CPI Price collection system documentation

Date: 21/1-2021

Version 1.0

Author: Anne Abelsæth ([aae@ssb.no](mailto:aae@ssb.no))

Contents

[1 Introduction 3](#_Toc62144103)

[2 The file Hierarchy 3](#_Toc62144104)

[3 The lookup folder 4](#_Toc62144105)

[3.1 The local offices folders 5](#_Toc62144106)

[4 The tablet applications 5](#_Toc62144107)

[4.1 The menu app 6](#_Toc62144108)

[4.1.1 Initiate new month 6](#_Toc62144109)

[4.1.2 Sync with HQ 6](#_Toc62144110)

[4.1.3 Collect prices 6](#_Toc62144111)

[4.2 The CPI\_app – main price collection app 7](#_Toc62144112)

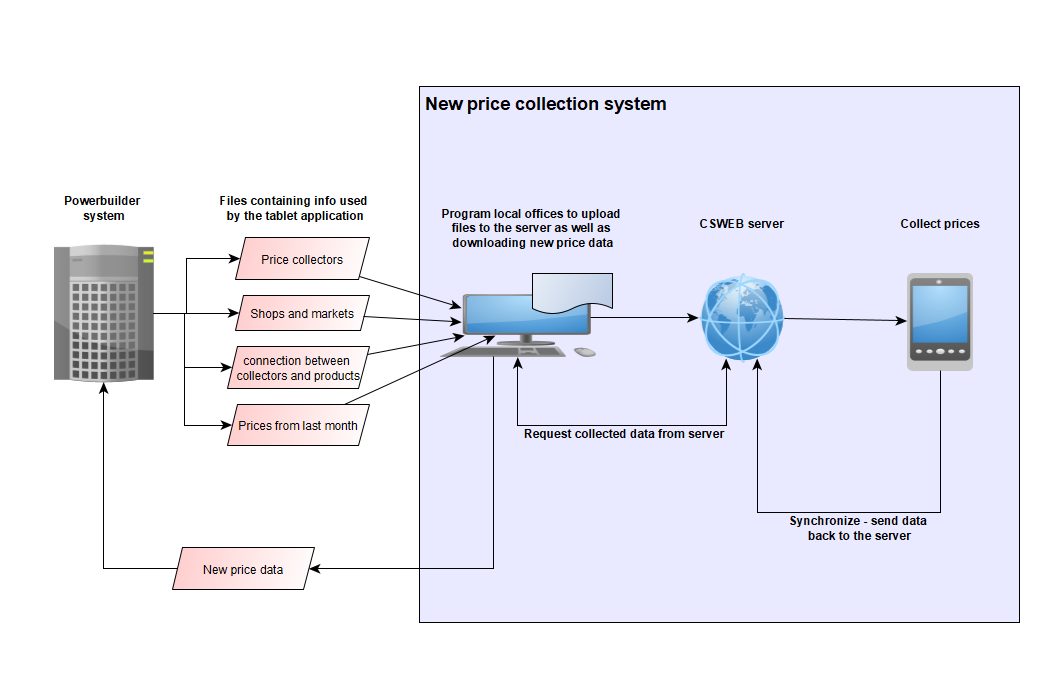
[4.2.1 At startup 7](#_Toc62144113)

[4.2.2 The PRODGROUP field 7](#_Toc62144114)

[4.2.3 The PRODUCT field 7](#_Toc62144115)

[4.2.4 The ENDQUEST and the DUMMY fields 7](#_Toc62144116)

# Introduction

The CPI price collection system is for collecting monthly prices. It only covers the data collection part. There are other systems covering data editing and further processing as well as administering the products and shops that are included in the survey. The system described here however, is closely linked with the existing powerbuilder system (see figure below)

The system is made in CSPro – a “Census and survey processing system” made by US Census Bureau and consists of 3 parts:

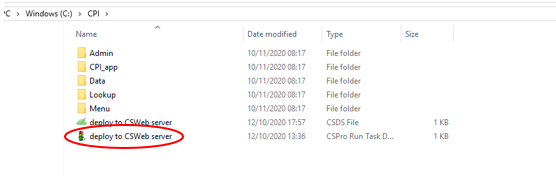
* The app on the tablets for collecting the prices
* CSWeb software running on IIS – the Windows’ web server. This is for synchronizing data and administer the tablets
* A CSPro app running on desktop computer of the local offices. This is used to
  + extract the collected data from the database on the server
  + upload the monthly files used by the tablets to navigate to the right shops and products and compare prices with last month.

