TWINNING CONTRACT

BA 15 IPA SR 01 17

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

MISSION REPORT

Activity 2.1.7:
IT upgrade of SBR I

Component 2: Business Statistics
Sub-component 2.1: Statistical Business Register

Mission carried out by

Søren Netterstrøm, Statistics Denmark

20 - 24 May 2019

Version: Final

STATISTICS DENMARK

Institut national de la statistique et des études économiques
Insee
Mesurer pour comprendre

Statistics Finland

CROATIAN BUREAU OF STATISTICS

Expert contact information

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

Table of contents

1. General comments 4
2. Assessment and results 4
3. What to do before the next mission on IT for the BC Counterpart 6
4. Topics for the next mission 6
Annex 1. Terms of Reference 7
Annex 2. Persons met 8
Annex 3. Draft model for creation of EGs and updating of SBR. 9
Annex 4. Changes to existing SBR system 14
Annex 5 Time-plan 15
Annex 6. Creating EGs from relations 16
Annex 7. Pseudo Language 21

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”. This was a mission in a series of mission devoted at the introduction of Enterprise Groups (EG) in SBR. Where the missions has been dealing with the concepts and EG and potential sources, this mission is the first of a set of mission on the IT aspects of EG in SBR.

The purposes of the mission were:

- Follow up from the previous mission (Activity 2.1.4) - prepared by BC
  - Review DB model and assign attributes
  - Further analysis of sources
  - Consider how data from sources may be loaded into EGDB
  - Consider flow and division of work for creation of EG
- Upgrading of IT applications in the SBR
  - Finalizing the DB model
  - Finalizing of the procedures for creation of EGs
  - Finalizing procedures for updating EGs in SBR
  - Examining the analyses of sources
  - Description of how sources are loaded into EGDB
  - Distribution of work, sequence, data exchange etc. in the process of creating EGs

The consultant would like to express his thanks to all officials and individuals met for the kind support and valuable information which he 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

2.1 Decisions

In order to describe the outline of the system to create EGs and include them in the SBR several decisions regarding the coverage and other issues needs to be taken.

The following was discussed and agreed during the mission and it is the recommendation of the expert that these decisions are followed. The model build during the mission is based on these decisions and if they are changed the model needs to be changed. One of the implicit goals has been to minimize the resources needed to build and maintain the system and data about enterprises groups and still to be able to use enterprise groups both for profiling and some analysis on the larger groups in BiH. If needed at a later stage, the model can be expanded and refined, but then it may be expected that more resources will be needed to maintain the register.

The potential connection to the Eurogroup register was discussed during the previous mission. Because it is at this stage still very unclear if, when (and how) BiH may get access to the Eurogroup register it was decided not to consider this aspect. The scope of the project is then

enterprise groups within BiH as well as truncated enterprise groups, that is enterprises in BiH where the parent is a foreign legal unit or where and enterprise in BiH is the parent of units abroad.

Several sources have been found that can be used to get relations between legal units, both domestically and abroad. However, for some of these sources, only the existence of the relation is available, but the share of ownership can not be found. However, this information is crucial in the construction of enterprise groups. It seems, that in a not to distant future, this information will be available in electronic form, but for the time being it have to be looked up manually in court registers.
If all relations between enterprises should be examined, this could be a rather time-consuming task. The main usage of the EGs is assumed to be profiling or other analysis of large units. For this reason, it was decided in the beginning only to consider EGs where at least one of the resident legal units has an employment above a threshold (to be defined). This will save resources but at the same time give the institutions some experience with enterprise groups and the specific model to be used. Also profiling and analysis could be started.

Some of the sources available may have a natural person as the parent (owner) of a legal unit. Based on this is was discussed, if natural persons should be considered or not. According to the EU recommendation manual, natural persons may be considered if information is available as head of an enterprise group. It was decided at this stage to omit natural persons from the process, however, they could later be added in a way similar to the way foreign legal units are considered.

