

# CCMS: Capi Case Management System

*Gerrit de Bolster and Peter Sinkiewicz, Statistics Netherlands*

## 1 Intro

Data collection with Blaise is not limited to the design of questionnaires. The process of data collection also needs a logistic system to get sample data and instruments to the interviewers' laptop, and to return the respondent data back to office. Most organizations already made their own custom toolset, adapted to their specific infrastructure and standards. For organizations new to Blaise, reinventing such toolset is rather inefficient. Making use of existing knowledge present in the Blaise community is an appropriate way to go. Started as an initiative of the BCLUB, the Case Management working group focused on the requirements for a CAPI case management system. CCMS is developed to meet those requirements and will be made available to the Blaise users.

*Figure 1. Logo of CCMS*



### 1.1 Some characteristics of CCMS

- Based mainly on Blaise technology
- Cheap: additional functionality provided by open software/freeware
- To be used at the office side, by the interviewers (and their supervisors)
- Fully automated synchronization of data between office and interviewer
- Build with the KISS principle in mind: Keep It Simple and Secure
- Robust and foolproof; self-recovering
- Secure

### 1.2 The scope of CCMS

CCMS is not a complete survey management system. Its functionality is limited to the logistics between the office, supervisors and interviewers and to the case management system on the laptops of the interviewers and the supervisors. So on the office side the functionality of CCMS is restricted: it only takes care of storing interviewer and supervisor data and interchanging those data with the other parties in a robust and secure way. Which respondent should be assigned to which interviewer is outside the scope of CCMS.

## 2 Scenarios

CCMS recognizes 3 parties: the office, the interviewers and when applicable a third in-between party e.g. a regional office or a supervisor. The surveys and the survey data of respondents are considered as “payload”. There are different scenarios possible in the communication and tasks between the 3 parties. Unlike the office and the interviewers a supervisor/regional office is not always present in all the scenarios.

Due to technical or organizational limitations no single solution can suit the needs of each organization. We have considered the situation where an internet connection is available to all or part of the interviewers and the situation where interviewers have to visit their ‘regional’ office. Besides transmitting data through the internet also transmitting data by means of a memory stick or memory card has been taken into account.

## **2.1 Scenario “A”**

In this scenario only two parties are involved: the office and the interviewers. The office provides the payload; the interviewer on its return will treat the payload and then return the results. In CCMS it is foreseen that the transport of the data between both parties can be organized in different ways: ideally the interviewer connects with its laptop on a regular basis through the internet to a (s)FTP-server located at the office, but the transport of the payload can also be accomplished in another way, like by a memory card through regular mail.

## **2.2 Scenario “B”**

This scenario involves a third party between the office and the interviewer, like a regional office or agency or a local supervisor. There can be various reasons for having such third party: a traditional regional infrastructure, outsourcing of the interviewing, limited ways of transport or limited communication as in mountain regions. The role of the third party can be as little as just being a serving hatch, passing on the payload. But also other tasks could belong to this party, like quality control, payment of remunerations and so on. For some of those tasks it is possible that this third party needs access themselves to the contents of the payload or part of it. This is the case when the third party has the task to divide the workload between a group of interviewers.

The main concern of CCMS is to offer a solution for the logistics. As it comes for instance to the reassignment of the payload to another interviewer it is considered best to be done on one place, at the office. Any suggestions of scenarios to be taken care of by CCMS are welcomed.

## **3 Data communication**

The data communication system must meet three basic requirements:

It has to be protected.

It must be robust.

It must be very easy to use (usability).

### **3.1 Protection**

There are several communication protocols available. They can be divided into two groups: protected protocols (sFTP, FTP-s, HTTPs) and unprotected (FTP, e-mail, memory sticks/cards). Using a protected protocol it is not necessary to protect the payload. However, when applying an unprotected protocol it is a must that the payload is protected by encryption. With Blaise build 4.8.2.1606 strong encryption (AES) was introduced for the zip and unzip commands in Blaise. With this encryption it is very easy to protect the payload. It can cause a problem if e-mail is used. Providers can have filters that block encrypted attachments. In that case the only solution is to use a provider that, by agreement, does not block the encrypted zip files.

