Installing/Upgrading Midrange Dynamics Products on the IBM i

1.1 Prerequisites

1.1.1 Operating System

IBM i (AS/400) System with OS/400 Release **V7R1M0** or higher

The current version can be checked using command WRKLICINF

1.1.2 PTFs

None

1.1.3 Java Virtual Machine

The MD products uses Java for various processes relating to Excel generation, FTP, Object Signing, PDF generation, SMTP, and Zipping. In order to carry out these processes, a JVM (minimum 1.7) must be installed. The MD products can be used without Java, but some features will not be available.

The installed JVMs can be checked by using command:
WRKLNK '/QOpenSys/QIBM/ProdData/JavaVM'

1.1.4 MD License Keys

In order to use the products, a valid License Key is required for the core product (MDCMS or MDXREF), which is based on the Serial Number of the system and the version of the Product.

To use one of the add-ons to the core product, such as MDOpen or MDWorkflow, a separate key will also be required.

Please provide Midrange Dynamics or a reseller with the serial number (obtained with command WRKLICINF or DSPSYSVAL QSRLNBR), partition number (from the Hardware Management Console, if applicable) and product version and continue with the installation once the key or keys have been provided.

If the keys are provided in file MDLICENSE.savf, include this file in the same IFS path or library as the product save files. When MDLICENSE.savf is included, the keys for the given serial number and partition number will be automatically applied to the product instance, and the keys for all other included systems will be stored in the product instance so that if a switch is made to a backup system, the backup keys will automatically be applied to the product, thus avoiding any delays.

1.1.5 User Authorization

The user performing the installation must have *SECADM and *ALLOBJ authority on the system.

Additionally, in order to enroll a user in MDSEC, the initial enrollment must be performed by a user with user class *SECADM or *SECOFR.

1.1.6 Allow Restore of Programs with Adopted Authority

The system value **QALWOBJRST** must include the choice of *ALL or *ALWPGMADP so that the product programs with the adopt attribute can be restored.

1.1.7 Disk Space

The product initially requires about 900 MB of space at installation time.

A fully built cross-reference database in MDXREF averages an additional 500 MB of space.

MDCMS history typically requires an additional 300 MB per year of use.

1.1.8 Exclusive Product Library Locks when Upgrading

When upgrading an existing version of the MD Products to a new build, the existing library instances for MDCMS, MDREP, MDSEC and MDXREF may not be in use.

Object locks can be checked by using command WRKOBJLCK OBJ(MDSEC) OBJTYPE(*LIB).
MDSEC is included in any MD process, so it is normally sufficient to check only for locks against this library instance.

If locks exist, you can cleanly end the jobs ahead of time, or specify parameter END(*YES) on the MDINSSAVF command to automatically end all jobs locking the product libraries.

1.2 Objects created by Installation Process

1.2.1 QGPL Objects

The following commands are placed by default into library QGPL:
MDINSSAVF – The MD Product Installer
MDSEC – MD Security Menu
MDXREF – MD Cross Reference
MDCMS – MD Change Management menu, if MDCMS is installed

For each command, a corresponding program and panel group are also placed in QGPL.

1.2.2 Libraries

Some or all of the following libraries will be created, depending on the product:

Library	Description	Initial Size
MDSEC	MD Security and core objects used by all MD products	150 MB
MDXREF	MD Cross-Reference used by MDXREF, MDCMS, MDOpen or MDWorkflow	50 MB
MDCMS	MD Change Management used by MDCMS, MDOpen or MDWorkflow	500 MB
MDREP	MD Repository Objects used by MDOpen or MDWorkflow	200 MB

All data accumulated by MDCMS, MDSEC and MDXREF are kept in standard DB2 tables within their respective libraries. It is highly recommended to regularly back up those 3 libraries.
MDOpen and MDWorkflow don't have their own database, but instead access the data in those 3 libraries using JDBC so that a central database is utilized to avoid synchronization issues.

MDREP doesn't contain critical site-specific data, so it isn't necessary to back up.

By default, the libraries are named as stated in the table. At installation time,
a 1-4 character Instance ID (ENV) can be defined which will be used as the suffix for the library names. For example, suffix T would mean that product MDCMS would be stored in library MDCMST.