It was decided to identify an EG by the identity of the group head. If the group head is a domestic enterprise (legal unit), then the ID of the EG is the same as the ID of the head. A special register is established for all foreign units that are found in the processing. In this register, an ID is created using the same structure as for EnterpriseID but starting with 4 to mark it as a foreign unit. This is then used as identifier of the (truncated) enterprise group. This model assumes, that most enterprise groups will have the same head of group from year to year and this way can be followed. In case a new unit is added on the top of an otherwise unchanged group, this will be recorded as the termination of the old group and the beginning of a new group, where the ideal output would be that the group is the same just with a new head. Mergers and split may like wise not be correctly captured. However, this approach allows a very automated approach to EGs. To take a more ideal approach would lead to a much more complex application (set of rules) and most likely the need to process several changes manually, because decisions could not be automated.
It is recommended that the impact of this decision is analysed when the system is updated, at least first and second time.

EGs may consist of units from more than one entity. For that reason, it does not make sense to process the relations between legal units / enterprises at the level of entity, the process of building EGs must be performed at the level of all of BiH. As a consequence, the first collection of data on relations will be distributed between RIS, FIS and BHAS, depending on who has access to the original source. Once the data is collected and normalised, all data on relations (including share of ownership) is turned over to BHAS. BHAS will then process the data to create the EGs and make the necessary updates of SBR, where two new tables will hold data on EGs.
Some unfinished transaction may occur. The rules for handling these needs to be decided.

There is also a need to discuss and decided don rules related to updating Enterprises and Local Units that is part of an Enterprise Group, especially in the case where the EG has units in more than one entity. Who is responsible and what are the consequences for the data exchange between the institutions?

In the EU manual it is recommended to have data on employment and turnover on the level of Enterprise Groups. For turnover the problem would be to find a source regarding this value, as turnover is not additive. Another problem is, that presently these data are only shared in categories between the entities and BHAS, so adding up employment in case of EGs covering multiple entities is not possible. It has thus been decided not to include information about employment and/or turnover at this stage.

2.2 The model for EGs in SBR

Based on the decision the consultant and the participants in the meeting build a model of how EGs may be constructed and maintained in SBR.
The work was partly based on the ideas put forward in the report from October 2016 on Enterprise Groups and the ideas discussed in the previous meeting. There was also input from BC regarding attributes for the different objects in the data model.
The experience gathered by BC when creating a test model proved to be very helpful.

During the mission, the total flow from cleaning up data on relations between legal units collected from different sources until to the update of EGs in SBR was covered. This was done in an iterative process, where further details were added for each iteration.
The output of that process will be covered in detail in the final mission report and form the basis for the work to be initiated by BC to implement the system.

In the discussion so far, the present situation, where there is a 1-1 correspondence between Legal Units and Enterprises in SBR has been covered. However, since one of the purposes of EGs is to start profiling of large companies, ´real´ Enterprises may be created. The consequences of this needs to be further examined. This may lead to the need for some manual inspection in the update process (unfinished transactions), but it should be insured that simple cases of adding and removing units can be handled without intervention.

Needed changes to the SBR online application and system for update was also discussed.

3. What to do before the next mission on IT for the BC Counterpart

A temporary workplan for the development of the system was created during the mission, see appendix 5.

The BC is expected based on this to have a first version of the implementation of Enterprise Groups developed.

4. Topics for the next mission

- Review rules for update of SBR with profiled EG’s
- Review of the model and progress of work
- Any other issues that may occur during implementation

The mission is expected to take place in the second half of October 2019.

Annex 1. Terms of Reference

Activity 2.1.7: IT upgrade of SBR I

1. Mandatory result and benchmarks for the component

Mandatory result:

- Volume of characteristics in the Statistical Business Register increased and quality of data improved in line with EU standards by 8th project quarter

Benchmarks:

- 5-year development plan for the statistical business register updated and adopted by 2nd project quarter
- Staff trained in profiling methods by 5th project quarter
- First test profiling created by 5th project quarter
- 5-year development plan for the statistical business register updated and adopted by 8th project quarter
- Enterprise groups data integrated into the statistical business register by 8th project quarter
- Plan developed for updating the statistical business register data model by 8th project quarter

2. Purpose of the activity

- Follow up from the previous mission (Activity 2.1.4) - prepared by BC
  - Review DB model and assign attributes
  - Further analysis of sources
  - Consider how data from sources may be loaded into EGDB
  - Consider flow and division of work for creation of EG
- Upgrading of IT applications in the SBR
  - Finalizing the DB model
  - Finalizing of the procedures for creation of EGs
  - Finalizing procedures for updating EGs in SBR
  - Examining the analyses of sources
  - Description of how sources are loaded into EGDB
  - Distribution of work, sequence, data exchange etc. in the process of creating EGs

