.panagenda.com/mclic/templates/panagenda.zip
- www.panagenda.com/mclic/templates/mc_analyze.zip

Use these templates to create the databases on a server of your choosing:



New Application

Specify New Application Name and Location

Server: YourServerName/Domain
Title: MarvelClient Config
File name: panagenda\panagenda.nsf
□ Create full text index for searching

Specify Template for New Application

Server: Local
Template: Application Library (8), Archive Log (6), Autosave, Bookmarks (8.5), Certificate Requests (8), Cluster Analysis (6), ConfigNTF, Database Analysis, Decommission Server Reports, Design Synopsis
File name: panagenda\panagenda.ntf
☑ Show advanced templates
☑ Inherit future design changes

New Application

Specify New Application Name and Location

Server: YourServerName/Domain
Title: MarvelClient Analyze
File name: panagenda\mc_analyze.nsf
□ Create full text index for searching

Specify Template for New Application

Server: Local
Template: -Blank-, -Blank Composite Application-, Agent Log (8), AnalyzeNTF, Application Library (8), Archive Log (6), Autosave, Bookmarks (8.5), Certificate Requests (8), Cluster Analysis (6)
File name: panagenda\mc_analyze.ntf
☑ Show advanced templates
☑ Inherit future design changes

Figure 2: Creation of a new Configuration and Analyze Database

While the destination filepath and filename are recommended to be “panagenda\panagenda.nsf” and “panagenda\mc_analyze.nsf“, both the path and filename are freely configurable for both databases. It is recommended to use filepaths and filenames that can be remembered easily, as they may need to be entered in several configuration documents later again.

ACL Settings

Please adjust the access control lists (ACLs) of the two databases as follows:

Configuration Database (panagenda.nsf)		
ACL-Entry	Recommended Rights	Role(s)
-Default-	**No access**	(none)
Normal Users	**Reader**, no other attributes (no "Write public documents", no "Replicate or copy documents",…)	(none)
Rollback Users (= user, for whom the rollback of tiles and bookmarks from the Analyze database is possible if necessary)	**Author**, no other attributes (no "Create documents", no "Delete documents", no "Write public documents", no "Replicate or copy documents",…)	(none)
Administrators	minimum **Editor**	[Admin]
Power Administrators	minimum **Editor**	[Admin], [AOnlineUpdate]
Servers	**Manager**	[Server]
Administration Server	**Manager** + "modify all name fields"	[Server]

Table 1: Access Control List Configuration Database – Settings

Analyze Database (mc_analyze.nsf)		
ACL-Entry	Recommended Rights	Role(s)
-Default-	**No access**	(none)
Normal Users	**Author** with "Create documents" (for the optional collection and updating of Location and Connection Documents also "Delete documents")	(none)
Administrators	minimum **Editor**	[Admin]
Servers (all or one only)	**Manager**	[Admin] (for details whether all of just one server should have this role, refer to "Installation Types" on page 6)
Administration Server	**Manager** + "modify all name fields"	[Admin]

Table 2: Access Control List Analyze Database – Settings

Role descriptions

The following tables describe the roles and their impact in the respective database:

Configuration Database	
Role	Description
[Admin]	Targeting Actions at certain users, groups or certifiers is based on Reader Fields. This ensures that MarvelClient only "sees" exactly those Actions that have been associated to the corresponding end-user. Users with the role [Admin] can see all documents in the Configuration database and are also able to edit them, target them at other users and save them. This is also why users with the role [Admin] are by default affected by all Actions – however, [Admin] users can naturally be excluded from Actions (see "Who Tab in Detail" on page 107).
[AOnlineUpdate]	The role [AOnlineUpdate] is required to carry out an Online Update in the Configuration database. Using online updates, new versions (DLLs and/or templates for example) and licenses can be downloaded and then seamlessly rolled out to clients.
[Rollback]	Users with the role [Rollback] are able to see the option *Advanced* → *Rollback/Restore* in the Navigator in order to verify which restore jobs are still outstanding or already completed.
[Security]	Users with this role can use the Action *ID-File* (*Advanced*). Note that this Action also requires a special license which can be obtained through panagenda support if you already have MarvelClient Migrate or Managed licensed.
[Server]	Servers with the role [Server] can see all documents in the Configuration database (similar to role [Admin]).
[ShowFolders]	Users with this role see the *Self-created Folders* section in the Navigator
[Staging]	The role [Staging] enables the view *Advanced* → *Staging*. This allows the support of mass migration projects with many hundreds to thousands of server-side changes, without having to create as many single (Mass Changes & RTC) Actions manually.
[XML]	All Actions generate XML code for MarvelClient to execute. This XML code is not visible in all tabs unless the role [XML] is granted in the ACL of the MarvelClient Configuration database. Note that he *Admin* tab always displays the XML code.

Table 3: Access Control List Configuration Database – Role Description

Analyze Database	
Role	Description
[Admin]	Documents uploaded into the Analyze database (or rather into any Domino database when using a Backup Action with a Domino database as target) are populated with the following reader names: (Notes) user name of the end-user, (Domino) server name on which the Analyze database resides, cluster members for that server (if so known to the uploading client), and role [Admin]. Thus, end-users can see only see their own documents in the Analyze database; users and servers with role [Admin] can see all documents.
[LessViews]	With with this role, users will only see a very small subset of views in the Analyze database as to prevent view indices from being built in the first place.
[Rollback]	Users with the role [Rollback] can start a Rollback Action from the User Profile Documents in the Analyze database in order to restore a particular backup of the workspace and/or bookmarks. The role triggers the display of a rollback image on the files tab. By clicking on the rollback image, it is possible to then select *desktop.xml* + *images.xml* for the recovery of a workspace and/or *bookmark.xml* for the recovery of bookmarks from a corresponding backup version in the subjacent table. The corresponding destination user, for whom the recovery is requested, will need to have author rights (no other attributes, i.e. "create documents" or "delete documents") in the Configuration database.
[Security]	Activates the menu option *Advanced* → *ID-File Export config.* in the navigation bar. This is to decrypt optionally collected ID file copies. Note that collecting ID Files requires a special license in the first place. Please contact panagenda support in case you need further details (**support@panagenda.com**).
[ShowFolders]	With this role the section *Self-created Folders* will be visible.

Table 4: Access Control List Analyze Database – Role Description

Final Steps

After setting up your databases, please open the Configuration database. You will be prompted to perform an Online Update. Click on the appropriate entry (*Administration → Online Update*) in the Navigator to open the panagenda MarvelClient Online Update Configuration. This form provides several options to perform an online update – if you need further information about these settings, please refer to "panagenda MarvelClient Online Update Configuration" on page 95.

Database Signing

Please ensure that your MarvelClient databases are properly signed before initial installation and after any future Online Updates. To do so, open the Domino Administrator, navigate to the location of your MarvelClient databases, right click on both databases and select "Sign".

Select your desired signing ID (user or server) and sign "All design documents".

Ideal signing IDs are IDs which have the right to run Agents on the server.

(Audit All) Agent

In order (significantly) reduce network traffic and the number of NRPC calls (IBM Notes clients and Domino servers communicate with each other through so called Notes Remote Procedure Calls) between MarvelClient and the Analyze database, information is not directly uploaded into fields within documents, but through zipped XML file attachments. These XML files need to be unzipped on the server-side to fill in all necessary fields for proper visualization of client configuration information in documents and views. The (Audit All) agent can do this automatically for you on a scheduled basis:

Enable the (Audit All) Agent in ONE appropriate Analyze database replica (the appropriate ONE depends on your environment – see: "Installation Types" on page 6).

Open the Analyze database in Domino Designer, navigate to "(Audit All)" (*Code* → *Agents*) and double click on it. Click on the "Schedule" button (*Properties* tab: *Basics*) to open the Agent's schedule dialog and select the appropriate server on which the (Audit All) agent shall run on a scheduled basis. Lastly you may want to adjust the agent schedule. It is recommended to not run the audit agent during peak working hours, since at that moment many clients update their Analyze database documents.

(Cleanup) Agent

The (Cleanup) Agent should be scheduled similar to the (Audit All) Agent); for details see "Cleanup Configuration:" on page 68.

Client Setup

Manual (Local)

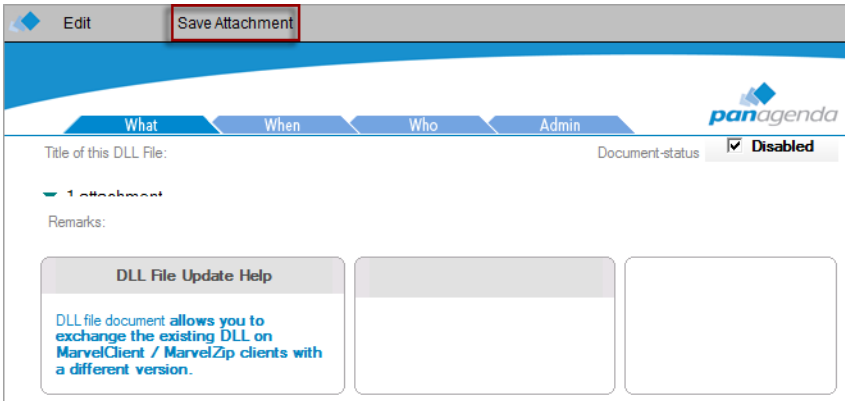
- In the Configuration database, click on *Administration* → *DLL Updates* and open the document for the respective operating system (newest at the top).

- **Save** the attachment as ***mc.dll/libmc.dylib/libmc.so*** into EITHER your IBM Notes program OR your IBM Notes d data ata directory. With multi user clients on Windows, the DLLs for new installations can also be put in the common data directory:

  - **Windows 7**: *C:\ProgramData\Lotus\Notes\Data\Common*

  - **Windows XP**: *C:\Documents and Settings\All Users\Application Data\Lotus\Notes\Data\Common* (the Common data directory is a directory *"Common" on the same level* as "Shared")

> ***Note that under Citrix the program directory is highly recommended, however not required.***



Title of this DLL File:

Document-status ☑ Disabled

1 attachment

Remarks:

DLL File Update Help

DLL file document allows you to exchange the existing DLL on MarvelClient / MarvelZip clients with a different version.

Figure 3: DLL Update document – *Save Attachment*

- In *notes.ini*, please search for an entry "EXTMGR_ADDINS=":

  - If such an **entry already exists**, please add "mc.dll", "libmc.so" or "libmc.dylib" accordingly – with comma but no space separation – for example: *EXTMGR_ADDINS=EntryOne,EntryTwo,mc.dll*

- If such an **entry does not exist** yet, add a respective new entry/line to your *notes.ini*: ***EXTMGR_ADDINS=mc.dll***

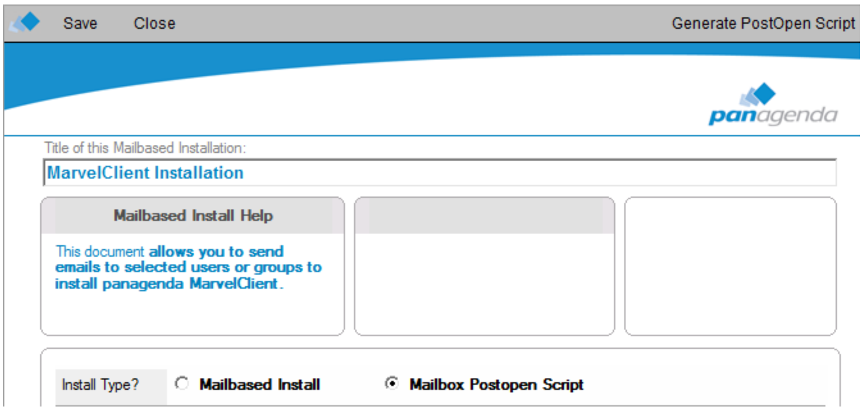
> ***Note that there must not be more than one EXTMGR_ADDINS line in notes.ini, as only the very first such line is relevant, any others are ignored by IBM Notes.***
>
> ***Also note that when adding a new line to notes.ini, the very last line of notes.ini an empty line (this has nothing to do with MarvelClient).***

- Please add another new line to *notes.ini* as follows: ***MC_DB=****NameOfYour(Test)Server***!!***DirectoryAnd\FilenameOfYourConfigurationDatabase****.nsf*. Server and file path are separated with a **double exclamation mark (!!)**. Server and file name must match the server and filename from the previous server installation. Please note, that MC_DB supports the placeholder *%notes_homeserver%* which is dynamically replaced at runtime with the then current end-users mailserver, for example: *MC_DB=%notes_homeserver%!!panagenda\panagenda.nsf*

> ***When the Notes client gets launched, MarvelClient follows the below execution sequence for resolving server names:***
>
> 1. ***MC_DB=*** *gets read and resolved*
>   - *If* ***MC_DB=%notes_homeserver%!!Directory\ConfigDB.nsf****, MarvelClient looks to the information in the Location Document or the MailServer= entry in the notes.ini on how to resolve that*
> 2. *If the above also cannot be resolved, MarvelClient looks for* ***MC_DB_LastReachable=****, which is auto-populated based on last available server connectivity*
> 3. *If the above cannot be resolved, MarvelClient looks for an entry* ***MC_DB_NotResolvable=*** *which can optionally be deployed to users to point to their home server*
> 4. *If the above cannot be resolved, MarvelClient looks for an entry* ***MC_DB_Unavailable=*** *which can optionally be deployed to point to a fallback server in case the users' homeserver is not available*

Deploying MarvelClient on Clients from the Configuration DB

The Configuration database allows for easy deployment of the local MarvelClient (*.dll*/*.dylib*/*.so*) file, including the necessary changes to *notes.ini*, on selected or all end-user clients. All you need to do is complete a form for creating an installation email or a *Mailbox Postopen Script* (=automatically installation of MarvelClient when end-users open their mail databases). Navigate to *Administration* → *Installation* and click on the *New Install* Button in the Action Bar of that view to open the respective form:



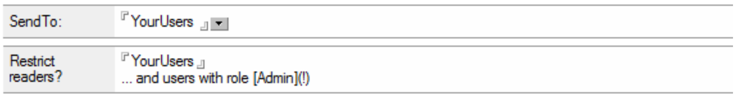
Title of this Mailbased Installation: MarvelClient Installation

This document **allows you to send emails to selected users or groups to install panagenda MarvelClient**.

Install Type? ○ Mailbased Install ● Mailbox Postopen Script

Figure 4: Mailbased Installation Form – Section 1

Start with naming your document and choose your preferred *Install Type*. If you select *Mailbox Postopen Script*, there won't be the option to define an email subject (several other email related options won't be available either).

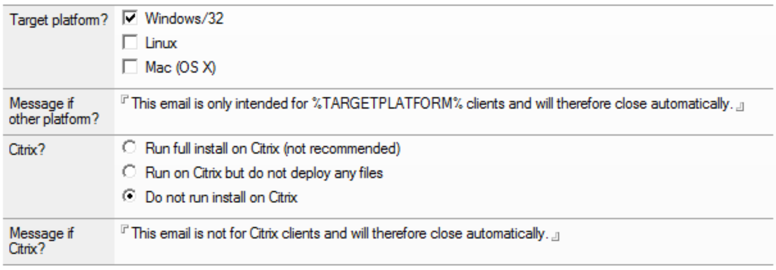


SendTo: YourUsers

Restrict readers? YourUsers
... and users with role [Admin](!)

Figure 5: Mailbased Installation Form – Section 2

Next specify to which users/groups you want to roll out MarvelClient. Depending on *Install Type*, you either have to enter the users/groups who should be sent the installation email (*SendTo*) or those to whom the *Mailbox Postopen Script* is restricted; an empty *Restrict readers?* field means that MarvelClient is installed for all users, that open a database to which the PostOpen Script code is added.

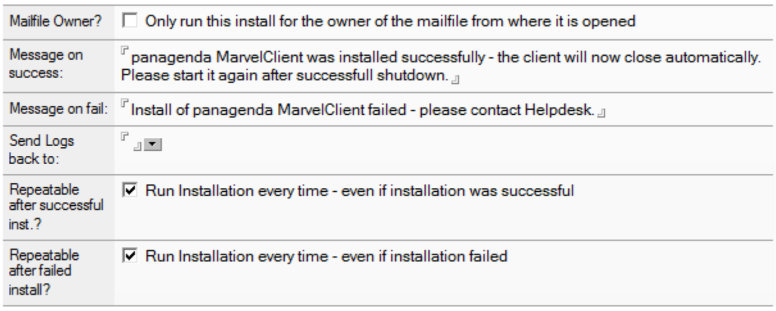


Target platform?	☑ Windows/32 □ Linux □ Mac (OS X)
Message if other platform?	This email is only intended for %TARGETPLATFORM% clients and will therefore close automatically.
Citrix?	○ Run full install on Citrix (not recommended) ○ Run on Citrix but do not deploy any files ● Do not run install on Citrix
Message if Citrix?	This email is not for Citrix clients and will therefore close automatically.

Figure 6: Mailbased Installation Form – Section 3

Select a *Target platform* and optionally edit the *Message if other platform* field (*Mailbased Install* only). Leaving the Message if other platform field empty will not display a message box to end users in case the platform the mail is opened on does not match any of the target platforms.

Furthermore specify the options for Citrix clients. Again you can edit the pre-defined *Message if Citrix* (only *Mailbased Install*). Leaving the Message if Citrix field empty will not display a messagebox to end users in case the mail is opened from within a Citrix session.



Mailfile Owner?	□ Only run this install for the owner of the mailfile from where it is opened
Message on success:	panagenda MarvelClient was installed successfully - the client will now close automatically. Please start it again after successfull shutdown.
Message on fail:	Install of panagenda MarvelClient failed - please contact Helpdesk.
Send Logs back to:	
Repeatable after successful inst.?	☑ Run Installation every time - even if installation was successful
Repeatable after failed install?	☑ Run Installation every time - even if installation failed

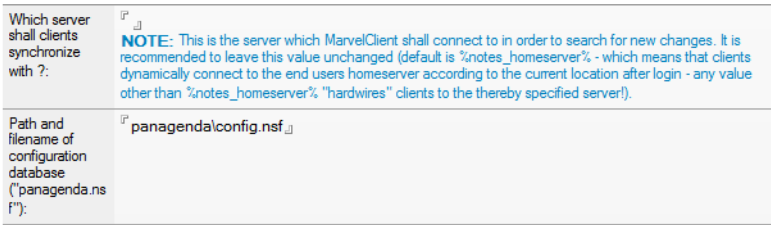
Figure 7: Mailbased Installation Form – Section 4

If *Mailbased Install* was selected as the preferred *Install Type*, you can select whether you want to *Only run this install for the owner of the mailfile from where it is opened.*

In the next two lines, you can optionally edit the *Message on success* and *Message on fail*.Leaving any of these two fields blank will not display a message box to end-users in case the install fails/succeeds.

Furtheron please specify whom install logs shall be sent back to – it is recommended to enter an email address here, like your own or a central administrator mailbox.

Last but not least you can select whether the installation should be repeated if installation was successful/failed (*Mailbased Install* only).

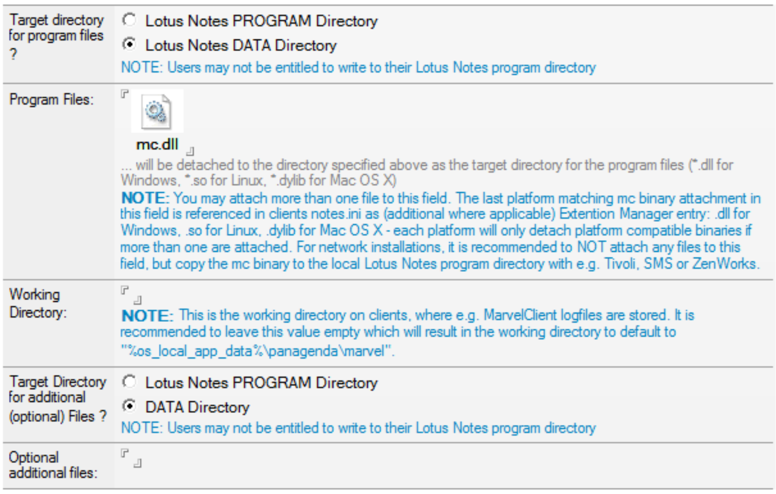


Which server shall clients synchronize with ?:	⌜ ⌟ **NOTE**: This is the server which MarvelClient shall connect to in order to search for new changes. It is recommended to leave this value unchanged (default is %notes_homeserver% - which means that clients dynamically connect to the end users homeserver according to the current location after login - any value other than %notes_homeserver% "hardwires" clients to the thereby specified server!).
Path and filename of configuration database ("panagenda.nsf"):	⌜panagenda\config.nsf⌟

Figure 8: Mailbased Installation Form – Section 5

If you chose for a dedicated server Install Type (refer to "Installation Types" on page 6) enter the corresponding dedicated server name in the field *Which server shall clients synchronize with?* – otherwise, leave the server as is (*%notes_homeserver%*). Please pay attention to the Note about the *%notes_homeserver%* in the form (and screenshot above). Also keep "Installation Types" on page 6 in mind.

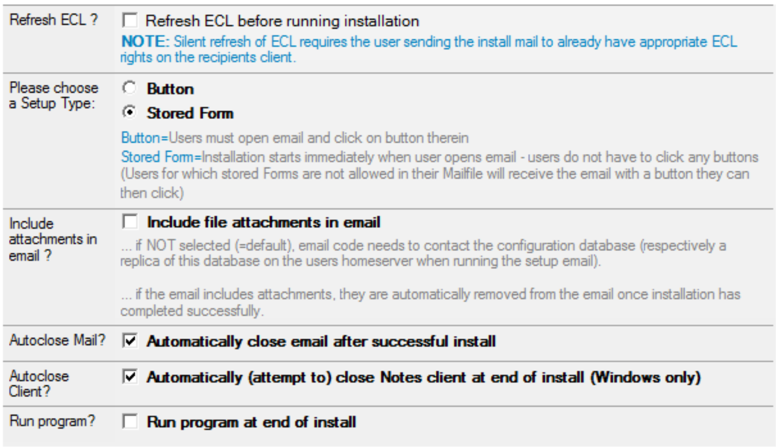
Next, specify the path and filename of your Configuration database (recommended: panagenda/panagenda.nsf – as described under "Server Installation" on page 9).



Target directory for program files ?	○ Lotus Notes PROGRAM Directory ● Lotus Notes DATA Directory NOTE: Users may not be entitled to write to their Lotus Notes program directory
Program Files:	⌜ mc.dll ⌟ ... will be detached to the directory specified above as the target directory for the program files (*.dll for Windows, *.so for Linux, *.dylib for Mac OS X) **NOTE**: You may attach more than one file to this field. The last platform matching mc binary attachment in this field is referenced in clients notes.ini as (additional where applicable) Extention Manager entry: .dll for Windows, .so for Linux, .dylib for Mac OS X - each platform will only detach platform compatible binaries if more than one are attached. For network installations, it is recommended to NOT attach any files to this field, but copy the mc binary to the local Lotus Notes program directory with e.g. Tivoli, SMS or ZenWorks.
Working Directory:	⌜ ⌟ **NOTE**: This is the working directory on clients, where e.g. MarvelClient logfiles are stored. It is recommended to leave this value empty which will result in the working directory to default to "%os_local_app_data%\panagenda\marvel".
Target Directory for additional (optional) Files ?	○ Lotus Notes PROGRAM Directory ● DATA Directory NOTE: Users may not be entitled to write to their Lotus Notes program directory
Optional additional files:	⌜ ⌟

Figure 9: *Mailbased Installation Form* – Section 5

Then choose to which directory you want to roll out your (the) MarvelClient (*.dll*/*.dylib*/*.so*). After that, attach the appropriate files to the *Program Files* field (for instance by opening the *Administration* → *DLL Updates* view, copying the file from the document of your choosing and pasting it into the *Program Files* flied). Please read all *Notes* under *Program Files* carefully.



Refresh ECL ?	☐ Refresh ECL before running installation **NOTE:** Silent refresh of ECL requires the user sending the install mail to already have appropriate ECL rights on the recipients client.
Please choose a Setup Type:	○ **Button** ● **Stored Form** Button=Users must open email and click on button therein Stored Form=Installation starts immediately when user opens email - users do not have to click any buttons (Users for which stored Forms are not allowed in their Mailfile will receive the email with a button they can then click)
Include attachments in email ?	☐ **Include file attachments in email** ... if NOT selected (=default), email code needs to contact the configuration database (respectively a replica of this database on the users homeserver when running the setup email). ... if the email includes attachments, they are automatically removed from the email once installation has completed successfully.
Autoclose Mail?	☑ **Automatically close email after successful install**
Autoclose Client?	☑ **Automatically (attempt to) close Notes client at end of install (Windows only)**
Run program?	☐ **Run program at end of install**

Figure 10: *Mailbased Installation Form* – Section 6

In the above screenshot, if checked, the first option will **refresh** the client **ECL** according to the Admin ECL on the users mailserver during installation. Note that this will display an ECL alert if the sending user does *not* have the rights to actually refresh/change the target client ECL.

For installs of type *Mailbased Install*, the next option *Setup Type* allows to choose for whether install emails are to be sent as stored form or as normal email including a button.

Note that when sending the mail as Stored Form, the signer ID that was used to sign the (M. Mailbased Install Template) form (not the M. Mailbased Install form) in the Configuration database is relevant for any possible ECL alerts on targeted clients – whereas for sending emails with a Button it is the sender of the Installation Mail.

Sending the Installation Mail as *Stored Form* is usually better, as the install occurs immediately after opening the email (even from the preview pane). Sending an Install Button decreases the initial success rate (since users must open the email And then click on the button), as well as increases tracking and tracking overhead.

Note that mail recipients who have Stored Forms disabled will see the button instead; therefore using the stored form Setup Type option is almost the best way to go.

Including file attachments in the email being sent ensures that the installation also works offline, such as when laptop users take the install mail with them when travelling, but do not open it before being offline. The file attachments are automatically removed after successful installation, provided that the installation is *not* sent to be repeatable after being successful.

