Compilation of Supply and Use tables and Input-Output tables in Albania

Lindita Cokaj
2 – 4 October 2018, Chisinau Republic of Moldova
1. SUT state of play
2. Data sources for SUTs and IOTs
3. NA Builder program
   - Introduction on NA Builder
   - Demonstration
State of play SUT/IOT

- 2015 – first release of annual SUTs in current values and derived SIOT, for years 2009-2011 according to NACE Rev. 1 (25 products, 25 activities),
- 2016 – published SUTs in current values and derived SIOT, for years 2012 and 2013, according NACE Rev. 2 (35 products, 35 activities),
- 2017 – published SUTs in current values, for year 2014. Experimental compilation of SUT in pyp,
- 2018 – work is ongoing for SUT 2015 in current values, SIOT and development of the compilation of SIOTs for domestic use and imported
State of play: SUT/IOT framework

SUT, T
A89 x P90

SUT_{pyp}, T/T-1
Experimental
A89 x P90

TTM
NT

SUT_{bp}, T
Experimental
A89 x P90

USE_{bp} (tot, imp, dom)
Experimental
A89 x P90

SIOT, T,
A35x A35
State of play SUT/IOT

- SUT compilation and release after annual estimates, not fully integrated in estimation of production approach and expenditure approach
- Third quadrant of SUT (income approach) is not compiled
- Output is not breakdown in sub categories P11, P12 and P13 (market output, output for own final use, other non-market output)
- Cif / Fob adjustment included in products and not as total required from EU transmission programme
- Not estimated transactions for non-residents in Albania, transactions for residents abroad
### Data sources for SUTs and IOTs

#### Supply table
- Production approach database at enterprise level,
- List of industrial products from Structure Business Survey (SBS), FTS and Custom data, Bop data,
- Agriculture data,
- Financial institutions data, Government.

#### Use table
- Cost structure (SBS),
- Annual accounts estimation on HFCE and Household Budget Survey (HBS),
- Government final consumption expenditure,
- GFCF estimates for expenditure approach,
- FTS and Custom data.
Way a National Accounts (NA) automation

- **Major challenge** - to design an IT system in which:
  - data are stored in databases
  - NA compilations take place in spreadsheet structures

- **This is what **NA Builder** aims to achieve:**
  - EXCEL application
  - sets up complex tabular structures in a simple, intuitive way,
  - refers to NA classification codes, rather than EXCEL column letters and row numbers
NA Builder is an application to:

- Build National Accounts tables, using classifications
- Add data to and take data from these tables using normalized lists
- Change data in these tables, using a variety of tools (manual edits, batch edits, view based edits, edit rules)
- Build and subsequently carry out compilation strategies as scripts, containing rules

NA Builder comes with settings files for SUT, QNA and IEA (others can be developed upon request)

NA Builder comes with an ACCESS database back-end
NA Builder is an application stored in an Excel .xlsxb workbook (xlsxb = binary xlsx).

It consists of 21 application sheets, of which:

- The sheets *Info*, *Dict*, *SB*, *Aggr*, *Log*, *Vw* and *Scrt* are the program sheets.
- The sheets *Templ*, *Part*, *ClasIt*, *Clas*, *Brdg*, *Cor*, *Frml* are the sheets where structure information is stored (on respectively: templates, partitions, classifications, bridges, correspondences and formulas).
- The sheets *Files*, *DataIn* and *DataOut* are the sheets where data are specified and stored.
- *The sheet Impl* contains the table definitions (to be implemented).
- The sheets *Rls*, *ScptRls* and *Scpt* are the sheets where rules and scripts are specified and stored.

