NNT Change Tracker – Upgrade Notes

June 2020 ©New Net Technologies

Contents

Introduction

This document provides detailed information about the process involved in upgrading the NNT software to the latest edition and the process involved in backing up the Change Tracker Mongo Database as well as restoring it if a roll back is required. Please view the New Version Release notes to ensure your NNT Software is on the latest edition.

Change Tracker Backup & Recovery Process

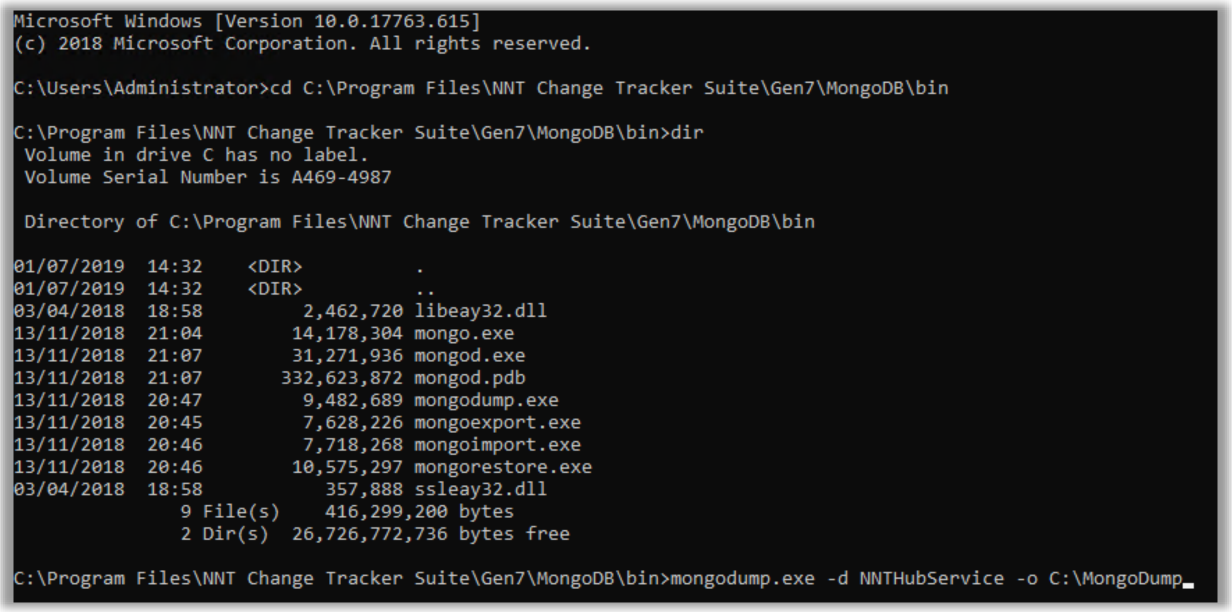
Backup Process

It is very important to take a backup of the Change Tracker Database before proceeding with the console upgrade as the need to roll back may arise in case things go wrong.

On the Change Tracker/MongoDB server, the Mongo Database installation contains a small application called MongoDump (usually located in C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin) that allows for a copy/backup of the database to be taken, this is best done with the Change Tracker hub (IIS Site) service shutdown. The MongoDump application is used via a simple command line which allows for the resulting backup file to be sent to either a local or remote destination, an example of the command is shown here:

```
mongodump.exe -d NNTHubService -o C:\MongoDump
```

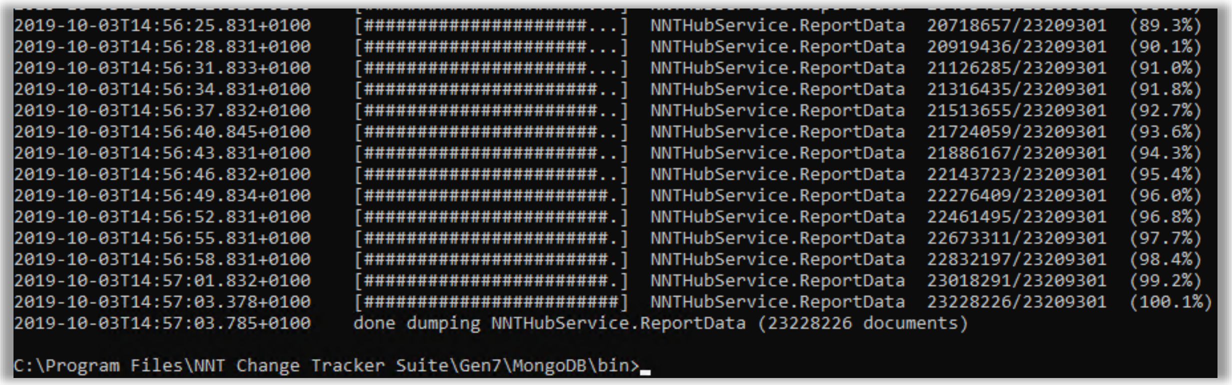
In the screenshot below you can see that an administrator command prompt has been loaded and the default MongoDB binary file location has been listed.



```
Microsoft Windows [Version 10.0.17763.615]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin

C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>dir
 Volume in drive C has no label.
 Volume Serial Number is A469-4987

 Directory of C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin

01/07/2019  14:32    <DIR>          .
01/07/2019  14:32    <DIR>          ..
03/04/2018  18:58         2,462,720 libeay32.dll
13/11/2018  21:04        14,178,304 mongo.exe
13/11/2018  21:07        31,271,936 mongod.exe
13/11/2018  21:07       332,623,872 mongod.pdb
13/11/2018  20:47         9,482,689 mongodump.exe
13/11/2018  20:45         7,628,226 mongoexport.exe
13/11/2018  20:46         7,718,268 mongoimport.exe
13/11/2018  20:46        10,575,297 mongorestore.exe
03/04/2018  18:58           357,888 ssleay32.dll
               9 File(s)    416,299,200 bytes
               2 Dir(s)  26,726,772,736 bytes free

C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongodump.exe -d NNTHubService -o C:\MongoDump
```

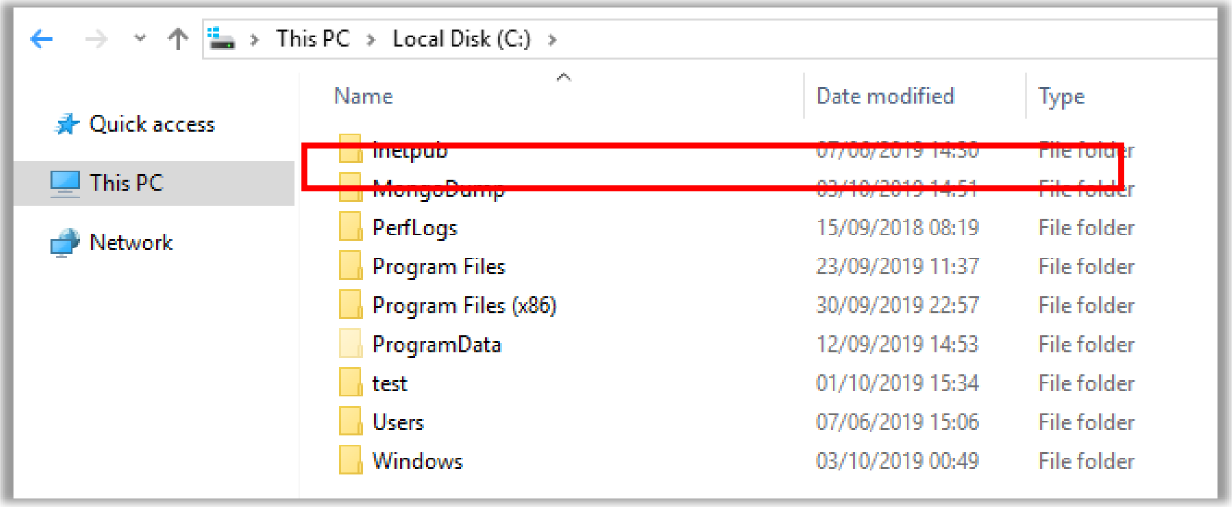
The time taken to complete the MongoDump/Backup all depends on the size of the Mongo Database, so please allow some time for completion.