Finally, select whether the email or the IBM Notes client shall be closed automatically after (successful) installation to aid users with the restart of their client.

The option to *Run a program at the end of install* allows to execute an executable or batch file if needed – this is optional and not needed for a normal MarvelClient install.

There are also two rich text fields: *(optional) Richtext before/after Button*. Those are for when sending installation emails with a button (or in case a user has stored forms disabled in his or her mailfile): if you want give the install mail a nicer look, add a company logo, explain the purpose of the install email – these two fields are there to customize the look and feel of the install mail.

Note that the notes.ini entries table at the bottom of the form is for any optional notes.ini variables that you may wish to role out together with MarvelClient. The necessary notes.ini entries EXTMGR_ADDINS, MC_DB and MC_WorkingDirectory are implicitly taken care of by the install code (be it Mail- or PostOpen Script-based) in any case – you do not need to configure these in the form's bottom-most table.

When you finished completing the form:

- The **Installation Email** is sent upon explicit confirmation when saving the *Mailbased Installation* document. This also means that a readily filled in installation document can be reused as often as needed: simply enter other/additional recipients, save, send and close the document again.

- If you want to embed PostOpen Script into end user mailfiles or a different database (a telephone book application everyone uses regularly, for example), then click on ***Generate PostOpen Script*** in the Action Bar of the install configuration document. Please add the resulting LotusScript code to the design of the mail database or template as per the instructions shown on the then appearing popup (Note that the code is only computed and displayed upon properly filling in all relevant fields).
- (Only appears after successful generating process). Subsequently, add the requested call to the "MC_Install" code to an appropriate place of the mailbox (or other application) design.

Deployment with a Third Party Software Push Tool

When using a software distribution solution to install MarvelClient, familiarize yourself with the previously described manual (local) installation (refer to: "Manual (Local)" on page 15):

The deployment with software distribution tools is very easy as MarvelClient only requires to drop one file into either the IBM Notes data or program directory and add a number of *notes.ini* entries. MarvelClient does not require any changes to the IBM Notes shortcuts (Windows desktop shortcuts), nor does it come with Windows startup components or any other complicated to bring out, non-standard elements.

Is MarvelClient installed properly?

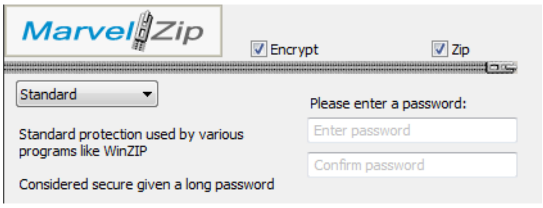
Leaving the IBM Notes end-user experience unchanged is vital to ensure broad support across the organization and especially in upper management when using a client management solution. This is why it's not easy to at first sight spot whether MarvelClient is properly installed. The following points, however, each allow for an easy check as to whether MarvelClient is up and running:

Local *log.xml*:

Amongst other files, a *log.xml* file is maintained by MarvelClient in the end-users' MarvelClient Working Directory (for example on Windows 7:

C:\Users\UserName\AppData\Local\panagenda\marvel see "The panagenda Working Directory" on page 44 for further details). If the file exists and was updated upon launching the client (check the last modified date/time of the file), then MarvelClient is properly installed.

MarvelClient Zip/Unzip:



☑ Encrypt ☑ Zip

Standard

Standard protection used by various programs like WinZIP

Considered secure given a long password

Please enter a password:

Enter password

Confirm password

Provided that the MarvelClient Zip/Unzip Module is licensed (trial versions usually do include it), the Zip/Unzip setting section is visible in the file-attach dialog when attaching a file to (for example) an email:

On IBM Notes Basic Clients:

If the Notes Basic client is closed with its window maximized and then launched again (with MarvelClient working), the main window will not be maximized during the client start up, but when the IBM Notes client has finished launching the user interface – without MarvelClient, the IBM Notes window is maximized from the very start.

On IBM Notes Standard Clients:

Without MarvelClient, the IBM Notes client will display the progress bar in the splash screen the very moment that it displays the login dialog. With MarvelClient, the IBM Notes client will not display the progress bar before the login dialog, but after the user has entered his or her password.

BASICS

The Configuration and Analyze database are very different: Whilst the Configuration database is responsible for the active part, for example the actual IBM Notes client management, including the configuration of when and what client data is actually collected in the Analyze database, the Analyze database itself can be seen as entirely passive: it provides smart inspection possibilities for all information collected by MarvelClient on IBM Notes clients to anyone who wants to analyze or audit the client infrastructure.

In principle, these two databases are entirely independent from one another (with the exception that the Configuration database needs to tell MarvelClient when and what to upload into the Analyze database). To configure MarvelClient, it does not matter whether you have no or outdated data in your Analyze database, as clients are always configured as to what they should look like and not "how they should look different from what was last analyzed". This allows for both a 1:some/many AND 1:any approach in client management, as opposed to just 1:1 and 1:some. The 1:any client management approach, especially in combination with client attributes such as "do this on laptops, that on desktops, and something else on Citrix/virtual clients", is the biggest cost saver when it comes to client management. Instead of selecting specific users and groups (perhaps even after having to spend the time to identify them in an analysis database) and change them, you simply specify what certain client(type)s should look like "365 days a year, 24 hours a day" – and MarvelClient simply takes care of it, independent of your Analyze database.

TIP

You should take a look at your Analyze database once in a while, as it clearly points out how far you are away from where you would like (or need) to be. The Analyze database is also the perfect place to find out how you can further optimize your existing client configurations – without thinking in terms of "these are the users I need to change", but rather in terms of "every laptop out there should have a local replica" or "no client out there should have outdated ODSs" and similar.

When it comes to the Configuration database, **Actions** are the very foundation of both the Configuration database itself and MarvelClient in general.

Depending on the scope of license, the Configuration database allows you to instruct MarvelClient on your IBM Notes clients to carry out various tasks, such as housekeeping, optimization, configuration, initial setup, upgrades, audits, and much more, virtually covering all aspects of your client management demand. Since not all customers require all of MarvelClients features from the very start, MarvelClient is available in various different modules, each enabling different (sets of) Actions. The further description of Actions starts with the chapter: "Actions: General" on page 103. Before we go there, however, it is important be familiar with the following topics:

Run Types/Times/Sections

There are several different "moments" which are very important for the ongoing use of MarvelClient: First of all, it is important to distinguish between Run Times and Run Sections:

- **Run Times** are for instance the client start ("init"), the period before or after login, the IBM Notes session itself ("background") and the closing of an IBM Notes client ("shutdown") – refer to "Figure 11: MarvelClient Run Times" on page 26
- **Run Sections** are "parts" of Run Times – examples are: "before download", "before sync" and "after write", all of which belong to the Run Time "After Login".

Each Run Time has 1-n Run Sections. Note that most Run Times only have one Run Section (for example "Shutdown"). The combination of Run Time and Run Section is called Run Type, such as "After Login – Before Download", "After Login – Before Sync".

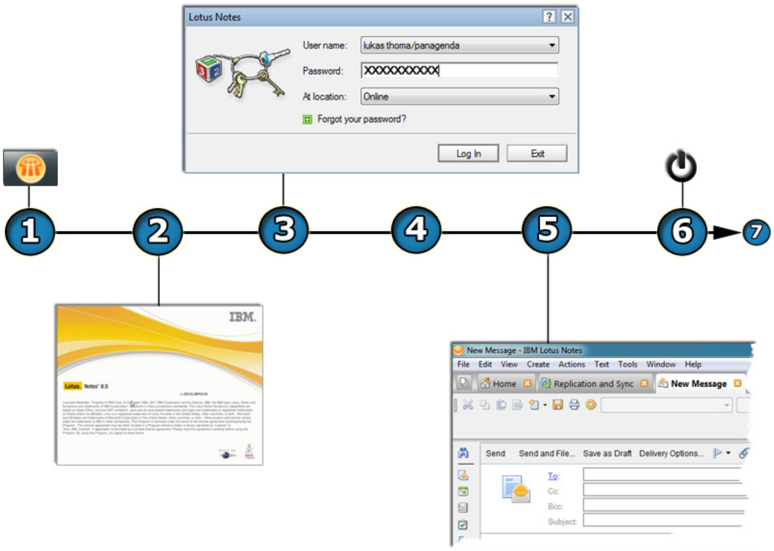
Every Action from within the Configuration database either has one supported Run Type, or allows you to choose from multiple possible Run Types.

For example, managing a desktop icon is only allowed during "After Login" in the exact same Run Section, but neither "After Login – Before Download", "Before Login" or during "Shutdown". When running your own Agent with MarvelClient, however, you may choose between various different Run Types. In most cases it is recommended to stick with the Run Type the Configuration database suggests by default – unless you really know what you are doing or have been instructed otherwise by panagenda support.

Note that Run Times are also of vital importance for the Analyze database, to precisely localize performance problems (also see page 55).

The following figures illustrate major Run Times and Run Sections:

Progress in detail:



Lotus Notes

User name: lukas thoma/panagenda

Password: XXXXXXXXXX

At location: Online

Figure 11: MarvelClient Run Times

1. The IBM Notes client is launched and the local MarvelClient file (for example *mc.dll* on Windows) is invoked by Notes as an IBM Extension Manager (EXTMGR_ADDINS parameter in *notes.ini*)

2. The Run Time Before login is comprised of the following Run Sections:

   - *Before Login – Init*(ialization)
   - (Analysis only): Time it takes the IBM Notes client to move from *Before Login – Init* to *Before Login – Pre*
   - *Before Login – Pre* – usually used only for MarvelClient Roaming

- (Analysis only): Time it takes the IBM Notes client to move from *Before Login – pre* to *Before Login*
- *Before Login* – Any Actions set to the Run Type *Before Login* are executed (such as preparing Location Documents to be displayed in the login dialog)

3. (Analysis only): Time it took from "before login" to "after Login" = either Password entry or **Single Sign on** (with SSO, this should take <100 ms; without SSO it takes as long as it takes the user to enter his or her password)

4. After login: (this is the main Run Time for MarvelClient Actions – all bullet points are equivalent to Run Types)

   - **After login - Before download** – Allows to execute Actions before local Actions (*actions.xml*) are synchronized with the Configuration database (new/changed/deleted Actions)
   - **After login - Before Sync** – Allows for the execution of Actions that need to access databases before MarvelClient actually opens them (desktop, bookmarks, etc.)
   - **After login - Before Roaming** – Executes Actions before the desktop and bookmark configuration (XML files) is actually roamed into the respective databases (bookmark.nsf and desktop*.dsk/ndk)
   - **After login** – This is where most Actions are executed – run times can be anything from 1 second to several seconds. Typically, all of After Login should be executed within 4-7 seconds – longer run times indicate performance problems
   - **After Login - After Write** – Allows to execute Actions after all changes (if applicable) have been made during "after login". This Run Type ensures that all databases needed by MC are already closed
   - **Background - Init** – Whilst all above Run Types are "blocking", *Background – Init* runs as a separate thread in parallel to the remaining Notes client startup. Especially used for MarvelClient Eclipse features

5. Client Session:

   - **On first server access after start** – By default, this MarvelClient feature is set to wait for when the client (background replicator etc.) or end-user first accesses his or her mailserver after Notes client startup. It runs in parallel to the client session and waits for the first server access by the end-user/replicator/policy engine or similar and will run any Actions configured for this Run Section **one minute after the end of "after login" at the earliest**

- **Background** – Runs every 60 minutes (by default) in parallel to the client session. Allows to enforce for instance *notes.ini* settings or Eclipse configurations whilst IBM Notes is up and running – also allows to **infuse new Actions after clients have started**, thus *Background* includes updating new/changed/deleted Actions from the Configuration database

6. **During shutdown** – Runs Actions during shutdown (for example backup for roaming)

7. (Client is closed)

On Reference Only – On first sight this particular Run Type seems as it has nothing to do with the above Run Times, as since such an Action will only be executed if referenced by another Action (refer to "Admin in Detail" on page 110). However, the Run Type *On Reference Only* is essential when reading through "Execution Order and Reference Chains" on page 36.

Note that some Run Types are not available on all operating systems supported by MarvelClient. Please check the release notes in the Configuration database to see which Run Types are (not) supported on which platforms (in the Navigator: Information → Release Notes).

Additional *notes.ini* entries

In addition to the mandatory *notes.ini* entry EXTMGR_ADDINS (and MC_DB if you installed the Configuration database with a filepath different from the default panagenda\panagenda.nsf), other entries may be relevant depending on the scope of license and what you want to achieve:

General Settings		
Entry	Values (examples)	Annotations
MC_WorkingDirectory=	**Allowed root directory placeholder:** %notes_data% %notes_client% %os_system_root% %os_program_files% %os_app_data% %os_common_app_data% %os_local_app_data% %os_user_profile% %os_temp% **Allowed sub-directory placeholders:** %notes_user% %os_user% %os_computer%	Working directory for panagenda MarvelClient *On Windows, the working directory defaults (=if not specified in note.ini): %os_local_app_data% \panagenda\marvel* *Example: C:\Users\ [OSUsername]\ AppData\local\ panagenda\ marvel* NOTE: On Mac OS X and Linux, the working directory defaults to */tmp/ panagenda/marvel*
MC_SkinningDirectory=	Refer to **MC_WorkingDirectory=** above	Skinning directory – contains files to skin the IBM Notes workspace (aka MarvelClient Skinning Edition) *Defaults to: %notes_data%\Skinning*
MC_AllowSyncDuringSetup=	**1** = force synchronisation **2** = sync is only executed when *KeyFileName=* is specified in *notes.ini* or ConfigFile (for example *setup.txt*) and when this entry refers to an existing ID file (setting this to 2 is **recommended**).	Required for synchronisation during initial setup of IBM Notes (by default, MarvelClient does not synchronize any changes during initial client setup)
MC_ZapNotes=	**1** (or non-existent)	Activates the automatic detection of hanging processes during Notes client startup.

Table 5: MarvelClient *notes.ini* variables – General Settings

Roaming		
Entry	Values (examples)	Annotations
MC_NetworkDirectory=	**H:\Roaming** or **\\UNCPath\...** **Allowed Placeholders:** %os_user%	Required for file-based roaming
MC_AllowSyncDuringSetup=	**2** (Prerequisite for roaming under Citrix and newly installed clients)	see Table 5 on page 29
MC_PreviousNotesUsername=	(read only)	Last logged in Notes user
MC_LastBackupDate=	date/time (UTC) – (read only)	date/time of the mos recent roaming backup. If this entry in *notes.ini* is deleted, a client will automatically roam the youngest roaming backup from the network directory upon next startup.

Table 6: MarvelClient *notes.ini* variables – Roaming

Realtime Control		
Entry	Values (examples)	Annotations
MC_RR_Enabled=	**1** or **0**	Realtime Control: enabled/disabled
MC_RR_UseTray=	**1** or **0**	Realtime Control: Display key symbol in systemtray – yes/no
MC_RR_Profile_Current=	**nn_rrProfilename** Example: 00_Default	Realtime control: Currently selected/enabled profile (1-n real time rules)
MC_RR_Running=	1 or 0 – (read-only)	Real time control: Active – yes/no?

Table 7: MarvelClient *notes.ini* variables – Realtime Control

Partial Analysis of User Address Books		
Entry	Values (examples)	Annotations
MC_UploadDB=	**Server!!Directory\File.nsf** of the Analyze database	**Required to optionally collect…**
MC_AddonColGetLocations=	1 or 0	…Location Documents
MC_AddonColGetConnections=	1 or 0	…Connection Documents,

Table 8: MarvelClient *notes.ini* variables – Partial Analysis of User Address Books

Partial Analysis of User Address Books

Entry	Values (examples)	Annotations
MC_AddonColGetAccounts=	1 or 0	...Accounts; (not recommended due to number of Accounts as per client)
MC_AddonColGetCertificates=	1 or 0	...Certificates, (not recommended due to number of Certificates as per client)
MC_AddonColKey=	We recommended to set this to <os:user>\<notes:user_abbreviated>\<os:computer>\<notes:version> along with above values	**In the left-hand example, any documents enabled for collection will be uploaded into the Analyze database as per combination of OS username, Notes username, computer name and Notes release.**

Table 8: MarvelClient *notes.ini* variables – Partial Analysis of User Address Books

Software & Hardware Analysis

Entry	Values (examples)	Annotations
MC_UploadDB=	**Server!!Directory\File.nsf** of the Analyze database	**Required to optionally collect...**
Note: Hard- and software analysis automatically contains information on whether end-users have admin rights on the OS level, as well as details on the processor and physical memory		
MC_AddonColGetHWDetails=	1 or 0	...hardware details
MC_ AddonColGetLogicalDriveInfo=	1 or 0	...drive details
MC_AddonColGetProductInfo=	1 or 0	...software (=products) info
MC_AddonColKey=	We recommended to set this to <os:user>\<os:computer> along with above values	**In the left-hand example, any hard- or software information enabled for collection will be uploaded into the Analyze database as per combination of OS username and computer name.**

Table 9: MarvelClient *notes.ini* variables – Software & Hardware Analysis

Temporary variables that are stored in *notes.ini* when using the MarvelClient databases	
Entry	Annotations
MC_DBShowHelp=	Display Help in Forms/Documents in the Configuration database – yes/no
$MC_DB_Language=	Selected language for the Configuration and Analyze database
$MC_DBLastView=	Last view opened in the Configuration database
$MC_UploadDBLastView=	Last view opened in the Analyze database

Table 10: MarvelClient *notes.ini* variables – Tempory Variables

References

Dynamic configurations that automatically adapt to every end-user's IBM Notes client are a basic requirement for flexible and scalable management of IBM Notes/Domino environments.

Two examples:

- Adding the public names and address book to the workspace of every end-user:

  Without a dynamic configuration, one MarvelClient Action would have to be created for each and every mail server. Using a dynamic configuration, it is just one Action that automatically adapts to every end-user's mail server according to the current location of each user.

- Backing up client data to a network drive into a sub-directory "operation system username" – here, too, a variable is needed.

To fulfill such requirements, MarvelClient provides so called References and Placeholders.

References can be used universally in virtually any Action in the Configuration database and are resolved PRIOR to the execution of an Action.

If resolving a reference fails, any Action referencing this Action will not be executed.

Placeholders

Basically, Placeholders are primarily used in *notes.ini* – for example: *%notes_data%*, *%os_user_profile%*, *%notes_user%*, *%os_user%* or *%env_nameofOSenvironmentvariable%.*

There are three Actions in the Configuration database, in which Placeholders are used too: *Backup/Monitoring*, *Roaming/Rollback* and *Location*. In Location Actions, for example, Placeholders are used to automatically resolve information from the public names and address book WHILST the Location Action is being executed:

- *%LOOKUP_MAILDOMAIN%*
- *%LOOKUP_MAILSERVER%*
- *%LOOKUP_MAILFILE%*
- *%LOOKUP_MAILFILE_AUTODETECT%*
- *%LOOKUP_INTERNETMAILADDRESS%*
- *%LOOKUP_SHORTNAME%*

Placeholders are replaced whilst a respective *notes.ini* entry is being read, respectively DURING EXECUTION of the Action.

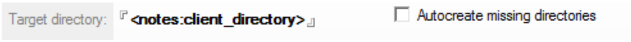
References – Syntax

The use of References is subject to the following syntax:

<Namespace:VariableName>

For most of the available variables you can use a *config.xml* file as a starting point (you can find this file in the MarvelClient Working Directory – refer to "The panagenda Working Directory" on page 44) – typically it is sufficient to look at the *config.xml* of one or two machines to get a hang of which Namespaces and Variable Names can be used. There are usually only differences between major Notes releases (7/8 for example) and platforms (Windows/Mac/Linux).

Example:

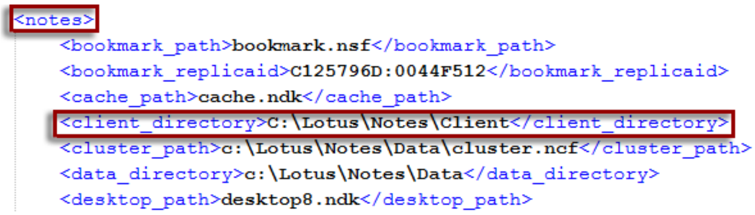


Target directory: <notes:client_directory>

□ Autocreate missing directories

Figure 12: References – Example 1

The above screenshot shows using a Reference to generically configure the *Target directory* field of a Backup/Monitoring Action to point to each user's <*notes:client_directory*>. Looking at a sample MarvelClient *config.xml* (located in the Working Directory – see "The panagenda Working Directory" on page 44) shows that this would be resolved to "C:\Lotus\Notes\Client" on the client from which *config.xml* was opened:



```
<notes>
    <bookmark_path>bookmark.nsf</bookmark_path>
    <bookmark_replicaid>C125796D:0044F512</bookmark_replicaid>
    <cache_path>cache.ndk</cache_path>
    <client_directory>C:\Lotus\Notes\Client</client_directory>
    <cluster_path>c:\Lotus\Notes\Data\cluster.ncf</cluster_path>
    <data_directory>c:\Lotus\Notes\Data</data_directory>
    <desktop_path>desktop8.ndk</desktop_path>
```

Figure 13: References – *config.xml*

Namespaces for MarvelClient References:

- **mc** – MarvelClient "System variables"; see <*mc*> ... <*/mc*> in *config.xml*
- **mz** – "System variables" for MarvelClient Zip/Unzip and Attachment blocking (formerly called MarvelZip); see <*mz*> ... <*/mz*> in the *config.xml*
- **user** – This Namespace is meant for your own variables (thus you can freely write your own variables into this Namespace; for example using an **ini & Variable* Action that sets "replica_type" to "stub")
- **ACTION-ALIAS** – Allows to reference results and variables of MarvelClient Actions; for example <*ACTION-ALIAS:replica_type*> allows to reference a *Desktop Icon & Replica* Actions in oder to query for whether a local replica exists for a given application.

Namespaces for IBM Notes and Operating System References:

MarvelClient also provides over 150 IBM Notes client and operating system details, which can be referenced, allowing for flexible rule sets to be compiled without any programming. The corresponding Namespaces are:

- **notes** – refer to *<notes> ... </notes>* in *config.xml*
- **os** – refer to *<os> ... </os>* in *config.xml*

Namespaces for "external" References:

- **env** – Available variables depend on OS environment variables of each target client (see *cmd.exe*/terminal → "set")
- **ini** – Available variables depend on *notes.ini* of each target client

VariableNames:

The available Variable Names from within the Namespaces ***mc***, ***mz, os*** and ***notes*** can be deduced from a representative *config.xml* (in each client's Working Directory or in the Analyze database if so collected). This xml file is recreated upon every Notes client start – after closing an IBM Notes client, any therein available variables are not read back into memory ever.

About Reference Chains

References can also build up so called Reference Chains.

For example:

- A *Desktop Icon & Replica* Action, which requires the result of a *Workspace Page* action (in order to add/move the icon on to a specific page), whereby the *Workspace Page* Action again references the result of a *Regular Expression* Action for a more complex computation of its page title.

If ANY Reference from within a Reference Chain cannot be resolved, the entire Action Chain is not executed at all.

Any circular Reference Chains (Action A references Action B references Action A) are excluded from execution in the first place.

Reference Chains and especially circular ones, however, are rather seldom, as chaining of Actions and References is rarely required or is handled automatically by the Configuration database itself (for example managing an icon on a specific workspace page is done automatically through References).

Execution Order and Reference Chains

In general, MarvelClient executes Actions in the following order:

1. Run Types are executed one after another, such as *Before Login* → *After Login* → *Shutdown*
2. Within each Run Type, Actions are executed by Priority, Actions with lower priority (weighting) come first

   Actions with same priority are executed by Title (alphabetically)

3. If an Action references another Action, the referenced Action is executed before the Action that references it. In case execution of the referenced Action fails, the referencing Action will not be executed further.

   If Action A, running before login, references Action B, set to run after login, Action A will bring Action B forward to run before login.

Stringing Actions together is rarely needed – if at all. A fairly popular use case however is "I only want to switch a user's Location Document to local if they have a fully replicated local replica". In this case, a Location Action needs to "ask" a Desktop Icon & Replica Action whether it has a full replica or not. For ease of use, the use case can be fulfilled easily with MarvelClient using references:

1. Create a Desktop Icon & Replica Action that manages the user's mailfile – the Action doesn't have to specifically manage a user's replica (it shouldn't remove the replica, though)
2. Give the Desktop Icon & Replica Action an alias such as "UserMailFile"
3. Then create a String Condition that checks for *<alias:replica_type>=full*, which in this particular above example means:
   <UserMailFile:replica_type>
   =
   full
4. In the Location Action that sets "server=local", make it only run based on the above Condition.

It is not a problem if, because of Run Types or execution priority, the Location Action runs BEFORE the "Mailfile" Desktop Icon & Replica Action – as soon as the Location Action checks for the Condition of the full local replica, it will automatically RUN the "Mailfile" Action. Note that in this example, if the Location Action were set to also run before login, it would fail, as bringing the Desktop Icon & Replica Action forward to *Before Login* is not possible (managing the mailfile and asking the user's mailserver for mailfile details is not possible prior to login) – the Desktop Icon & Replica Action would fail and so would the referencing Location Action.

Summarizing order of execution in words means:
Run Types come before References, come before priorities, come before titles by alphabet.

Note: If ANY Reference fails (for instance a referenced Action cannot be brought forward or may not execute due to conditions, or a variable cannot be resolved), or if there are circular References, ALL Actions within the Reference chain are skipped.

MarvelClient Roaming and Rollback

Roaming/Profiling Key

MarvelClient Backup and Roaming/Rollback jobs also make use of Placeholders. Therein used backup (Profiling) and roaming/rollback (Roaming) Keys consist of one or multiple Placeholders (since Backup and Roaming/Rollback Actions need to preserve the unresolved key for further use during execution of the Action, References cannot be used as they would be resolved prior to executing the respective Action). The following Placeholders can be used for Profiling Keys in Backup Actions and Roaming Keys in Roaming/Rollback Actions:

- ***%notes_user%, %notes_user_abbreviated%***
- ***%notes_version%, %notes_version_major%***
- ***%os_user%***
- ***%os_computer%***

By using "\" (backslash) as separator you can combine any number of those Placeholders (several sub-keys), such as:

%os_user%\%notes_version_major%

or

%os_user%\%notes_user_abbreviated%\%os_computer%\%notes_version%

TIP

The latter example represents the default Profiling Key for Backup/ Monitoring Actions.