In the current CCMS we applied sFTP and memory sticks/cards and AES 256 bits protected zip files with a key of 64 characters. Each interviewer/supervisor has his own key provided by the office.

### **3.2 Robust**

During a contact between the interviewer and the office (or supervisor) files are exchanged in two directions: from the office to the interviewer and vice versa. This synchronization must be robust. In other words: it must not be possible that a communication fails without being noticed. The best way to secure such synchronization is to work with confirmation messages. Especially when indirect communication is involved (e-mail, memory sticks/cards by mail) this is a must.

The synchronization is always initiated by the interviewer (client side) by starting this function in the CCMS. There are different reasons and preconditions for an interviewer to start this synchronization. In case of an online connection the connection must be established. In case the exchange takes place with offline media it can be the reception of a memory sticks/card from the office.

At the office side CCMS will be used to check on uploaded files from the interviewer and preparing files for download. In the majority of cases the office will have a scheduled plan for this action. This

is for certain the case if the synchronization takes place by sending memory sticks/cards as the process will not be started for each stick/card separately.

Given this situation it is better to perform on the client side (interviewer, supervisor) first the download and subsequently the upload. Applying this order, the confirmation of the downloaded packages can be included immediately in the upload. The confirmation of the upload by the office will be included in a later download. To secure the synchronization there must also be procedures installed that warn the parties involved when confirmations seems to be missing. They could be triggered by time and/or by e.g. a broken sequence number.

All interchange actions are recorded in a logging. The logging on the interviewer side can be send to the office on initiative from the office to check what happened in case of a problem. If necessary the office can update the logging on the interviewers' side. The logging is also used to register confirmations and to avoid double actions.

### 3.3 Usability

The data communication must not be too complicated to use especially on the side of the interviewer. Just a click on the button will invoke all the necessary actions. The system not only supports the interchange of surveys and respondent data but also the communication between the interviewer and the office/supervisor. Hence an additional separate e-mail system is not necessary.

## 4 Laptop access

The protection of the laptop itself is not an issue for the CCMS. It is assumed that the laptops are protected according to modern standards. Therefore the stored surveys, respondent data and statements are not additional protected. However, the use of CCMS on the laptop has its own (limited) protection to avoid unauthorized use (deliberately or accidental).

In the laptop part of CCMS there are 2 files essential for its operation: the **configuration** file and the **interviewer** settings file. More than one interviewer can use the same laptop. Each interviewer has its own sub-folder with surveys and respondent information.

The configuration file contains the supported data communication modes, data communication access info, used symbols and some general settings. It is a Blaise file with a code/value structure of which the value (a string) is encrypted with a special developed encryption method. To decrypt this value a main key and a local key are needed. The main key of 255 characters is integrated in the Blaise setup, the local key is stored in the settings file of the interviewer.

The interviewer settings file contains the interviewer number (for verification), the local key to access the configuration file and the 64 characters ZIP key to access the AES 256 bits protected zip files used for the interchange of surveys and data with the other parties. It is also a Blaise file with a code/value structure of which the value (a string) is encrypted with the same special developed encryption method as used for the configuration file. To decrypt this value a main key and a local key are needed. The main key (the same as used for the configuration file) is integrated in the Blaise setup; the local key must be typed in by the interviewer in combination with the interviewer number at startup of CCMS. If the CCMS is not able to decrypt the interviewer settings file the access to the CCMS is denied.

As well the local key as the zip key of an interviewer can be changed at the office side and sent to the interviewer within CCMS. The system takes into account that some time may be needed before a new key is operational on the interviewers (or supervisors) side.

## **5 Open architecture**

### **5.1 The software**

As CCMS should be applicable for different organizations the kind of software involved plays an important role. It must be clear that Blaise has the preference. E.g. instead of using batch files it is possible to create a text file that is used by a Maniplus setup to invoke all kind of commands. In case there is no direct or indirect Blaise solution for hand (like in the case of sFTP or the installation program) CCMS is using freeware and/or open source solutions to perform the task. This makes CCMS accessible to more organizations. Of course CCMS should also be able to support licensed communication software. To achieve that goal, the interface with the data communication software is placed in a separate Maniplus setup containing procedures with a standardized call. It can easily be replaced by any other implementation.