This way, several instances of the MD products can reside on the same system. The runtime is differentiated by stating the suffix when running a product command.

1.2.3 IFS Folders

If MDCMS is installed, a folder named MDCMS is created in the IFS at the root of the selected ASP device. This folder contains Java components, Mail Bodies, Remote Server Scripts and is the location for the archiving of replaced objects and commitment repository of Remote Server objects.

If you are upgrading from a prior version of MDCMS, the Mail Body templates of the new version will be copied to a sub-folder named for the version and Script examples will be copied to a sub-folder named for the version.

1.2.4 User Profiles

A user profile is created to own the objects in the product libraries. The profile is created without the ability for users to sign on or otherwise make use of the profile. By default, the name of this user profile is MDOWNER, but can be changed at installation time by specifying a different value for MDINSSAVF parameter OWN.

If the product owner profile already exists, it is left as is.

MDCMS programs use adopted authority from the owner profile, which has *ALLOBJ authority, so that the actual users authorized to perform functions in MDCMS do not need to have any special system authorities to accomplish the task of making changes to your business applications.

Any programs providing access to a command line do not have adopted (*OWNER) authority.

None of the programs have parameter “Use adopted authority” set to *YES, ensuring that authority won’t be inherited from your internal calling programs.

The following 2 Programs are owned by QSECOFR:

MDSEC/MDLOAUT – apply object authority to deployed objects. If the user defined on the job description for an application level is always the owner for all objects in that level, then the owner of MDLOAUT may be changed to another profile, such as MDOWNER, that doesn’t have *SECADM special authority.

MDCMS/MDLRFIL – authorize DDM access to MDCMS databases on remote OS/400 partitions for cross-system analysis and synchronization. If this feature won’t be used, then the owner of MDLRFIL may be changed to another profile, such as MDOWNER, that doesn’t have *SECADM special authority.

If you are upgrading an existing installation of MDCMS, and a user or users were specifically granted object authority to objects in MDSEC, MDSEC or MDXREF, then those special rights will need to be reapplied after the upgrade is complete. This is typically granted for the user profile for DDM and FTP connections that is defined in the OS/400 Location settings when connecting to this system.

1.3 The Installation Steps

1.3.1 Downloading and extracting the Save Files

1) Sign into the Midrange Dynamics Service Desk portal at https://servicedesk.mdcms.ch/

   You will need to be registered to use the portal. If not yet registered, request registration from: https://www.midrangedynamics.com/request-service-desk/

2) Proceed to the Downloads section and select the MDChange (MDCMS) product group. Then click on the download link for the intended version (the newest non-Beta version is generally recommended).

   If you additionally wish to install MDOpen or MDWorkflow, those are located separately under their sections of the Downloads page and each have their own installation instructions.

3) Save the zip file to a local directory on your PC.

4) Extract the save files to a local directory on your PC.

1.3.2 Option 1 - Installing from Save Files in IFS

1) Copy the save files to an IFS folder on your IBM i system.

   NOTE: One of the easiest ways to copy the save files to an IFS folder on your IBM i is to upload them using IBM's Integrated File System action in Access Client Solutions.
   Alternatively, you may be able to use FTP. A recommended FTP client is **FileZilla**, which is available for free from the internet.

2) If command MDINSSAVF already exists in an IBM i library on your system, and it was created since version 8.5, skip to section 1.3.4

3) Enter command ***CRTSAVF QTEMP/MDINSSAVF***

4) Enter command
   CPYFRMSTMF FROMSTMF('/x/mdinssavf.savf')
   TOMBR('/qsys.lib/qtemp.lib/mdinssavf.file') MBROPT(*REPLACE)
   where **x** is the name of the IFS folder containing the save files

5) Enter command
   RSTOBJ OBJ(*ALL) SAVLIB(QGPL) DEV(*SAVF) SAVF(QTEMP/MDINSSAVF)

   The objects may be restored to a different library than QGPL, if desired. However, the CHGCMD command will need to be used on the MDINSSAVF command to change the library for the program and panel group.

1.3.3 Option 2 - Installing from Save Files in a Library

1) Copy the save files to a Library on your IBM i system.