Once the process has completed, you should see something like this on the command prompt:



```
2019-10-03T14:56:25.831+0100    [#####################...]  NNTHubService.ReportData  20718657/23209301  (89.3%)
2019-10-03T14:56:28.831+0100    [#####################...]  NNTHubService.ReportData  20919436/23209301  (90.1%)
2019-10-03T14:56:31.833+0100    [#####################...]  NNTHubService.ReportData  21126285/23209301  (91.0%)
2019-10-03T14:56:34.831+0100    [######################..]  NNTHubService.ReportData  21316435/23209301  (91.8%)
2019-10-03T14:56:37.832+0100    [######################..]  NNTHubService.ReportData  21513655/23209301  (92.7%)
2019-10-03T14:56:40.845+0100    [######################..]  NNTHubService.ReportData  21724059/23209301  (93.6%)
2019-10-03T14:56:43.831+0100    [######################..]  NNTHubService.ReportData  21886167/23209301  (94.3%)
2019-10-03T14:56:46.832+0100    [######################..]  NNTHubService.ReportData  22143723/23209301  (95.4%)
2019-10-03T14:56:49.834+0100    [#######################.]  NNTHubService.ReportData  22276409/23209301  (96.0%)
2019-10-03T14:56:52.831+0100    [#######################.]  NNTHubService.ReportData  22461495/23209301  (96.8%)
2019-10-03T14:56:55.831+0100    [#######################.]  NNTHubService.ReportData  22673311/23209301  (97.7%)
2019-10-03T14:56:58.831+0100    [#######################.]  NNTHubService.ReportData  22832197/23209301  (98.4%)
2019-10-03T14:57:01.832+0100    [#######################.]  NNTHubService.ReportData  23018291/23209301  (99.2%)
2019-10-03T14:57:03.378+0100    [########################]  NNTHubService.ReportData  23228226/23209301  (100.1%)
2019-10-03T14:57:03.785+0100    done dumping NNTHubService.ReportData (23228226 documents)

C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>_
```

Depending on where you have specified for the backup output to be placed in, you should now be able to locate the backup as seen in the screenshot below:



Name	Date modified	Type
inetpub	07/06/2019 14:30	File folder
MongoDump	03/10/2019 14:51	File folder
PerfLogs	15/09/2018 08:19	File folder
Program Files	23/09/2019 11:37	File folder
Program Files (x86)	30/09/2019 22:57	File folder
ProgramData	12/09/2019 14:53	File folder
test	01/10/2019 15:34	File folder
Users	07/06/2019 15:06	File folder
Windows	03/10/2019 00:49	File folder

Restore Process

Should the Change Tracker Server fail completely, the recovery approach would be:

- Stand-up a new Change Tracker server (virtual or physical).
- Installed the IIS roles and features required by Change Tracker.
- Install the same Change Tracker version as previously running (this will also install a blank Mongo database).
- Stop IIS and restore the Mongo database from backup.
- Rename machine to match the previous Change Tracker server.

The Mongo Database’s backup is restored to a running instance of the Mongo Database by using the MongoRestore application. The MongoRestore Application is found in the same location as the MongoDump executable (default location is C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin).

If the same Change Tracker version as previously is in place, before running the MongoRestore command, you will need to follow the steps below:

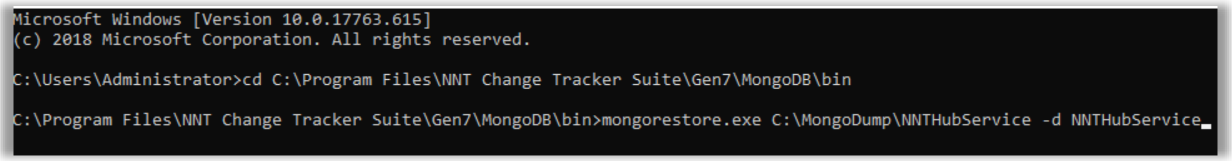
- Stop the **MongoDB** service.
- Navigate into **C:\ProgramData\Change Tracker Generation 7\MongoDB** (this is the default location).
- Rename the folder called **db** to **db-old** to keep a copy of the original db folder and create a new/empty folder called **db** OR delete the **db** folder and re-create it.
- Start the **MongoDB** service.

The steps above will clear the existing database and this will prepare it for a restore.

Once the steps above have been followed through with, the following command can be utilized to begin the restore process:

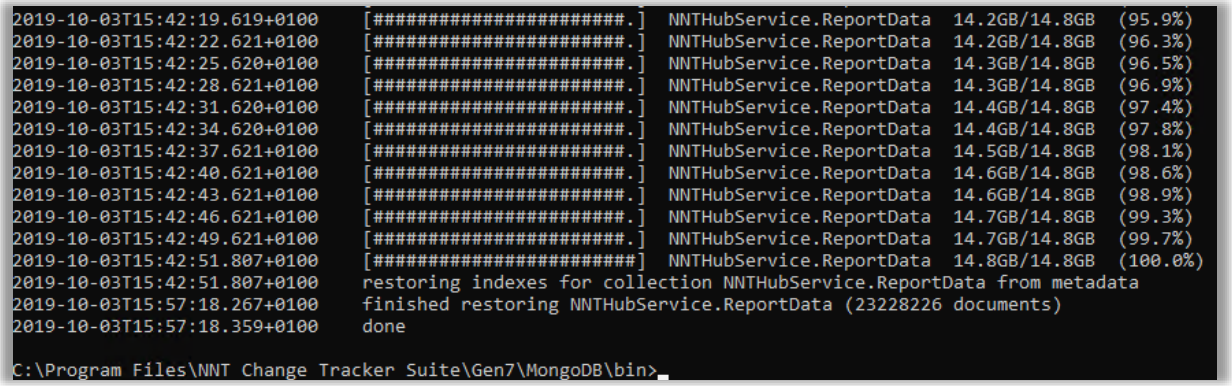
```
mongorestore.exe C:\MongoDump\NNTHubService -d NNTHubService
```

Please view an example of the command below:



```
Microsoft Windows [Version 10.0.17763.615]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin

C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongorestore.exe C:\MongoDump\NNTHubService -d NNTHubService
```

Once again, the time taken to complete the MongoRestore process all depends on the size of the Mongo Database, so please allow some time for completion. Once the restore has completed, you should see something similar to the screenshot below:



```
2019-10-03T15:42:19.619+0100    [#######################.]  NNTHubService.ReportData  14.2GB/14.8GB  (95.9%)
2019-10-03T15:42:22.621+0100    [#######################.]  NNTHubService.ReportData  14.2GB/14.8GB  (96.3%)
2019-10-03T15:42:25.620+0100    [#######################.]  NNTHubService.ReportData  14.3GB/14.8GB  (96.5%)
2019-10-03T15:42:28.621+0100    [#######################.]  NNTHubService.ReportData  14.3GB/14.8GB  (96.9%)
2019-10-03T15:42:31.620+0100    [#######################.]  NNTHubService.ReportData  14.4GB/14.8GB  (97.4%)
2019-10-03T15:42:34.620+0100    [#######################.]  NNTHubService.ReportData  14.4GB/14.8GB  (97.8%)
2019-10-03T15:42:37.621+0100    [#######################.]  NNTHubService.ReportData  14.5GB/14.8GB  (98.1%)
2019-10-03T15:42:40.621+0100    [#######################.]  NNTHubService.ReportData  14.6GB/14.8GB  (98.6%)
2019-10-03T15:42:43.621+0100    [#######################.]  NNTHubService.ReportData  14.6GB/14.8GB  (98.9%)
2019-10-03T15:42:46.621+0100    [#######################.]  NNTHubService.ReportData  14.7GB/14.8GB  (99.3%)
2019-10-03T15:42:49.621+0100    [#######################.]  NNTHubService.ReportData  14.7GB/14.8GB  (99.7%)
2019-10-03T15:42:51.807+0100    [########################]  NNTHubService.ReportData  14.8GB/14.8GB  (100.0%)
2019-10-03T15:42:51.807+0100    restoring indexes for collection NNTHubService.ReportData from metadata
2019-10-03T15:57:18.267+0100    finished restoring NNTHubService.ReportData (23228226 documents)
2019-10-03T15:57:18.359+0100    done

C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>
```

Upgrading the Change Tracker Console

This section will cover the steps required to upgrade your Change Tracer Console, please view the New Version Release notes to ensure your NNT Software is on the latest edition. Also, please be aware that a split architecture should not affect the upgrade procedure and moving forward .NET 4.7.2 Framework will be required from Change Tracker R2 version 7.2.1.19 onwards (on the Change Tracker Server only).