# The file Hierarchy

The whole of the system contains the following folders. Not all of them are relevant to the different parts of the system:

* **admin folder**: This contains CSPro programs for the local offices to extract data from the server and to upload the txt files from the powerbuilder. Also, there are extra functionality for the CPI admin for changing the basket of products every 5th year.
* **CPI\_app folder:** the main price collecting app source files. This, together with the menu folder constitutes the part of the system that is installed on the tablets.
* **Documentation folder:** Contains this document and any other documentation related to the system.
* **LookUp folder:** Where all the lookup files for all local offices are stored. This also need to be present at the CSWeb server.
* **Menu folder:** The program to start up the price collection app on the tablets.

In addition there are some files

* **Deploy to CSWeb server files:** Script to redeploy the system to the CSWeb server. To be run if updates are done to the CPI\_app and/or Menu folders. Requires CSPro to be installed. To run it: double click on the file with the traffic light icon:  
    
  

# The lookup folder

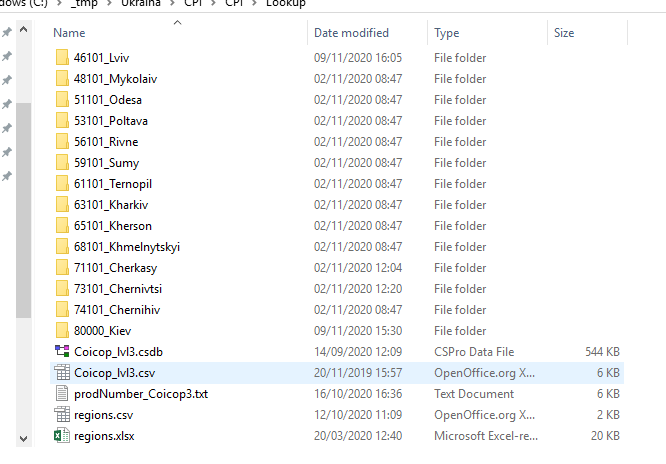
At the top of the lookup hierarchy, there are the following important files:

* **Coicop\_lvl3.csdb:** This is a CSPro database file containing 3-level coicop code with descriptions for grouping the products to make it easy for the price collectors to navigate in the shops.  
  If this list has to be changed when a new basket is introduced, please refer to the document “Change 3-lvl coicop code.docx” to see how.
* **prodNumber\_Coicop3.txt:** a file connecting the products number in use at SSSU with the COICOP numbers. This file must be changed every 5th year when there is a new basket of products.
* **Regions.csv:** the file containing oblasts and mistos where prices are collected. If a local office is added or removed, this file should be changed as well as the file hierarchy containing the names of the offices.
* **MistoNames.txt:** this file is used by the admin application and must be modified in the same situation as for the Regions file

(The other files in this folder can be ignored, as they are generated by CSPro)

## The local offices folders

There is one folder for each of the local offices in the survey. These folders contain all the files from the powerbuilder system and contains information about the regions of where to collect prices from, 3-level COICOP to group the products, names and ID of shops, price collectors etc.



Inside the local office folder are the following files (MM = month, YY = year):

* BLTA\_MMYY: prices collected last month from shops
* BLTB\_MMYY: Prices collected last month from markets
* BLTm\_MMYY: prices collected last month from shops for monitoring survey
* BLTmB\_MMYY: prices collected last month from markets for monitoring survey
* D\_perexid\_MMYY: info to collect price collectors to product numbers
* D\_reestr\_MMYY: Price collector IDs and names
* D\_rinok\_MMYY: info about markets and IDs of responsible price collectors
* D\_shop\_MMYY: Info about shops and IDs of responsible price collectors

These 8 files from powerbuilder have to be updated every month using the admin app at the local offices (see document “User manual for the Admin program.docx”).

(The distribution of these files are handled by the admin application, but if – for some reason they need to be looked at at the server, the path to these files is: C:\inetpub\wwwroot\csweb\files\CPI\Lookup)

# The tablet applications

The tablet application consists of two parts: A menu application, and the “CPI\_app” which handles the actual price collection.

## The menu app

This app is the one that the price collector starts up on the tablet, and it does most of the work preparing the files and prices to collect for the current month.

The code is self-documented, but here is a brief overview:

### Initiate new month

Prerequisites: The powerbuilder files must have been uploaded to the server. See document “User manual for the Admin program.docx”

First the system asks for location (Oblast and Misto respectively) and interviewer info. This is mostly to ensure that the system is consistent even if the tablet is moved from one area to another or if the interviewer swaps tablets. *But it is important that the tablet is personal/belongs to the same price collector for the whole month!*

Most of the logic takes place in the postproc of M\_ENUMERATOR\_NO because at this point enough information is collected to be able to create personalised files with the products of which the price collector is responsible for:

1. A list[[1]](#footnote-1) of the products of which the price collector is responsible for is extracted from the d\_perexid\_MMYY file.
2. Lists of the shops and markets for this price collector is also extracted from d\_shops\_MMYY
3. The dataset PRODUCTS (in CSPro database format) is created. This is used to populate the roster of products for a given shop with last months info with the format and this is the dataset that it actually used by the CPI\_APP application.

