

# Axiell Move 3.6.28 for the iPhone

Server version 3.0.20309.1



## Axiell ALM Netherlands B.V.

---

Copyright © 2013-2021 Axiell ALM Netherlands B.V.® All rights reserved. Adlib® is a product of Axiell ALM Netherlands B.V.®

The information in this document is subject to change without notice and should not be construed as a commitment by Axiell ALM Netherlands B.V. Axiell assumes no responsibility for any errors that may appear in this document. The software described in this document is furnished under a licence and may be used or copied only in accordance with the terms of such a licence. While making every effort to ensure the accuracy of this document, products are continually being improved.

As a result of continuous improvements, later versions of the products may vary from those described here. Under no circumstances may this document be regarded as a part of any contractual obligation to supply software, or as a definitive product description.

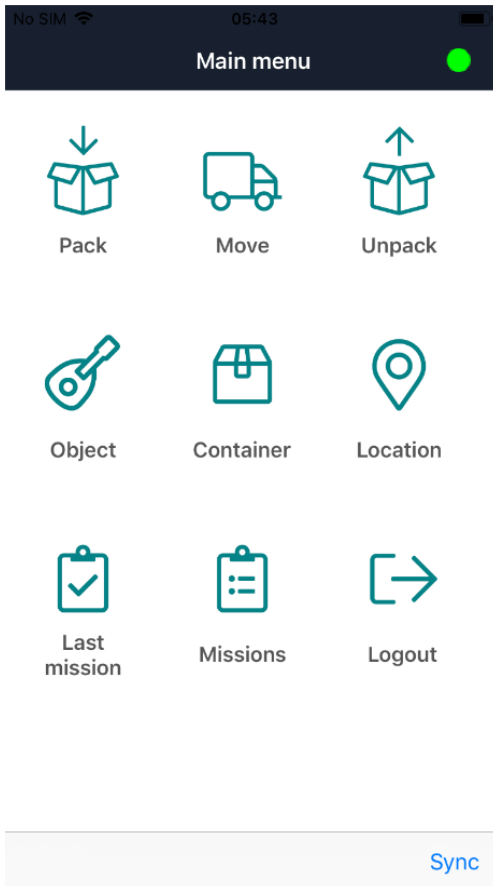
---

# Contents

|  |           |
|--|-----------|
| <b>1. Introduction .....</b>                             | <b>1</b>  |
| 1.1. Capabilities of Axiell Move .....                   | 3         |
| 1.2. Using the iPhone/scanner combination .....          | 3         |
| 1.3. New in 3.6.28 .....                                 | 5         |
| <b>2. Working with Axiell Move .....</b>                 | <b>7</b>  |
| 2.1. Starting Axiell Move.....                           | 9         |
| 2.2. The Main menu.....                                  | 11        |
| 2.3. Retrieving object or container information .....    | 13        |
| 2.4. Packing.....  | 16        |
| 2.5. Moving.....   | 23        |
| 2.6. Unpacking .....                                     | 28        |
| 2.7. Missions .....                                      | 34        |
| 2.8. Synchronizing offline transactions.....             | 45        |
| <b>3. Installation .....</b>                             | <b>49</b> |
| 3.1. Installing the Axiell Service (on the server) ..... | 49        |
| 3.1.1. AdmoveServerSetup.xml.....                        | 53        |
| 3.2. Registering login details.....                      | 78        |
| 3.3. Setting up the missions functionality .....         | 81        |
| 3.4. Setting up the iPhone .....                         | 85        |
| 3.5. Initializing Axiell Move .....                      | 87        |
| 3.6. Installing an update of the software.....           | 92        |
| 3.7. Security considerations.....                        | 92        |
| 3.8. Troubleshooting.....                                | 93        |



# 1. Introduction



The Axiell Move suite consists of an app for iOS 9 or later, called Axiell Move (aka the AdMove Client or previously known as the Movement module), to be installed on an Apple iPhone (from versions 5 and 5s) or iPod touch 5 or iPad, possibly fitted with a Linea Pro 5 or later (or Infinea Tab M for iPad mini and iPad Air) scanner sleeve (offered by [Infinite Peripherals\\*](#)) to scan barcodes or RFID tags (although the iPhone camera can handle scanning barcodes very well too if no separate scanning device is connected), plus the Axiell Move server software. The Axiell Move app can be downloaded from the Apple App Store.

In combination with Axiell (Adlib) Museum, Emu or Mimsy software, Axiell Move supports the logistic process of movements of (packed or bare) objects in museums or archives.

Axiell Move Premium has added functionality for so-called missions: missions are bundles of move or pack tasks assignable to specific co-workers, enabling you to plan and distribute the work to be done with Axiell Move.

The iPhone/scanner combination provides a fast barcode and RFID scanner with which identification numbers of objects, containers (packages) and locations can be scanned easily and quickly. The scanned codes will then be processed by Axiell Move. (An RFID tag - Radio Frequency Identification - is a small programmable chip with antenna, integrated in a label.)

If no scanner sleeve has been fitted around the Apple device, the camera of the device will automatically be used to scan barcodes.

The iPhone, iPod or iPad has a built-in Wi-Fi transmitter and receiver to enable a wireless connection to a local Wi-Fi hotspot which in turn is connected to a computer and your network: optionally, 3G/4G connectivity will be tried when the Wi-Fi connection fails (if the Apple device supports 3G/4G). Via the network, Axiell Move will then call an Axiell Move server on the database server to obtain direct access to a live database in which the on-site scanned and registered changes will then be saved automatically.

It is also possible to work offline with Axiell Move: this means that no continuous connection to a live database is required, and that changes will only be saved in the database when the Apple device is connected to the network and you've given your permission to synchronize the changes.

\* Since the Linea Pro scanner sleeve is not made by Axiell, its availability for specific Apple devices and versions depends on supplier Infinite Peripherals. Should a new scanner sleeve for an old Apple device no longer be available in the future, then you can always fall back on using the camera of the device as the barcode scanner.

---

## 1.1. Capabilities of Axiell Move

In brief, you have the following possibilities using the iPhone\* and Axiell Move:

- registration of the packaging of one or more objects. Overpacks containing multiple packaged objects, can be registered as well. You actually establish a relation between an object and its packaging.
- registration of movements of containers or bare objects. By scanning a new location or entering it manually, and subsequently scanning the containers or objects moving to this location, the registered location of all those items will be changed.
- registration of the unpacking of one or more objects or smaller containers. You can do this per item or for all items in a package at once. You actually break the relation between the items and their packaging.
- addition of supplemental information. During the registering of transactions you may enter several types of notes. It's also possible to take one or more pictures of the moved object, container or location, after which reproduction records for those photos will automatically be created, which in turn will be linked to the relevant object, container or location record.
- Axiell Move premium also offers the possibility to process packaging and movement work bundled in so-called missions used to organise and distribute such tasks among coworkers.

All scanned or entered information, like container codes, location changes and notes, will immediately be sent by Axiell Move (when you are working online) and processed in the relevant database by the Axiell Move server.

\* Unless there's a difference between the operation of the software on iPhone, iPod or iPad, we'll use "iPhone" whenever we mean "iPhone, iPod or iPad" in the rest of this manual, for the sake of succinctness.

---

## 1.2. Using the iPhone/scanner combination

- For data entry of barcode fields, you can use just the iPhone camera or the iPhone/scanner combination. In the latter case, keep the large button at the top of the right side of the scanner sleeve pressed down and aim the scanner on what you want to scan. Do not hold the scanner too close. When you hear a beep, you can let

go of the button. The scanned value will appear in the active *Barcode* field.

The scanner can only work if currently no charging cable is connected to the iPhone/scanner combination and if the Axiell Move app is active.

If you are not using a scanner sleeve, you only need to click (press) the grey *SCAN* button next to a *Barcode* field in the Axiell Move screen briefly to activate the camera. The image of the camera will appear on screen. Now reposition the camera in front of the barcode and allow the camera to focus on it. As soon as Axiell Move recognizes the barcode, the camera image disappears, you'll hear a beep and the record data identified by the barcode will be retrieved from the database, if possible.

- When scanning isn't possible, for instance because the barcode on a packaging has been damaged or when you want to fill in a text field, you can enter text via the virtual keyboard of the iPhone. Make the desired field active by clicking (pressing) it and then start typing. Finish your input by clicking *Return* or *Search* on the virtual keyboard.
- The Axiell Move app checks every 5 seconds if there is still a connection with the Axiell Move server. First a connection through Wi-Fi will be tried, and if that fails, a connection through 3G/4G\* will be attempted. If successful, the client will try to send all subsequent transactions to the server directly. If unsuccessful, your transactions will not be processed in the database immediately, instead they will be registered in a log file which can be synchronized with the database later on when you're back online.

\* By default, the iPhone uses a 3G or 4G dial connection to send data over the internet when no Wi-Fi network is available. This may cause unintended expenses though. Therefore it's good to know that this functionality can be switched off at any time: start the *Settings* app on the iPhone and select the *Mobile data* or *Cellular* option.

Now switch *Mobile data* or *Cellular data* off by dragging the slider to the left. (Green means: switched on.)

- If the Axiell Move server has been set up securely, so that it is not accessible outside the company network, then any external Wi-Fi network connection or 3G/4G dial connection will fail, unless you've set up and activated a VPN connection on your iPhone. Only an active VPN connection will then allow Axiell Move to access its server on the company network. Once a VPN configuration has been made (see chapter 3.3) you can switch it on or off via the



VPN slider in the *Settings* app.

Note that with VPN activated, normal internet access on your iPhone may be inhibited. Switching VPN off should solve the problem.

- The scanner sleeve contains a battery which must remain charged and the same goes for the iPhone of course. The best way to keep both devices charged is to place the combined device in the supplied charging cradle at the end of your working day.

---

### 1.3. New in 3.6.28

- You can now also set the home/normal location of an object or a container (if configured on the server side).
- Axiell Move Premium: if you're online, you can now release an already loaded mission from your device, so that it can be picked up by someone else.
- Axiell Move Premium: mission items (individual actions) can be refused (because they can't be executed normally for example) and you may add a comment to your refusal.
- Axiell Move Premium: a move is now split up into a pickup action and a delivery action, pickup representing the loading of containers or objects onto a trolley while delivery represents the offloading of those items onto the final destination. This provides a better registration of the move process.
- Axiell Move Premium: pop-up confirmation is now optional (via the *Confirm pickup & delivery* option in the Axiell Move settings). Without confirmation you can work faster.
- Axiell Move Premium: besides scanning each object or package to pick up or deliver, you can work even faster and just open a sub-menu for a move action and click *Pick up* or *Deliver*. The destination location must have been scanned first though, for safety reasons.
- Management of iOS 13 dark mode.
- Some display issues on last iOS version have been fixed.
- Miscellaneous bug fixes.



## 2. Working with Axiell Move

The first registration of objects, containers/packages and locations must take place in one of the appropriate Adlib, Emu or Mimsy Axiell products. This manual only illustrates the combination of Axiell Move and Axiell (Adlib) Museum 4.5.

In Museum application 4.5, objects can be described in the *Internal* and *External object catalogue* data sources; containers and locations can be described in the *Locations and containers* data source – the content of the *Location or container* field determines if the record pertains to a container or a location – and package types in the *Container types* data source. (In older or newer applications, these data source and fields may have different names.) All objects, containers and locations that you may want to work with in Axiell Move, must have been registered in the Museum application first: you cannot do that registration from within Axiell Move. With Axiell Move you apply changes to existing records.

So data scanned or entered via Axiell Move on the iPhone, will be added to existing object and container records, or it will replace existing data in there. In principle, some data can be edited or entered in Axiell Move as well as in the Museum application, but each application has its own advantages: in Axiell Move you can register the packing and moving of objects quickly and efficiently on-site, while the Museum application is meant for the registration and management of entire records.

The essence of Axiell Move is that it allows you to change the registered *Current location* of an item to a new current location, because in reality you are moving that item as well. In this definition, an item can be a bare or packaged object or an empty container. The basic concept is that all movable items have a current location and that the current location of an item inside a container its packaging is: a vase packed in a small crate, has the crate as its location, while that crate itself has a location of its own as well, maybe a shelf or a larger overpack. That's why the three basic actions available in Axiell Move – packing, moving and unpacking – are all in fact a type of moving: in the record of the moved (meaning packed, moved or unpacked) items, the *Current location* will be changed and the *Location history* will be updated.

In more technical wording we can say that every record of a movable item has to be linked to a location or container record and that, at the same time, a reverse link between the location or container record and the item record exists. During the move of an item, those links

will be broken, after which a new link between the item record and the new location or container record will be created.

- In an object record in the Museum application, the current location or package as well as the normal location and planned future movements, can be found on the *Location | Future movements* screen tab, but only the current location can be changed by Axiell Move; the normal location and any planned future movements are in no way involved in transactions in Axiell Move. When you move an object with Axiell Move, or put it in a package, the new location or container will become the new current location of the object. The details of the previous location or container of the object will then be transferred automatically to the *Location history* tab. This way you can always track where an object has been, when and for how long.
- In a container record in the *Locations and containers* data source in the Museum application, the current location or overpack can be found on the *Container details* screen tab. When you move an empty or filled package with Axiell Move, or put it in an overpack, the new location or overpack will become the new current location of the package. The details of the previous location or overpack of the package will then be transferred automatically to the location history on the same tab.
- In a location record in the *Locations and containers* data source in the Museum application, details about the location type and the location hierarchy can be found on the *Location details* screen tab. With Axiell Move you can't change anything in that hierarchy.

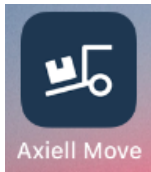
In an object or container record in the Museum application it is possible in principle to link to another current location or (over)package manually, but then the location history won't be updated automatically (!) and you would have to do that manually as well.

With the *Change locations* procedure in Museum applications 4.2 and higher you can apply location changes in batch to object records (not to container records), and also have the location history updated automatically. For more information about this, see the *Change locations* topic in the Axiell Collections Help.

---

## 2.1. Starting Axiell Move

1. Swipe through the apps on your iPhone to locate the Axiell Move app. Click the icon to start the app.

2. Axiell Move opens with the login screen. Note that the language of the user interface is the same as the language set for the iPhone itself, but limited to either English, Dutch, French or German. Tap the *User name* entry field to activate it and to open the virtual keyboard: now enter your user name (without any domain). Instead of typing your user name you can also scan a barcode (using the scanner sleeve) representing your user name, if you have such a barcode. Then click *Return* to activate the next entry field

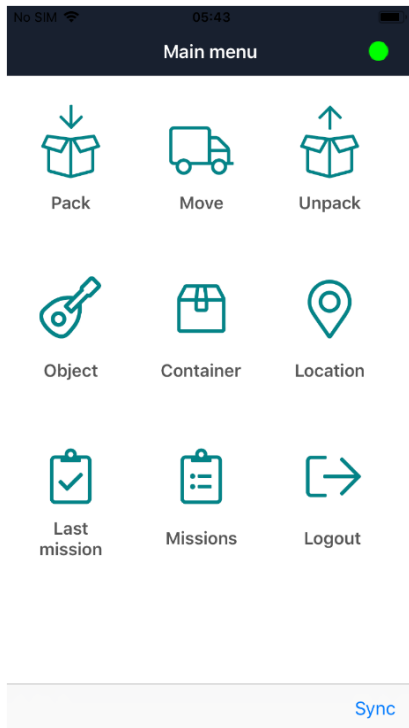
and enter your password as well. For the password only asterisks will appear, so that no-one can read your password from the screen. Then click the *Login* button on screen to access Axiell Move.

The user name and the encrypted password of the last 10 users on this iPhone, will be remembered by Axiell Move. This allows users to log on to the client even if the iPhone is offline. The name of the last or current user will have been filled in already in the *User name* field when you start or activate Axiell Move, but the password must always be entered manually again, even when you're logged in and return to the login screen.

When you or someone else has logged on to Axiell Move once before by entering a user name and password, and one or more fingerprints for Touch ID have been registered on your iPhone (see chapter 3.3) then every time you start Axiell Move you'll be offered the possibility to log in via Touch ID by simply resting your (registered) fingertip on the Home button.

The message will disappear of itself once your fingerprint is recognized. If you cannot or don't want to log in via your fingerprint, then simply click *Cancel* to log in the normal way. Further note that using Touch ID means that you automatically log in under the name and password of the previously logged in user (even if you're somebody else) and that everyone with access to your iPhone who knows your access code, will be able to register their own fingerprint allowing them to log on to Axiell Move. This also means that you should never share your access code to an iPhone with fingerprint recognition or that an iPhone with fingerprint recognition shouldn't be shared with colleagues at all or that your shared iPhone with fingerprint recognition is allowed to log in via a single general user name and password so that all users may log in via their fingerprint and transactions may be logged under that single name. An Apple device without fingerprint recognition does not have these limitations.

## 2.2. The Main menu



Axiell Move opens with the *Main menu*. From here, you choose the task at hand: packing (to link objects to containers), transport (to move objects, packaged or not) or unpacking (to break the link between objects and a container). You can also retrieve object information, container information or location information (for your information or for checking); these three options are only active though if there is a connection with the database and you have chosen to work online. See chapter 1.2 for information about how to go online if the iPhone is currently offline. The Premium version of Axiell Move further offers the *Last mission* icon to open the last mission you were working on and the *Missions* icon to download new missions from the server or select one of the earlier downloaded missions to start or continue working with. Use *Logout* to quit working in the app and to prevent others from registering Move transactions under your account on this phone.

- If you are working offline, your changes will temporarily be stored in a transaction log file on the iPhone until you can actually process the changes in the database (see chapter 2.7). The dot in the top right of the screen indicates whether you have access to the Axiell Move service or not: if it is green, you have access; if it is red, you have no access and the program automatically defaults to offline mode. The connection with the Axiell Move server will be checked every 5 seconds, during which time the dot may turn orange briefly. Press the green dot to go offline temporarily: the dot turns blue. Press the blue dot again to go back online. Having no access (red dot) can be a consequence of a missing wireless network connection or a consequence of an incorrectly set up Axiell Move service.

Note that during offline work, no information can be retrieved from the database either: if you scan the barcode of an object now, then its accompanying data (and possibly an image) from the record cannot be displayed. Only the scanned barcode of the object will be shown, until you scan another barcode.

Aside from not being able to retrieve information, offline mode in principle allows you to register the same transactions as you would online. However, some limitations to those transactions cannot be managed immediately when you are working offline, for instance when you would try to unpack an object that isn't registered as being packed in the first place. Such checks will then only be performed when you synchronize the transaction log file with the database.

- With the *Sync* option (bottom right) you'll open the *Synchronization* screen. When due to offline work a transaction log file has been created, you can view that log here and still process it in the database now or later (if you're online) or remove it if the transactions have become redundant.

## Sync

- Using the < *Main menu* option on the top of other screens, you can always return to this screen to start a different task. If you are deep in the processing of a particular task, you'll sometimes have to return to a previous step first, via <, before you get the option to return to the main menu. So use < to go back a step.
- When you are finished with your registration tasks and you don't want others with access to this iPhone to work with Axiell Move using your login details, then simply return to the main menu and click *Logout*. This will log you off and the password field will be



emptied. You can log in again by re-entering the password and clicking *Login* or *Return*. A different user may log in by using his or her own user name and the accompanying password.

- If you want, you can close the app completely as well. You'll have to do that the iOS way: quickly press the Home button of the iPhone twice, swipe left or right to the active Axiell Move app and swipe it upwards to really close it.

---

## 2.3. Retrieving object or container information



Object



Container



Location

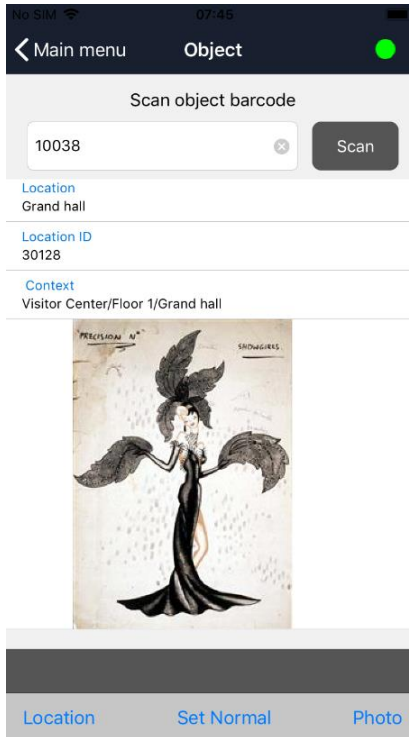
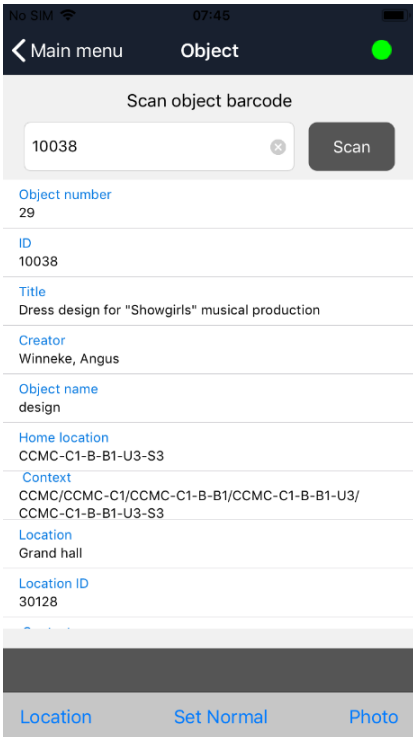
Via *Object*, *Container* and *Location* in the *Main menu* of Axiell Move, you can retrieve information about objects, containers and locations from the database, if you are working online. This allows you to check whether the object or box in front of you, is the right one. You can also add a photo, correct the current location of an object or container, or correct both the current and normal location of an object or container simultaneously.

