

# smatrix MANUAL

Version 1.5

All rights reserved especially (also in extracts) for translation,  
reprinting, reproduction by copying or other technical means.

dawin gmbh  
Belgische Allee 50  
D-53842 Troisdorf

*“**Plant phenotyping** is the comprehensive assessment of complex plant traits such as growth, development, tolerance, resistance, architecture, physiology, ecology, yield, and the basic measurement of individual quantitative parameters that form the basis for the more complex traits. Examples for such direct measurement parameters are image-based projected leaf area, chlorophyll fluorescence, stem diameter, plant height/width, compactness, stress pigment concentration, tip burn, internode length, color, leaf angle, leaf rolling, leaf elongation, seed number, seed size, tiller number, flowering time, germination time etc. ”*

Definition from: <http://www.lemnatec.com/plant-phenotyping/>

## Version History

Date	Manual version	Applicable smatrix version
22/3/2017	1.0	1.0.0.9 or later
11/4/2017	1.1	1.0.1 or later
20/11/2017	1.2	1.5.0 or later
5/11/2018	1.3	1.7.0 or later
2/4/2019	1.4	1.7.1 or later
09/4/2020	1.5	1.8.0 or later

1. Contents	
1 Introduction .....	5
2 Installation .....	6
2.1 Prerequisites .....	6
2.2 smatrix installation.....	6
2.2.1 Silent installation.....	7
3 Using smatrix step by step .....	8
3.1 Speech input guidelines.....	8
3.1.1 General.....	8
3.1.2 Position of the microphone .....	8
3.1.3 Valid identifiers and how to pronounce them .....	9
3.1.4 How to pronounce numeric values .....	11
3.1.5 How to pronounce date values.....	11
3.2 Step by Step.....	11
3.2.1 Verify that a headset is connected.....	11
3.2.2 Start smatrix .....	12
3.2.3 Discover the start screen features.....	13
3.2.4 Check audio devices.....	15
3.2.5 Setup Sheet .....	19
3.2.6 Select Excel-file .....	20
3.2.7 Select Excel sheet.....	20
3.2.8 Sheet autoconfiguration .....	21
3.2.9 Sheet configuration with a template .....	21

3.2.10	Mapping of the identifiers.....	22
3.2.11	More on the mapping of the identifiers.....	24
3.2.12	Mapping of the criteria.....	26
3.2.13	Mapping of the criteria with subsamples.....	29
3.2.14	Mapping of multi-level criteria.....	32
3.2.15	Rows selection.....	35
3.2.16	Parameters.....	36
3.2.17	Save as template.....	38
3.2.18	Save the configuration.....	39
3.2.19	Input Data in Sheet.....	40
3.2.20	Ready <-> Pause.....	41
3.2.21	Entering data.....	42
3.2.22	Entering the 'not available' value.....	43
3.2.23	Entering multiple values.....	43
3.2.24	Clearing cells.....	44
3.2.25	Filling empty cells.....	45
3.2.26	Undo and Redo.....	46
3.2.27	Navigating.....	47
3.2.28	Saving and exiting.....	48
3.2.29	Setting speech feedback.....	48
3.2.30	Setting options.....	49
3.2.31	Inserting a new column.....	52
3.3	Counting Survey.....	53

3.3.1	What is it? .....	53
3.3.2	Configuration .....	54
3.3.3	Voice commands for counting .....	57
3.4	Creating a user profile .....	57
4	Customizing smatrix.....	58
4.1	Editing smatrix defaults.....	58
4.2	Customizing criteria names and data ranges.....	59
4.2.1	Object names.....	60
4.2.2	Criteria names.....	62
4.2.3	Value Ranges .....	64
4.2.4	Customizing Autoconfiguration.....	74
4.2.5	Smatrix configuration XML syntax.....	77