Upgrading Change Tracker Generation 7 to R2

If you still have the older Change Tracker Generation 7 installed and are interested in upgrading to R2, please follow the steps provided on this section or contact the Support Team at NNT to assist you.

- Please take a backup of MongoDB first.
- Ensure your Mongo Database is utilizing the wiredTiger storage engine.
- Ensure your Mongo Database is on the latest version that Change Tracker supports which is version 3.6.12, if you are on an older version of MongoDB, please follow the steps provided on the Upgrading the MongoDB Versions section.
- Ensure your Change Tracker Generation 7 (old) version is on the latest version which is 7.0.1.57. This should be as simple as running the installer and going through the pages presented, it should not ask for any configurations.
- Now, you can install the latest Change Tracker R2 version, please view the NNT Website for the latest R2 Version. This should be as simple as running the installer and going through the pages presented, it should not ask for any configurations.
- Finally, take a copy of the original C:\inetpub\wwwroot\Change Tracker Generation 7 Hub\bin\Configs\localhost.json file to hold a backup of it and replace all of the contents in that file with the contents contain on Appendix A – Localhost.json, save the file and restart the IIS Change Tracker Site. ***NOTE:*** *If have implemented a split architecture (Change Tracker & MongoDB on different servers), you will need to specify the MongoDB parameters in the file again, you should have a reference point from copied localhost.json file.*

Upgrading Change Tracker R2 From a Prior Version (7.2.1.17) to the Latest

If your current Change Tracker R2 version is on 7.2.1.17 or lower, you will need to follow the steps below to upgrade to the latest Change Tracker R2 edition:

- Please take a backup of MongoDB first.
- Stop IIS.
- Delete ALL report templates *.json files from C:\inetpub\wwwroot\Change Tracker Generation 7 Hub\bin\Setup\ReportTemplates.
- Install the latest Change Tracker R2 Version. This should be as simple as running the installer and going through the pages presented, it should not ask for any configurations.
- Run the Appendix B script against the Mongo Database to rename the templates in Mongo making it compatible with the newer Change Tracker R2 build. You will need to launch the Mongo Shell and to do that, you will need to launch a command prompt window, navigate into the MongoDB binary files directory (default location C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin) and then run the **Mongo** command.
- Start IIS.
- Now, on Change Trackers **Settings > Report Layout Templates** page, there will be various Updated templates available - Click 'Update Template' on each.

Upgrading Change Tracker R2 From Version 7.2.1.17 to Version 7.2.1.21

If your current Change Tracker R2 version is on 7.2.1.17 or higher, you will need to follow the steps below to upgrade to the latest Change Tracker R2 edition:

- Please take a backup of MongoDB first.
- Download the latest Change Tracker R2 Installer from the NNT website.
- Install the latest Change Tracker R2 Version. This should be as simple as running the installer and going through the pages presented, it should not ask for any configurations.

Upgrade Change Tracker R2 From Version 7.2.1.21 to Change Tracker Version 7.3

The Change Tracker version 7.3 is a major advancement in what is possible with Change Tracker's architecture. Using Change Tracker 7.3, customers can scale their Change Tracker deployment as their environment grows, adding more resources and high availability. View a short video on the NNT website which explains the benefits of the 7.3 architecture - https://www.newnettechnologies.com/mega-scale-change-control.html

If you are running a Change Tracker Hub with a split Mongo database, i.e. a database that is located on a different system to be the Change Tracker Hub, then please contact NNT Support for advice on upgrading to Change Tracker 7.3.

For technical support, including answers to questions not addressed in this product release notes, visit the NNT Helpdesk portal and review the knowledge base:

https://supportnntws.atlassian.net/servicedesk/customer/user/login?destination=portals

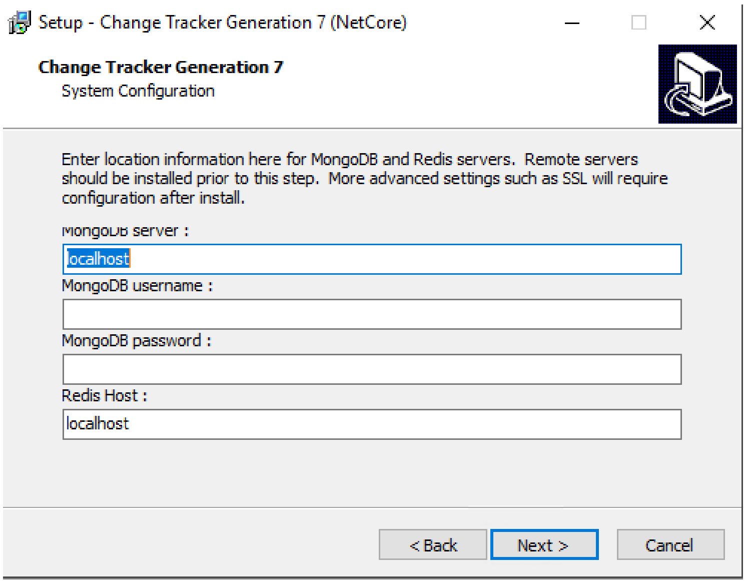
If you do not have access to the NNT Helpdesk portal please contact NNT Support directly on support@nntws.com

Change Tracker 7.3 pre-requisites

- Before upgrading to Change Tracker 7.3 ensure that you are on version 7.2.1.21 of Change Tracker. Upgrade details can be found here.
- Change Tracker now uses .NET Core and so this package must be downloaded and installed before commencing the upgrade to 7.3. .NET Core Package.
- Identify the current Change Tracker database path. The default location for the Change Tracker database which will be offered by the installer has changed and so this path must be altered to the current database location during the 7.3 install. The Mongo database default path is C:\ProgramData\Change Tracker Generation 7\MongoDB. For confirmation, check the mongod.conf file which contains the database and log location. The default location for the mongod.conf is C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\conf.

Change Tracker 7.3 Upgrade Process

- Confirm the .NET Core Package has been installed
- Please take a backup of MongoDB first.
- Download the latest Change Tracker R2 Installer from the NNT website.
- Install the latest Change Tracker 7.3 Version. Run through the NNT Change Tracker 7.3 installer. The following screenshots show the key points of the installer.



Change Tracker Generation 7
System Configuration

Enter location information here for MongoDB and Redis servers. Remote servers should be installed prior to this step. More advanced settings such as SSL will require configuration after install.

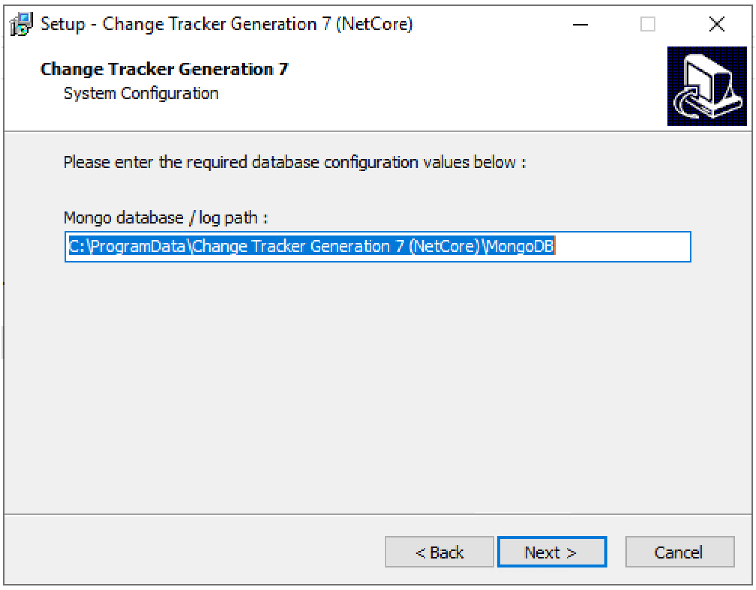
MongoDB server : localhost

MongoDB username :

MongoDB password :

Redis Host : localhost

Leave the MongoDB server and Redis Host as localhost.