Note that in the above second example, a backup is created for each combination of OS user, Notes user name, OS computer name and Notes Version – meaning that if a user has a laptop and a desktop, with two Notes Releases on each, MarvelClient will create four backup sets.

You could easily change the Backup Key to not include the Notes release, for example, by simply removing *\%notes_version%* from above example, resulting in the profiling key being *%os_user%\%notes_user_abbreviated%\%os_computer%*.

In a similar fashion, you could reduce the number of physical backup set by changing the Backup/Profiling Key to just *%os_user%\%os_computer%*, resulting in one backup as per combination of OS username and computer name.

Note that if multiple end-users use one and the same client, that in above example one would not be able to distinguish the backup of Notes user A from the backup of Notes user B, as the Notes username is not used as part of the Backup/Profiling Key.

You are advised to NOT change the primary Backup/Profiling Key, which in almost any case should always be *%os_user%*.

OS Usernames usually do not change, whereas Notes user names do, due to renames, for example. If the Notes username were used as the first, primary key, a user could not roam his/her previous configuration from before a rename.

When executing a Roaming/Rollback Action, MarvelClient scans all backup sets in the respective source directory or database, filtering out any backups for which the primary key does not match.

Example directory structure in combination with a primary key *%OS_USER%* and the current user being OSUser1:

- *H:\Backup\OSUser1\Computer1\...*
- *H:\Backup\OSUser1\Computer2\...*
- *H:\Backup\OSUser2\Computer1\...*
- *H:\Backup\OSUser2\Computer2\...*

In above example, only the first two backup sets will be looked at in detail, for which then the remaining subkeys (if any) are compared – only backup sets that match ALL roaming keys (in case more than one was specified) are further checked for whether they are more recent than the last backup of the client currently being launched.

> ***Note: The primary key for a Roaming/Rollback Action must be the same as the primary key of the Backup/Monitoring Action it relies upon. A Roaming/Rollback Action can make use of less keys than its corresponding Backup/Monitoring Action.***

Example: Backup/Monitoring Action

%os_user%\%notes_user_abbreviated%\%os_computer%\%notes_version%

+ Roaming/Rollback Action

%os_user%

= Default Roaming configuration where a user will always get the most recent configuration on any machine or Notes release

Backup Action

%os_user%\%notes_user_abbreviated%\%os_computer%

+ Roaming Action

%os_user%\%notes_version_major%

= Roaming configuration where a user would only roam within major releases, but not from one release to another (i.e. 8.x to 8.x and 7.x to 7.x but not 7.x ⇔ 8.x)

If more than one backup set is found in the results, then the **most current set** will be used (if none of the found backup sets is actually more current that the local client's last backup, no roaming/rollback will take place).

- Please be aware that Placeholders such as *%notes_user%* cannot be resolved prior to login. In such a case, a roaming Action would fail or worse produce unexpected results (by roaming based on the then resolved Notes username being *Unknown!Unknown*). In this case, use the Placeholder *%os_user%* as primary key

- In case you roam via a **UNC path**, make sure that you also append *<os:user>* to the source directory, such as *\\uncserver\backup\<os:user>*

  If you were to just use *\\uncserver\backup* as the source directory, ALL users would always have to scan ALL other user's backup sets, which would significantly increase the time needed to run through otherwise just a few relevant backups.

  If you need further help with setting up UNC-path based roaming, please contact panagenda support (support@panagenda.com)

TIP

It is not mandatory for the individual parts of the Roaming/Profiling keys to be composed of Placeholders. You can also choose a (static) string for the sub-key. One of several use cases could be saving a certain file in the Analyze database just once; and if you wanted to prevent this file from being included in the backup sets for monitoring purposes. Example: *%notes_user%\IDFILE_BACKUP*

File-based Backups and Roaming Sets

File-based backup and roaming sets also introduce a so called *revision.xml* file in the backup directory, storing further meta information, such the Backup/Profiling Key that was used when creating the backup, when which files last changed, whether a file was originally unzipped, whether the source was a single file vs. multiple files zipped up, a checksum to detect whether files have effectively changed, and more:

Example:

```
<document>
  <revisioning>
     <header>
       <key>%OS_USER%\%NOTES_USER_ABBREVIATED%\%OS_COMPUTER%\%NOTES_VERSION%</key>
       <key_conv>Administrator\max_mustermann!panagenda\PC\8-5-3</key_conv>
       <updated>20091105T174454Z</updated>
       <notes_user>CN=max mustermann/O=panagenda</notes_user>
     </header>
     <file>
       <name>bookmark.xml</name>
       <revision>
               <number>0</number>
               <updated>20120207T174454Z</updated>
               <name>rev_1f98156e_bookmark.xml.zip</name>
               <org_name>bookmark.xml</org_name>
               <checksum>628130eafd625fc6a2f6423a17b61ec8</checksum>
               <zipped>yes</zipped>
       </revision>
       <revision>
               <number>1</number>
               <updated>20120206T192220Z</updated>
               <name>rev_3c92156e_bookmark.xml.zip</name>
               <org_name>bookmark.xml</org_name>
               <checksum>628130eafd625fc6a2f6423a17b61ec8</checksum>
               <zipped>yes</zipped>
       </revision>
     </file>
   </revisioning>
</document>
```

The above backup example consists of one single file (the compressed *bookmark.xml*) in two so called "revisions" (~versions).

Domino Database-based Backups and Roaming Sets

As previously mentioned, Backup/Monitoring Actions allow to save client configuration data to either a file set (network drive/SAN/NAS) or a Domino database.

File-based backup was described further up (see "File-based Backups and Roaming Sets" on page 41) – as for storing backups or roaming sets in a in a Domino database, the individual backup sets are then created as documents (respectively updated). These can be searched for in the view "v_upload" (in Domino Designer "(I. Upload)" in the Analyze database). Here, they are sorted by converted backup key, in order to allow for quick access (without having to open unnecessary documents). All information necessary to find the matching backup set, is displayed in this view. The meta information mentioned above (under "File-based Backups and Roaming Sets" on page 41) can be found in the "t_xml" field of each such backup document. Files are stored as Notes file attachments in each backup document.

notes.ini Entries (for Backup and Roaming)

When writing backup or roaming sets, MarvelClient maintains the following three *notes.ini* entries:

- *MC_LastBackupDate*
- *MC_LastProfilingKey*
- *MC_LastProfilingKey_Conv*

In above example these would be stored as follows:

- *MC_LastBackupDate=20120207T174454Z*
- *MC_LastProfilingKey=%OS_USER%\%NOTES_USER_ABBREVIATED%\%OS_COMPUTER%\%NOTES_VERSION%*
- *MC_LastProfilingKey_Conv=Administrator\max_mustermann!panagenda\PC\8-5-3*

If a roaming Action has identified the most recent backup in a source directory or database, the date/time of that backup set is compared to above "MC_LastBackupDate". If "MC_LastBackupDate" is empty or older than the date/time of the most recent backup set found, roaming will take place.

The panagenda Working Directory

The panagenda Working Directory is the place where MarvelClient stores information about all Actions it has synchronized with the Configuration database, as well as files for upload into the Analyze database and for troubleshooting purposes (like the log-file for example).

By default, panagenda MarvelClient creates these files and folders in the OS users profile (*%os_local_app_data%*), such as:

- ***C:\Users\UserName\AppData\Local\panagenda\marvel*** (on Windows Vista/7)

or:

- ***C:\Documents and Settings\UserName\Local Settings\Application Data\panagenda\marvel*** (on Windows XP)

If needed, the MarvelClient Working Directory can be changed by adjusting "MC_WorkingDirectory" in *notes.ini* such as:

- *MC_WorkingDirectory=%notes_data%\mc*

It is advisable to make sure that the Working Directory is already specified properly in *notes.ini* PRIOR to client startup as opposed to dynamically changing it via the Configuration database – in the latter case, MarvelClient would first start logging in the previous Working Directory and then switch into the new/changed directory, which may create confusion (and only "half" of *log.xml* being uploaded into the Analyze database).

Within the Working Directory the following files are located:

- ***actions.xml*** = list of all Actions that apply to the specific user. By default, this file gets updated at every IBM Notes client start, shutdown and every 60 minutes whilst the client is up and running. This file also stores information for whether one time Actions have already been processed – thus if *actions.xml* is deleted, ALL one time Actions will be carried out again (provided that they match whatever other Conditions have been set for the respective Actions)

- ***log.xml*** = log of what happened on the local machine with regards to MarvelClient. This file gets recreated upon each IBM Notes client start.

Within the Working Directory there is also the ***temp*** sub-directory, which contains the following files (these files can be safely deleted any time, except for when they are uploaded into the Analyze database):

Note that depending on configuration, other files, such as keyfile.xml or log_cleanup.xml me be added.

- ***bookmark.xml*** = list of the users' bookmarks
- ***client_ecl.xml*** = list of the users' ECL entries
- ***config.xml*** = list of various local configuration details, such as the users' Notes client, operating system and MarvelClient settings
- ***databases.xml*** = list of found local databases from in- and outside the Notes data directory
- ***desktop.xml*** = list of desktop icons
- ***images.xml*** = icons displayed on the users workspace (=the pictures on the database tiles)
- ***names.xml*** = locations configuration and replicator page details
- ***notes_ini.xml*** = users' *notes.ini*
- ***toolbars.xml*** = list of used toolbars

▶ **All of these files are updated at every IBM Notes Client restart.**

You may also find a sub-directory "roaming" in the Working Directory, which is used by MarvelClient Roaming.

Optionally there is also a ***roaming*** directory, where the roaming files from your MarvelClient server are stored (*desktop.xml*, *bookmark.xml* and *notes_ini.xml*).

Note: MarvelClient log and Action files store date/times in UTC format.

How to Read the MarvelClient Log

As described in the previous topic, MarvelClient maintains a detailed log of what it found and did (not do) into ***log.xml*** in the panagenda Working Directory. It's recommended to also collect this file in the MarvelClient Analyze database, allowing you check every end-user's log from a central point of information.

While testing MarvelClient Actions, you may find an Action does not execute and you want to know what happened. Or perhaps an end-user is complaining of long Notes client startup times and you want to see what exactly is happening on this machine.

Basics:

Looking at a log file, each line has a millisecond indicator at the beginning, showing the number of milliseconds that have elapsed since the log was started:

```
<le><t>273</t><l>2</l><m>before login done; duration: 63</m></le>
<le><t>2283</t><l>2</l><f>LDR::MCAfterLogin</f><m>waiting for local replica scan
     to finish</m></le>
<le><t>2283</t><l>2</l><f>LDR::MCAfterLogin</f><m>thread done</m></le>
```

Also note that at the beginning of each *log.xml* file, you will find the following important lines:

```
<le><t>0</t><l>2</l><m>Version 3.0.2.1045 C:20110414T101124 B:20110414T102608</
     m></le> //The MarvelClient release
<le><t>0</t><l>2</l><m>now: 20120725T131632Z</m></le> //When was the log file
     started in UTC time - a quick peek tells you whether this is a rather recent
log or quite an outdated one
<le><t>0</t><l>2</l><m>client: Release 8.5.3FP2|July 12, 2012</m></le> //The
     exact IBM Notes release
```

Further details on Standard vs. Basic client etc. can be found in *config.xml* (from within the Working Directory).

Performance issues:

In general, look at larger millisecond jumps (> 1,000 for example) – those are usually the ones that indicate performance issues. If you happen to have a log file with no significant jumps but ongoing small jumps from line to line it is quite definitely an end-user who is starting Notes along with a cold (Windows) boot and the disk is going berserk trying to initialize Windows, Notes, and lots of other Applications/Drivers etc. In a normal *log.xml* on a halfway decent machine, you should find that the ms-indicator doesn't jump for many (like 60+) lines in a row.

In the following example, it took 799 ms to open the local mailfile replica – most presumably because it is a several GB mail file – more details to be found in *databases.xml*, then.

```
<le><t>3680</t><l>2</l><f>ND::Database_Impl::Open</f><m>opening
    database: (C12572D7:005E2FC0) 'mail\aname.nsf'</m></le>
```

```
<le><t>4479</t><l>2</l><f>AM::A_Viewreader::OpenSourceDB</f>
    <m>database open</m></le>
```

In 99% of all cases you should be able to easily determine wether you have performance bottlenecks because of your Domino server performance (for instance when the Configuration and/or Analyze database are being opened), or towards the data directory (especially if the Notes data directory is on a network drive).

TIP

When searching for "duration" in a *log.xml* file, you will find summaries of how long specific sections took to process – you can also find these performance indicators in the Analyze database (see: page 55).

Errors:

Actions are logged with their name and their Action-ID:

```
<le><t>2872</t><l>2</l><i>2</i><f>AM::ActionMan::ExecuteAction
    </f><m>action 'ActionName', ID
    '4832984CECFB9116C1257A610044278B'</m></le>
```

Searching for a particular Action title or ID helps you identify whether it ran or not, or whether it was not even downloaded by the client yet. You can also search for the string "error" in *log.xml* to find out whether any errors occurred, which might explain for why an Action was not downloaded in the first place, for example.

ANALYZE DATABASE

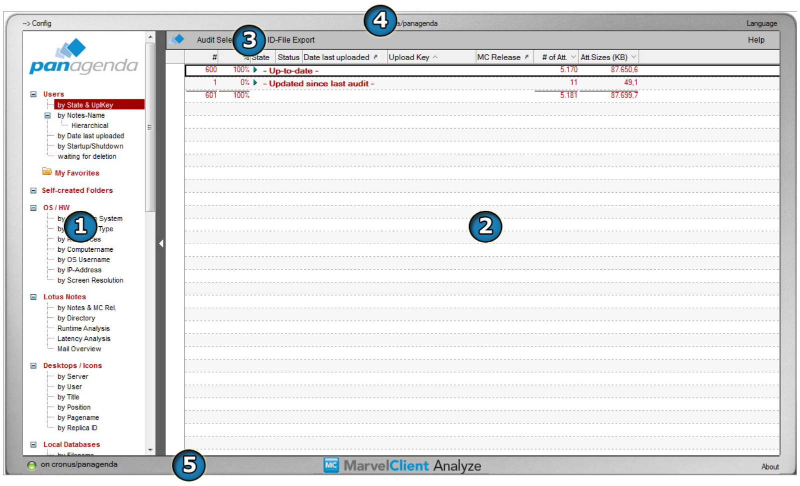
Introduction

The Analyze database gives you insight into all the information collected by MarvelClient across all your IBM Notes clients – as a reminder, any "configuration" work is done from the Configuration database. This also means, that editing documents in the Analyze database does not affect any client configuration but only distorts the collected data (apart from XSLT Configuration documents in the Analyze database – please contact panagenda support before modifying any of those documents).

Basically, the Analyze database stores one profile as per combination of operating system user name, IBM Notes user name (including certifiers – "/? is replaced with "!?, " " (blank) with "_"), machine name and IBM Notes release (for example "fvolger\florian_vogler!panagenda\VOF-W500\8-5-3"). For Citrix systems the machine name is replaced with the word "Citrix", to avoid the collection of countless (similar) profiles across multiple Citrix servers (refer to "Roaming/Profiling Key" on page 38 for details on that).

Whilst the information in the Analyze database can be used for various purposes (such as understanding "what is out there", performance optimization, software deployment, upgrade projects, or localization of root causes for helpdesk incidents), it does not allow for any conclusions about end-user productivity.

The following screenshot shows the MarvcelClient Analyze database:



#	%	State	Status	Date last uploaded	Upload Key	MC Release	# of Att.	Att.Sizes (KB)
600	100%	- Up-to-date -					5.170	87.650,6
1	0%	- Updated since last audit -					11	49,1
601	100%						5.181	87.699,7

Figure 14: Analyze database – Overview

1. The left hand frame shows the **navigation bar**

2. The display in the **main area** depends on your current selection in the navigation bar

3. The buttons/actions displayed in the **action bar** depend on the currently opened view or document in the main area

4. In the topmost area of the Analyze database frameset you will find a link on the left hand side which **opens the Configuration database** (--> *Config*) in the middle the **current user** is displayed and on the right hand side you see a button to open the **language selection** (currently English and German are available)

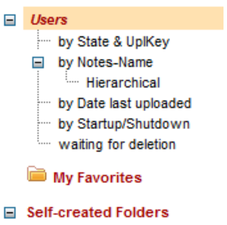
5. The bottom most frame shows whether you are working **local** (red light + *on [local]*) **or on server** (green light + *on YourServer/YourCompany*) on the left hand side, and on the right hand side you can check which **release of the Analyze database** you are using (*About* button)

The Navigator

The information uploaded into the Analyze database is visualized through various views. Whilst you can always deep-dive into particular user configurations, we recommend to use the Analyze database views for a better understanding of how large the delta is between where you would like to, need to, or should be, and what the most recent client configurations look like. Looking at single end-user configurations should only be necessary to spot root causes for why certain clients do not match your desired target state, or to clarify what is needed to adjust client configurations accordingly.

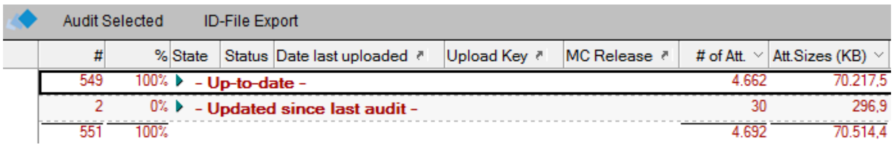
The Analyze database holds various different sections to display information and for administration purposes (Note that some sub-views are self-explanatory and thus not described below):

Users (and Self-created Folders)



Within the views of the ***Users*** section, you will find the main User Profiles, which are uploaded/updated from your IBM Notes clients, sorted by different criteria. Each of these shows the same three view actions in their **action bar**: *Audit Selected*, *ID-File Export* (role [Security] needed – see "Advanced" on page 71) and *Help*. Double-clicking on a User Profile in one of these views opens the respective **User Profile Document** (see "User Profile Document" on page 72).

by State & UplKey provides you with a quick overview on whether (and which) entries have (not) already been audited manually or by the (optional) Audit All Agent (also see "(Audit All) Agent" on page 13). Selected User Profile Documents can be audited manually by clicking on the *Audit Selected* view action.



#	%	State	Status	Date last uploaded	Upload Key	MC Release	# of Att.	Att.Sizes (KB)
549	100%	▶ - Up-to-date -					4.662	70.217,5
2	0%	▶ - Updated since last audit -					30	296,9
551	100%						4.692	70.514,4

Figure 15: Views in the Analyze Database: *by State & UplKey*

In the by ***Notes-Name\Hierarchical*** view, all certifier components are resolved from back to front and hence categorized top-down (such as panagenda ⇨ Europe ⇨ Austria instead of Austria ⇨ Europe ⇨ panagenda).

The *by Startup/Shutdown* view only provides useful information if you upload the *config.xml* file to the Analyze database both at Notes client startup AND shutdown; the view then allows you to determine whether end-users restart their IBM Notes clients almost every day or whether you have hibernating end-users that only restart their clients once in a while.

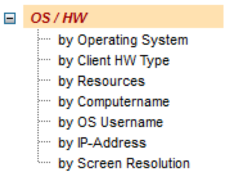
You will find further details on the ***waiting for deletion*** view under "Administration" on page 66.

If you have User Profiles of special interest, drag and drop them to the My Favorites folder to make them faster accessible and thus easier to revisit.

Self-created Folders:

Here, all your self-created folders are listed – if any. If the views provided from out of the box within the Analyze database are sufficient, and you have therefore not created any additional folders, this section remains empty.

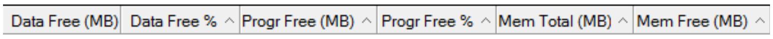
OS/HW



The views in this section are based on operating system and hardware information collected through *config.xml*. In the **action bar** of these views you will find two view actions: *Audit Selected* and *Help*. Double-clicking on a document in one of the views opens the corresponding **User Profile Document** (see "User Profile Document" on page 72).

by Client HW Type distinguishes between desktop, laptop and Citrix clients. Note that you can optionally also detect VMWare and Windows Virtual PC hosts, for details see Configuration database *Examples → Audit/Monitoring → Detect VMWare/Virtual PC*.

In order to quickly locate resource problems quickly, the ***by Resources*** view offers you all you need. Click on a particular column header to sort by free disk space (in MB or percent) in the data or program directory, or by total or available physical memory.

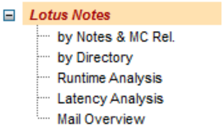


Data Free (MB)	Data Free % ^	Progr Free (MB) ^	Progr Free % ^	Mem Total (MB) ^	Mem Free (MB) ^

Figure 16: *by Resources* – Columns and Sorting

Note that end-users with **less than 10% of free disk space** in either the data or program directory are likely to suffer from performance problems, as such little free disk space heavily adds to fragmentation of files on disk. End-Users with **less than 2 GB of total physical memory** are most likely not enjoying daily usage of an IBM Notes Standard (Eclipse) client. End-Users with **less than 500 MB of available Memory** are most probably suffering from OS paging when switching between applications.

Lotus Notes

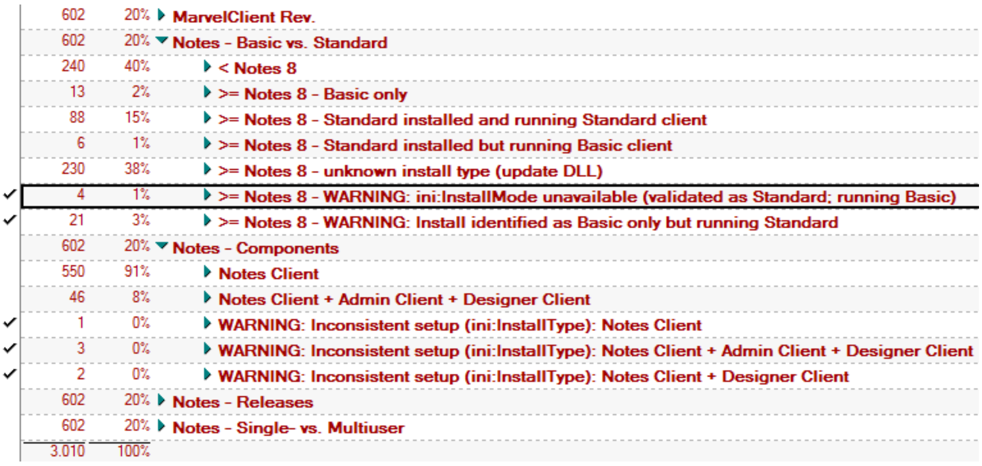


- *Lotus Notes*
  - by Notes & MC Rel.
  - by Directory
  - Runtime Analysis
  - Latency Analysis
  - Mail Overview

These views visualize various details about your IBM Notes clients. In the respective **action bars** you will find: *Audit Selected* and *Help*. All views are based on the **User Profile Document**. Whilst some of the herein displayed information can also be found in your public address book (for example Notes Releases), MarvelClient aims for presenting everything needed from one central location ("one-stop-shopping").

by Notes & MC Rel.(ease) displays Notes client release/version information – this might be very useful during the course of an upgrade project, in order to check whether all clients were already upgraded successfully, or to localize exactly which IBM Notes clients haven't been upgraded yet.

In addition to monitoring all current Notes client releases, this view is also very helpful as it alerts you of various *Warnings* regarding possible inconsistent installations in your environment:



	602	20%	▸ MarvelClient Rev.
	602	20%	▾ Notes - Basic vs. Standard
	240	40%	▸ < Notes 8
	13	2%	▸ >= Notes 8 - Basic only
	88	15%	▸ >= Notes 8 - Standard installed and running Standard client
	6	1%	▸ >= Notes 8 - Standard installed but running Basic client
	230	38%	▸ >= Notes 8 - unknown install type (update DLL)
✓	4	1%	▸ >= Notes 8 - WARNING: ini:InstallMode unavailable (validated as Standard; running Basic)
✓	21	3%	▸ >= Notes 8 - WARNING: Install identified as Basic only but running Standard
	602	20%	▾ Notes - Components
	550	91%	▸ Notes Client
	46	8%	▸ Notes Client + Admin Client + Designer Client
✓	1	0%	▸ WARNING: Inconsistent setup (ini:InstallType): Notes Client
✓	3	0%	▸ WARNING: Inconsistent setup (ini:InstallType): Notes Client + Admin Client + Designer Client
✓	2	0%	▸ WARNING: Inconsistent setup (ini:InstallType): Notes Client + Designer Client
	602	20%	▸ Notes - Releases
	602	20%	▸ Notes - Single- vs. Multiuser
	3.010	100%	

Figure 17: Notes Releases – Inconsistent Installations

Thereby, the *by Notes & MC Rel.* view offers you an easy way to check whether your client landscape is actually properly configured, which is vital for Policies, SmartUpgrade and other tooling provided by IBM Notes/Domino (in many environments, 5 to 10 percent of all users are misconfigured; sometimes even up to 100% if corporate software deployment did not take care of the necessary *notes.ini* entries).

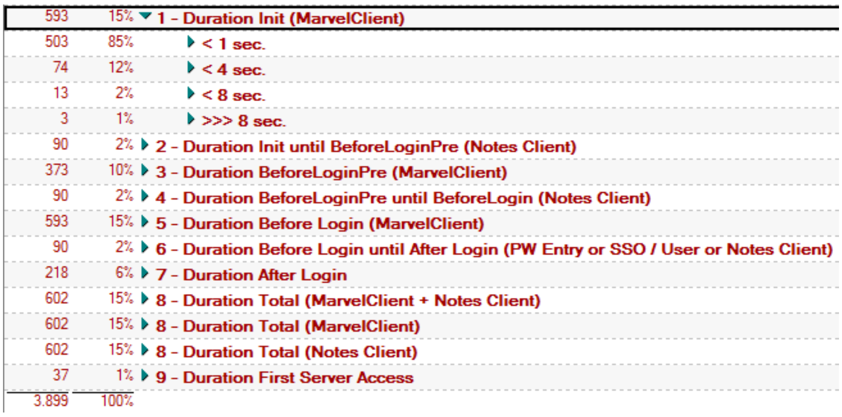
The ***by Directory*** view can be used to check where exactly Notes relevant folders are stored (IBM Notes data, program and Shared data directories), as well as *notes.ini* (to monitor the delta towards company standards, for example).