## 1 Introduction

smatrix is a software product for speech-enabled recording of plant phenotyping data. Phenotyping data is often collected in Excel sheets. The format of the data is either operator-defined or exported from trial data management programs (e.g. the ARM software from Gylling Data Management Inc.)

smatrix highlights are:

- The smatrix user enters phenotyping data by voice dictation. Feedback is given by the system through speech synthesis.
- smatrix stores the phenotyping data in an Excel sheet.
- The smatrix operator can configure the range of data rows/columns for the current session.
- For each column, a data range can be configured. A data range represents the values that are valid for this column. For each value, alternative pronunciations can be provided.
- A particular walkthrough can be set i.e. row by row for a selection of columns or just a single column.
- smatrix implements a few simple rules to auto-detect the layout of exported sheets.
- smatrix is available in several languages.

## 2 Installation

### 2.1 Prerequisites

smatrix has the following prerequisites:

- Windows 7, 8 or 10.
- Microsoft Office 2010 or later.
- For viewing the manual, a pdf viewer like Acrobat Reader or Foxit Reader.

### 2.2 smatrix installation

smatrix relies for speech recognition and speech synthesis on software from Nuance Communications Inc. dawin provides installable language packs for the individual languages.

The following packages have to be installed:

- SmatrixSetup.msi: this is the smatrix installation package. The default install directory is “C:\Program Files (x86)\Smatrix”
- LanguagePack\_*lang*.msi: this is a language pack for the speech software. *lang* is the language code ( *eng* for British English, *enu* for American English, *ged* for German, *frf* for French, *dun* for Dutch, *spe* for Spanish, *ptb* for Brazilian Portuguese, *ptp* for Portuguese Portugal). Only the needed languages have to be installed.

IMPORTANT: When installing the language packs, accept the default install directories.

- When upgrading from an earlier smatrix version, **you have to** uninstall the old version of the smatrix software.
- When upgrading from Smatrix version 1.0.x, **it is best** to uninstall all the previous language related msi's ( 2 base packages + 2 extra per language) and instead install the language packs (1 pack per language) :

- VoconHybrid\_v4\_9\_base.msi: this is the base installation package for the Vocon Hybrid speech recognizer.
- VoconHybrid\_v4\_9\_*lang*.msi: this is a language data package for the Vocon Hybrid speech recognizer. *lang* is the language code ( *eng* for British English, *enu* for American English, *ged* for German, *frf* for French).
- VocalizerExpressive\_v1\_5\_0\_base.msi: this is the base installation package for the Vocalizer Expressive speech synthesis.
- VocalizerExpressive\_v1\_5\_0\_*lang*.msi: this is a language data package for the Vocalizer Expressive speech synthesis.

### 2.2.1 Silent installation

Each package can be installed silently (i.e. in a script without user interaction) with the following command:

```
msiexec /i packagename.msi /quiet
```

Alternatively, a progress bar can be shown while still not allowing any user interaction with the following command:

```
msiexec /i packagename.msi /passive
```

Silent uninstall can be accomplished using:

```
msiexec /x packagename.msi /quiet
```

### 3 Using smatrix step by step

#### 3.1 Speech input guidelines

##### 3.1.1 General

The speech recognition software used by smatrix is a continuous speech recognizer. This means that the software is optimized for natural speech without pauses between the individual words. A speech pause will always be interpreted as the end of the speech command.

Contrary to system as Siri or Google Now, smatrix has a limited vocabulary. It recognizes only the phrases that are valid in a specific application context. Out of context speech will not or incorrectly be recognized. The speech recognizer will always look for the in-context phrase that matches best the utterance. This can sometimes surprise the user.

##### 3.1.2 Position of the microphone

As a rule of thumb, the microphone should be positioned at a distance of about 1 finger width in the corner of the mouth.

If the headset has a noise-canceling microphone, make sure that the correct side points in the direction of your mouth. Typically, a printed dot is visible on this side.

Outside, always use a windscreen around the microphone.