### **5.2 Languages**

CCMS supports different languages to be able to be used by different organizations. It concerns the language of the user interface of the software, as well at the interviewer side as at the office side. The surveys have to be designed supporting their own language(s). To be able to support different languages CCMS works with external language files.

On the interviewer side CCMS is multilingual. Not only can an existing language file be replaced by another language file, it is also possible to add more language files. This is useful in countries with more than one official language or in the case interviewers with a different language are hired to address a special minority. In the current version 2 languages are included: Dutch and English. The language can be changed within the user interface by the interviewer.

On the office side only one language file is supported at a time. It can easily be replaced by another language file.

### **5.3 The surveys**

To maintain an open architecture while supporting all kind of different surveys (using different Blaise data models) is a challenge on its own. The (primary) key may consist of one or more fields varying from survey to survey. Therefore the key information of a form of an interview is stored in a general identification field in the respondent information of CCMS. This identification field has been defined as one single string which holds the values of the different key fields, separated by a semi-colon (;). When starting up an interview this key is automatically applied by the CCMS.

Another challenge is formed by the administration forms in which an interviewer can report about the interview itself. Some organizations include this form as a parallel in the questionnaire itself and some define it as a separate data model. In CCMS it is possible to indicate that the administration form is not included and one can specify which separate form should be used.

Support of an audit trail can be added to the surveys by choice. It is possible to activate the audit trail for all surveys, a selection of surveys; for all interviewers or only a subset of the interviewers. It is even possible to activate the audit trail for only a part of the forms (respondents) of a survey.

There are still some wishes to be fulfilled. In some organizations a main survey form is used for all respondents in the sample. Based upon the answers automatically a follow-up task should be generated for another survey. And for sure there are more wishes.

## **6 Technology applied**

### **6.1 Data package & objects**

The data is interchanged between the parties using data packages. These data packages (extension .cmp) are encrypted zip files. They can only be unpacked by the specific interviewer using the 64 character long zip key unique for each interviewer. The office has all the keys and the supervisor those of the interviewers within the supervisor's group.

Each data package contains one or more so-called data objects (extension .cmo). These objects only contain the data (properties and “payload”) and not the methods. When starting the development of CCMS the choice has been made to use separate Manipula programs which contain procedures for these methods, as procedures enable the use of parameters for import, transit and export. The procedures are used as alien procedures from other Manipula programs of CCMS, and will hereinafter referred to as Manipula libraries or Alien libraries. Older Blaise versions expected these libraries to be stored in the same folder of the application were the other Manipula programs were stored. Due to that restriction the methods were not added to the objects. The latest version of Blaise now supports the storage of alien procedures in other folders. However, during development of CCMS this separate storage of methods proved to be very developer friendly so the construction was not changed.

A data object (in reality a zip file) contains object properties and survey data. The properties are used by the methods (procedures) defined in the Manipula libraries to interpret what should be done with the data (payload). There are different types of data objects. Each type has a different role in the system. The type is included in the file name of the object:

<type>{<code>}<date>T<time>.cmo

<code> = see “Object types”

<date> = yyyyymmdd

<time> = hhmmss

Sample: QNR{Commute1}20110602T150600.cmo

The following types of the objects have been defined:

LNG - Languages

NTF – Notification

<code> = function (logging, messages, received, zipkey, etc.)

QNR - Questionnaire

<code> = survey code

RPD - Respondents

<code> = survey code

SWU - Software update

At this moment only the NTF, QNR and RPD are implemented in CCMS. The properties of the data object are stored as an xml file (\_object.xml). The other files are stored in a zip file in the data object (data.zip).

## 6.2 Alien libraries

In CCMS procedure libraries (prepared Manipula setups) are used for the processes around the data objects. These libraries have the advantage that it is possible to use parameters for input, transit and output when calling the procedures stored inside the library. In some cases these alien procedures are invoking procedures of other alien libraries. The procedures (and functions) in such library can read from and/or write to files including temporary files. The alien libraries are linked to a certain object type and the same library will be used in the different environments (office, supervisor and interviewer). Furthermore a common library was added to support all kind of actions regarding the logbook, so this Manipula library is not connected to a specific data object type.