Note that the information displayed in this view is 100% reliable, since it is not derived from the Windows registry or "from outside" IBM Notes, but MarvelClient obtains these details from "inside" each client at runtime – in essence, the IBM Notes client tells MarvelClient which folders and files it is using after it has been launched.

TIP

This information can either also be passed on to corporate software deployment tools, to remove any otherwise related guesswork and prevent lengthy and error-prone coding work out of client upgrades, or the information can be used in conjunction with MarvelClient Upgrade (see: http://www.panagenda.com/en_uk/marvelclient_upgrade_module for further information).

The ***Runtime Analysis*** views are based on several runtime measurements:

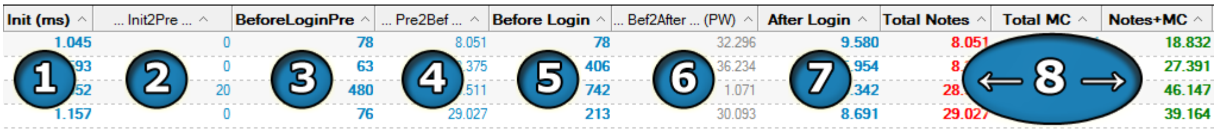


593	15%	▼ 1 - Duration Init (MarvelClient)
503	85%	▶ < 1 sec.
74	12%	▶ < 4 sec.
13	2%	▶ < 8 sec.
3	1%	▶ >>> 8 sec.
90	2%	▶ 2 - Duration Init until BeforeLoginPre (Notes Client)
373	10%	▶ 3 - Duration BeforeLoginPre (MarvelClient)
90	2%	▶ 4 - Duration BeforeLoginPre until BeforeLogin (Notes Client)
593	15%	▶ 5 - Duration Before Login (MarvelClient)
90	2%	▶ 6 - Duration Before Login until After Login (PW Entry or SSO / User or Notes Client)
218	6%	▶ 7 - Duration After Login
602	15%	▶ 8 - Duration Total (MarvelClient + Notes Client)
602	15%	▶ 8 - Duration Total (MarvelClient)
602	15%	▶ 8 - Duration Total (Notes Client)
37	1%	▶ 9 - Duration First Server Access
3.899	100%	

Figure 18: Analyze Database: Runtime Analysis – Categories

The vertical categories in this view help you to quickly localize clients with performance problems. When you expand these categories until an end-user is shown, you will see that runtime details are spread across various columns (horizontally). Bold columns are MarvelClient runtimes, non-bold ones are IBM Notes runtimes. The order of the columns (from left to right) is the same as the order of the vertical categories:

Example: *1 - Duration Init (MarvelClient)* = ***Init (ms)***:



Init (ms)	... Init2Pre ...	BeforeLoginPre	... Pre2Bef ...	Before Login	... Bef2After ... (PW)	After Login	Total Notes	Total MC	Notes+MC
1.045	0	78	8.051	78	32.296	9.580	8.051		18.832
93	0	63	375	406	36.234	954	8.		27.391
52	20	480	511	742	1.071	342	28.		46.147
1.157	0	76	29.027	213	30.093	8.691	29.027		39.164

Figure 19: Analyze database: Runtime Analysis – Columns

1. Duration Init (MarvelClient)/***Init (ms)***: Should typcially be done within ~100 ms. Longer run times indicate a disk bottleneck (such as launching IBM Notes in parallel to a Windows boot storm – consequently a quick MarvelClient is hard to achieve). Usually this happens when IBM Notes is started right after booting the operating system, or when a virus scanner runs during client startup

2. Duration Init until BeforeLoginPre (Notes client)/*...Init2Pre...*: *IBM Time*. Initialization duration of the IBM Notes client (in ms). Should typcially be done in <100 ms

3. Duration BeforeLoginPre (MarvelClient)/***BeforeLoginPre***: Time period (in ms) for the execution of all *Before login – Init* Actions (such as Roaming). Should typically be done in <100 ms UNLESS roaming is used. If roaming is used, it is executed right here and can take a couple of seconds. Long runtimes (without roaming) indicate network disk bottlenecks. Without MarvelClient Roaming this runtime should be finished within less then 100 ms – longer durations indicate waiting for the operating system (boot storms for example). With MarvelClient Roaming, anything longer than a couple of seconds (2-4) indicates disk bottlenecks and/or too large roaming sets (and hence optimizable through compaction, for example)

4. Duration BeforeLoginPre until BeforeLogin (Notes client)/*...Pre2Bef...*: *IBM Time*. Should be done in <4 seconds (<4,000 ms). Longer runtimes indicate bottlenecks (disk fragmentation, elder Notes 8 releases, outdated ODSs, etc.)

5. Duration Before Login (MarvelClient)/***BeforeLogin***: Execution of Actions with the Run Type *Before login* (such as preparing Location Documents to be displayed in the login dialog). Without any Actions, this time period should take <100 ms

6. Duration Before Login until After Login (Pw Entry or SSO / User or Notes client)/ ...Bef2After...(PW): Password entry/Single-Sign-On (with SSO, this should take <100 ms; without SSO it takes as long as it takes the user to enter his/her password)

7. Duration After Login/**After Login**: Time period between Login and loading of Eclipse and/or other components. This is the main section for MarvelClient Actions (c.f.: After login on page 27). Right after this period, control is given back to the IBM Notes client.

   The total duration of After Login should not exceed 5 seconds; if it does, this indicates disk bottle necks, communication issues towards the Domino server, anti-virus programs, OS disk indexing or other root causes which should be sought after

8. The columns ***Total Notes***, ***Total MC*** and (total) ***Notes+MC*** display the respective accumulated durations

- First Server Access: Three columns show whether the MarvelClient feature *On First Server Access* (see: "On First Server Access" on page 27) is available/enabled, od, if so, how long it took to execute it: ***FSAC?*** shows you, whether this feature is available and activated, ***FirstSrvAcc (UTC)*** shows when exactly Fist Server Access was triggered, and ***FirstSrvACC*** shows how long MarvelClient needed to execute all Actions within the Run Type *On fist server access after start* (including synchronization with the Configuration database)

Continuing with the views:

Latency measurements displayed in the view ***Latency Analysis*** are quite similar to a "ping", where latency is not just measured in total for how long it took to "poke" a server and receive a response, but is measured in both directions: How much time passed while transferring a standardized package from the client to the Sever, and vice versa (upload vs. download). The results are categorized by performance (*PerfCat* – in ms). Optimal values are between 0 and 10 ms. In corporate networks, both the up- and download latency are usually the same due to synchronous connections, whereas "home office" users usually have higher upload than download latencies, due to good download bandwidth, but bad/lesser upload bandwidth on a DSL connection. If latency exceeds > 30 ms, end-users are suffering from severe(!)delays when working with databases on servers. Opening up a server-side mailfile, for example, can well take 3-4 seconds with a latency of 40 ms, same for creating an email, a calendar entry, switching views, and so on. Users with such bad latencies should be configured for local replicas of often used applications.

- For further details regarding the interpretation of ***Runtime Analysis*** and ***Latency Analisis*** refer to this presentation: http://tinyurl.com/axu8o2j

The view ***Mail Overview*** shows various details about the usage of mail databases in a very fast and easy way: First of all, it shows you who works on servers vs. who works on a local replica. For local replicas, the view shows whether the local path matches the filepath on the user's mailserver – if it does not, this can lead to alerts not working for calendar entries, for example.

Note that for all details to be properly displayed in this view, a Location Action is required in order to lookup each user's public addressbook details; otherwise you will see documents categorized as [- *Serverpath unknown* -] (this also applies for when you have actually configured the necessary Location Action, but users launch their client in Offline mode).

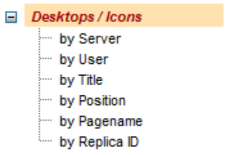
Views based on User Profiles vs. Detail Documents:

The before described *Lotus Notes* section in the Analyze Navigator frame is the last section from within the Analyze database that displays main User Profile Documents. All of the following views are "*detail document* views", which display desktop/workspace details, *notes.ini* details, database details and more based on separate documents maintained by the Audit and Cleanup agents.

The Audit agent (whether scheduled or run manually) unzips client configuration details and splits them into one or more detail documents for each desktop, *notes.ini*, database details and similar such objects. This is because Notes/Domino views only allow for the display of a maximum of 32 KB of data as per document in a view column (since text field lists have some overhead, and various view columns in the Analyze database concatenate multiple list fields into one column, the real splitting limit is around 5 KB). Since many users have more icons, databases and similar than can be fit into 32 KB – respectively 5 KB - of data, many users have more than one detail document as per main User Profile Document and object.

Note that none of the detail document views will display any view action bar.

Desktops/Icons



The *Desktops/Icons* views provide insight into the desktop/workspace configuration of your IBM Notes clients. Double clicking on a documents in one of the Desktop/Icon views opens the **Desktop Details Document** which is further explained further down.

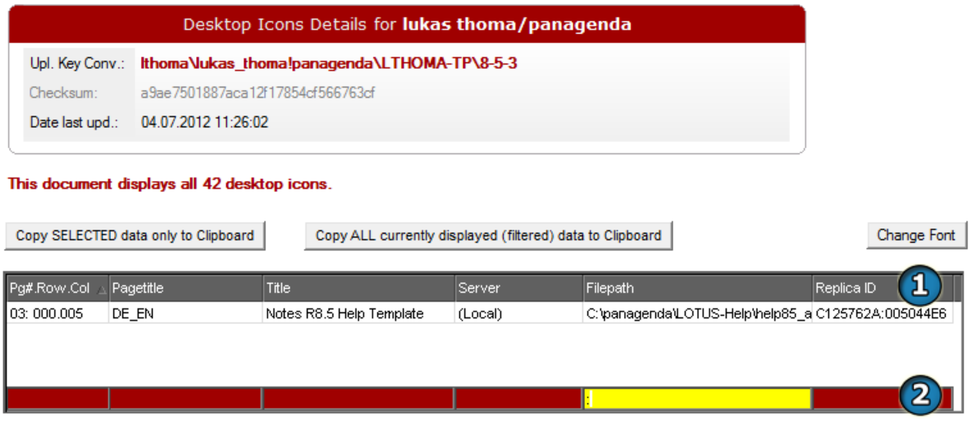
The ***by Server*** view is essential for finding out which end-users will be affected by a server consolidation – who uses links to the relevant server(s) etc. (this, of course, also considerable eases troubleshooting during the course of consolidations).

This view also allows to quickly identify old, outdated names of servers that have been decommissioned long ago, but for which users (or rather clients) still have links on their desktops – naturally you can then easily create a respective Mass Change of type *Delete* in the Configuration database to remove any such old, outdated links from all desktops. Best of all, with MarvelClient you don't have to worry about who actually has any such old links – simply run this "remove old server links" Action for all clients and MarvelClient will take care of it seamlessly.

by Title can come in handy when carrying out a security audit: In this view, you can easily check who (still) has a link to certain confidential databases, for example.

Use ***by Position*** and ***by Pagename*** to find out what icons people have at the top left of page one or two, for example, or what titles their first three pages have. This is especially useful to determine how many users will consider their workspace as "having changed overnight" when you start to standardize end-user desktop configurations.

Desktop Details Document:



Desktop Icons Details for **lukas thoma/panagenda**

Upl. Key Conv.:	lthoma\lukas_thoma!panagenda\LTHOMA-TP\8-5-3
Checksum:	a9ae7501887aca12f17854cf566763cf
Date last upd.:	04.07.2012 11:26:02

This document displays all 42 desktop icons.

Copy SELECTED data only to Clipboard
Copy ALL currently displayed (filtered) data to Clipboard
Change Font
1
2

Pg#.Row.Col	Pagetitle	Title	Server	Filepath	Replica ID
03: 000.005	DE_EN	Notes R8.5 Help Template	(Local)	C:\panagenda\LOTUS-Help\help85_a	C125762A:005044E6

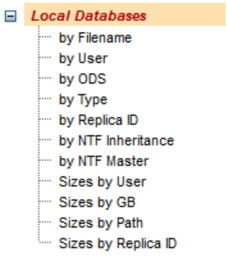
Figure 20: Desktop Details Document

Each Desktop Details Document allows you to investigate all desktop icons of ONE particular end-user; whilst the same information is available from within an embedded view in User

Profile Documents, the information is presented much more user friendly within (not just Desktop Icon, but any) Detail documents: Detail documents allow you to **sort** (1) each column ascending and descending and the last line serves as a **filter** (2). On the screenshot for instance you see a search for ":" (colon) in the *Filepath* column (as a result, only links pointing to databases that reside outside the IBM Notes data folder are displayed; i.e. only absolute links to databases).

> ***Note that desktop icons may point to databases that no longer exist, or to which a user no longer has access. If you would like to do a full automatic cleanup of such old and outdated links, panagenda GroupExplorer might be of further interest – see:*** http://www.panagenda.com/en_uk/groupexplorer ***for more).***

Local Databases



The views within the *Local Databases* section provide you with many important details of each end-users' local databases. Double-clicking on a document in one of these views opens the corresponding **Database Details Document**, providing investigation options similar to a Desktop Details Document (see: "Desktop Details Document:" on page 59). The table from within Database Details Documents consists of the following columns: *Filepath*, *ODS*, *Replica ID*, *Size (MB)* and *Last Modified* (again you can use these to sort and filter an end users' list of databases)

by Filename and ***by User*** provide a very easy way to see, whether users have databases located outside of the IBM Notes data directory (=**[Absolute]**) and which databases are stored inside their respective Notes data directory or any subfolder of it (=**[Relative]**). This makes it easy to, for example, identify unwanted "local" databases on network drives, database copies or replicas on USB sticks, and more. Furthermore with this view you can make sure that not a single important email will be lost when centrally archiving email, by easily spotting mail archives that are stored outside of IBM Notes data directories.

by ODS is a very important view, especially when it comes to IBM Notes client performance: All local databases are categorized by ODS version, allowing you to immediately see client performance optimization potential (**Note**: An upgrade from ODS version 43 to ODS version 51 equates to up to 80%(!) less file I/O).

In the view ***by Type*** you will find two main categories on the "highest level": *Copy/Replica* and *Managed Replicas*. This allows you to easily find out how many and which end-users are using what kind of replicas.

by NFT Inheritance shows you which local databases have which template assigned.

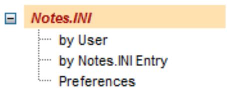
by NTF Master shows you which local databases are a template.

These two views are helpful to spot client performance issues and possible problems with Domino Policies, for example. A multi-user client with a Shared data directory should not have any "StdR4PersonalAddressbook" template for example, as a design refresh would then not use the shared (assumed more recent) template, but the (assumed outdated) one from the local data directory.

Sizes by User gives you an indication of how large end-user IBM Notes data directories are; huge data directories can be further investigated by opening a respective detail document and descending sorting of database sizes in order to immediately pinpoint the largest local databases or replicas.

Using the other ***Sizes by...*** views allows you to easily localize large(r) databases, as well as understanding the spreading of database sizes in general.

Notes.INI



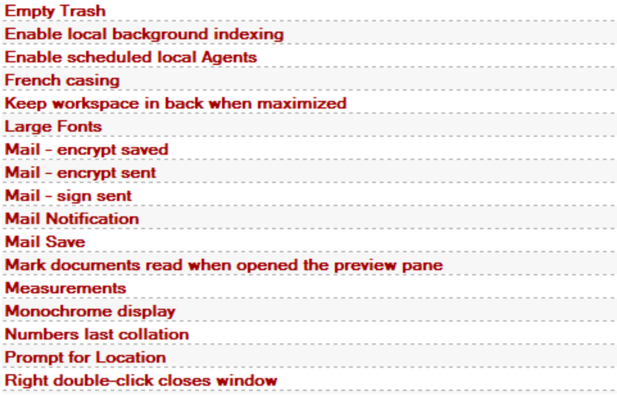
In the views of the *Notes.INI* section you can observe all *notes.ini* entries that were collected and audited by MarvelClient. The three different sub-views (***by User, by Notes.INI Entry*** and ***Preferences***) offer different perspectives on the collected data.

Example questions that could be answered using these views:

- How many/which end-users (do not) have port compression enabled?

- Who signs/encrypts emails upon sending and/or saving?
- Which clients have anti-virus registered as EXTMGR_ADDINS or NSF_HOOKS?
- How many users have a particular "$..." variable which is important for one of your applications? (for example for telephony integration, or a CRM system)?
- Are ther users that have "UseBasicNotes=1" in theri *notes.ini*?

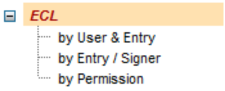
The ***Preferences*** view "decodes" the corresponding *notes.ini* setting for ease of use, as the Preferences line in *notes.ini* is otherwise "fairly hard to read":



Empty Trash
Enable local background indexing
Enable scheduled local Agents
French casing
Keep workspace in back when maximized
Large Fonts
Mail - encrypt saved
Mail - encrypt sent
Mail - sign sent
Mail Notification
Mail Save
Mark documents read when opened the preview pane
Measurements
Monochrome display
Numbers last collation
Prompt for Location
Right double-click closes window

Figure 21: Views Analyze Database: *Note.INI Preferences*

ECL



ECL
by User & Entry
by Entry / Signer
by Permission

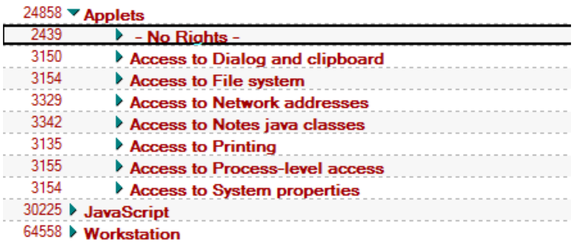
These views are based on the analysis of each Notes client's ECL (=Execution Control List).

by User & Entry focusses on the end-user and thus the main category is his/her username, followed by ECL entries and then respective permissions (for example: *YourUser/ YourCompany* → *- Default -* → *Applets* → *- No Rights -*).

Within ***by Entry/Signer*** the category on the highest level is the ECL entry. This view is perfectly suited to answer questions like "for how many users/whom does - *Default* – have dubious rights?".

The view ***by Permission*** is categorized by ECL Section ("Applets", "JavaScript", "Workstation"), followed by the respective Permissions within each Section, to then display each ECL entry that is granted the respective Permission (for instance:

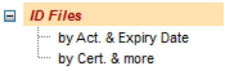
Applets → - No Rights - → - Default - → YourUser/YourCompany).



24858 Applets
2439 - No Rights -
3150 Access to Dialog and clipboard
3154 Access to File system
3329 Access to Network addresses
3342 Access to Notes java classes
3135 Access to Printing
3155 Access to Process-level access
3154 Access to System properties
30225 JavaScript
64558 Workstation

Figure 22: Views Analyze Database: *ECL by Permission*

ID Files



ID Files
by Act. & Expiry Date
by Cert. & more

ID file details are not collected by default – this requires setup of an appropriate Action

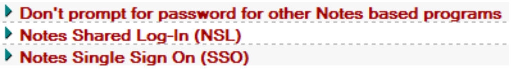
.

Note that collecting ID File Details is not to be confused with collecting ID files as such! – the collection and display of ID file details does not require collecting actual ID files.

by Act. & Expiry Date shows when user and certifier IDs were first activated and when they expire.

by Cert. & more allows you to

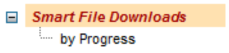
- identify whether all IDs with the same certifier were actually created with the same certifier ID (yes, we've seen companies create a new certifier ID which LOOKS the same, but is in fact very different – for example because no one could remember the password for the original certifier ID)
- see who is using Single Sign On (SSO) or Notes Shared Login (NSL)
- see for which ID files "Don't prompt for password for other Notes based program" is en- or disabled



Don't prompt for password for other Notes based programs
Notes Shared Log-In (NSL)
Notes Single Sign On (SSO)

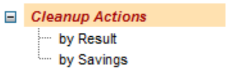
Figure 23: Views Analyze Database: *ID Files* – Password

Smart File Downloads


Smart File Downloads
by Progress

If so used, this section allows you to track the progress of "Smart File Downloads".

Cleanup Actions


Cleanup Actions
by Result
by Savings

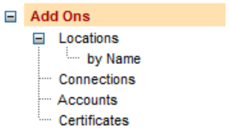
If so used and uploaded into the Analyze database via the corresponding *log_cleanup.xml*, this section shows the results of Cleanup Actions.

The ***by Result*** view is essential for analyzing simulated Cleanup Actions for as to what would happen if you were to not simulate such an Action: Which outdated or redundant replicas were found, which would have been deleted or moved, and which will (not) be restored again.

The ***by Savings*** view visualizes storage savings (potential if simulated) as per User Profile and in total. Simulations are marked with the info icon in the column labeled *Sim?*.

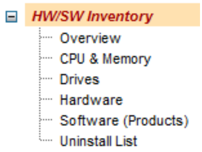
Note that you also have to upload the log_cleanup.xml file from the MC Temporary Directory to the Analyze database (=change your Backup Actions accordingly) – otherwise these views will remain empty.

Add Ons


Add Ons
Locations
by Name
Connections
Accounts
Certificates

These views display ***Locations***, ***Connections***, ***Accounts*** and ***Certificates***, if so collected – note that these (documents) are not collected by default. Also note that we do not recommend to collect certificates, since each IBM Notes client has approximately 300 of them. Collecting any of these documents from clients will copy/update those documents into the Analyze database just as they are stored in each users' personal addressbook (aka names.nsf).

HW/SW Inventory

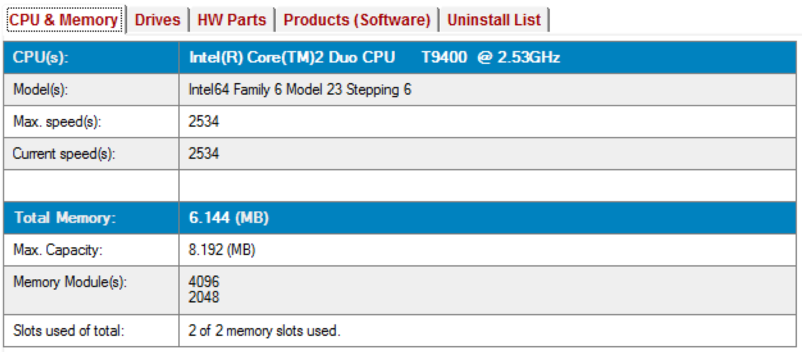


For Windows machines, you also have the option to collect more detailed hard- and software information (use the *Configure HW and SW Inventory* example Action in the Configuration database to set this up <u>View</u>: *MarvelClient → All Actions → Examples;* <u>Category</u>: *Audit/Monitoring → Optional Add-Ons*).

We advice you to NOT perform this extended HW/SW Inventory on a daily basis (but rather only on demand or once a month, for example), as such an audit can take up to two minutes.

Even though the audit is carried out in the background whilst the client is up and running, it does consume CPU and disk resources by interfacing with WMI (Windows Management Instrumentation). Also, hard- and software usually doesn't change all too often, as users do not have sufficient rights to install or uninstall software, for example.

Hard- and software inventory results are stored in separate **HW/SW Inventory documents**:



CPU(s):	Intel(R) Core(TM)2 Duo CPU T9400 @ 2.53GHz
Model(s):	Intel64 Family 6 Model 23 Stepping 6
Max. speed(s):	2534
Current speed(s):	2534
Total Memory:	6.144 (MB)
Max. Capacity:	8.192 (MB)
Memory Module(s):	4096 2048
Slots used of total:	2 of 2 memory slots used.

Figure 24: HW/SW Inventory Document – *CPU & Memory*

Apart from comprehensive hardware information (c.f. *CPU & Memory*, *Drives* and *HW Parts* tab) you also get plenty of details about the installed software across your Windows clients.

The *Products (Software)* tab also reveals the so called "product codes", which are needed in case you need to uninstall certain software without knowing which package/product code it was installed with in the first place.

Last but not least, the *Uninstall List* tab displays all uninstall information from the Windows registry. In general, the contents of the *Products (Software)* and *Uninstall List* tab should be similar; however, some elder Windows machines do not properly report back on the Products (Software) tab because of missing WMI patches (which is why we added the *Uninstall List* to make sure that independent from possible WMI issues, all Windows clients properly report back on installed software components).

Currently available *HW/SW Inventory* views are:

Overview shows who has/does not have OS Administrator rights,

CPU & Memory displays which CPUs are out there, how much memory the machines have, how much memory they could have, and how many memory slots are used out of how many (particularly useful before client upgrades – knowing that many machines have a free slot makes purchasing additional physical memory easy)

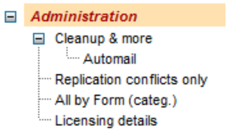
Drives lists which drives are mapped on clients and free disk space on each drive (honoring quotas)

Hardware shows hardware components and their status from the operating system's perspective (for example *OK*, *Stopped*, *Degraded* – for details, see Description of status property on http://tinyurl.com/cqkm9y9 for further details)

Software (Products) shows installed software categorized by *Vendor/Prod.*[uct]/*Ver.*[sion]

Uninstall List shows software that can be uninstalled categorized by *Vendor/Prod.*[uct]/ *Ver.*[sion]

Administration



This section holds views and forms/documents for administration purposes related to the Analyze database itself.