2) If command MDINSSAVF already exists in an IBM i library on your system, and it was created since version 8.5, skip to section 1.3.4

3) Enter command
RSTOBJ OBJ(*ALL) SAVLIB(QGPL) DEV(*SAVF) SAVF(x/MDINSSAVF)
where **x** is the name of the library containing the save files

The objects may be restored to a different library than QGPL, if desired.

1.3.4 Invoke the MDINSSAVF Command

Enter command ***MDINSSAVF*** and press F4 to review command parameters and make any necessary changes (press F1 or use the parameter table on next page for more information).

Installations may occur in an interactive or batch job. Interactive is recommended for new installations for improved monitoring and prompting of the process.

If using MDINSSAVF to upgrade an existing product to a newer version, the distribution and installation can occur as part of an RFP. In this case ensure the following:

- MDINSSAVF should be a Post-Installation (3) attribute command attached to the *IFS or *FILE attribute defined with the IFS path or save file library as the target object library.
- Run for Modifications = Y
- Keep MD Libs in Libl = N
- Frequency = R
- MDINSSAVF command should begin with SBMJOB so that it runs separately from the RFP
- END(*YES) to terminate any locks
- DLY(30) to allow RFP time to finish before starting with installation
- USER for SBMJOB should be a profile with *SECADM authority
- INLLIBL for SBMJOB should include only QTEMP and QGPL (or library where MDINSSAVF command exists)

Example command definition:

```
 Appl.....: MD                                Run for Modifications: Y  Y/N
 Lvl......: 50                                        Recompiles...: N  Y/N
 Attribute: INSTALL     Attribute, *RFP               Deletes......: N  Y/N
                                                      Updates......: N  Y/N
 Type.....: 3  Post-Installation             Ignore Errors........: Y  Y/N
 Sequence.:   1                              Keep MD Libs in Libl.: N  Y/N
 Frequency: R  O=Object, R=RFP

Command
SBMJOB CMD(MDINSSAVF PATH('##OBJLIB##') END(*YES) DLY(30)) JOB(MDINSTALL) USER(QSO)
INLLIBL(QTEMP QGPL)
```

Option 20 from the MDCMS Setup Menu can be used to automatically create the application settings to manage the update of the MDCMS product.

MDINSSAVF Parameter Table

Parameter	Label	Description
PRD	Core Product to Install	Specifies which Core Product Group should be installed *CMS – The MDCMS product group, which includes MDXREF and MDSEC, will be installed. *XREF – The MDXREF product group, which includes MDSEC, will be installed. Use when MDXREF will be used as a standalone product.
REP	Incl. MDREP for Open/Workflow	Specifies whether or not the MDREP library will be installed. The MDREP library is used for the interface components between MDCMS and MDOpen or MDWorkflow. Must be *YES if MDOpen or MDWorkflow will connect to this partition. *YES – The MDREP interface library will be installed *NO – The MDREP interface library will not be installed
LTYP	Save File Location Type	Specifies if the product save files are in an IFS folder or in a library. *IFS – The save files are located in an IFS folder. Specify the full path of the folder in parameter PATH *LIB – The save files are located in a library. Specify the name of the library in parameter LIB
PATH	IFS-Path containing Save Files	Specifies the IFS path containing the save files used to install the Midrange Dynamics products. The contents of the path must contain a save file for each library of the selected Product Group. If MDREP is included, then it must also be in the path.
LIB	Library containing Save Files	Specifies the Library containing the save files used to install the Midrange Dynamics products. The contents of the library must contain a save file for each library of the selected Product Group. If MDREP is included, then it must also be in the library.

