Matrix42

Empirum

UEM Agent Windows

2203.1.2 Standard Feature Release

Version 1.0.12

22/03/2022

Copyright © Matrix42 AG

This documentation is copyright protected. All rights are reserved by Matrix42 AG.

Any other use, in particular the disclosure to third parties, storage in a data system, dissemination, processing, presentation, performance and demonstration are prohibited. This applies to the entire document, as well as parts thereof.

Subject to change. Reprint, also in excerpts, is permitted only with the written consent of Matrix42 AG.

The software described in this document is subject to a permanent development due to which there may be differences in the documentation and the actual software. This documentation is not entitled to the actual functionality of the software.

Apple and **Mac OS X** are registered trademarks of Apple Inc.
Citrix® software or **Citrix® server** are Trademarks and Registered Trademarks of Citrix Systems, Inc. in the United States and other countries.
cygwin is copyrighted by Red Hat Inc. 1996-2003.
expat is copyrighted by Thai Open Source Software Center Ltd.
gSOAP is copyrighted by Robert A. van Engelen, Genivia, Inc. All rights reserved.
Iconv is copyrighted by 1999-2003 Free Software Foundation, Inc.
Iperf is copyrighted by the University of Illinois, except for the gnu_getopt.c, gnu_getopt_long.c, gnu_getopt.h files, and inet_aton.c, which are under the GNU General Public License.
Libmspack (C) 2003-2004 by Stuart Caie <kyzer@4u.net>.
OpenSSL This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit.
PuTTY is copyrighted by Simon Tatham. Portions copyright Robert de Bath, Joris van Rantwijk, Delian Delchev, Andreas Schultz, Jeroen Massar, Wez Furlong, Nicolas Barry, Justin Bradford, Ben Harris, Malcolm Smith, Ahmad Khalifa, Markus Kuhn, and CORE SDI S.A.
RSA Data Security, Inc. MD5 Message-Digest Algorithm is copyrighted by RSA Data Security Inc. Created 1991. All rights reserved.
rsync is an open source utility that provides fast incremental file transfer. rsync is freely available under the GNU General Public License version 2.
runcontrol The Initial Developer of the Original Code is James Clark. Portions created by James Clark are Copyright (c) 1998 James Clark. All rights reserved.
SNMP++ Copyright (c) 1996 Hewlett-Packard Company.
VMware, the **VMware "boxes" logo and design, Virtual SMP, VMotion vSphere, vSphere Hypervisor (ESXi), ESX, View, ThinApp, vCenter** and **vCloud** are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions.
Windows, Windows XP, Windows Vista, Windows 7, Windows 8, **Windows 8.1**, **Windows 10**, **Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2 and Windows Server 2016** are registered trademarks of Microsoft Corporation.

Others, at this point not explicitly listed, company, brand and product names are trademarks or registered trademarks of their respective owners and are subject to trademark protection.

Table of Contents

1. UEM Agent Windows 5
   1.1. Integration of the Marketplace download 5
   1.2. System requirements 5
      1.2.1. Microsoft .NET Framework 5
      1.2.2. Microsoft Visual C++ 2019 Redistributable Package 6
2. New features and changes 7
3. Installation, Update and Rollout 8
   3.1. UEM Agent Auto Update 8
   3.2. Downgrade to Advanced Agent 10
4. Installation window (Z-Dialog/Task List) 11
   4.1. Important changes compared to the Advanced Agent: 11
   4.2. Installation query (Z-Dialog) 11
   4.3. Postpone 11
   4.4. Installation on shutdown 12
      4.4.1. Software installation only at shutdown 12
   4.5. Assignment of optional packages 13
      4.5.1. Optional packages kiosk 14
5. Tips and Troubleshooting 15
   5.1. If the Software Kiosk or Installation UI cnnot be displayed 15
   5.2. Anti Virus Programs 15
   5.3. How to use the global silent switch 15
   5.4. Shutdown the computer after OS installation 15
   5.5. Hide the option "Install on Shutdown" 16
   5.6. Suspend mode 16
   5.7. Script files check for kiosk packages 16
   5.8. Configure behaviour if Windows requires a reboot 16
   5.9. Set Rights of Agent Cache 17
   5.10. Behavior in case of problems during package download 17
6. Customization of the user interface 18
   6.1. Kiosk 18
   6.2. StatusUI 18
7. Installation of the UEM Agent via Silverback 20
   7.1. Requirements 20
   7.2. Importing the UEM Agent MSI into Silverback 20
   7.3. Automatic assignment of newly enrolled computers 21
8. Pre-Logon Window 23
9. Validation of packages before installation 24
   9.1. Enabling package validation for UEM Agent 24
   9.2. Effect of incorrect validation of packets 24

Table of Contents

9.3. Observing the sequence 25
9.4. Enabling package validation via Empirum console 25
10. Check for signing 26
10.1. Activate signing for UEM Agent 26
11. Installation and login after reboot on Full disk encryption 27
12. Alternative Package Sources 28
12.1. Loading Packages into Microsoft Azure Blob Container 28
12.1.1. Create a Storage Account 28
12.1.2. Create Blob Container under the newly created Storage Account 29
12.1.3. Copy software packages to Azure Blob Container 30
12.1.4. Important notes 30
13. Info page 32
14. Survey 33
15. Auto Update Flow Diagram 34
16. UEM Depot Management 35
16.1. General 35
16.2. Installation 35
16.3. Configuration 35
16.3.1. Available synchronizations 35
16.3.2. Exclude directories 37
16.3.3. Exclude files 38
16.4. Protocoll und Master Server 38
16.5. Open points 39
16.6. Troubleshooting 39

1. UEM Agent Windows

Matrix42 supplys a new, enhanced software management agent. This agent is based on the Advanced Agent and is now called “UEM Agent Windows”. We plan to supply additional technical previews and versions that have been officially released for productive operations as soon as they are available through the Matrix42 Marketplace accordingly.

We also plan to supply the respective current version of the UEM Agent, together with each release. Details can be found in the corresponding release notes.

Please use the most current version for your tests.

New versioning logic. The UEM Agent is developed independently from the Empirum main release in other cycles. There are three release types of the UEM Agent:

- ESR: Extended Support Release
- SFR: Standard Feature Release
- TP: Technical Preview

The main release includes the last available ESR. The current versions are available for download on the Marketplace. Matrix42 always supports the respective current UEM Agent version.

1.1. Integration of the Marketplace download

The UEM Agent is provided as a ZIP file. Unzip the ZIP file and copy the content to the Empirum directory of the Empirum Server (overwrite any existing files).

Included are:

- UEM Agent package for import (Configurator\PackageStore\{ID})
- UEM Agent MSI (Configurator\Packages\Matrix42\UEM Agent Windows\MSI\XXXX)
- Matrix42UEMAgentUpdater (Configurator\User\UEMAgentUpdater)

Import the package using the import function of the EMC in the software depot.

The Matrix42UEMAgentUpdater is used when updating the Agent. As of Empirum 18.0.3 it is integrated.

The current documentation is contained in the Help directory of the UEM Agent package.

1.2. System requirements

1.2.1. Microsoft .NET Framework

- Version 4.6.1 for UEM-Agent versions before 1910.0
- Version 4.7.2 for UEM-Agent versions from 1910.0 onwards
  - Microsoft .NET Framework 4.7.2 supports the following operating systems:
    - Windows Server 2016 (version 1709)

- Windows 10 Anniversary Update (version 1607) (x86 and x64)
- Windows 10 Creators Update (version 1703) (x86 and x64)
- Windows 10 Fall Creators Update (version 1709) (x86 and x64)
- Windows Server 2012 R2 (x64)
- Windows 8.1 (x86 and x64)
- Windows Server 2012 (x64)
- Windows Server 2008 R2 Service Pack 1 (x64)
- Windows 7 Service Pack 1 (x86 and x64)

The .NET package is automatically installed by the UEM Agent during the installation via package or Agent Push. Only when installing via MSI this has to be installed before.

1.2.2. Microsoft Visual C++ 2019 Redistributable Package

- Version 14.30.30708.0

The redistributable package is automatically installed by the UEM Agent.

2. New features and changes

The complete list of new features and bug fixes can now be found in a central location in the Marketplace at https://marketplace.matrix42.com/details/uem-agent-windows-release/

3. Installation, Update and Rollout

The UEM Agent can be installed directly or as update via a running Advanced Agent. Important – Please note that a downgrade to the Advanced Agent is only possible with Empirum 17.0.2.

Note:

The UEM agent does not install the Empirum Advanced Agent 17.0 or 18.0 package if the UEM Agent is also assigned. Reinstallation of the Advanced Agent via the reinstallation function is not possible in this case. To return to the Advanced Agent, remove the UEM Agent package and assign only the Advanced Agent package.

Also make sure that no older version of the UEM Agent is assigned when the newer one is installed.

The installation can be performed via package, agent push or the UEMAgent.bat file.

To install the latest UEM Agent via agent push, the variable MX42_AGENT_PUSH_PACKAGE_FOLDER must be assigned the path to the required UEM Agent version. Example: UEM Agent Windows\1808.0

Based on the new versioning logics, it is easier to run multiple versions in parallel, for instance, to be able to test them during roll-out of a new version step by step.

MSI packages (64bit and 32bit) are provided for the initial installation of the UEM Agent. These now include the appropriate vcredist versions (x64 and x86 for 64bit MSI and x86 for 32bit MSI). The parameters are the same as with the Advanced Agent.

3.1. UEM Agent Auto Update

As of version 1811.0, the UEM Agent supports an automatic update. Three variants for updating the UEM Agent are supported. All variants use a utility program to perform the update process smoothly.

- Installation of the assigned version.
- Automatic installation of the latest UEM Agent version integrated and released in the software depot.
- Automatic installation of the latest UEM Agent version integrated in the software vault (also not released). This can be used for pilot rollouts.

The behavior is controlled with the variable "MX42_UEM_AGENT.Auto Update".

- No - Use assigned version
- Yes - Use latest approved version
- Yes - Use latest available version (Pilot rollout)

In addition to this option, since Empirum version 20.0.2 it is possible to select the release type for the Auto Update. This ensures that only agents of the specified type are considered for the Auto Update.

The behavior is controlled with the variable "MX42_UEM_AGENT.Release Type".

- ESR
- SFR
- TP

If the variable is not set, or if an Empirum version older than 20.0.2 is used, the agent only considers versions of its release type when performing an Auto Update. The change of the release type requires in this case an installation via assignment or a push.

If a fully automatic update is selected (Yes..), it is recommended not to assign a UEM Agent version, as the assigned version may no longer correspond to the actually installed version and therefore the status is yellow.

Important note: The control of the automatic installation (Yes) via the release of the package in the depot is only possible starting with Empirum 18.0.3. With older versions the release option is not considered and thus the most current version known in the depot is installed.

Agent update procedure:

1. The UEM Agent checks the variable with every normal polling. There is no check:
   1. 30 minutes after system startup
   2. When the user logs in
   3. During manual testing for new software by the user
   4. Outside the maintenance window configured in the template
   5. When the Software Kiosk is open
2. If the variable is configured to "Yes", the UEM Agent compares the versions available in the depot (SWDepot.dds) with the locally installed version.
   1. "Yes - Use latest approved version" only released versions are taken into account (from Empirum 18.0.3)
   2. "Yes - Use latest available version (Pilot rollout)" all versions available in the depot will be taken into account.