Change Tracker Generation 7
System Configuration

Please enter the required database configuration values below :

Mongo database / log path : C:\ProgramData\Change Tracker Generation 7 (NetCore)\MongoDB

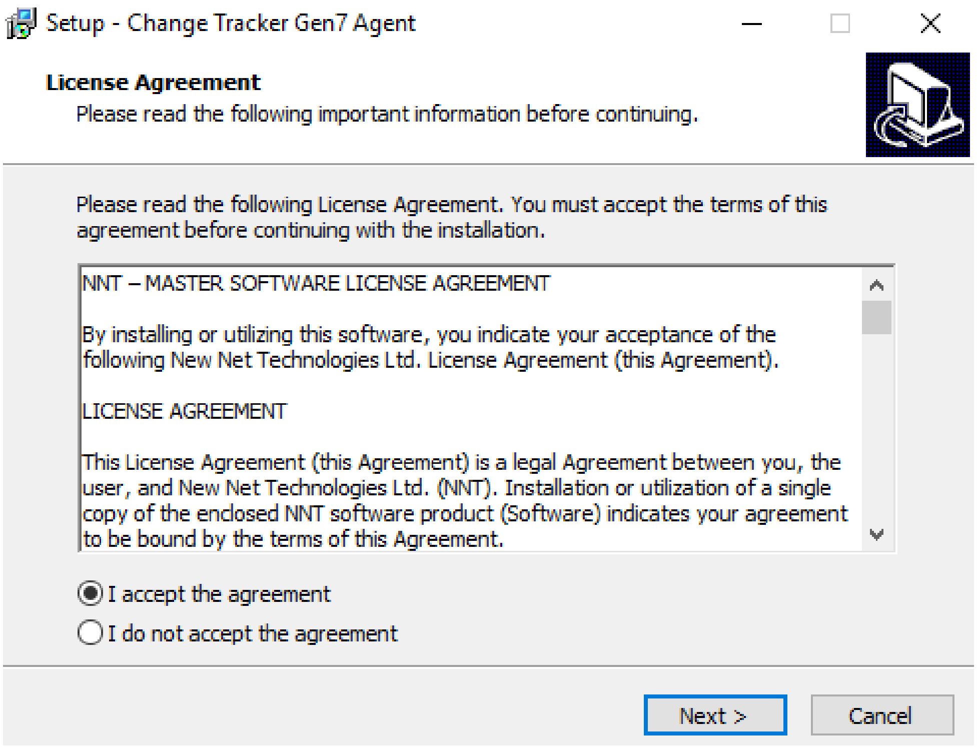
NOTE: The default location for the mongo db path of 7.3 is different to that of Change Tracker's R2 default database location. As listed in the pre-requisites, at this point in the 7.3 installation paste the current location for the Mongo database into this field.

Upgrading the Agents

This section will cover the steps involved in upgrading the Agents. Please view the New Version Release notes to ensure your NNT Software is on the latest edition.

Windows

Upgrading the Windows Agent version to a newer edition is fairly simple, please run the new version installer as an Administrator and go through the install pages, no configuration is required as this has been done before so the new agent version will be aware of this.



License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

NNT – MASTER SOFTWARE LICENSE AGREEMENT

By installing or utilizing this software, you indicate your acceptance of the following New Net Technologies Ltd. License Agreement (this Agreement).

LICENSE AGREEMENT

This License Agreement (this Agreement) is a legal Agreement between you, the user, and New Net Technologies Ltd. (NNT). Installation or utilization of a single copy of the enclosed NNT software product (Software) indicates your agreement to be bound by the terms of this Agreement.

● I accept the agreement

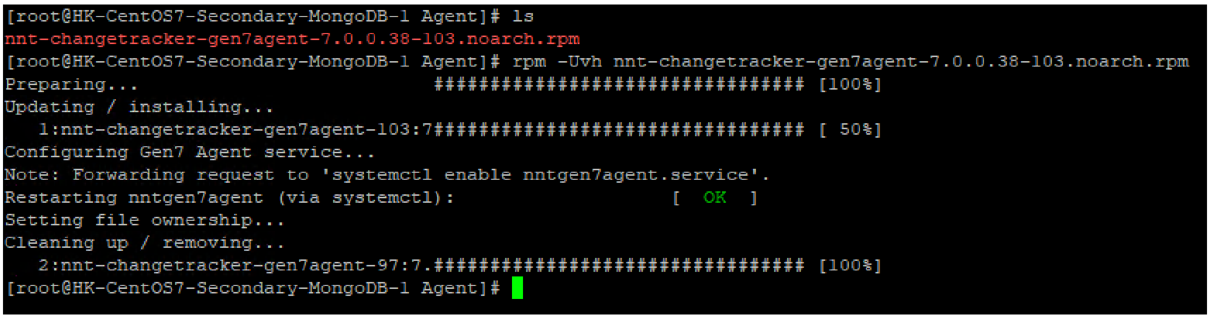
○ I do not accept the agreement

Linux

Upgrading the Linux Agent version to a newer edition is fairly simple, please run the following command to upgrade:

```
rpm -Uvh <Agent Package>
```

Please view the screenshot below to see the upgrade process:

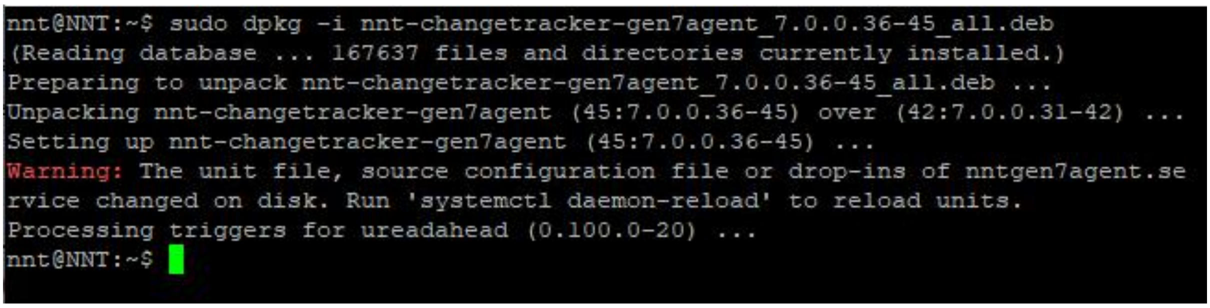


```
[root@HK-CentOS7-Secondary-MongoDB-1 Agent]# ls
nnt-changetracker-gen7agent-7.0.0.38-103.noarch.rpm
[root@HK-CentOS7-Secondary-MongoDB-1 Agent]# rpm -Uvh nnt-changetracker-gen7agent-7.0.0.38-103.noarch.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:nnt-changetracker-gen7agent-103:7################################# [ 50%]
Configuring Gen7 Agent service...
Note: Forwarding request to 'systemctl enable nntgen7agent.service'.
Restarting nntgen7agent (via systemctl):                   [  OK  ]
Setting file ownership...
Cleaning up / removing...
   2:nnt-changetracker-gen7agent-97:7.################################# [100%]
[root@HK-CentOS7-Secondary-MongoDB-1 Agent]#
```

Debian Linux

Upgrading the Debian Linux Agent version to a newer edition is fairly simple, please run the following command to upgrade:

```
dpkg -i <Agent Package>
```



```
nnt@NNT:~$ sudo dpkg -i nnt-changetracker-gen7agent_7.0.0.36-45_all.deb
(Reading database ... 167637 files and directories currently installed.)
Preparing to unpack nnt-changetracker-gen7agent_7.0.0.36-45_all.deb ...
Unpacking nnt-changetracker-gen7agent (45:7.0.0.36-45) over (42:7.0.0.31-42) ...
Setting up nnt-changetracker-gen7agent (45:7.0.0.36-45) ...
Warning: The unit file, source configuration file or drop-ins of nntgen7agent.se
rvice changed on disk. Run 'systemctl daemon-reload' to reload units.
Processing triggers for ureadahead (0.100.0-20) ...
nnt@NNT:~$
```

Upgrading the MongoDB Versions

This section will cover the steps required to upgrade your MongoDB Version to the latest supported edition. If your MongoDB version is fairly low, you will be required to upgrade in increments to get to the latest version. Please view the Package Locations section to get a link to the MongoDB downloads page and also take a backup of the database before proceeding.