3. Expected output of the activity

- Plan developed for updating the SBR data model
- Adjustment to the SBR made
- Input provided to the ToR of next activity

Annex 2. Persons met

Agency for Statistics of BiH (BHAS)

- Dzenita Mustafic, SBR Coordinator
- Mevlija Odobasic, SBR
- Ivana Tavra Colo, SBR
- Branislava Cvijetic, IT BHAS
- Senija Facic, Branch Office Brcko District
- Vedad Osmanovic, IT Branch Office Brcko District

Institute for Statistics of Federation of BiH (FIS)

- Emira Beširević, SBR Coordinator
- Enisa Rastić, Head of SBR and Economic Classifications Department
- Envera Hurić, Senior Advisor in SBR and Economic Classifications Department
- Razija Bičakčić, IT expert for SBR

Institute for Statistics of Republika Srpska (RSIS)

- Nada Malinović, SBR Coordinator
- Jelena Baroš, Senior Statistician SBR
- Stevan Marjanović, Senior Statistician SBR
- Pero Kazanović, IT RISIS

Twinning Project Administration

- Katja Møller Hjelvang, RTA
- Đemka Šahinpašić, RTA Assistant
- Haris Imamovic, Interpreter

Annex 3. Draft model for creation of EGs and updating of SBR.

Based on the discussion during the workshop, the following model for the creation of EGs and their inclusion into SBR was developed.

It should be noted, that this is a first draft of the processing and may need further considerations and refinements.

The work is split up in the 9 steps

1. Normalise and standardise input from sources to a common format and using standard classifications.
2. Update of (creation of) register of Foreign Legal Units
3. Filter out relations not meeting the criteria for further processing
4. Check for circular references
5. Assign Share of ownership (manual or(semi)- automatic)
6. Combine data collected by each institution.
7. Use the algorithm to create Enterprise Groups.
8. Create transactions to update SBR
9. Process the transactions

Step 1 to 5 is performed by the institution that receives data from the external source, this may be FIS, RS or BHAS.
Step 6and 7 needs to be handled at the level of BiH and are performed to BHAS.
For Step 8 and 9 it should be further analysed whether it is simpler to perform the process of update SBR part individually in each institution or in one place and then be distributed in some way to other institutions. The critical point may be handling of unfinished transactions,

Step 1. Normalise and standardise

Data about relations between legal units may come from a number of sources.
Both the structure and the classifications used may differ between the sources. To be able to run the following steps it is nessecary to have a common format and to use a standard set of classifications.

The expected output from this stage is transactions with the following content

- Parent Unit identified by Tax_ID in FIS and BHAS or RS_ABR_ID in RS for domestic units.
  Foreign units are handled in step2
- Child unit identifies in the same way
- *Country Code*

The remaining attrubutes are only used for Foreign Units. At least 1 F_id must be available, but as many attributes as available should be filled.

- *F_id1*
- *F_id2*
- *F_id3*

} *foreign ID numbers, from original administrative sources for foreign legal units*

- *Name*

- *Address*
- *Telephone, email*
- *activity code*
- *Share of Ownership (if available)*
- *Source*

It should be noted, that data about natural persons (as parent or child) is not included but such records in the original sources are skipped (filtered out).

Also relations between to foreign units may be filtered out.

Step 2. Update (or Create) register of foreign units

ID for foreign units is added to the transactions created in step1

Central in this proces is the register of Foreign Legal Units (FLE).

For each transaction where parent or child is a foreign unit, based on Country Code and F_Idn a lookup is performed in FLE. A unit is foud if country code matched the country code in FLE and (F_ID1 matches F_ID1, F_ID2 or F_ID3 in FLE or F_ID2 matches F_ID1, F_ID2 or F_ID3 in FLE or F_ID3 matches F_ID1, F_ID2 or F_ID3 in FLE).
If a match is found, FLE may be updated if the transaction has data on name, adress etc that is not found in the register or an ID not already recorded.

If match, the ID from FLE (*FE_SBR_ID)* is inserted in the transaction.

