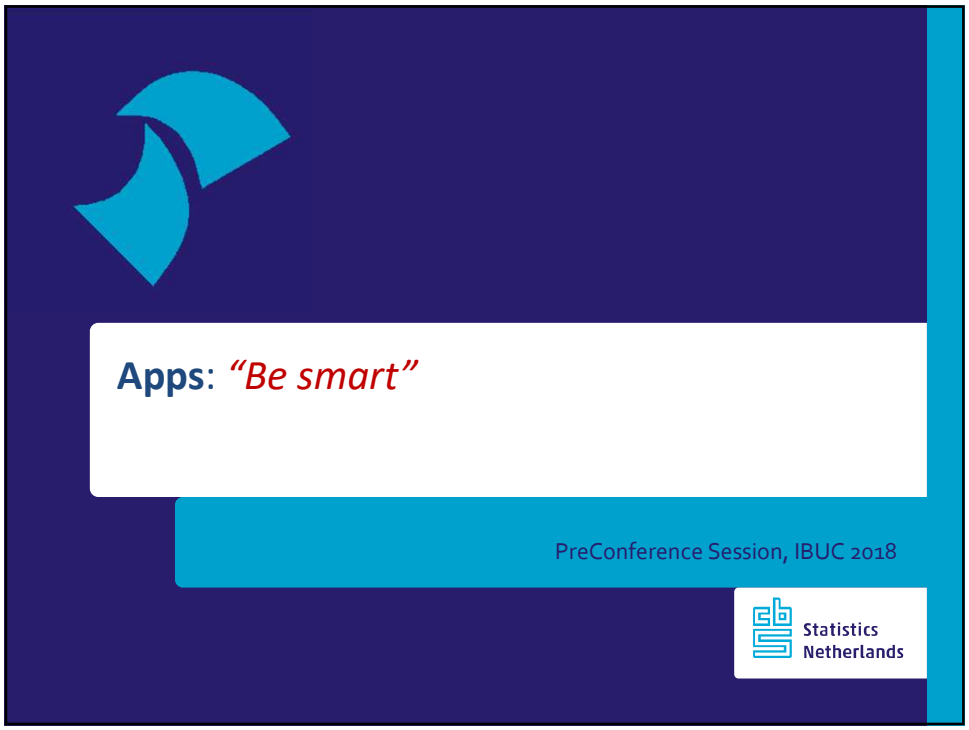


Overview

- Introduction to Blaise apps
- How to set up:
  - Server(s) (Server Park, Roles, Users)
  - Device
- App features
- OfflineCAPI block
- (Auto)Synchronizing of surveys and data
- Encryption
- Customization
  - Dep App branding; Starter Kits
- 5.5 extensions
  - Restrict functions in app
  - Intercepting data at data server
  - Maniplus support

Blaise Apps

- Available apps
  - Dep App (Windows)
  - Android App
  - iOS App
- Native apps; they don't use a browser to display things
- Support for multiple run modes, but main purpose is to run surveys offline in a distributed environment
  - CAPI
  - Download and upload of items if a connection is available
- Can be used by end users as well

App features

- Data Entry Client
  - Run any Blaise 5 survey, online and offline
- Administrative tasks
  - Login to server
  - Get list of available surveys
  - Download and install surveys
  - Download of cases
  - Upload of cases
  - Browse data that is stored on device
  - Removal of cases
  - Removal of surveys
  - Alter settings

Server Setup (1)

- Set up server park(s)
  - Run Mode
  - Data Entry Server
    - Should be publicly accessible
    - Port 8033 (by default)
    - Binding http or https (requires certificate)
- Users
  - Grant access to server park(s)
  - Grant ‘Download packages to apps’ and ‘Survey Data Access’ privileges
  - In 5.5 possibility to grand access to individual app features

Server Setup (2)

- Run mode
  - Determines how survey is executed by the apps
    - Client/Server
      - Rules execution and data storage is done at server
    - ThickClient
    - ThickClient with fallback
      - Survey runs on device, but data access is done at the server; fallback stores the data on device if no connection is available
    - Disconnected
      - Survey runs completely offline and data is stored on device

Server Setup (3)

- Install survey in appropriate server park
- App package
  - Package that is downloaded by the apps
  - Will be created if survey is installed in Disconnected or ThickClient server park
  - Stripped down version of regular package (*.bpkg)
    - Contains no survey data
      - Data file is replaced by empty SQLite database
    - Contains only app related files
  - <survey name>.app.bpkg

