

# MCM-Service Installation

## Integration with RAIssoft ePDok – Manual

Release Date: 21.11.2019

INS-000061.en – 1.0.1



## Content

<b>1. Scope</b> .....	<b>2</b>
<b>2. Terms</b> .....	<b>2</b>
<b>3. Preparation</b> .....	<b>2</b>
3.1. Update Mobility & Care Manager.....	2
3.2. DB Backup.....	3
3.3. License for MCM-Service .....	3
<b>4. Installation procedure</b> .....	<b>3</b>
4.1. Run preparation script.....	3
4.2. Install MCM-Service .....	3
4.3. Open Firewall.....	3
4.4. Test service installation .....	3
<b>5. Integration with ePDok</b> .....	<b>3</b>
5.1. Admin settings in ePDok.....	3
5.2. Check Patient list in in MCM.....	4
5.3. Merge patients in database .....	4
5.4. Cleanup.....	5

## 1. Scope

This document describes the installation procedure for the software interface “MCM-Service”. MCM-Service is a REST-API for the Mobility & Care Manager database. This enables the integration into 3<sup>rd</sup> party applications such as RAIssoft ePDok.

Installation of the Mobility & Care Manager application is not covered by this document (see M-000011).

## 2. Terms

Term	Meaning
<b>MCM</b>	Mobility & Care Manager
<b>DB</b>	Database
<b>MCM-Service</b>	Product name of the REST-service
<b>epDok</b>	Nursing documentation software from RAIssoft

## 3. Preparation

### 3.1. Update Mobility & Care Manager

Make sure that the latest revision of MCM is installed.

## 3.2. DB Backup

Take a backup of the database. By default the DB is named “MobilityMonitoring”.

## 3.3. License for MCM-Service

Apply for an online license for the MCM-Service.

# 4. Installation procedure

## 4.1. Run preparation script

When updating from a MCM version < 1.2.6.0, then there are no Patient numbers. This provides a way to mark all existing patients with a dummy “number” string (e.g. ‘MCM’).

```
update [dbo].[Patient]
set [PatientNumber] = 'MCM', [Lastname] = [Lastname]+'_'
where [PatientNumber] = null or [PatientNumber] = '';

select p.Lastname, p.Firstname, p.PatientNumber from [dbo].[Patient] p;
```

[Script\\_Merge-Patients\\_01Preparation.sql](#)

## 4.2. Install MCM-Service

Run the setup wizard for the MCM-Service on the application server. If needed, also install the .Net Core runtime environment.

## 4.3. Open Firewall

Open ports 5000 – 5001 for incoming TCP traffic on the server.

## 4.4. Test service installation

In a web browser, enter the URL of the newly installed MCM-Service.

Example: <http://localhost:5000/swagger>

# 5. Integration with ePDok

## 5.1. Admin settings in ePDok

Enter the MCM-Service endpoint in the ePDok settings.

When successful, then the wards (departments) and patients from ePDok are synchronized to Mobility & Care Manager.

## 5.2. Check Patient list in in MCM

Check if for each patient in the Mobility & Care Manager database, there are two entries. One with the dummy patient number 'MCM' from the preparation script in step 4.1.

## 5.3. Merge patients in database

Run the following script:

```

/* Merge Patients from MCM with Patients synced by MCM-Service -- STEP 2 */

/* Move all data assigned to patient
   from: Patient with Lastname ending with '_'
   to: Patient with same Lastname and Firstname and Patientnumber is not 'MCM'
*/

-- Move Data
update [dbo].[MonitoringFile]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [MonitoringFile].PatientId =
pMCM.PatientId;

-- Move Comments
update [dbo].[Comment]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [Comment].PatientId = pMCM.PatientId;

-- Move [dbo].[CustomMonitoringEvent]
update [dbo].[CustomMonitoringEvent]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [CustomMonitoringEvent].PatientId =
pMCM.PatientId;

-- Move [dbo].[Event]
update [dbo].[Event]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [Event].PatientId = pMCM.PatientId;

-- Move [dbo].[MobilityToleranceSettings]
update [dbo].[MobilityToleranceSettings]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [MobilityToleranceSettings].PatientId =
pMCM.PatientId;

-- Move [dbo].[Observation]

```

```
update [dbo].[Observation]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [Observation].PatientId = pMCM.PatientId;

-- Move [dbo].[Observation]
update [dbo].[Observation]
set [PatientId] = pRai.[PatientId]
from Patient pRai
join Patient pMCM on (substring(pMCM.Lastname, 0, len(pMCM.Lastname)) = pRai.Lastname
and pMCM.Firstname = pRai.Firstname)
where pRai.PatientNumber not like 'MCM' and [Observation].PatientId = pMCM.PatientId;
```

[Script\\_Merge-Patients\\_02.sql](#)

## 5.4. Cleanup

Run the following script to cleanup the database.

```
/* Merge Patients from MCM with Patients synced by MCM-Service -- STEP 3 */

/* cleanup after Merging Patients */
/* delete Patients where Lastname ends with '_' and PatientNumber = 'MCM' */

-- select Patients without data (and PatientNumber is 'MCM'):
select p.PatientNumber, p.Lastname, p.Firstname, f.* from Patient p
left join dbo.MonitoringFile f on f.PatientId = p.PatientId
where MonitoringFileId is null and p.PatientNumber = 'MCM'
order by p.Lastname asc;

-- now delete!
delete p
from Patient p
left join dbo.MonitoringFile f on f.PatientId = p.PatientId
where MonitoringFileId is null
and Lastname like '%_' and PatientNumber = 'MCM';

-- remove PatientNumber "MCM":
update Patient
set PatientNumber = null
where PatientNumber = 'MCM';
```

[Script\\_Merge-Patients\\_02.sql](#)