If there is no match in FLE, a new unit is created in FLE. The *FE_SBR_ID* is created using the same procedure as is used for Enterprise ID in SBR, but first digist is set to 4. The second digit may reflect the institution responsible for creation of the unit.
The ID of the new unit is then transferred to the transaction.

FLE must be shared between all institutions. Each institue will have a local copy of FLE. It is suggested, that whenever an institution is ready to perform step 2, they reserve FLE by informing the partners). After completion of step 2, they then send the updated FLE to the partners. It is important to insure, that the same Foreign Unit is recorded only once.

FLE will have *FE_SBR_ID, Country Code, date of creation (in FLE) and Source of information. It will also hold F_Id1, F_ID2 etc. from the transaction (except share of ownership).*

Step 3. Filter out relations not meeting the criteria for further processing

Because data on share of ownership is not available in all sources, it is decided only to process units with an employment above a certain threshold.

The first step is for each transaction to determine, by consulting SBR, if any domestic unit (parent or child) is having an employment above (or equal to) the threshold.

All Domestic IDs (both child and parent) for these transactions are added to a table of IDs (if not already there).
To this table is then, from SBR, added to IDs of Legal Units (Tax_ID or RS_ABR_ID as relevant) that is part of an existing Enterprise group (active or inactive).

To capture all units across BiH, this step may need to be synchronize between the partners to ensure that units captured by one partner is known to all partners.

Finally, for each transaction it is examined if Parent or Child ID is in the list. In not, the transaction is not further processed.

Step 4. Check for Circular references

Before proceeding, it should be checked that the material does not contain circular references, both directly and indirectly. I.e. 1 is parent of 2, 2 is parent of 3 and 3 is parent of 1.
If this is the case, the reason for this should be examined and corrected.

Step 5. Assign Share of ownership (manual or(semi)- automatic)

In this step, for all transactions not having information about share of ownership, this information must be filled.

This may be a manual process, looking up the information in court registers or other sources available.

It may be considered to reuse information from previous year, that is, if the parent-child relation existed in the previous year, the share may just be copied. It may be further examined, that the child has the set same of owners as in the previous year.

Step 6. Combine data collected from each institution.

Now all transactions are transferred to BHAS for the final processing.

Combine all transaction from all institutions.
If there are any duplicates, they should be examined and removed.

Step 7. Use algorithm to create EG

Using the algorithm in annex 6 the Enterprise groups are created. (Taken from the mission report on 2.1.5 17-26 October 2016 (from previous twinning project) the Enterprise groups are created.

Note that the input to that process is the transactions created in step 6.

The output from this process is

Relations
RootID (Tax_ID or RS_ABR_ID or FE_SBR_ID)
ChildID (Tax_ID or RS_ABR_ID or FE_SBR_ID)
Type Of Control (Direct or indirect)

RootIDs is the same as enterprise ID.

Step 8 Create transactions to update SBR

In order to update SBR, transactions are created by comparing the relations created in step 7 with the existing relations between Enterprises and EnterpriseGroups.

First, in the relations, Tax_ID and RS_ABR_ID is converted to Enterprise_ID
The root may be domestic or Foreign. All relations where child is a Foreign Unit is not processed.

The table in SBR contains fk_.. as UID, make a copy where these are replaced by Enterprise_ID and EG_ID.

Now by comparing this three types of transactions may be constructed

1. New relation (does not exist in SBR)
2. Continuation (exist in SBR and relations)
3. Discontinued relation (exists only in SBR)

The transactions should contain EG_ID (parent), Enterprise_ID (child) and type of control.
If parent is a foreign unit, the name, address etc. as needed to build/update an EG should be included as well.

Step 9 Update SBR

For new relations (type1), it must first be examined if the relation is a relation to an existing EG. If this is not the case, a new EG should be build.
Next the relation between Enterprise and EG should be established.

For continuation (type 2) if should be examined, if the old relation is terminated. If that is the case, a new relation should be build.
Type of control (direct/indirect), what if that is changed ??
In case of relations to a foreign unit, it may be considered to test (and update) name, address etc.

For discontinued relations (type 3), the relation in SBR is closed.

For transactions of type 1 and 3, a new transaction, check activity code, should be created. These transactions are processed when all transactions of type 1 and 3 has been processed.