Device Setup

- iOS and Android
  - Download Blaise Data Entry app from Apple App Store and Google Play Store respectively
- DepApp (Windows)
  - Install Blaise; run depapp.exe from Blaise installation folder
  - Don't install Blaise; copy the following files to a folder on your device; run depapp.exe from there
    - DepApp.exe
    - DepApp.exe.config
    - System.Data.Sqlite.dll
    - Msvcrt100.dll
    - StatNeth.Blaise.Shared.Windows.dll (if branding file is used)

App Settings

- Apps require logon
- Log on to data entry server
  - Specify user credentials
  - Data entry server
  - Port
  - Binding (http / https)
  - Run offline
    - If true then no attempt will be made to connect to the server. Only offline functions are available
  - Deploy location (Dep App only)
    - Location where survey files and settings are stored

List of surveys

- Get list of surveys that are available to the user
  - Divided into:
    - Available on server
      - Surveys installed in a client/server Server Park
    - Available for download
      - If installed in a Server Park with run mode set to Thick-Client and Disconnected
    - Installed on device
      - Surveys that are already installed on the device

Installing surveys

- During installation of a survey in

- Download and install surveys
  - App package is downloaded and survey is installed at deploy location
- Remove locally installed survey
  - Deletes survey and survey data
- Browse local data
- Start survey
- Show survey info

Browsing data

- Enabled when Run mode is Disconnected or Thick Client with fallback
- Can be used to view locally stored data
  - Fields to be displayed can be set in a bdvx. Bdvx can be deployed via Custom Package
- From the Browse Data screen you can:
  - Start a case
  - Download cases
  - Upload cases
  - Delete selected case

Demo: app usage

Downloading and uploading data

- Available when device is connected
- First few versions of app had very rough methods for dowqnlaod an upload of cases
  - Record filter had to be entered in app to download cases
    - Error prone: need to use the exact syntax, for example InterviewerName = ‘peter’
    - You really do not want that interviewers have do this!
  - Uploads all records
- Needed a way to control this better
  - OfflineCapi block
    - Gives you more control over the download and upload process
    - Highly recommended if you are going to use the apps!

OfflineCapi block

- Special block recognized and used by the apps
- Provides very basic case management functions
- Important fields:
  - ToWhom
    - Can be used to assign cases to interviewers
      - Login name should match name in ToWhom field
  - CaseStatus
    - The following statuses are used
      - Downloaded; downloaded but not started
      - Incomplete; started but not completed
      - Completed; reached the end of the questionnaire
      - Uploaded; data was uploaded to the server
      - Resend; case will be resend to the user
      - Suspended (new in 5.4.5); case will not be downloaded anymore

OfflineCapi block – 5.4.5 version

```
TYPE
  TCaseStatus = (Downloaded(1), Touched(2), Refused(3),
    RefusedIncomplete(4), Incomplete(5), Complete(6),
    Appointment(7), Problem(8), Uploaded(9), Resend(10), Suspended(11))

BLOCK BOfflineCapi
  FIELDS
    ToWhom: STRING[50], EMPTY
    Status: TCaseStatus, EMPTY
    Date: DATETYPE
    Time: TIMETYPE
    UploadDate: DATETYPE
    UploadTime: TIMETYPE

RULES
  ToWhom.KEEP
  Status.KEEP
  Date.KEEP
  Time.KEEP
  UploadDate.KEEP
  UploadTime.KEEP

ENDBLOCK
```

OfflineCapi block

- New fields in 5.4.4 and later:
  - Upload date; records date of last upload
  - Upload time; records time of last upload
  - Needed to preserve the current status on the device; in older versions the status was changed to uploaded

- New status in 5.4.5
  - Suspended
    - Can be set at server
    - Case will not be downloaded anymore
    - Case will be deleted from device on download cases

OfflineCapi block – Controlling cases

- Assigning cases to interviewers
  - Prepopulate your data file and assign user to the special ToWhom field
  - Apps will only download the cases for the user that has logged in
- Resending cases to interviewers
  - Completed cases will not be downloaded anymore
  - Set status to Resend to (re)enable download
- Suspending cases
  - Case will not be downloaded anymore
- Transfer cases to another interviewer
  - Set ToWhom field at the server to the new interviewer
  - Cases will now be downloaded to the new interviewer and removed from the device of the previous user.
  - In general: cases will be removed if the case is not assigned anymore to the user