   If there is no newer version to install, the polling will continue normally.
3. If the variable is configured to "No" or not, the agent update will be performed if a newer version than the locally installed one is assigned.
4. The target version is downloaded
5. The UEM Agent is set to suspend mode.
6. The update utility Matrix42UEMAgentUpdate.exe (in the directory Configurator\User\UEMAgentUpdater) is started.
7. Matrix42UEMAgentUpdate stops the UEM Agent and all associated processes.
8. Matrix42UEMAgentUpdate uninstalls the current UEM Agent.

9 Matrix42UEMAgentUpdate installs the desired (assigned or latest) UEM Agent.

10 Matrix42UEMAgentUpdate exits itself.

11 The UEM Agent is automatically active after installation. Suspend mode is no longer active.

Please see the flow diagram at the end of the document.

A downgrade to an older UEM Agent version can only take place if the variable is configured to "No - Use assigned version" and the desired, older, Agent version is distributed via "Reinstallation". Please note that "Reinstallation with previous deinstallation" must not be used under any circumstances.

3.2. Downgrade to Advanced Agent

Note: The downgrade to the Advanced Agent is only supported with the Advanced Agent of Empirum 17.0.2 or newer.

4. Installation window (Z-Dialog/Task List)

The UEM Agent provides a modern interface to make it easy for the end user to see what options he has to execute or postpone management actions.

4.1. Important changes compared to the Advanced Agent:

The number of postponements is specified by days and no longer by "polling" intervals. Two times means that the user can postpone on two days, after that there is no possibility to postpone. The change of day takes place at 0:00 o' clock.

Only the days on which a postponement was made are counted. For example, if the user does not work on weekends, these days are not counted. If an end date for a rollout is desired, this distribution option should also be used.

In order to keep the number of interruptions as low as possible and to clearly show the possibilities to non-experienced users, the end user is only shown a single number with possible shifts. This is always the least possible.

If a package with 3 days and one with 7 days (or a date in the distance future) is assigned to the computer for installation, the end user has three days to postpone. Then, at the latest, both packages will be installed.

Packages with the FORCE and HIDE distribution options are ignored.

Tip: since the user is no longer disturbed by every polling when he has postponed, this interval can now be set shorter in order to obtain changes closer to the time.

4.2. Installation query (Z-Dialog)

The new installation dialog now shows the user the packages to be installed in a tiled view. Depending on the Agent Template configuration, the user can now execute the action immediately or switch to the Postpone dialog.

4.3. Postpone

If at least one day is still available for moving, the end user can select postpone.

The window that appears shows how many days can be moved and when the next reminder is planned.

The user can now postpone to predefined times. We offer 30 minutes, 2 hours, 4 hours, at shutdown and the next day. The displayed time is rounded up to a full 15 minutes. If a maintenance window is specified in the Agent Template, this is taken into account and only possible installation times are offered. The user can request a new query at any time by clicking on the tray icon "Check for newly approved software".

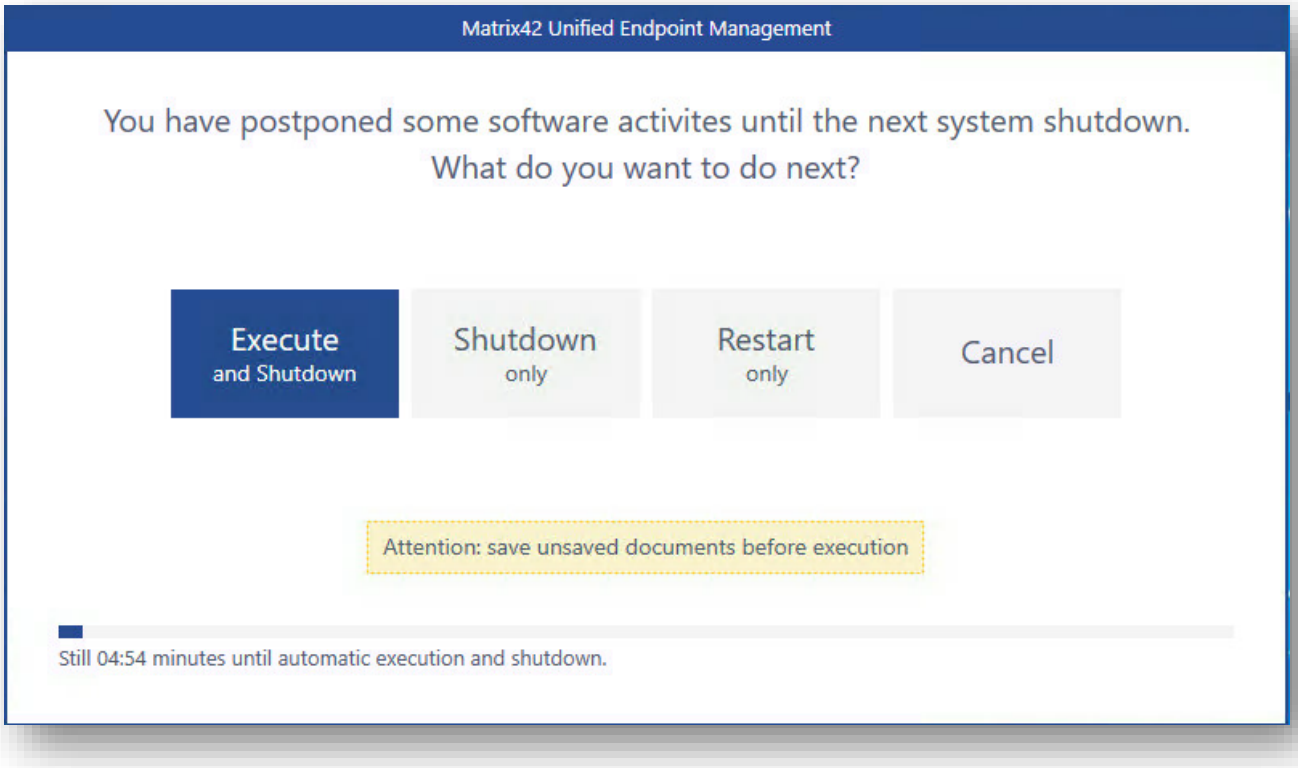
Selecting a time will not execute any action, the user will be reminded again at the selected time. Exceptions are software push and packages with the option "Ignore installation timeframe".

Back "returns to the installation query

4.4. Installation on shutdown

The user can postpone installations until shutdown if it is allowed in the agent configuration. If this option is selected in the postpone dialog box, no query will be made until the computer shuts down. This selection is reset at 0:00 so that users who do not shut down the computer will be informed at least once a day that installation tasks are pending. If the user chooses "Shutdown" in Windows, he will be informed about pending installation tasks and asked to start them. The user is then logged off for security reasons and all tasks, including any necessary restarts, are performed. The user can also start the "Installation on Shutdown" process directly from the tray icon of the agent if it is configured and a software installation is pending.

Software packages that can only be installed when the user is logged in are not installed.



4.4.1. Software installation only at shutdown

If software is only to be installed during shutdown, for example to avoid notifications and updates in call centers or at customer counters during opening hours, software packages can either be executed completely in the background or installed only during shutdown.

Example configuration:

- The agent template is configured in such a way that the maintenance window is outside working hours (if necessary, completely deactivated).
- An icon is placed on the desktop, with which the user shuty down the computer:
- "C:\Program Files\Matrix42\Universal Agent Framework\ERIS_UI.exe" /installshutdown
- Packages that can/should be pre-installed in the background are assigned with the "Hide" and "Ignore installation time frame" distribution options.
- All packages to be installed on shutdown only are assigned with default distribution options.

Users are no longer interrupted and there is no "normal" installation. To shut down, the user always chooses the icon and "Execute". If no software is available for installation, the system will shut down after approx. 2 minutes, otherwise after the installation.

Please note that no forced installation takes place after a specified time.

4.5. Assignment of optional packages

The UEM Agent supports the optional package assignment feature introduced in Empirum version 21.0.3.

The administrator can set the distribution option "Optional" in combination with an end date in the Empirum Management Console, as well as via the SDK.

The user will be offered this package optionally for (de)installation and will only be executed with his consent if no expiration date is specified or the expiration date is in the future. When the expiration date is passed, the package is no longer optional and is processed according to the other settings.

Important Notice:

The following behavior should be noted

- Packages with the Optional distribution option and an expiration date are downloaded to the agent cache. Packages without an expiration date are downloaded only during the installation.
- Optional packages are always installed at the end of an installation process, that is, after the mandatory packages.
- If the user selects the "Install on shutdown" option, the selected optional packages are treated as required assigned packages during execution. If a reboot is required during installation, the agent will perform it according to the settings in the agent template and continue (de)installing all pending packages without a new user prompt.
- If the installation of multiple packages is triggered with "Start Selected", all packages will be installed. If a reboot is required during installation, the agent will perform it according to the settings in the agent template and continue (de)installing all pending packages without a new user prompt. The maintenance window is ignored.
- Older agents that do not yet include this functionality will interpret optional packages as normally assigned packages and execute them accordingly.

To enable the user to select optional packages, the Z dialog of the UEM Agent has been completely revised. The user sees the number of pending optional and required packages

in the overview. Furthermore, he sees the information whether the pending (de)installations require a restart.

The user has the possibility to postpone or start the (de)installation for all packages according to the configuration in the agent template.

You can reach the detail view by clicking the "Show details" button or the area around the number of packages. The detail view allows the selection of optional packages. Also, all selected packages can be started from there.

Important Notice:

- The Z dialog opens only if at least one required package is pending installation, or if the user manually checks for new software and at least one optional package is assigned
- Optional packages can only be selected and executed in the detail view

4.5.1. Optional packages kiosk

As of Empirum 21.0.3 (incl. all Hotfixes), it is possible to use the "AgentUI" -> "Show optional assigned Packages in Kiosk" option of the agent template to display an optional packages kiosk in addition to the classic software kiosk. This shows only assigned packages with the distribution option "Optional".

The behavior is as follows:

- All packages that have the "Optional" distribution option are displayed.
- Packages with the distribution option "Optional" and an end date are displayed until the date is reached.
- Only the assigned distribution options (Install/Reinstall or Uninstall) are allowed to the user.
- If both the Classic Software Kiosk and the Optional Packages Kiosk are enabled, the distribution options (Install and Uninstall) from both are merged.

5. Tips and Troubleshooting

5.1. If the Software Kiosk or Installation UI cnnot be displayed

The new user dialogs are based on a locally executed web page that is displayed with an embedded chrome browser. To do this, it is necessary to set the Internet properties so that no proxy is used locally and the local computer is classified as secure. In addition the "localhost" should not be part of the trusted site list. Use GPOs to ensure this for all computers.

5.2. Anti Virus Programs

Some antivirus programs prevent the installation or execution of the UEM Agent. Please use the AV options to exclude the"%Program Files%\Matrix42\UniversalAgent Framework" directory.