Select the desired option and simply scan the barcode or RFID tag of an object, container or location and selected data will be retrieved and displayed. Swipe upwards in the displayed data to show the rest of it, if not everything is visible at once.

Scan either by clicking the *Scan* button on screen and positioning the iPhone camera over the barcode or keep the scanner sleeve button pressed down while you aim the front of the scanner at the barcode.

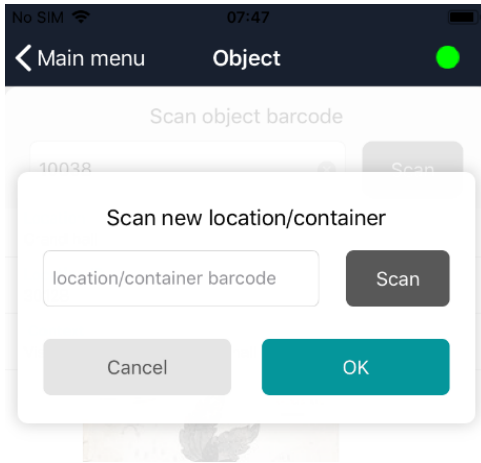
You can use this screen to retrieve details from other records of the same type as well, by just scanning another object, container or location.

Instead of scanning, you may also click the *Barcode* entry field and enter the barcode number manually, followed by *Return* or *Search*.



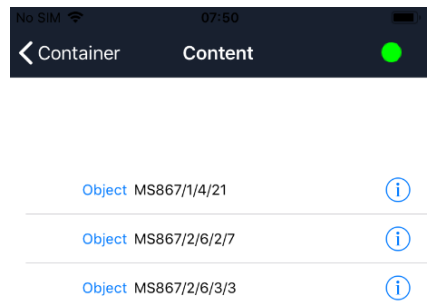
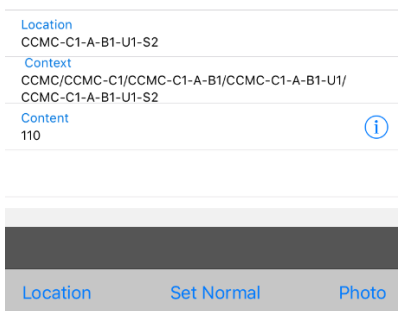
With the *Photo* option, in the bottom right of the screen, you can take a picture or make a video on the spot, of the relevant object, container or location. After taking the photo, select the *Use photo* option to upload the picture to the *\images* folder of your Adlib system and have a reproduction record created for it in the *Multimedia documentation* database, which in turn will be linked to the relevant object-, container or location record: Axiell Move will notify you briefly of the upload.

The *Location* option (only available for objects and containers) allows you to scan a new location for the object or container. This might be useful when you've located the item on a different location than the one registered in the relevant record and you'd like to correct that error immediately. Click *OK* to accept the scanned location or click *Cancel* to cancel the entry of a new location.



*Set Normal* (only available for objects and containers) does the same thing as the *Location* option, but it also changes the normal (home) location of the current item to the newly scanned location. So after clicking *Set Normal*, you'll get to scan a single location and this location will then be registered in the object or container record as both the *Normal location* as well as the *Current location*. The change is reflected immediately in the retrieved item details in the app too.

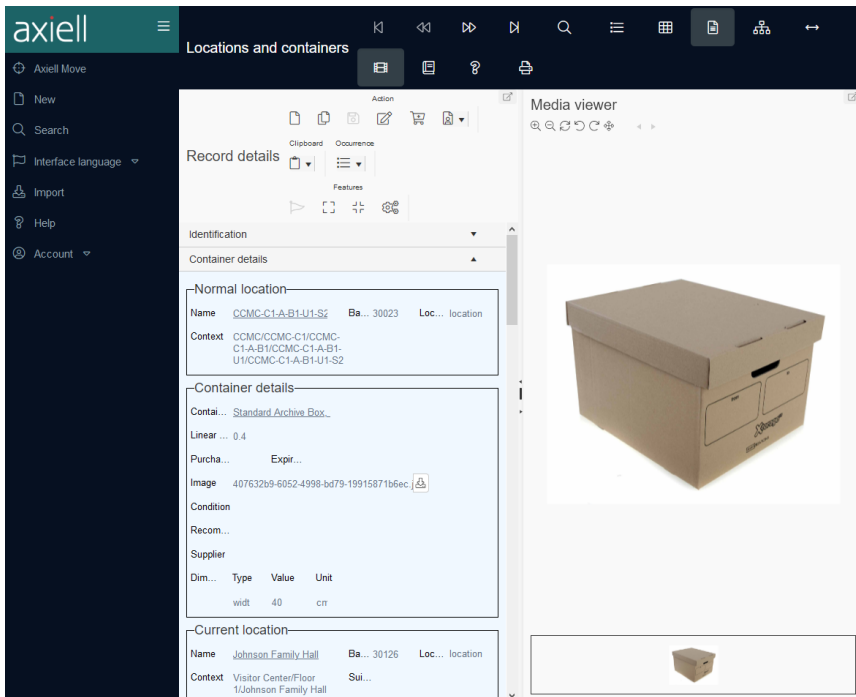
On the *Location* and *Container* screens, amongst others, you will also find the number of items which have this location or container as their current location. Click the ⓘ icon behind *Content* to display the actual list of containers and/or objects: swipe up to see the entire list (max. 50 items by default, but this can be increased in the Axiell Move server configuration). In this list you may click ⓘ again behind an object or container for more information about it.



Click the < icon or swipe right to return to the previous screen.

## 2.4. Packing

With the *Pack* option you can link containers to objects or link a larger container/overpack to already packaged objects (you then link to the package, not the object). Containers must have a unique reference, like a barcode for instance. So every individual container must have its own database record in which details about the relevant packaging have been registered. In Museum application 4.5 this database is called *Locations and containers*: both container records as well as location records are stored in here.



The screenshot displays the Axiell Move application interface. The main window shows the 'Locations and containers' record details for a container. The interface includes a sidebar with navigation options (New, Search, Interface language, Import, Help, Account) and a top navigation bar with various icons. The main content area is divided into several sections:

- Record details:** Includes 'Identification', 'Container details', and 'Current location' sections.
- Normal location:**
  - Name: CCMC-C1-A-B1-U1-S2
  - Ba...: 30023
  - Loc...: location
  - Context: CCMC/CCMC-C1/CCMC-C1-A-B1/CCMC-C1-A-B1-U1/CCMC-C1-A-B1-U1-S2
- Container details:**
  - Contai...: Standard Archive Box
  - Linear...: 0.4
  - Purcha...: Expir...
  - Image: 407632b9-6052-4998-bd79-19915871b6ec (with download icon)
  - Condition:
  - Recom...:
  - Supplier:
  - Table with columns: Dim..., Type, Value, Unit. Row: width, 40, cm
- Current location:**
  - Name: Johnson Family Hall
  - Ba...: 30126
  - Loc...: location
  - Context: Visitor Center/Floor 1/Johnson Family Hall

On the right side, there is a 'Media viewer' showing a 3D rendering of a cardboard box. Below the media viewer, there is a small thumbnail image of the same box.

It is handy (but not mandatory) if a container has a location, so that of an empty package and especially of an object you pack in it, a location is always known. In any case it creates the necessity to register the relocation of both empty as well as filled containers, so that the location of any content can always be deduced.

With a packaging transaction, the location of a package will not be transferred to the record of the object being packed, nor vice versa. Only the following changes will be transferred:

- When an object is being packed, the location of the package remains what it was, while the new current location of the object becomes the current package. So the location registered in an object record can be a *location* or a *container*. This is indicated in the *Location/package* field of the *Current location* and the *Location history* of the object.
- When a smaller, already packed object is being packed in an outer overpack, again the location of the overpack remains what it was and the location of the object doesn't change either, while now the new current location of the smaller package becomes the outer overpack. So a container record has its own current location and location history, and that location can be a *location* or a *container*.

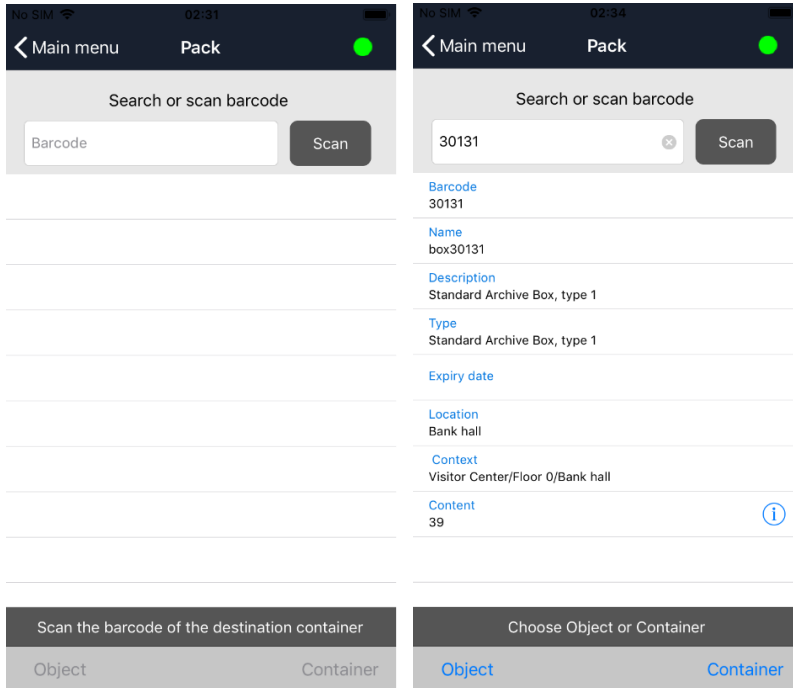
If you want to find out in the Museum application what the location of a packed object is, you must search for it in the relevant container record in the *Locations and containers* database. If the location itself is an overpack, then open that record to find the location of the overpack, etc. For a displayed record in *Locations and containers* you can also open the *Hierarchy browser* (F9) for a succinct presentation of the complete container/location hierarchy of the current container or location.

1. Choose the *Pack* option in the *Main menu* to get going.



Pack

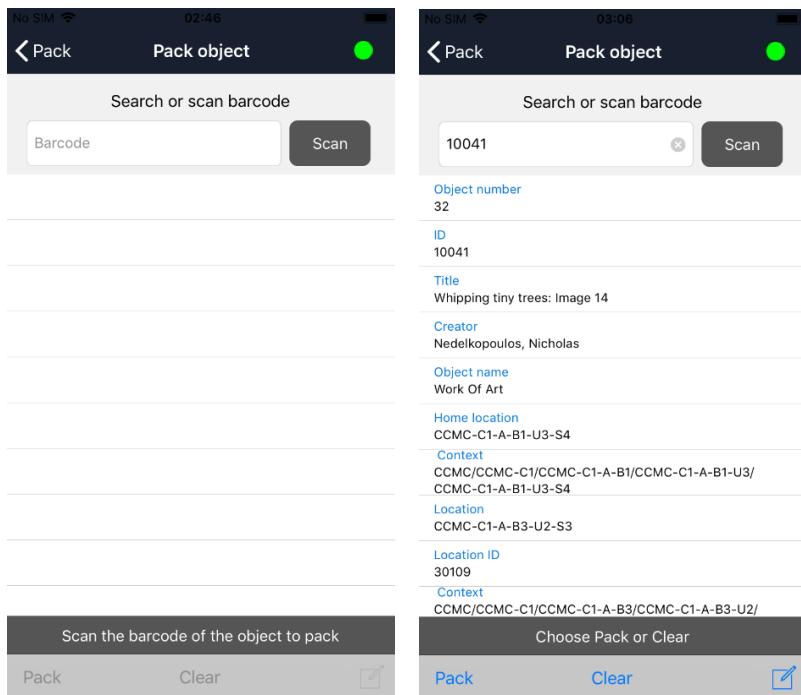
2. The *Pack* screen opens. To link a package to one or more objects, you will first have to scan the barcode of the relevant container or manually enter the unique identification code (barcode) of the packaging via the virtual keyboard (the cursor must be blinking in the *Barcode* field). Some properties of the container will be retrieved (if possible) from the database and displayed. If the barcode cannot be found in the database, you'll be notified so: it is possible that there is no container record with this barcode yet. In that case you'll have to register it through the Museum application first.



### 3. Now you can do five things:

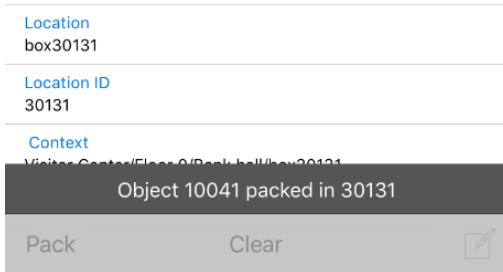
- a. **Object:** click *Object* if you want to link this container to one or more (unpacked) objects. Then proceed to step 4.
  - b. **Container:** click *Container* if you want to link this overpack to one or more already packaged objects. Continue with step 4.
  - c. You could scan another container if you don't want to do anything with the currently selected container after all.
  - d. Click the ⓘ icon behind *Content* to display the actual list of containers and/or objects in this container: swipe up to see the entire list. In this list you may click ⓘ again behind an object or container for more information about it.
  - e. **< Main menu:** click < *Main menu* or swipe to the right if you don't want to do anything with this container after all and you want to stop packing.
4. Scan an unpacked object (if you chose *Object* in step 3) or smaller package of one or more objects (if you chose *Package* in step 3)

to enter its identification number in the *Barcode* field. After scanning an object or smaller package, some of its details are shown (if you are working online): if your confirmation for the transaction is required\* you can use those to check whether you have the right item in front of you.



Aside from going back to the previous screen because you don't want to pack the scanned object after all, the following three options are now available to you (if your confirmation for packing is required\*):

- a. **Pack:** to confirm that you want to link the selected package to this object or smaller packaging of objects, click *Pack*. It takes a moment before the registration has finished and you'll see a message stating that the relevant item has been packed.



If you want, you can scan the next item immediately to pack it in the current container too by clicking *Pack* again after scanning, etc.

In the relevant object record in the object catalogue of your Museum application, you can observe that the container has been linked to this object as its current location. The details of the previous location have been transferred to the location history of the object.

In case you have been packing an already packaged object in an overpack, you can observe that the overpack has been linked to the smaller package as its current location, in the record of the smaller package in the *Locations and containers* data source. The details of the previous location have been transferred to the location history of the smaller package.

- b. **Clear:** click *Clear* if you don't want to link the currently displayed item to the package, in order to pick another item. You'll then be able to scan a different unpacked object if you also scanned an unpacked object just before, or scan a different already packaged object if you also scanned an already packaged object just before.
- c. **Notes:** click the *Notes* icon if you want to register remarks about the packing.



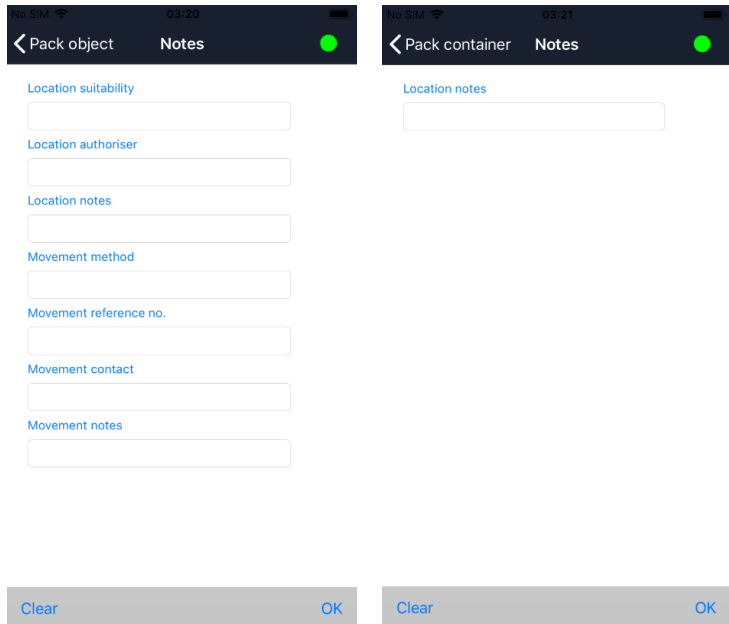
You can enter those notes in a new screen. The screen is different for objects and smaller packages (see screenshots below):



**Objects** – By default there are seven notes fields available. The first three fields are about the container (a type of “location”) in which you are momentarily packing objects, and this data will be stored in the object record with the current location. The last four fields, on the other hand, are about the actual act of packing (a type of “movement”), and this data will be stored in the object record with the previous location in the location history.

**Smaller containers** – By default there’s only a single notes field available here. This field is about the overpack (a type of “location”) in which you are momentarily packing a smaller packaged object, and these notes will be stored in the record of the smaller package with the current location.

Click *OK* to keep the entered data or click *Clear* to throw the entered data away. Click < or swipe right to return to the previous step, if required. The data you keep will only be saved in the item record when you actually pack the scanned item by clicking the *Pack* button.



5. Click the < *Pack* option at the top of the screen or swipe right, if you are done with the current container. You'll be able to scan a new container or return to the main menu.

By the way, a means of transport like a truck can also be registered as a "package" with a location. When you load all overpacks in the truck, you are sort of packing them in the truck, and during unloading you are sort of unpacking them from the truck. With the *Move* option you can change the location of the truck after transportation.

\* In the configuration file for the Axiell Move server there's a setting which determines if confirmation is required for the last step in packing, moving and unpacking. If not set, a scanned object is packed automatically and you cannot check its details beforehand or add notes.

## 2.5. Moving

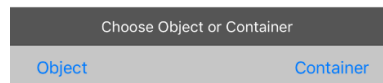
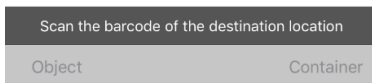
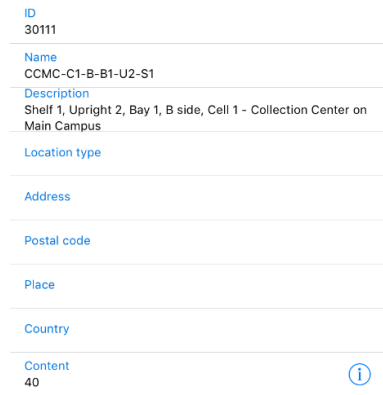
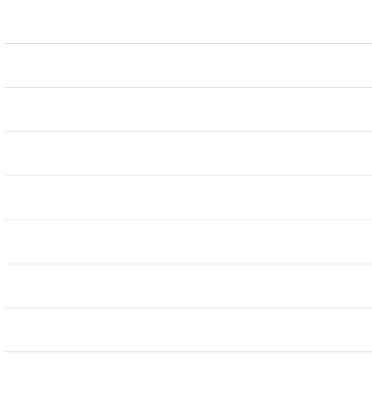
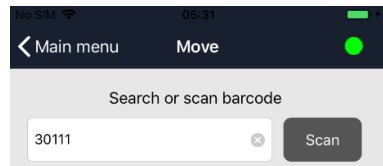
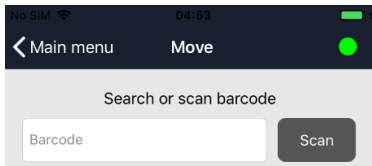
With the *Move* option you register the new location of objects (packed or not) which have just been moved in the real world. You register the items (crates, boxes and any unpacked objects) as they arrive after the relocation, still before you unpack anything. The point is to register the new location of all these (usually packed) items on reception, even if that is just a temporary storage space. Later, when you unpack objects, you can assign a new location per object again.