The activity code of an Enterprise is a 2-digit NACE-code. The rules for updating this code needs to be established.
A starting point may be to examine the activity code (first 2 digits) of all active Enterprises with an active relation. If all enterprises as the same code, then this should be the code of the EG.
This will lead to some unfinished transactions that must be dealt with manually.

However, if the number of unfinished transactions becomes large, further refinement could be considered.
It may be considered to establish a rule, that activity codes in chapter C to F (10-39) has priority, so if they are all the same (or there is only 1), this code is used. It would handle cases, where there is 1 unit in Manufacture and other units is wholesale, administration or holding companies etc. In such cases it may be natural to see the EG as Manufacture.

For all EGs the activity status should be updated.
If the EG have 1 or more active relations to an active Enterprise, the EG is active,
else if the EG have 1 one more active relations to dormant units, the EG is dormant,
else the EG is inactive (as all relations will be to inactive units).

Content of SBR

In SBR two new tables are added.

Enterprise Group
- UID
- EG-Id
- Multi-national (Foreign head, Foreign children, both, fully domestic)
- Multi-Entity (All (Active) domestic units in same Entity, mixed)
- Name
- Activity Code
- Activity Status (active, dormant, inactive)

EG_Enterprise_relation

- Fk_EG
- FK_Enterprise
- Relation start (date when relation is established)
- Relation termination (date when relation is terminated)
- Relation type (direct or indirect control)
- (Head of group ?)

It should be noted, that start/termination is only updated when relations are processed and does not reflect if the EG of Enterprise is active, dormant or inactive.

Annex 4. Changes to existing SBR system

Update of SBR, consequences

If the activity code of an Enterprise (that is part of an Enterprise group) is changed the activity code of the EG may need to be changed (using the same rules as above).
If the name of an Enterprise changes, and the enterprise is Head of group, then Name of group may change.

Online system

A new window needs to be added to the only system holding information about EG.

Similar to the way used for Enterprises (and other units), the top part of the window will display information about the EG. The bottom part will list all relations to enterprises with ID of Enterprise, name, date of start of relation and date of termination of relation, date of cessation of enterprise.

In the Enterprise window, the ID Enterprise Group should be added and in the lower part a list of all relations to EGs (actual and historical) should be added as a new tab.

Note, that there can only be one relation where termination data is null.

Annex 5 Time-plan

The following time table was set up on the last day. It may be seen as a tentative schedule, however, the time table should allow for reasonable time for each task.

Task	descr	resp	before
Analyse Central Bank data	Information about foriegn unit, code list for country code	Dzenita Mustafic	15 June 2019
Step 1,2 procedure for CBB pseudocode	Incl. lookup and creating of new unit in SBR_FE	Dzenita Mustafic	15 July 2019
Code Step 1,2 for CBB	Create SQL procedure	Branislava Cvijetic	1 August
Analyse da from xx.	These tasks are repeated for each source		
Step 1,2 procedure for xx pseudocode			
Code Step 1,2 for xx			
Pseudocode for step 3 and 4	Revise description in report, agree on threshold	Nada Malinović	15 July 2019
Code for step 3and 4	Create SQL procedure	Pero Kazanović	1 August
For RS, insert share (step5)	Create SQL procedure	Pero Kazanović	1 September
For FIS/DBr, step5	Determine rule	Emira Beširević	1 July 2019
For FIS/DBr, step5	Create SQL procedure	Vedad Osmanovic	1 August
Step 6 and 7	Merge and algorithm Psudocode (review)	Dzenita Mustafic	1 September
Step 6 and 7	Create SQL procedure	Branislava	1 October
Step 8,9	Review of rules (pseudocode)	Emira Beširević	1 August
Step 8,9	Create SQL procedure	Razija Bičakčić	1 October
Update SBR application		Pero Kazanović	1 October

Annex 6. Creating EGs from relations

(this is copied from the report October 2016)

Input to the process is a table (LEG_REL) that contains
Parent_ID
Child_ID
Ownership%
Example

Parent	Child	Percent
1	2	60
1	3	60
2	4	90
4	5	80
3	6	30
4	6	30
7	6	40

STEP 1
From this table a new table (GLeU_POP) with all ID-numbers are created
At this stage Controlled and root are blank