The ***Cleanup & more*** option links to the configuration of the (Cleanup) Agent. The Cleanup Agent can perform the following two Actions from within the Analyze database – either on demand, or on a scheduled basis:

- Cleanup
- Automailing

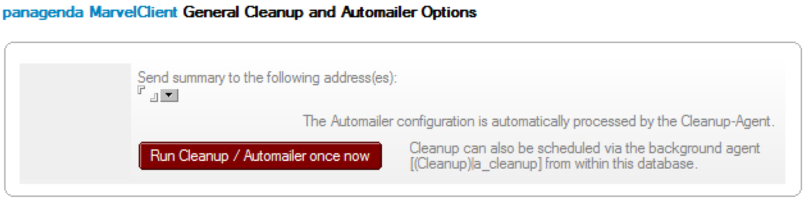
Cleanup allows you to delete User Profile Documents (and any associated detail documents) that have not been updated in a configurable number of days, such as "delete any User Profiles that have not been updated in the last 30 days". Also, Cleanup can make sure that no more than (again, configurable) a certain number of profiles are stored as per IBM Notes user, meaning that if a user logs on to many different machines, all but the last 2 (or more) are automatically removed from the Analyze database.

The Automailer option (also handled by the Cleanup Agent) allows for two things:

1. optionally building the delta between a public addressbook in order to identify who has already uploaded data into the analyze database and who has not
2. optionally taking (1.) a step further to automatically send install emails to users that haven't updated their analyze record over a certain number of days

The following steps guide you through these options and filling in the *Cleanup & more* configuration form/document accordingly:

General Cleanup and Automailer Options:



panagenda MarvelClient General Cleanup and Automailer Options

Send summary to the following address(es):

The Automailer configuration is automatically processed by the Cleanup-Agent.

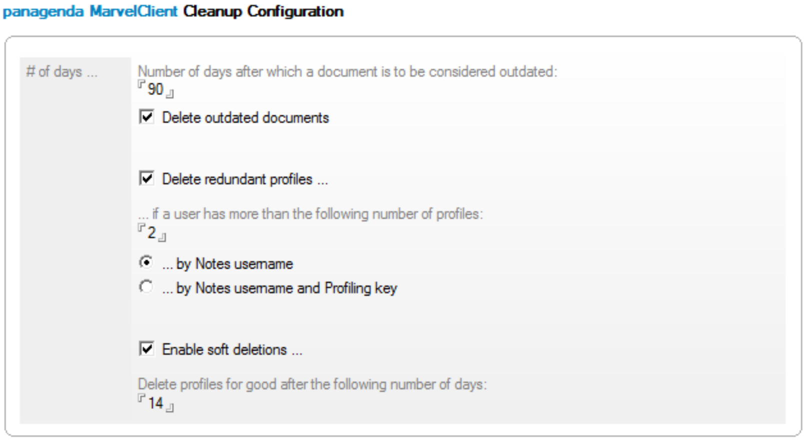
Cleanup can also be scheduled via the background agent [(Cleanup)|a_cleanup] from within this database.

Figure 25: *Cleanup & more* – General Cleanup and Automailer Options

Start filling in the form by specifying one or multiple email addresses to which you want Cleanup and Automailer summaries to be sent.

Once you have completed filling in the entire form/document, the *Run Cleanup/Automailer once now* button allows you to execute Cleanup and/or Automailing on demand. The button was added to the beginning of the form for in case you want to run *Cleanup/Automailer* at a later stage when re-opening a readily filled in configuration.

Cleanup Configuration:



panagenda MarvelClient Cleanup Configuration

# of days ...

Number of days after which a document is to be considered outdated:
90

☑ Delete outdated documents

☑ Delete redundant profiles ...

... if a user has more than the following number of profiles:
2

● ... by Notes username
○ ... by Notes username and Profiling key

☑ Enable soft deletions ...

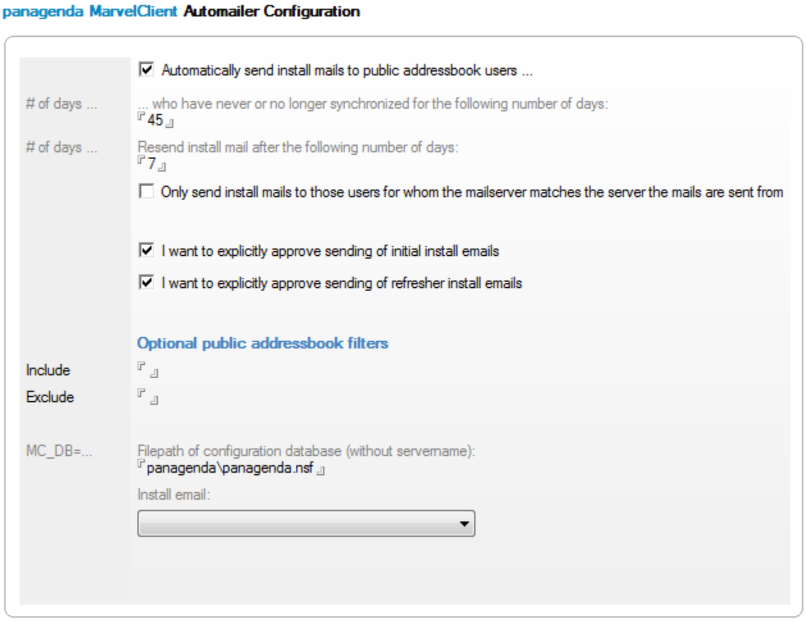
Delete profiles for good after the following number of days:
14

Figure 26: Cleanup & more – Cleanup Configuration

Here you define which documents should be considered outdated (defaults to > 90 days) and whether they should be deleted during Cleanup (if you choose that outdated documents should not be deleted, the Cleanup summary will only tell you how many User Profiles are outdated). Furthermore you can specify whether redundant profiles (more than X as per IBM Notes user or as per IBM Notes User and Profiling Key) should be deleted when the Cleanup Agent runs (this, for instance, is recommended in environments where users log on to several different machines within one or just a few working days).

If you *Enable soft deletions...* you will find the deleted documents under *waiting for deletion* (in the *User* section) until the number of days has elapsed, that are to be specified in the last field of the *Cleanup Configuration* section.

Automailer Configuration:



panagenda MarvelClient **Automailer Configuration**

☑ Automatically send install mails to public addressbook users ...

# of days ... — ... who have never or no longer synchronized for the following number of days: 45

# of days ... — Resend install mail after the following number of days: 7

□ Only send install mails to those users for whom the mailserver matches the server the mails are sent from

☑ I want to explicitly approve sending of initial install emails

☑ I want to explicitly approve sending of refresher install emails

Optional public addressbook filters

Include

Exclude

MC_DB=... — Filepath of configuration database (without servername): panagenda\panagenda.nsf

Install email:

Figure 27: Cleanup & more – Automailer Configuration

Within this configuration section, you can setup the Cleanup Agent to automatically build the delta between the public addressbook and the Analyze database, as to identify which users have already uploaded data into the Analyze database, and which have not (or are no longer updating a once uploaded User Profile after a certain number of days).

Most of the fields in this section should be self-explanatory, with the following exceptions:

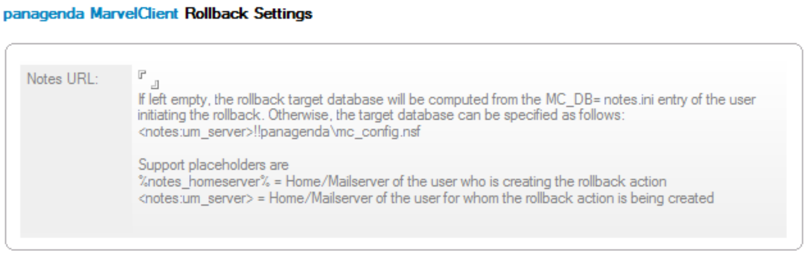
- By default, both "I want to explicitly approve …" checkboxes are checked, meaning that the Automailer component will NEVER send out any emails to end-users, unless so approved in the *Cleanup & More\Automailer* view.
- The Include and Exclude are listfields which support "Like" patterns (for details, see Domino Designer Help: http://tinyurl.com/d2tl2xw. In short, "Like" patterns support wildcards like ? (any one character), # (one digit from 0 to 9), * (any string = zero or more characters), and more)

  For example "test*/panagenda" would include any usernames that start with the word "test" and end with "/panagenda".

  Note that Exclude overrules Include.

- You must specify a valid Install email document in the Automailer configuration, even if the Automailer will not send any emails – this is because at any later point in time you might want to approve sending of such email and a then missing Install email would lead to unexpected (=no) results.

Rollback Settings:



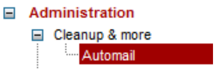
panagenda MarvelClient Rollback Settings

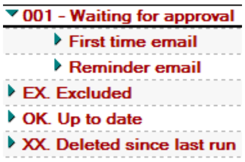
Notes URL: ◸ ◿
If left empty, the rollback target database will be computed from the MC_DB= notes.ini entry of the user initiating the rollback. Otherwise, the target database can be specified as follows:
<notes:um_server>!!panagenda\mc_config.nsf

Support placeholders are
%notes_homeserver% = Home/Mailserver of the user who is creating the rollback action
<notes:um_server> = Home/Mailserver of the user for whom the rollback action is being created

Figure 28: Cleanup & more – Rollback Settings

The last field in the *Cleanup & more* configuration form allows you to optionally specify in which target database optional (Roaming/)Rollback Actions are supposed to be created (these can be created from within a User Profile Document, see "Rollback mode:" on page 75 for more). If this field is left blank, the target Configuration database will always correspond to *<ini:MC_DB>* of the user who is creating the rollback instruction.

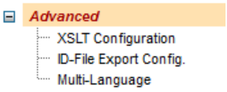




The sub-view ***Automail*** displays the results from the last Automailer run (included with scheduled Cleanup if so enabled). Use the button ***Toggle approval state*** from the action bar, to approve first tim and reminder emails (if applicable and sending of such emails is desired)

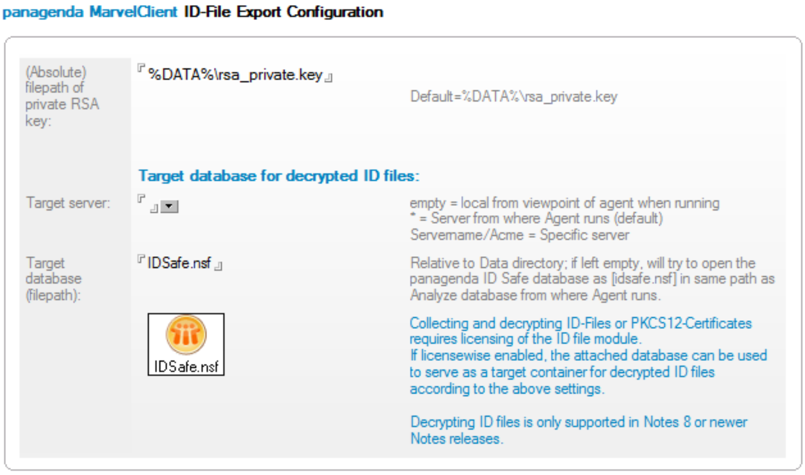
The other views in the *Administration* section of the Navigator allow you to check whether replication conflicts have occurred (***Replication conflicts only***) or list all documents from inside the Analyze database categorized by form (***All by Form (categ.)***). Last but not least, the view ***Licensing details*** shows the state of MarvelClient licenses across your clients, as well as allows you to count total number of unique users across all User Profile Documents uploaded into this Analyze database.

Advanced


Advanced
XSLT Configuration
ID-File Export Config.
Multi-Language

We strongly recommend that you consult panagenda support before modifying any of the ***XSLT Configuration*** documents (support@panagenda.com or your certified panagenda partner).

The ***ID-File Export Config.***[uration] option allows to centrally decrypt user ID files that can optionally be collected with MarvelClient. Note that this requires a special license together with at least MarvelClient Migrate or Manage, and that for security reasons such a license is only available upon request from panagenda.


panagenda MarvelClient **ID-File Export Configuration**

(Absolute) filepath of private RSA key: %DATA%\rsa_private.key — Default=%DATA%\rsa_private.key

Target database for decrypted ID files:

Target server: (empty) — empty = local from viewpoint of agent when running
* = Server from where Agent runs (default)
Servername/Acme = Specific server

Target database (filepath): IDSafe.nsf — Relative to Data directory; if left empty, will try to open the panagenda ID Safe database as [idsafe.nsf] in same path as Analyze database from where Agent runs.

IDSafe.nsf

Collecting and decrypting ID-Files or PKCS12-Certificates requires licensing of the ID file module.
If licensewise enabled, the attached database can be used to serve as a target container for decrypted ID files according to the above settings.

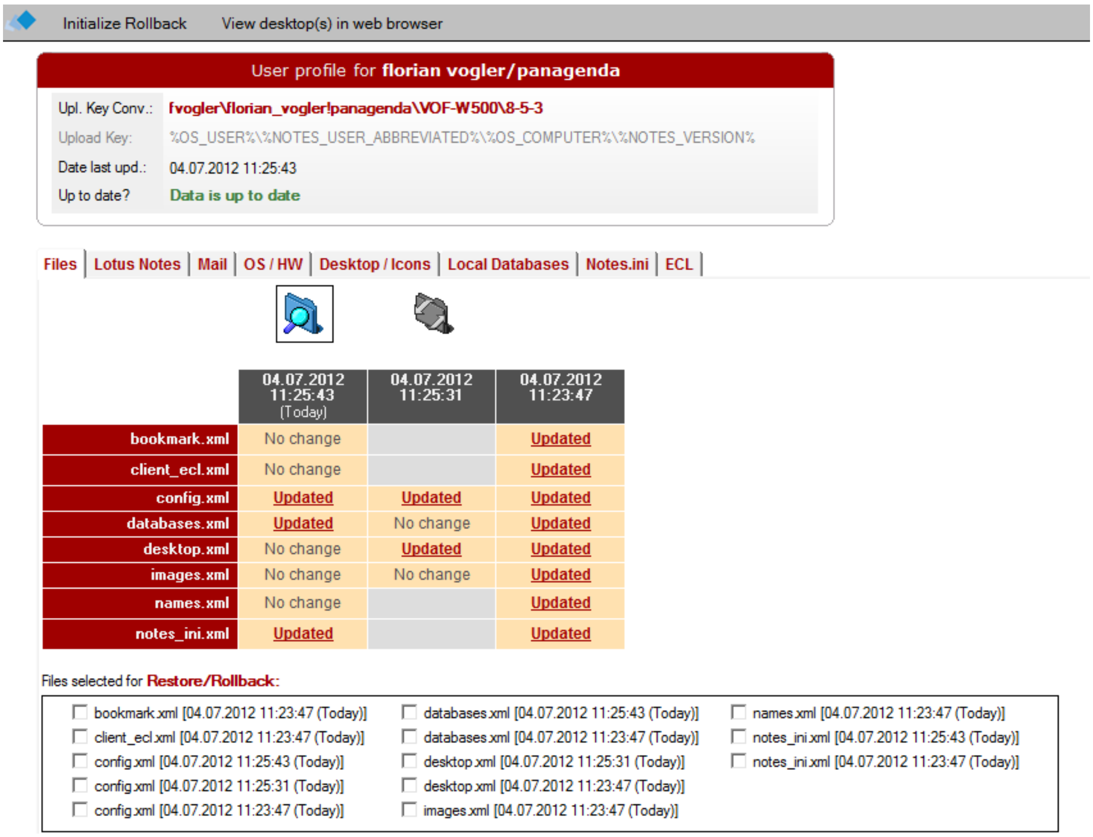
Decrypting ID files is only supported in Notes 8 or newer Notes releases.

Figure 29: ID-File Export Configuration

The ***Multi-Language*** view lists all language documents necessary for selecting different languages in the Analyze database. You should not edit any of these document unless so instructed by panagenda support.

Note that new languages and updates of existing language definitions (same for XSLT documents) are automatically maintained through Online Update.

User Profile Document



User profile for **florian vogler/panagenda**

Upl. Key Conv.:	fvogler\florian_vogler!panagenda\VOF-W500\8-5-3
Upload Key:	%OS_USER%\%NOTES_USER_ABBREVIATED%\%OS_COMPUTER%\%NOTES_VERSION%
Date last upd.:	04.07.2012 11:25:43
Up to date?	Data is up to date

	04.07.2012 11:25:43 (Today)	04.07.2012 11:25:31	04.07.2012 11:23:47
bookmark.xml	No change		Updated
client_ecl.xml	No change		Updated
config.xml	Updated	Updated	Updated
databases.xml	Updated	No change	Updated
desktop.xml	No change	Updated	Updated
images.xml	No change	No change	Updated
names.xml	No change		Updated
notes_ini.xml	Updated		Updated

Files selected for **Restore/Rollback**:

- ☐ bookmark.xml [04.07.2012 11:23:47 (Today)]
- ☐ client_ecl.xml [04.07.2012 11:23:47 (Today)]
- ☐ config.xml [04.07.2012 11:25:43 (Today)]
- ☐ config.xml [04.07.2012 11:25:31 (Today)]
- ☐ config.xml [04.07.2012 11:23:47 (Today)]
- ☐ databases.xml [04.07.2012 11:25:43 (Today)]
- ☐ databases.xml [04.07.2012 11:23:47 (Today)]
- ☐ desktop.xml [04.07.2012 11:25:31 (Today)]
- ☐ desktop.xml [04.07.2012 11:23:47 (Today)]
- ☐ images.xml [04.07.2012 11:23:47 (Today)]
- ☐ names.xml [04.07.2012 11:23:47 (Today)]
- ☐ notes_ini.xml [04.07.2012 11:25:43 (Today)]
- ☐ notes_ini.xml [04.07.2012 11:23:47 (Today)]

Figure 30: User Profile Document – Files

Usually, User Profile Documents are meant to only be used for spot tests, detailed inspections, or to rollback certain user settings (for support calls where users ask for a configuration from yesterday or last week or a different computer, for example). Since you don’t want to inspect all your end-user configurations one by one, the various views in the Analyze database should be used before – if not instead of – looking at any detailed end-user configurations.

Nevertheless, in case you do open a User Profile Document, you will find two buttons in the action bar of every such document: ***Initialize Rollback*** (provided that you have the role [Rollback] in the Analyze database) and ***View desktop(s) in web browser***. Use the first one to create a Roaming/Rollback Action in the Configuration database and the latter one to display the desktop(s) stored in the currently open User Profile Document in a web browser:

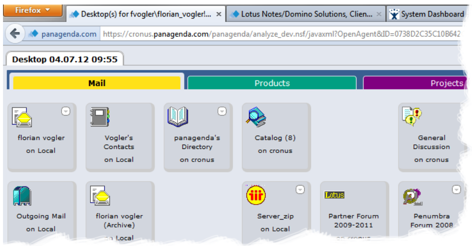

Firefox
Desktop(s) for fvogler\florian_vogler!...
Lotus Notes/Domino Solutions, Clien...
System Dashboard
panagenda.com
https://cronus.panagenda.com/panagenda/analyze_dev.nsf/javaxml?OpenAgent&ID=0738D2C35C10B642
Desktop 04.07.12 09:55
Mail
Products
Projects
florian vogler on Local
Vogler's Contacts on Local
panagenda's Directory on cronus
Catalog (8) on cronus
General Discussion on cronus
Outgoing Mail on Local
florian vogler (Archive) on Local
Server_zip on Local
Partner Forum 2009-2011
Penumbra Forum 2008

Figure 31: Users desktop in web browser

TIP

Desktop icons with titles **colored red** represent invisible ones (such as those created by the IBM Domino Administrator for caching) – these invisible icons also explain why, when adding new icons on the desktop, the client often does not add them to the first free visible space on the desktop, but leaves assumed blank space in between; it's just that the assumed blank space happens to be occupied by such invisible icon(stack)s.

Various general information is displayed in the upper area of a User Profile Document, as well as when it has been updated the last time and whether the displayed ***Data is up to date*** or not (***Displayed data is OUTDATED***); if outdated, audit the document from the view or wait for the scheduled Audit agent to do its' magic.

The next section within each User Profile Document is split up into multiple tabs:

Files

On the *Files* tab you get a clear overview on when which files were last uploaded/updated. The rendering of the file table greatly depends on the configuration of your respective *Backup/Monitoring Actions* (for example when a backup occurs, which files are then uploaded, number of revisions, revision handling).

In general, if a file was changed at a given time, the appropriate cell shows ***Updated***; if the file wasn't updated since the last upload (which is checked via hash codes), it doesn't get transferred again through the network and the appropriate cell shows ***No change***; if it's impossible to check for whether a file has changed (for instance if multiple Backup/Monitoring jobs do not upload the same set of files), the appropriate cell gets marked **grey**.

If you click on an "*Updated*" cell, the corresponding XML file will be opened (with which program depends on your operating system's default settings) – provided that you are in default **File Open mode**. Switching form the default File Open mode to **Rollback mode** and vice versa is done by clicking on the icons above the table: The icon with the folder and the magnifying glass represents the File Open mode, the other one (folder with arrows) represents Rollback mode:

In the *Files* tab you get a clear overview, when which data was uploaded to the Analyze database. How this table looks in detail depends very strongly on the configurations of your respective *Backup/Monitoring* Actions.

In general, if a file was changed at a given time, the appropriate cell shows ***Updated***; if the file wasn't updated since the last upload (which is checked via hash codes), it doesn't get transferred again through the network and the appropriate cell shows ***No change***; if it's impossible to check weather the file has changed (for instance if there several *Backup/ Monitoring* jobs that have uploaded not the same set of files), the appropriate cell(s) get(s) marked **grey**.

Figure 32: Rollback Mode

TIP

It is recommended to open the MarvelClient XML files in **notepad**, **notepad++**, **ultraedit** or similar as opposed to a web browser.

Note that the rollback icon is only visible for users with role [Rollback] in the ACL of the Analyze database.

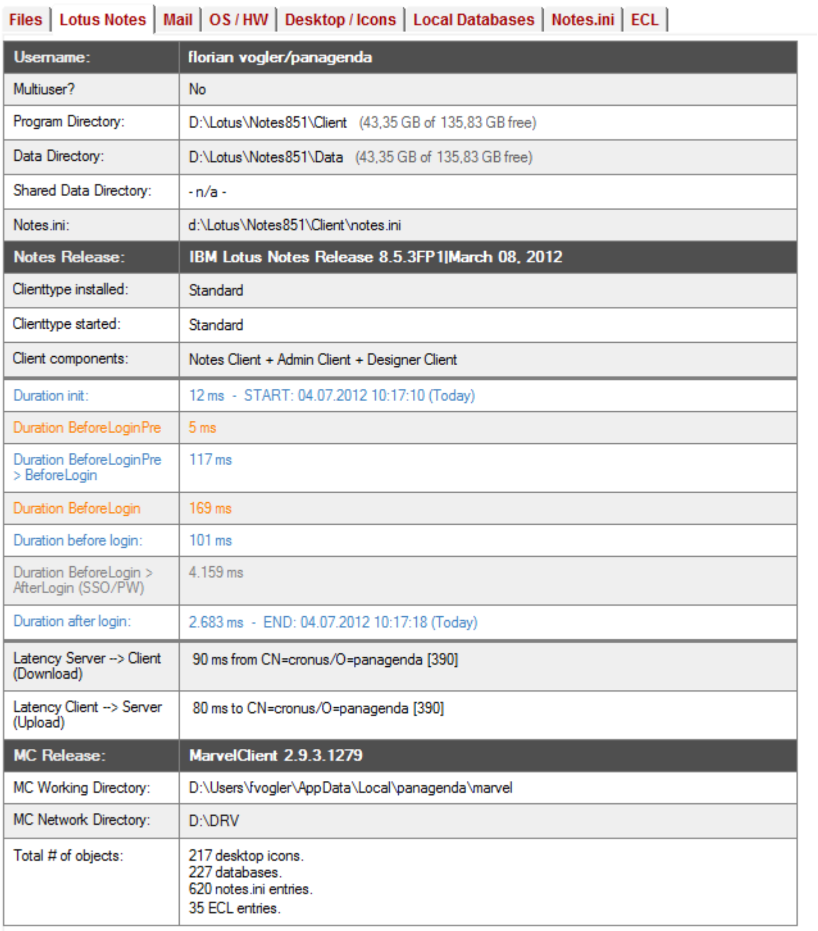
Rollback mode:

Clicking on "*Updated*"-cells in Rollback mode selects the corresponding files for a Roaming/ Rollback Action (requires [Rollback] role in the ACL of the Analyze database). Further down in this document, the (Files selected for Restore/Rollback) list shows which files have been selected for rollback from the above file table (=checked ones).

You MUST NOT select any files in the below list (which is only possible if you put the document in edit mode anyway, and editing the document is something you shouldn't do either). You should only select files from the above file table - otherwise we can't guarantee that the correct files will be rolled back.

Click on the *Initialize Rollback* button when you are done with selecting files, as to generate the appropriate Roaming/Rollback Action in your Configuration database. Please note, that you can only roll back **desktop icons**, **images** and **bookmarks**! Technically, you can also rollback parts of *notes.ini*, but contrary to file-based roaming before login, not all *notes.ini* entries will stick when roaming via the Analyze database (which happens after login). Roaming the Eclipse configuration is possible too, but noticeably increases the backup footprint (=size) in the Analyze database. Roaming of ID files or user personal addressbooks is not possible via the Analyze database.