ENV	Product Instance	Specifies the instance ID of the product. The ID is appended to the names of the product libraries in order to allow multiple instances of the products on the same partition. *DFT – A suffix is not appended to the library names. They will be named MDSEC, MDCMS, etc.. The suffix to be used. Up to 4 characters are allowed and each character must be accepted as part of a library name.
FENV	Copy Data from Instance	Specifies the location of an instance of the products, if it exists, that should be used for the copy of the data to the new version. *SAME – The same instance, or library suffix, that was defined in parameter ENV *DFT – Copy the data from the default instance, which doesn't contain a suffix *NONE – Do not copy any data to the new version. The new version will be a clean installation. Specify the product instance, or library suffix, containing the data to be copied to the new version.
BSFX	Backup Library Suffix	Specifies the suffix to be appended to the libraries containing the version to be replaced by this installation, if a prior version exists for the instance defined in parameter ENV. This way, the prior version isn't lost and can be reactivated by renaming the libraries from the backup suffix to the instance suffix.
BREP	Replace existing Backup Libs	Specifies if existing backup libraries using the same suffix as defined in parameter BSFX should be automatically replaced. *YES – Any existing libraries with the same name as the new Backup libraries will be automatically replaced. *NO – Libraries with the same names as the new Backup libraries will not be replaced, and the installation will not occur if any exist.
ASPD	Install in ASP Device	Specifies the auxiliary storage pool (ASP) device to which the product libraries should be restored. *SYSBAS – The product libraries will be restored to the base system ASP. character-value – the product libraries will be restored to the indicated ASP device.

END	End Jobs locking Product Libs	Specifies if all jobs that have a lock on one of the product libraries should automatically be ended immediately. *NO – If a lock exists for one of the libraries, the installation process will be cancelled. *YES – Each job that has a lock on one of the product libraries will be ended immediately. If ending fails for a job, the installation process will be cancelled.
DLY	Delay before Starting	Specifies the amount of time, in seconds, that the installer job should wait before beginning with the installation process. This is helpful if the MDINSSAVF command is submitted by an MDCMS RFP as a Post-Install command for the save files. By delaying, the RFP can complete normally so that locks from the RFP will no longer be on the product libraries for automatic batch installs of MDCMS upgrades. 0 – immediately begin with the installation decimal-value – The number of seconds to wait before beginning. 30 seconds is recommended for a submitted install from an RFP Post-Installation command.
OWN	Product Object Owner	Specifies the user profile to be used to own the objects in the MD product libraries and IFS folders. If the profile doesn't exist yet, the installer will create it. The profile should have *ALLOBJ, *JOBCTL, and *SPLCTL special authorities to ensure that deployments function correctly.
SECU	User to Register in MDSEC	Specifies a user profile to add to MDSEC. This user will be granted authority to administer the MDSEC authorities and will have the user roles based on parameter ROLE. *NONE A user profile will not be added to MDSEC. *USER The current job user should be registered in MDSEC. character-value A valid user profile ID to register in MDSEC.
ROLE	MDSEC Roles for User	If a user is provided in parameter SECU, that user will be granted certain roles based on the value of this parameter. *NODEV The user will be granted all roles except for those that begin with MD_PGMR. This allows for all administrative and management functions but doesn't require a developer license. *ALL The user will be granted all roles that exist in MDSEC including for development.

		*USER The user will be granted only role MD_USER for basic access to the MD products.
LOCT	Location Title	Specifies a title to apply to the instance of MDCMS being installed. This appears at the top of each product screen to help differentiate it from other locations/instances. *SAME The current title is retained character-value A Title to be displayed within the product.
CCSID	Default CCSID	Specifies the default CCSID to be used by interactive and batch jobs operating from within the MD products. *SAME The current CCSID is retained character-value A valid 2-5 digit CCSID that is language specific (not Hex or UTF)
SBSD	Subsystem for MD Jobs	Optionally specify the subsystem to be used for MD batch jobs. If the subsystem doesn't exist, it will be created in the specified library. If the library parameter value is *LIBL, QGPL will be used. character-value The name and library of a subsystem description. If the library is specified, but doesn't exist, it will be created in the ASP device specified in parameter ASPD.
JOBQ	Job Queue for MD Jobs	Optionally specify the job queue to be used for MD batch jobs. If the job queue doesn't exist, and a value was provided for parameter SBSD, the job queue will be created and added to the subsystem. If the library parameter value is *LIBL, QGPL will be used. The JOBQ parameter will be ignored if the SBSD parameter is not provided. character-value The name and library of a job queue. If the library is specified, but doesn't exist, it will be created in the ASP device specified in parameter ASPD.
SAPLIB	SAP Data Library	Specifies the SAP application data library that will be the target for updates to SQL tables, indexes and views. If a value is provided, the installer will automatically configure the SAP application, level and attributes. *NONE The configuration for SAP database management should not be created. character-value The name of the SAP database library.