ID number	Controlled	Root
1		1
2	1	1
3	1	1
4	1	2
5	1	4
6		6
7		7

For each row in LEG_POP it is checked if there is a row in LEG_REL where Child =ID Number and Percent > 50.
For these the rows, Controlled is changed to 1 as in the example above and Parent is set as root
For other rows IDNumber (1,6,7) root is equal IDNumber

STEP 2

Next step will examine and update root in LEG_POP.
For each row
If root <> idnumber
Old_root = root
New_root = select root from Leg_POP where IDNumber=old_root
If new_root <> old_root
Set root=new_root
End if
End if
This step is repeated until no rows change root

Example
4 has 2 as root
2 has 1 as root next iterarion (4 gets 1(
1 has 1 so no change

Now LEG_POP should look like

ID number	Controlled	Root
1		1
2	1	1
3	1	1
4	1	1
5	1	1
6		6
7		7

STEP 3
Now it is time to look for indirect control.
To do this we create a new relation table (Root_LEG)
From LEG_REL all rows where control of child (in LEG_POPI is blank
Root is taken from Leg_POP, Child and percent from LEG_REL
Select A.Root as Parent, ,B,Child, b.Percent
from LEG_POP A Join LEG_REL B on A.IDNUMBER = B.Parent
where A.Control is NULL
This looks like

Parent	Child	Percent
1	6	30
1	6	30
7	6	40

Now summarize over Parent and Child to get

Parent	Child	Percent
1	6	60
7	6	40

For each row in LEG_POP where control is blank it is checked if there is a row in Root_LEG where Child =ID Number and Percent > 50.
For these the rows, Controlled is changed to 2 and Parent is set as root

ID number	Controlled	Root
1		1
2	1	1
3	1	1
4	1	1
5	1	1
6	2	1
7		7

Now repeat step 2 to find real roots (nit need in this example)