A disadvantage of using these libraries is that if a runtime error occurs when executing an alien procedure, it is not reported to the calling program: it just stops processing the commands. This makes debugging difficult.

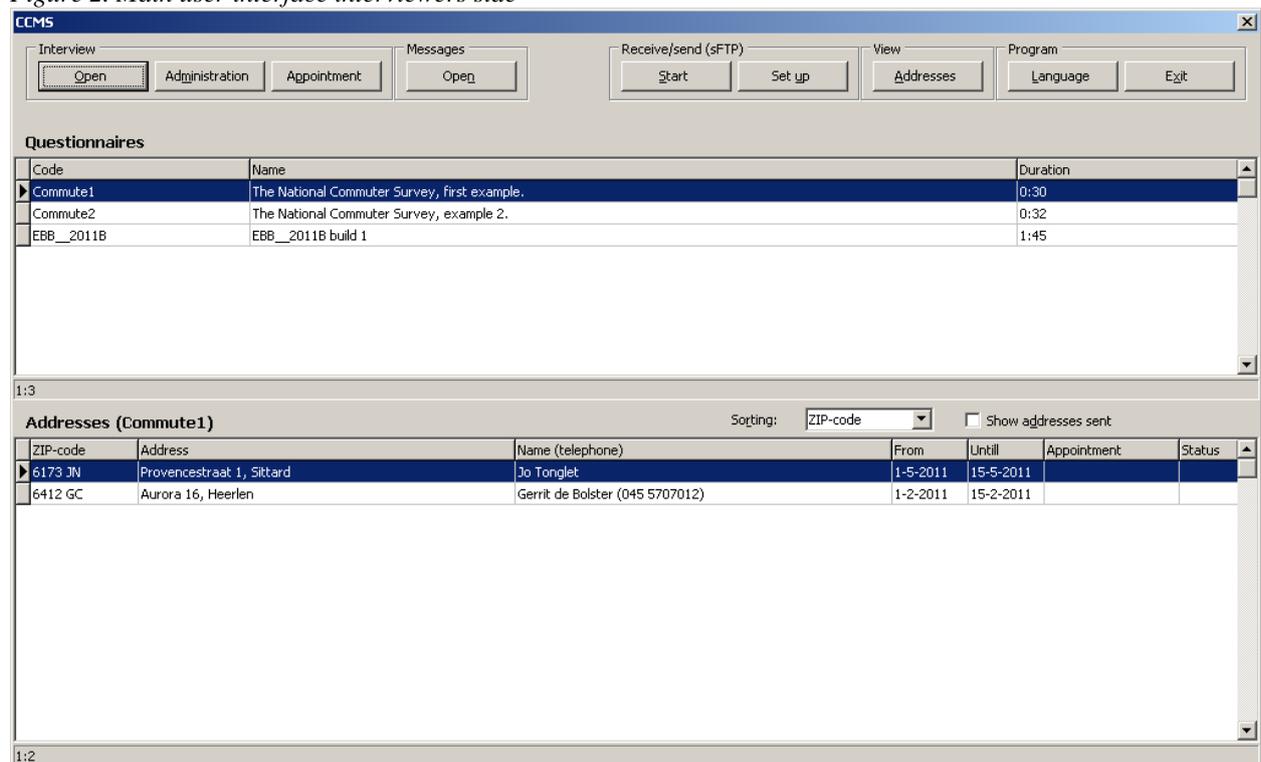
During development it appeared that some behavior of Manipula did not meet our needs. The actual Blaise version at that moment did not formally support that the alien procedures were stored in a different folder than the calling program. It was also impossible to use such alien if its path contained a space. For that reason it was decided to store the Manipula libraries in the folder with the other Manipula programs of CCMS and not to include them in the objects. As it appeared during further

development this decision made it easier to cope with (unavoidable) changes as test objects and data packages did not need to be recreated.

### 6.3 User interface

The user interface at the office side comes with a classical Windows menu. For the interviewers another approach has been chosen. The main user interface consists of a stand-alone dialog window, which functions as a dashboard. The size of this dialog is based upon the screen size of a netbook. A small VBScript is applied to place this dialog in the middle of the screen whatever it size may be.