OfflineCapi block – Status codes

Code	⇩	⇧	Description
Empty	✓	✓	Case has no status yet
Downloaded	✓		Case is downloaded but hasn't be touched
Touched	✓	✓	Case has been opened
Incomplete	✓	✓	Case has been opened, but has been closed before reaching the end of the questionnaire
Complete		✓	Case has been completed
Uploaded	✓		Case has been uploaded. Is used in 5.4.3 and earlier versions. Has been replaced by upload date and time colums to preserve original status
Resend	✓		Case will be/has been resended to the app
Suspended			Case will not be downloaded (anymore). Case will be deleted from the devices if a sync occurs

⇩ = Will be downloaded ⇧ = Will be uploaded

OfflineCapi status – Overwrite behavior

Status on device	Overwrite behavior
Empty	Case will be overwritten by new download
Downloaded	Case will be overwritten by new download
Touched	Case will only be overwritten if status in download is resend
Incomplete	Case will only be overwritten if status in download is resend
Complete	Case will only be overwritten if status in download is resend
Uploaded	Is used in 5.4.3 and earlier versions. Case will only be overwritten if status in download is resend
Resend	Case will be overwritten by new download

Downloading cases

- OfflineCAPI block
  - Downloads cases that are assigned to the user that has logged on
  - Cases with status Resend will be overwritten
  - Touched, Incompleted and Completed cases will not be overwritten
  - CaseStatus becomes 'Downloaded'

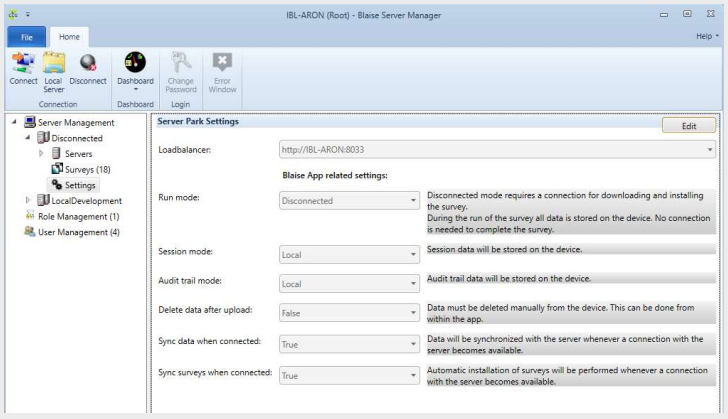
- Without OfflineCAPI block
  - Record filter must be specified in app
  - Option to overwrite existing records. Default = no

Upload of cases

- Uploads survey, session and audittrail data
- Session data is written to survey database before upload (5.4.5)
  - Takes care about partial cases
- OfflineCAPI block
  - Only relevant cases will be uploaded
    - Touched, Incomplete, Completed
    - Upload date and time will be recorded
- Without OfflineCAPI block
  - Uploads all locally stored cases
- Data server writes data to the main survey database
  - Overwrites already existing data!

App related Server Park Settings

- Delete data after upload
- Sync surveys/data when connected



Automatic syncing Surveys

- Synchronizes surveys when connected
  - Fires when wifi or ethernet connection is available
    - Start of app
    - Refresh of survey list
  - Installs missing surveys
  - Installs newer versions of surveys
  - Deletes obsolete surveys
    - If survey is removed from the server, then it will also be removed from the app

Automatic synchronization Data

- Requires OfflineCapi block
- Fires when wifi or ethernet connection is available
  - Start of app
  - Whenever a survey is downloaded
  - Refresh of survey list
- Performs a full sync
  - Downloads relevant cases
  - Uploads relevant cases
  - Deletes cases that aren't assigned to the user anymore
  - Deletes cases which have the 'Suspended' status at server

Demo: automatic syncing

Delete after upload behaviors

- Deletes cases after a succesful upload
  - OfflineCapi block
    - With upload date and time fields
      - Deletes cases with CaseStatus = ‘Completed’
    - Without upload date and time fields
      - Deletes cases with ValidationStatus = ‘Clean’
  - Without OfflineCapi block
    - Deletes cases with ValidationStatus = ‘Clean’

Updating app and surveys

- Update DepApp
  - Replace depapp.exe
  - Surveys and survey data will be preserved
- iOS and Android app cannot be updated
  - Each version has its own id
  - Surveys and cases have to be installed and downloaded again in a new version
- Reinstall of surveys
  - Survey definition is always overwritten
  - Data handling: respects the overwrite data setting as specified during installation of the survey on the server
    - Overwrite data:
      - No, preserves data
      - Only if data is incompatible
      - Yes, data will be overwritten
    - Harmless changes update is also supported