1. Select the *Move* option in the *Main menu*.



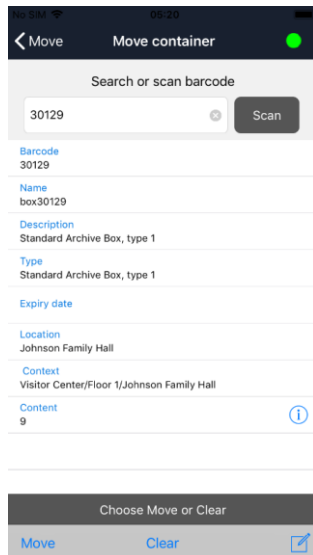
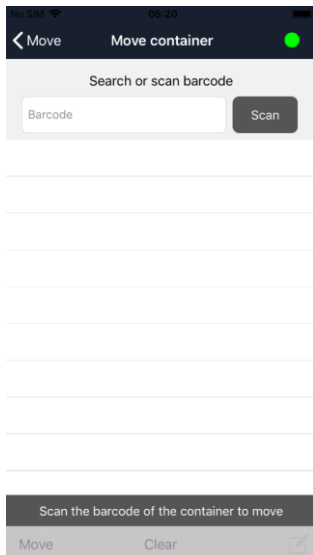
Move

2. The *Move* screen opens. Scan the barcode or RFID tag of the new location: this is the location (not a container) where the shipped or received items will be stored now.



If the barcode cannot be found in the database, you'll be notified so: it's possible that there is no location record with this barcode yet. If that is the case, you'll have to register it through the Museum application first.

3. Now you can do five things:
  - a. **Object:** you want to link this location to one or more bare objects. Click *Object* and proceed to step 4.
  - b. **Container:** you want to link this location to one or more packaged objects. Click *Container* and continue with step 4.
  - c. You could scan another location if you don't want to do anything with the currently selected location after all.
  - d. Click the ⓘ icon behind *Content* to display the actual list of containers and/or objects on this location: swipe up to see the entire list. In this list you may click ⓘ again behind an object or container for more information about it.
  - e. **< Main menu:** on second thought you don't want to do anything with this location and you want to stop moving. Then click < *Main menu* or swipe right at the top of the screen.
4. Scan, depending on your choice in step 3, the barcode of an object or container: if you are working online and if your confirmation for the transaction is required\* you can check the details.



Besides the option to go back to the previous screen, if you do not want to move the scanned item after all and possibly to scan a new location, the following three options are now available to you (if your confirmation for moving is required\*):

- a. **Move:** to confirm that you want to link the new location to this object or container of objects, click *Move*. It takes a moment before the registration has finished and you'll see a message stating that the relevant item has been moved. If you want, you can immediately scan the next item that has been moved to the currently selected location, after which you must click *Move* again to register it, etc.  
In the relevant object record (of an unpacked object) in the object catalogue of your Museum application, you can observe that the currently selected location has been linked to this object as its current location. The details of the previous location have been transferred to the location history of the object.  
In case you have been moving a packaged object, you can observe that the currently selected location has been linked to this container as its current location, in the record of the container in the *Locations and containers* data source. The details of the previous location have been transferred to the location history of the container.
- b. **Clear:** click *Clear* if you don't want to move the currently displayed item, in order to pick another item. You'll then be able to scan a different unpacked object if you also scanned an unpacked object just before, or scan a different packaged object if you also scanned a packaged object just before.
- c. **Notes:** click the *Notes* icon if you want to register remarks about the movement.



You can enter those notes in a new screen. The screen is different for bare objects and smaller packages (see screenshots below):

**Bare objects** – By default there are seven notes fields available. The first three fields are about the currently selected location, and this data will be stored in the object record with the current location. The last four fields, on the other hand,

are about the actual act of moving, and this data will be stored in the object record with the previous location in the location history.

**Packaged objects** – By default there's only a single notes field available here. This field is about the currently selected location, and these notes will be stored in the record of the container with the current location.

Click *OK* to keep the entered data or click *Clear* to throw the entered data away. Click < or swipe right to return to the previous step, if required. The data you keep will only be saved in the item record when you actually move the scanned item by clicking the *Move* button.

The image shows two screenshots of the Axiell Move app's 'Notes' screen. The left screenshot is for 'Move object' and the right is for 'Move container'. Both screens have a dark header with a back arrow, the title, and a green indicator light. The 'Move object' screen has several input fields: 'Location suitability', 'Location authoriser', 'Location notes', 'Movement method', 'Movement reference no.', 'Movement contact', and 'Movement notes'. The 'Move container' screen has a single 'Location notes' input field. At the bottom of each screen is a grey bar with 'Clear' on the left and 'OK' on the right.

5. Click < at the top of the screen or swipe right if you are done with movements to the current location. You'll be able to scan a new location or return to the main menu.

\* In the configuration file for the Axiell Move server there's a setting which determines if confirmation is required for the last step in packing, moving and unpacking. If not set, a scanned item is moved automatically and you cannot check its details beforehand or add notes.

## 2.6. Unpacking

With the *Unpack* option you break the link between one or more items (smaller packaged objects or bare objects) and the current (over)-pack. You always provide the new location of every unpacked item. The location of the (over)pack remains what it was. By the way: any smaller packages which you take from the overpack do not necessarily have to be unpacked themselves too.

1. Choose the *Unpack* option in the *Main menu* to begin.



Unpack

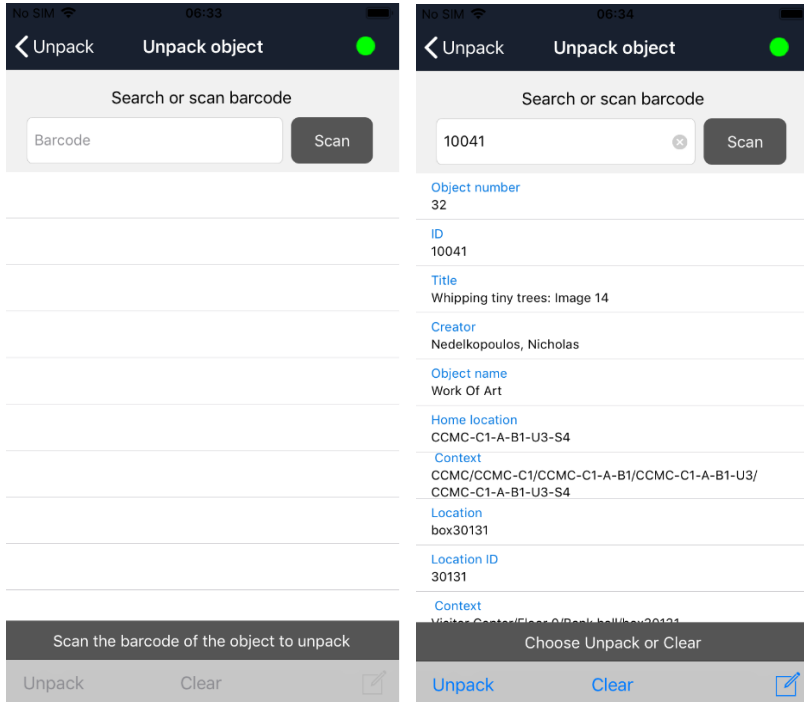
2. The *Unpack* screen opens. First scan the barcode of the new location for the items you are about to unpack. Axiell Move will check whether the location has been registered in the database and then displays it on screen.

|  |  |
|--|--|
| <p>No SIM 06:15</p> <p>Main menu Unpack</p> <p>Search or scan barcode</p> <p>Barcode <input type="text"/> Scan</p>   | <p>No SIM 06:16</p> <p>Main menu Unpack</p> <p>Search or scan barcode</p> <p>30112 <input type="text"/> Scan</p>   |
| <p>ID</p> <p>30112</p> <p>Name</p> <p>CCMC-C1-B-B1-U2-S2</p> <p>Description</p> <p>Shelf 2, Upright 2, Bay 1, B side, Cell 1 - Collection Center on Main Campus</p> <p>Location type</p> <p>Address</p> <p>Postal code</p> <p>Place</p> <p>Country</p> <p>Content</p> <p>0</p> | <p>ID</p> <p>30112</p> <p>Name</p> <p>CCMC-C1-B-B1-U2-S2</p> <p>Description</p> <p>Shelf 2, Upright 2, Bay 1, B side, Cell 1 - Collection Center on Main Campus</p> <p>Location type</p> <p>Address</p> <p>Postal code</p> <p>Place</p> <p>Country</p> <p>Content</p> <p>0</p> |
| <p>Scan the barcode of the destination location</p> <p>Object Container All</p>  | <p>Choose Object, Container or All</p> <p>Object Container All</p>   |



If the barcode of the location cannot be found in the database, you'll be notified so: it's possible that there is no location record with this barcode yet. If that is the case, you'll have to register it through the Museum application first.

3. Now you can do six things:
  - a. **Object:** you want to unpack one, or just a few, bare objects from the container at hand, to the currently selected location. Click *Object* and proceed to step 4.
  - b. **Container:** you want to unpack one, or just a few, smaller packages from the overpack before you, to the currently selected location. Click *Container* and proceed to step 4.
  - c. **All:** you want to unpack all items at once, to the currently selected location. Click *All* and continue with step 5.
  - d. Click the ⓘ icon behind *Content* to display the actual list of containers and/or objects on this location: swipe up to see the entire list. In this list you may click ⓘ again behind an object or container for more information about it.
  - e. You could scan another location if you don't want to do anything with the currently selected location after all.
  - f. **< Main menu:** on second thought you don't want to do anything with this location and you want to stop unpacking. Then click *< Main menu* at the top of the screen or swipe right.
4. If you chose *Object* or *Container* in the previous step, then now scan the barcode of the object or the smaller container. If you are working online, some details of the item are displayed and if your confirmation for the transaction is required\*, you can use those details to check if you've scanned the right item.



Apart from going back to the previous step because you don't want to unpack the current item after all, the following options are now available to you (if your confirmation for unpacking is required\*):

- a. **Unpack:** to confirm that you want to break the link between the selected item and the current (over)pack, and assign the new location to the item, click *Unpack*. It takes a moment before the registration has finished and you'll see a message stating that the relevant item has been unpacked. If you want, you can immediately scan a next item of the same type (container/object) and unpack it to the new location by clicking *Unpack*, etc.

Note that only the relation between the current (over)pack and the relevant item in it has been broken: if the item is a package itself, then for now the link between that smaller package and its content remains as it is.

The details of the previous packaging have been transferred to the location history of the unpacked item.

- b. **Clear:** click *Clear* if you don't want to unpack the currently displayed item from the package, in order to pick another item. You'll then be able to scan a different bare object if you also scanned a bare object just before, or scan a different packaged object if you also scanned a packaged object just before.
- c. **Notes:** click the *Notes* icon if you want to register remarks about the unpacking.



You can enter those notes in a new screen. The screen is different for bare objects and smaller packages (see screenshots below):

**Objects** – By default there are seven notes fields available. The first three fields are about the new location, and this data will be stored in the object record with the current location. The last four fields, on the other hand, are about the actual act of unpacking (a sort of “moving”), and this data will be stored in the object record with the previous location in the location history.

**Containers** – By default there's only a single notes field available here. This field is about the new location, and these notes will be stored in the record of the container with the current location.

Click *OK* to keep the entered data or click *Clear* to throw the entered data away; click *<* or swipe right in case you didn't return to the previous screen automatically. The data you keep will only be saved in the item record when you actually unpack the scanned item by clicking the *Unpack* button. Proceed to step 6.

The image shows two screenshots of the Axiell Move app's 'Notes' screen. The left screenshot is titled 'Unpack object Notes' and contains several input fields: 'Location suitability', 'Location authoriser', 'Location notes', 'Movement method', 'Movement reference no.', 'Movement contact', and 'Movement notes'. The right screenshot is titled 'Notes' and contains a single input field for 'Location notes'. Below the screenshots are two buttons, each with 'Clear' and 'OK' options.

5. If you chose *All* in step 3, then now scan the barcode of the (over)pack from which you want to unpack all items. If you are working online, some details of the package will be retrieved. You can click the ⓘ icon to display the actual list of smaller packages and/or objects in the current (over)pack.

The image shows a screenshot of the Axiell Move app's 'Unpack all' screen. At the top, there is a search bar with the text 'Search or scan barcode' and a 'Scan' button. Below the search bar, the barcode '30136' is displayed. The screen then lists details for the package: 'Name: box30136', 'Description: Standard Archive Box, type 2', 'Type: Standard Archive Box, type 2', 'Expiry date', 'Location: CCMC-C1-A-B1-U2-S2', 'Context: CCMC/CCMC-C1/CCMC-C1-A-B1/CCMC-C1-A-B1-U2/CCMC-C1-A-B1-U2-S2', and 'Content: 122'. An ⓘ icon is next to the 'Content' field. At the bottom, there is a button labeled 'Choose Unpack or Clear' with 'Unpack' and 'Clear' options.

Apart from going back to the previous step because you don't want to unpack all items from the scanned overpack after all, the following options are now available to you (if your confirmation for unpacking is required\*):

- a. **Unpack:** to confirm that you want to break the link between the current (over)pack and all items in it, and assign the new location to all those items, click *Unpack*. It takes a moment before the registration has finished and you'll see a message stating that the items have been unpacked. If you want, you can scan a next overpack immediately and unpack all items in it to the new location by clicking *Unpack*, etc.  
Note that only the relations between the current (over)pack and all items in it has been broken: if items are packages themselves, then for now the links between those smaller packages and their contents remain as they are. The details of the previous (over)packaging have been transferred to the location history of the unpacked items.
  - b. **Clear:** click *Clear* if you don't want to unpack the displayed overpack, in order to pick another overpack.
6. Click < at the top of the screen or swipe right once or twice if you are done with the selected location. You'll return to the *Main menu*.

\* In the configuration file for the Axiell Move server there's a setting which determines if confirmation is required for the last step in packing, moving and unpacking. If not set, a scanned item is moved automatically and you cannot check its details beforehand or add notes.

## 2.7. Missions

Choose the *Missions* option (Axiell Move Premium only) to download new missions or to start or continue working on downloaded missions. Or choose *Last mission* to continue working where you left off last time.

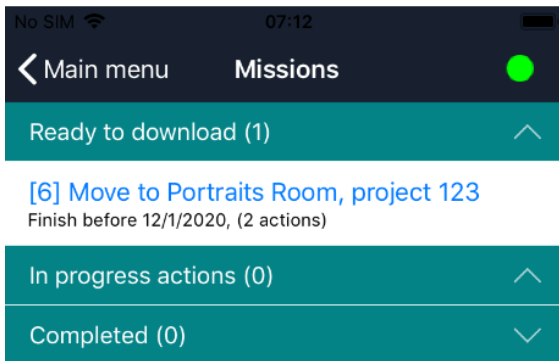


Last  
mission

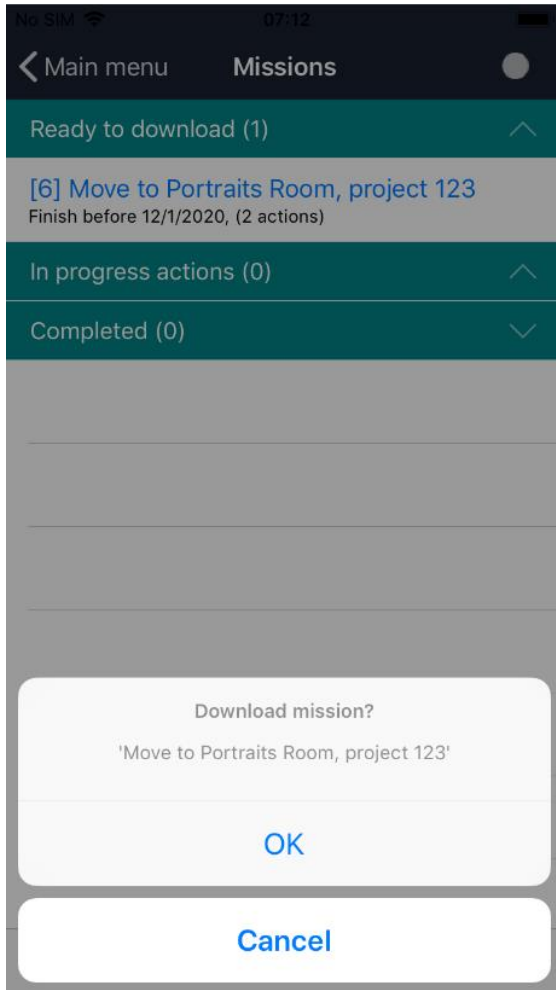


Missions

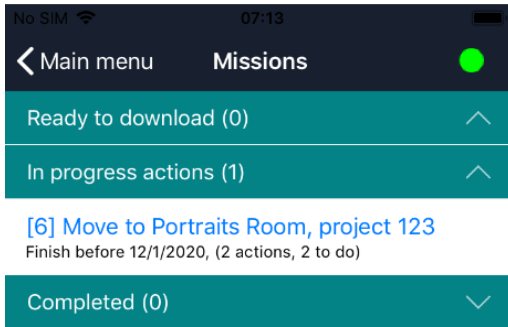
1. Choose the *Missions* option to open the *Missions* screen.



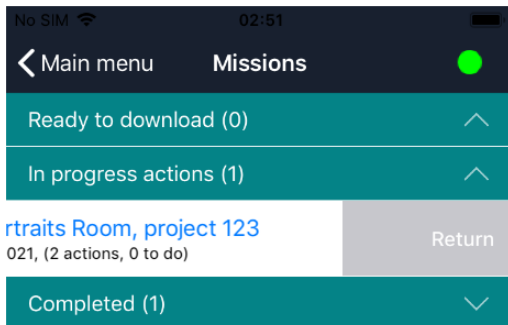
You'll get a list of downloadable missions (the *Status* in the record in Collections must be *ready for download*) and already downloaded missions. Missions may have been assigned to you in advance or may not have been assigned to one person in particular. A mission must have been downloaded to your device before you can start working on it. Click an already downloaded mission to start or continue working with it or click a downloadable mission to assume the work: you'll still have to confirm the downloading.



2. The downloaded mission is now listed under *In progress actions* and is ready for operation. Once downloaded, the mission can only be processed by you and other co-workers won't be able to download the same mission.

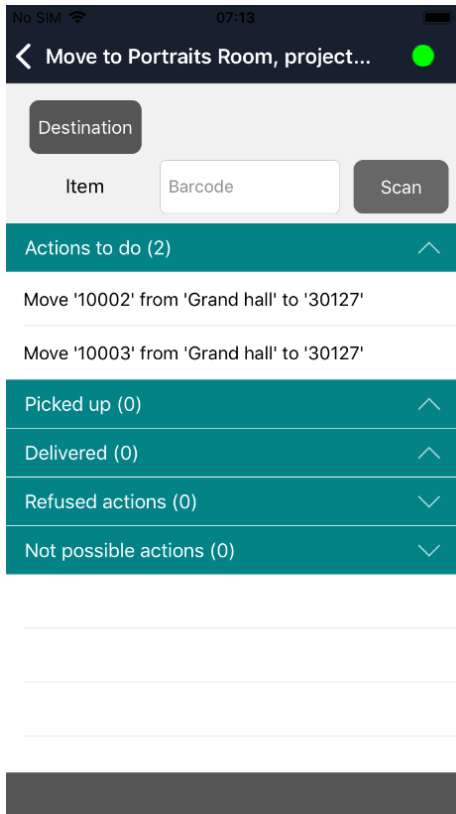


However, you can also release an already downloaded mission, if on second thought, someone else should do it. To do so, slide the relevant mission all the way to the left to reveal the *Return* option. (This option is only available when the device is online to ensure that all stored transactions are sent back to the server before the mission is released from the device.) The mission is then automatically returned and the status of the record is put back to *ready for download*. You can even use this option later in the mission process, if you can't complete it yourself, but take note that in real life the move process might be completed only half, with items on a trolley, so make sure you have that covered when you use the *Return* option halfway.