5.3. How to use the global silent switch

To ensure more standardized installations and minimize disruptions for the end user, there is the option to set a global “silent” key in the registry, which ensures that all packages installed via the software kiosk use the same mode. Ideally, this should be the “0” (/S0) mode. Progress is displayed to the user in the kiosk in larger increments.

With Empirum 17.0 Update 2 the Setup.exe will know the new value /S4. This suppresses all output except for process queries and prompts (user input).

Current restrictions:

- Mode “0” and “1” does not work for packages where an user entry is required.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\Agent]
"GlobalSilentLevel"=dword:00000004
```

In this example, all installations are set to “/S4”.

5.4. Shutdown the computer after OS installation

```
If the registry key below is set to 1, the "InstallAtShutdown"
mode is active for the agent. This can be set in UEMAgent.bat to
turn off the computer completely after the OS installation.
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"InstallAtShutdown"=dword:00000001
```

Example in UEM Agent.bat:

```
REG ADD HKLM\SOFTWARE\MATRIX42\AGENT /v InstallAtShutdown /t
REG_DWORD /d 1 /f
```

5.5. Hide the option "Install on Shutdown"

The option "Install on Shutdown" can be hidden from the user by disabling it in the agent configuration.

Alternatively, you can configure this via registry key, which has priority:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"AllowPostponeUntilShutdown "=dword:00000000
```

5.6. Suspend mode

With a registry key the UEM Agent can be set to a mode in which no polling, download or installation actions are performed. Internally this mode is used for the Auto Update. If required, this mode can be used, for example, to prevent the agent's possible data consumption when establishing a VPN connection via a satellite telephone.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\Agent]
"Suspenduntil" (STRING) = end date of suspend mode
```

The value specifies the end of the suspend mode. For example, 2018-12-24T18:15. Any specified value that is not recognized by the agent as an ISO time value in the past will pause the agent.

5.7. Script files check for kiosk packages

The check for script files on the depot server can be activated via the following registry key:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"UseCheckFileForKiosk"=dword:00000001
```

5.8. Configure behaviour if Windows requires a reboot

The UEM Agent detects if Windows Update requires a reboot.

You can configure the behaviour with the following registry value.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"WindowsUpdateRebootCheck"=dword:00000002
```

Possible values:

- 0: The agent ignores the required reboot and continues its tasks.
- 1: The agent suspends its tasks until a reboot is performed or the reboot request is withdrawn. (This is the behaviour like before version 2009)
- 2 or value not available: After a delay the agent initiates the reboot behaviour as configured in the agent configuration.

The delay until the reboot behaviour is initiated can be configured by the following registry value.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"WindowsUpdateRebootDelaySeconds"=dword:00000120
```

The value is interpreted in seconds. If the value is missing the agent waits 15 minutes.

5.9. Set Rights of Agent Cache

The behaviour of setting the agent cache rights can be defined by the following registry key:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"SetNTFSCacheRights"=dword:00000001
```

Possible Values are:

- 0: No rights are set by the agent
- 1: The standard access rights are set by the agent (default behaviour)
- 2: The standard access rights are set by the agent, but no rights for "Everyone" are set to the "User" folder.

The default access rights are:

- "System" user and "local administrators" have full access to all folders.
- Additionally, "everyone" has the following rights:
  - Read right on the folders "Packages", "UserValues", "Log", "EmpInv" and "User".
  - Write right to the folders "Log" and "EmpInv".
  - Execute right to the folder "User".

5.10. Behavior in case of problems during package download

If there are problems while downloading a package (e.g. temporary network errors, access problems), the Agent can be configured to retry the process automatically. To do this, you can set the number of retries (default: 5) and the pause between retries in seconds (default: 60). The default can be overwritten in the registry:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"RetryTransferRepetitionLimit"=dword:00000005
[HKEY_LOCAL_MACHINE\SOFTWARE\Matrix42\AGENT]
"RetryTransferPause"=dword:00000060
```

6. Customization of the user interface

The colors can be adjusted in the file UI.json. A template of this file can be found in the folder %Programme%\Matrix42\UEM Agent\CustomizingSamples. The file is automatically made available to all clients by being stored in the user directory.

6.1. Kiosk

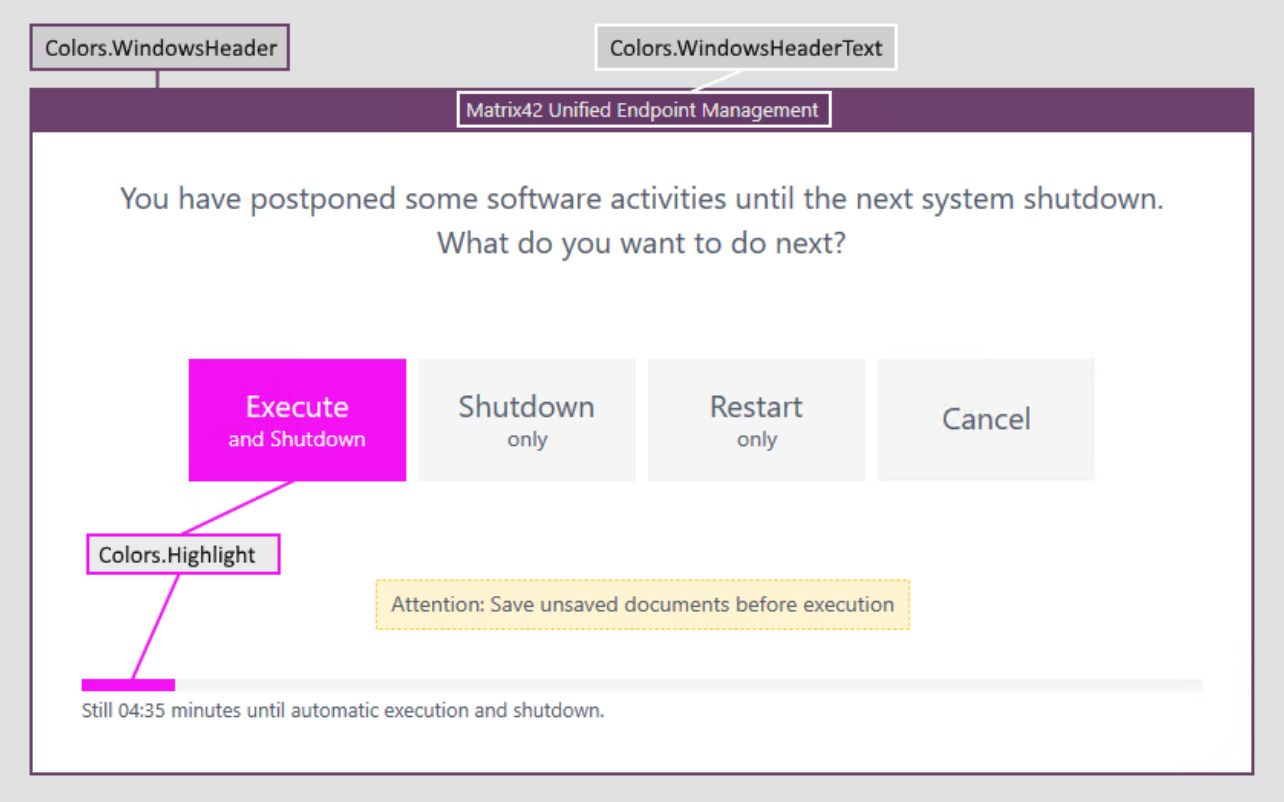
To apply the settings, click the "Update group memberships and software list" button in the kiosk settings window.

Color codes or standard color names in English can be used.

To change graphics, they must be stored on the Empirum master server in the Empirum\Configurator\user directory and, if necessary, distributed to the subdepot servers via synchronization. In the current version the logo can be adapted. The kiosk expects an image with the name Customlogo.png, ideally in size 301x44, other sizes are scaled to the width.

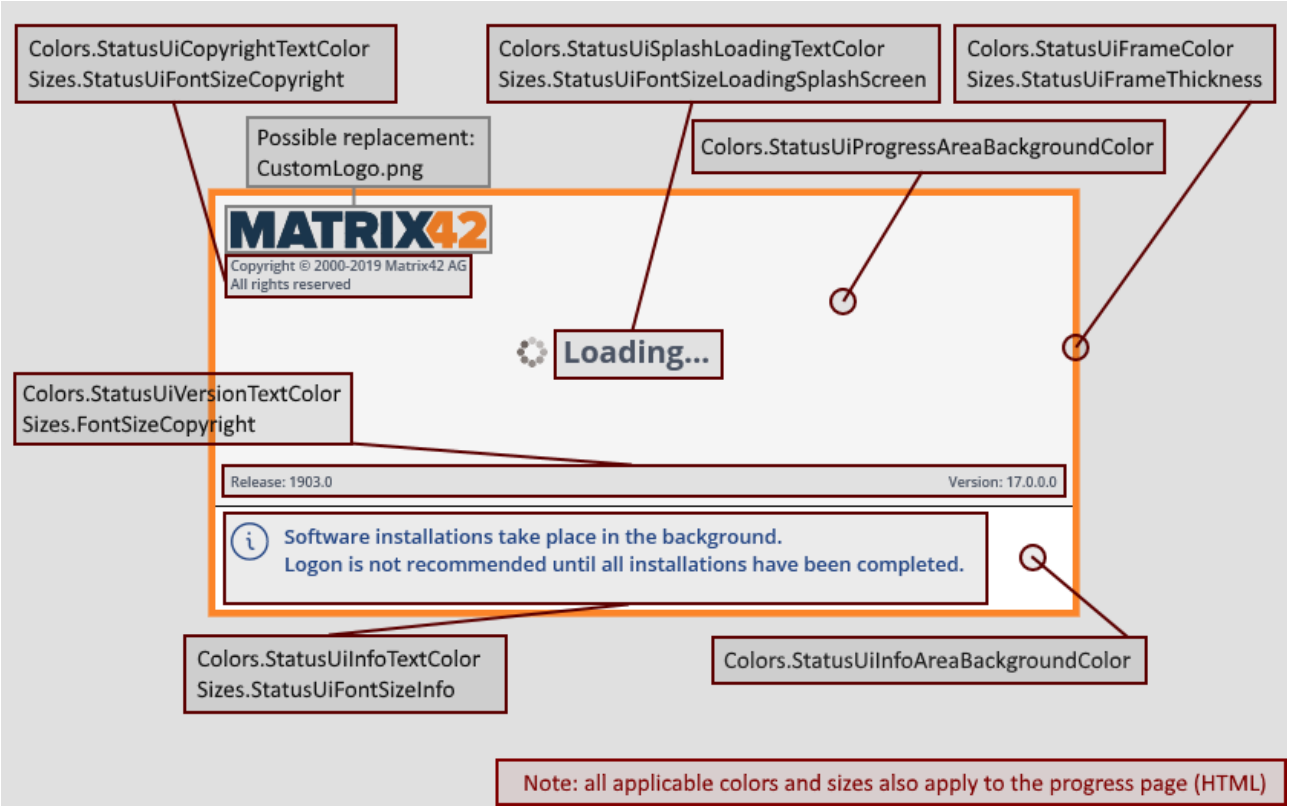
The file M42UEM_UI_Custom_Glossary.xml is available for changing the texts. If the agent finds this file in the user directory, the contents of the file overwrite the texts of the standard glossary.