### 3.1.3 Valid identifiers and how to pronounce them

During a smatrix survey, the surveyor enters data describing the characteristics (trait) of particular objects (plant, parcel, pot, ...). Each object is identified by a single or a combination of multiple identifiers, e.g.: a unique parcel number, row and position-in-row numbers for a pot in a glasshouse.

For smatrix, a valid identifier is either:

- A **single** letter **A-Z**
- Or a sequence of characters **ending** with a **sequence of digits with an optional letter suffix**. In this case only the last 6 digits and optional suffix are meaningful for smatrix.
- Or one of the above followed by a **single space** and a subsample number.

The following table shows some identifiers and the relevant meaningful part.

Identifier	Meaningful	Comment
<b>3546789</b>	546789	Only last 6 digits
<b>3546789D</b>	546789D	Only last 6 digits + suffix
3546- <b>789D</b>	789D	Digit sequence starts after last non-digit
3546-A		Not valid
<b>G</b>	G	Single letter identifier
<b>G 1</b>	G 1	Single letter identifier + subsample number
X <b>235</b> 1009	235 1009	Digit sequence starts after last non-digit + subsample number

Identifiers can be pronounced as follows:

Identifier	Identifier Pronunciation	Subsample Pronunciation
100	one hundred	N/A
100	one zero zero	N/A
105A	one hundred and five A	N/A
105A	one zero five A	N/A
20 1	twenty	one
<u>2452456</u> 6	Four five two four five six	six
<u>2452456</u> 23	Four hundred fifty-two thousand four hundred fifty-six	twenty-three OR two three
ABC45- <u>5</u>	five	N/A

**Note:**

When pronouncing less than the meaningful digits, Smatrix looks for the first object with a number ending in those digits.

### 3.1.4 How to pronounce numeric values

Examples:

105 = one hundred and five OR one zero five

25,7 = twenty-five point seven OR two five point seven

0,23 = zero point two three NOT zero point twenty-three

#### General rule:

- Before the decimal point: natural pronunciation or digit by digit
- After the decimal point: ALWAYS digit by digit

### 3.1.5 How to pronounce date values

Examples:

Today, tomorrow, yesterday = referenced date

The 20<sup>th</sup> of January = the 20<sup>th</sup> of January of the current year

The 20<sup>th</sup> of January 2018 = the 20<sup>th</sup> of January of the year 2018

## 3.2 Step by Step

### 3.2.1 Verify that a headset is connected

For speech recognition, a headset is required. A built-in microphone is OK for testing the application in an office environment but for production work a good headset with noise-canceling microphone is a necessity.

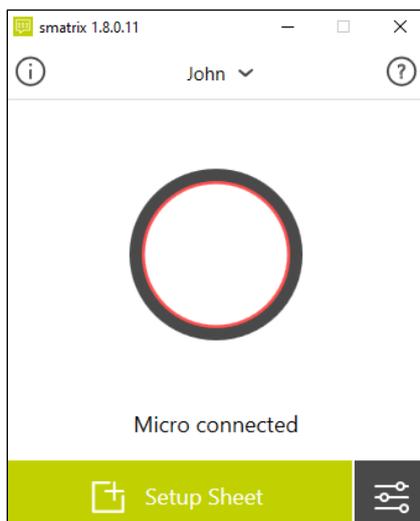
### 3.2.2 Start smatrix

Start the smatrix desktop application from the Windows menus or through the search function.