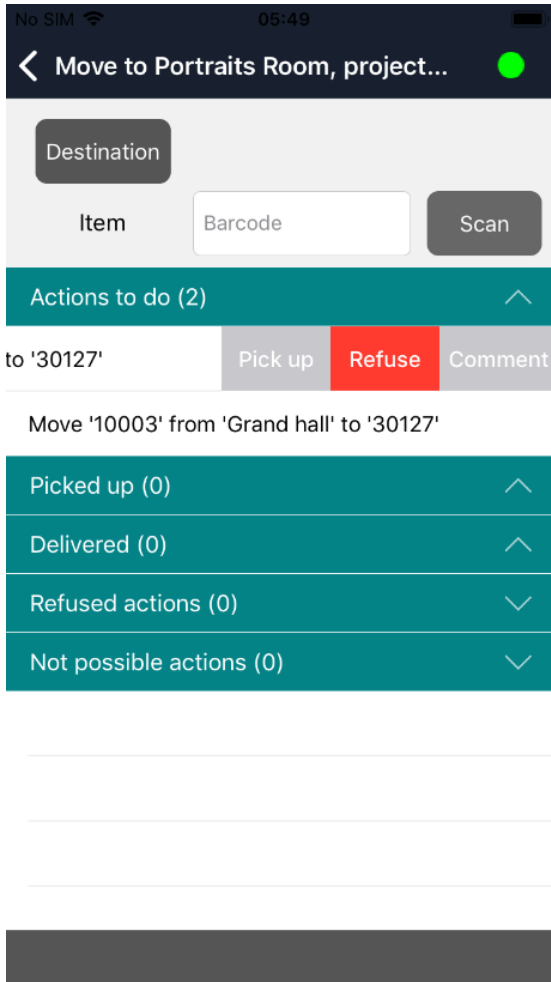


3. Click the downloaded mission to see the actual mission items it contains: the actions/tasks to perform. In this example there are two tasks which involve moving some object from one location to another.

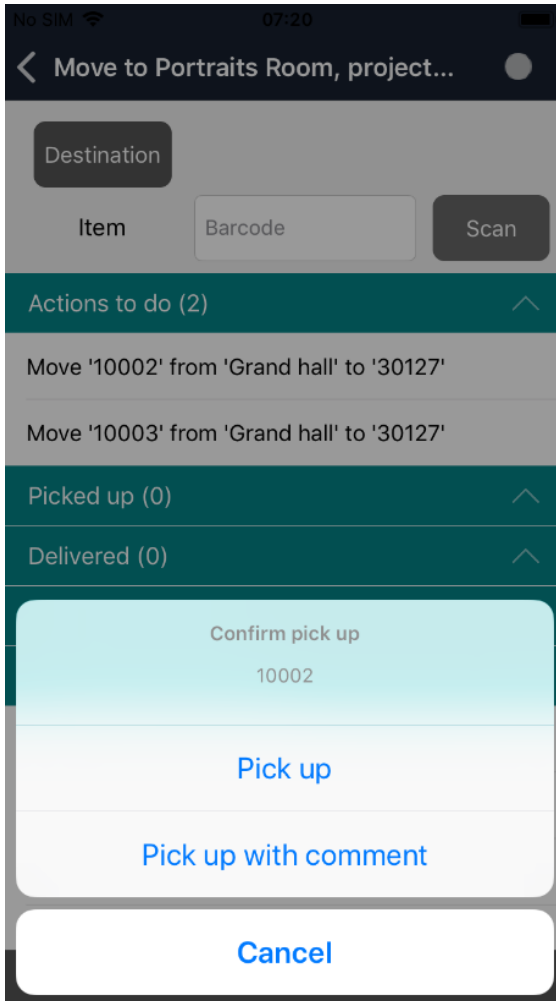




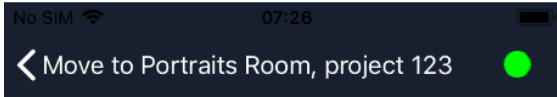
4. The actual tasks can now be performed in different ways. One way (which requires scanning the item), is to start performing the actual tasks by scanning one of the objects or containers to be moved, by clicking the *Scan* button. A move is divided up in a pickup action and a delivery action, allowing you to register your actions just after you performed them in real life. So typically you pick up all items to be moved first, put them on a trolley or something like that and register the pickup in the app. Another way (which doesn't require scanning the item) is to swipe left just a centimetre or so, to open a submenu with the options *Pick up*, *Refuse* and *Comment*. Click *Pick up* to start the pickup action without scanning the item. (If you swipe too far, the *Comment* option will be activated, but you can leave that screen by clicking *Cancel* if that was not what you wanted).



In both cases, you'll still have to confirm the pickup if the *Confirm pickup & delivery* option in the Axiell Move settings has been switched on: it allows you to confirm your action and possibly add a comment about the pickup itself which will be stored in the *Notes* field in the relevant mission item record in Collections. Otherwise no confirmation will be asked and the action is performed immediately.



The *Comment* option in the swipe-left submenu allows you an alternative way to add a comment. The comment added this way will only be saved in the mission item record after the pickup is completed. After entering and saving a comment, you can revisit it by clicking *Comment* again and you'll also see it again when clicking *Pick up with comment* in the confirmation. When adding a comment, click the *Comments* field to activate the virtual keyboard.



## Move 10002

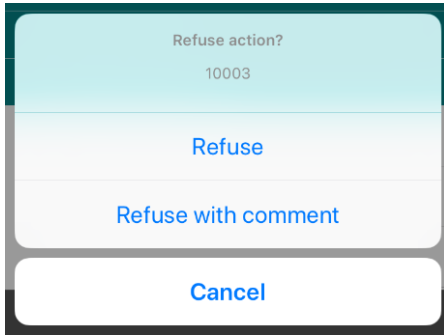
From            Grand hall  
To                30127

Comments

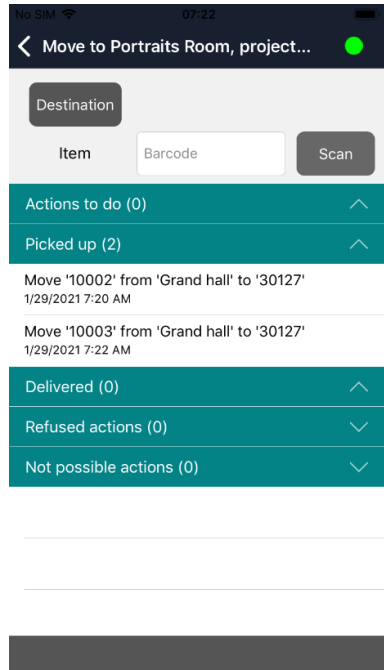
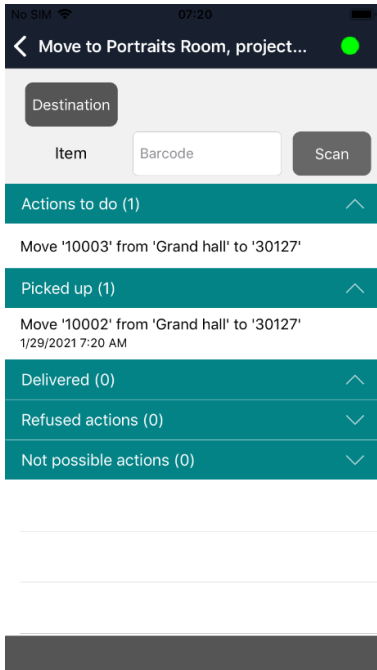
SAVE

Cancel

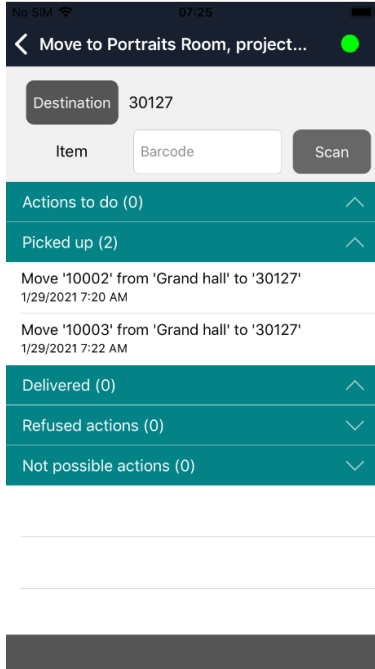
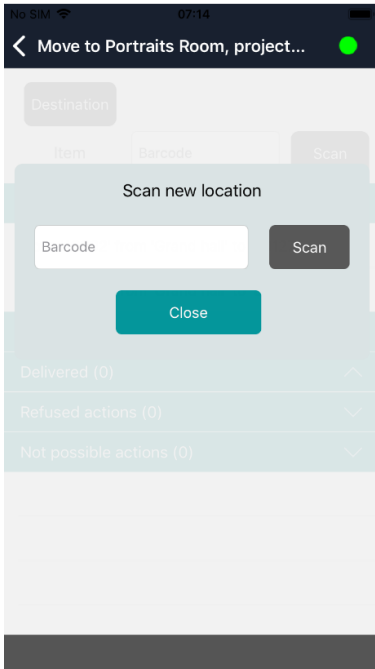
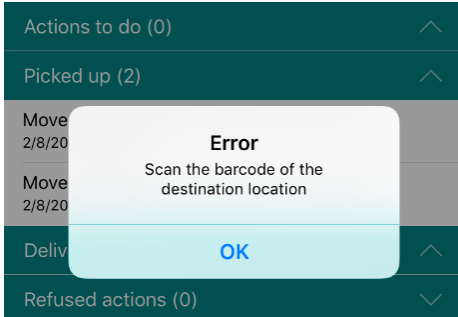
The *Refuse* option in the submenu discussed above is a way to refuse the selected action, for example because the item is missing, broken, too fragile or too heavy to be picked up by you. After clicking *Refuse* you still get to confirm the action and you may add a comment to explain the refusal. The status of the mission item record will change to *Cancelled*. In the app, refused actions will be listed underneath *Refused actions*.



The pickup means that the status of the mission item record in Collections will be changed to *Picked* and the action itself will be moved to the *Picked up* section in the Move app. Repeat this step for all actions to do. Note that a new status in a record which is already being displayed in Collections may only become apparent once you reload the record.



- 5. Picked up items can be delivered. But before you register the delivery in the app, scan the destination location by clicking the *Destination* button in the app. This is just a safety measure to make sure you're delivering the item to the correct location. If you haven't scanned the destination location yet and you try to deliver an item anyway, an error message will pop up to warn you to do that still.

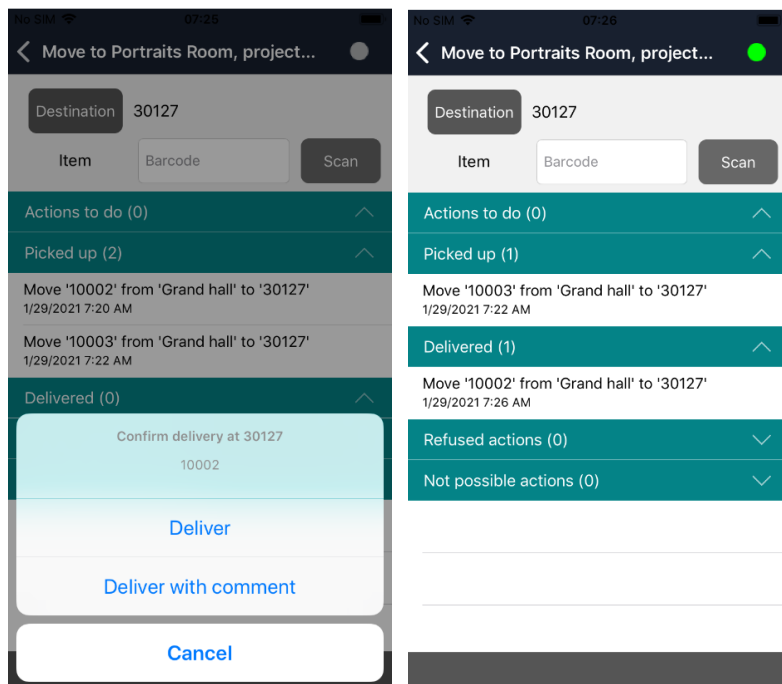


6. Now register the delivered items one by one, but only do this once you've delivered the items in real life. Again there are two ways of doing this. The first one is to click an action in the *Picked up* section and scan the item barcode.

The second way is faster because you don't need to scan the item. Swipe left just a centimetre or so, to open a submenu with the options *Deliver*, *Refuse* and *Comment*. Click *Deliver* to start the deliver action without scanning the item.

(If you swipe too far, the *Comment* option will be activated, but you can leave that screen by clicking *Cancel* if that was not what you wanted).

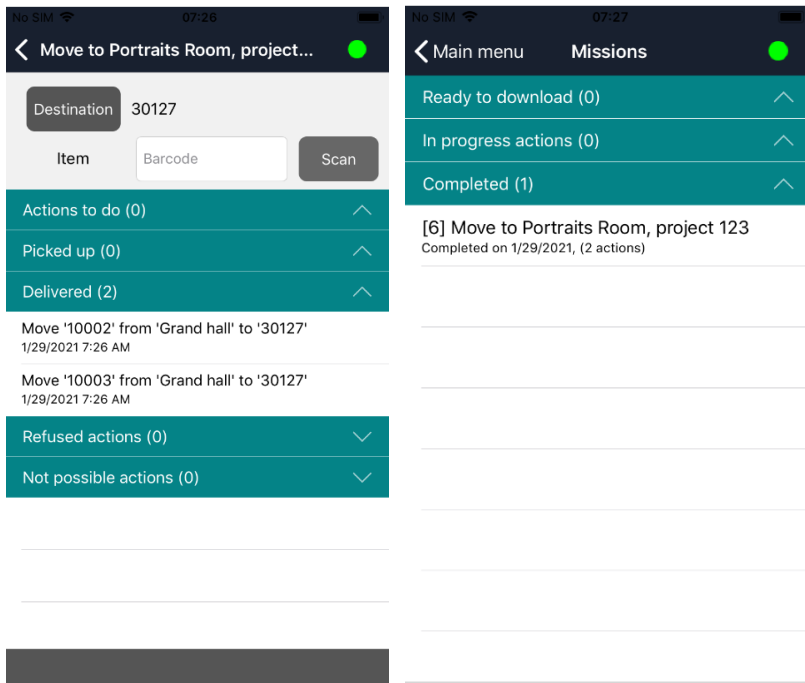
In both cases, as with the pickup of an item, you get to confirm the delivery only if the *Confirm pickup & delivery* option has been switched on in the Axiell Move settings, also allowing you to add a comment about the delivery itself which will be registered in the mission item record.



The *Refuse* option in the submenu discussed above is a way to refuse the selected action, for example because the item is broken or too heavy to be delivered by you. After clicking *Refuse* you still

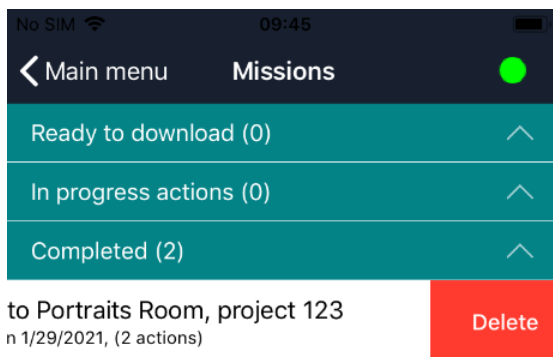
get to confirm the action and you may add a comment to explain the refusal. The status of the mission item record will change to *Cancelled*. In the app, refused actions will be listed underneath *Refused actions*.

- The delivery will be processed in the database, which means that the status of the mission item record in Collections will be changed to *Finished* and the action itself will be moved to the *Delivered* section in the Move app. Repeat this step for all actions to do. You may continue with the other *Actions to do* now or some time later until the entire mission is complete. Once the mission is complete and you return to the *Missions* overview (swipe right), the completed mission will be listed in the *Completed* section. Once completed you cannot view the parts of that mission anymore. And you're done with the mission work. (Note that the *Not possible actions* section is currently not being used.)

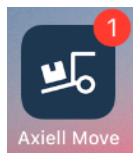




- Completed missions can be removed from the app by swiping the relevant complete mission left: this won't delete the mission from the database though, so you can still look it up in Collections.



## 2.8. Synchronizing offline transactions



If you have been working offline, your transactions will have been stored temporarily in a transaction log file on the iPhone. If you're not going back online during your current session and synchronize your offline transactions, then next time you start Axiell Move you may notice a small round badge with a number attached to the Axiell Move icon: this (optional) badge indicates that offline transactions awaiting your attention have been stored on the iPhone whilst the number indicates the number of stored offline transactions.

When you are back online, you can have all of the transactions from this file processed in the database at once. When you are logging in, the dot in the top right of the screen indicates whether you have access to the Axiell service or not: if it is green, you have access; if it is red, you have no access and the program automatically defaults to offline mode. Having no access can be a consequence of a missing wireless network connection or a consequence of an incorrectly set up Axiell service.

The *Synch* option in the bottom right corner of the *Main menu* opens the *Synchronization* screen in which you can still process stored trans-

actions in the database. The option is always active so that you can check any logged transactions.

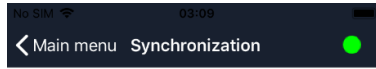
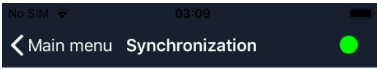
## Sync

You then have the following options:

- **Direct synchronization** – Click the icon in the bottom left corner of the screen to synchronize immediately.



Depending on the number of transactions, the processing can take a while. Once synchronization has been completed, the stored transactions will be removed. The number of processed transactions and the number of errors is displayed as well. The nature of any occurring errors will be shown too. Click *Main menu* to continue with your regular work.



|              |   |
|--------------|---|
| Transactions | 1 |
| Processed    | 0 |
| Errors       | 0 |

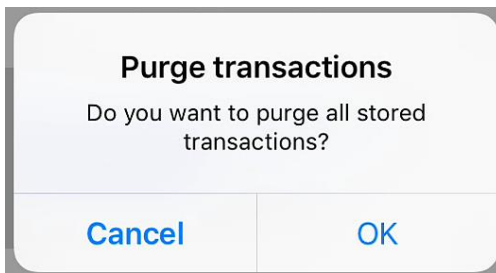
|              |   |
|--------------|---|
| Transactions | 0 |
| Processed    | 1 |
| Errors       | 0 |

Status for mission '1' set to 'ReadyForDownload'

2021-02-08 03:07:41 axiellcollections



- **Postpone synchronization** – If you don't want to synchronize just yet after all, for whatever reason, just click *Main menu* to continue with your regular work. You can click the *Sync* option in the *Main menu* at any time to have the transaction log processed in the database still.
- **Purge the transactions made offline** – If you want to irreversibly delete all transactions made on this iPhone while you were offline, for instance because you've been testing Axiell Move and you don't want to process the changes into the database, then click the garbage bin icon in the *Synchronization* screen. Subsequently click *OK* to delete all transactions made offline, or click *Cancel* to keep them still. In the latter case you must decide later if you want to synchronize the data after all or remove it.



After synchronization, the transaction log file is always removed, even if errors occurred during synchronization.

### ■ Synchronizing obsolete transactions

It is very well possible that you'll be synchronizing transactions which are wholly or partly obsolete. This might occur if after your offline transactions, other users with other iPhones have registered newer transactions in the database already. This is not a problem though. This is because on synchronization of offline transactions, the date and time of those transactions are taken into account by the Axiell Move server. If a transaction to be synchronized has taken place before the most recently registered transaction in the database, then the older transaction is simply inserted at the correct position in the location history.

Suppose you register offline that you are taking an object from room B to room C and you do not synchronize your data just yet. A day later you realize it should have been room D instead of C. You are online now and you register room D as the new current location of the object. In the database, room B will be transferred to the location history of the object. Only afterwards you still synchronize your earlier offline transaction. The Axiell Move server will note that the movement to room C took place before the movement to room D and will therefore maintain room D as the current location and register the older movement to room C in the location history of the object, at the chronologically correct position in that history.

Nonetheless it is of course best to work online as much as possible and to synchronize any offline transaction logs as soon as possible, especially if several co-workers are using Axiell Move. This is because all non-synchronized offline transactions might be unknown to your co-workers. It can be very confusing if objects cannot be found at the currently registered location because the most recent movement hasn't been registered in the database yet.

### ■ Repairing synchronization errors

During offline work, all transactions are basically approved, since Axiell Move does not have access to the database and is not able to check transactions anyway. This means that incorrect offline transactions go unnoticed: if you were to scan a package instead of an object or if you were to pack a location in an object, the error won't surface as long as you're working offline.

However, during online synchronization all checks will still be performed. Any errors in the transactions or errors during the synchronization of correct transactions (e.g. when a record to be updated is in edit mode already), are registered in the *admoved.log* in the *\Logs* subfolder underneath *\AdmoveServer* on the server\*, accompanied by the date and time. The details of the failed transaction itself are included. If the iPhone reports errors during synchronization, you will have to open the log file on the server to see what went wrong. Using that information, you should be able to reconstruct the problem and repair it. A repair could mean that you still make the correct transaction using Axiell Move, or that you edit object, location or package records manually. Other than this recommendation, it is not possible to provide a general instruction for correcting errors.

\* Note that the name and the location of the logging file have been specified during the Axiell Move server setup and may therefore be different.