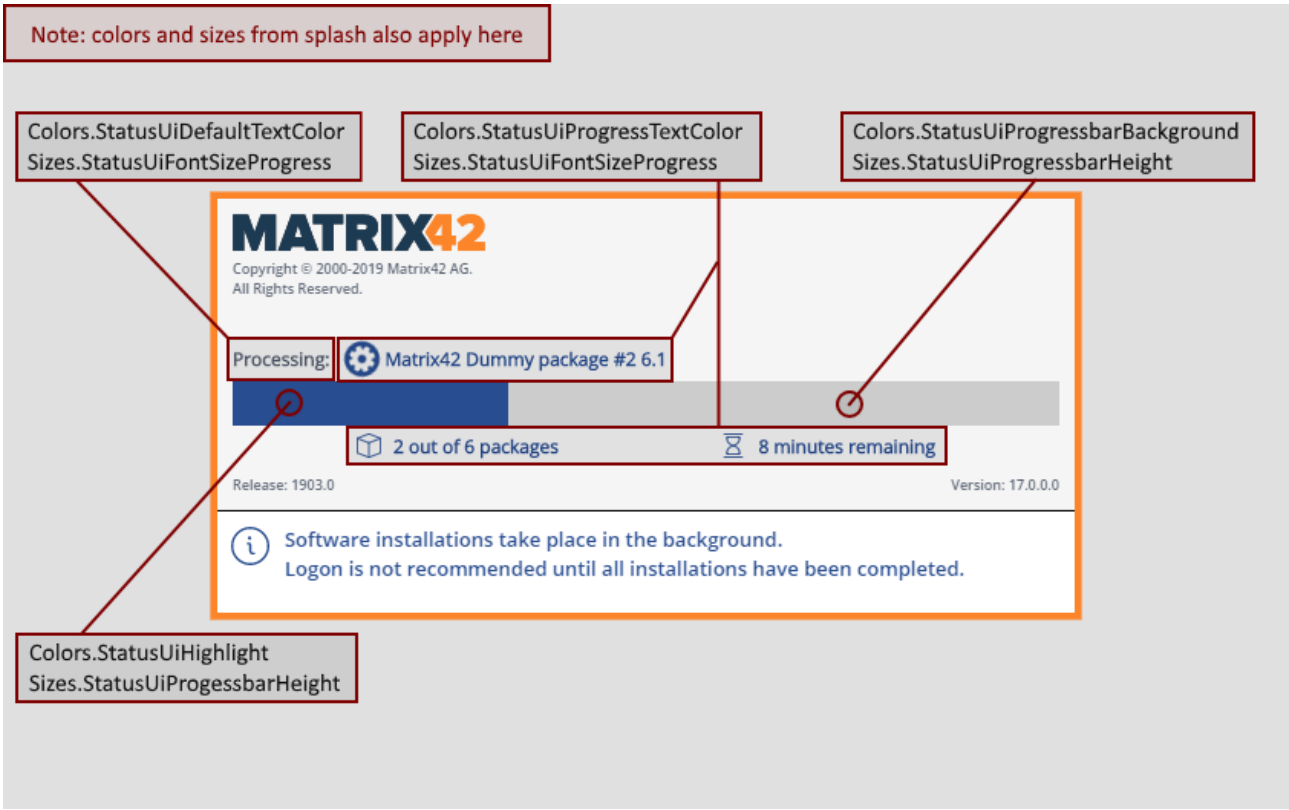
These adjustments also apply to the "Install at Shutdown" dialog.



6.2. StatusUI

You can use the StatusUI_Custom_Glossary.xml file to change the texts. The same rules apply for the file and the logo as for the kiosk.


Colors.StatusUiCopyrightTextColor
Sizes.StatusUiFontSizeCopyright
Colors.StatusUiSplashLoadingTextColor
Sizes.StatusUiFontSizeLoadingSplashScreen
Colors.StatusUiFrameColor
Sizes.StatusUiFrameThickness
Possible replacement:
CustomLogo.png
Colors.StatusUiProgressAreaBackgroundColor
MATRIX42
Copyright © 2000-2019 Matrix42 AG
All rights reserved
Loading...
Colors.StatusUiVersionTextColor
Sizes.FontSizeCopyright
Release: 1903.0
Version: 17.0.0.0
Software installations take place in the background.
Logon is not recommended until all installations have been completed.
Colors.StatusUiInfoTextColor
Sizes.StatusUiFontSizeInfo
Colors.StatusUiInfoAreaBackgroundColor
Note: all applicable colors and sizes also apply to the progress page (HTML)


Note: colors and sizes from splash also apply here
Colors.StatusUiDefaultTextColor
Sizes.StatusUiFontSizeProgress
Colors.StatusUiProgressTextColor
Sizes.StatusUiFontSizeProgress
Colors.StatusUiProgressbarBackground
Sizes.StatusUiProgressbarHeight
MATRIX42
Copyright © 2000-2019 Matrix42 AG.
All Rights Reserved.
Processing:
Matrix42 Dummy package #2 6.1
2 out of 6 packages
8 minutes remaining
Release: 1903.0
Version: 17.0.0.0
Software installations take place in the background.
Logon is not recommended until all installations have been completed.
Colors.StatusUiHighlight
Sizes.StatusUiProgessbarHeight

7. Installation of the UEM Agent via Silverback

To integrate a new Windows 10 computer into the management, the initial distribution of the UEM Agent with the enrollment in Silverback is an alternative. During the installation a "Mini-Inventory" is created and transmitted to the Empirum Server. The latter can use the "filter actions" to assign the computer directly to a group in order to install further standard software.

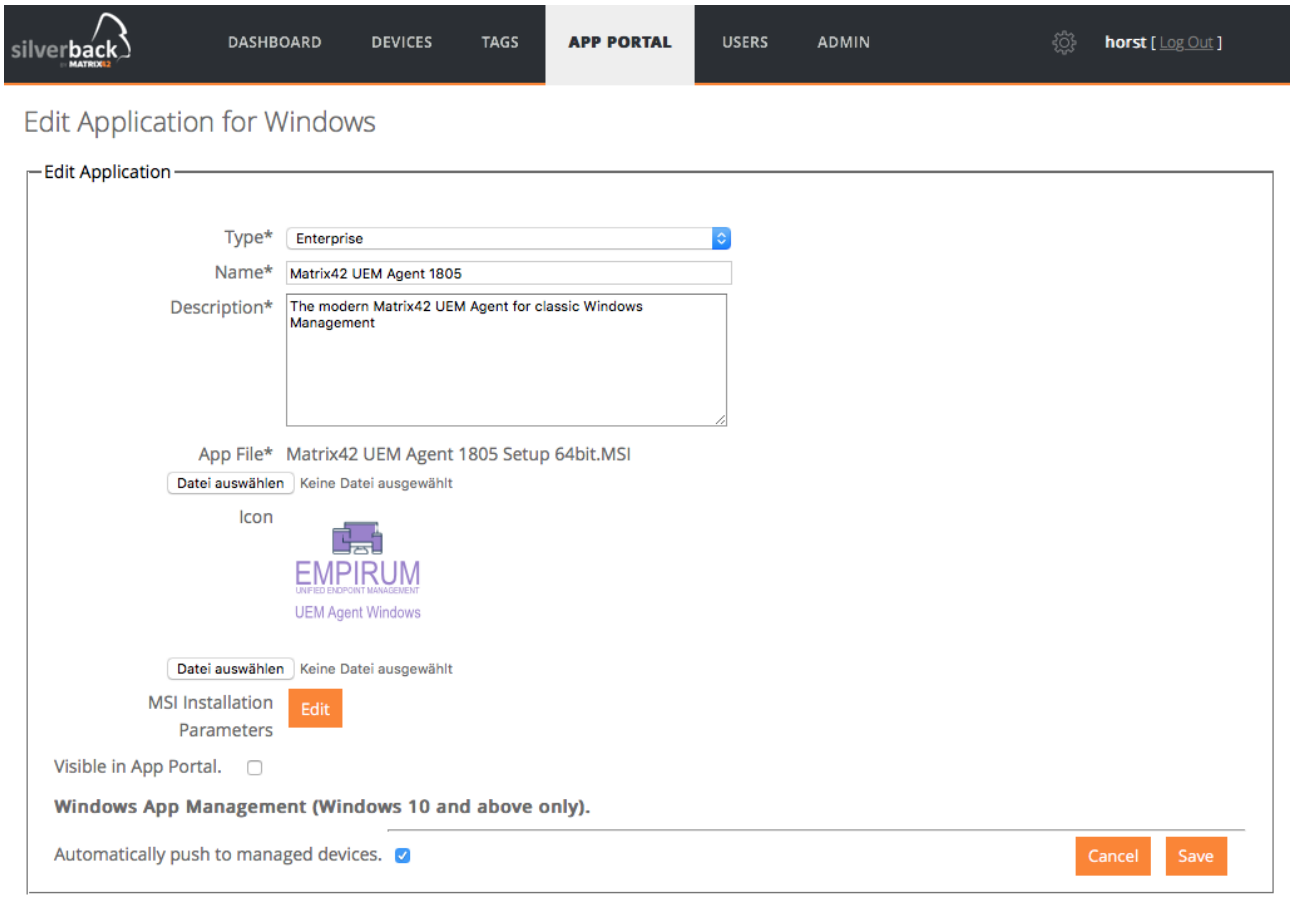
7.1. Requirements

- Computers require Windows 10 1803 or later
- The UEM Agent Platform (x64) must match the used Windows version (x64)
- The UEM Agent must be able to connect to the Empirum (depot) server. Usually this will be done via https.

7.2. Importing the UEM Agent MSI into Silverback

In the admin area of Silverback Console, navigate to "App Portal" - "Windows" and select "New Application".

Import the latest UEM Agent MSI and enter the installation parameters.



Sample MSI installation parameters:

```
/quiet /l*v "c:\Windows\Temp\UEMAgentMSI.log"
Server="srv001.abc.com" User="dom\agentuser"
Password="E2BA23794CCCBDB7ACD5CBBC8A728E1EA057EBDF4577507EB2421C7E
```

```
2B8327CA4593CE11B902D9B623167BED8BDC5A5F" Protocol="https"
Port="443"
```

The password is encrypted with the Empirum tool Empcrypt.

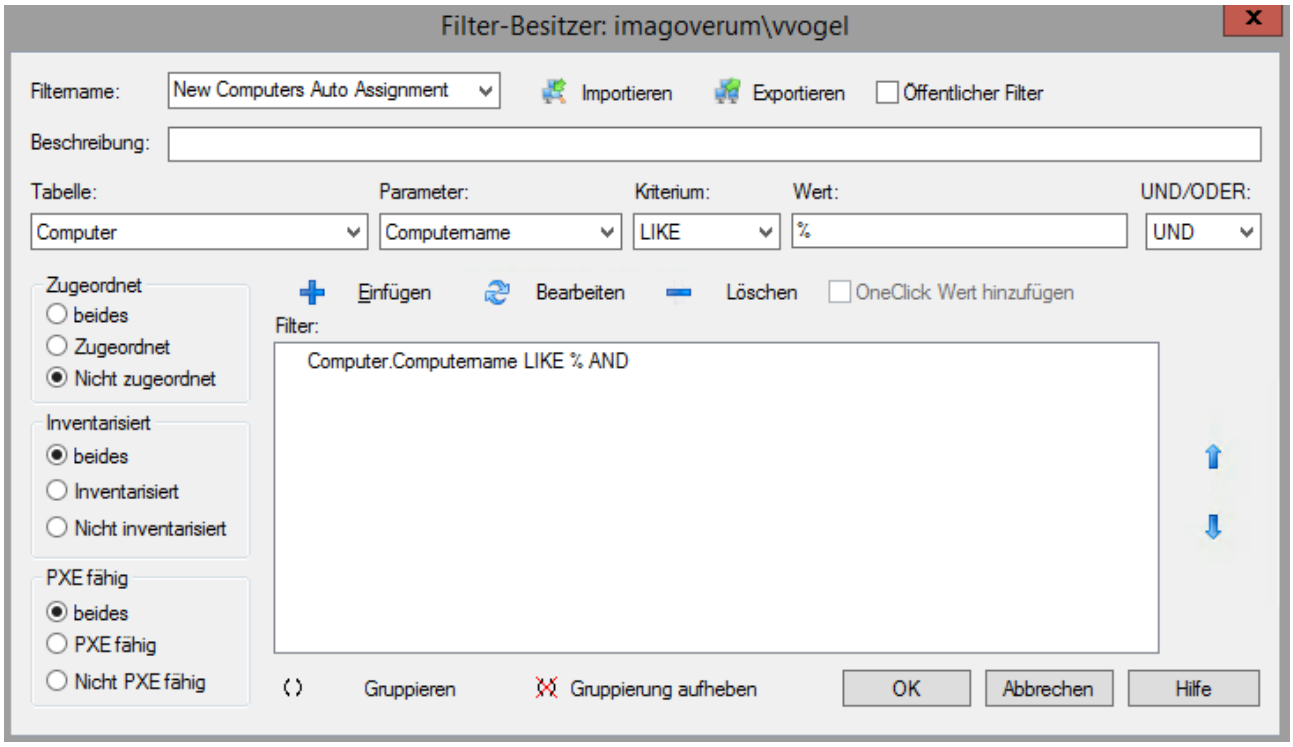
After successful testing, select "Automatically push to managed devices" to automatically install the UEM Agent on all Windows 10 computers during enrollment.

Note: If Windows UAC is switched on, the UEM Agent is only started automatically after a restart.

7.3. Automatic assignment of newly enrolled computers

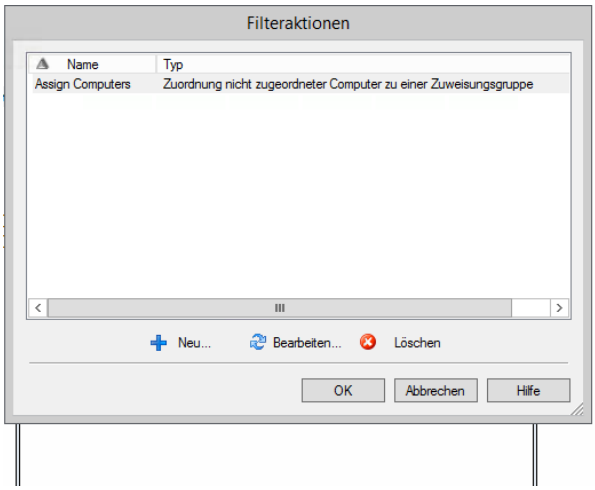
To provide computers with additional basic software immediately, all newly enrolled computers can be assigned to an assignment group with a "filter action". This also assigns a default agent template.

Create a new filter (example: New Computers Auto Assignment). In the simplest case, use "Computer name like %" and "Not Assigned" as criteria to assign all new computers. Of course, you can also use other filters with naming rules for assignment to different groups.



Create an assignment group in which all new computers are to be assigned and set a default Agent Template and the desired software packages (for example Inventory).

Select "Filter actions" in the context menu of the filter and create a new action "Assignment of unassigned computers to an assignment group". There you select "Only for new objects" and "Activate" in the further course of the wizard.


Filteraktionen
Name
Typ
Assign Computers
Zuordnung nicht zugeordneter Computer zu einer Zuweisungsgruppe
Neu...
Bearbeiten...
Löschen
OK
Abbrechen
Hilfe

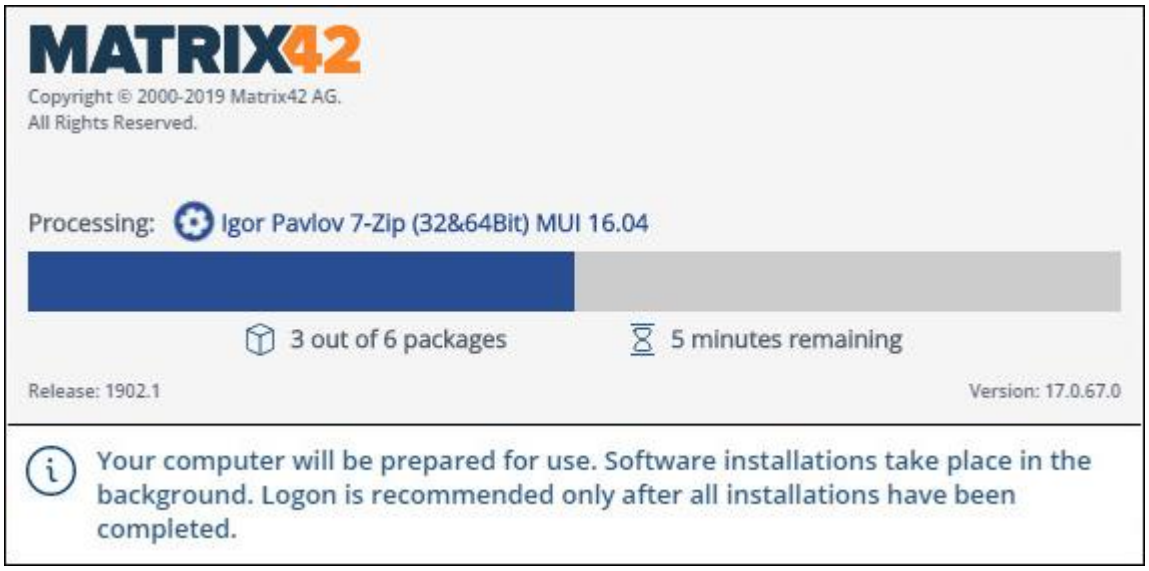
Filteraktion
MATRIX42
Aktionstyp
Alarm
Software-Verteilung über Konfigurationsgruppen
Software-Verteilung über Zuweisungsgruppen
Zuordnung nicht zugeordneter Computer zu einer Konfigurationsgruppe
Zuordnung nicht zugeordneter Computer zu einer Zuweisungsgruppe
Nur für neue Objekte
Beim ersten Ausführen alle Objekte einbeziehen
matrix42 AG
< Zurück
Weiter >
Abbrechen
Hilfe

8. Pre-Logon Window

With version 1901.0 (TP) of the UEM Agent, a new application is stored in the installation folder of the UEM Agent package. The new StatusUI.exe replaces the existing ErisLogonUI.exe.

The new StatusUI application shows the number of pending packages, a progress bar based on the estimated times of the packets and the remaining time until a user can log in without being disturbed by a restart in later operation.

Packages which are flagged with "Hide" in the distribution commands will be shown as "We are preparing your system." This text can be individually adapted under the key "WEB_StatusUiHiddenPackageName" in the file StatusUI_Custom_Glossary.xml



9. Validation of packages before installation

The UEM Agent offers the possibility to validate packages before the installation. The validation is done via SHA-256 checksums which are generated per package.

The program CreateHash creates a PackageHashes.json file in the user directory on the Empirum Master Server. The file serves as a checkpoint for the UEM Agent to verify the downloaded package on a client.

Starting with Empirum Version 19.0 the creation of PackageHashes.json files on the Empirum server is automated by a service. This makes the program CreateHash obsolete.

Use of the CreateHash at first use:

- Copy the unpacked directory CreateHash to \Empirum\AddOns\.
- Execute the batch file "Create Hashes for all packages.bat".
- The program parses the SWDepot.dds according to the specified package paths and creates the PackageHashes.json file. The file is stored in the Configurator\User directory.
- The UEM Agent generates a hash value for the package to be installed before installation.
- The UEM Agent checks the generated hash value against the hash value in the PackageHashes.json file to ensure that the package is valid.
- If there are changes in the repository, the batch file "Create Hashes for all unhashed packages.bat" can be executed. The execution of the second batch file updates the PackageHashes.json file.

9.1. Enabling package validation for UEM Agent

To activate validation on the client side, set "CheckPackageHash" as DWORD to a value greater than 0 in the registry.

Example:

```
REG ADD HKLM\SOFTWARE\MATRIX42\AGENT /v CheckPackageHash /t
REG_DWORD /d 1 /f
```

Behavior of the UEM Agent:

- If Key CheckPackageHash does not exist, no validation is performed.
- If Key CheckPackageHash exists, the value is set to 0, then no validation.
- If Key CheckPackageHash exists, the value set to 1, then Validation.
- If the validation of the packages via hash is activated, the result of the validation can be viewed in the SWDepot log of the EMC under the mode "Validation Status".

9.2. Effect of incorrect validation of packets

If the package validation detects a difference between the hash values on the server and on the client, the FailedInstallationRetries counter is incremented for this package. This behavior can be controlled specifically with the CountHashValidationErrors key as DWORD.

- If the key does not exist or has a value unequal to "1", the counter is not incremented.

- If the key exists and has the value "1", the counter is incremented.

Example call for a behavior change:

```
REG ADD HKLM\SOFTWARE\MATRIX42\AGENT /v CountHashValidationErrors
/t REG_DWORD /d 0 /f
```

9.3. Observing the sequence

If, in the order of the UEM Agent package list to be processed, a package has a negative validation, all subsequent packages are no longer executed until the validation is positive.

9.4. Enabling package validation via Empirum console

Starting with Empirum version 19.0.0, the validation of packages can be activated via the console. The setting is available under Configuration/Software Management/Empirum Agent/Software Depot in the tab Other settings for the UEM Agent from version 1903.0.

With this option you can activate the check of the packages to be distributed.

The check is performed before the installation by comparing the hash value generated on the server with the hash value generated locally before the installation.

10. Check for signing

The UEM Agent offers the possibility to check the swdepot.dds for a valid signature before downloading and installing packages.

The program CreateHash signs the file PackageHashes.json and the swdepot.dds in the user directory on the Empirum Master Server. The files are checked for their digital signature on a client after the download.

Starting with Empirum version 19.0.2 the creation and signing of the PackageHashes.json as well as the signing of the swdepot.dds on the Empirum server is automatically performed by a service.

Use of the CreateHash at first use:

- Copy the unpacked directory CreateHash to \Empirum\AddOns\.
- Open the "Create Hashes and Sign Files.bat".
- Add your thumbprint certificate to the -th parameter.
- Add the path to your PackageHashes.json und swdepot.dds
- Run the batch file.
- The files PackageHashes.json and swdepot.dds get a digital signature at the end of the files. The files remain readable.

The batch file must always be executed if changes are made in the depot.

10.1. Activate signing for UEM Agent

To activate signing on the client side, set the following in the AgentTemplate under the thumbprint (without spaces and special characters)

/AgentTemplate/SoftwareDepot/SignatureThumbprint

Example:

```
<SignatureThumbprint>2774e273994767b7505123b9f363ec072bbbfec8
</SignatureThumbprint>
```

11. Installation and login after reboot on Full disk encryption

During a pending reboot, the UEM agent checks whether " Disc Encryption" is installed on the client. If the prerequisites are fulfilled, it prepares the system to start with automatic login after the reboot in order to continue the installation. This process is repeated for each subsequent reboot requested by a package. At the end, the agent resets the system to the default state, "Disc Encryption" is active and expects the user to log in at the next system startup.

Supported products are

- BitLocker
- EgoSecure

The desired product must be set in the agent template.

In addition, a valid pre-boot authentication file (PBA) must be assigned in the agent template for EgoSecure. After the assignment it gets the same name as the template and is stored in the subdirectory "Endpoint Security" of the user directory.

12. Alternative Package Sources

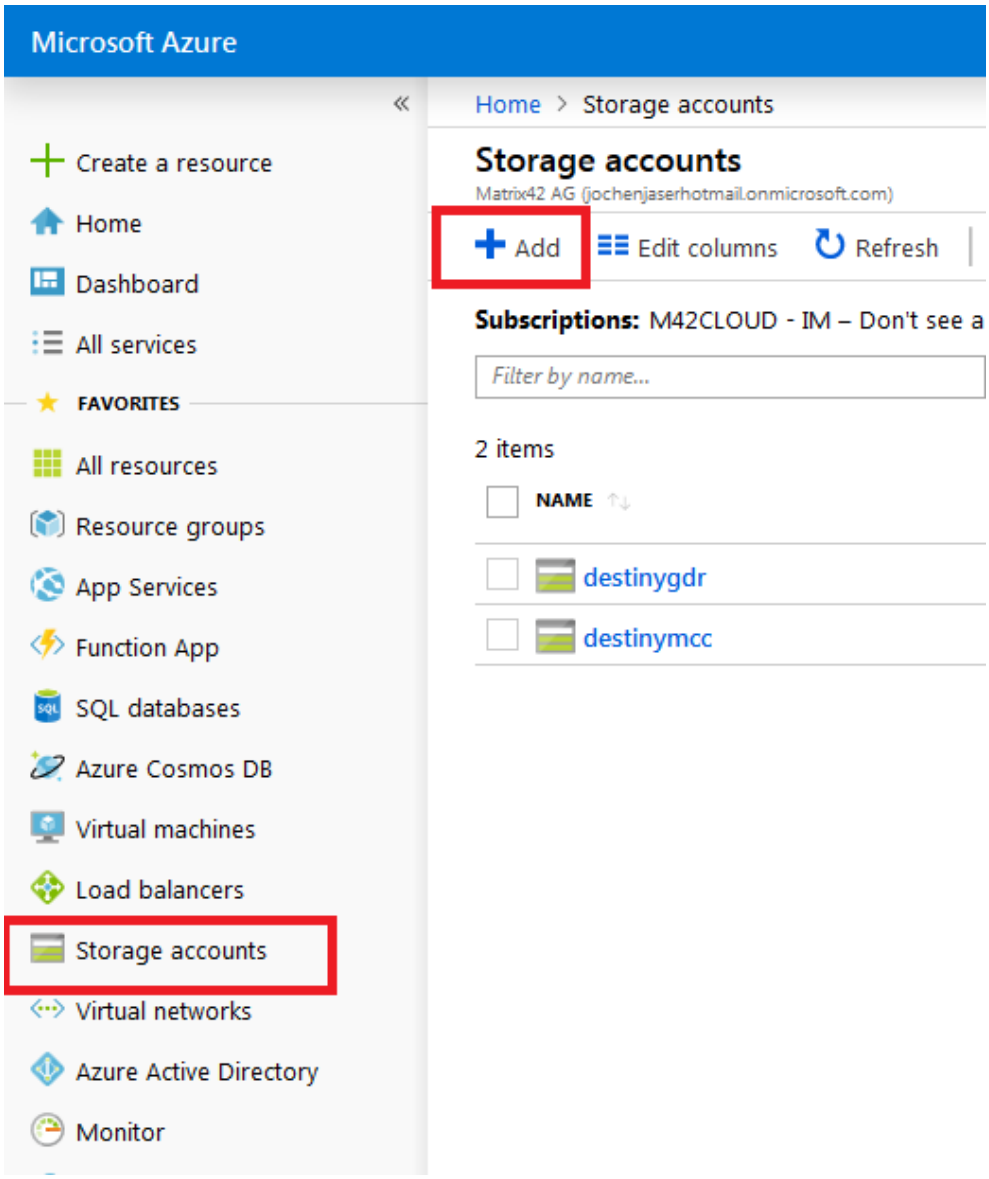
Using alternative package sources, the agent can load the mass data for software packages from a special source. Currently Microsoft Azure Blob is supported.

12.1. Loading Packages into Microsoft Azure Blob Container

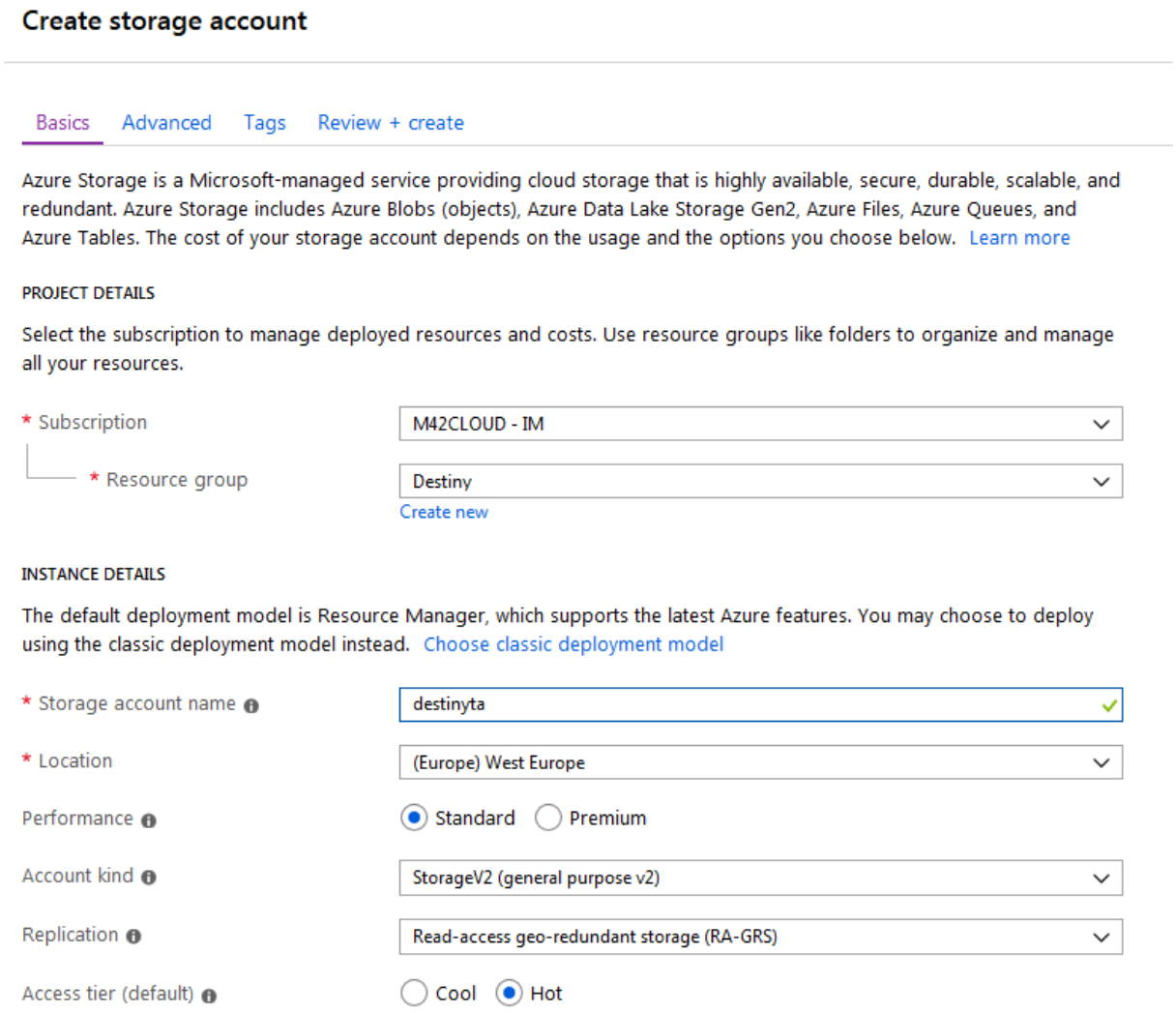
In preparation, the packages must be loaded into a cloud storage and the access data must be entered in the agent configuration. (See Matrix42 Online Help)

12.1.1. Create a Storage Account

- Log in to the Azure Portal
- https://azure.microsoft.com/en-us/account/
- In the Resources menu, select "Storage Accounts" and create a new storage account according to the Azure specifications.

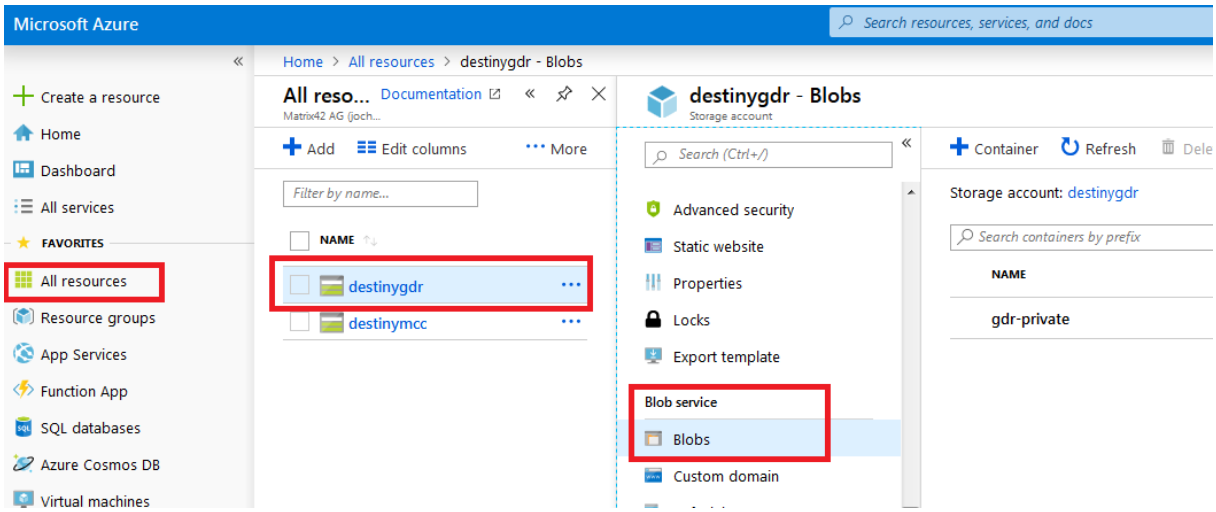


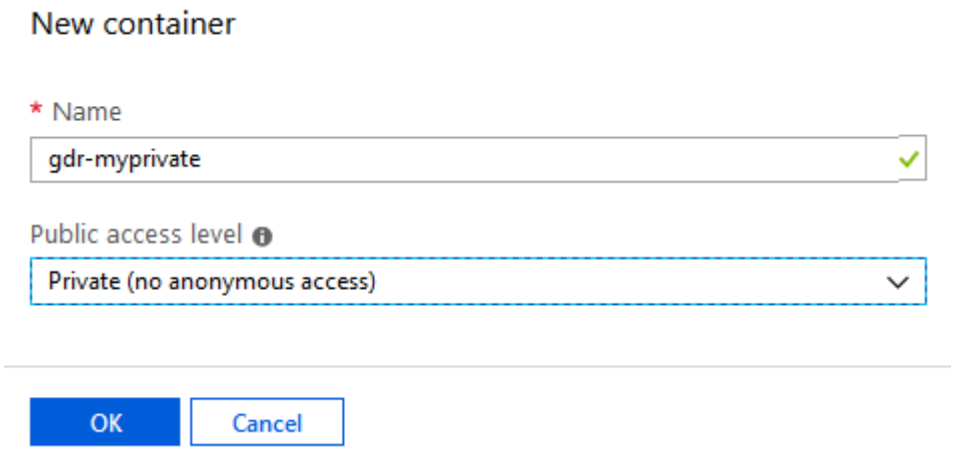
- Fill in the form and confirm the creation of a new storage account by clicking "Review and Create" and then "Create".



12.1.2. Create Blob Container under the newly created Storage Account

- Select the storage account you created in step 2.
- Select the menu item "Blobs" under "Blob service".
- To create a blob container click on "+ Container".
- Select "Private (no anonymous access)" as Public access level.
- The blob container is created





12.1.3. Copy software packages to Azure Blob Container

- To make packages available in the Azure Blob Container, the packages must be uploaded manually from the Empirum server to the cloud.
- We recommend to use the tool "Microsoft Azure Storage Explorer".
- Copy all packages below Empirum$\Configurator\Packages into the blob container excluding the patch management data (directory PatchManagement).
- Now the feature can be activated in the agent configuration from Empirum version 19.0.1.

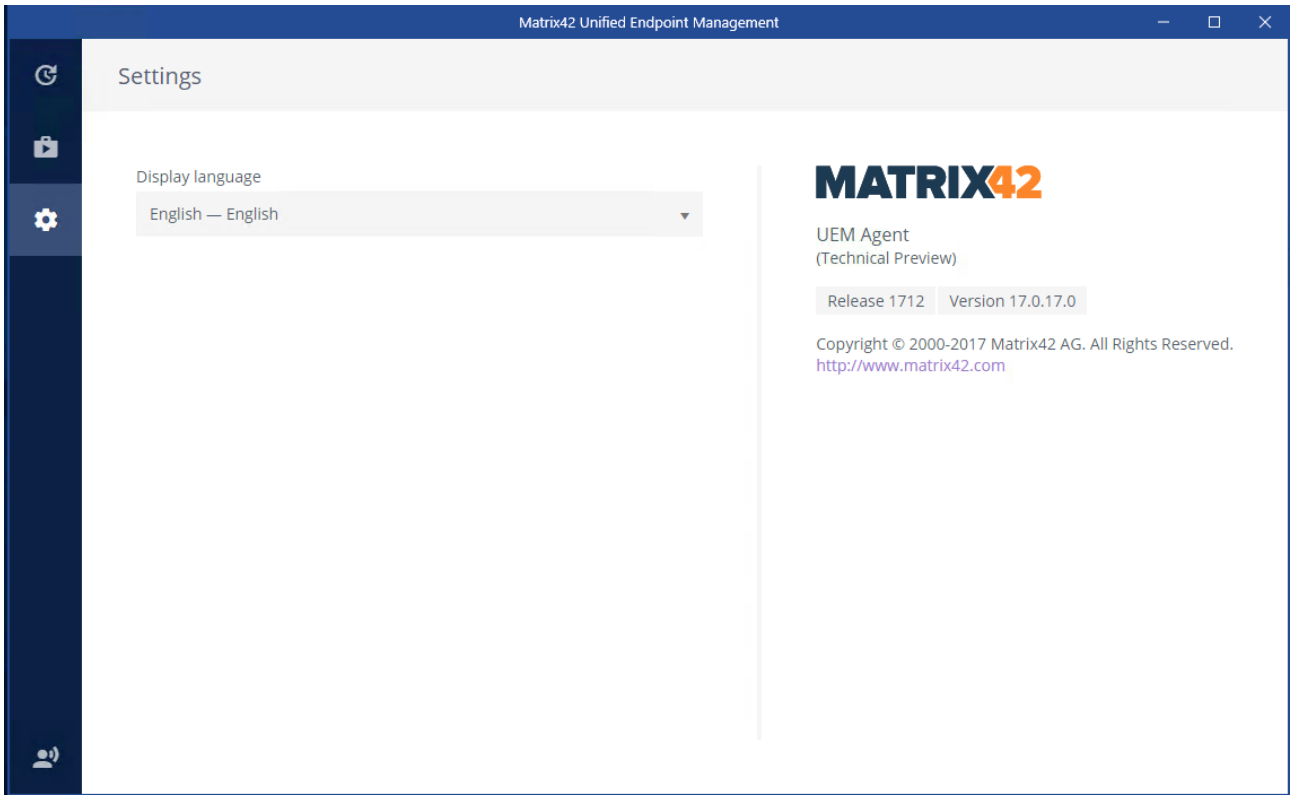
12.1.4. Important notes

- The UEM Agent will continue to download configuration and metadata (DDS, INI, DCC, etc.) via the current Empirum server.
- The packages are obtained from the Azure Blob Container. Currently there is no fallback. If the package does not exist in the blob container, the UEM agent will not try to get the package from the Empirum server.
- Currently packages have to be uploaded manually into the Azure Blob Container.

- Patch management data is still retrieved from the Empirum server.

13. Info page

The information page displays the version in use and allows to change the diplay language.