Lotus Notes

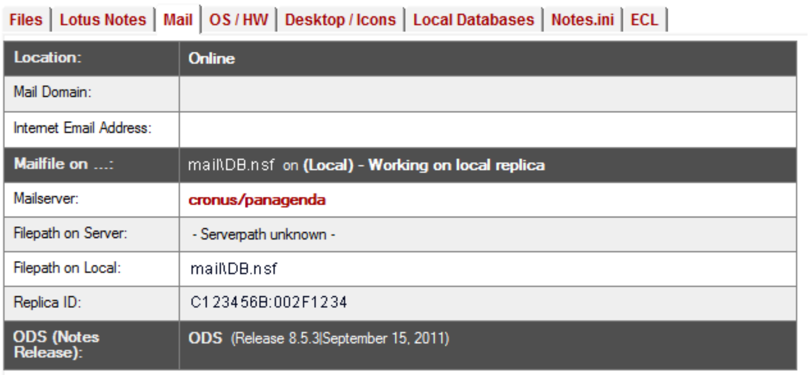


Username:	florian vogler/panagenda
Multiuser?	No
Program Directory:	D:\Lotus\Notes851\Client (43,35 GB of 135,83 GB free)
Data Directory:	D:\Lotus\Notes851\Data (43,35 GB of 135,83 GB free)
Shared Data Directory:	- n/a -
Notes.ini:	d:\Lotus\Notes851\Client\notes.ini
Notes Release:	IBM Lotus Notes Release 8.5.3FP1|March 08, 2012
Clienttype installed:	Standard
Clienttype started:	Standard
Client components:	Notes Client + Admin Client + Designer Client
Duration init:	12 ms - START: 04.07.2012 10:17:10 (Today)
Duration BeforeLoginPre	5 ms
Duration BeforeLoginPre > BeforeLogin	117 ms
Duration BeforeLogin	169 ms
Duration before login:	101 ms
Duration BeforeLogin > AfterLogin (SSO/PW)	4.159 ms
Duration after login:	2.683 ms - END: 04.07.2012 10:17:18 (Today)
Latency Server --> Client (Download)	90 ms from CN=cronus/O=panagenda [390]
Latency Client --> Server (Upload)	80 ms to CN=cronus/O=panagenda [390]
MC Release:	MarvelClient 2.9.3.1279
MC Working Directory:	D:\Users\fvogler\AppData\Local\panagenda\marvel
MC Network Directory:	D:\DRV
Total # of objects:	217 desktop icons. 227 databases. 620 notes.ini entries. 35 ECL entries.

Figure 33: User Profile Document: Lotus Notes Tab

The IBM Notes tab in User Profile Documents displays various details about MarvelClient and the user's IBM Notes client (release, components, directories etc.), including runtime (duration) measurements of various moments during client startup.

Mail



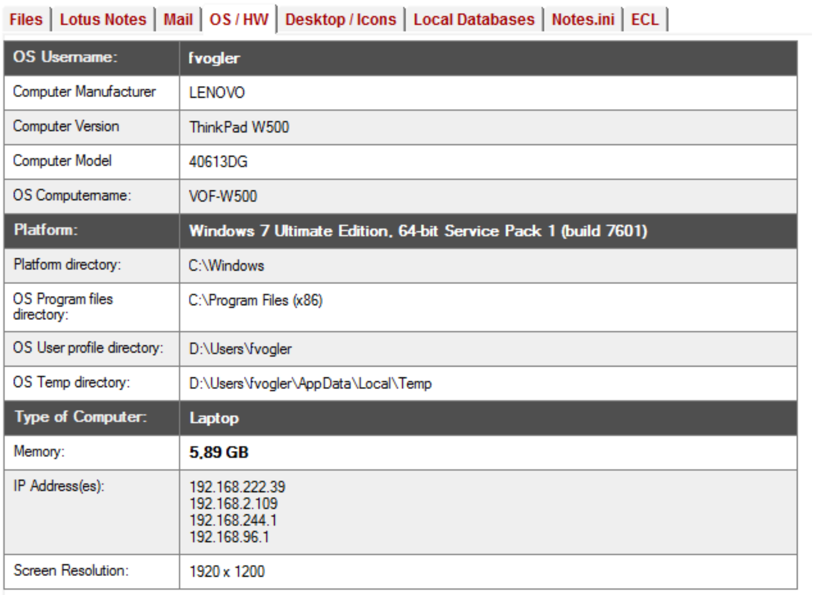
Location:	Online
Mail Domain:	
Internet Email Address:	
Mailfile on ...:	mail\DB.nsf on (Local) - Working on local replica
Mailserver:	cronus/panagenda
Filepath on Server:	- Serverpath unknown -
Filepath on Local:	mail\DB.nsf
Replica ID:	C123456B:002F1234
ODS (Notes Release):	ODS (Release 8.5.3|September 15, 2011)

Figure 34: User Profile Document: Mail Tab

The mail tab holds details about the mail configuration of the currently opened User Profile.

Note that some of the fields require to run at least one Location Action on clients, as Location management triggers a Namelookup towards the user's mailserver in order to lookup the necessary server-side configuration details, which are subsequently uploaded into the Analyze database. It is highly recommended – if not advisable – to execute Location management for your IBM Notes clients anyway, as to prevent a variety of help desk calls (because of an otherwise wrong Mailfile path, Mailserver, or Internet Email Address, for example).

OS/HW



OS Username:	**fvogler**
Computer Manufacturer	LENOVO
Computer Version	ThinkPad W500
Computer Model	40613DG
OS Computername:	VOF-W500
Platform:	**Windows 7 Ultimate Edition, 64-bit Service Pack 1 (build 7601)**
Platform directory:	C:\Windows
OS Program files directory:	C:\Program Files (x86)
OS User profile directory:	D:\Users\fvogler
OS Temp directory:	D:\Users\fvogler\AppData\Local\Temp
Type of Computer:	**Laptop**
Memory:	**5.89 GB**
IP Address(es):	192.168.222.39 192.168.2.109 192.168.244.1 192.168.96.1
Screen Resolution:	1920 x 1200

Figure 35: User Profile Document: OS/HW Tab

The most important details about your end-user's hardware and operating system can be found on the OS/HW tab.

Note that some details shown in above screenshot, such as Computer Version and Computer Model, require the optional HW/SW inventory agent to run on clients (refer to: "HW/SW Inventory" on page 65).

Desktop/Icons – Local Databases – Notes.ini – ECL

The other tabs provide first insight into the actual configuration of desktop icons, local databases, *notes.ini* entries and ECL for the currently open User Profile. Again, please note that these tabs (or any of the User Profile Documents as such) are not meant to continuously analyze single end-user configurations (apart from on demand spot tests). As to save time and reduce efforts you are advised to primarily use the views in the Analyze database to spot areas that require your attention.

Last but not least, at the bottom of each User Profile Document you will see a number of file attachments (for example *rev_4e323c23_desktop.xml.zip*; "4e323c23" is the hash code of the file in this example). Which files and how many are attached to the document depends on the configuration of the corresponding Backup/Monitoring Action(s). The attachments are regularly unzipped by the Audit Agent (if so scheduled) and written into the fields of both the User Profile Document as well as the various detail documents.

CONFIGURATION DATABASE

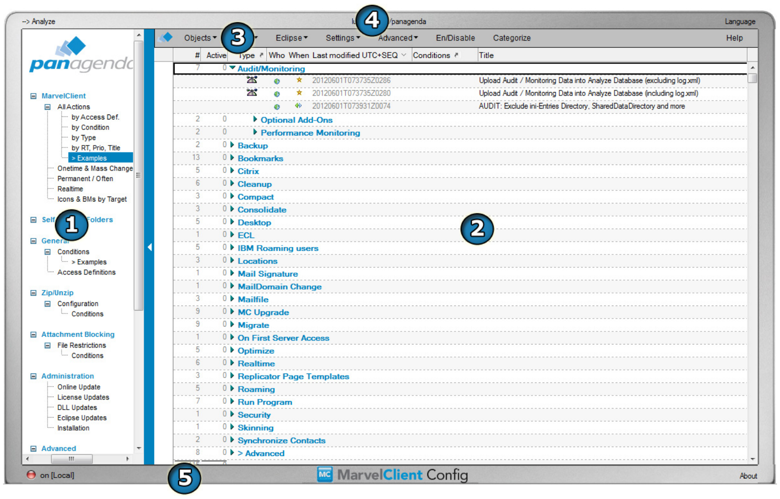
Introduction

This database is the “configuration center” of panagenda MarvelClient: In general, Actions are configured within the Configuration database and the local MarvelClient file (*.dll/.dylib/.so*) communicates with this database to know how to manage your IBM Notes clients. More precisely: When a Notes client is started, the *Download View* (I. Download) is checked for new, changed, deleted or disabled Actions compared to all already locally stored Actions (if any, Actions are stored in the *actions.xml* file in the MarvelClient Working Directory), which are then updated accordingly.

TIP

Within this database, you will also find everything needed to deploy MarvelClient in your environment.

The following screenshot shows the MarvcelClient Configuration database:



#	Active	Type	Who	When	Last modified UTC+SEQ	Conditions	Title
7	0	Audit/Monitoring					
					20120601T073735Z0286		Upload Audit / Monitoring Data into Analyze Database (excluding log.xml)
					20120601T073735Z0280		Upload Audit / Monitoring Data into Analyze Database (including log.xml)
					20120601T073931Z0074		AUDIT: Exclude ini-Entries Directory, SharedDataDirectory and more
2	0	Optional Add-Ons					
2	0	Performance Monitoring					
2	0	Backup					
13	0	Bookmarks					
5	0	Citrix					
6	0	Cleanup					
3	0	Compact					
3	0	Consolidate					
5	0	Desktop					
1	0	ECL					
5	0	IBM Roaming users					
3	0	Locations					
1	0	Mail Signature					
1	0	MailDomain Change					
3	0	Mailfile					
9	0	MC Upgrade					
9	0	Migrate					
1	0	On First Server Access					
5	0	Optimize					
6	0	Realtime					
3	0	Replicator Page Templates					
5	0	Roaming					
7	0	Run Program					
1	0	Security					
1	0	Skinning					
2	0	Synchronize Contacts					
8	0	> Advanced					

Figure 36: Configuration Database – overview

1. The left frame of the Configuration database shows the **navigation bar**
2. The content in the **main area** depends on your current selection in the navigation bar
3. Similarly, the buttons displayed in the **action bar** depend on the currently open view or document within the main area

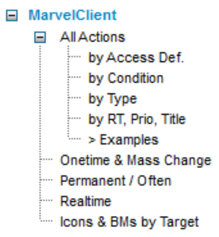
4. In the topmost area of the Configuration database frameset, on the left hand side you can see a link which **opens the Analyze database** (--> *Analyze*), in the middle the **current user** is displayed and on the right hand side you see a button to open the **language selection** (currently English, German are available)

5. In the bottom-most frame, on the left hand side you see whether you are working on **local** (red light + *on [local]*) or on a **server** (green light + *on YourServer/YourCompany*), and on the right hand side you can check which **release of the Configuration database** you are using (*About* button).

Navigation and Configuration

Within the Configuration database there are different sections for configuration, administration and information:

MarvelClient



- ⊟ MarvelClient
  - ⊟ All Actions
    - by Access Def.
    - by Condition
    - by Type
    - by RT, Prio, Title
    - > Examples
  - Onetime & Mass Change
  - Permanent / Often
  - Realtime
  - Icons & BMs by Target

All Actions displays all of the Actions already defined in the Configuration database, categorized according to freely definable categories. The categorization of Actions in this view is optional and does not affect their execution in any way; some customers just like to organize their Actions in custom categories.

The ***by Access Def.***[inition] view shows whether and, if so, which Actions are restricted to (which) users, groups and/or Access Definitions. In essence this view displays which Actions target which groups of users. A **green** category means that the therein contained Actions are RESTRICTED to the respective user(groups), a **red** category means that the therein contained Actions are EXCLUDED from the respective user(groups).

by Condition categorizes all Actions by Condition(s) (= number of Conditions that must apply, and which ones). Conditions, which must all be true, are represented in **dark blue**, whereas Conditions, where only one out of potentially multiple has to be true, are displayed in light blue. The primary category also visualizes whether any "all of n" and "one of m" Condition sets are concatenated through a logical AND or OR.

by Type: In total, there more than 40 different types of Actions with often many further options from within each type of Action (not including Conditions, Access Definitions, Finger Prints, File Properties, File Restrictions and other configuration options).

In the ***by RT, Prio, Title*** view, all Actions are categorized by Run Type, priority and title (for further details regarding Run Types and Runtime order see: "Execution Order and Reference Chains" on page 36).

→ ***Examples***: This view provides a rich variety of example Actions categorized by topics/areas of interest.

TIP

Action Examples: Whatever you want to do with MarvelClient – in many cases you will find a readymade example in this view (if not, drop us a note, please)!

All of the provided example Actions are pre-configured for special business cases and usually only require slight modifications to meet your individual requirements. Just copy and paste an appropriate example Action and it will appear in the *All Actions* view under the respective category; alternatively, you can also use the view action *Examples* → *Create Action(s) from Example(s)* from within the *All Actions* view to copy & paste example Actions into your particular configuration.

Note: Pasted example Actions are <u>disabled by default</u>!

Onetime Actions only displays Actions configured to be executed once, ***Permanent/Often*** displays the opposite (executed more than once or always), and ***Realtime*** shows those Actions, that run in realtime (=Actions, that don't run during client startup, but span the entire Notes session, such as relocate database access to a better server or restrict database operations in realtime).

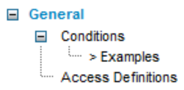
Icons & BMs by Target: Desktop icons and bookmarks are categorized by workspace page or bookmark folder that the respective icon or bookmark is associated to..

Note that this view is of little use without a MarvelClient Manage license

Self-created Folders:

Here, all your self-created folders are listed, if any. If you have not (yet) added any other folders to the Configuration database, then this section remains empty.

General



⊟ General
⊟ Conditions
> Examples
Access Definitions

Within this section, you can manage ***Conditions*** and ***Access Definitions***. Conditions enable you to restrict Actions to, for example, reusable target client or computer characteristics (such as laptop, desktop, IP address range, and many more). Access Definitions allow you to restrict Actions to reusable persons, groups and/or certifiers (for example */Sales/ACME).

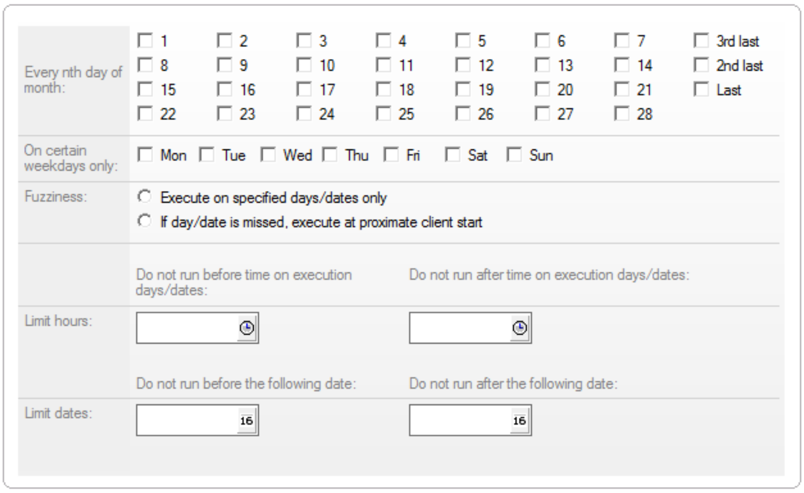
The view ***Conditions → Examples*** lists popular Conditions, which can be copied and pasted just like Example Actions: Simply copy and paste Conditions of your choice, which are then added to the *Conditions* view.

Note that pasted Conditions are always disabled for in case you need to make changes prior to enabling them

Time Condition

New Time Conditions can either be created directly from within the *When* tab of an Action (Note that on the *When* tab the *All- or At least ONE* checkbox must be checked for the corresponding button to be displayed) or via the Action Bar of the Conditions view (in the *General* section of the Navigator). **To save you time, you might also want to check the General → Conditions → Examples view** for whether there is already a ready-made Condition that you can simply copy and paste.

The main setting area of a Time Condition provides the following options:



Every nth day of month:	□ 1	□ 2	□ 3	□ 4	□ 5	□ 6	□ 7	□ 3rd last
	□ 8	□ 9	□ 10	□ 11	□ 12	□ 13	□ 14	□ 2nd last
	□ 15	□ 16	□ 17	□ 18	□ 19	□ 20	□ 21	□ Last
	□ 22	□ 23	□ 24	□ 25	□ 26	□ 27	□ 28	
On certain weekdays only:	□ Mon	□ Tue	□ Wed	□ Thu	□ Fri	□ Sat	□ Sun	

Fuzziness: ○ Execute on specified days/dates only
○ If day/date is missed, execute at proximate client start

	Do not run before time on execution days/dates:	Do not run after time on execution days/dates:
Limit hours:		
	Do not run before the following date:	Do not run after the following date:
Limit dates:		

Figure 37: Time Condition

In the upper area of a Time Condition you must give your document a title, define its scope (*If condition is met* OR *If condition is NOT met*), and specify whether the Condition shall be static (*Compute once only*) or dynamic (*Always compute again*).

Static vs. dynamic conditions are best explained in conjunction with IP Range Conditions (see further below): Let’s assume that you have several Actions associated to a particular IP address range and during execution of those Actions, the client’s network connection is interrupted. A static Condition would only be evaluated once, and all Actions would then run, independent of whether the IP-adress changes. A dynamic Condition would be re-evaluated upon execution of each associated Action and hence if the network connection were interrupted, all Actions until then would run, whereas all other after the connection loss would not. The same principle may apply to Time Conditions: If 10 Actions are tied to not run before 8 am in the morning, dynamic resolution would potentially run a subset of those Actions if the user starts the client at 7:59:55 am and 5 seconds later any left over Actions tied to the same Time Condition would not be executed. Using a static Time Condition would ensure that all 10 Action run as long as the very first one is executed before 8 am in this particular example.

Example: You need a *Time Condition* for an Action that should be executed every Monday

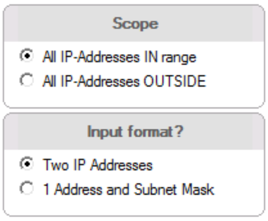
- Give your *Time Condition* a title
- Check *If condition is met* as the *Scope* of the Condition (checked by default)
- Check *Always compute again (=dynamic)* (checked by default); note that if 10 Actions would use this condition and during client startup the weekday would shift from Mon

23:59:59 to Tue 00:00:00, some Actions would execute, others would not – if you'd like all or none of the associated Actions to execute, choose *Compute Once Only (=static)*. Again, you should not choose static in conjunction with the Run Type *Background* (see: "Run Types/Times/Sections" on page 25), if applicable.

- Check *Mon* (*On certain weekdays only:*)
- Select a *Fuzziness* option
- Click on the *Save* button in the Action Bar

IP Range Condition

New IP Range Conditions can either be created directly from within the *When* tab of an Action (Note that on the *When* tab the *All- or At least ONE* checkbox must be checked for the corresponding button to be displayed) or via the Action Bar of the Conditions view (in the *General* section of the Navigator). **To save you time, you might also want to check the *General* → *Conditions* → *Examples* view** for whether there is already a ready-made Condition that you can simply copy and paste.



Scope

● All IP-Addresses IN range
○ All IP-Addresses OUTSIDE

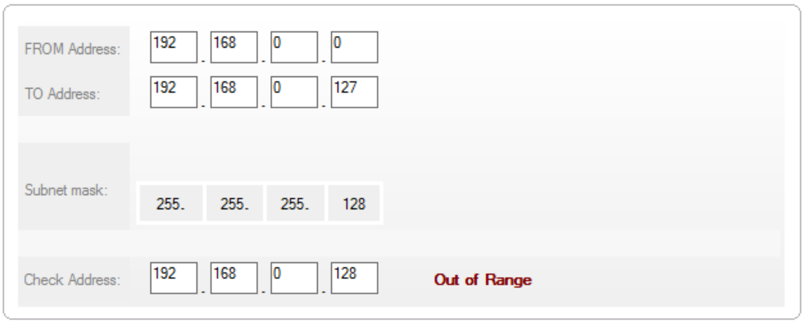
Input format?

● Two IP Addresses
○ 1 Address and Subnet Mask

IP Range Conditions come in handy to distinguish between, for example, LAN-, WAN- or VPN-connections, different cities, countries or offices.
When defining an IP-Address range you can choose between two different input formats:

When configuring an IP segment using **two IP addresses**, the specified address range must be compatible with the subnet masks. A valid address range is, for example 10.0.0.0 to 10.0.0.127, whereas 10.0.0.0 to 10.0.0.128 is invalid. The latter will trigger an appropriate warning.

The field *Check Address* allows you to enter one IP address in order to verify whether it is covered by the specified segment:



FROM Address: 192 . 168 . 0 . 0

TO Address: 192 . 168 . 0 . 127

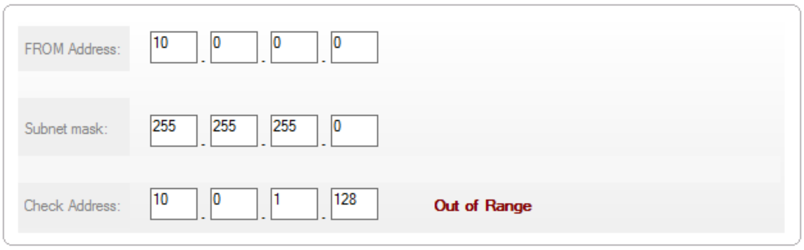
Subnet mask: 255. 255. 255. 128

Check Address: 192 . 168 . 0 . 128 Out of Range

Figure 38: IP Range Condition – FROM TO IP

Note that the <u>Check Address</u> field does not at all affect the Condition as such, but is there for your convenience to quickly check whether a "test" address is (or is not) within the specified range.

Alternatively, an IP range/segment can also be defined by specifying an IP address and a subnet mask:



FROM Address: 10 . 0 . 0 . 0

Subnet mask: 255 . 255 . 255 . 0

Check Address: 10 . 0 . 1 . 128 Out of Range

Figure 39: IP Range Condition – FROM IP + Subnet Mask

String Condition

New String Conditions can either be created directly from within the *When* tab of an Action (Note that on the *When* tab the *All- or At least ONE-checkbox* must be checked for the corresponding button to be displayed) or via the Action Bar of the Conditions view (in the *General* section of the Navigator). **To save you time, you might also want to check the *General* → *Conditions* → *Examples* view** for whether there is already a ready-made Condition that you can simply copy and paste.

In the upper area of the document, you must give your document a title, define the scope (*If condition is met* OR *If condition is NOT met*) and whether the Condition shall be static

(*Compute once only*) or dynamic (*Always compute again*) – for details on static vs. dynamic Conditions, see "Time Condition" on page 83.

A string condition is defined through the fields ***Condition*** (string), ***Result*** (string or regular expression) and ***Match Case*** (exact or not). By using *References* ("References – Syntax" on page 33), String Conditions provide various easy and powerful options to query clients for whether or not they meet certain criteria:

Example:

- Condition: ***<ini:Location>*** / Result: ***^office \(network\)$*** / Match exact case: **Not checked**
  This will match "Office (Network) as well as "oFfiCe (nEtWORk) or any other casing of the string "office (network).
  "Office" or "Office – New York" will not match

- Condition: ***<ini:Location>*** / Result: ***^Office.*$*** / Match exact case: **Checked**
  This will match any Location name beginning (^) with the string "Office" and any other characters thereafter, if any.
  Due to match exact case checked, "office …" would not match.

- In a regular expression, "^" means "begins with", "$" means "ends with". The following characters must be escaped (= they must be prefixed with a "\" backslash) if they are to be literally compared: **\, *, +, ?, |, {, [, (,), ^, $, ., #**. This explains, why "\(Network\)" instead of just "(Network)" was specified in the first example.

- Also note that the result of a string comparison must always be a complete match. Partial comparisons therefore have to make use of ".*" ("dot asterisk") to indicate any unknown string components before or after the string you are searching for, depending on what you are searching for.

  - ***^Office.*$*** = anything that starts with the word "Office" (as well as "Office" only)

  - ***^.*Office$*** = anything that ends with the word "Office" (as well as "Office" only)

  - ***^.*Office.*$*** = anything that contains the word "Office" (as well as "Office" only)

Last but not least, the result field supports **Perl regular expressions** (for further information see http://www.regular-expressions.info/quickstart.html)

Numeric Condition

New Numeric Conditions can either be created directly from within the *When* tab of an Action (Note that on the *When* tab the *All- or At least ONE* checkbox must be checked for the corresponding button to be displayed) or via the action bar of the Conditions view (in the *General* section of the Navigator). **To save you time, you might also want to check the *General* → *Conditions* → *Examples* view** for whether there is already a ready-made Condition that you can simply copy and paste.

In the upper area you must give your Numeric Condition a title, define the scope (*If condition is met* OR *If condition is NOT met*) and specify whether the Condition shall be static (*Compute once only*) or dynamic (*Always compute again*).

A Numeric Condition is defined through the fields ***Condition*** (string with numerical result that is to be compared with *Result*), ***Comparison*** (selection list) and ***Result*** (number). As with String Conditions, the fact that you can use *References* ("References – Syntax" on page 33) allows to query clients for a rich variety of criteria, for example:

- *Condition*: ***<os:memory_physical_available_mb>*** / *Comparison*: **>=** (greater than or equal to) / *Result*: **1024**
  Checks for wheter a client has at least 1 GB of physical memory

- Condition: ***<os:disk_data_free_mb>*** / Comparison: **>=** (greater than or equal to) / Result: **2048**
  Checks for whether a client has at least 2 GB of free disk space in its data directory

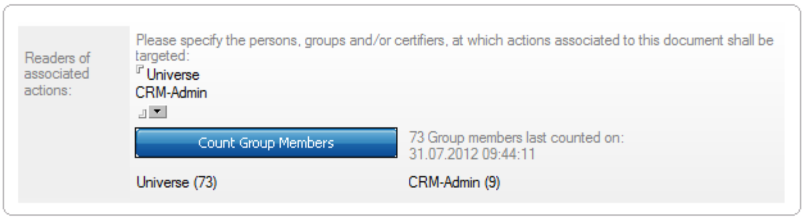
- Condition: ***<notes:version_number>*** / Comparison: **>=** (greater than or equal to) / Result: **8050302**
  Checks for whether a IBM Notes client is at least 8.5.3FP2

Access Definition

New Access Definitions can either be created directly from within the *When* tab of an Action (note that on the *When* tab the *All- or At least ONE* checkbox must be checked for the corresponding button to be displayed) or via the action bar of the *Access Definitions* view (in the *General* section of the Navigator).

In the upper area, you can find the mandatory *Title* field. In the main configuration area, you can enter one or multiple individuals (person names), groups, and/or certifiers in the field Readers of associated Actions.

Note that any changes to existing Access Definitions are automatically bequeathed into all associated Actions.