Switching the MongoDB Storage Engine

Before upgrading MongoDB, you will be required to switch the storage engine utilized to wiredTiger as this engine is more efficient and allows for database compression. Please check your MongoDB configuration file to inspect what is currently configured. By default, the Change Tracker MongoDB instances configuration file is location in C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\conf\mongod.conf and the line you should be looking for is **storageEngine=wiredTiger**.

If that line exists already, then you do not need to do anything as your database is already using the recommended storage engine. However, if you find that it has not been configured to **wiredTiger**, you will need to perform a few steps to configure it. Please perform a MongoDB Backup and then once the backup has completed, please stop the MongoDB service and edit the MongoDB configuration file to include the following line: **storageEngine=wiredTiger**

Once you have successfully backed up the database and have added in the recommended storage engine to the configuration file, please navigate into the database directory, by default it is located in C:\ProgramData\Change Tracker Generation 7\MongoDB. Now, you have got two options to pick from, one option is to rename the **db** folder to something like **db-old** (this keeps a copy of the current database) and then create a new empty folder called **db** and the second option is to delete the **db** folder and re-create it to ensure the folder is empty. Please follow one option and once completed, please start the MongoDB service and then execute the MongoRestore.exe executable to restore the database backup point (further information can be found in the restore section – page 4-5), this will restore your database ensuring that it utilizes the wiredTiger storage engine.

Once you have confirmed/applied the MongoDB storage engine, please follow the relevant section steps below that apply to your environment.

If your current MongoDB Edition is on version **3.2.*** please start from this point

If your current MongoDB version is on 3.2.*, please ensure it is on the latest edition of 3.2 which is 3.2.22 (if your MongoDB is already on this version, please follow the steps on the next section):

- **mongodb-win32-x86_64-2008plus-ssl-3.2.22-signed.msi**

Once you have downloaded that version of MongoDB, please follow the subsection called The Steps Required. Once you have successfully upgraded the MongoDB instance to the latest 3.2 edition, please follow the steps provided on the next section to install the latest edition of 3.4.

If your current MongoDB Edition is on version **3.4.*** please start from this point

If your current MongoDB version is on 3.4.*, please ensure it is on the latest edition of 3.4 which is 3.4.23 (if your MongoDB is already on this version, please follow the steps on the next section):

- **mongodb-win32-x86_64-2008plus-ssl-3.4.23-signed.msi**

Once you have downloaded the version above, please follow the subsection called The Steps Required. When you have followed through with The Steps Required section, you will need to ensure that your Change Tracker MongoDB instance is compatible with version 3.4. Open up a command prompt window and navigate into the MongoDB binary files location, please view the command below (the path below is the default location):

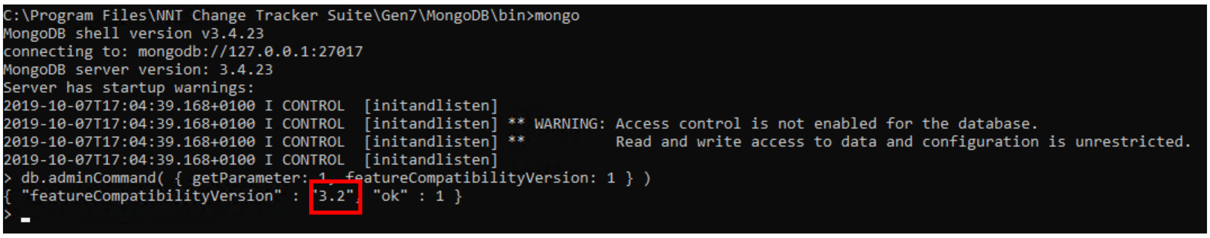
```
cd C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin
```

Once you have navigated into the MongoDB binary files location, please launch Mongo Shell using the command below:

```
mongo
```

Once Mongo Shell has launched, please execute the command below to confirm if the database is compatible with version 3.4:

```
db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
```

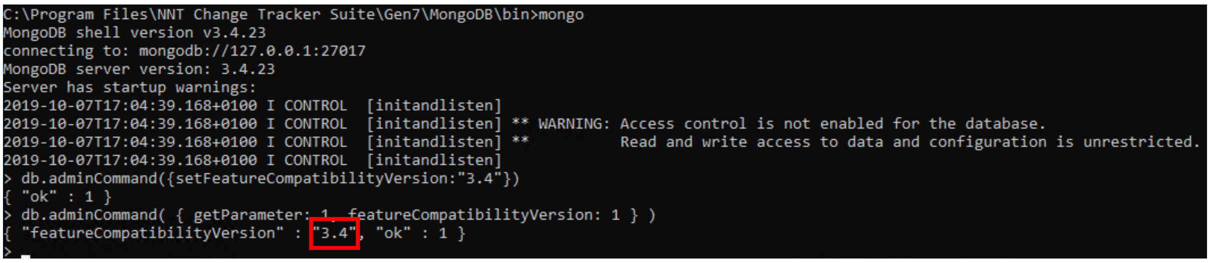


```
C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongo
MongoDB shell version v3.4.23
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.4.23
Server has startup warnings:
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten]
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten]
> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{ "featureCompatibilityVersion" : "3.2", "ok" : 1 }
>
```

In the screenshot above, you can see that the command has been executed and it seems like the database is not compatible with 3.4 as the command has returned “3.2”. In order to make it compatible, we will need to execute the command below on Mongo Shell.

```
db.adminCommand({setFeatureCompatibilityVersion:"3.4"})
```

Now, if you execute the command from before to confirm the compatible version, it should output “3.4” as seen in the screenshot below:



```
C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongo
MongoDB shell version v3.4.23
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.4.23
Server has startup warnings:
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten]
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2019-10-07T17:04:39.168+0100 I CONTROL  [initandlisten]
> db.adminCommand({setFeatureCompatibilityVersion:"3.4"})
{ "ok" : 1 }
> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{ "featureCompatibilityVersion" : "3.4", "ok" : 1 }
>
```

At this point, you can move onto the next section to upgrade to the latest 3.6 edition.