The “initiate new month” choice must be chosen at least once before the price collection for a given month start, it is, however, no problem if it is chosen more than once, and it MUST be chosen again if for some reason any of the lookup files has been changed.

### Sync with HQ

Sends the data file to the server and downloads all the lookup files. (The same is actually also done in the “initiate new month” option, but here the main goal is to receive lookup files)

### Collect prices

Before prices can be collected, the price collector has to choose what shop to collect prices for. When that is done, the system checks whether the price collection already has started for this shop and use this info to assemble the pff file[[2]](#footnote-2). Once that is done, it launches the CPI\_APP to start the actual price collection.

## The CPI\_app – main price collection app

After the shop is chosen in the menu app, the CPI\_App is launched (and it will be launched every time a shop is chosen)

### At startup

Information from the pff file is retrieved and if not already done: the roster containing the product is filled. Also, if the shop has been started/is partially saved, the menu for choosing 3-level COICOP codes is set.

All of the fields on the cover page are automatically set – either by populating them with information from the pff file, or using the system date and time information

### The PRODGROUP field

The app loops through the PRODUCTS dataset generated in the menu app to find all 3-level COICOP codes relevant for the chosen shop. These will be added to the options to choose from for the field.

It also loops through the elements of the field one more time to check whether price collection for each price group has been finished, and if so adds the word “Finished” in front of it, so that the price collector does not have to open it to check.

### The PRODUCT field

This field is looped through again and again until all prices are entered or “quit” is chosen.

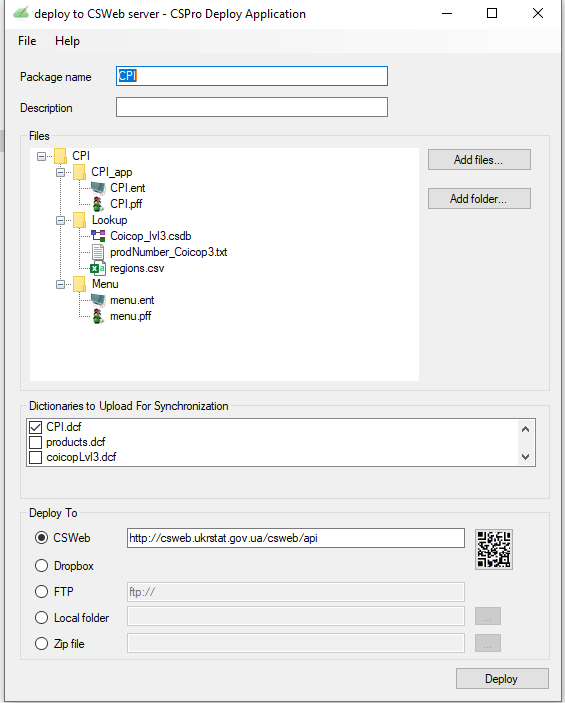
In the preproc, all the products for the 3-lvl coicop are entered into the dropdown of the field while in the postproc the actual entry and checking of price is done. Once it is acceptable with price either same as last month or with a reason for the price change, it is saved in the CPI\_DICT Cspro database and will be uploaded next time “sync” is pressed.

### The ENDQUEST and the DUMMY fields

For CSPro to save the record as complete – not partially saved, the focus must move and end up in the last field – DUMMY - in the form. But to know whether it is the end of the record the ENDQUEST variable is given value 0 if the user has chosen “quit” and the record is complete (ie all prices for the shop has been collected). Otherwise it gets the value 1, and the record will be saved as partial.

# Deployment to server

If changes are done to the application, it must be redeployed to the server. This is done using the scripts in the CPI folder:

* **Deploy to CSWeb server.pff:** The file to run the deploy script. Double click on this to deploy
* **Deploy to CSWeb server csds:** this is the file containing the details of files to deploy, and can be opened by double clicking on it (given that CSPro is installed on the computer)  
    
    
    
  + In the Files area, the files belonging to the deployment is specified. Normally, all the lookup files should be included here, but as this application has extremely many lookup files changing from month to month, and as there is a full hierarchy of folders, only the common lookup files are deployed, and the rest is uploaded using the admin application.
  + “Dictionaries to Upload for Synchronization”: Only the CPI.dcf is needed, as the other dictionaries are just temporary to use at the tablet.

There are also deployment files to deploy to folder structure, but this will probably never be used.

1. In this connection, a list is a programming structure to hold data – not a list as understood in normal English [↑](#footnote-ref-1)
2. A pff file is kind of a configuration file for CSEntry telling it what files to use and to shuffle information between programs. It also tells CSEntry about startup mode: add shop or modify shop [↑](#footnote-ref-2)