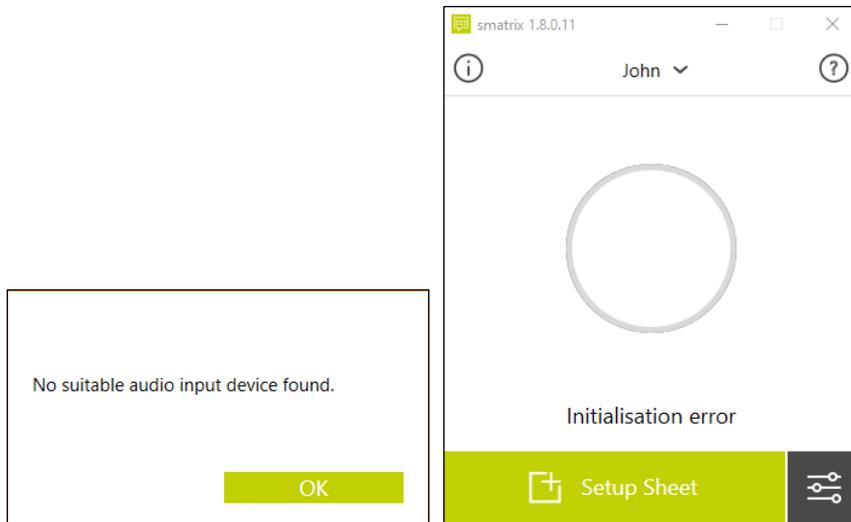
The smatrix splash screen will be shown for 10 seconds. On the splash screen, you will find the version of smatrix that is installed, and the license information.



Then the start screen appears with a "live" sound ring.

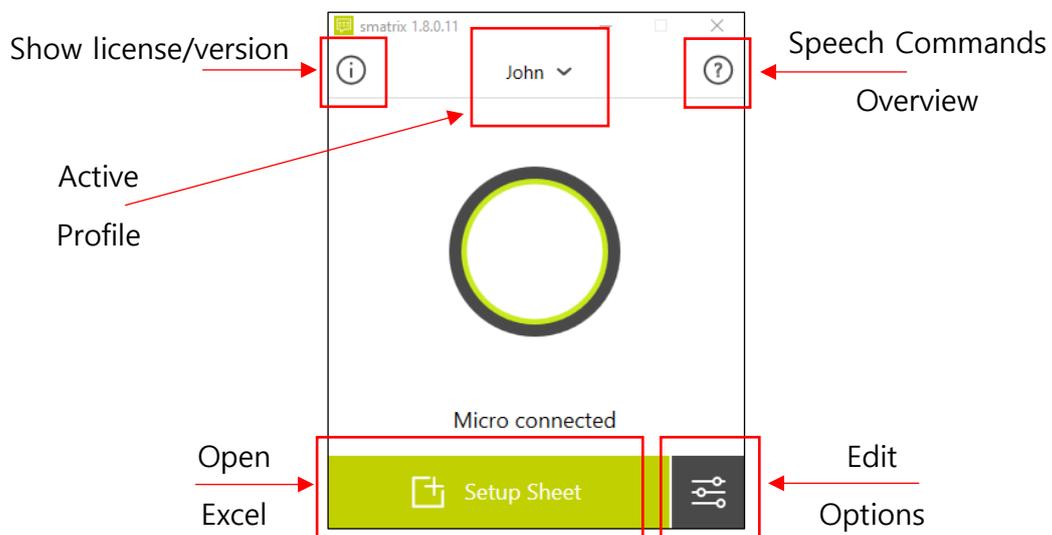


When no sound input device was found, an error message and a start screen with a stale gray sound ring are shown.



### 3.2.3 Discover the start screen features

The main feature on this screen is the live sound ring. The inner ring colors green when the sound level is in the correct range for speech recognition. Red means that the level is either too high or too low.