If your current MongoDB Edition is on version **3.6.*** please start from this point

If your current MongoDB version is on 3.6.*, please ensure it is on the latest edition of 3.6 that our software supports which is 3.6.12:

- **mongodb-win32-x86_64-2008plus-ssl-3.6.12-signed.msi**

Once you have downloaded the version above, please follow the subsection called The Steps Required. When you have followed through with The Steps Required section, you will need to ensure that your Change Tracker MongoDB instance is compatible with version 3.6. Open up a command prompt window and navigate into the MongoDB binary files location, please view the command below (the path below is the default location):

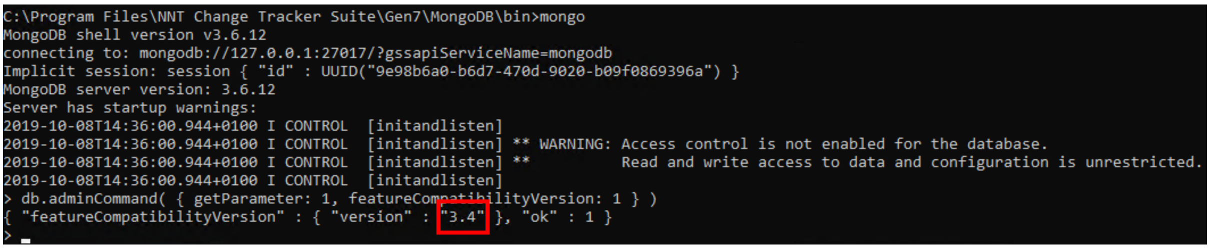
```
cd C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin
```

Once you have navigated into the MongoDB binary files location, please launch Mongo Shell using the command below:

```
mongo
```

Once Mongo Shell has launched, please execute the command below to confirm if the database is compatible with version 3.6:

```
db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
```

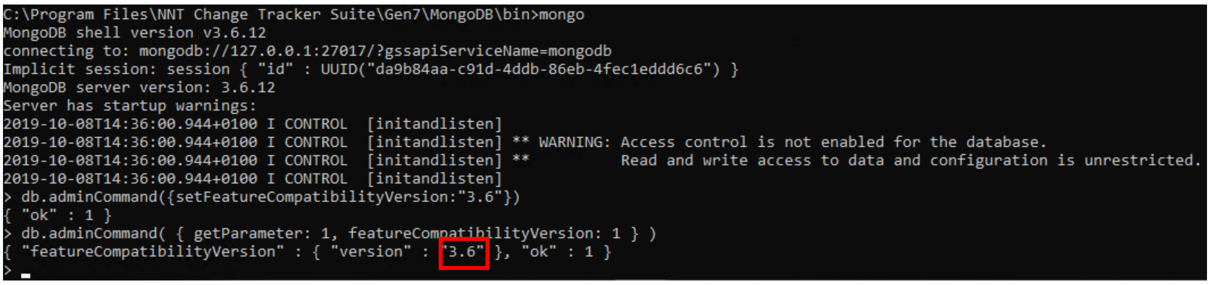


```
C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongo
MongoDB shell version v3.6.12
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("9e98b6a0-b6d7-470d-9020-b09f0869396a") }
MongoDB server version: 3.6.12
Server has startup warnings:
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten]
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten]
> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{ "featureCompatibilityVersion" : { "version" : "3.4" }, "ok" : 1 }
> _
```

In the screenshot above, you can see that the command has been executed and it seems like the database is not compatible with 3.6 as the command has returned "3.4". In order to make it compatible, we will need to execute the command below on Mongo Shell.

```
db.adminCommand({setFeatureCompatibilityVersion:"3.6"})
```

Now, if you execute the command from before to confirm the compatible version, it should output "3.6" as seen in the screenshot below:

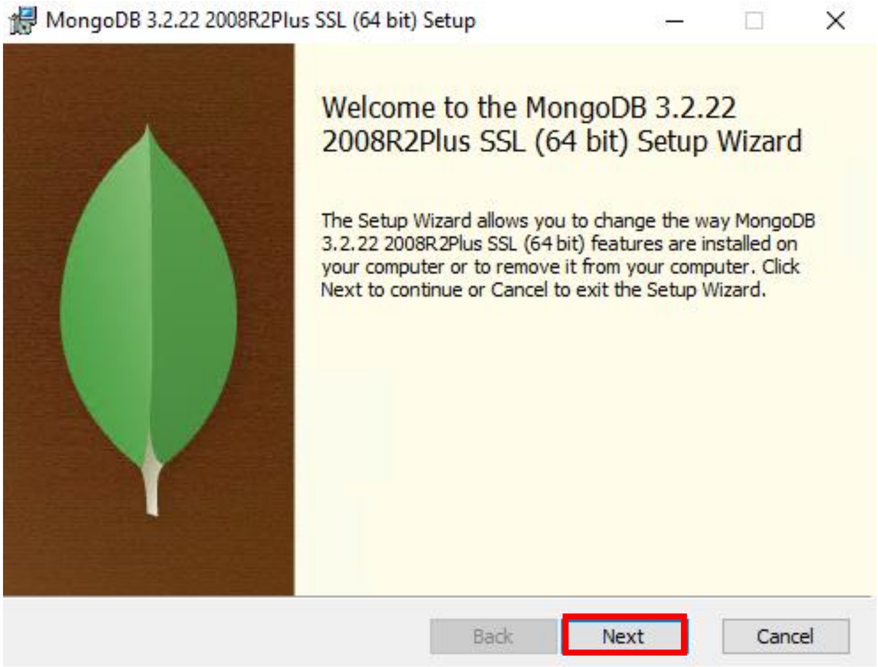


```
C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\bin>mongo
MongoDB shell version v3.6.12
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("da9b84aa-c91d-4ddb-86eb-4fec1eddd6c6") }
MongoDB server version: 3.6.12
Server has startup warnings:
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten]
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2019-10-08T14:36:00.944+0100 I CONTROL  [initandlisten]
> db.adminCommand({setFeatureCompatibilityVersion:"3.6"})
{ "ok" : 1 }
> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{ "featureCompatibilityVersion" : { "version" : "3.6" }, "ok" : 1 }
> _
```

The Steps Required

In this example, MongoDB version 3.2.11 will be upgraded to the latest 3.2 edition which is 3.2.22.

Please run the downloaded MongoDB executable and press on the next button to continue to the next step.

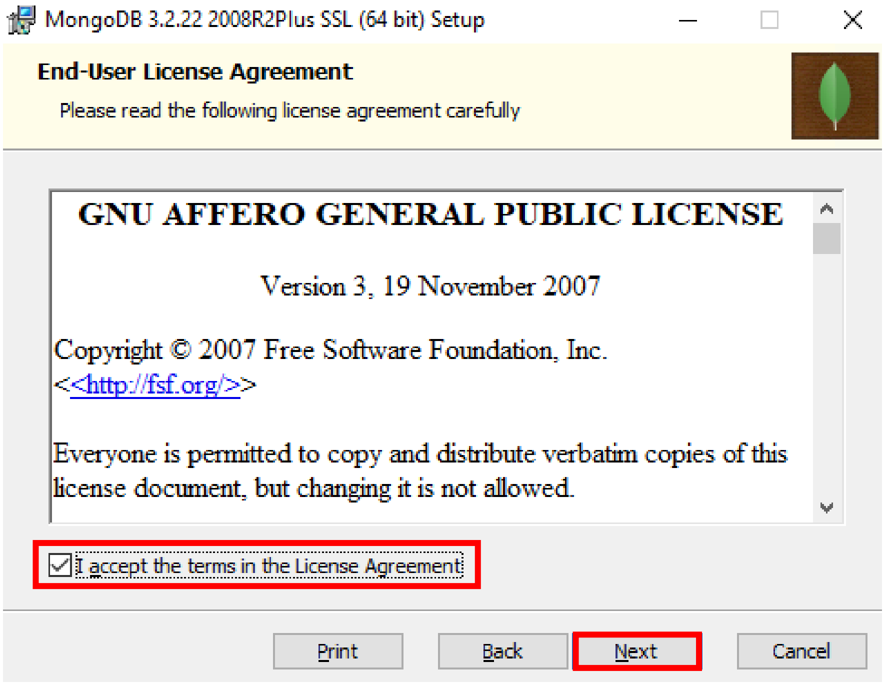


MongoDB 3.2.22 2008R2Plus SSL (64 bit) Setup

Welcome to the MongoDB 3.2.22 2008R2Plus SSL (64 bit) Setup Wizard

The Setup Wizard allows you to change the way MongoDB 3.2.22 2008R2Plus SSL (64 bit) features are installed on your computer or to remove it from your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Please review the end-user license agreement and then press on the next button.