Figure 2. Main user interface interviewers side



The layout of the dialog at the interviewer side is different too (see figure 2). Similar to modern interfaces the action buttons are placed on the top of the window, more or less like a ribbon. Below these buttons the lookups and other controls are placed. The main interface shows a split screen with the active surveys and the related addresses of respondents with status information. Changing the focus on a survey will change the set of addresses automatically to the one related to the focused survey. These addresses can be sorted in different order according to a choice in a dropdown list. By a click on an action button the split screen is replaced by a single list of all addresses. The related survey is now shown in a column of the list.

For the message function a separate sub dialog can be activated by a button in the ribbon, again with the buttons on the top of the screen. Below the buttons a list of the messages is shown. The interviewer can choose between received and sent messages.

### 6.4 Synchronization processes

This is a global description of scenario “A”. Actions e.g. concerning the logbook are left out.

Naming conventions: An interviewer is receiving (downloading) from the office and sending (uploading) to the office. Therefore the folders are called “Off2Int” and “Int2Off”, as well on the interviewers side, on the (s)FTP server and key/card and on the office side. The office is receiving the data packages uploaded by the interviewers from the “Int2Off” folder and sending data packages to be downloaded by the interviewers to the “Off2Int” folder.

Figure 3a. Data communication using a (s)FTP server

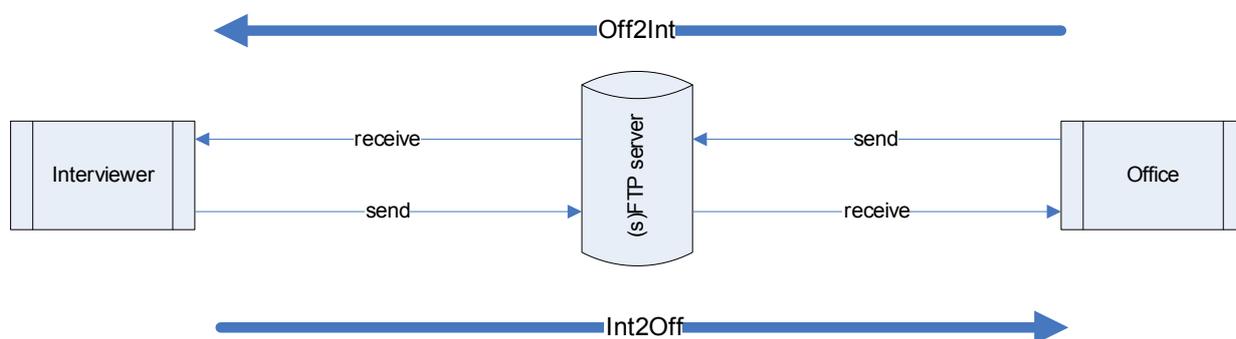
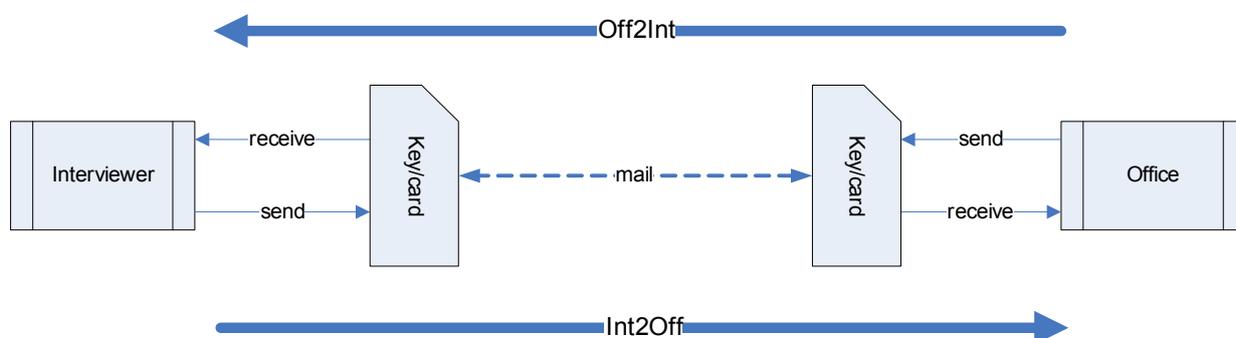


Figure 3b. Data communication using a Key/card



#### 6.4.1 Synchronization process at the interviewer side

See appendix 1.

When the interviewer starts the synchronization process the data packages intended for this interviewer are copied or downloaded (depending on the setting: key/card or sFTP in the current CCMS) to the “Off2Int” folder (process 1[1/2]). These packages can be recognized as the file name contains the identification of the interviewer. The data packages are unpacked (process 2a) using the zip key from the interviewer and the resulting data objects are stored in the receive folder. If successful the data package is moved to the archive folder (process 2b). The data objects are then processed using the init method associated to this type of data object (process 3). A confirmation data object for the received data objects is created in the send folder as well as data objects with completed statements and messages. The data objects are packed in a data package and stored in the “Int2Off” folder (process 4). The data packages stored in this “Int2Off” folder are copied or uploaded (depending on the setting: key/card or sFTP in the current CCMS) to the key/card or remote upload folder (process 5[1/2]). If successful the data packages are moved to the archive folder (process 6).

#### 6.4.2 Synchronization process at the office side

See appendix 2.

The data packages uploaded by the interviewers to the sFTP server are downloaded (or copied) to the “Int2Off” folder (process 1a). Data packages received on key/card are imported to the office system and copied to the same “Int2Off” folder (process 1b). The key/card contains an “Int2Off” and a “Off2Int” folder and only the data packages from the “Int2Off” folder are imported. It is therefore not necessary for an interviewer to clean the key/card received from the office before using it to send data packages to the office as any data packages sent by the office are stored in the “Off2Int” folder. De

data packages in the “Int2Off” folder are unpacked using the zip keys of the interviewers and the data objects are stored in the interviewers’ receive folder (process 2a). If successful the data packages are removed from the FTP server (process 2b). The received data objects are processed using the init method associated to this type of data object (process 3). Resulting data is placed in a shared folder to serve as input for the internal statistical processes. A confirmation data object containing the received data objects is created in the send folder as well as data objects with new surveys, respondent information and messages (process 4). The data objects are packed in a data package and stored in the “Off2Int” folder (process 5). The data packages are uploaded to the download folder on the FTP server (process 6a) and copied to the “Off2Int” folder on the formatted (process 8) key/cards (process 6b). The download folder on the FTP server is synchronized with the “Off2Int” folder removing data packages that already have been successfully download by the interviewers (process 7).

When keycards are used instead the contents of that card resembles the contents of the folders of the sFTP-server: all the data packages for all the interviewers are copied to the “Off2Int” folder of all formatted key/cards; hence this replication process is simple. So after the replication all the key/cards are identical with causes the packing in envelopes and sending the key/cards to the interviewers to be foolproof. An interviewer can only open its own data packages as they are encrypted with the unique zip key of the interviewer. The same applies to the download folder on the sFTP server.

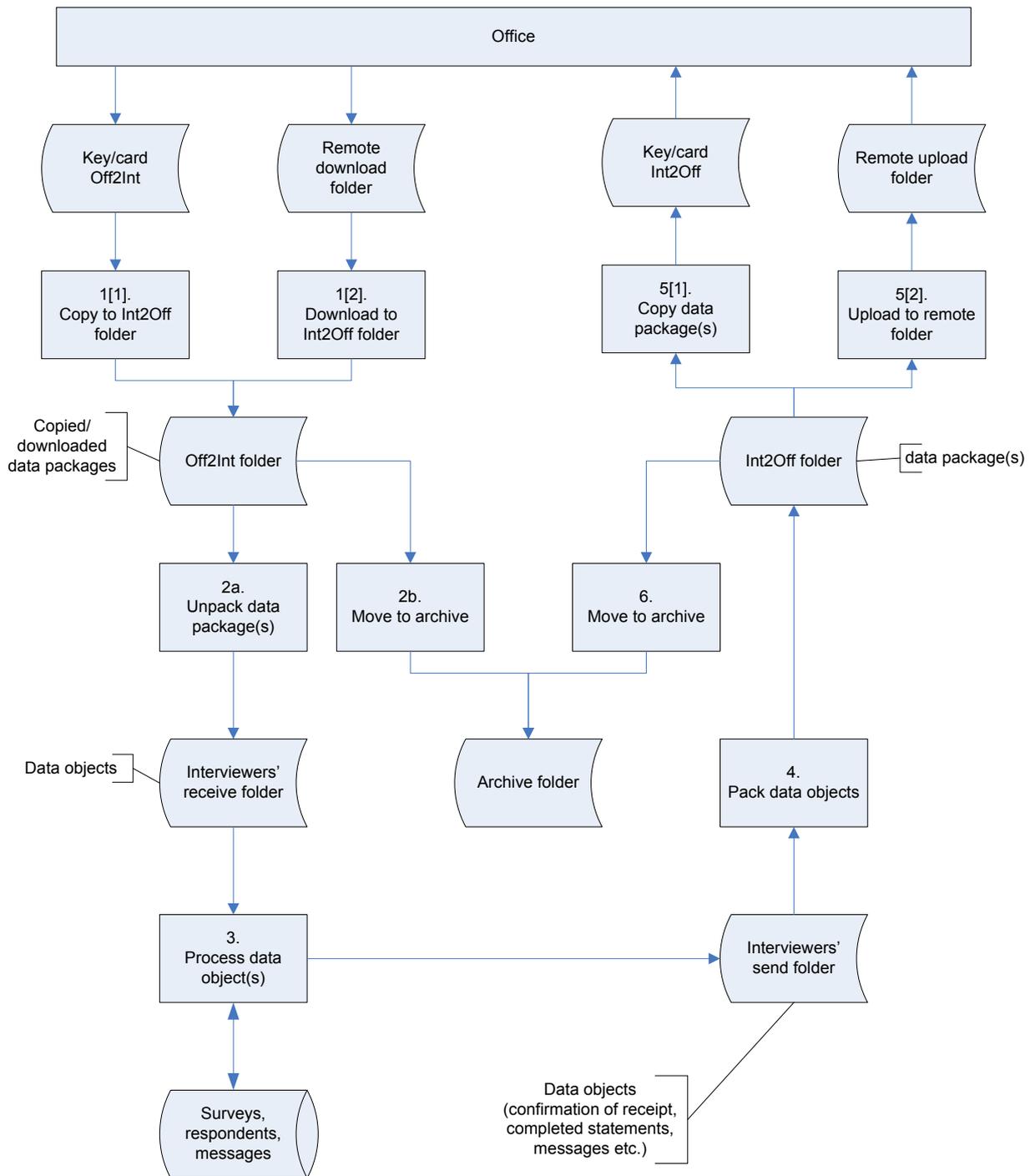
## 7 Summary

Within the defined scope the management system described in this paper is a complete solution for CAPI Case Management at very low costs. A working version is available for free. It includes both the applications as described in scenario A: for the laptop and for the office. It is based on Blaise in combination with freeware. The interchange of information between the office and the interviewer (and vice versa) is robust. The first version supports 2 different transportation media: sFTP and memory keys or cards by classic mail (or a combination of both). Even for the sFTP server a freeware solution is available. It is very flexible concerning the languages applied although the first version only supports the “left to right” languages.