Encryption

- iOS and Android
  - Data files are always encrypted using SQLCipher AES 256bit encryption
- DepApp
  - Data files aren't encrypted; can be encrypted using SQLite's build in encryption based on CryptoAPI

Customizing

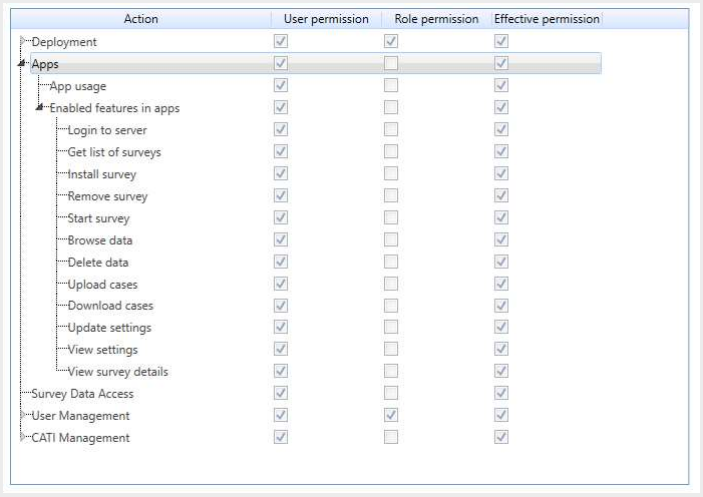
- Dep App
  - Use DepApp branding file
    - Can be used to change font sizes and color schema
  - HasReadOnlySettings in DepApp.exe.config
    - Only UserName / Password are editable
    - Provide DepApp.exe.config with settings for Host, Port, etc
- Other possibilities
  - Starter Kits: based on API's
    - Available for Windows, Android and iOS
    - API's contain all functions that the apps have
    - Create your own wrapper
  - Maniplus (5.5)
    - Includes all the functions that the apps have
    - Create your own screens and run them in the apps

Demo depapp branding and ReadOnly settings

5.5– Grant access to individual features

- Possiblity to control which features are activated in the app
  - Some organizations do not want that interviewers can do adminstrative tasks in the apps, like installing surveys, deleting surveys, uploading, downloading and deleting cases
  - Tasks should be performed automatically
- Grant access to individual app functions in Server Manager
  - Allowed features will be read and activated whenever an user/interviewer logs on
- Surveys and cases can be installed and removed automatically by activating the sync functions in server park

5.5– Grant access to app functions



5.5 – Intercepting data

- Purpose:
  - to give you more control over the update and read process when data arrives at the server or is going to be delivered to the requesting client respectively
- Important for supporting multimode where data can come in from multiple modes or will be read by multiple modes
- How?
  - By installing a special Manipula setup at the data server
  - Data that is going to be written or is going to be delivered (read) becomes available as a temporary file in Manipula
  - Write your own logic in Manipula to deal with the data that is going to be written or read

5.5 – Intercepting data

```
SETUP DataProcessor

SETTINGS
   IsDataProcessor = Yes
   AutoRead = no

USES
   MyMeta (var)

TEMPORARYFILE
  IncomingData: MyMeta
SETTINGS
  Interchange = Import

TEMPORARYFILE
  Database: MyMeta
SETTINGS
  Interchange = Export

OUTPUTFILE
   Output1: MyMeta ('OutputFile1.txt', ASCII)

MANIPULATE
   IncomingData.RESET
   WHILE NOT IncomingData.EOF DO
      IncomingData.READNEXT
      IF val(AppData.GETVALUE('ID')) <= 3 THEN
         Output1.WRITE
      ELSE
         Database.WRITE
      ENDIF
   ENDWHILE
END
```

5.5 – Support for Maniplus

- Create and run your Maniplus setup in the apps
  - Design your own dialogs/screen and start surveys from it
  - App features like get list of surveys, installing surveys, download and upload of data will also be available in Maniplus
- Most of the Maniplus functionality will also available in the iOS and Android apps
  - Show data in a datagrid
  - Select records and change values
  - etc...