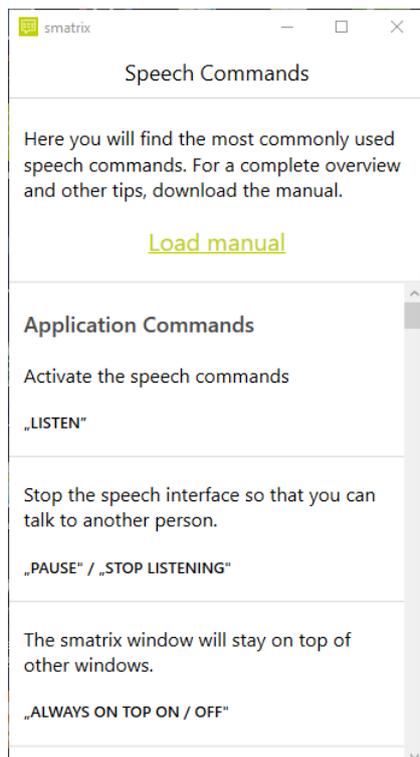


When you start smatrix for the first time, an initial profile is created associated with your windows account. The active profile can be edited or a new profile can be created through the "Edit Options" button.

If multiple profiles exist, the profile of choice can be selected from the "Profile" drop-down list.

The Excel sheet you want to work with can be opened and configured by activating the "Setup Sheet" button.

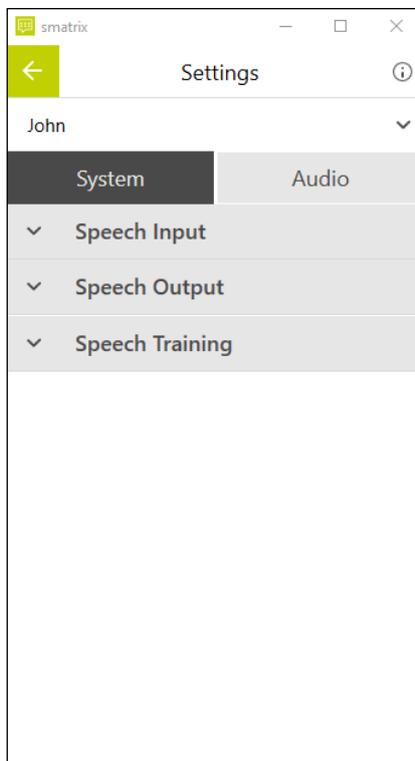
From the start screen, you can also get a speech command overview. This opens as a separate window that you can position somewhere on your screen as a reference.



Via the "Load manual" hyperlink, the present manual can be opened.

### 3.2.4 Check audio devices

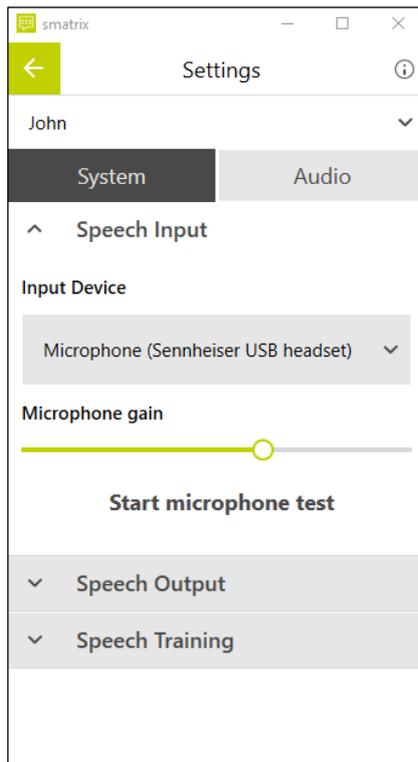
From the start screen, select the “Edit Options” button. Select the Audio tab on the “Settings” screen.



On this screen, a number of audio related options can be set.

The chosen options will be stored in the active profile when you exit the screen via the back-arrow.

Expand the “Speech Input” section:



Drop-down list of available sound input devices.

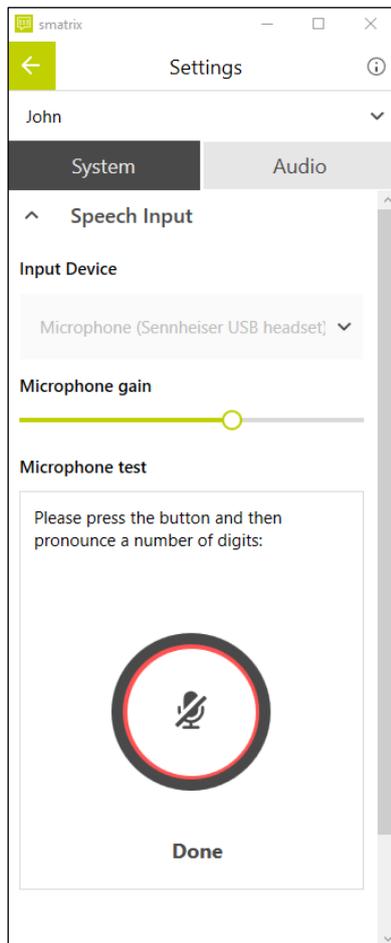
You can change the input volume with the slider.

It is important to set the input level at the right level. The speech recognizer is triggered when the sound input level reaches a minimum energy level.

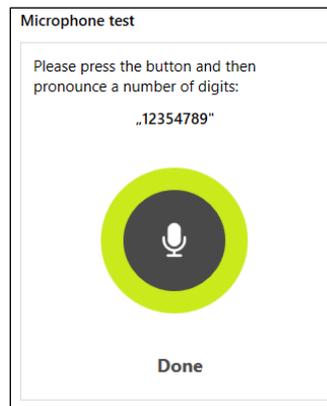
- If the volume is set too low, the user will have to speak loudly to trigger the speech recognizer which is very tiring after a while.
- If the volume is too high, the speech recognizer will be triggered by background noise and/or will not correctly estimate when the user stops speaking. This causes the speech recognizer to be out of sync with the user’s perception. Also, speech input can be deformed (clipped) due to too much amplification. This results in bad speech recognition quality.

In order to set the input volume correctly, a microphone test procedure is available.

Start the microphone test:



After pushing the “Microphone” button, you can say a number of digits. Observe the sound ring carefully.



As soon as you speak, the ring should turn green and remain green as long as you speak. Be careful to position the microphone correctly in the corner of your mouth, 1 finger-width away.

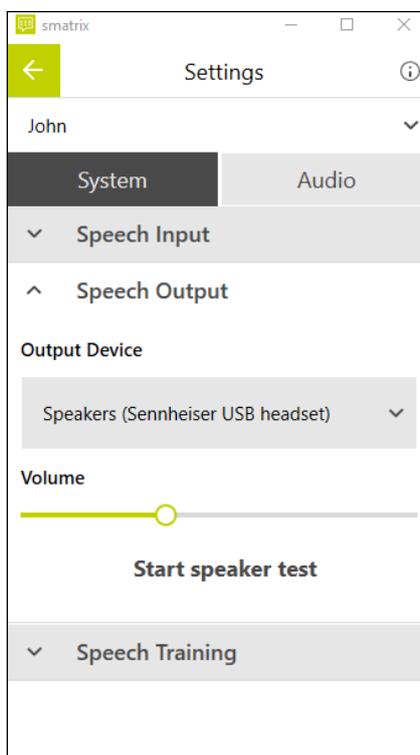
Adjust the volume to the minimum level where:

- 1: You do not have to make a voice effort to get a green level when you speak.
- 2: The sound ring does not turn green while you are silent due to environmental noise.

3: The sound ring should never turn red when you are talking.

Press “Done” when you have finished adjusting. You can always come back later to this screen to make further adjustments.

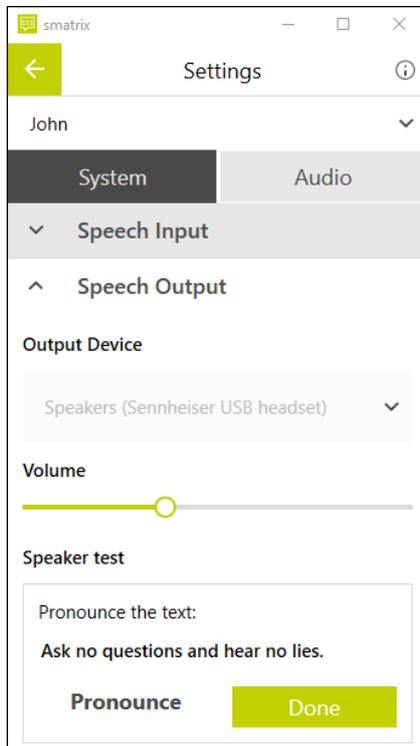
Minimize the “Speech Input” section and proceed to the “Speech Output” section.



Drop-down list of available output devices.

You can change the output volume with the slider. The slider affects the windows sound level for this device.

To set the most comfortable level for you, start the speaker test.



Push the "Pronounce" button to have the Speech Synthesis engine pronounce the phrase just above. You can alter this phrase to a phrase of your liking by typing over it.

Repeat this procedure several times and use the slider to set the volume to a comfortable level.

Note: while in data entry mode, you can still change the sound level with the "Louder" / "Softer" voice commands. These commands do not affect the amplification but instead influence the sound level generated by the speech synthesizer.

Use the back arrow  to save the changed settings and go back to the start screen.

### 3.2.5 Setup Sheet

Push "Setup Sheet" to go to the Excel setup form. The setup is proceeding along several steps.

In our examples, we will use sheets exported by ARM. They have well defined header data in the sheets and particular sheet names. See the Reference section for sheets with a user-defined layout.

### 3.2.6 Select Excel-file

**1. Select Excel**

Please select the Excel workbook and sheet into which the data will be entered.

Excel workbook

Choose Excel workbook ⋮

Smatrix remembers the directory and file name of the last Excel you opened.

After a file is selected, Excel will be started and the selected workbook is opened.

### 3.2.7 Select Excel sheet

Excel sheet

Ratings ▾

The sheet can be selected from the Smatrix drop-down list or by selecting the sheet in the opened Excel application.

The next action will be to configure the selected sheet. Configuring in this context means to specify which columns will be filled for each plot and what are the data ranges for each column.

The configuration data will be stored upon exiting this screen in a parameter file associated with the sheet.

The filename for this parameter file is: *excel\_file\_name.sheet\_name.params.xml*

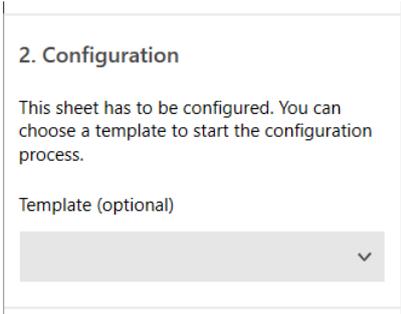
When you want to share the same configuration for a lot of sheets with essentially the same structure, the parameter file can be saved as a template.

### 3.2.8 Sheet autoconfiguration

When a sheet is opened, smatrix will check whether the sheet was already configured in the past based on the presence or absence of the *excel\_file\_name.sheet\_name.params.xml* file in the same directory as the Excel file.

If no parameter file is found, smatrix will do its best to autoconfigure the sheet. This autoconfiguration is never complete or it can be just plain wrong. Therefore, you can complete or override it in the next steps.

### 3.2.9 Sheet configuration with a template



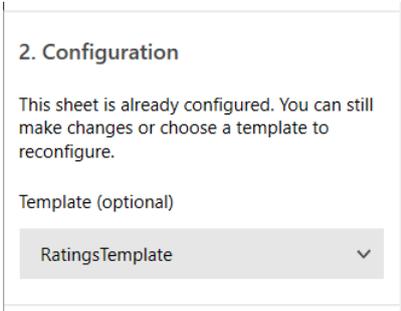
**2. Configuration**

This sheet has to be configured. You can choose a template to start the configuration process.

Template (optional)

If you want to reuse the parameters saved in a template, you can select the template here.