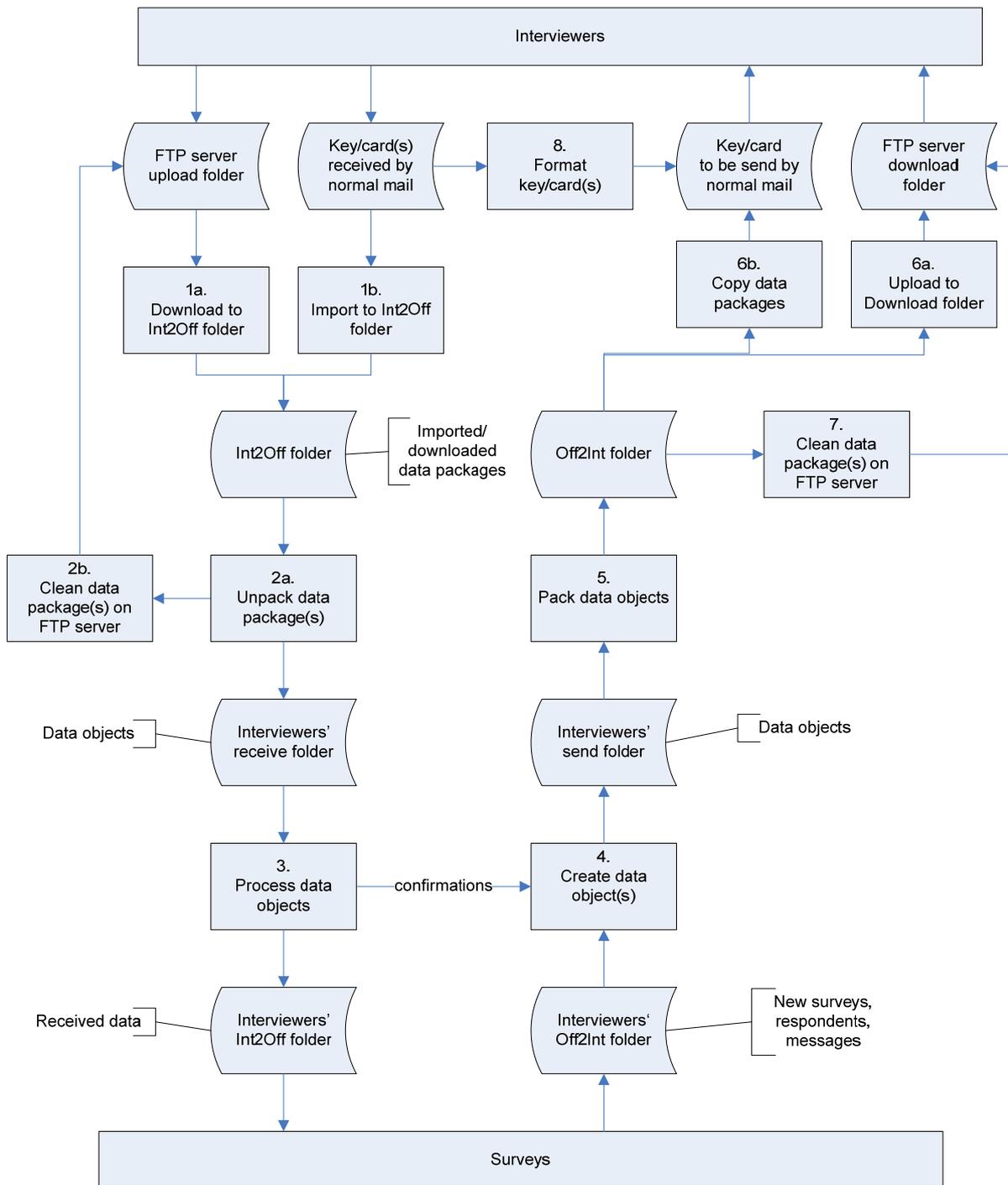
The development is set up as open software. Every interested person is invited to join the “CCMS community”. Not only to use it for free but also (if possible) to contribute. The software (including the sources and documentation as far as it is available) is distributed using DropBox ([www.dropbox.com](http://www.dropbox.com)). Contact the authors of the paper for more information.

# Appendixes

## 1. Synchronization process at the interviewer side



## 2. Synchronization process at the office side



### **3. System requirements**

The same system requirements that apply to the Blaise software are applicable for using CCMS. CCMS does support the management of questionnaires created with older versions of Blaise like 4.6. However, for executing CCMS on the laptop/netbook of the interviewer the Manipula program (Manipula.exe) of Blaise 4.8.3.1735 (or higher) is required. CCMS in the office will also work with a Blaise 4.8.2 version, e.g. build 1700. At the office side in addition to the Manipula program CCMS also uses the Blaise dataviewer (dataview.exe). From the licensing point of view Blaise licenses are required as described on the website of Statistics Netherlands. Hence Capi end-user licenses will be required for the interviewer laptops/netbooks. Creating questionnaires or making adaptations to CCMS can only be done when the organization possesses a developer license as well.

For the data communication (client and server) and even the installer there is enough suitable freeware available on the Internet. In the first version some samples are included.

CCMS (interviewer and office side) will run on Windows XP as well as on Vista and Windows 7 (32 and 64 bits).