MongoDB 3.2.22 2008R2Plus SSL (64 bit) Setup

End-User License Agreement

Please read the following license agreement carefully

GNU AFFERO GENERAL PUBLIC LICENSE

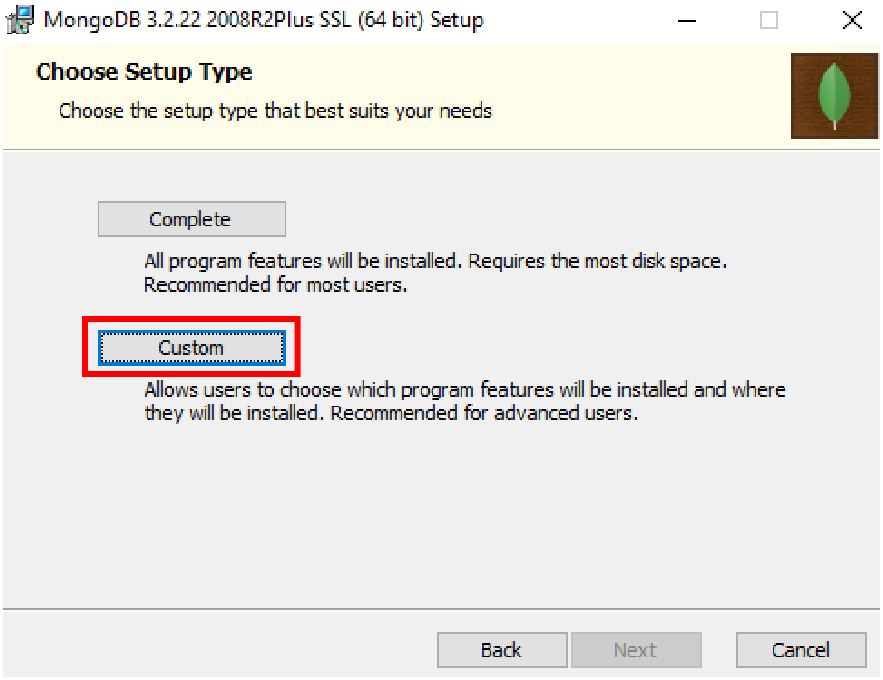
Version 3, 19 November 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

☑ I accept the terms in the License Agreement

MongoDB will by default install its components into the default directory so please ensure you select the **Custom** button to specify the Change Tracker Database path.



MongoDB 3.2.22 2008R2Plus SSL (64 bit) Setup

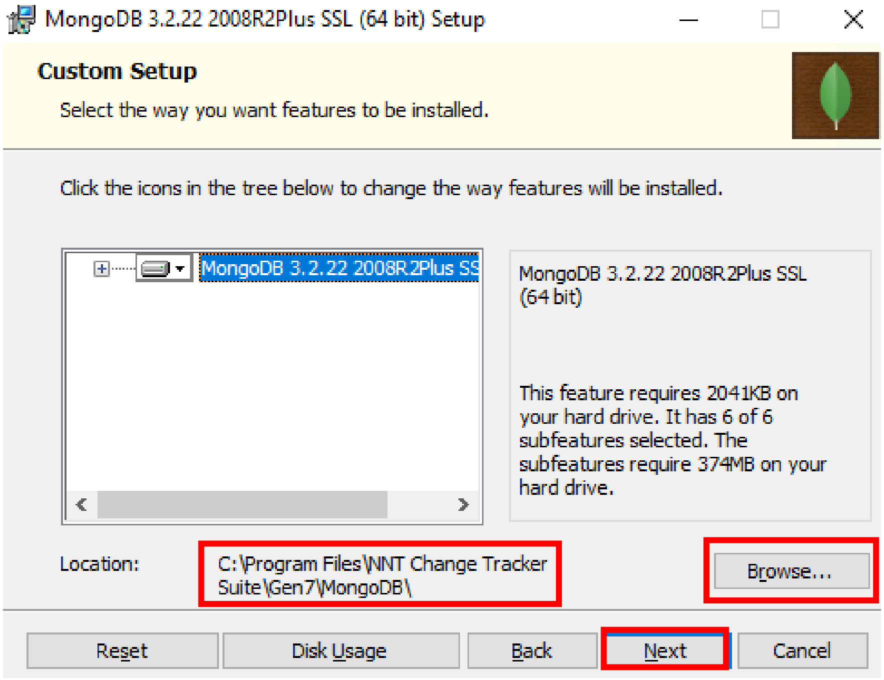
Choose Setup Type

Choose the setup type that best suits your needs

All program features will be installed. Requires the most disk space. Recommended for most users.

Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.

Once you have selected the custom option, please ensure you select the **Browse** button and input the following directory (this is the default Change Tracker MongoDB directory): **C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB**



MongoDB 3.2.22 2008R2Plus SSL (64 bit) Setup

Custom Setup

Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

MongoDB 3.2.22 2008R2Plus SSL (64 bit)

This feature requires 2041KB on your hard drive. It has 6 of 6 subfeatures selected. The subfeatures require 374MB on your hard drive.

Location: C:\Program Files\NNT Change Tracker Suite\Gen7\MongoDB\

Once the steps above have been followed through with, you can click the next button and finish up the installation.

Package Locations

If you have not already done so, please go to the Customer Portal area on our website and register for an account. The customer portal contains manuals, software downloads and a Knowledge Base with lots of useful information.

- https://www.newnettechnologies.com/index.php?option=com_comprofiler&view=registers&Itemid=573

Change Tracker Server

Once you have logged into the NNT website, please navigate to the following link to access the Change Tracker Servers downloads area:

- https://www.newnettechnologies.com/index.php?option=com_docman&view=list&layout=default&slug=all-variants-2008-r2-2012-r2-2016&own=0&Itemid=1159&lang=en

Alternatively, you can follow the path below on the website:
NNT ▶ Downloads ▶ Change Tracker™ Gen7 R2 ▶ Software Installers ▶ Server Installers ▶ Windows ▶ All Variants (2008 R2, 2012 R2 & 2016)

NNT Agents

Once you have logged into the NNT website, please navigate to the following link to access the Agent Installers area:

- https://www.newnettechnologies.com/change-tracker-gen-7-agent-installers.html

Alternatively, you can follow the path below on the website:
NNT ▶ Downloads ▶ Change Tracker™ Gen7 R2 ▶ Software Installers ▶ Agent Installers

MongoDB Versions

On the following website, you will be able to locate all of the MongoDB version binaries:
NOTE: *The latest MongoDB instance Change Tracker R2 supports is version 3.6.12*

- https://www.mongodb.org/dl/win32/

Dot Net Framework 4.7.2

From the release of Change Tracker version 7.2.1.19, there is a requirement to have a minimum Dot Net Framework of 4.7.2 installed on the Change Tracker management station. A download of this version of Dot Net can be found here:

- https://support.microsoft.com/en-gb/help/4054530/microsoft-net-framework-4-7-2-offline-installer-for-windows

NOTE: *Upgrading the .NET Framework may require a system restart.*

Dot Net Core 3.1 Runtime (v3.1.2) - Windows Hosting Bundle

All 7.3 versions of Change Tracker will require dot net core to be installed. The current Change Tracker compatible version of dot net core can be found here:

- https://dotnet.microsoft.com/download/dotnet-core/thank-you/runtime-aspnetcore-3.1.2-windows-hosting-bundle-installer

Customer Support

For technical support, please visit the NNT Helpdesk portal:
https://supportnntws.atlassian.net/servicedesk/customer/user/login?destination=portals

If you do not have access to the NNT Helpdesk portal, please contact NNT Support directly on support@nntws.com