Readers of associated actions: Please specify the persons, groups and/or certifiers, at which actions associated to this document shall be targeted:
Universe
CRM-Admin

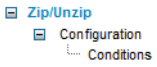
73 Group members last counted on: 31.07.2012 09:44:11

Universe (73) CRM-Admin (9)

Figure 40: Access Definition

The button *Count Group Members* is for informational purposes only, showing you how many users are member of any specified groups (this includes resolving of nested groups) when clicking the button. This helps to anticipate how many users will be affected by (respectively will be excluded from) an Action using said group or Access Definition. As you can see in above example screenshot, all members of “CRM-Admin” (9) are also members of “Universe” (73), totaling in 73 members.

Zip/Unzip



The *Zip/Unzip* → ***Configuration*** view lists all the Actions that are related to the configuration of MarvelClient Zip/Unzip, such as global or database specific Zip/Unzip settings.

The sub-view ***Conditions*** displays all of your existing Database Scope Conditions, if any. In all Zip/Unzip views, the action bar allows you to create Zip/Unzip related Actions and Conditions respectively.

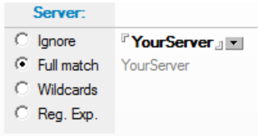
Database Scope Condition

Database Scope Conditions allow you to define a (set of) database(s), to be further associated to database specific Zip/Unzip (and Attachment Blocking) settings.

When creating a Database Scope Condition, specify its respective *Scope* first:

- *If condition is met* = database(s) that match the Database Scope specified below in this Condition
- *If condition is NOT met* = any databases that do not match the Database Scope specified below in this Condition

The Database Scope is defined by selecting one or more of the criteria ***Server***, ***Directory***, ***Filename***, ***Replica-ID*** and ***Template name***.



Server:

○ Ignore
● Full match
○ Wildcards
○ Reg. Exp.

YourServer
YourServer

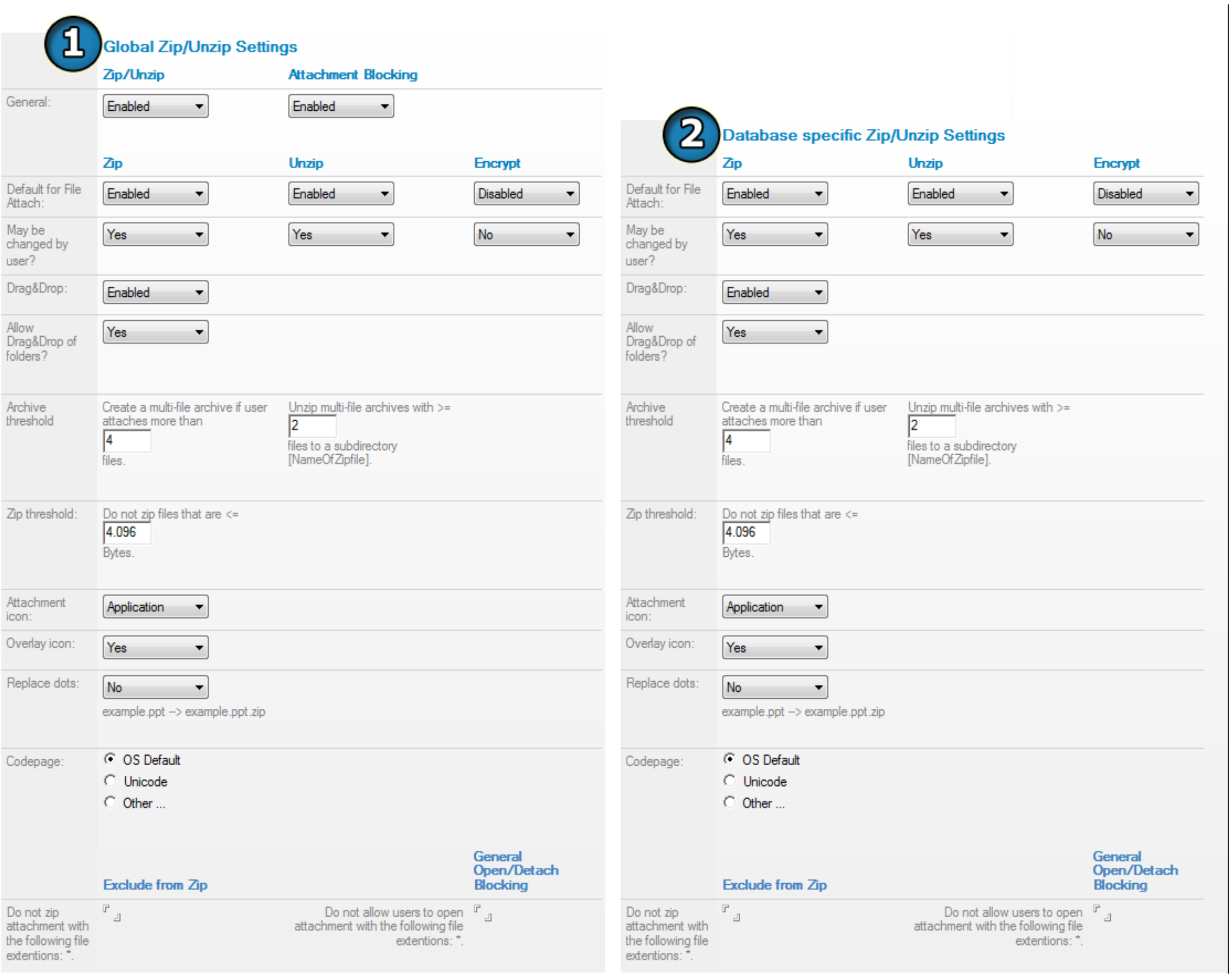
Apart from *Replica-ID*, where you only have the options *Ignore* and *Full match*, all other criteria allows you to either *Ignore* the value or perform a *Full match*, *Wildcard*, or *Reg.*[ular] *Exp.*[ression] comparison.

Wildcards allow for the use of "*" and "?", where "*" means 0-n characters, "?" means 1 character, for example:

- "**Server0?/ACME**" would match "Server01/ACME", "Server02/ACME", "Server0A/ACME", "Sever0X/ACME", etc., but not "Server011/ACME" or "Server01A/ACME"
- "**Server0*/ACME**" however would well match all of what "Server0?/ACME" matches, as well as "Server011/ACME" and "Server 01A/ACME"

Zip/Unzip Settings (Setting Action)

Database specific Zip/Unzip Settings make use of Database Scope Conditions to let you define the behavior of Zip/Unzip based on specific databases. If the conditions are not met for at least one DB-Specific Settings the behavior will default to the **Global Zip/Unzip Settings**. The Global Zip/Unzip Settings (1) and the Database specific Zip/Unzip Settings (2) forms are quite similar:



Global Zip/Unzip Settings

	Zip/Unzip	Attachment Blocking	
General:	Enabled	Enabled	
	Zip	**Unzip**	**Encrypt**
Default for File Attach:	Enabled	Enabled	Disabled
May be changed by user?	Yes	Yes	No
Drag&Drop:	Enabled		
Allow Drag&Drop of folders?	Yes		
Archive threshold	Create a multi-file archive if user attaches more than 4 files.	Unzip multi-file archives with >= 2 files to a subdirectory [NameOfZipfile].	
Zip threshold:	Do not zip files that are <= 4.096 Bytes.		
Attachment icon:	Application		
Overlay icon:	Yes		
Replace dots:	No example.ppt --> example.ppt.zip		
Codepage:	● OS Default ○ Unicode ○ Other ...		
	Exclude from Zip		**General Open/Detach Blocking**
Do not zip attachment with the following file extentions: *.		Do not allow users to open attachment with the following file extentions: *.	

Database specific Zip/Unzip Settings

	Zip	Unzip	Encrypt
Default for File Attach:	Enabled	Enabled	Disabled
May be changed by user?	Yes	Yes	No
Drag&Drop:	Enabled		
Allow Drag&Drop of folders?	Yes		
Archive threshold	Create a multi-file archive if user attaches more than 4 files.	Unzip multi-file archives with >= 2 files to a subdirectory [NameOfZipfile].	
Zip threshold:	Do not zip files that are <= 4.096 Bytes.		
Attachment icon:	Application		
Overlay icon:	Yes		
Replace dots:	No example.ppt --> example.ppt.zip		
Codepage:	● OS Default ○ Unicode ○ Other ...		
	Exclude from Zip		**General Open/Detach Blocking**
Do not zip attachment with the following file extentions: *.		Do not allow users to open attachment with the following file extentions: *.	

Figure 41: Zip/Unzip Settings – Global and Database Specific

Specify the settings of your choice, give the document a title and don't forget to set up and select a **Database Scope Condition** for Database specific Zip/Unzip Settings on the *When* tab of the form – then, save the Settings document.

Details about Zip and Encryption:

The Zip/Unzip module is compatible with the de facto standard defined by PKWARE's ZIP Application Note (Version 6.3.3, see: http://www.pkware.com/documents/casestudies/APPNOTE.TXT).

The used encryption depends on the selection in the attachment file dialog:

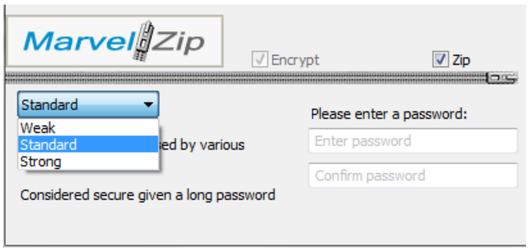


Figure 42: Zip/Unzip Encryption – Attachment File Dialog

- ***Weak*** – Also known as "Standard Zip 2.0 encryption" or "Password-Protection". See Chapter 6.1 of PKZIP AppNote.txt specification (see: http://www.pkware.com/documents/casestudies/APPNOTE.TXT).
  Description in attachment file dialog: "Default password protection for old zip archives. Not recommended (unsecure)"
  - Vulnerable to known-plaintext attacks
    (see: http://math.ucr.edu/~mike/zipattacks.pdf)

- ***Standard*** and ***Strong*** – Uses Advances Encryption Standard (=AES, see: http://en.wikipedia.org/wiki/Advanced_Encryption_Standard). 128-bit for *Standard*, 256-bit if *Strong* is selected.

  Description in attachment file dialog:
  - *Standard*: "Standard protection used by various programs like *WinZIP*. Considered secure given a long password"

  - *Strong*: "Strong encryption. Recipient needs MarvelZIP to unzip this archive"

***Description for Strong is misleading** and will be updated - most tools (aside form Windows Explorer that only supports password protected zip files) nowadays will handle the encryption method used by MarvelZip just fine.*

Attachment Blocking

⊟ Attachment Blocking
⊟ File Restrictions
Conditions

Within these views you will find everything needed to set up your custom Attachment Blocking configuration. The view ***File Restrictions*** lists all such configurations for black- or whitelisting of when users add file attachments to IBM Notes documents.

For example, you can ensure that users do not attach files larger than a certain size, or that they may not attach any **.exe* and **.dll* files, or even create combined blocking rules, such as "no video files larger than 2 MB". Also, you can limit such restrictions to certain databases only, such as only mailfiles, all or files that belong to the CRM system, or similar. This allows you to flexibly create custom File Restrictions that match your different applications and respective requirements.

TIP

The action bar allows you to create new File Restrictions (Attachment Blocking Action). Within File Restrictions you are provided the option to **combine different file blocking Conditions**.

In the ***Conditions*** view, you find a list of all existing file blocking Conditions and corresponding **action bar** menu options to create your own Conditions: Filesize Definitions, Filename Patterns, Fingerprints (=tamper proof digital file patterns for filetype-detection; i.e. an **.exe* file, which was renamed to **.txt* will still be reliably detected as an **.exe* file) and Database Scope Conditions (see: "Database Scope Condition" on page 90) to limit File Restrictions to a certain (set of) database(s).

Filesize Definitions

Filesize Definitions are used to prevent too large (or too small) files from being attached to documents, such as emails or teamrooms for example.

Select one of the following scopes

- *If GREATER than...*
- *If smaller than...*

and specify the desired file size (in bytes).

Filename Patterns

Filename Patterns are used to prevent certain filenames from being attached to documents, such as emails, for example.

Select one of the following scopes

- *If filename MATCHes*
- *If filename does NOT match*

and specify one or multiple file patterns (using regular expression), such as **^(.*)\.exe$**.

Note that if you enter multiple patterns in one such Condition, only ONE of your entered filename patterns must match for the Condition to be met in an associated File Restriction.

Fingerprints

Fingerprints are used to prevent certain types of files from being attached to documents, such as emails – even if they are renamed (for example from *file.exe to file.txt*), fingerprints ensure proper detection of the respective file type.

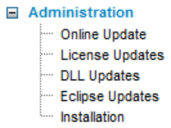
Note that Fingerprints are also known as "Linux Magic Numbers" and there is a vivid community out there that regularly adds new fingerprints on various websites.

Select one of the following scopes

- *If fingerprint matches*
- *If fingerprint does NOT match*

Then specify the fingerprint pattern.

Administration



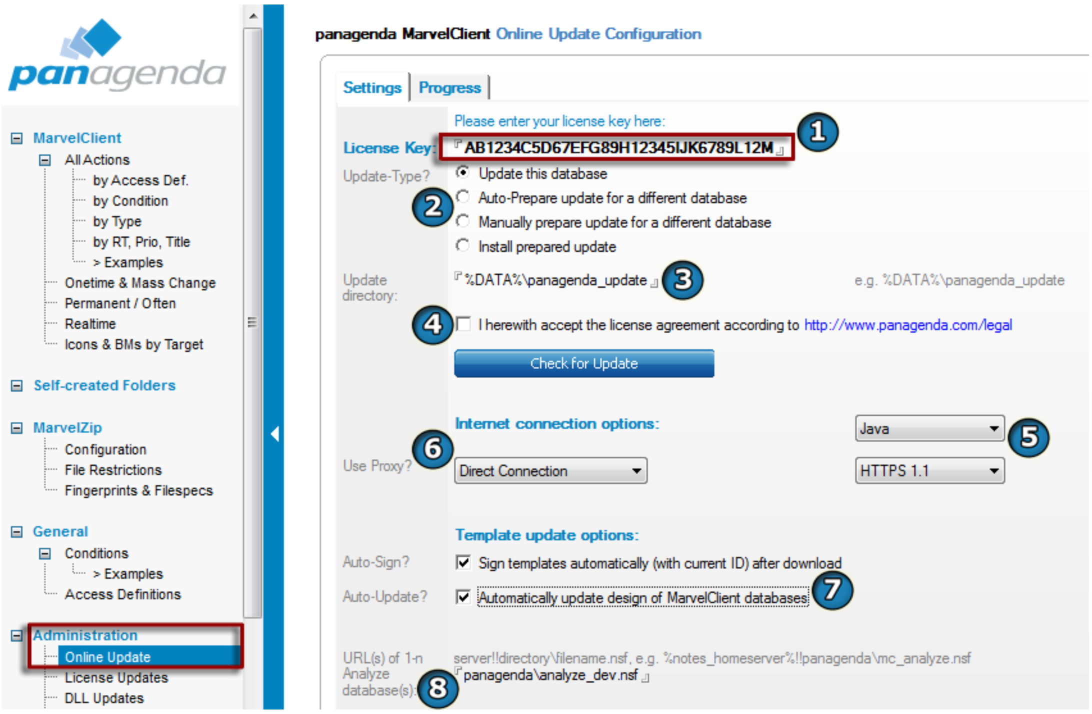
The *Administration* section provides documents and views for installation, deployment and updating of your MarvelClient environment.

Note that if you don't have the role [AOnlineUpdate] you won't be able to open the Online Update document.

panagenda MarvelClient Online Update Configuration

Use this document to perform various updates of the MarvelClient databases.

(Online) Update this database:



panagenda MarvelClient Online Update Configuration

Please enter your license key here:

License Key: AB1234C5D67EFG89H12345IJK6789L12M

Update-Type?
● Update this database
○ Auto-Prepare update for a different database
○ Manually prepare update for a different database
○ Install prepared update

Update directory: %DATA%\panagenda_update e.g. %DATA%\panagenda_update

□ I herewith accept the license agreement according to http://www.panagenda.com/legal

Internet connection options:

Use Proxy? Direct Connection Java HTTPS 1.1

Template update options:

Auto-Sign? ☑ Sign templates automatically (with current ID) after download

Auto-Update? ☑ Automatically update design of MarvelClient databases

URL(s) of 1-n Analyze database(s): server!!directory\filename.nsf, e.g. %notes_homeserver%!!panagenda\mc_analyze.nsf
panagenda\analyze_dev.nsf

Figure 43: Online Update Configuration – *Update this database*

1. Please enter your 32 character license key
2. Select *Update this database*
3. You may adjust the update directory if desired or leave the default which is *%DATA%\panagenda_update*. The update directory is usually only important if you prepare updates for different Configuration databases
4. Please read and accept the license agreement (http://www.panagenda.com/legal)

5. The standard protocol version is HTTP 1.0, in some cases your Proxy server may require an HTTP 1.1, HTTPS 1.0 or HTTPS 1.1 connection. An Online Update of type Java is required for customers in Asia. All other customers can select between LotusScript (=recommended) or Java based Online Update. A Java based Online Update is usually faster, whereas a LotusScript based Online Update allows to establish the update connection based on system settings

6. Choose Direct connection, proxy connection or connection according to system settings (LotusScript based Online Update only). When selecting Proxy connection please make sure that you enter the Proxy server including the port number, such as "MyProxy.MyDomain.com:3128"

7. The update process can also use the Notes ID currently in use to automatically sign design elements after a design update, which is recommended only if an administrator or signing ID is used to perform the online update. For existing installations, the Online Update can not only download new templates but also update the design of your existing databases automatically

8. Specify the path to one or multiple Analyze database(s), such as *%notes_homeserver%!!panagenda\mc_analyze.nsf or MyHubserver!!admin\analyze.nsf.*

Note that during an Online Update, %notes_homeserver% is automatically replaced with the current mail server of the user running the update. The path of the Analyze database is needed to check whether the design of it is up to date. If you have multiple Analyze databases that do NOT replicate, please specify multiple URLs. If your Analyze databases do replicate, then you should only specify ONE master replica.

- **Finally click on *Check for Update.***

If you did not select for the Configuration database to be signed in step 7 above, then, as soon as Online Update has completed, please open your IBM Domino Administrator, navigate to the location of your MarvelClient databases on the Files tab and right-click on them and select "Sign". Then select your desired signing ID and sign all design documents. Ideal signing IDs are IDs which have the right to run Agents on the server (also see "(Audit All) Agent" on page 13 and the information about the Cleanup Agent under "Administration" on page 66)

Prepare and Install Update:

Another option of updating your MarvelClient databases is to perform an update with locally stored (auto-prepared) update package. This may be useful if the client running the Online Update may not access the Configuration database at the same time (aka you are not allowed to access the internet from the environment where you are allowed access to the MarvelClient database). In this case you can **auto-prepare** an update package on a client that has a connection to the internet. Enter your license key, select *Auto-Prepare update for a different database* and follow the steps described above ("(Online) Update this database:" on page 95 – from step 3 on; ignore step 7). When you are done with setting everything up, click on *Check for Update.*

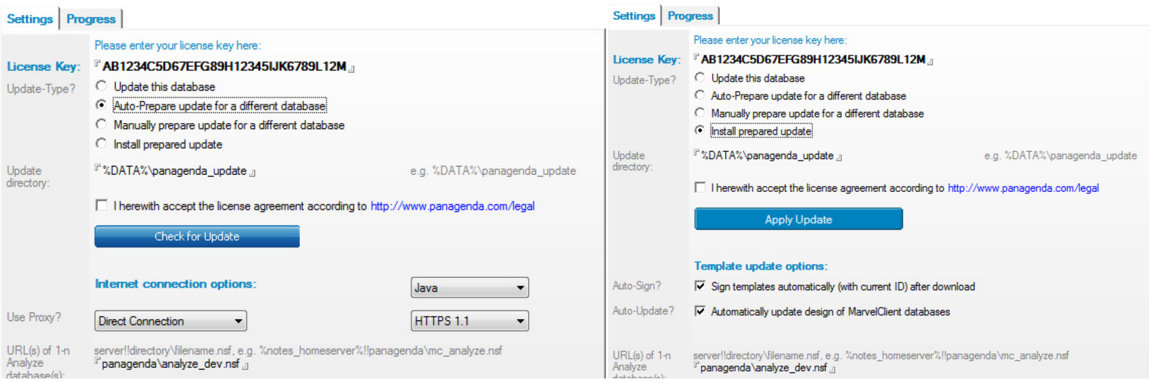
Alternatively you can **manually** prepare the update package using the download files and following the instructions from: http://update.panagenda.com/pub/panaweb.nsf/getlicenseinfo?openagent&key=***YourLicenseKey***

You can go there faster by selecting *Manually prepare update for a different database,* accepting the license agreement and clicking on the *GO!* button – provided that the *License Key* field is filled in correctly.

Note: There is no need to download the license file for existing installations unless there has been a change in license since the last Online Update.

Using the files from the *Update directory* (whether **auto-prepared** or downloaded **manually** – if downloaded manually, please make sure that you unzip(!) all such downloaded files!), you can then perform updates in environments without internet connection: Copy the files to a client in the closed environment, open the Online Update document in the respective Configuration database, select ***Install prepared update*** and point the *Update directory* to the directory with the prepared update files. After that follow the steps 4, 7 and 8 described above ("(Online) Update this database:" on page 95).

- **Finally click on Apply Update.**



Please enter your license key here:

License Key: AB1234C5D67EFG89H12345IJK6789L12M

Update-Type?
○ Update this database
● Auto-Prepare update for a different database
○ Manually prepare update for a different database
○ Install prepared update

Update directory: %DATA%\panagenda_update   e.g. %DATA%\panagenda_update

□ I herewith accept the license agreement according to http://www.panagenda.com/legal

Internet connection options: Java

Use Proxy? Direct Connection   HTTPS 1.1

URL(s) of 1-n Analyze database(s): server!!directory\filename.nsf, e.g. %notes_homeserver%!!panagenda\mc_analyze.nsf
panagenda\analyze_dev.nsf

Please enter your license key here:

License Key: AB1234C5D67EFG89H12345IJK6789L12M

Update-Type?
○ Update this database
○ Auto-Prepare update for a different database
○ Manually prepare update for a different database
● Install prepared update

Update directory: %DATA%\panagenda_update   e.g. %DATA%\panagenda_update

□ I herewith accept the license agreement according to http://www.panagenda.com/legal

Template update options:

Auto-Sign? ☑ Sign templates automatically (with current ID) after download

Auto-Update? ☑ Automatically update design of MarvelClient databases

URL(s) of 1-n Analyze database(s): server!!directory\filename.nsf, e.g. %notes_homeserver%!!panagenda\mc_analyze.nsf
panagenda\analyze_dev.nsf

Figure 44: Online Update Configuration – Auto-Prepare for a Different Database and Install

Don't forget to sign your databases if not already done in step 7!

Continuing with the view descriptions of the *Administration* section:

The view ***License Updates*** displays your MarvelClient license document(s).

Check this view after each (Online) Update to ensure that the latest document is enabled.

During initial setup, your first downloaded license document is automatically enabled. In case Online Update detects a license update during a subsequent update, your new license is not automatically enabled, since we would then have to also automatically disable your current license. As you may want to test or review the new license document first, Online Update does not change your existing licensing configuration.

Note: the Last modified UTC+SEQ column shows the last modified date/time for each license document, which is how you can easily spot the most current license.

DLL Updates shows a list of all your licensed local MarvelClient files (*.dll*/*.dylib*/*.so*), ***Eclipse Updates*** shows a list of your MarvelClient Eclipse plugin versions (if applicable).

TIP

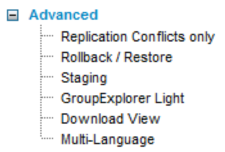
IMPORTANT: After updating existing MarvelClient installations, please also check these "Update" views as, similar to how Online Update handles license updates, Online Update does not automatically enable any newly downloaded MarvelClient files or Eclipse plugins.

Note, that enabling MarvelClient files only makes sense if some or all of your end-users have write permissions in the respective directory of where MarvelClient was initially deployed. Existing MarvelClient releases will update themselves with enabled MarvelClient updates from the Configuration database, provided that the directory where the current MarvelClient (.dll/.dylib/.so) file resides can be written to by the then current operating system user.

Also note that replacing an existing MarvelClient file requires two IBM Notes client restarts, as during first restart, the new (.dll/.dylib/.so) file is downloaded (with the "old" one still running), and upon the second client restart the new file is then launched.

The view ***Installation*** lists all documents (if any) used for installing MarvelClient by email or database PostOpen scripts. Use the *New Install* button from the **action bar** to create a new install document (see: "Deploying MarvelClient on Clients from the Configuration DB" on page 17).

Advanced



Advanced
- Replication Conflicts only
- Rollback / Restore
- Staging
- GroupExplorer Light
- Download View
- Multi-Language

The ***Replication Conflicts only*** view shows whether the Configuration database holds any replication or save conflict documents.

Rollback/Restore (requires role [Rollback]) shows a list of *Roaming/Rollback* Actions, categorized by *0 - Waiting for processing* (waiting to be processed by MarvelClient) and *1 - Processed/done* (processed and completed by MarvelClient).

Under ***Staging*** (requires role [Staging]) you see a list of Actions configured for staging, categorized by *0 - Waiting for processing* (waiting/"staged"), and *1 - Processed* (i.e. consolidated in one "compressed collection Action") – please contact support@panagenda.com for further information about *Staging*; please note that staging is only useful if you need to automatically feed the Configuration database with hundreds to thousands of configuration instructions as opposed to creating such many Actions manually.

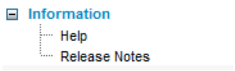
Please refer to the GroupExplorer Administrator's Guide for any further information about ***GroupExplorer Light***.

The ***Download View*** serves the synchronization of the client component of MarvelClient with the Configuration database, for example the downloading of new, modified, disabled or deleted Actions and Conditions.

IMPORTANT: You should not delete or edit any documents in this view unless you really know what you are doing!