## 3. Installation

---

### 3.1. Installing the Axiell Service (on the server)

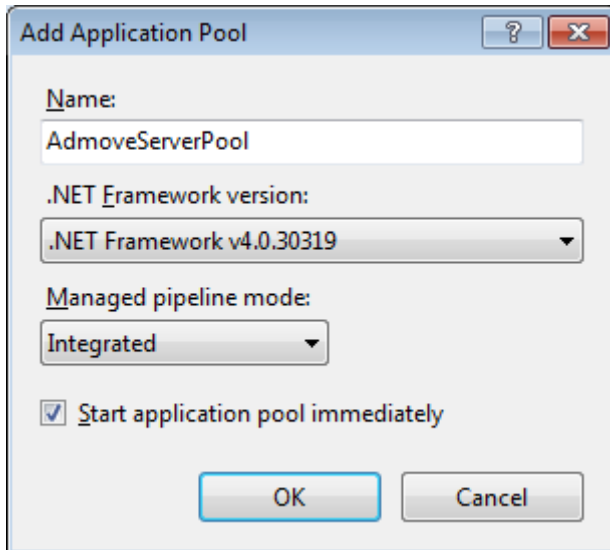
1. Preferably, a wireless network (Wi-Fi) is available to you. This network should have good coverage in all the places where you would like to work with the Axiell Move, and the server which will process your changes in the database must of course be accessible too, but only from within the local company network though, for security reasons. Outside the local Wi-Fi range, the iPhone user will then have to use a VPN connection via 3G/4G or external Wi-Fi network to connect to the server. However, there's an option to work offline as well.
2. An Axiell (Adlib) Museum system (at least model version 4.4 or a custom version) operating on an Adlib SQL Server database must have been installed on a server. Axiell Move Premium further requires *Missions* and *Missions items* data sources to have been added to the model application (see chapter 3.3), to allow managers to specify missions and their items. See the [Installation guide for Museum, Library and Archive](#) for more information about installing the Museum application and setting up the SQL database to be accessible by that application on work stations.

When setting up the user authentication by SQL Server or Active Directory, take into account that the database must be approachable by both users of the Museum application on work stations, as well as by users of Axiell Move on iPhones. For the second case you should know that the SQL Server will be approached by means of a web service (specific to Axiell Move) via a general IIS user account\*. (That user account must be part of the IIS\_IUSRS group for access to IIS itself.) In principle, that IIS user only needs to be able to read and write in the database, and in that regard it would be safest to only assign the `db_datareader` and `db_datawriter` *Database role memberships* to the *SQL Server database user* with the same name as the general IIS user. However, within your organisation there will also be co-workers who must be allowed to apply changes to the database structure via Axiell Designer. Depending on whether you use SQL Server authentication or Windows authentication, Designer will then use either the general IIS user name or the personal/AD group name of that user to access the database. In both cases it is very important that this user has the `db_owner` *Database role membership*: if that is not the case and the relevant user creates a new

index, then the table name won't get the required *dbo.* prefix. So you'll have to make a choice between SQL Server authentication, which means that you'll have to assign the *db\_owner* role to the general IIS user (with a possible security risk), and Windows authentication (with which the general IIS user only needs to have read and write access) to assign the *db\_owner* role to specific individual users and make absolutely sure that no-one else but them (to prevent the earlier mentioned problem) can make changes to the database structure via Axiell Designer.

\* See the *User authentication* chapter in the [installation guide for the Web API](#) for more information about the general IIS account.

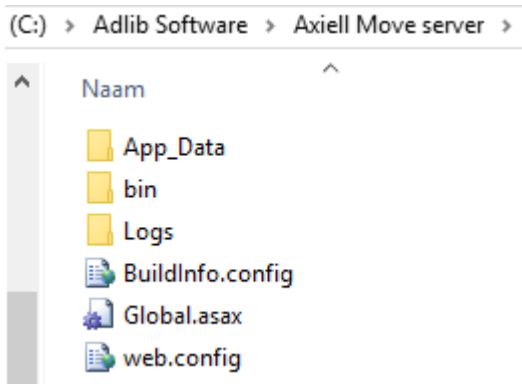
3. In step 4 you'll have to install an Axiell Move server which is going to handle the communication between the Axiell software on the iPhone and the SQL database. Preferably, this service must run in its own application pool. (An application pool is sort of a protected environment for services.) Now create the relevant .NET 4.0 application pool (underneath *Application pools* in IIS). Choose an appropriate name. (Note that 'Admove' is a different name for 'Axiell Move'.)



Then open the *Advanced settings* for the new application pool and enter the general IIS account name behind *Identity*.

See the installation guide for the Web API (click the link in the previous step) for more information about IIS.

4. For the installation, a number of files and folders have been made available to you. Place these items in a folder on the server and name that folder appropriately. After this, you'll still have to create an empty `\Logs` folder on the same level as `\App_Data` and `\bin` (see the screenshot below).



5. You need to make at least one so-called *application* in IIS, to secure your Axiell Move server and to create an Internet address. In IIS, under the web site in which you'd like to accommodate your Axiell Move server, add an *AxiellMoveServer* (or similarly named) *application*: the name will become part of the URL to the Move server, so choose a proper name. Further, select the application pool for it that you created earlier and the path to the physical folder on your system containing the Axiell Move files and subfolders (the main folder containing `web.config`, amongst others). See chapter 2.1 in the *Installation guide for the Web API* for detailed information about creating an IIS *application*.
6. Co-workers using the iPhone, must also have sufficient access rights (also see step 7) to the physical folder and/or share on which the Axiell Move server has been installed (at least read access). If that is already the case, you may continue with step 7. If it isn't the case (for example when normal network users have been excluded from access to as many folders as possible), you should either assign the required (NTFS) rights for all those users to that folder via Windows Explorer, or you must provide so-called physical path credentials for the *application*, with which the physical folder will then be approached instead of with the individual user accounts. In IIS, right-click the *application* name and in the pop-up menu select *Manage application > Advanced settings*. Put the cursor in the entry field behind *Physical Path*

*Credentials* and click the button with the ellipsis character (...) on the right. In the *Connect as* window, mark the *Specific user* option, click the *Set* button and enter the general IIS account name and password.

Most likely this account does already have the required rights to the physical folder, especially if that folder is located on the same server as the IIS account; on the other hand, if the physical folder is located on a different share or server, then you may still need to assign sufficient rights to the general IIS account on that share or server.

7. Via Windows Explorer, the application pool *identity* (the general IIS account) under which your application runs, must be given write access to the *AdmoveServerSetup.xml* configuration file in the *\App\_data* subfolder underneath the *\Axiell Move server* folder on the server: write access is required even though the file won't be changed by the Axiell Move server, otherwise an exception will follow. Write access for the general IIS account must also be assigned to the entire *\Logs* subfolder, so that the log files can be updated by the server. You change the access rights to a folder or file by right-clicking it, by opening the *Properties* from the pop-up menu and adding the access rights (*Modify, Read & execute, Read and Write*) for the relevant user account on the *Security* tab.
8. Finally, you'll have to adjust the *AdmoveServerSetup.xml* configuration file in the *\App\_data* subfolder on the server, to suit your particular environment. You can edit the file in a simple text editor like Windows Notepad or Wordpad. At the least you'll have to adjust the several path options in this configuration file, so that the *Adlib \data* and *\images* subfolders can be found by the server. You may also have to check whether all set field names and field tags match the definitions in your *Adlib* data dictionary (the *.inf* database structure files), but if Axiell ALM Netherlands has produced your 4.4/4.5 model application or customer specific application for Axiell Move, this check should be redundant. An example of this configuration file and a clarification of the available settings can be found in the paragraph below.



### 3.1.1. AdmoveServerSetup.xml

Below, an example of the *AdmoveServerSetup.xml* server configuration file (version 4.0):

```
<?xml version="1.0" encoding="utf-8"?>
<AdmoveSetup xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <DefaultLanguage>en-GB</DefaultLanguage>
  <Logging>
    <Enabled>true</Enabled>
    <LogFileName>Logs\admove.log</LogFileName>
  </Logging>

  <AuthenticationConfiguration>
    <type>Database</type>
    <databasePath>\\ourserver\Adlib SQL\data</databasePath>
    <database>borrower</database>
    <userIdField>borrower_name</userIdField>
    <passwordField>borrower_number</passwordField>
  </AuthenticationConfiguration>

  <DatabaseConfiguration database="photo">
    <databasePath>\\ourserver\Adlib SQL\data</databasePath>
    <database>photo</database>
  </DatabaseConfiguration>

  <ImageServerConfiguration name="admoveImages">
    <serverType>FileSystem</serverType>
    <path>\\ourserver\Adlib SQL\images</path>
    <cachePath>C:\TEMP\cache</cachePath>
    <folderMappingList>
      <!-- Museum branch 1: -->
      <folderMapping>
        <lowerLimit>1</lowerLimit>
        <upperLimit>200000000</upperLimit>
        <folder>\\ourserver\Adlib SQL\images\m1</folder>
        <mediaLowerLimit>1</mediaLowerLimit>
        <mediaUpperLimit>200000000</mediaUpperLimit>
      </folderMapping>
      <!-- Museum branch 2: -->
      <folderMapping>
        <lowerLimit>200000001</lowerLimit>
        <upperLimit>400000000</upperLimit>
        <folder>\\ourserver\Adlib SQL\images\m2</folder>
        <mediaLowerLimit>200000001</mediaLowerLimit>
        <mediaUpperLimit>400000000</mediaUpperLimit>
      </folderMapping>
      <!-- Museum branch 3: -->
      <folderMapping>
        <lowerLimit>400000001</lowerLimit>
        <upperLimit>600000000</upperLimit>
        <folder>\\ourserver\Adlib SQL\images\m2</folder>
```

```

    <mediaLowerLimit>400000001</mediaLowerLimit>
    <mediaUpperLimit>600000000</mediaUpperLimit>
  </folderMapping>
</folderMappingList>
<ImagePlugin type="Adlib.Imaging.Plugin.ImagePlugin,
  AdlibImagePlugin">
  <database>photo</database>
  <referenceField>priref</referenceField>
  <profileField>item.profile</profileField>
  <umidField>item.umid</umidField>
  <profileValue>7</profileValue>
  <wsAddress>http://www.ourmuseum.com/</wsAddress>
  <profiles>
    <profile id="7">image</profile>
  </profiles>
</ImagePlugin>
</ImageServerConfiguration>

<Object>
  <Path>\\ourserver\Adlib SQL\data\collect</Path>
  <StorageAdapl>..\adapls\mystoradapl</StorageAdapl>
  <Priref name="priref" property="Priref" tag="%0"> </Priref>
  <IdField>object_number</IdField>
  <ImageField>reproduction.reference</ImageField>
  <Details>
    <Field name="object_number">
      <Label lang="en-GB">Id</Label>
      <Label lang="nl-NL">Id</Label>
    </Field>
    <Field name="title">
      <Label lang="en-GB">Title</Label>
      <Label lang="nl-NL">Titel</Label>
    </Field>
    <Field name="creator">
      <Label lang="en-GB">Creator</Label>
      <Label lang="nl-NL">Vervaardiger</Label>
    </Field>
    <Field name="current_location">
      <Label lang="en-GB">Location</Label>
      <Label lang="nl-NL">Standplaats</Label>
    </Field>
  </Details>
  <MovementNotes>
    <Field name="current_location.suitability">
      <Label lang="en-GB">Location suitability</Label>
      <Label lang="nl-NL">Geschiktheid standplaats</Label>
    </Field>
    <Field name="current_location.authoriser">
      <Label lang="en-GB">Location authoriser</Label>
      <Label lang="nl-NL">Autorisator standplaats</Label>
    </Field>
    <Field name="current_location.notes">
      <Label lang="en-GB">Location notes</Label>
      <Label lang="nl-NL">Bijzonderheden standplaats</Label>
    </Field>
  </MovementNotes>
</Object>

```

```

</Field>
<!-- the next 4 fields are written to the history -->
<Field name="movement.method">
  <Label lang="en-GB">Movement method</Label>
  <Label lang="nl-NL">Methode verplaatsing</Label>
</Field>
<Field name="movement.reference">
  <Label lang="en-GB">Movement reference no.</Label>
  <Label lang="nl-NL">Referentienr. verplaatsing</Label>
</Field>
<Field name="movement.contact">
  <Label lang="en-GB">Movement contact</Label>
  <Label lang="nl-NL">Contactpersoon verplaatsing</Label>
</Field>
<Field name="movement.notes">
  <Label lang="en-GB">Movement notes</Label>
  <Label lang="nl-NL">Bijzonderheden verplaatsing</Label>
</Field>
</MovementNotes>
<CurrentLocation>
  <Field name="current_location" property="Id" tag="2A">
    <Label lang="en-GB">Current location</Label>
  </Field>
  <Field name="current_location.date" property="Date" tag="2C">
    <Label lang="en-GB">Date</Label>
  </Field>
  <Field name="current_location.time" property="Time" tag="2G">
    <Label lang="en-GB">Time</Label>
  </Field>
  <Field name="current_location.executor" property="User"
    tag="2R">
    <Label lang="en-GB">Executor</Label>
  </Field>
</CurrentLocation>
<CurrentLocationCopyFields>
  <Field name="current_location.suitability" property=
    "Suitability" tag="2E">
    <Label lang="en-GB">Suitability</Label>
  </Field>
  <Field name="current_location.authoriser" property=
    "Authorizer" tag="2F">
    <Label lang="en-GB">Authorizer</Label>
  </Field>
  <Field name="current_location.notes" property=
    "HistoryNotes" tag="2D">
    <Label lang="en-GB">Notes</Label>
  </Field>
</CurrentLocationCopyFields>
<LocationHistory>
  <Field name="location.history" property="Id" tag="ST">
    <Label lang="en-GB">Location</Label>
  </Field>
  <Field name="location.history.date.start" property=
    "Date" tag="SS">

```

```

    <Label lang="en-GB">Start date</Label>
  </Field>
  <Field name="location.history.time" property=
    "Time" tag="SH">
    <Label lang="en-GB">Start time</Label>
  </Field>
  <Field name="location.history.date.end" property=
    "EndDate" tag="SE">
    <Label lang="en-GB">Removal date</Label>
  </Field>
  <Field name="location.history.removal_time" property=
    "EndTime" tag="Sh">
    <Label lang="en-GB">Removal time</Label>
  </Field>
  <Field name="location.history.executor" property=
    "User" tag="2V">
    <Label lang="en-GB">Executor</Label>
  </Field>
</LocationHistory>
<LocationHistoryCopyFields>
  <Field name="location.history.suitability" property=
    "Suitability" tag="S3">
    <Label lang="en-GB">Suitability</Label>
  </Field>
  <Field name="location.history.authoriser" property=
    "Authorizer" tag="SP">
    <Label lang="en-GB">Authorizer</Label>
  </Field>
  <Field name="location.history.notes" property=
    "HistoryNotes" tag="LM">
    <Label lang="en-GB">Notes</Label>
  </Field>
</LocationHistoryCopyFields>
</Object>

<Package>
  <Path>\\ourserver\Adlib SQL\data\location</Path>
  <Preref name="record_number" property="Preref" tag="%0">
</Preref>
  <IdField>barcode</IdField>
  <SearchLinkField>name</SearchLinkField>
  <ItemTypeField>package_location</ItemTypeField>
  <ImageField>image</ImageField>
  <HierarchyLimit>50</HierarchyLimit>
  <Details>
    <Field name="barcode">
      <Label lang="en-GB">Id</Label>
      <Label lang="nl-NL">Id</Label>
    </Field>
    <Field name="package_location">
      <Label lang="en-GB">Item type</Label>
      <Label lang="nl-NL">Soort item</Label>
    </Field>
    <Field name="name">

```

```

        <Label lang="en-GB">Name</Label>
        <Label lang="nl-NL">Naam</Label>
    </Field>
    <Field name="description">
        <Label lang="en-GB">Description</Label>
        <Label lang="nl-NL">Beschrijving</Label>
    </Field>
    <Field name="package_type">
        <Label lang="en-GB">Package type</Label>
        <Label lang="nl-NL">Soort verpakking</Label>
    </Field>
    <Field name="current_location_package">
        <Label lang="en-GB">Location</Label>
        <Label lang="nl-NL">Standplaats</Label>
    </Field>
    <Field name="purchase_date">
        <Label lang="en-GB">Purchase date</Label>
        <Label lang="nl-NL">Aankoopdatum</Label>
    </Field>
</Details>
<MovementNotes>
    <Field name="current_location_package.notes">
        <Label lang="en-GB">Location notes</Label>
        <Label lang="nl-NL">Bijzonderheden standplaats</Label>
    </Field>
</MovementNotes>
<CurrentLocation>
    <Field name="current_location_package" property=
        "Id" tag="2Q">
        <Label lang="en-GB">Current location</Label>
    </Field>
    <Field name="current_location_package.date" property=
        "Date" tag="2C">
        <Label lang="en-GB">Date</Label>
    </Field>
    <Field name="current_location_package.time" property=
        "Time" tag="2G">
        <Label lang="en-GB">Time</Label>
    </Field>
    <Field name="current_location_package.executor" property=
        "User" tag="2R">
        <Label lang="en-GB">Executor</Label>
    </Field>
</CurrentLocation>
<CurrentLocationCopyFields>
    <Field name="current_location_package.notes" property=
        "HistoryNotes" tag="2D">
        <Label lang="en-GB">Notes</Label>
    </Field>
</CurrentLocationCopyFields>
<LocationHistory>
    <Field name="location_history.location_package" property=
        "Id" tag="2U">
        <Label lang="en-GB">Location</Label>

```

```

</Field>
<Field name="location_history.date.start" property=
  "Date" tag="SS">
  <Label lang="en-GB">Start date</Label>
</Field>
<Field name="location_history.time" property=
  "Time" tag="SH">
  <Label lang="en-GB">Start time</Label>
</Field>
<Field name="location_history.date.end" property=
  "EndDate" tag="SE">
  <Label lang="en-GB">Removal date</Label>
</Field>
<Field name="location_history.removal_time" property=
  "EndTime" tag="Sh">
  <Label lang="en-GB">Removal time</Label>
</Field>
<Field name="location_history.executor" property=
  "User" tag="2V">
  <Label lang="en-GB">Executor</Label>
</Field>
</LocationHistory>
<LocationHistoryCopyFields>
  <Field name="location_history.notes" property=
    "HistoryNotes" tag="LM">
    <Label lang="en-GB">Notes</Label>
  </Field>
</LocationHistoryCopyFields>
</Package>

<Location>
  <Path>\\ourserver\Adlib SQL\data\location</Path>
  <Prifref name="record_number" property="Prifref" tag="%0">
  </Prifref>
  <IdField>barcode</IdField>
  <SearchLinkField>name</SearchLinkField>
  <ItemTypeField>package_location</ItemTypeField>
  <ImageField />
  <HierarchyLimit>50</HierarchyLimit>
  <Details>
    <Field name="barcode">
      <Label lang="en-GB">Location</Label>
      <Label lang="nl-NL">Standplaats</Label>
    </Field>
    <Field name="package_location">
      <Label lang="en-GB">Item type</Label>
      <Label lang="nl-NL">Soort item</Label>
    </Field>
    <Field name="description">
      <Label lang="en-GB">Description</Label>
      <Label lang="nl-NL">Beschrijving</Label>
    </Field>
    <Field name="address">
      <Label lang="en-GB">Address</Label>

```

```

        <Label lang="nl-NL">Adres</Label>
    </Field>
    <Field name="address.postal_code">
        <Label lang="en-GB">Postal code</Label>
        <Label lang="nl-NL">Postcode</Label>
    </Field>
    <Field name="address.place">
        <Label lang="en-GB">Place</Label>
        <Label lang="nl-NL">Plaats</Label>
    </Field>
    <Field name="address.country">
        <Label lang="en-GB">Country</Label>
        <Label lang="nl-NL">Land</Label>
    </Field>
    </Details>
</Location>

<Missions>
    <Path>C:\ourmuseum\adlib\data+mission</Path>
    <!-- fields used in the mission database -->
    <UserField>user</UserField>
    <DescriptionField>de</DescriptionField>
    <DueDateField>du</DueDateField>
    <DueTimeField>tu</DueTimeField>
    <DeviceNameField>dn</DeviceNameField>
    <LoadedByUserField>lu</LoadedByUserField>
    <LoadedOnDeviceDateField>ld</LoadedOnDeviceDateField>
    <LoadedOnDeviceTimeField>lt</LoadedOnDeviceTimeField>
    <StatusChangedDateField>sd</StatusChangedDateField>
    <StatusChangedTimeField>sh</StatusChangedTimeField>
    <StatusField>ms</StatusField-->
    <NotesField>notes</NotesField>
</Missions>

<MissionActions>
    <Path>C:\ourmuseum\adlib\data+request</Path>
    <MissionIdField>mission_id</MissionIdField>
    <ActionTypeField>action_type</ActionTypeField>
    <ActionStatusField>request.status</ActionStatusField>
    <SourceItemField>object_number</SourceItemField>
    <SourcePackageField>package_source.id</SourcePackageField>
    <DestinationLocationField>location_destination.id
    </DestinationLocationField>
    <DestinationPackageField>package_destination.id
    </DestinationPackageField>
    <StatusChangedDateField>status_change.date
    </StatusChangedDateField>
    <StatusChangedTimeField>status_change.time
    </StatusChangedTimeField>
    <NotesField>request.details</NotesField>
</MissionActions>

<AdmoveUISettings>
    <AllowUnpackAll>true</AllowUnpackAll>

```

```

    <ConfirmAll>true</ConfirmAll>
    <EnableMissions>true</EnableMissions>
    <ShowPurposes>false</ShowPurposes>
    <ClearInfo>true</ClearInfo>
  </AdmoveUISettings>
</AdmoveSetup>

```

The differences with version 3.0 of this file are located in the following settings: <Missions>, <MissionActions> and <EnableMissions>.

## ■ Settings

The main sections in this XML file are:

- DefaultLanguage
- Logging
- AuthenticationConfiguration
- DatabaseConfiguration
- ImageServerConfiguration
- Object
- Package
- Location
- Purpose
- Missions
- MissionActions
- AdmoveUISettings

The AuthenticationConfiguration and the ImageServerConfiguration sections are based on their counterparts in the *adlibweb.xml* configuration file for the Adlib API, as described on: <http://api.adlibsoft.com/site/documentation/the-adlibweb-xml-file>.

## DefaultLanguage

```
<DefaultLanguage>en-GB</DefaultLanguage>
```

| XML node        | Explanation  |
|-----------------|--|
| DefaultLanguage | the language in which the Axiell Move interface texts must be displayed when the user has switched the i-Phone to a language for which no translations of interface texts are available. Further down this setup file you can see field labels being available in four languages by default: British English ( <i>en-GB</i> ), Dutch ( <i>nl-NL</i> ), French ( <i>fr-FR</i> ) and German ( <i>de-DE</i> ). In principle, the Axiell |



|  |  |
|--|--|
|  | <p>Move interface is now displayed in the current iPhone language (while in version 2.0 this language had to be set in the app itself), but by default that is only possible if that iPhone language is one of the four mentioned languages: if not, the default language set here will be used (which of course must be one of the available translations too).</p> <p>Note that new translations of interface texts can always be added to <i>AdmoveServerSetup.xml</i> and <i>Admove-Texts.xml</i>.</p> |
|--|--|

### The Logging section

```
<Logging>
  <Enabled>true</Enabled>
  <LogFileName>Logs\admove.log</LogFileName>
</Logging>
```

| XML node           | Explanation   |
|--------------------|---|
| <b>Enabled</b>     | switches storage of error reports in a logging file on ( <i>true</i> ) or off ( <i>false</i> ). When something goes wrong during your work with Axiell Move and an error is reported on the screen, that error message can automatically be saved in a logging file too, for later reference. It is recommended you switch this option on. In version 2.0, error messages were always stored in the logging file. |
| <b>LogFileName</b> | the name of the logging file in which error reports must be stored. The file name may be preceded by a relative path (relative to the root Axiell Move folder). The path you specify here, must already exist. In version 2.0 this logging file was called <i>errorLog.log</i> by default.  |

### The AuthenticationConfiguration section

```
<AuthenticationConfiguration>
  <type>Database</type>
  <databasePath>\\ourserver\Adlib SQL\data</databasePath>
  <database>borrower</database>
  <userIdField>borrower_name</userIdField>
  <passwordField>borrower_number</passwordField>
</AuthenticationConfiguration>
```

| XML node             | Explanation  |
|----------------------|--|
| <b>type</b>          | the authentication mode (the storage type for authentication details for access to the client software), with possible values: <code>Database</code> or <code>ActiveDirectory</code> . For the <code>Database</code> type, the extra settings below may apply; <code>ActiveDirectory</code> doesn't require extra settings because the configuration will be taken from Active Directory. In the case of <code>ActiveDirectory</code> , users will have to log on using the name of the local network domain followed by a backslash and their own user name, for example: <code>ourmuseum\erik</code> . This full name will also be written to the management details of records edited by Axiell Move, as the name of the person who applied the change. |
| <b>databasePath</b>  | the absolute (UNC) path to the folder containing the Adlib <code>.inf</code> files (database structure files).   |
| <b>database</b>      | the name of the database containing the authentication details. This name is taken from the file name of the <code>.inf</code> file for the relevant database (without the <code>.inf</code> extension).   |
| <b>userIdField</b>   | the name of the user name field in the <code>database</code> (the English field name).   |
| <b>passwordField</b> | the name of the password field in the <code>database</code> (the English field name).  |
| <b>groupsField</b>   | (optional, not in the example and currently not in use by the client software) the name of a groups/roles field (if present in the <code>database</code> ) which may contain one or more groups or roles to which a particular user belongs. These roles will be returned to the client software on the iPhone that could use this information to shield off certain functionality.  |
| <b>defaultGroups</b> | (optional, not in the example and currently not in use by the client software) in the child nodes: all the user groups or roles to which all logged-in users will belong. This element can be specified if no groups field exists in the <code>database</code> . The roles from the <code>group</code> nodes will be returned to the client software on the iPhone that could use this information to shield off certain functionality.  |
| <b>/group</b>        | (optional, not in the example and currently not in use by the client software) the name of a default user group or role, if <code>defaultGroups</code> are being used.   |

## The DatabaseConfiguration section

```
<DatabaseConfiguration database="photo">
  <databasePath>\\ourserver\Adlib SQL\data</databasePath>
  <database>photo</database>
</DatabaseConfiguration>
```

| XML node                      | Explanation  |
|-------------------------------|--|
| <b>Database-Configuration</b> | should contain the reference to an Adlib reproductions database, typically <code>photo</code> or <code>media</code> , which must be accessible to the Adlib Image plugin for a digital asset management system (DAMS). This section is only required if indeed a DAMS is being used to manage images. The <code>database</code> attribute uniquely identifies this database configuration. The alias doesn't need to be identical to the actual Adlib database name, but the <code>database</code> XML node for the Image plugin in the <code>ImageServerConfiguration</code> section must match it. |
| <b>databasePath</b>           | specifies the UNC path to the folder containing the <code>.inf</code> file of the Adlib database configured in this section, typically the <code>\data</code> subfolder.   |
| <b>database</b>               | the actual name of the Adlib reproductions database, which is the name of the relevant <code>.inf</code> file without its extension, typically <code>photo</code> .  |

## The ImageServerConfiguration section

```
<ImageServerConfiguration name="admoveImages">
  <serverType>FileSystem</serverType>
  <path>\\ourserver\Adlib SQL\images</path>
  <cachePath>C:\TEMP\cache</cachePath>
</ImageServerConfiguration>
```

or for example:

```
<ImageServerConfiguration name="admoveImages">
  <serverType>FileSystem</serverType>
  <ImagePlugin type="Adlib.Imaging.Plugin.ImagePlugin,
    AdlibImagePlugin">
    <database>photo</database>
    <referenceField>priref</referenceField>
    <profileField>item.profile</profileField>
    <umidField>item.umid</umidField>
    <profileValue>7</profileValue>
    <wsAddress>http://www.ourmuseum.com</wsAddress>
    <profiles>
      <profile id="7">image</profile>
```

```

    </profiles>
  </ImagePlugin>
</ImageServerConfiguration>

```

or for example:

```

<ImageServerConfiguration name="admoveImages">
  <serverType>FileSystem</serverType>
  <path>\\ourserver\Adlib SQL</path>
  <cachePath>C:\TEMP\cache</cachePath>
  <folderMappingList>
    <!-- Museum branch 1: -->
    <folderMapping>
      <lowerLimit>1</lowerLimit>
      <upperLimit>200000000</upperLimit>
      <folder>\\ourserver\Adlib SQL\images 1</folder>
      <mediaLowerLimit>1</mediaLowerLimit>
      <mediaUpperLimit>200000000</mediaUpperLimit>
    </folderMapping>
    <!-- Museum branch 2: -->
    <folderMapping>
      <lowerLimit>200000001</lowerLimit>
      <upperLimit>400000000</upperLimit>
      <folder>\\ourserver\Adlib SQL\images 2</folder>
      <mediaLowerLimit>200000001</mediaLowerLimit>
      <mediaUpperLimit>400000000</mediaUpperLimit>
    </folderMapping>
  </folderMappingList>
</ImageServerConfiguration>

```

| XML node                         | Explanation   |
|----------------------------------|---|
| <b>ImageServer-Configuration</b> | <p>the image server configuration. This server is used to retrieve object, location and package images. Currently, only a single image server configuration is supported. The <code>name</code> attribute provides a name for this image server configuration. You can leave it as is, or change it. There can be more than one <code>ImageServerConfiguration</code> section.</p> <p>You have to choose between two configuration types:</p> <ul style="list-style-type: none"> <li>You use the first one when images must be retrieved from the file system or from SQL Server through the Axiell Move server directly. In this case you only need to specify the XML elements <code>serverType</code>, <code>path</code> and <code>cachePath</code> (and any optional elements in between). This type of configuration is applied most often.</li> <li>The second must be applied when you are using a third-party DAMS (digital asset management system)</li> </ul> |

|                   |  |
|-------------------|--|
|                   | <p>to manage images. Image files won't be accessible through the file system. Axiell ALM Netherlands provides a so-called image plug-in (on the server) per DAMS type (each supplier has its own type), which takes care of the communication between the Axiell Move server and the DAMS to stream images to the client on the iPhone. In that case you specify the <code>ImagePlugin</code> XML elements plus the <code>serverType</code>: <code>FileSystem</code> (the latter despite the fact that the images are not located in the file system). So <code>Path</code> and <code>cachePath</code> can be left out because they aren't being used: if you do specify them, they will be ignored. Note that the example above is not generally applicable to all types of DAMS: each type has its own settings. The type from the example applies to a Cambridge Imaging Systems DAMS, although even then there might be differences between implementations.</p> |
| <b>serverType</b> | <p>(<code>MsSql</code>   <code>Oracle</code>   <code>FileSystem</code>) specifies the storage system (SQL, Oracle or the file system), from which images must be retrieved. If you use an Adlib plugin for a DAMS, you must set this option to <code>FileSystem</code>.</p>  |
| <b>server</b>     | <p>(optional, not for the <code>FileSystem</code> server type, not for a DAMS) provides the name of the server on which your Adlib SQL Server database can be found.</p>   |
| <b>username</b>   | <p>(optional, not for the <code>FileSystem</code> server type, not for a DAMS) specifies the name with which, in combination with <code>password</code>, the images in a SQL database can be accessed via the Axiell Move server. It concerns the same <code>username</code> and <code>password</code> which have been set in SQL Server, for making a connection to the database.</p>   |
| <b>password</b>   | <p>(optional, not for the <code>FileSystem</code> server type, not for a DAMS) specifies the password with which, in combination with <code>username</code>, the images in a SQL database can be accessed via the Axiell Move server. It concerns the same <code>username</code> and <code>password</code> which have been set in SQL Server or Oracle, for making a connection to the database.</p>   |
| <b>database</b>   | <p>(optional, not for the <code>FileSystem</code> server type, not for a DAMS) specifies the Adlib SQL Server database from which images must be retrieved, if they are not stored in the file system.</p>   |
| <b>path</b>       | <p>specifies the (UNC) path to the main images folder (this applies only if your images have been stored in the file</p>   |

|                          |   |
|--------------------------|---|
|                          | system, not in an Adlib SQL database nor in a DAMS); a local file system path is possible as well, but only if this folder is located on the same server as the Axiell Move server, or has been mapped as drive on this server.   |
| <b>cachePath</b>         | specifies the (UNC) path to the folder in which all images resized by the server will be stored* (not for a DAMS); a local file system path is possible as well, but only if this cache folder is located on the same server, or has been mapped as drive on this server.<br>* The Axiell Move server (like the Web API, <i>wwwopac.ashx</i> ) does not resize the images to be retrieved for every request, but reduces or enlarges the image only when first requested, and then saves the result in a cache folder so that for a future identical request a ready-made resized image can be retrieved. |
| <b>folderMappingList</b> | (optional) for uploading images pertaining to an object in an enterprise configuration where each branch must have its own images folder and its own dataset within the Media database, add a single <code>folderMappingList</code> containing as many <code>folderMapping</code> sections as there are branches.   |
| <b>folderMapping</b>     | a single <code>folderMapping</code> for a branch (partial collection) in an enterprise environment must contain the record number range to check the processed object record for, the target folder for uploaded images and the record number range of the Media dataset for this particular branch.  |
| <b>lowerLimit</b>        | the current <code>folderMapping</code> will be applied if the object record for which the user is currently uploading an image, has a record number higher than or equal to the number you specify here.  |
| <b>upperLimit</b>        | the current <code>folderMapping</code> will be applied if the object record for which the user is currently uploading an image, has a record number lower than or equal to the number you specify here.   |
| <b>folder</b>            | specifies the (UNC) path to the images sub folder specific to this branch (this applies only if your images have been stored in the file system, not in an Adlib SQL database nor in a DAMS); a local file system path is possible as well, but only if this folder is located on the same server as the Axiell Move server, or has been mapped as drive on this server. A relative path will automatically be combined with the path from the <code>path</code> setting.   |
| <b>mediaLowerLimit</b>   | for each uploaded image, a reproduction record will automatically be created in the Media database. Since each branch has its own dataset (which is determined by a   |

|                        |  |
|------------------------|--|
|                        | record number range), specify here the lowest possible record number in the dataset for the current branch, so that the record can be created in that dataset.   |
| <b>mediaUpperLimit</b> | for each uploaded image, a reproduction record will automatically be created in the Media database. Since each branch has its own dataset (which is determined by a record number range), specify here the highest possible record number in the dataset for the current branch, so that the record can be created in that dataset.  |
| <b>ImagePlugin</b>     | this section contains DAMS-specific settings for an Adlib image plugin to allow streaming of images from the DAMS to the Axiell Move client.<br>The <code>type</code> attribute must always be: <code>Adlib.Imaging.Plugin.ImagePlugin</code> , <code>AdlibImagePlugin</code>  |
| <b>database</b>        | alias of the Adlib reproductions database, as defined in the <code>DatabaseConfiguration</code> section. For a DAMS, this Adlib database links metadata (from a record) indirectly to some version of the media file (aka media type rendition) it pertains to, stored in the DAMS, by means of the <code>umidField</code> and (in this particular case) the <code>profileField</code> .   |
| <b>referenceField</b>  | name of the field in the reproductions database which is referenced from within the Adlib (museum) object records database, to link to the image record. <code>preref</code> is the record number field.   |
| <b>profileField</b>    | name of the repeatable field in the reproductions database which contains one or more references to rendition profiles** in which this image <i>could</i> be streamed. Each <code>profileField</code> field occurrence must have an accompanying <code>umidField</code> field occurrence.  |
| <b>umidField</b>       | name of the repeatable field in the reproductions database, which contains one or more unique material identifier(s) for the media mapping from the <code>preref</code> / <code>profile</code> combination in the reproduction record to the <code>umid</code> / <code>profile</code> combination in the DAMS: so a single reproduction record may contain several <code>umid</code> / <code>profile</code> combinations in which the media type can be rendered by the DAMS. Each <code>umidField</code> field occurrence must have an accompanying <code>profileField</code> field occurrence. |
| <b>profileValue</b>    | specifies the rendition profile identifier which must be used for the stream rendition of all images. A requested image can only be streamed with this profile if one of the occurrences of the <code>profileField</code> in the reproductions database actually references this profile.  |
| <b>wsAddress</b>       | the DAMS web service address (URL).  |

|                 |   |
|-----------------|---|
| <b>profiles</b> | this section defines the available profiles for the plugin.   |
| <b>profile</b>  | <p>defines a profile identifier for a media type, like an image, video, thumbnail, audio, etc.</p> <p><b>**</b> A rendition profile is one of possibly multiple versions in which a media file can be streamed: in this context, a version of an image can be understood as its resolution and/or file type, for example.</p> <p>The <code>id</code> attribute must contain an identifier matching one of the profiles defined in the DAMS.</p> |

Note that of images always a small version will be retrieved, so that any large image files won't lead to performance issues.

Also see: <http://api.adlibsoft.com/site/documentation/the-adlibweb-xml-file> for more information about the image server configuration settings. You don't need to set up a separate Web API (`wwwopac.ashx`) image server though.

## The Object, Package and Location sections

```

<Object>
  <Path\\ourserver\Adlib SQL\data\collect</Path>
  <StorageAdapl>..\adapls\mystoradapl</StorageAdapl>
  <Priref name="priref" property="Priref" tag="%0"> </Priref>
  <IdField>object_number</IdField>
<!--<SearchLinkField>name</SearchLinkField> -->
  <ImageField>reproduction.reference</ImageField>
  <HierarchyLimit>100</HierarchyLimit>
  <Details>
    <Field name="object_number">
      <Label lang="en-GB">Id</Label>
      <Label lang="nl-NL">Id</Label>
    </Field>
    ...
  </Details>
  <MovementNotes>
    <Field name="current_location.suitability">
      <Label lang="en-GB">Location suitability</Label>
      <Label lang="nl-NL">Geschiktheid standplaats</Label>
    </Field>
    ...
  </MovementNotes>
  <CurrentLocation>
    <Field name="current_location" property="Id" tag="2A">
      <Label lang="en-GB">Current location</Label>
    </Field>
    <Field name="current_location.date" property="Date" tag="2C">
      <Label lang="en-GB">Date</Label>
    </Field>
  </CurrentLocation>

```



```

...
</CurrentLocation>
<CurrentLocationCopyFields>
  <Field name="current_location.suitability" property=
    "Suitability" tag="2E">
    <Label lang="en-GB">Suitability</Label>
  </Field>
...
</CurrentLocationCopyFields>
<LocationHistory>
  <Field name="location.history" property="Id" tag="ST">
    <Label lang="en-GB">Location</Label>
  </Field>
...
</LocationHistory>
<LocationHistoryCopyFields>
  <Field name="location.history.suitability" property=
    "Suitability" tag="S3">
    <Label lang="en-GB">Suitability</Label>
  </Field>
...
</LocationHistoryCopyFields>
<HomeLocation>
  <Field name="location.default.name" property="Id" tag="NL">
    <Label lang="en-GB">Home location</Label>
  </Field>
...
</HomeLocation>
</Object>