Appendix A – Localhost.json

```
{
 "incomingEventQueue": {
  "queueType": "MongoQueue",
  "connectionString": "mongodb://localhost",
  "queueName": "IncomingEventQueue",
  "dbName": "NNTHubService",
  "sortField": "DateUtc.Ticks",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "agentTaskQueue": {
  "queueType": "MongoQueue",
  "connectionString": "mongodb://localhost",
  "queueName": "AgentTaskQueue",
  "dbName": "NNTHubService",
  "sortField": "AgentTask._id",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "backgroundTaskQueue": {
  "queueType": "MongoQueue",
  "connectionString": "mongodb://localhost",
  "queueName": "BackgroundTaskQueue",
  "dbName": "NNTHubService",
  "sortField": "SubmittedDateTimeUtc",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "_exclusionField": "_t",
  "_exclusionItems": "GenerateReportDataBackgroundTask",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "notificationQueue": {
  "queueType": "MongoQueue",
  "connectionString": "mongodb://localhost",
  "queueName": "NotificationQueue",
  "dbName": "NNTHubService",
  "sortField": "DateUtc",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "failoverEventQueue": {
  "queueType": "DotNetQueue",
  "queueName": "FailoverEventQueue"
 },
 "sequenceGenerator": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "eventRepository": {
  "repositoryType": "MongoEventRepository",
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "hashRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "policyRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "cacheService": {
  "serviceType": "SimpleCacheService"
 },
 "authRepository": {
  "repositoryType": "MongoAuthRepository",
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "directoryRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "credentialsRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "commandRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "fileRepository": {
  "repositoryType": "MongoFileRepository",
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "backupRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "backgroundTaskRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "savedUserDataRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "configRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
  "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
  "verifySslCertificate": "no"
 },
 "organizationRepository": {
  "connectionString": "mongodb://localhost",
  "dbName": "NNTHubService",
  "userName": "T_USERNAME",
  "password": "T_PASSWORD",
  "useSSL": "no",
     "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
     "verifySslCertificate": "no"
   },
   "performanceDataRepository": {
     "connectionString": "mongodb://localhost",
     "dbName": "NNTHubService",
     "userName": "T_USERNAME",
     "password": "T_PASSWORD",
     "useSSL": "no",
     "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
     "verifySslCertificate": "no"
   },
   "pendingNotificationRepository": {
     "connectionString": "mongodb://localhost",
     "dbName": "NNTHubService",
     "userName": "T_USERNAME",
     "password": "T_PASSWORD",
     "useSSL": "no",
     "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
     "verifySslCertificate": "no"
   },
   "ipAddressActivityRepository": {
     "connectionString": "mongodb://localhost",
     "dbName": "NNTHubService",
     "userName": "T_USERNAME",
     "password": "T_PASSWORD",
     "useSSL": "no",
     "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
     "verifySslCertificate": "no"
   },
   "reportDataRepository": {
     "connectionString": "mongodb://localhost",
     "dbName": "NNTHubService",
     "userName": "T_USERNAME",
     "password": "T_PASSWORD",
     "useSSL": "no",
     "certificatePath": "C:\\nnt-example-path\\mongodb-cert.crt",
     "verifySslCertificate": "no"
   }
 }
```

Appendix B - Template Upgrade Script

```
// Report layout template names have been rationalised in 7.2.1.17
// Run these queries to update existing records before allowing the
7.2.1.17 IIS application to startup for the first time
//
========================================================================
=============================================

use NNTHubService

// Compliance Full -> Compliance Detail

// Find the number of Compliance Full Templates being used.
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Compliance Full'}).count()

// Update the Template Name
db.ReportData.update({'ScheduledReportItem.EmailDelivery.Attachment1Tem
plateName': 'Compliance Full'}, { $set:
{'ScheduledReportItem.EmailDelivery.Attachment1TemplateName':
'Compliance Detail'} }, { multi: true })

// Find the number of Compliance Detail Templates being used
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Compliance Detail'}).count()


// Planned Change Instance -> Planned Change Detail

// Find the number of Compliance Full Templates being used.
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Planned Change Instance'}).count()

// Update the Template Name
db.ReportData.update({'ScheduledReportItem.EmailDelivery.Attachment1Tem
plateName': 'Planned Change Instance'}, { $set:
{'ScheduledReportItem.EmailDelivery.Attachment1TemplateName': 'Planned
Change Detail'} }, { multi: true })

// Find the number of Compliance Detail Templates being used
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Planned Change Detail'}).count()


// Device Monitoring -> Device Detail

// Find the number of Compliance Full Templates being used.
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Device Monitoring'}).count()

// Update the Template Name
db.ReportData.update({'ScheduledReportItem.EmailDelivery.Attachment1Tem
plateName': 'Device Monitoring'}, { $set:
{'ScheduledReportItem.EmailDelivery.Attachment1TemplateName': 'Device
Detail'} }, { multi: true })
```

```
// Find the number of Compliance Detail Templates being used
db.ReportData.find({'ScheduledReportItem.EmailDelivery.Attachment1Templ
ateName': 'Device Detail'}).count()


// Updating the fs.files Report Template meta data
// ================================================

// Compliance Full -> Compliance Detail

// Find the number of Compliance Full Report Templates, filename and
metadata counts should match
db.fs.files.find( { 'filename': 'ComplianceFull.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Compliance Full'
}).count()

// Update the filename
db.fs.files.update({'filename': 'ComplianceFull.json'}, { $set:
{'filename': 'ComplianceDetail.json'} }, { multi: true })
// Update the metadata
db.fs.files.update({'metadata.ReportTemplateName': 'Compliance Full'},
{ $set: {'metadata.ReportTemplateName': 'Compliance Detail'} }, {
multi: true })
db.fs.files.update({'metadata.FileName': 'ComplianceFull.json'}, {
$set: {'metadata.FileName': 'ComplianceDetail.json'} }, { multi: true
})

// Find the number of Compliance Detail Report Templates, filename and
metadata counts should match
db.fs.files.find( { 'filename': 'ComplianceDetail.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Compliance Detail'
}).count()


// Planned Change Instance -> Planned Change Detail

// Find the number of Planned Change Instance Report Templates,
filename and metadata counts should match
db.fs.files.find( { 'filename': 'PlannedChangeInstance.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Planned Change
Instance' }).count()

// Update the filename
db.fs.files.update({'filename': 'PlannedChangeInstance.json'}, { $set:
{'filename': 'PlannedChangeDetail.json'} }, { multi: true })
// Update the metadata
db.fs.files.update({'metadata.ReportTemplateName': 'Planned Change
Instance'}, { $set: {'metadata.ReportTemplateName': 'Planned Change
Detail'} }, { multi: true })
db.fs.files.update({'metadata.FileName': 'PlannedChangeInstance.json'},
{ $set: {'metadata.FileName': 'PlannedChangeDetail.json'} }, { multi:
true })

// Find the number of Planned Change Detail Report Templates, filename
and metadata counts should match
```

```
db.fs.files.find( { 'filename': 'PlannedChangeDetail.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Planned Change
Detail' }).count()


// Device Monitoring -> Device Detail

// Find the number of Device Monitoring Report Templates, filename and
metadata counts should match
db.fs.files.find( { 'filename': 'DeviceMonitoring.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Device Monitoring'
}).count()

// Update the filename
db.fs.files.update({'filename': 'DeviceMonitoring.json'}, { $set:
{'filename': 'DeviceDetail.json'} }, { multi: true })
// Update the metadata
db.fs.files.update({'metadata.ReportTemplateName': 'Device
Monitoring'}, { $set: {'metadata.ReportTemplateName': 'Device Detail'}
}, { multi: true })
db.fs.files.update({'metadata.FileName': 'DeviceMonitoring.json'}, {
$set: {'metadata.FileName': 'DeviceDetail.json'} }, { multi: true })

// Find the number of Device Detail Report Templates, filename and
metadata counts should match
db.fs.files.find( { 'filename': 'DeviceDetail.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Device Detail'
}).count()

// Sti Error Template -> Error Template

// Find the number of Templates, filename and metadata counts should
match
db.fs.files.find( { 'filename': 'StiErrorTemplate.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Sti Error Template'
}).count()

// Update the filename
db.fs.files.update({'filename': 'StiErrorTemplate.json'}, { $set:
{'filename': 'ErrorTemplate.json'} }, { multi: true })
// Update the metadata
db.fs.files.update({'metadata.ReportTemplateName': 'Sti Error
Template'}, { $set: {'metadata.ReportTemplateName': 'Error Template'}
}, { multi: true })
db.fs.files.update({'metadata.FileName': 'StiErrorTemplate.json'}, {
$set: {'metadata.FileName': 'ErrorTemplate.json'} }, { multi: true })

// Find the number of Templates, filename and metadata counts should
match
db.fs.files.find( { 'filename': 'ErrorTemplate.json' }).count()
db.fs.files.find( { 'metadata.ReportTemplateName': 'Error Template'
}).count()
```