Multi-Language contains all language documents the application uses. You should not edit or update (let alone delete) any documents in this view unless so instructed by panagenda support. Also note that language packs are automatically maintained by Online Update.

Information



Information
- Help
- Release Notes

Under ***Help*** you can find selected help documents and ***Release Notes*** hold important releases for MarvelClient updates.

The Action Bar

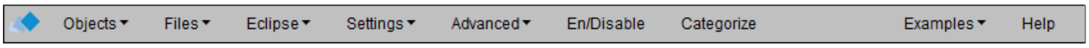


Figure 45: Action Bar – MarvelClient → All Actions

The Actions available from within the Action bar vary from view to view and document to document – many of them were already described above. Whilst the *All Actions* views come with an **action bar** that provides options to create new MarvelClient Actions, the Action Bar may look different from view to view (the *Realtime* view, for example, only shows Realtime related Actions).

Most of the views in the MarvelClient section of the Navigator show the following menu options or a subset of it:

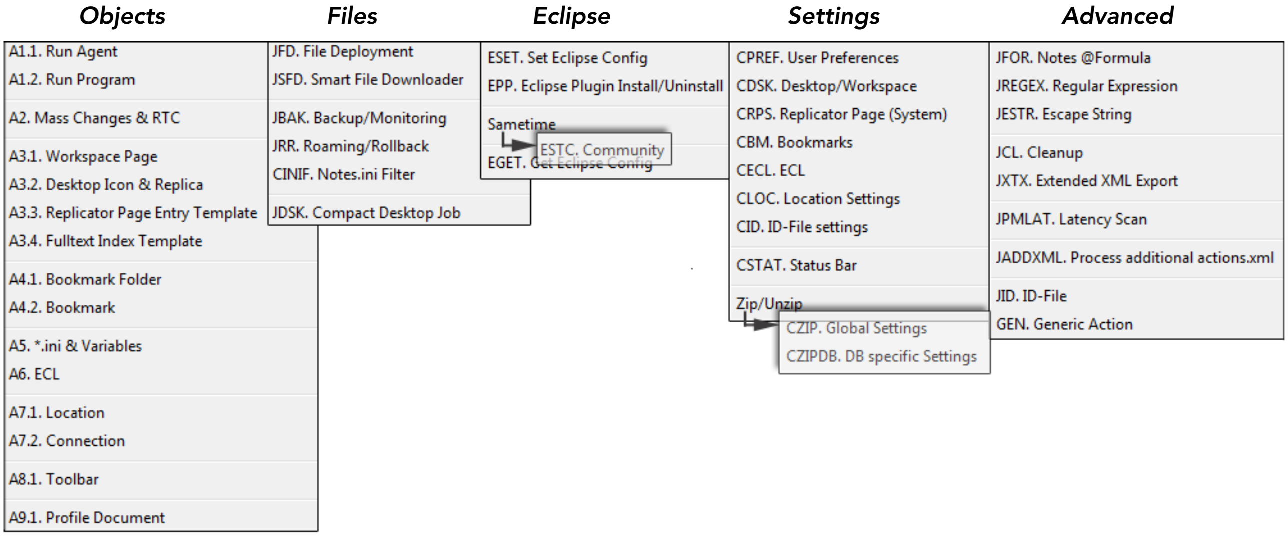


Figure 46: Action Bar – MarvelClient → All Actions – Menu Objects

The ***Objects*** menu includes all Action types for the central management of typical IBM Notes client objects, such as workspace pages, desktop icons, replicas and replicator page settings, bookmarks and bookmark folders, *notes.ini*, ECLs, Location and Connection Documents, as well as Action types for starting Agents and programs without emails, buttons and end-user interaction.

The ***Files*** menu allows for the configuration of special file-based Actions, such as the compression of "desktop*.dsk/.ndk", the distribution of files to clients (including optional, automatic unzip on targeted clients to reduce network traffic), as well as uploading freely configurable files to network drives or Domino databases, down to smart roaming of such uploaded files.

The ***Settings*** menu allows the creation of configuration Actions for settings in IBM Notes clients and MarvelClient, such as options for the workspace (show unread, show server names, stack icons), specifying a default location for the IBM Notes login dialog (through to even optionally overriding the Location after login) and zip/unzip settings.

The ***Eclipse*** menu provides several Actions related to the management of Eclipse-based Notes clients (>8, Standard), such as Sametime settings and installation of additional Eclipse plugins.

The ***Advanced*** menu offers advanced Actions for advanced administrators, such as computing strings with regular expressions or IBM Notes @Formulas, running latency measurements, or the processing of additional *actions.xml* files (keyword *Staging*, or controlling MarvelClient by using xml files which are generated and provided by third party suppliers, such as help desk solutions).

The various different Action types along with their respective options and use cases are described in the following chapters.

ACTIONS: GENERAL

The What, When, Who and Admin Tabs

Every Action has four tabs at its very top:

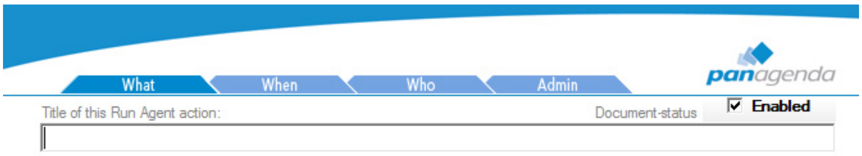


Figure 47: Actions - *What, When, Who* and *Admin* Tab

- ***What***: This tab differs from Action type to Action type and configures the respective settings of each Action itself.

- ***When***: This tab houses the execution control for each Action, including the configuration of how often an Action shall execute (such as once only or permanently). The *When* tab is also where Actions can be tied to Conditions such as *Time Conditions* (see: "Time Condition" on page 83), *IP Range Conditions* (see: "IP Range Condition" on page 85), *String Conditions* (see: "String Condition" on page 86) and *Numeric Conditions* (see: "Numeric Condition" on page 88).

- ***Who***: On this tab each Action can be assigned to users, groups and/or certifiers. If there are no individual settings, the Action is applied to all users (subject to further limitations through Conditions on the *When* tab).

Note: The Who tab affects distribution AND execution while the When tab only affects execution!

- ***Admin***: This tab allows for documentation of comments and advanced configurations such as alias and priority, as well as the preparation of Actions for *Staging*. If anything is set on the *Admin* tab (apart from remarks), it will be **marked with an asterisk (*)**, so that when opening an Action it is instantly evident that this specific tab contains special settings.

Regardless of which tab is selected from within an Action, the action bar always shows the buttons Edit (read mode) or Save and Close (edit mode).

What Tab in Detail

Along with the general fields *Enabled/Disabled* (for document-status) and *Title of…* (this Action) at the top of each Action, the *What* tab is described for each Action in more detail in the appropriate Action chapter (not part of this MarvelClient basic guide).

Enabled/Disabled:

Checkbox for activating/deactivating Actions. Activated Actions are automatically distributed to all end-users in accordance to the *Who* tab – the *When* tab does not restrict the distribution but solely the execution of an Action.

Deactivated Actions are automatically deleted on workstations as long as these were previously activated and already downloaded to a client.

- The automatic deletion of documents which were formerly activated and distributed, and later on deactivated, is also applied to Conditions. Please note that the execution of any Actions that reference any such deactivated or deleted condition(s) will fail due to the inactive reference which can then not be resolved.

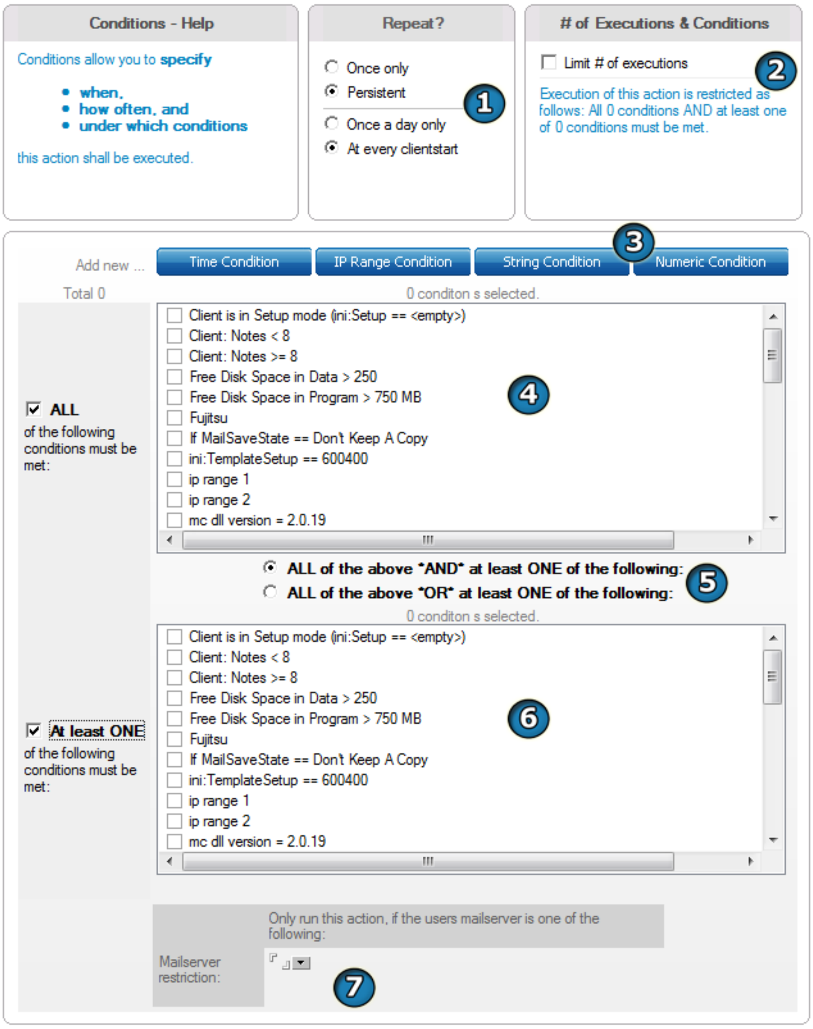
***Title* field and its influence on execution:**

The first field of each type of document (Actions, Conditions, etc.) is a title field for explicit naming of each document. The title may affect the execution order of Actions as Actions with the same Run Time and priority are executed in alphabetical order.

- Runtime selection overrides referencing, which overrides priority which in turn overrides title.
- If an Action requires the result from another Action which is supposed to be executed later during that same Run Time, the execution is then brought forward.
- If however an Action requires the result from another Action which is set to run at a later Run Time, the execution of the Action referencing that later Action will fail as an Action cannot bring another Action forward across Run Times.
- On the other hand referencing an Action, which was already executed at an earlier Run Time, will work across runtime moments.

When Tab in Detail

The *When* tab allows you to configure how often and when an Action is supposed to be executed (or not) – without programming skills. If a user is not excluded from an Action via the *Who* tab, the Action will be downloaded and the settings on the *When* tab (and where appropriate the *What* as well) **determine the execution of the Action**.



Conditions - Help

Conditions allow you to **specify**

- **when,**
- **how often, and**
- **under which conditions**

this action shall be executed.

Repeat?

○ Once only
● Persistent

○ Once a day only
● At every clientstart

# of Executions & Conditions

□ Limit # of executions

Execution of this action is restricted as follows: All 0 conditions AND at least one of 0 conditions must be met.

Total 0

0 conditon s selected.

☑ **ALL** of the following conditions must be met:

□ Client is in Setup mode (ini:Setup == <empty>)
□ Client: Notes < 8
□ Client: Notes >= 8
□ Free Disk Space in Data > 250
□ Free Disk Space in Program > 750 MB
□ Fujitsu
□ If MailSaveState == Dont Keep A Copy
□ ini:TemplateSetup == 600400
□ ip range 1
□ ip range 2
□ mc dll version = 2.0.19

● **ALL of the above "AND" at least ONE of the following:**
○ **ALL of the above "OR" at least ONE of the following:**

0 conditon s selected.

☑ **At least ONE** of the following conditions must be met:

□ Client is in Setup mode (ini:Setup == <empty>)
□ Client: Notes < 8
□ Client: Notes >= 8
□ Free Disk Space in Data > 250
□ Free Disk Space in Program > 750 MB
□ Fujitsu
□ If MailSaveState == Dont Keep A Copy
□ ini:TemplateSetup == 600400
□ ip range 1
□ ip range 2
□ mc dll version = 2.0.19

Only run this action, if the users mailserver is one of the following:

Mailserver restriction:

Figure 48: Action – *When* Tab

1. ***Repeat?***
   Here you can choose how often an Action is supposed to be executed.

2. ***# of Executions & Conditions***
   Only editable <u>if *Repeat* is not set to *Once Only*</u> (see *Repeat?* above). Enter the number of executions in the number field. 10, for instance, means that the Action will only be executed 10 times.

- **Retire Option**

  Only visible with role [Staging]. The default setting is *Retire locally* (= remember in the local *actions.xml* file, how often the Action has already been executed, how many more times it is supposed to be executed, etc.). Any changes to this field are only to be carried out if so instructed by panagenda (support@panagenda.com) or your certified panagenda partner.

3. **Buttons for adding extra conditions**
   These buttons allow you to create custom Conditions, for more information see "Time Condition" on page 83, "IP Range Condition" on page 85, "String Condition" on page 86 and "Numeric Condition" on page 88.

   After saving and closing a Condition which was created using any one of these buttons, the lists of available conditions (4) and (6) need to be updated manually. To do so, **press the F9 key** (function key).

   Existing (both enabled and previously disabled) Conditions can be viewed and edited in the Configuration database under *General* → *Conditions*.

4. **List of conditions which must all be true (n/n) for an Action to be executed**
   Ticking the check box *All* allows you to select any number of Conditions which all need to be true (=fulfilled) for an Action to be executed. The list of available Conditions can be extended by using the above buttons for adding extra conditions or from the *General* → *Conditions* view.

5. **Linking lists of conditions (n/n and 1/n)**
   This field allows to link the conditions from (4)+(6) together using a logical AND or OR.

6. **List of conditions of which at least one must be true (1/n) for an Action to be executed**
   After selecting the check box "At least ONE", this field allows you to choose any number of Conditions, of which at least one must be true (=fulfilled), for an Action to be executed. The list of available Conditions can be extended by using the above buttons for adding extra conditions or from the *General* → *Conditions* view.

7. **Limiting the execution to end-users who have their mailfile on (a) specific mail server(s)**
   In this field you can specify one or more mail server names. When the field is not blank, the Action will only be executed if the mail server of the target users matches one of the specified server names. Note that a users's mailserver is derived from her/his then current Location Document, not the public address book.

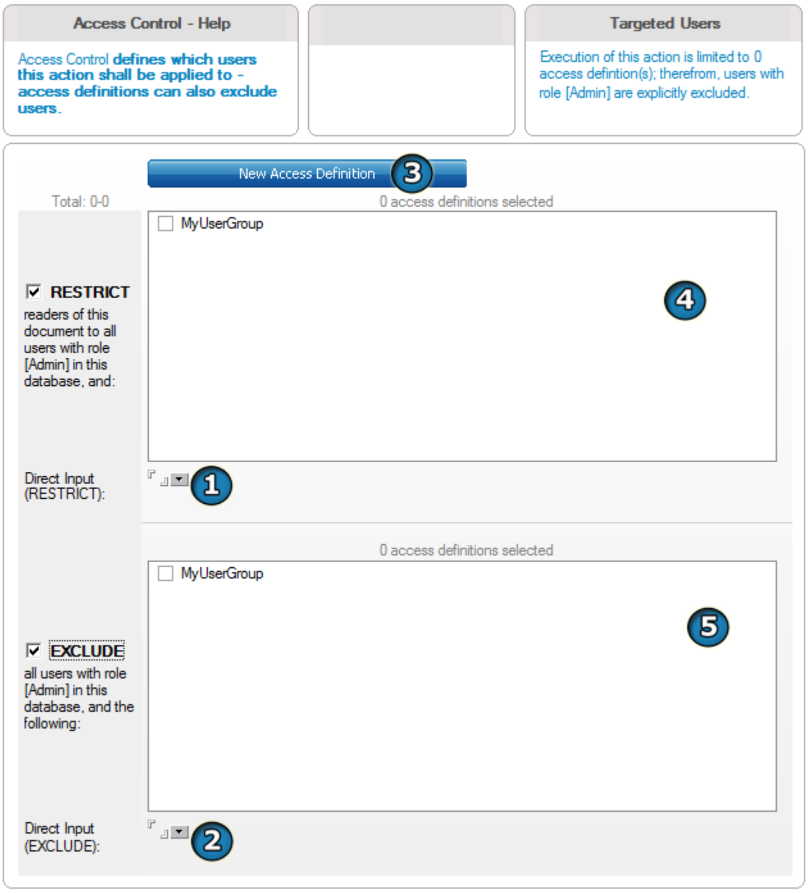
> ***Note: Using <notes:um_server> in this field does not make sense as <notes:um_server> matches every user's mailserver***

Who Tab in Detail

Using reader fields, the *Who* tab – and in particular the *RESTRICT* section – determines which end-users/groups can see (=download) an Action. This allows you to easily deploy Actions to specific users, groups and/or certifiers (see 1. in screenshot description below).

4 and 5 (in screenshot below) allow you to reuse the same of users/groups/certififers across multiple Actions through so called Access Definitions which are similar to group documents in the public addressbook but stored in the MarvelClient Configuration database. New Access Definitions can be created directly from within an Action (see 3. in screenshot description below). Access Definitions can also be created and edited from the *General* → *Access Definitions* view. Access Definitions hence allow administrators to manage re-usable "groups" even if they don't have the permissions to create groups in the public addressbook or in case they do not want to create additional group documents in the public addressbook (see "Access Definition" on page 88)

Using the *EXCLUDE* function, users, groups and/or certifiers can again be excluded from an Action. Instead of having to create a group "Everyone except the board of management", simply check the *EXCLUDE* option and enter "BoardOfManagement" into the direct input field (see 2) (assuming that the group "BoardOfManagement" is stored in your public addressbook and contains all members of the board).



Access Control - Help

Access Control defines which users this action shall be applied to - access definitions can also exclude users.

Targeted Users

Execution of this action is limited to 0 access defintion(s); therefrom, users with role [Admin] are explicitly excluded.

New Access Definition

Total: 0-0

0 access definitions selected

☐ MyUserGroup

☑ RESTRICT readers of this document to all users with role [Admin] in this database, and:

Direct Input (RESTRICT):

0 access definitions selected

☐ MyUserGroup

☑ EXCLUDE all users with role [Admin] in this database, and the following:

Direct Input (EXCLUDE):

Figure 49: Action – *Who* Tab

1. **Direct input (*RESTRICT*)**
   After activating the check box *RESTRICT* enter one or multiple person names, group names and/or certifiers. For example enter "*/Sales" or "*/HQ/ACME" depending on how reasonable it is in relation to your certifier structure.

Note that checking the RESTRICT option always includes all users that have the role [Admin] in the Configuration database. This is necessary because the respective restriction is populated into a reader field. If users with role [Admin] were not automatically included in the list of readers, Administrators could create documents they can then no longer see/edit after saving them.

2. **Direct input (*EXCLUDE*)**
   After activating the check box *EXCLUDE* enter one or multiple person names, group names and/or certifiers. For example enter "*/Sales" or "*/HQ/ACME" depending on how reasonable it is in relation to your certifier structure.

If neither the RESTRICT nor the EXCLUDE option is checked, ALL users will download the respective Action. The EXECUTION, however, is determined by the configuration on the What AND When tabs.

3. **Button for adding Access Definitions**
   This button allows you to create reusable groups in the configuration database itself (see *"Access Definition" on page 88*).

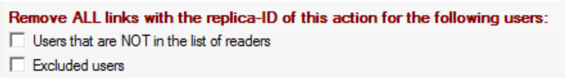
   After saving and closing an Access Definition which was created using this button, the lists of available *Access Definitions* (4.) and (5.) need to be updated manually. To do so, **press the F9 key** (function key).

   Existing *Access Definitions* can be viewed and edited in the Configuration database under *General* → *Access Definitions*.

4. **List of *Access Definitions* for which the Action is valid**
   After activating the check box *RESTRICT* this field allows you to select from existing enabled *Access Definitions* – if any. The total of all users from the selected *Access Definitions* on this list (and if applicable in combination with 1. = direct input of "Readers") will see (aka download) this Action.

5. **List of *Access Definitions* which are excluded from the Action**
   After activating the check box *EXCLUDE*, this field allows you to select from existing enabled Access Definitions. The total of all users from the selected *Access Definitions* on this list (and if applicable in combination with 2. = direct input of excluded users/groups/certifiers) will be excluded from downloading this Action.

Special options on the *When* tab from within Desktop Icon & Replica Actions:



Remove ALL links with the replica-ID of this action for the following users:

□ Users that are NOT in the list of readers

□ Excluded users

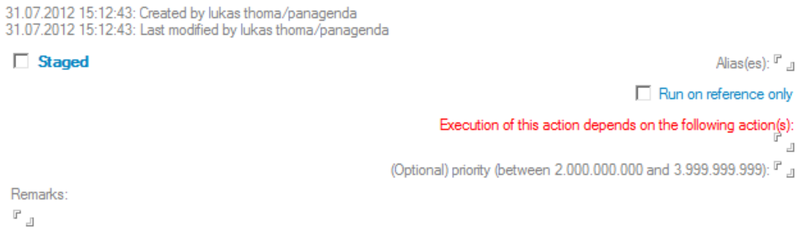
If the ***RESTRICT* check box** is selected the first option allows for the implicit deletion of any links to the respective database for all users that are NOT in the *RESTRICT* scope.

If the ***EXCLUDE* check box** is selected the second option allows for the implicit deletion of any links to the respective database for all those users who are excluded from the Desktop Icon & Replica Action.

If either one of the above special options is checked, the deletion scope can be extended to not just delete desktop icons for the respective desktop icon, but to also include any physical replicas.

Admin in Detail

The *Admin* tab allows you to store comments/remarks along with an Action, as well as specify advanced configuration options such as creating Actions dependencies, configuring a special priority, or other advanced settings:



31.07.2012 15:12:43: Created by lukas thoma/panagenda
31.07.2012 15:12:43: Last modified by lukas thoma/panagenda

□ Staged

Alias(es): ⌜ ⌟

□ Run on reference only

Execution of this action depends on the following action(s): ⌜ ⌟

(Optional) priority (between 2.000.000.000 and 3.999.999.999): ⌜ ⌟

Remarks: ⌜ ⌟

Figure 50: Action – *Admin* Tab

The *Admin* tab also displays the "history" of each Action; in other words, which user created the Action and who last modified it.

If anything special is configured on the *Admin* tab (apart from remarks), it will be **marked with an asterisk (*)**, so that it is instantly clear, that this tab contains special settings when opening an Action.

Staged:

Only visible for users that have the role [Staging].
Only relevant for real mass changes, as to avoid the manual creation tens of hundreds if not even thousands of individual Actions of the type *Mass Changes & RTC*, for example. Only to be performed if so instructed by panagenda (contact: support@panagenda.com) or one of our certified partners.

- In simple terms, the *Staging* of Actions enables the aggregation of many hundreds/ thousands of commands into one collective, zipped Action. Therefore "Staged" Actions do not support the *Who* tab, as they are to be aggregated into one "Super-Action" which has its own *Who* tab.

Alias(es):

This field allows you to specify zero, one or multiple aliases (Multiple string input) to create dependencies between Actions (aka create Action Chains) or reference the return values of an Action from within other Actions. Not all Actions have meaningful return values; those that do are documented in the respective Action documentation (not part of this MarvelClient basic guide).

> ***The alias(es) of one Action must be unique to one destination user. When MarvelClient loads all Actions into memory, it creates an index of all respective aliases. Aliases which are used for more than one Action are deleted from the index. As a result, resolving any references towards the relevant deleted alias will fail.***

Run on reference only:

If your Action shouldn't run on its own at a particular Run Type, but only if referenced from within another Action, check *Run on reference only* here (Note that this option is only visible if the respective Action doesn't offer *On Reference only* as part of the **Run Type Selection** on the *What* tab – some Actions do not offer for Run Type selection on the *What* tab, you can then use this option on the *When* tab)

***Execution of this action depends on the following action(s)*:**

Here is where you can specify one or multiple aliases of Actions that ALL need to be (successfully) executed before executing the Action currently being viewed/edited.

***Priority*:**

Every Action has a recommended priority range which is displayed to the left of the priority field. If the priority field is left blank, the priority of the Action will be the exact mean value of the recommended priority range. This allows you to easily have Actions execute before or after other Actions.

> ***Actions with the same priority and Run Type will be executed in alphabetical order of their title.***

The recommended priority range should only be exceeded by expert users with thorough knowledge of MarvelClient. When saving an Action with a priority outside of the recommended priority range you will receive a warning message. Note that this is not a validation error message which will prevent you from saving the document, but a warning only.

Exceeding the default priority range is helpful in certain scenarios, such as when you want to create a location only after a *Desktop Icon & Replica* Action has created a local replica of the user's mail file. By default, location Actions run before Desktop Icon & Replica Actions; priorities can override this.

> ***Keep in mind: <u>Run Time selection</u> overrides <u>referencing</u> which overrides <u>priority</u> which in turn overrides <u>title</u>.***

Default Priorities Explained

While assigning a default priority of 3 billion to something may seem odd, it does make sense when dealing with heavily populated MarvelClient Configuration databases.

For example, if Actions were to be numbered in steps of 1, 2, 3, there would be no room in the future to insert other Actions, without having to renumber everything.

Because of this, we recommend that Actions are assigned "gapped" numbers, such as 1.000, 2.000, 3.000 and so on. Stretching this concept into gaps of 10,000 or more leads to easily filling a range of a couple of billions.

By default, Location Documents start with a default priority of 7,500, Workspace pages start with 1,500,000,000, icons have 3,000,000,000. This is done in an effort to assist with logical ordering of Actions: a readily populated Location Document is required for managing the mailfile icon (stack and replica, if applicable), for example, and workspace pages need to be in place before managing icon(stack)s.

Remarks:

This field can be used to store comments or customer specific documentation.

APPENDIX

List of Figures

List of Tables

Notes: