TWINNING CONTRACT

BA 15 IPA SR 01 17

Support to the reform of the statistics system in Bosnia and Herzegovina

MISSION REPORT

Activity 2.C.2:
IT application development for producer prices II

Component 2: Business Statistics

Mission carried out by
Søren Netterstrøm, Statistics Denmark

29 October – 02-November 2018

Version: Final

Expert contact information

Søren Netterstrøm
Statistics Denmark
Copenhagen, Denmark
Email: sne@viv.dk

Table of contents

List of Abbreviations

BHAS	Agency for Statistics of Bosnia and Herzegovina
BiH	Bosnia and Herzegovina
CBBH	Central Bank of Bosnia and Herzegovina
EC	European Commission
EU	European Union
FBiH	Federation of Bosnia and Herzegovina
FIS	Institute for Statistics of Federation of Bosnia and Herzegovina
MS	EU Member State
RSIS	Institute for Statistics of Republika Srpska
RTA	Resident Twinning Adviser
ToR	Terms of Reference

1. General comments

This mission report was prepared within the EU Twinning Project " Support to the reform of the statistics system in Bosnia and Herzegovina. It was the second mission to be devoted to the development of an IT application for price indexes (CCPI and SPPI) within Component 2: Business Statistics of the project.
The purposes of the mission were:

- Review and further development of the application specification
- Further development of the application

The consultant would like to express his/her thanks to all officials and individuals met for the kind support and valuable information which he/she received during the stay in Bosnia-Herzegovina and which highly facilitated the work of the consultant.

This views and observations stated in this report are those of the consultant and do not necessarily correspond to the views of EU, BHAS, FIS, RSIS, CBBH, Statistics Denmark, INSEE, Statistics Finland and Croatian Bureau of Statistics.

2. Assessment and results

The overall goal of this component is to develop an application to be used by both CPPI and SPPI as discussed in the first mission of the component.

Based on the specifications made during the first to missions, the consultant was able, during the first three days to finalize the development of the application with a few outstanding issues, to be further elaborated in the final report.

During the last mission it was agreed that BC should carefully reads and comment on the specification of the application. However, as the consultant did not get any feedback, the application is built on the specifications as they stand. This may lead to some changes or corrections to meet all needs of the users once the application is in test phase or production.

During the short time available for meeting with BC, the application was demonstrated with special focus on the data entry part, that was not covered during the previous mission.

There was a brief discussion of the handling of non-response units. This will eventually lead to a small change in the application to distinguish between that a questionnaire is detected as non-response and that is has been imputed. There was also a very brief discussion about the possibility of introducing a simple method for automatic imputation, using the general development of those units that has responded. Needs further investigation.
Another issue that turned up is, that SPPI is using average of 2015 as BASE (=100) rather than 4th quarter of 2014. The system will be expanded to meet this demand (and other demands for rebasing in the future).

There was a brief discussion about how the system can be implemented including a discussion of the security level. It was agreed that during the final mission, one of the main tasks should be to make a first implementation of the system for one or two surveys in close cooperation between the consultant and IT staff at BHAS, FIS and RSIS. The manual will also be expanded in relation to this issue.

3. Outstanding issues

The application is ready to be tested, however there are still several outstanding issues to be solved.

The following issues is known, but others may come as a result of the testing to take place.

1. Version of indexes, when elementary indexes is added or removed.

2. Terminate response unit during data entry

3. Handling of non-response

4. Creating and using shadow indexes

5. Test that data exists for all elementary indexes during check

The necessary changes to the application will be made before or during the next mission,

4. Work to be done before next mission

.

Before the next mission the BC Counterpart should **carefully** read the specifications, to be included in the final report, when available, to ensure that it meets the demands of both the entities and BHAS as well as both surveys. Any comments should be sent in written form to the consultant the latest 1 week before next mission.

The consultant will within a week provide:
A revised version of the application
A database with data from FIS regarding SPPI, based on the data received in September 2018
A database with data from FIS regarding CPPI, based on the data received on the first mission.
A database with data from RS regarding CPPI, based on data received at the end of this mission.
A database with data from DiBR regarding CPPI, based on data received during the study visit to Copenhagen
Databases for BHAS regarding CPPI and SPPI, with data imported.

If data regarding SPPI are received from RS, these will also be processed and returned.

The BC counterpart should also test the delivered application. This includes but are not limited to testing the data entry part, check that calculations are giving expected results and that the method for exchange of data is working properly.

In case of problems, the consultant should be contacted and as far as possible, the issues raised will be solved shortly.

5. Topics for the next mission

The next mission will be carried out in week 50 (10.12-14-12) in Sarajevo.

The main topic of that mission will be to finally install the system, solve any outstanding issues. IT staff that will be responsible for local support will be given a technical presentation of the system to help them to better understand how the system is constructed.

The need for one or two mission in the spring of 2019 has been discussed, but no final decisions are made on this issue.

Annex 1. Terms of Reference

<div style="border:1px solid black; padding:1em;">

Terms of Reference

EU Twinning Project BA 15 IPA ST 01 17

Component 2: Business Statistics
29.October – 02.November 2018
Hosting institution: RSIS, Veljka Mladjenovića 12d, Banja Luka

</div>

Activity 2.C.2: IT application development for producer prices II

1. **Purpose of the activity**
   - Follow up from the previous mission:
     - The specification, annex 3, examined and open questions considered as a preparation for further work
     - Preparation of a test scenario
   - Discussion and revision of the specification document
   - Further development of the IT application
   - Clarification about how the database should be installed (databases, security etc,) Presentation of the specification annex 3. (BC experts).

2. **Expected output of the activity**
   - IT application for producer price indices developed

Agenda

Location: Banja Luka, Veljka Mladjenovica 12d

Day	Place	Event	Purpose / Details
29 October Monday	RSIS	Ms expert	Further development of the IT application (MS expert)
30 October Tuesday	RSIS	Ms expert	Further development of the IT application (MS expert)
31 October Wednesday	RSIS	MS expert	Further development of the IT application (MS expert)
01 November Thursday	RSIS	Meeting with BC Experts	Discussion about IT application (MS and BC expert)
02 November Friday	RSIS	Meeting with BC Experts	Agreement on summary mission report Identification of work to be done until next activity. Input to ToR for the next activity Conclusions and next steps

Annex 2. Persons met

Institute for Statistics of Republika Srpska (RSIS)

- Jelena Glamočika, RSIS
- Aleksandra Đonlaga, RSIS
- Pero Kazanovic, RSIS
- Danica Babic, RSIS

Institute for Statistics of Federation of BiH (FIS)

- Edina Dulic, FIS
- Merima Hadzalic, FIS
- Ahmet Fazlic, FIS, IT

Agency for Statistics of BiH (BHAS)

- Fahir Kanlić, BHAS
- Dženita Babić, BHAS
- Alen Bajramović, BHAS

Twinning Project Administration

- Katja Møller Hjelvang, RTA
- Đemka Šahinpašić, RTA Assistant

Annex 2. Specifications of the IT system

Application for CPPI and SPPI in BiH

The purpose of this paper is to describe an application to be used by BHAS, FIS and RSIS to compile the CPPI and SPPI.

The application should cover the data entry and data validation processes and the calculation of elementary indexes and aggregated indexes.

It should be possible to extract predefined reports from the system as well as defining new reports.

Data are collected by RSIS and FIS for the two entities and by BHAS for District of Brčko. RSIS and FIS may compile indexes for their entity. Once data entry has been completed (and validated), data on price changes are sent to BHAS, that will produce indexes for the whole of BiH.

For CPPI and SPPI data are always collected in a way where for each item the price for both previous period and current period are collected. The system should take advantage of this.

The system will be build using Visual Basic (Visual Studio 2017) and can use either MS Access or SQL Server as databases.

For each survey there will be a separate database. For CPPI there will be one database for the quarterly survey and another for yearly surveys (potentially one for each). The structure for SPPI is not completed, but it will be similar.

Further there will be local databases, so for each database there will be one in RSIS, one in FIS and one in BHAS to cover District of Brčko. Finally BHAS will have a special data base to hold the data for all of BiH.

Before the system can be used, a database must be created, and some basic tables and other information must be filled. Appendix 2 contains information needed to do this.
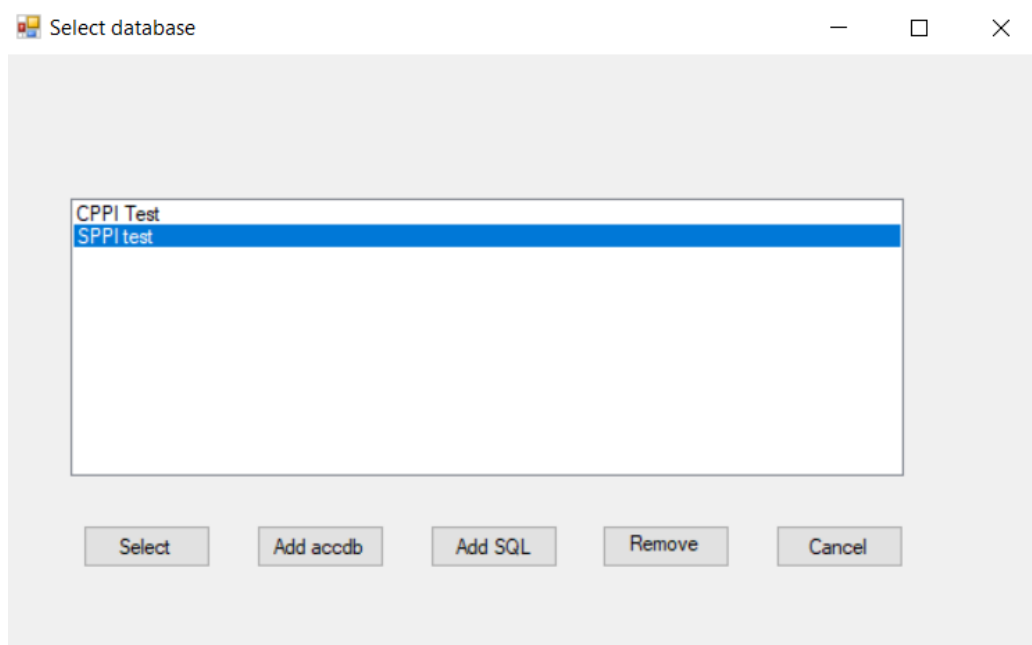
Indholdsfortegnelse

Starting the system, logon

Select database

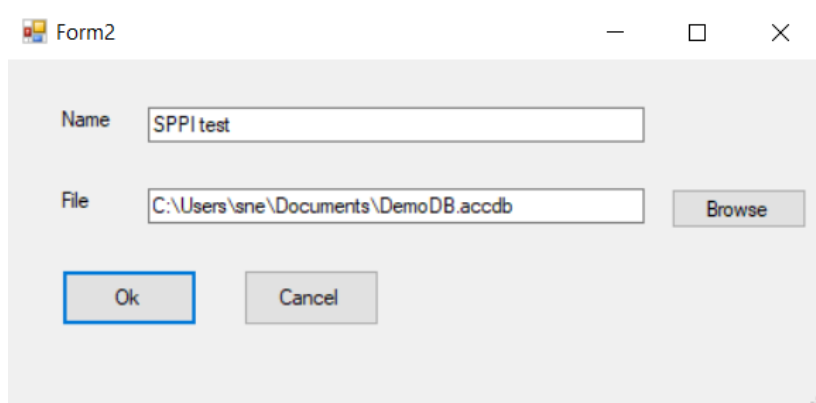When the user starts the application, the first screen will prompt the user to select a database.

The user then selects the database to work with.

The use can also add a new database (accdb or SQL) or remove a database from the list.

Cancel terminates the application.

To add a new database the user fill this

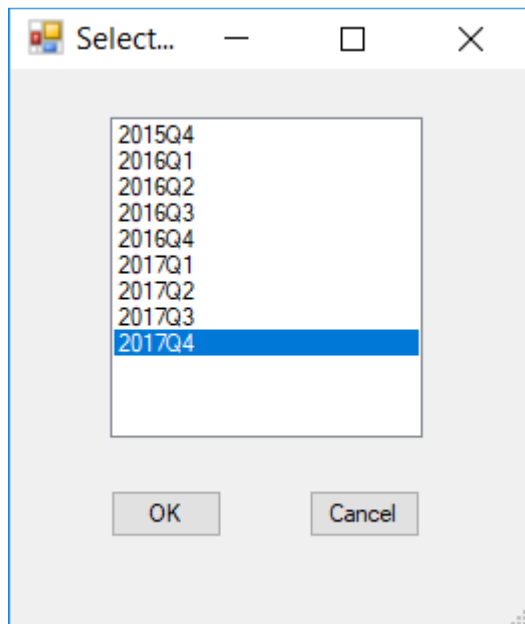The name is the one presented on the first screen and file the full path to the database.

Similar for SQL server.

The system will test, that the user (using the name for Windows Login) has the right to access the database. If not the user is informed and the system terminates.
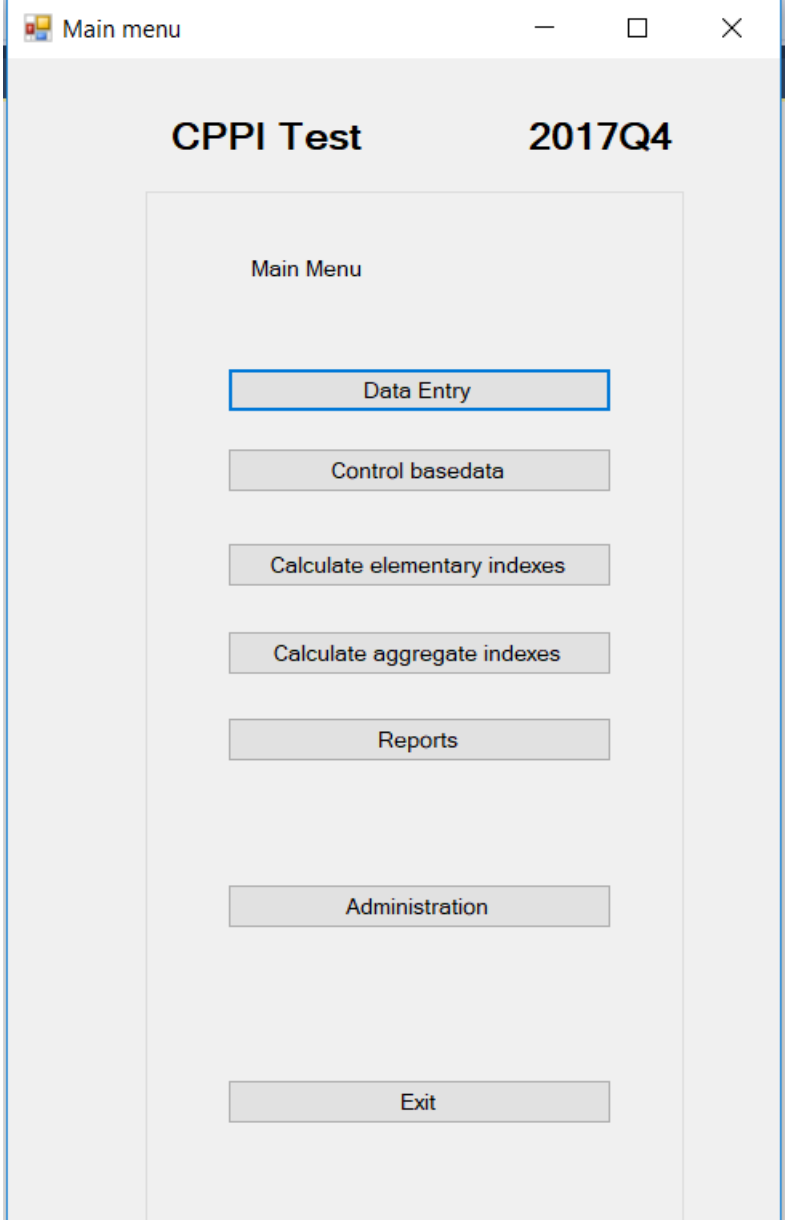
Select Period
Next step is to select the period to work with.

Standard is the last period in the system, but any period can be selected.

Main Menu

When database and period has been selected. The main menu is displayed.

Data Entry

Used to enter data from a questionnaire or edit/review data already entered.

Control Base Data

This will give a status on the progress of data entry. If data entry is completed the user may request reports to further validate data. These reports will be created as Excel files.

The actual number and content of these reports has to be decided.

One option is to use the report feature (see later) to mark one or more reports to be produced to perform the final validation of data.

15

Calculate Elementary Indexes

Once data entry is completed, the next step is to calculate the elementary indexes.

This will produce a report (Excel file) listing all elementary indexes for both current and all previous periods.

There will also be a report giving the indexed as month to month development.

This should be used to further examine that data are valid before proceeding with next step.

Calculate indexes

Next and final step is to calculate all aggregated level indexes.

This will produce a report (Excel file) listing all indexes for both current and all previous periods.

Reports

The system will contain standard reports and the use may add further reports to the system (see below).

When selecting reports, a dialog will show up, where one or more reports can be selected.

Administration

This will open a new menu used to perform various administrative function like setting up a new quarter, maintaining weights etc.

Exit

To terminate the application.

Data Entry

When selecting Data Entry, the user must first select the response unit. The user may optionally select another period, but in that case, the user can not change any data.

The window for data entry is then displayed.

If the unit has been submitting data in the previous period, the display is filled with data from that period. Product, sub product, product name, measurement method. Prices (previous and current) are copied from current price of previous period. Reason of change is set to 0 (no change) and comments are excluded.

Product names can be very long. In the overall display only the first 40 characters are displayed.

If there is a change in a price, the user may enter the new price and should also set reason for change.

If the reported price of current quarter does not match what is prefilled, it should be checked if a new sub product should be introduced (or if this should just be considered a minor reporting error).

The user should however not have the option to change previous price unless it is 0 or blank (in the case of the very first period or a new product has been added).

The user may (but activating the button) add comments as needed.

If there is a need to verify or modify the product name, the user should select to expand the row. A window will pop up where there is enough room to enter even very large text.

If a product is not reported for the actual quarter, the user should **terminate** the corresponding row. When doing so, the user must select if the termination is permanent or temporary. If permanent, the item should not be included in the next questionnaire, but if temporary is remains on the questionnaire.

If new products are added, the use select **add** product, A new window is displayed where the user must enter productid (from a list), product name, measurement unit, prices for previous and current period and reason for change (if any). When done, a new line will be displayed on the main window. Sub Product is created automatic by assigning the next free number for the product ID within the response unit.

Product replacement is done by terminating (permanently) the old product and then adding a new product.

Once the user has completed editing, the user must press the button **check**. This will run the tests described below. In case of absolute errors these must be corrected before data can be finally saved. In case of warning the user chose to ignore warning (after checking the reason).

If there are no errors and warning has been ignored (or there are no warnings), the user now has the option to **save** data.

In some cases, it may not be possible to finalize a questionnaire because further information must be obtained by contacting the user or other means. In that case the process can be **put on hold**. Data that has been entered are saved. The questionnaire will have an indicator it has been held. The user may enter a comment (note) for the reason of this, that will be show when the questionnaire is selected for further processing.

Change in quality

If the reason for change of price is 5 (other reason), could this indicate a shift in quality. In the case the previous price reported is different from last reported current prices (prefilled price) this may be the case.

This could be handled ether by allowing to insert the new previous price in this case or by creating a new sub product replacing the existing (but having the same text).

However, if price indexes are calculated using the ratio of the price in current period to last quarter of previous your (or any other base period not being the previous period), this method can not be applied, ads there may not be a price in the base period. So a consequence of this is, that between two base periods, there can not be any change in the list of products (and subproducts) for any response units. Such changes can only take place when the base period is equal to the previous period.

See also discussion of index calculation.

Checks performed on a questionnaire.

Absolute errors:

Price change (or no price change) should match reason for change.
Tech note: The system will contain a table of reason codes and for each reason code indicate if price should be unchanged, changed (increased or decreased). This allows for reason codes to be different for different surveys or to be changed over time.

Each item must have a price for previous and current period.

Warnings

Change of price should not be outside a given interval (in %) (by product)
Tech note: In the table of products, the allowed interval of changes is given for each product. However, if no interval is given a standard interval (set in options will be used). Confidential Interval

Outliers are detected by comparing current price with average for previous period
Tech note: In the table of products, the allowed interval in % is given for each product. However, if no interval is given a standard interval (set in options will be used).

Inliers mean that data has not changed for a given period.
Tech note: The number of periods to look at are set in options.
The check can either be performed on a sub product by sub product base or the whole of the questionnaire. In the latter case this means, that there are no changes to sub products or any prices in the selected period for the response unit. Method is determined in options.

Error report

If there a no errors or warnings, the user is just informed in a message box.

If there are errors an error report is giving the reason for each error or warning and a message with the number of errors and warnings are displayed.

Handling no response

If a response unit has not submitted any questionnaire or not filled in the questionnaire, the response unit may either be terminated (in case the company has ceased to exist or does not perform any relevant activity) or marked as a nonresponse for the current period.

If the unit is terminated, the user can terminate the unit from the list of response units. The Questionnaire and all prices for the current period is removed and the unit is marked as terminated. When creating the next period (see later), the unit will not be part of that process.

If the unit is nonresponse, the questionnaire will be marked as no-processed non-response. This again is done from the list of response units.

When all units that has a respond is processed, units marked as nonresponse should be handled.

It is to be further discussed, how this should be done. It could be done unit by unit, estimating a new price and saving. Some help could be obtained by calculating a new value based on the average relative change of prices in the same product (or service group) that are not imputed. Another option could be a more automated process, using a similar method.

Regardless of what is decided, when processed the questionnaire should be marked as process non-response.

Calculation of elementary Indexes

Prerequisites

Elementary indexes can only be calculated when data entry is completed, that is for all response units (questionnaires for current period) the status must be completed, or non-response processed. Note that there is no questionnaire for terminated units.

Weights for products by response units must be checked.

If there is at least one weight for a given product, then there must be a weight for all response units reporting that product and there must not be any weight for response units that do not report the product.

If this requirement is not fulfilled, a list of products with errors are produced and the user must use the tools under administration to enter/correct the weights for those products.

If the base period is not equal the previous period, each price (defined by response unit, product and subproduct) must have a price for the base period (that is in fact a previous price in the period following the base period).

Method

The index calculation is done 4 steps.

1. Prices are summarized for each response unit using geometric average.
   tech note: table RUPrices (`Period, ResponseUnitID, ProductID, PrevPrice, CurrPrice`)
2. The ratio between current price and previous price is calculated
   tech note: table RURelative (`Period, ResponseUnitID, ProductID, Change`)
3. Using weighted geomean (Jevson index) or if no weights are available geomean relative changes for each product is calculated
   Tech note: `EARelatives(Period, ProductID, Change)`
4. Finally, the chained index is calculated as (index for previous period) * (change in current period).

Note: In the example above, changes are the changes between previous price and current price as reported in the current period. In step 4 this is reflected by using chained indexes.

See also Appendix 4 Calculation methods for further information on different options.

Output

When indexes are calculated, reports are created.

See further in appendix 1 for how to create reports and select then to be created at this stage.

The following reports has so far been identified

1. table of all elementary indexes for all periods in the system.

2. a table with number of reports units, number of individual prices (total), and nonresponse/terminations

3. for each index the number of individual prices, number of response units contributing, min(current price) and max(current price) at individual price level

There will be a special mark in the case there are no prices or less than 3 reporting units.

4. for each index number of individual prices distributed by reason for change

The user may create any other needed reports using the report feature.

Handling elementary indexes that can no calculated due to missing input.

In the rare case that there is no input for a elementary index, then it must be determined if this is a temporary problem or if the index cannot be calculated in the future.

In the first case (temporary), it should be possible to insert an imputed value for the index, the user must determine the method and value to be inserted. This should be marked, as it will then be necessary to re-evaluate when data again becomes available (comparing current price with last known price).

If this is not temporary, a substitution index should be selected. The index will then be calculated using the ratio of chance for that index, both for the current period and future periods.

The item will then no longer be active. It will not possible to enter new prices for this index.

Table 2 above should include information about the number of indexes that are calculated this way.

When calculating indexes, it will be checked, that there is at least one value for each index to be compiled in the reported prices. If not a warning should be issued.

Calculation of aggregate indexes

Prerequisites

Elementary indexes have been calculated (and verified).

That there are no errors in the index structure definition.

Method

Aggregated indexes are calculated as Laspeyere indexed relative to a base period.

Tech note: The system will have a table of periods numbering the periods from 1 to n.
For each period there will be a text (3rd Quarter 2015) and information about the base period to be used for index calculation of aggregate indexes.

For each base period there is a separate set of weights.

Tech note: If an index is based on other aggregate indexes, then the definition is expanded to use the base for these aggregates, until the expanded definition only contains elementary indexes. Consequently, the order in which the indexes are calculated does not matter.

Output

Like for elementary indexes, some reports can be defined to be created when aggregate indexes are created.

This includes

1. table of aggregate indexes for all periods in the system.

Reports

The system contains a set of standard reports that the user may produce as needed.

A report is a query into the database that is presented as an excel spreadsheet.

When calculating elementary or aggregate level indexes, some reports are automatically produced, but the same report, and others, may be produced from here.

When selecting Reports, the system displays a list of the reports defined in the system.

Just select the report you would like and select **Run**

A report may require one to five parameters, like Period, Response Unit ID, or Product ID. In that case a window pops up requesting the information needed.

Editing and creating new reports

The user may edit an existing report or add a new report to the system. How this is done is explained in appendix 1.

Administration

Administration covers tasks for maintenance of the system.

Administration is only available for users logged on as administrators.

Set Options

Options are group into Data control, Calculation Methods, Rebase options and type of DB.
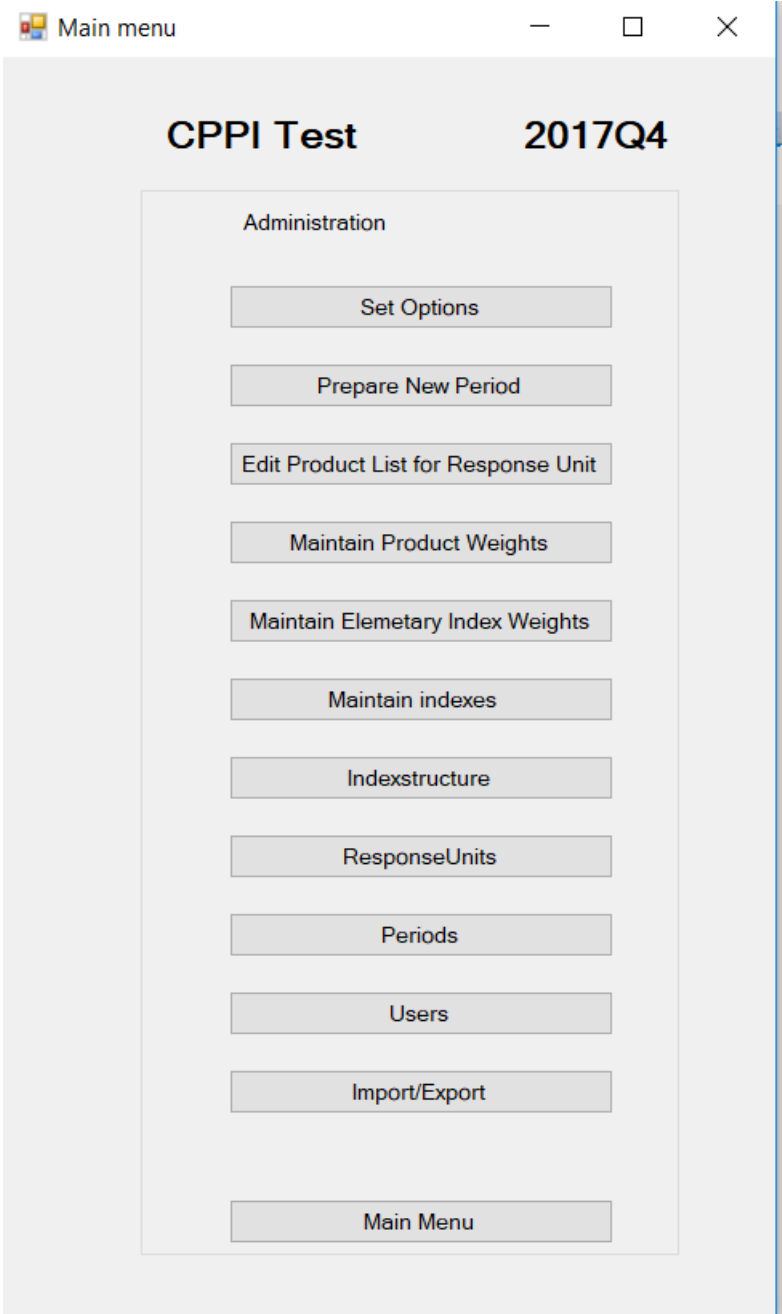

Prices changes are setting a default interval for range of relative change between previous and current price.

Outliers are testing, that the price of current period if not outside a giving range compared to the average price of the same product as reported in the previous period.

Inlier detection set the rule for inlier detection, if any. Either by price or on the full questionnaire. The number of periods set the limit of the test, that is how many periods to test for no change.

This is used to set the options for calculations, see also appendix 4

When rebasing the indexed, this can be either a single period or the average of periods (a year).

Type of database is to distinguish between the databases uses for entities (including data entry, and the database used by BHaS to compile data for all BiH.

Prepare New Period

This function will make the system ready to process a new period.

Periods are automatically numbered. The user must add a title or name like 2017Q1 or January 2018.

The user must also select the period that will be used as base when creating Laspeyere indexes. For each base period used, there must be a set of weights. This is checked when indexes are created.

Questionnaires may be different for each user, as they are using the sub-products the user was responding the previous period.

The table to hold individual responses are prepared by making a copy of the responses from last quarter. Response Unit ID, Product ID, Sub Product ID and product name, pricing method and measurement unit are copied from previous period. Previous and current price are taken from current price of previous period. Reason for changed is set to No Change (0). Comments will be NULL.

For each Response Unit, status of Questionnaire is set to Not processed.

The product weights by response unit copied assuming there will be no changes, but they may have to be recalculated if there are non-responses or new response units.

Response units that is marked as terminated is not processed.

Maintain Product List for Response Units

When starting a new period, there may be a need to maintain the list of products each response units, i.e. the copy of individual prices created when initializing the period.

This is most important, if this data is used to produce individual questionnaires.

If a new response unit is introduced, it may be needed to select the products/services the unit is supposed to the able to respond to.

If products (and elementary aggregates) are added or removed (or assigned a shadow product), then the product should be removed/replaced or added as appropriate.

Maintain Product Weights

The weights used in step 3 of the elementary index calculation is for each product given the weight for each response unit contributing to that product.

These weights are only valid for 1 period.

The sum of the weights for a given product must be 1.

To maintain these weights first bring up a list of all products that have weights associated and then select the product (or add a product).

For the product selected, a list of all Response Units contributing is presented with the base of weight (turnover) and the actual weight and all Response Units having a weight for the product even if they do not contribute in this period.

For units not contributing, verify that this is correct and then remove unit from the list.

For new units add the base (turnover) and potentially correct base for old units.

When bases are correct, press 'Calculate' the new weights and then save.

Maintain Elementary Index Weights

Each elementary index must have a weight that are used when creating aggregate indexes.

These weights are established for each (weight) base period and should not be changed once established. If they are changed, all aggregate indexes from the base period and up should be recalculated.

When producing aggregate indexes, these are always produced from elementary indexes, even if they may be defined as aggregation of other aggregates. For this reason, there is no need to provide weights for aggregates.

The base for indexes may be turnover or any other relevant measure.

It is normal to change the indexes to a percentage (with 2 decimal places). This calculation automatic. As Laspeyere does not require the sum of indexes to be 1 (or 100 %) there will be no attempts to make sure the sum of indexes is 100%, due to rounding errors they may be slightly above or below.

Maintain indexes
All elementary indexes are related to a product (service etc.), so the list of elementary indexes include also data on products.

Aggregates indexes are compiled from elementary indexes but has no relation to a single product. See Index Structure below.

For each index the system has the following information

Index Code (= Product ID for elementary indexes)

Index Type (Base or Aggregate)

Short title (max 100 characters), used when listing indexes in Excel Sheets

Long Title (no max), detailed description, used an initial text for questionnaires

Measurement unit (Elementary index only)

Pricing method (Elementary index only)

Confidence interval, low and high (base unit only), optional

Confidence interval, if present are use for validation during data entry. It is given as % and measured against the geometric index if prices of previous period.

In a given base period, it is no possible to add or remove Indexes, but aggregate level indexes may be added or deleted. In that case the index-structure should be revised as well.

Also refer to the discussion in appendix 2 regarding versions of indexes.

Index Structure

Aggregated Indexes are created base on elementary indexes or other aggregates.

Typically, the are created in a hierarchy, but they may be constructed any way. The only requirement is that the same elementary index will only be used once when compiling an aggregate.

Indexes are examined and created from this form

The first box list all aggregate indexes. The Second box list the indexes that are aggregated (sources) to produce the index selected in the first box.

The last Box show all other indexes (potential)

Use <== to move an index from potential to sources or ==> to move and index from sources to potential. One the structure has been done, press save.

The system checks, that the resulting definition is not causing any overlap.

Response Units

Response Units are initially taken from SBR. When setting up the system for the first use,

Response units should be loaded from an Excel sheet or other electronic source, see appendix 2.

It is assumed, that the same response units are kept for a long time, but if there is a bulk change, the methods in appendix 2 may be used to transfer from an external source.

The purpose of this item is to maintain the list. This includes changes in address, phone, fax or contact person as well of occasionally adding a new response unit if needed.

Response Units can not be deleted or removed (except if they have never reported anything), but they can be terminated. In the latter case they will not be part of any future survey.

Normally termination is done in connection with data entry, see above

Periods

Periods are created in the process of prepare new period.

The purpose of this is just to allow to change the name of the period or the base period for index calculation if needed.

Base period can only be changed to a period after the current base period.

Status and locked

For each period a survey go through the following states

0 Prepared (and data entry in progress)

1 Data Entry completed (and verified)

2 Bases indexes have been calculated

3 Aggregated indexes have been calculated

If needed, status can be changed

When a new period is initialized, the system will put a lock on all previous periods.

When a period is locked, it is not possible to edit individual prices or recalculate indexes or change any weights. You can still extract reports and inspect data. If needed the lock can be removed.

Users

When users are defined, only users in the list of users have access to the database.

An administrator is a user who have access to the functions describer under administration. Ensure that the system always have at least one administrator.

The user is identified using the name from login to Windows.

When the first user is added, it will become the current user as administrator.

If there are no administrators, the system will assume all users are administrators.

Import/Export

The price-indexes for BiH are produced in cooperation between the statistical institutes BHaS, FIS and RSIS.

For the same price index there should be one database in FIS, one database in RSIS and two databases in BHaS, one for District of Brčko and one to summarize all data for BiH.

It must be possible for the entities to send data to BHaS when a period has been finalized. BHaS can then compile the result for the whole of BiH.

It is important, that the databases are synchronized in relation to the index structure, at least for elementary indexes and are referring to the same periods.

Import/Export is used for these exchanges. An Export will create an Access database holding the information to be exchanged, then Import is used to transfer the data to the target database.

Export will export :

Response Units (for all periods) giving a random identifier (1st digit dependes on the entity) and information if the unit is terminated and reason for termination

ResonseUnitID	Terminated	ReasonForTerminate	
1705546	FALSK		
1533423	FALSK		
1579518	FALSK		

Individual prices for one period

ResponseunitID	Period	ProductID	SubProduct	PrevPrice	CurrPrice	ReasonForChange	
1373535	1	49.41.06	2	1	1	0	
1373535	1	49.41.06	1	1	1	0	
1829800	1	49.41.08	1	1	0.974359	0	

Product weights (inckuding turnover)

ResponseUnitID	Period	ProductID	Weight	Turnover	
1961952	1	49.41.02	0.0176	27.73	
1790479	1	49.41.03	1	1269	

Turnover is necessary in order to be able to calculate weights on BiH level.

Questionnaire Status

ResponseUnitID	Period	Status
1705546	1	2
1533423	1	2

The texts associated with each sub product

ResponseU	ProductID	SubProduc	SPText	Measurem	PricingMet	ReasonForNew
1705546	49.41.01	1	Ostrozebbe (Be) - Breza; 21	tura	UG	
1705546	49.41.06	1	Mausteten (De) - Sarajevo;	tura	UG	

Data as exported to BHAS will have a random identifier (that will be the same over time) replacing the ResponseUnitID. Previous and Current Price will given as a ratio. The exact mechanism has still to be agreed.

Issues related to the transfer of data between entities and BHAS.

These issues has not been examined in detail so far. Potentiel problems could be, that for one or more products, there is no data available from 1 of the entities. How to handle this case?

Appendix 1 Creating Reports

Reports can be created (or edited) by the user.

A report is just an SQL-query, that delivers rows of data that are the inserted in a spreadsheet, with appropriate title and headings.

The user may of cause make such a query using the data model as described in appendix 5. It may however be better to use a tool like Microsoft Access to build the query and then copy (an modify) the resulting SQL statement as needed.

  The **ID** can be anything but must be unique. It is used as the sort order when showing the list of reports.

The **Title** is what is listed

The **Heading** will be written in the first row of the report. It may include parameters, see below. If omitted Title is used

SQL is the actual SQL statement. It may include parameters

1 to 5 **parameters** may be defined. Each parameter has a name, a type and a title

Name is used to identify the parameter. Before using the heading or SQL query, the system will replace @Name with the value given for that parameter.

Type identifies how the value should be obtained. Basically, there are strings and integers. You may check with the data model or with the SQL statement created using Access.

Beside string and integers some common types are defined: Period(ID), Response Unit(ID) and Product(ID), Index(ID). For these types, the system will use a drop-down combo box listing Periods, Response Units, Product or Indexes respectively, so the user will have to select from that list.

See the example above where period is of type period and the inserted both in heading and SQL

A report may have a mark telling that it should be run when new elementary/aggregate indexes are calculated. Such reports may have only 1 parameter of type Period or no parameter

Appendix 2 Setting up the system for the first time

35

To start a new survey from scratch, the following steps should be taken.

From the select database, use create new accb to create and empty database (or if SQL server is user, create an empty database using SQL Server Management tools).

Next is to add the database to the list of databases.

Select the database. Confirm to create all tables needed (this prompt comes if the DB is empty)

Next is to set base period, that is the period before the first period where data is collected. For instance IV Quarter 2014, if first real period is I Quarter 2015.

You can no start populating the database

- Set Options
- Create base indexed
- Create index structure
- Set weights for base indexes
- Maintain Reason codes (for change of price)
- Now, select prepare new period to set the first period where data is collected
- Create response units
- Assigned products to Response units

You may no start entering data and set product weights for base indexes etc.

Appendix 3 Adding or removing elementary and aggregated indexes

In the lifetime of an index, there may be the need to change the structure, adding or removing indexes but keeping the sane indexes at the top levels.

This will happen if there are new products/services available or a product/service has become insignificant or non-existing.

Elementary indexes

Elementary indexes can only be added or deleted at the point of time when new weights are assigned. (Note new weights may be assigned without change in elementary indexes).

In case any elementary index is added from period x, then all 'old' elementary indexes to be rebased to have period x-1 = 100 and all new should have index in period x-1 set to 100.

If no elementary index is added, even if some are remove, there is no need to rebase indexes (but it may be done anyway in connection with a change of weights).

If in the period between to weight changes it is not possible to get (enough) prices for a product, a substitution index should be selected for such an index. The index will then follow the development of the substitution index, that is if the substation index increases with x%, the index will increase with the x% too.

Aggregate indexes, consequences of add/deleting elementary indexes.

When elementary indexes are deleted, then aggregates including these indexes (including the total) may have to be redefined. In most cases this is like done by simply removing the deleted elementary index for the definition of the aggregate or by terminating the aggregate index.

In the same way, when elementary indexes are added, the definition of aggregate indexes must be revised, at least to include the new indexes in the total index.

Other changes to aggregate indexes

Aggregates indexes may be redefined, added or removed at any point of time. If redefined, the index should be recalculated.

Aggregate indexes are often using a base year, that is different from the base of elementary items. When adding a new aggregate index, if possible it should be calculated from the base year of aggregates (that is all aggregates should have the same base year).

If it is not possible to recalculate the index (it is based mainly of new elementary items) it may be necessary to rebase the aggregates to a point, from where the new aggregate can be calculated.

What base years are needed in the system.

The PPI manual contains the following definition, that seems to be in line with the discussion above

Base period
The base period generally is understood to be the period
with which other periods are compared and
whose values provide the weights for a price index.
However, the concept of the "base period" is not a
precise one and may be used to mean rather different
things. Three types of base periods may be distinguished:
(i) the *price reference period*, that is, the period
whose prices appear in the denominators of the
price relatives used to calculate the index, *or*
(ii) the *weight reference period*, that is, the period,
usually a year, whose values serve as weights for
the index. However, when hybrid expenditure
weights are used in which the quantities of one
period are valued at the prices of some other period,
there is no unique weight reference period,
or
(iii) the *index reference period*, that is, the period for
which the index is set equal to 100.
The three reference periods may coincide but frequently
do not.

In the system as defined, the price reference period is always the previous period (using chained indexes)

The weight reference period is relevant for changes in weights of elementary indexes.

There will be two indexes for reference periods, elementary index reference period (iiia) and aggregate level index reference period (iiib).

(ii), (iiia) and (iiib) may be the same or different. (ii) and (iiia) are linked as described above.

Tech note

The system table defining period contains the name of the period and the weight base period used when calculating aggregates indexes for that period.

To handle substantial changes in the index structure, where there are changes in elementary indexes, it should be possible to have versions of the index structure. For a given period, a specific version is then used, calculating elementary and aggregate indexes using that version. A new version is needed whenever elementary items are added or deleted.

When possible, a new aggregate may be defined in old versions (based in the elementary items of that period). As discussed above, when it is not possible, a aggregates should be rebased.

Appendix 4 Calculation methods

Two different methods of calculation are used in the application.

Elementary indexes

The calculation of Elementary indexes is performed this way:

For the first period in the system, that will be the base period until there is a new version of the index structure, for each elementary item the index is set to 100.

When there is a change in index structure, all base indexes are rebased to 100 and any new indexes are added with the value of 100.

For other periods the following steps are taken using chained indexes (for each period):

1. Individual prices are summarized to prices on reponse unit level for each product using unweighted arithmetic mean. Both current at previous price is computed this way.
    The formula used is
    Exp(Sum(Log(Price)) / Sum(number of prices)) for each product/response unit.

2. Prices (previous and current) on response unit level are now turned into Elementary Aggregates for each product.
    If the product has product weights a weighted arithmetic mean is used, for those where there a not any weighted the unweigthed method is used (as above)
    The formula for weighted arithmetic mean used is
    Exp(Sum(Log([Price^Weight))), for each product. Sum of weigths are 1.

3. In the next step, the ration between Previous. Price and Current Price for each product is calculated as (CurrPrice/PrevPrice)

4. Finally the index is calculated as the index of previous period * the ratio calculated in 3.

Aggregated indexes

Aggregated indexes are calculated using Laspyere indexes.

The indexes may be defined as aggregate of other aggregates, but before calculation this is changed by substitution to ensure that all aggregates are based on a set of elementary indexes.
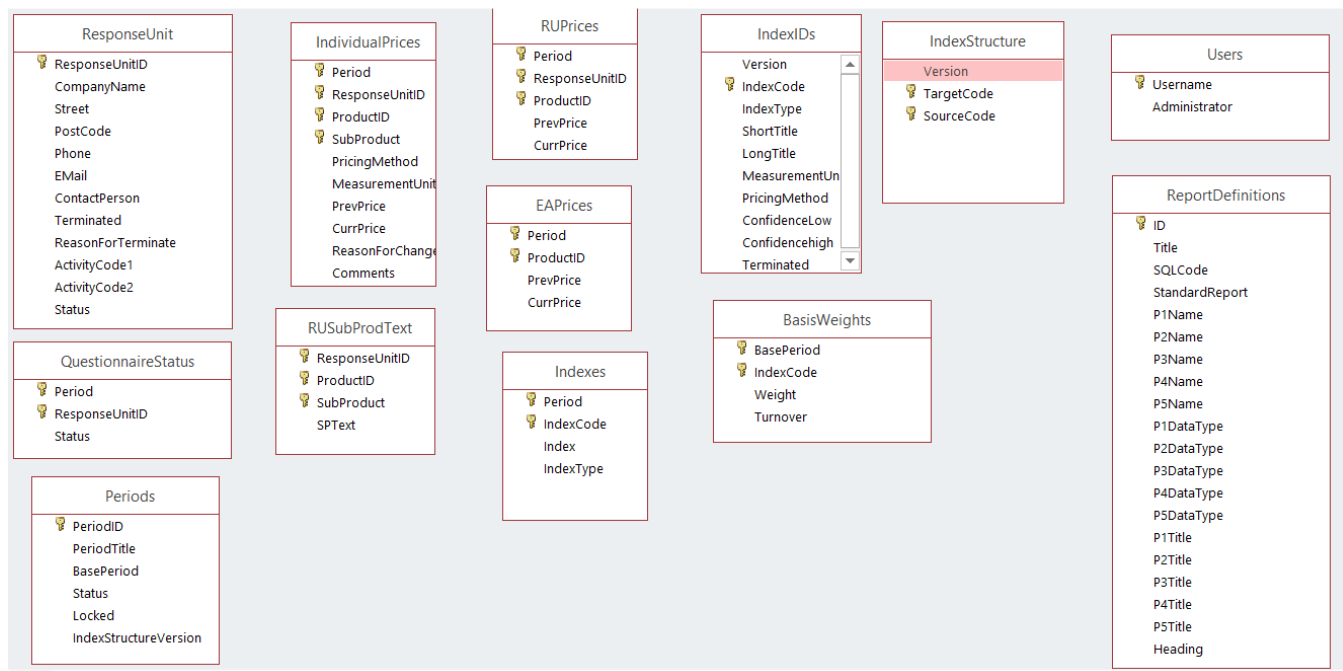
By doing so indexes can be calculated in any order.

The actual calculation is performed in two steps.

1. Calculate (using Laspiere) the change in the index relative to the base year for calculation (weights).
    Sum([Index for period]/[index for base year]*[index weight]) /sum(weights) for all elementary indexes making the aggregate
2. Multiply the index for base period with the change to get the new index.

Appendix 5 Database model

The database at this moment contains the following tables (may be revised)

ResponseUnit contains information about the response units

QuestionnaireStatus has for each period the status for each response unit that has been sent a questionnaire.

Periods is used to give each period a title, a base period, a status and if the period is locked. It also gives information about the version of IndexIds and IndexStructure to use.
PeriodId is just numbering the periods to make it easy to refer to previous period.

IndividualPrices hold the data collected for each Period,ResponseUnit, Product and Subproduct.

RUSubProdText is used to contain the test associated with subproducts. A subproduct is defined at the level of a response unit. Used mainly to be able to produce individual questionnaires.

RUPrices aggregated at Response Unit and Product level.

EAPrices aggregated at Product Level = Base Index Level (Elementary aggregates). Also used in the control of individual prices for the following period.

Indexes are the actual calculated indexes, both elementary and aggregates (Index Type)

IndexIDs hold the ID for each Index (code), the type of index and other information. Since Elementary Indexes are equal to products, some product related information is also included. Long text is used as the initial text for questionnaires to new response units. However, as the response unit may change the text (reporting a similar product), it may be better to use the short title (not so detailed) to identify an index.

IndexStructure for each target (Aggregated index) has one row for each source index (elementary or aggregate). Note that IndexIds and IndexStructure has versions.

BasisWeights holds the weight related to a given base period for elementary items.

Users are simply a list of the users and if they are administrators.

ReportDefinitions are used to hold the definitions of reports as described in appendix 1. Note that some reports may be marked as Standard Report. In this case the report is used by the system, ie. Reports produced after calculation of indexes and should not be removed or changed.

A few tables with codes and title for some variables should be added. As an example. The should be table with the allowed reason codes for change of price (see also under data entry, validation).

If Activity codes are retained in Response Units a table holding valid Activity code should be there.

This list may not be complete.