```

| XML node                         | Explanation   |
|----------------------------------|---|
| <b>Object, Package, Location</b> | the settings below must be made in separate sections for objects, packages and locations.   |
| <b>Path</b>                      | <p>the absolute (UNC) path to the objects (<i>collect</i>), packages (<i>location</i>) or locations (<i>location</i>) database table <i>.inf</i> file. No need to specify the <i>.inf</i> extension.</p> <p>You can also specify a dataset behind the database table name: this is often required in multi-tenancy setups where each branch or museum has its own record number range (dataset) within the all-encompassing database table. Separate the database table name from the dataset name with <code>&amp;gt;</code> (the encoded <code>&gt;</code> character), like so for example:</p> <pre>&lt;Path&gt;\\ourserver\Adlib SQL\data\collect&amp;gt;fullcatalogue&lt;/Path&gt;</pre> |
| <b>StorageAdapl</b>              | the reference to an ADAPL <i>.bin</i> file (without extension) which must be executed just before storage of a record that has been adjusted by Axiell Move in the  |

|                       |  |
|-----------------------|--|
|                       | <p>current database, instead of the default storage adapl. This setting is optional: by default, the storage adapl set up for the database (via Designer) will be executed when records have been edited by Axiell Move (different from version 2.0 by the way, which never executed the default storage adapl). If you still want a newly written storage adapl specific to transactions to be executed after Axiell Move transactions, then specify here the relative path to that adapl (relative to the folder in which the <i>.inf</i> file of the current database is located).</p> <p>The default storage adapl will add management details (edit date, time and user name) to adjusted records, but it may also perform a lot of other checks and actions which are not required for changes made by Axiell Move. Moreover, you may like a note to be added to the management details indicating that the relevant change was made by Axiell Move. That's why sometimes it may be desirable to create a separate storage adapl for object, package and location records (usually based on the standard storage adapls for the relevant databases). The choice is yours.</p> <p>Further note that the execution of the adapl specified here, will be handled by software which currently does not really recognize <i>all</i> existing ADAPL functions: so it's possible that an existing storage adapl in Adlib works perfectly fine, while executed by the Axiell Move server it would suddenly generate an error message. In that case it's likely that one of the used ADAPL functions has not been implemented yet. Please report such issues to our helpdesk so we can fix the problem as soon as possible.</p> |
| <p><b>Priref</b></p>  | <p>has no value, only the following attributes:</p> <ul style="list-style-type: none"> <li>- <i>name</i>, the record number (English) field name (<i>priref of record_number</i>) in this database;</li> <li>- <i>property</i>, the property name (<i>Priref</i>) in the Axiell Move server which maps to the actual <i>priref</i> value;</li> <li>- <i>tag</i>, the Adlib field tag (<i>%0</i>) of the record number field in this database.</li> </ul>   |
| <p><b>IdField</b></p> | <p>the name of the field which identifies records in this database (<i>object_number, bar_code</i>). The index on this field will then be used to be able to find the relevant objects, packages or locations. If you specify an <i>IdField</i> you won't have to provide <i>IdFields</i>: always use one of both options. See the <code>&lt;IdFields&gt;</code> element if you'd like more than one field to be</p>   |

|                               |  |
|-------------------------------|--|
|                               | <p>searched.</p> <p>Note that the collect <code>bar_code</code> field is called <code>barcode</code> in the location database.</p>   |
| <p><b>IdFields</b></p>        | <p>From Axiell Move server version 2.0.150127.4, this element can be used instead of <code>IdField</code> if you want more than one field to be searched for objects, packages or locations. Within this element you must then list one or more fields as desired, for example as follows:</p> <pre data-bbox="445 424 896 544">&lt;IdFields&gt;   &lt;Field&gt;bar_code&lt;/Field&gt;   &lt;Field&gt;object_number&lt;/Field&gt;   &lt;Field&gt;alternative_number&lt;/Field&gt; &lt;/IdFields&gt;</pre> <p>The indexes of all specified fields will be searched, firstly of the field listed on top (<code>bar_code</code> in this example), then the second specified field, etc. This functionality can prove to be essential for the Enterprise model of the Adlib software in combination with a single Axiell Move server, in which for some partial collections no barcode or object number may be present in records.</p> <p>Note that the collect <code>bar_code</code> field is called <code>barcode</code> in the location database.</p>   |
| <p><b>SearchLinkField</b></p> | <p>the name of the field in the <code>location</code> database to which object records link in order to obtain location or package record data. The setting is mandatory (twice) and must appear in the <code>&lt;Package&gt;</code> and <code>&lt;Location&gt;</code> sections of <code>AdmoveServerSetup.xml</code> only. Which field name to enter here, depends on your database structure. Typically, the <code>current_location</code> field in <code>collect</code> links to either the <code>barcode</code> or <code>name</code> field in <code>location</code>. Whatever field is set up in the relevant link definition, you must set in the <code>SearchLinkField</code> option.</p> <p>The difference between having the <code>barcode</code> and the <code>name</code> field as lookup field in the link definition, is that the <code>name</code> field allows the the Museum application user to enter and validate location and package <b>names</b> during object data registry, instead of having to use barcodes which may have no meaning. To de Movement module on the iPhone, the used linked-to field makes no difference: it always looks up data using the scanned barcode.</p> |

|                         |  |
|-------------------------|--|
| <b>ImageField</b>       | the name of the field which identifies image records in this database ( <code>reproduction.reference</code> in <code>collect</code> , <code>image</code> in <code>location</code> )  |
| <b>HierarchyLimit</b>   | (optional, the implicit default value is 50) the maximum amount of items that will be retrieved and listed (in one long list) when the user asks for the contents of a location or container in the Move app. You can include this setting in the <code>Package</code> and <code>Location</code> sections only, and set it to a higher value if desired. Do note that setting it to a higher value might be bad for performance when such lists are retrieved in the app.  |
| <b>ItemTypeField</b>    | (not applicable to the <code>Object</code> section) the name of the field ( <code>package_location</code> ) in the locations and packages database ( <code>location</code> ), which contains the type of location: a record in the <code>location</code> database can describe a location or a package.  |
| <b>ContentItemField</b> | (not applicable to the <code>Location</code> section) optional, the name of the object or package field of which the value must be displayed if in Axiell Move the user clicks the ⓘ icon for a selected package or location to request information about the objects the package contains or the packages that can be found on the location. In the <code>Object</code> section you register the object field which must be displayed if the user requests the contents of a package: you would normally choose between <code>object_number</code> and <code>bar_code</code> . In the <code>Package</code> section on the other hand, you register the package field which must be displayed if the user requests the contents of a location: you would then probably choose between the <code>name</code> field (package names) or <code>barcode</code> (the barcode of these packages). |
| <b>ShowHierarchy</b>    | (not applicable to the <code>Object</code> section) optional, suppresses the retrieval of items in a package or from a location if set to <code>false</code> . By default, all items will be retrieved indeed, but if a location contains hundreds of objects, this might have a negative impact on the performance of Axiell Move: in that case you can switch the retrieval of details about those items off to improve the performance. As a consequence, the user won't be able to request this information in the client any more.<br>The option is available from version 2.0.15288.1.   |

|  |  |
|--|--|
| <p><b>AllowMoveToCurrentLocation</b></p> | <p>(not applicable to the <code>Location</code> section) optional, allows (when set to <code>true</code>) a package or object to be 'moved' to the location where the item already resides. You can use this option by way of inventory or location check: by moving an item to the current location, the location history of the item will be updated for future reference of the last date and time the location of the item was checked. By default (<code>false</code>), moving an item to the current location will result in a message stating that you're trying to move an item to its present location, after which that transaction won't be executed.</p>   |
| <p><b>Details</b></p>                    | <p>(in the child nodes) the list of all the fields that will be shown on the record detail screens in Axiell Move on the iPhone. Each <code>Field</code> node (containing an English field name in the <code>name</code> attribute) may have several multilingual <code>Label</code> elements (containing user interface translations of the field label). All fields must already be present in the database structure.</p>   |
| <p><b>MovementNotes</b></p>              | <p>the list of all the fields that can be edited when notes are attached to a transaction (for packing, unpacking and moving). After a move of an object, the <code>movement</code> fields in this section will be registered with the previous location in the object record, while the other fields will be registered with the new current location. All fields must already be present in the database structure.</p>  |
| <p><b>CurrentLocation</b></p>            | <p>the list of all the basic current location fields (only two of those shown in the example above) that must be updated in the record in the current database when an object or package is placed at another location. Each <code>Field</code> node (containing an English field name in the <code>name</code> attribute) may have several multilingual <code>Label</code> elements (containing user interface translations of the field label). At the moment, a current location change includes a location name, date and time of the placement and the user that performed the placement at the new location. All fields must already be present in the database structure. The field <code>names</code> and <code>tags</code> for the <code>CurrentLocation</code> and <code>LocationHistory</code> settings can in principle be changed if other fields apply, but the <code>property</code> attribute values (<code>Id</code>, <code>Date</code>, <code>Time</code>, <code>User</code>, <code>Device</code>) must remain the same (these are property names in</p> |

|  |  |
|--|--|
|  | <p>the Axiell Move server, mapping to a field value). A <code>Field</code> specification with the <code>property</code> attribute <code>Device</code> is optional: in the target field which you specify here, the so-called device id (the identification of the iPhone as determined in the settings for the Axiell Move client) will be stored with each transaction. It is possible that there is no existing target field to be used yet: if this is the case, you'll have to add such a field to the relevant database and to the screen with the other details of the current location, using Adlib Designer.</p>   |
| <p><b>CurrentLocation-CopyFields</b></p> | <p>the list of some extra current location fields that must be updated in the object record when an object is placed at another location. Each <code>Field</code> node (containing an English field name in the <code>name</code> attribute) may contain several multilingual <code>Label</code> elements (containing interface translations of the field label). At the moment, a current location change includes the authoriser, location suitability and location notes as the extra fields. All fields must already be present in the database structure.</p> <p>The field names and tags for the <code>CurrentLocationCopyFields</code> and <code>LocationHistoryCopyFields</code> settings can in principle be changed if other fields apply, but the <code>property</code> attribute values must remain the same (these are property names in the Axiell Move server, mapping to a field value).</p>   |
| <p><b>LocationHistory</b></p>            | <p>the list of all the basic location history fields that must be updated in the record in the current database when an object or package is placed at another location. Each <code>Field</code> node (containing an English field name in the <code>name</code> attribute) may contain several multilingual <code>Label</code> elements (containing user interface translations of the field label).</p> <p>At the moment, a location history change includes a location name, begin and end date and time of the placement period in this location and the user that performed the placement at the old location. All fields must already be present in the database structure.</p> <p>The field names and tags for the <code>CurrentLocation</code> and <code>LocationHistory</code> settings can in principle be changed if other fields apply, but the <code>property</code> attribute values (<code>Id</code>, <code>Date</code>, <code>Time</code>, <code>User</code>, <code>Device</code>) must remain the same (these are property names in the Axiell Move server, mapping to a field value). Note that the <code>CurrentLocation</code> settings map to the</p> |

|  |  |
|--|--|
|  | <p>LocationHistory settings.</p> <p>A Field specification with the property attribute Device is optional: in the target field which you specify here, the so-called device id (the identification of the iPhone as determined in the settings for the Axiell Move client) will be stored with each transaction. It is possible that there is no existing target field to be used yet: if this is the case, you'll have to add such a field to the relevant database and to the screen with the other details of the location history, using Adlib Designer.</p>  |
| <p><b>LocationHistory-CopyFields</b></p> | <p>the list of some extra location history fields that must be updated in the object record when an object is placed at another location. Each Field node (containing an English field name in the name attribute) may contain several multilingual Label elements (containing user interface translations of the field label).</p> <p>At the moment, a location history change includes the authoriser, location suitability and location notes as the extra fields. All fields must already be present in the database structure.</p> <p>The field names and tags for the CurrentLocationCopyFields and LocationHistoryCopyFields settings can in principle be changed if other fields apply, but the property attribute values must remain the same (these are property names in the Axiell Move server, mapping to a field value). Note that the CurrentLocationCopyFields settings map to the LocationHistoryCopyFields settings.</p> |
| <p><b>HomeLocation</b></p>               | <p>the list of all the basic normal location fields (typically only one though) that must be updated in the record in the current database when an object or container is placed at another normal location via the <i>Set Normal</i> option. Each Field node (containing an English field name in the name attribute) may have several multilingual Label elements (containing user interface translations of the field label).</p> <p>The field names and tags for the HomeLocation setting can in principle be changed if other fields apply, but the property attribute values (Id, Date, Time, User, Device) must remain the same (these are property names in the Axiell Move server, mapping to a field value).</p>   |

## The Purpose section

```
<Purpose>
  <purposes fieldName="location.purpose">
    <purpose>Temporary Storage</purpose>
    <purpose>Photography</purpose>
    <purpose>Permanent Storage</purpose>
    <purpose>Loan</purpose>
    <purpose>Exhibition</purpose>
    <purpose>Conservation</purpose>
  </purposes>
</Purpose>
```

| XML node        | Explanation   |
|-----------------|---|
| <b>Purpose</b>  | this section contains the Mimsy-specific option to offer a set of possible reasons (to be selected from a drop-down list with the transaction in the Move client) why the user performs a certain transaction. The last selected purpose will be preselected for the next transaction. Adlib users can ignore this section (leave it or remove it), because it hasn't been implemented in Adlib applications or core software. For Adlib, the <code>&lt;ShowPurposes&gt;</code> option underneath <code>&lt;Ad-moveUISettings&gt;</code> should always be set to <code>false</code> . |
| <b>purposes</b> | specifies the field name in which the purpose selected by the user must be stored, in the <code>fieldName</code> attribute, and lists the drop-down options in the <code>&lt;purpose&gt;</code> nodes underneath it.  |
| <b>purpose</b>  | specifies a single option in the drop-down list. You may include as many purposes as you like.  |

## The Missions and MissionActions sections

```
<Missions>
  <Path>C:\ourmuseum\adlib\data+mission</Path>
  <!-- fields used in the mission database -->
  <UserField>user</UserField>
  <DescriptionField>de</DescriptionField>
  <DueDateField>du</DueDateField>
  <DueTimeField>tu</DueTimeField>
  <DeviceNameField>dn</DeviceNameField>
  <LoadedByUserField>lu</LoadedByUserField>
  <LoadedOnDeviceDateField>ld</LoadedOnDeviceDateField>
  <LoadedOnDeviceTimeField>lt</LoadedOnDeviceTimeField>
  <StatusChangedDateField>sd</StatusChangedDateField>
  <StatusChangedTimeField>sh</StatusChangedTimeField>
  <StatusField>ms</StatusField>-->
  <NotesField>notes</NotesField>
```



```

</Missions>
<MissionActions>
  <Path>C:\ourmuseum\adlib\data+request</Path>
  <MissionIdField>mission_id</MissionIdField>
  <ActionTypeField>action_type</ActionTypeField>
  <ActionStatusField>request.status</ActionStatusField>
  <SourceItemField>object_number</SourceItemField>
  <SourcePackageField>package_source.id</SourcePackageField>
  <DestinationLocationField>location_destination.id
  </DestinationLocationField>
  <DestinationPackageField>package_destination.id
  </DestinationPackageField>
  <StatusChangedDateField>status_change.date
  </StatusChangedDateField>
  <StatusChangedTimeField>status_change.time
  </StatusChangedTimeField>
  <NotesField>request.details</NotesField>
</MissionActions>

```

| XML node                | Explanation  |
|-------------------------|--|
| <b>Path</b>             | specifies the path to the database structure file (.inf) containing either the mission or mission items field definitions. These are the <i>mission</i> and <i>request</i> database structures respectively, which come with your Axiell Move Premium adjusted Museum application. |
| <b>&lt;...Field&gt;</b> | these nodes list all fields/tags used by the missions functionality.   |

## The AdmoveUISettings section

```

<AdmoveUISettings>
  <AllowUnpackAll>true</AllowUnpackAll>
  <ConfirmAll>true</ConfirmAll>
  <ShowDomainInput>true</ShowDomainInput>
  <ShowPurposes>>false</ShowPurposes>
  <ClearInfo>true</ClearInfo>
</AdmoveUISettings>

```

| XML node                | Explanation  |
|-------------------------|--|
| <b>AdmoveUISettings</b> | this section contains options for the user interface of all Axiell Move clients together.  |
| <b>AllowUnpackAll</b>   | (true   false) use this setting to decide whether users are allowed to unpack all smaller packages or bare objects from an overpack at once and assign the same location to all those smaller items. If not (false), then the relevant option in the user interface won't be active. |

|                        |   |
|------------------------|---|
| <b>ConfirmAll</b>      | ( <code>true</code>   <code>false</code> ) it is possible to have an on-screen confirmation after scanning an object, packaging or location for a transaction: this confirmation comes in the shape of the scanned ID and some retrieved data from the relevant record. Set this option to <code>true</code> if you want such confirmations to be displayed to all users. On the other hand, if you think it slows down the work or doesn't add any value, then choose <code>false</code> . |
| <b>EnableMissions</b>  | ( <code>true</code>   <code>false</code> ) allows you to enable the missions functionality in Axiell Move Premium: if this option hasn't been set to <code>true</code> , then the <i>Missions</i> and <i>Last mission</i> icons won't be present in the main menu.  |
| <b>ShowDomainInput</b> | ( <code>true</code>   <code>false</code> ) only for the old Museum Tracker Windows app this option has the function to display an extra field on the login screen, to be able to enter the network domain of the user separately (since the keyboard of that device didn't contain a slash which would have been required to be able to enter the domain and user name in a single field).  |
| <b>ShowPurposes</b>    | ( <code>true</code>   <code>false</code> ) has no function for Adlib applications and for Adlib this option must therefore always be set to <code>false</code> .  |
| <b>ClearInfo</b>       | ( <code>true</code>   <code>false</code> ) makes sure (for <code>true</code> ) that after a transaction the displayed information (about a successfully packed object for example) won't be displayed again if you close this screen and then reopen it for a similar transaction because you'd like to pack another object.  |

## 3.2. Registering login details

In the `AuthenticationConfiguration` section of the `AdmoveServerSetup.xml` configuration file in the `\App_data` subfolder underneath the `\Axiell Move Server` folder on the server, you have set the authentication mode (the storage type for authentication details for access to the client software) to either `Database` or `ActiveDirectory`:

- `ActiveDirectory`: for logging on to Axiell Move the (Windows) user names and passwords, with which users normally log on to their computer in the local network, will be used. This means that user names and passwords don't need to be registered anywhere else, yet cannot be anything else than the Active Directory user names and passwords either.

- For the Database authentication type, the extra settings in the AuthenticationConfiguration section of the configuration file apply. In the example below, the borrower\_name and borrower\_number fields in the borrower database have been set, but you can use other fields in a different database as well (a separate database would indeed be a better solution).

```
<AuthenticationConfiguration>
  <type>Database</type>
  <databasePath>\\ourserver\Adlib SQL\data</databasePath>
  <database>borrower</database>
  <userIdField>borrower_name</userIdField>
  <passwordField>borrower_number</passwordField>
</AuthenticationConfiguration>
```

Axiell Designer is the proper tool to get information about field names and database names or for creating a new database table for authentication details. See the [Designer Help](#) for all documentation. For the example of the settings above, your user names and passwords must be registered as follows:

1. On your computer, start the Library Loans Management application in Adlib or Axiell Collections and open the *Borrowers* data source (this is the earlier set up `borrower` database).
2. Open a new record. The user name for login is found in the read-only *Name* field (`borrower_name`). This name is put together from the editable *First name* and *Surname* fields. You can leave the *First name* field empty, to obtain an easier user name. For example, enter `user1` in the *Surname* field.

The password for login is found in the *User number* field (`borrower_number`). Enter the desired password in this field. Save the record. From now on this user name and password can be used to log on to Axiell Move on the iPhone.

Note that all existing *User numbers* and accompanying *Names* in this database can be used to log in (which is not optimal and the reason why a separate authentication database would be better). However, do not just change or remove existing data here, because it is used for other purposes as well!

In principle you could use a single user name and password for all actual iPhone/scanner users, but then you will never be able to tell who executed certain transactions. So a different user name and password for each iPhone/scanner user is recommended.

| Record details   |                      |
|------------------|----------------------|
| Personal details |                      |
| Name and number  |                      |
| User number      | MyMovePW2748#        |
| Site             | <a href="#">main</a> |
| Name             | Lange, Erik          |
| Contact person   |                      |
| Term of address  |                      |
| Title            |                      |
| Surname          | Lange                |
| First name       | Erik                 |

---

### 3.3. Setting up the missions functionality

Axiell Move Premium requires *Missions* and *Mission items* data sources and their associated databases to have been added to your application, to allow managers to specify missions and their items. Even though there's no model application version of the required files yet, we can offer you an (unpolished) development version that should help you on your way, but do note that any changes you make to these files will be overwritten if in the future you decide to install a model version of these files.

A zip file with the required files can be requested through our helpdesk if you have an Axiell Move Premium license only.

Before you start doing anything with it, please create a backup of your existing `\Adlib software` folder and your SQL database. It's good to have the old situation to revert to should anything go wrong in the following steps.

You'll find the zip file contains the following files:

- *mission.inf* – the database definition for mission data: copy this file to your Axiell (Adlib) Museum application `\data` folder.
- *request.inf* – the database definition for mission items data: if your `\data` folder already contains a *request.inf* it means (other) workflow functionality has been implemented in your application and you should keep on using the existing *request.inf* in your application. (The *request.inf* can be used for both mission item data as well as for workflow activity data.) So copy this file to your Museum application `\data` folder only if it isn't present in that folder yet.
- *adlib.pbk* – a stripped application definition which contains *Mission* and *Mission items* data sources only. Copy this file to some temporary folder in your main Museum application folder and **DO NOT** overwrite any existing *adlib.pbk* file. The stripped version will serve as a source file for further work only.
- a `\screens\mission` subfolder containing 9 screen files. Copy the `\mission` subfolder to your Museum application `\screens` folder. So the mission screen files will remain in their own subfolder and won't mingle with the other screen files. The included screen files are:
  - *missionitem\_details.fmt* – mission item record detail screen;

- *mission\_details.fmt* – mission record detail screen;
- *zm\_missionitem\_details.fmt* – zoom screen for mission items;
- *zm\_mission\_details.fmt* – zoom screen for mission;
- *lnk\_missionitem.fmt* – link screen for mission items;
- *lnk\_mission.fmt* – link screen for mission;
- *br\_missionitem.fmt* – (brief) result set screen for mission items, also used as the connection screen for the *Add mission item(s) to mission* function in the *Missions* view that you open with the *Axiell Move* option in the main menu;
- *br\_mission.fmt* – (brief) result set screen for missions;
- *manage.fmt* – management details screen.

Further steps to take:

1. Open the *Application browser* of Axiell Designer.
2. Copy the *Missions* and *Mission items* data sources from the stripped *Mission* application definition file (.pbk) in your temporary folder to the definition of your target application: you can right-click a data source and select *Copy* in the pop-up menu to copy it, and then you can right-click the target application title (*Adlib XPlus 4.5* for example) and select *Paste* from the pop-up menu to paste the copied data source. After pasting a data source you can still drag it to the desired position in the data sources list. Save the changes to the target application definition.
3. From Collections 1.7 users can print data from selected mission records, on the condition that one or more custom output formats have been set up for the *Missions* data source. So if you'd like to offer this functionality, you'll have to create one or more output formats and associate them with the *Missions* data source.
4. Still in the *Application browser*, open the *\data* folder and select any earlier existing database definition, that of *collect* for example. On the *Database properties* tab, check out the *Data storage* section. These details have to be copied to the new *mission* and *request* (.inf) database definitions. The easiest way to do that is by changing one character in the *Data Source Name* or *Server* – remember what it was originally – and leave the entry field. Designer will now ask you if you want to apply the changes to all database definitions: click *Cancel* to refrain from doing that. Now change the single character back to what it was and leave the entry field again. This time choose *OK* when Designer asks whether

it should apply the change to all databases. This will effectively copy the correct data storage details to the *mission* and *request* database definitions.

5. On the *Database properties* tab of the earlier existing database definition you'll also see the *Locale* option. Remember the setting and open the *mission* database definition. If it states a different locale then set it to the language and country that you remembered from the other database definition. Once you leave the field, Designer will again ask if it should apply this change to all database definitions. Choose *OK* so that this locale will also be set for the *request* database definition. Save all changes in the affected database definitions.
6. Now you must have Designer create all relevant tables in the SQL database for the new *mission* and *request* databases: right-click the *mission* database definition and select *Clear database* in the pop-up menu. Confirm your choice. Now do the same for the *request* database definition but only if it didn't exist before! If it already existed, the SQL tables will be in place already and they might contain data, so then you don't want to clear this database. The *Clear database* option has now created the appropriate, empty tables in the SQL database.
7. *AdmoveServerSetup.xml* in the `\App_Data` subfolder of your Axiell Move Server folder now needs to be extended with settings to enable the missions functionality and to tell the server which fields to read from and write to. You can use English field names or tags. Change the `<Path>` value (twice) in the XML below to an appropriate full path, accessible to your Axiell Move Server, to the location of the relevant database, *mission* or *request*. Then copy the `<Missions>` and `<MissionActions>` sections from below to your *AdmoveServerSetup.xml* file, underneath the root `<AdmoveSetup>` node. Move any XML end tags isolated on a single line to the line above it: so two lines like
 

```
<DestinationPackageField>package_destination.id
</DestinationPackageField>
```

 should become a single line without spaces, the two XML tags enclosing the value. You already have an `<AdmoveUISettings>` section: copy the `<EnableMissions>` and `<AllowManualMissionActions>` lines from below to the existing section.

```
<!-- Setup for the missions -->
<Missions>
  <Path>G:\Adlib\Model4.5\data+mission</Path>
  <!-- fields used in the mission database -->
```

```

<UserField>user</UserField>
<DescriptionField>de</DescriptionField>
<DueDateField>du</DueDateField>
<DueTimeField>tu</DueTimeField>
<DeviceNameField>dn</DeviceNameField>
<LoadedByUserField>lu</LoadedByUserField>
<LoadedOnDeviceDateField>ld</LoadedOnDeviceDateField>
<LoadedOnDeviceTimeField>lt</LoadedOnDeviceTimeField>
<StatusChangedDateField>sd</StatusChangedDateField>
<StatusChangedTimeField>sh</StatusChangedTimeField>
<StatusField>ms</StatusField>-->
<NotesField>notes</NotesField>
</Missions>

<MissionActions>
  <Path>G:\Adlib\Model4.5\data+request</Path>
  <MissionIdField>mission_id</MissionIdField>
  <ActionTypeField>action_type</ActionTypeField>
  <ActionStatusField>request.status</ActionStatusField>
  <!-- <SourceItemField>object_number</SourceItemField>-->
  <SourceItemField>mission_object_number</SourceItemField>
  <SourcePackageField>package_source.id</SourcePackageField>
  <DestinationLocationField>location_destination.id
  </DestinationLocationField>
  <DestinationPackageField>package_destination.id
  </DestinationPackageField>
  <StatusChangedDateField>status_change.date
  </StatusChangedDateField>
  <StatusChangedTimeField>status_change.time
  </StatusChangedTimeField>
  <NotesField>request.details</NotesField>
</MissionActions>

<!-- Client (tracker) interface settings -->
<AdmoveUISettings>
  ...
  <EnableMissions>true</EnableMissions>
  <AllowManualMissionActions>true</AllowManualMissionActions>
</AdmoveUISettings>

```

## 8. Recycle you Axiell Collections application pool and/or restart Adlib for Windows to check out the results of your changes.

The *Missions* interface in Axiell Collections requires some explanation too. See the [Axiell Collections online Help](#) for all relevant documentation.



---

## 3.4. Setting up the iPhone

### ■ Selecting a local network

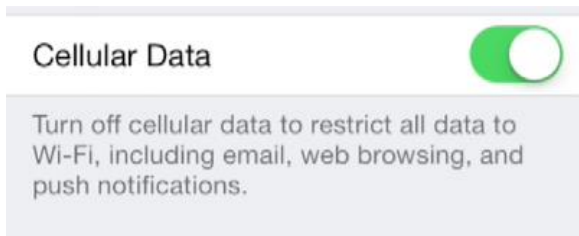
The iPhone must first be set up to be able to contact the local wireless network. You do this as follows:

1. Click iOS *Settings* icon and then select the *Wi-Fi* option.
2. Choose your local company network from the list of available networks and enter the password (your system administrator knows the password).

### ■ Submitting data through Wi-Fi only

By default, the iPhone uses a 3G or 4G dial connection to send data over the internet when no Wi-Fi network is available. This may cause unintended expenses though. Therefore it's good to know that this functionality can be switched off at any time:

1. Open the *Settings* app again and select the *Cellular* or *Mobile data* option.
2. Now switch the *Mobile data* or *Cellular data* off by dragging the slider to the left. (Green means: switched on.)



### ■ Creating a VPN configuration

If people have to be able to work online with Axiell Move, outside the reach of your local Wi-Fi network, you'll have to be able to count on an external Wi-Fi network or you'll have to allow a 3G/4G dial connection to send data over the internet (see the *Mobile/Cellular data* option discussed above). However, if the Axiell Move server has been set up securely, the iPhone will need an active VPN connection to actually be able to contact the server through 3G/4G or via an external Wi-Fi network.

You create and activate a VPN configuration via *Settings > General > VPN > Add VPN Configuration*. At the top of the configuration, set the VPN connection type according to how your VPN server has been set

up. Further, provide a name (*Description*) for this VPN configuration, the name of your company VPN Server, the *Account* under which to contact the VPN server, the *Active Directory Password* of the account user, and switch *Send All Traffic* on. You can probably leave the *Proxy* setting to *Off*. After creating the VPN configuration, you can switch the VPN connection on or off at any time via the *Settings > VPN* slider.

Cancel Add Configuration Save

L2TP PPTP IPSec

Description Required

Server Required

Account Required

RSA SecurID

Password Ask Every Time

Secret

Send All Traffic

PROXY

Off Manual Auto

### ■ Setting the date and time

It is very important that the current date and time have been set correctly on the iPhone. This is because that date and time will be associated with your transactions and an incorrect date (a date in the past for example) may cause new transactions to be registered in the location history, instead of as the most recent. You set the date and time via the *Settings* app: in it, select the option *General > Date & Time* and make the proper settings.

### ■ Registering your fingerprint for Touch ID

Anyone with the access code to your Apple device supporting fingerprint recognition, may register a fingerprint for Touch ID, a means to

pay for purchases and to quickly log in to various apps by simply resting the registered fingertip on the Home button of the Apple-device. Axiell Move also supports logging in this way, although you must realize that any registered fingerprint cannot be linked to any particular user and password: logging in via Touch ID always means logging in under the previously logged in user name and password.

This also means that you should never share your access code to an iPhone with fingerprint recognition or that an iPhone with fingerprint recognition shouldn't be shared with colleagues at all or that your shared iPhone with fingerprint recognition is allowed to log in via a single general user name and password so that all users may log in via their fingerprint and transactions may be logged under that single name.

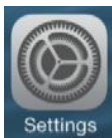
A fingerprint can be registered as follows:

1. In the *Settings* app, select *Touch ID & Passcode*.
2. Enter the passcode to the iPhone, if requested.
3. Click *Add a fingerprint* and follow the instructions on screen.
4. At the top of the Touch ID settings, select the general apps for which Touch ID is allowed to be used. Axiell Move won't be listed here.
5. If co-workers use this iPhone too, it is recommended to leave the passcode validation for the *Touch ID & Passcode* settings switched on. So, do not click *Turn Passcode Off*.  
On the other hand, if no access code has been set up yet, please do so now.

---

### 3.5. Initializing Axiell Move

Via the iOS *Settings* app you'll have to make a few settings for Axiell Move before you can start working with it. In *Settings*, scroll down to the bottom of the list and click *Axiell Move*.



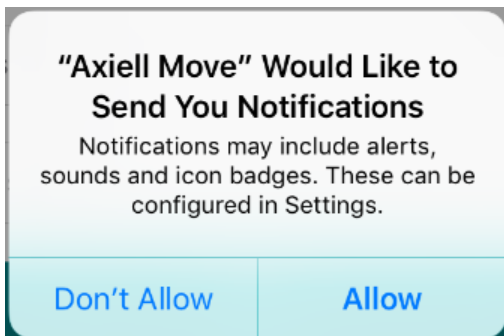
Axiell Move

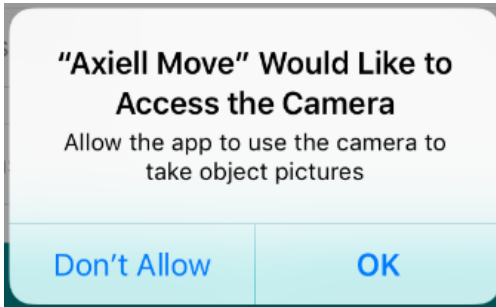
To start with, you'll have to indicate whether Axiell Move is ever allowed to use the camera to scan barcodes. (The microphone won't be used so you can leave that option switched off.)



Use the *Notifications* menu to decide if you'd like to see Axiell Move alerts in the notification center and on the lock screen of the iPhone and also to determine if you'd like a red (alert) badge to be attached to the Axiell Move icon, showing the number of stored transactions to be synchronized (if any).

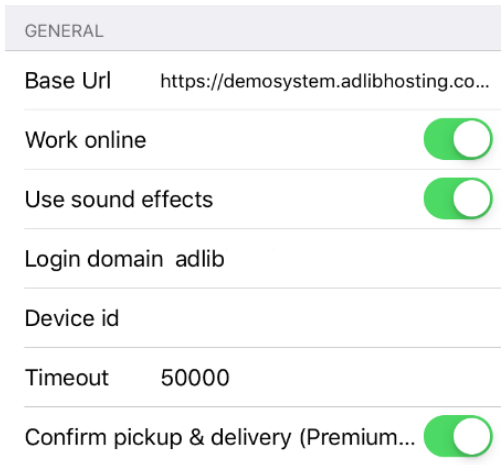
When you first start the Axiell Move app, you might be asked to grant the app permission for sending notifications and access to the camera (required for scanning barcodes). Both permissions can be changed via the *Settings* for Axiell Move.





Scroll down in the Axiell Move settings to set the following options as well:

- **Base Url** – Enter the URL to the Axiell Move server which will be processing all transactions.



Depending on your website URL and the virtual folder underneath which you've installed the Axiell Move *application* on the server, that URL will be comparable to something like:

`https://mymuseum.com/axiellmove/handler`. (Note that older versions of Axiell Move should still point to `admovehandler.ashx` instead of `handler`.) First, put the cursor in the *Base Url* field and then enter the URL using the virtual keyboard. You only need to do this once.

Start the Axiell Move app itself to test the URL (and your connection to the server). If the Move server cannot be found, then the *Cannot connect to server...* warning will appear at the

bottom of the login screen. If the entered URL really is correct, then you may have made an error in the installation of the server.

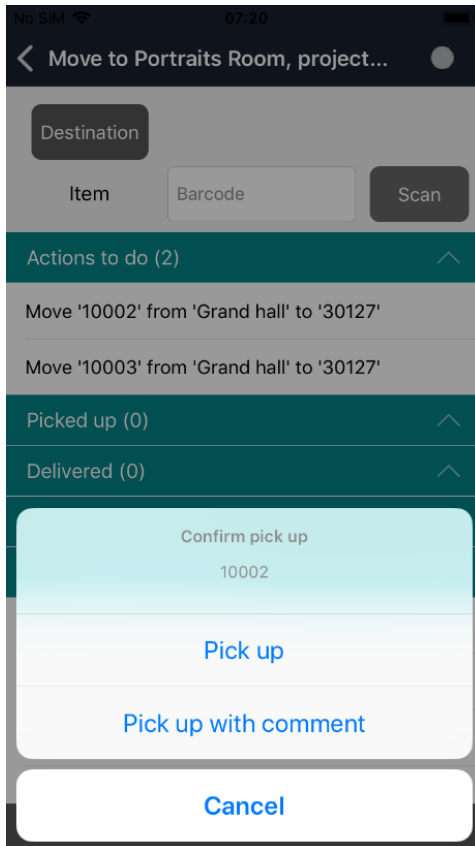
- **Work online** – Indicate your preferred work situation. Drag the slider to the right (the background turns green) if you want to work online whenever possible. Drag the slider to the left if you always want to work offline: during offline work, no information can be retrieved from the database and transactions will be stored in a log file for synchronization at a later time.
- **Use sound effects** – After each transaction, a message will be displayed on the screen to indicate whether the transaction succeeded or not. If you would also like to get an audible confirmation (so that you won't need to read the messages), then use this option to switch sound effects on: you'll hear a ping sound if the transaction was successful and a buzzer sound if the transaction failed.  
The sounds the scanner is making cannot be switched off here.
- **Login domain** - If your Axiell Move system is using Active Directory-authentication, then you'll have to enter the name of the local network domain here. In all other cases you don't need to enter any domain name.
- **Device id** – This is an identification of the current device. Optionally you can have this id (together with the user name) stored in records updated by Axiell Move (for the current location and location history), so that at any later time you can search Adlib for all transactions performed with a certain device.  
By default the id of your iPhone will be present here but you may enter another text if you like, or empty the property instead. If you use a device id, you'll have to set the two Adlib target fields in which the relevant id must be saved with each transaction, in the Axiell Move server configuration file: these can be existing fields or new fields which you added to the relevant database and screens via Adlib Designer. If you do use a device id but do not set any target fields in the configuration file, then the id won't be stored in records.
- **Timeout** - Provide the number of milliseconds after which a request will be cancelled. Sometimes the server can be so busy that it might take a long time before your transaction or search is executed, and sometimes the request cannot be executed at all. To prevent the client from waiting forever, you set a time-out interval. 5000 milliseconds for example, means that the request will be cancelled after 5 seconds. We recommend to set this value to 15000 or higher. After any time-out error message you should

always check whether the transaction has actually been processed in the database or not, before you try to execute the transaction again via the iPhone.

- **Confirm pickup & delivery** – When switched on, pickup and delivery tasks from mission items, need to be confirmed by the user. The advantage being that inadvertently clicked actions can still be canceled and that confirmed actions can be complemented with a comment if desired.

Without confirmation though, you can work faster.

(Missions are only present in the Premium version of Axiell Move.)



---

## 3.6. Installing an update of the software

The Axiell software gets improved regularly. It is possible that at any time just an update of the app becomes available, or only of the server software or of both.

### ■ The app

When a new version of the Axiell Move app becomes available, it will be retrieved by your iPhone automatically\* and installed over the earlier installed version. Your settings and user history will be saved so that you may continue with your work immediately after the update.

\* The update will only be applied automatically if the *Updates* option underneath *Settings > iTunes & App Store* has been switched on. If the option has been switched off, you'll have to open the *App Store* yourself to check whether an update for the app is available and then start the update manually.

### ■ The server software

The server software can easily be upgraded manually: from the Axiell Helpdesk you will have received a zip file containing a new `\bin` folder and possibly new versions of the `global.asax` and `web.config` files.

Open your earlier installed `\Axiell Move Server` folder, rename the existing `\bin` to something like `\bin-old` so that you have a backup and then paste the new `\bin` folder right next to it. You can now replace the `global.asax` and `web.config` files by the new files.

Since your `AdmoveServerSetup.xml` configuration file in the `\App_Data` subfolder has remained untouched, your configuration will still be intact.

---

## 3.7. Security considerations

By default, Axiell Move sends its data in unencrypted URLs. This means that anyone "listening in" on the (wireless) signal can attempt to decode user IDs and passwords in the passing data and even spoof movements. This might not be a problem on a LAN, but when you're working on a WAN this security issue is probably not acceptable. The solution is to use the secure HTTP protocol to transmit data (HTTPS). Then proceed as follows:

- Install an X.509 certificate on your Axiell Move server. Such certificates can be requested from appropriate certification authorities.
- Make sure that you have an Axiell Move version of 27-04-2014 or later. These versions of client and server actually encrypt all



transmitted data if the certified server URL starts with https://. This is done by POSTing encoded commands to the server instead of embedding them unencrypted in the URL (using GET). Note that GET requests are still supported, so that older clients won't fall over and won't need to be changed.

- In the *Base URL* setting in the Axiell Move settings on the iPhone, simply change the "http://" part (the so-called protocol identifier) in the URL to "https://" and leave the rest of the URL as it is.

---

## 3.8. Troubleshooting

If you encounter any problems during your work with Axiell Move and the iPhone/scanner combination, you may be able to solve them yourself quite easily.

- **Records haven't been updated after online synchronization**
  - During offline work, it is in principle possible to prepare incorrect transactions which will only surface during online synchronization when all proper checks will be performed. Any errors in the transactions or errors during the synchronization of correct transactions (e.g. when a record to be updated is in edit mode already), are registered in the *admoved.log* text file in the *\Logs* subfolder underneath *\Axiell Move Server* on the server, accompanied by the date and time. The details of the failed transaction itself are included. If Axiell Move reports errors during synchronization, you will always have to open the log file on the server to see what went wrong. Using that information, you should be able to reconstruct the problem and repair it. A repair could mean that you still make the correct transaction using Axiell Move, or that you edit object, location or package records manually. Other than this recommendation, it is not possible to provide a general instruction for correcting errors.
- **Axiell Move app no longer responds**
  - Try to close the app and restart it. On the iPhone you do this by pressing the Home button twice (after which all opened apps are shown), to subsequently close the Axiell Move app by swiping it upwards. Then restart Axiell Move.

- **The scanner is no longer functioning**
  - The scanner sleeve itself should be charged, not just the iPhone. Press the thin window with the dimmed or green LEDs at the bottom of the scanner sleeve (this is a button as well). One or more green LEDs should light up for a few seconds: four green lights means it is fully charged.
  - The scanner sleeve can be reset. You do this by keeping the scan button (right side, almost on top) pressed down and simultaneously pressing the thin window with the dimmed or green lights at the bottom of the scanner sleeve. Keep both buttons pressed down for about four seconds until the sleeve beeps. Now check if scanning from within Axiell Move is functional again.
- **A barcode is not being scanned properly**
  - Maybe the barcode has become unreadable because the lines have faded or because of stains.
  - The scanner sleeve itself should be charged, not just the iPhone. Press the thin window with the dimmed or green LEDs at the bottom of the scanner sleeve. One or more green LEDs should light up for a few seconds: four green lights means it is fully charged.
- **Error: object, package or location cannot be found**
  - If *admoved.log* reports a problem similar to the following: *Error executing command 'searchlocation': Adlib.Exceptions.TagNotFoundException: Tag 'pg' not found in database 'collect' in record '#####'*, this is an indication that the mentioned field tag has not been defined in the data dictionary of the relevant database yet. The solution is to manually create a field definition for this tag via Axiell Designer or to use the Axiell ValidateDatabase tool to have all field definitions for such undefined tags created automatically.

Should the problem persist even after having tried the solutions above, then please contact the Axiell Helpdesk.