*Framework (can be exported to a separate workbook)*
Data storage in **NA Builder**

Tabular data in sheets …

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Excel addresses `<sheet>!<<column letter><row number>` are “translated” as 4-tuples:

\{<sheet name>;<partition name>; <row classification code>; column classification code>\}

... and data from sheet “DataIn” can be inserted in the tables

**… are stored in “normalized” form in sheet “DataOut”**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

… and data from sheet “DataIn” can be inserted in the tables
In the database each dataset in *DataIn* or *DataOut* is assigned:

- User
- Type
- Framework
- Revision

*NA Builder* comes with an ACCESS database implementation (other databases are possible as well)

---

### Database fields

<table>
<thead>
<tr>
<th>Database fields</th>
<th>NA Builder fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Sheet</td>
</tr>
<tr>
<td>Type</td>
<td>Partition</td>
</tr>
<tr>
<td>Framework</td>
<td>Row code</td>
</tr>
<tr>
<td>Revision</td>
<td>Column code</td>
</tr>
</tbody>
</table>

---

### Example:

<table>
<thead>
<tr>
<th>Database fields</th>
<th>NA Builder fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBS</td>
<td>NFC_Data</td>
</tr>
<tr>
<td>SUT</td>
<td>P1</td>
</tr>
<tr>
<td>SUT2012</td>
<td>DA</td>
</tr>
<tr>
<td>3</td>
<td>C_1.211</td>
</tr>
</tbody>
</table>
Multi-user aspects

Structure $S$ for framework $F$ of type $T$

Data entry User

Local copy of NA Builder with structure $S$ loaded

Data sources $1 \ldots m$ prepared in EXCEL as:
- Normalized lists
- Tables

Data entry User

Local copy of NA Builder with structure $S$ loaded

Data sources $m+1 \ldots n$ prepared in EXCEL as:
- Normalized lists
- Tables

Data balancing User

Local copy of NA Builder with structure $S$ loaded

DataIn, DataOut as normalized EXCEL lists

NA Database

Database containing separate datasets for each revision for User 1, User 2 and User 3

Database consolidation generates new revisions for each user
## Rules

- Rules are mini programs contained within NA Builder to carry out particular tasks
- Each rule has a particular syntax
- Script = list of rules in a particular order

### Reclassification rules
- COPY
- DIAG
- PROJECT
- RECODE

### Balancing rules
- ABSORB
- DISAGGR
- PRORATE
- RAS

### Time series rules
- COMPARE
- CROSSCUT
- DENTON
- DISTRIBUTE
- FOLD
- INDEX
- UNFOLD

### Calculation rules
- ADD
- CALCULATE
- DIVIDE
- FORMULA
- MULTIPLY
- RAISE

### Input Output Table rule
- IO, allowing:
  - Product table, product technology assumption
  - Product table, industry technology assumption
  - Industry table, assumption of fixed industry sales structures
  - Industry table, assumption of fixed product sales structures
  - Product table, hybrid product technology assumption
  - Product table using Almon method
## Security issues

### System info status:
- **Hidden** = columns Z:DC, DM:DT, EA:FB of sheet SB hidden
- **Not hidden** = columns Z:DC, DM:DT, EA:FB of sheet SB not hidden

### Lock status:
- **Unlocked** = interface unprotected, settings unprotected
- **Partial locked** = interface protected, settings unprotected
- **Locked** = interface protected, settings protected

### NA Builder can be obtained in three versions:
- **Unsecure** = Unlocked, not hidden, settings files unprotected, help unprotected
- **Weakly secure** = Partially locked, hidden, settings files unprotected, help protected
- **Strongly secure** = Locked, hidden, settings files protected, help protected
Obtaining **NA Builder**

The aim of **NA Builder** is to aid small NA Departments with little or no IT support in their full implementation of SNA 2008 using modern IT techniques.

- **NA Builder** is freeware, i.e. it can be used by the institution to which permission of use has been granted by the copyright holder for an unlimited period of time free of charge.
- It is forbidden to pass on the application to another institution or person without written consent of the copyright holder.
- **NA Builder** may not be used for any commercial (consultancy or any other) activity.

More information on this website: https://sites.google.com/site/nabuilder1/

Copyright notice: NA Builder © Gosse Hommes, 2015, all rights reserved

This EXCEL application has been developed by:

Gosse Hommes (Hommes Consultancy)
Putten
Netherlands
gh.hommes@xs4all.nl
Thank you!

Visit us: www.instat.gov.al

Contact us:
Lindita Çokaj: lbecolli@instat.gov.al