STEP 3 (and Step2)is repeated until there are no more changes made

Output from Step 3

Parent	Child	Percent
7	6	40

Summarized we get the same table.
The process stop as there will be no more changes.

```
SET NOCOUNT ON
--
--  This procedure requires that the table LEG_REL exists.
--
--
--  STEP 1
-- From this table a new table(LEG_POP) with all ID-numbers are created
--
IF OBJECT_ID ('dbo.LEG_POP', 'U') IS NOT NULL
  DROP TABLE dbo.LEG_POP

select  distinct parentID as ID into parents from dbo.LEG_REL
select  distinct childID as ID  into children from dbo.LEG_REL

select  ID, null as Controlled, null as Root into LEG_POP from parents
 union select  ID, null as Controlled, null as Root from children

DROP TABLE dbo.parents;
DROP TABLE dbo.children;
--
-- Determine Direct Control
--
--For each row in LEG_POP it is checked if there is a row in LEG_REL where Child =ID Number and Percent > 50.
--For these the rows, Controlled is changed to 1 d Parent is set as root
--For other rows  IDNumber root is equal IDNumber
--
DECLARE @POPID int
DECLARE @PARENTID int
DECLARE @pct INT
DECLARE @indirect AS INT
DECLARE @ROOTCHANGED as int
DECLARE @OldRoot as int
DECLARE @NewRoot as int


DECLARE c1 CURSOR  FOR
  select a.ID, b.PArentID, B.PCT from dbo.LEG_POP A left join dbo.LEG_REL B on A.ID = B.CHILDID

OPEN c1 -- open the cursor
FETCH NEXT FROM c1 INTO  @POPID, @PARENTID , @pct

WHILE @@FETCH_STATUS =0
   BEGIN
      if @parentID is NULL OR  @pct < 51
        update dbo.LEG_POP set Root = @POPID WHERE ID = @POPID
     else
       update dbo.LEG_POP set Root = @PARENTID, Controlled = 1  WHERE ID = @POPID

      FETCH NEXT FROM c1  INTO  @POPID, @PARENTID, @pct
END
CLOSE c1 -- close the cursor
DEALLOCATE c1 -- Deallocate the cursor

SET @indirect = 1       -- used to control when process is done (no more indirect relations are made)

WHILE @indirect = 1
  BEGIN
      SET @indirect = 0
--
--STEP 2 Get the real root
--
```

```
	set @ROOTCHANGED = 1	-- used to control when real root is found

	While @ROOTCHANGED = 1
	  BEGIN
	    DECLARE c2 CURSOR  FOR
	        select a.ID, a.root, b.root as newroot from dbo.LEG_POP A left join dbo.LEG_POP B on A.Root  = B.id

	    OPEN c2 -- open the cursor

	    FETCH NEXT FROM c2   INTO  @POPID, @oldroot, @newroot

	    set @ROOTCHANGED = 0
	    WHILE @@FETCH_STATUS =0
	      BEGIN
	        if @oldroot <> @newroot
	          BEGIN
	            update dbo.LEG_POP set Root = @NewRoot WHERE ID = @POPID
	            set @ROOTCHANGED = 1
	          END
	        FETCH NEXT FROM c2  INTO  @POPID, @oldroot, @newroot
	      END
	    CLOSE c2   -- close the cursor
	    DEALLOCATE C2
	  END

--STEP 3
--Now it is time to look for indirect control.
--
	IF OBJECT_ID ('dbo.LEG_REL_COPY', 'U') IS NOT NULL
	  DROP TABLE dbo.LEG_REL_COPY;

	SELECT c.Root as  ParentID, A.ChildID, sum(A.Pct) as pct
	    INTO LEG_REL_COPY
	    FROM LEG_REL a join LEG_POP b on a.childID = b.id left join LEG_POP c on a.ParentID = c.id
	    WHERE b.Controlled is null
	    GROUP BY c.Root,a.ChildID

--  For each row in LEG_POP where controlled is NULL is checked if there is a row in LEG_REL_COPY where Child =ID Number
--  and Percent > 50.
-- For these the rows, Controlled is changed to 2 and Parent is set as root


	DECLARE c3 CURSOR FOR
	select a.ID, b.PArentID, B.PCT from dbo.LEG_POP A  join [dbo].LEG_REL_COPY B on A.ID = B.CHILDID
	    where a.Controlled is null
	OPEN c3
	FETCH NEXT FROM c3    INTO  @POPID, @PARENTID , @pct
	WHILE @@FETCH_STATUS =0
	  BEGIN
	    if   @pct > 50
	      BEGIN
	        UPDATE  dbo.LEG_POP set Root = @PARENTID, Controlled = 2  WHERE ID = @POPID
	        SET @indirect=1
	      END
	    FETCH NEXT FROM c3   INTO  @POPID, @PARENTID, @pct
	  END

	CLOSE c3 -- cloose the cursor

	DEALLOCATE c3 -- Deallocate the cursor

END
DROP TABLE dbo.LEG_REL_COPY;
```

Annex 7. Pseudo Language

Pseudo Language is an idea originally developed during a major revision of the Danish SBR. It has also been used in SBR projects in Croatia, BiH and Kosovo.

The main idea of Pseudo Language is to create a bridge between the subject matter specialists responsible for giving the rules for updating SBR and the IT specialist who will transform these rules into code using a formal computer language.

The basic concept is to create a basic structure to the set of rules and to enforce some strict logic without having a rigid formal language. To achieve this goal a limited set of fixed constructions are used and mixed with descriptions in 'normal' language.

The most important structure is
IF … THEN
….
ELSE
….
END IF

An example

IF unit has no employment for 2 years THEN
  Mark unit as statistical dead
ELSE
  Mark unit as statistical active
END IF

Note than IF, THEN, ELSE and END IF are written in capital letters.
In the example above both condition and actions are written in plain language. When appropriated and it may be more formal like IF Sex = 1 THEN rather than IF Male THEN, or a by using comments
IF Sex = 1 THEN * Male

Another structure is

CASE … THEN
…
CASE … THEN
…
CASE … THEN
…
END CASE

Like
CASE Unmarried THEN
..
CASE Married THEN
…
CASE Divorced THEN
..
CASE Widowed THEN
.
END CASE

Comments or notes are an important part of Pseudo Language and should be used. There are basically three types of text, Conditions (written between IF/CASE and THEN, actions (written after THEN) and then explanatory text. The latter is important as Pseudo Language is also intended to be part of the documentation. By putting in notes and comments may be a good way to make the document easier to read and understand after a while for yourself or other persons.