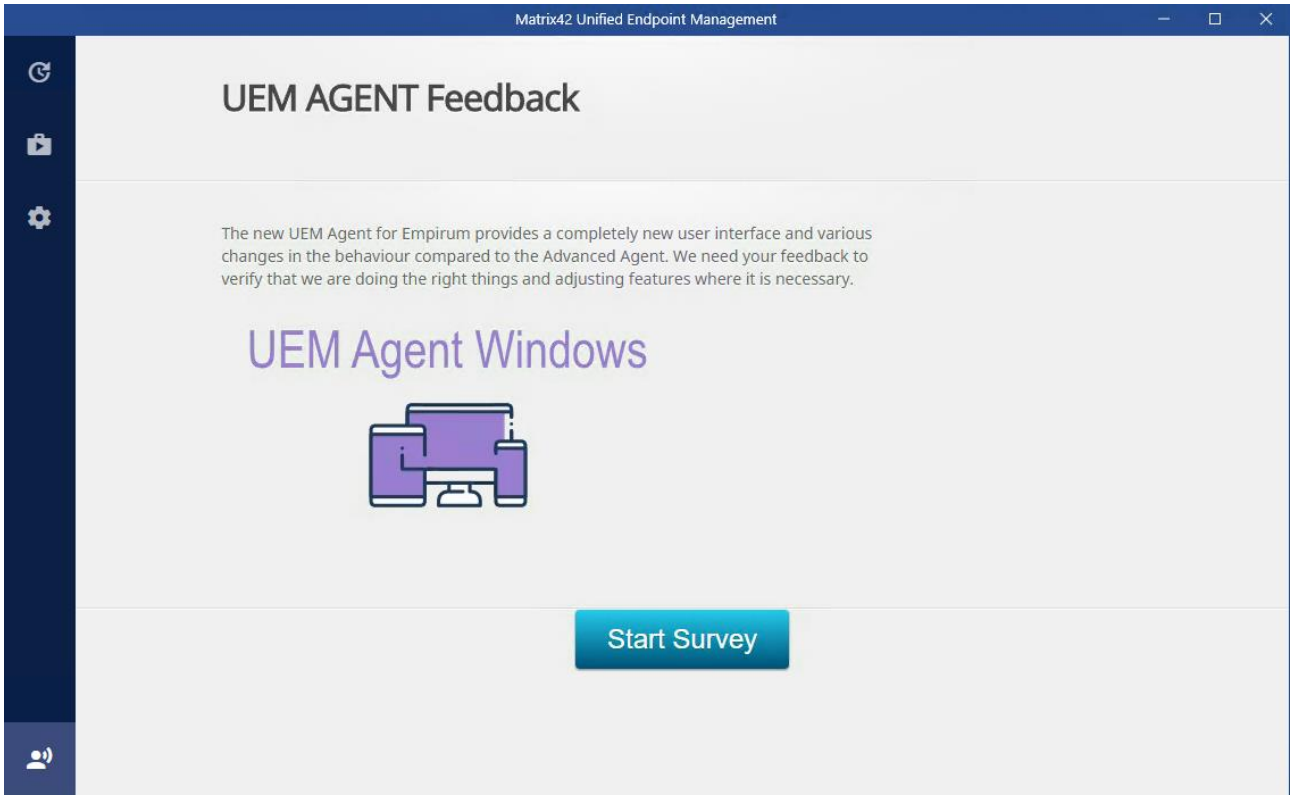


14. Survey

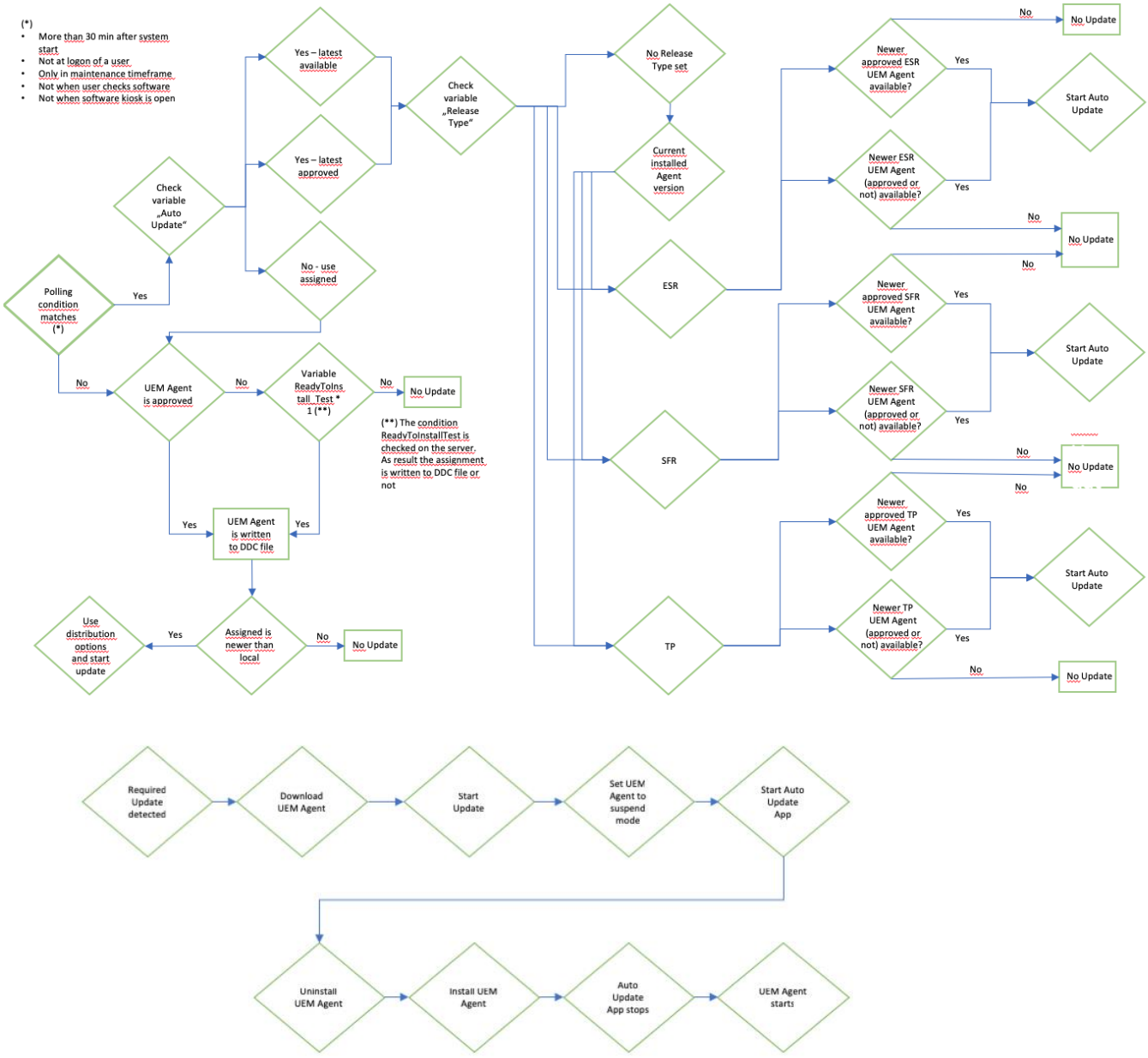
The survey page is only visible with an working internet connection.

The survey can be removed by the administrator or a separate feedback page can be included by entering a URL. If the registry key Feedback_URL does not exist under HKLM\Software\Matrix42\Agent, the standard Matrix42 survey is displayed. If the key is empty the item is completely hidden.

Your own web pages must be configured to be displayable within a frame.



15. Auto Update Flow Diagram



16. UEM Depot Management

16.1. General

The new UEM Depot Management offers a new UEM Depot Sync based on the Matrix42 Universal Agent Framework. With this framework, synchronization can now be performed via SMB as well as via HTTP(S).

The final status of the last synchronization can be viewed for each run in the EMC in the tab "Empirum Server Monitor".

Synchronizations, which sometimes have a longer runtime, send an additional information about the start of the sync run. For more information, see Available synchronizations.

16.2. Installation

For the initial setup of the UEM Depot Sync the following steps have to be performed:

- Configure the server as a subdepot
  - Computer properties > Computer role > Empirum Depot Server
- Installing the UEM Agent
- Configure variables
  - SUBDEPOT > ROOTPATH
  - SUBDEPOT > USER_1
  - SUBDEPOT > PASSWORD_1
- Install the current version of the "Empirum Subdepot" package on the server (create the directories and set the access rights)
- Configuring the Sync Jobs
  - See section Configuration

16.3. Configuration

The UEM Depot Sync can be configured using variables. A detailed description of the variables can be found here The Sync Jobs described below are supported.

It is recommended not to let the agent poll the depots too often, as this will lead to an increased load on the system. It is recommended to set an interval of at least 60 minutes.

16.3.1. Available synchronizations

16.3.1.1. CLIENTDATA

This sync job configures the synchronization of the client data directories EmpInv and. Log.

16.3.1.2. CONFIGURATIONDATA

This sync job configures the synchronization of the configuration data like DDC, DDS, INI, Configuration/User. Only the DDC and INI files of the computers that are assigned to the Subdepot are synchronized.

16.3.1.3. OS_CONFIGDATA

This sync job configures the synchronization of the OS metadata. Only the OS metadata of the computers that are assigned to the Subdepot will be synchronized.

Additionally, the file DeviceMappings.xml is copied to the Values folder.

16.3.1.4. OS_IMAGEDATA

Empirum Version 21.0.2 or higher is required as backend!

This sync job configures the synchronization of the OS mass data.

The start time of the run is visible in the log in the Empirum Management Console.

Hashsums will be calculated for OS mass data. The hashsums are stored in the file "OsProperties.json" in the "\Empirum\EmpInst" folder. They are used for a quicker synchronization and are required for the job to run.

For imported Operating Systems the (re-)generation of the hashsums can be started from the context menu.

Note

In order to force a complete new hash calculation at the next run, you can set the registry value RenewPackagesHashesOnce:

```
REG ADD HKLM\SOFTWARE\MATRIX42\
Platform\Service\Extension\ObjectStore\Data /v RenewOsHashesOnce
/t REG_DWORD /d 1 /f
```

16.3.1.5. PACKAGES

This sync job configures the synchronization of the package directory.

The start time of the run is visible in the log in the Empirum Management Console.

The files "SwDepot.dds" and "PackageHashes.json" are required. These are synchronized by the sync job "ConfigurationData". If they are not available, the synchronization of packages is not possible.

In order to speed up the matching of packages from source and target, the sync job uses the PackageHashes.json (see section Validation of packages before installation) During synchronization, the job checks whether a package is listed in the PackageHashes.json and whether the stored hash values of source and target match. If they are identical, no files are copied. If they differ, the package is synchronized. All packages that are not listed in PackageHashes.json are synchronized.

Note

If a package listed in the PackageHashes.json is changed in the target directory (e.g. files are deleted or edited), this change will not be detected by the check of the saved hash value and there will be no synchronization to the source directory!

In order to force a complete new hash calculation at the next run, you can set the registry value RenewPackagesHashesOnce:

```
REG ADD HKLM\SOFTWARE\MATRIX42\
Platform\Service\Extension\ObjectStore\Data /v
RenewPackagesHashesOnce /t REG_DWORD /d 1 /f
```

The following folders from the packages folder will not be synchronized:

- Matrix42\EndMessage
- Matrix42\OS-Message
- Matrix42\LinuxAgent
- Matrix42\EmpirumAgent_for_Mac
- Matrix42\InventoryMacOSX
- PatchManagement (Part of the sync Job "Patches")

The following files from the packages folder will not be synchronized:

- Matrix42\SyncJobs\SyncTemplate.inf

16.3.1.6. PATCHES

This sync job configures the synchronization of the patch directory.

The start of the run is transmitted at the beginning.

16.3.2. Exclude directories

Using the SYNC_*_EXCLUDED_FOLDERS variable, directories can be excluded from the sync. The syntax rules described below apply here.

The following directory and file structure are assumed for the examples:

ROOT\

Test1.txt

TEMP1\Folder1\Test1.txt

TEMP2\TEMP1\Folder2\Test2.png

TEMP3\TEMP1\Folder1\Test2.txt

16.3.2.1. Rule 1: Backslash \ for relative paths

If a directory starts with a backslash \ , the excluded path is relative to the Sync Root Folder (see variable ROOTPATH). Otherwise, all folders containing this part of the path are excluded

Example

- "\TEMP1\Folder1" will only exclude the folder "ROOT\TEMP1\Folder1" and the "Test1.txt" inside.
- "TEMP1\Folder1", on the other hand, will exclude all "Folder1" that are located under the root directory "ROOT", but also under all subdirectories and have "TEMP1" as their parent folder.

- Specifically, the folders "ROOT\TEMP1\Folder1" and "ROOT\TEMP3\TEMP1\Folder1" and the files they contain would be excluded.

16.3.2.2. Rule 2: Filtering with wildcards *

You can use * to specify placeholders for folders.

Example

- "*\TEMP1" würde die Ordner "ROOT\TEMP2\TEMP1\Folder2", "ROOT\TEMP3\TEMP1\Folder1" und alle darin enthaltene Dateien ausschließen.
  - Specifically, the folders "ROOT\TEMP1\Folder1" and "ROOT\TEMP3\TEMP1\Folder1" and the files they contain would be excluded.

16.3.3. Exclude files

Files can be excluded from the sync using the SYNC_*_EXCLUDED_FILES variable. The syntax rules described below apply here.

For the examples the directory and file structure of section Exclude directories is assumed.

16.3.3.1. Rule 1: Backslash \ for relative paths

If a directory or file starts with a backslash \, the excluded path is relative to the Sync Root Folder (see variable ROOTPATH). Otherwise, all files containing this part of the path are excluded.

Example

- "\Test1.txt" will only exclude the file "Test1.txt" in "ROOT.
- "\TEMP1\Folder1\Test1.txt" will only exclude the file under "ROOT\TEMP1\Folder1\Test1.txt.
- "Test1.txt" would exclude all files "Test1.txt" below "ROOT" and its subdirectories. Specifically, the files "ROOT\Test1.txt" and "ROOT\TEMP1\Folder1\Test1.txt" would be excluded.

16.3.3.2. Rule 2: filtering with wildcards *

You can use * to specify placeholders for files.

Example

- Using "*.txt" would exclude the files "ROOT\Test1.txt", "ROOT\TEMP1\Folder1\Test1.txt" and "ROOT\TEMP3\TEMP1\Folder1\Test2.txt".
- Using "Test2.*" would exclude the files "ROOT\TEMP2\TEMP1\Folder2\Test2.png" and "ROOT\TEMP3\TEMP1\Folder1\Test2.txt".

16.4. Protocoll und Master Server

The protocol and the Master Server are configured by means of an assigned Agent Template: In concrete terms, this means that the agent template of the UEM agent installed on the sub depot is used for the connection to the master server. This also includes the functions of bandwidth throttling and minimum bandwidth.

16.5. Open points

The following features are currently not yet supported by UEM Depot Management:

- Installation time frames are not considered
- Options in the scheduler:
  - "End after x occurences"
  - "Stop job if the end time is reached"
- In the log there is always a zero for "Size of target directory".

16.6. Troubleshooting

The UEM Depot Sync is writing log files in the folder *%ProgramData%\Matrix42\Logs\UEM Depot* on the depot server. They contain detailed informations, warnings and errors for the synchronizations.

Matrix42 AG
Elbinger Straße 7
60487 Frankfurt am Main
Germany
Tel.: +49 69 6677 3838 0
Fax: +49 69 6677 8865 7
E-Mail: info@matrix42.com
Web: https://www.matrix42.com/en/