Using https

- In general:
  - Server management uses port 8031
  - Port 8033 is used by dataentry, resource, session, data, cati and audittrail roles
- Https can be enabled for both ports

Using https: set up order

– We recommend to set up things in a certain order:

1. Determine the servers that you are planning to use and their roles
   Determine management, data entry, web, resource, session, audittrail, cati and data server
2. On each server remove the unneccesary roles with server manager! This is important!
   - If your server for instance only needs to have web, dataentry and resource role
     - Open Command in Admin mode
     - Goto Blaise installation folder
     - Execute: ServerManager – role:dataentry,web,resource (specify just the roles needed)
     - Restart Blaise 5 Service

- 1 and 2 were needed in 5.3 and earlier versions. Is not needed anymore in 5.4

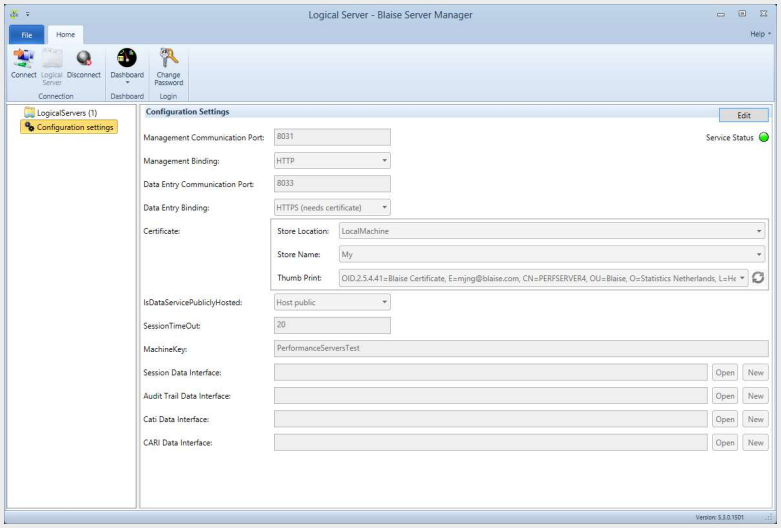
1. Install certificates on each server
2. On each server that uses https setup Logical Server and bind certificate to port 8033
3. Once everything has been prepared then you can create a server park

Using https: installing certificates

– Installing the certificates
- Server certificates
  - Certificate for the server that is going to use https on port 8033
  - Server certificates should be created by using the local/internal server names
  - Server certificates should be installed in the Local Machine – Personal store
- Root certificate
  - A certificate is created by a Certificate Authority. The name of the CA should be available in the Trusted Root Certification Authorities store
  - If you have created selfsigned certificates for your servers then you need to install the root certificate that has been used to generate the server certificates in the Local Machine Trusted Root Certification Authorities store of that server

- On each server where you want to use https on port 8033
  - Start ServerManager in Administrator mode
    - Goto Logical Server – Configuration Settings
    - Specify https for Data Entry Binding
    - Specify certificate information
    - Restart service

Local Server configuration

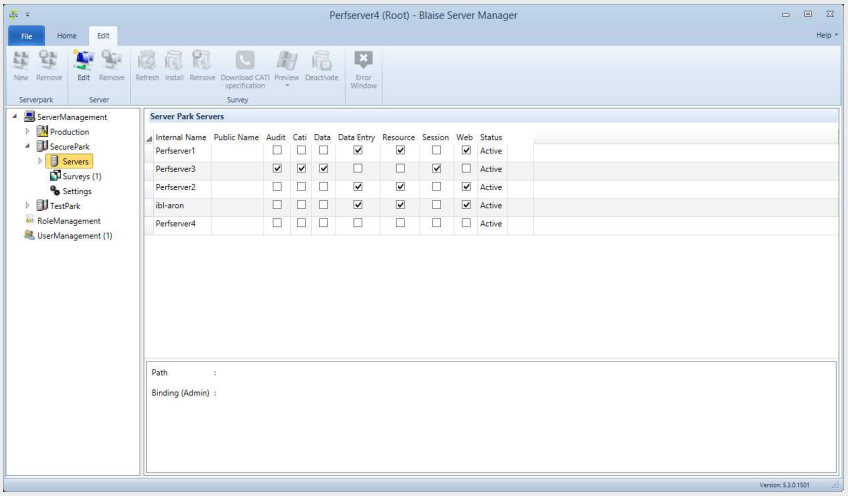


Using https: Setup server park

- Start Server Manager on Management Server
  - Define ServerPark; set Run Mode
  - Add servers and specify their roles
    - Use names as used with the certificates
    - Don't specify public names for the servers

- In DepApp connect with one of the Data Entry Servers directly (if possible) or with the firewall using a public name. Firewall needs to redirect to 1 of the Data Entry Servers by using their local names

Example Server Park



Perfserver4 = Management Server