The parameters in the template will serve as an initial configuration. This configuration can be kept or modified in further steps.



**2. Configuration**

This sheet is already configured. You can still make changes or choose a template to reconfigure.

Template (optional)

### 3.2.10 Mapping of the identifiers

In this step we define which columns contain identifiers and what they are identifying. Also, we can specify which (if any) column contains subsample identifiers.

A special notation is available for columns that contain **multiple identifiers in the same column**. These identifier values **have to be** separated by a '/'.

First, we identify the column range for the identifiers. A columns range can be specified as:

C, E	columns C and E
F, Y-AA	columns F and the columns in range Y to AA
A/3	column A contains id-group with 3 identifiers separated by '/'
B, A	columns B and A

The sequence in which the columns are specified is relevant. For more information see 3.2.11.

If subsamples are collected, we need to specify the column containing the subsample numbers.

Some examples:

	A	B	C
1		Characteristic 1	Characteris
2	Parcel	0-100%	1,3,5,7,9
3	100		
4	101		
5	102		
6	103		
7	104		
8	105		
9	106		

#### Mapping IDs

ID column range

A

Subsample ID Column

	A	B	C
1	Subsamples	5	
2			
3			
4		Characteristic 1	Characteristic 2
5	Parcel	0-100%	1,3,5,7,9
6	100 1		
7	100 2		
8	100 3		
9	100 4		
10	100 5		
11	100 6		

Mapping IDs

ID column range

Subsample ID Column

	A	B	C	D
1				Characteristic 1
2	Object	Parcel	Sub	0-100%
3	ABCD-1	A		1
4				2
5				3
6		B		1
7				2
8				3
9		C		1
10				2
11				3

Mapping IDs

ID column range

Subsample ID Column

In the next step, we specify what those identifiers are actually identifying. Because smatrix has a voice-interface, it needs to know how the user will name the objects being surveyed: are they pots, plants, parcels, rows, treatments? The possible object names can be configured in the headers.xml file (see reference). When configuring, we make a choice by selecting a concept from the drop-down list. Smatrix will *try* to preconfigure the name by looking at the headers in the table.

Example:

	A	B	C
1		Characteristic 1	Characteris
2	Parcel	0-100%	1,3,5,7,9
3	100		
4	101		
5	102		
6	103		
7	104		
8	105		
9	106		

### Mapping IDs

ID column range ⓘ

Subsample ID Column ⓘ

Counting ⓘ

Column A - Name

By selecting the name “Parcel,Plot,Plot (Sub)”, you tell smatrix that the id’s in column A refer to parcels or plots.

This name is configured in headers\_en.xml as:

```
<subject id="S2" name="Parcel,Plot,Plot (Sub)">
  <pronounce>parcel</pronounce>
  <pronounce>plot</pronounce>
</subject>
```

Therefore, you will be able to navigate to a particular row in the table by saying ‘plot 103’ or ‘parcel one zero six’.

### 3.2.11 More on the mapping of the identifiers

#### *Hierarchical identifiers*

When multiple identifiers are mapped, the sequence of mappings represents a hierarchy.

The mapping “A (field), B (parcel in field), C (plant in parcel)” represents the concept that a field identified by a value in column A **contains** a number of parcels, each identified by a value in column B. Each parcel **contains** plants identified by values in column C.

The mapping “B (parcel), A (field in parcel), C (plant in field)” would assume that parcels contain fields which contain plants.

This hierarchical meaning has an effect on the navigation commands.

Given the first mapping, the command: "plant 5" will navigate to the plant with id-value **equal to 5 in the current** field & parcel. The command "parcel 3" will navigate to the first plant in the parcel with id-value **equal to 3** in the current field. All the other parcels with the same id-value in other fields are not considered.

The commands "next/previous plant 5" disregard the hierarchy and search **from the current sample on forwards/backwards** for one with a plant id-value **ending in '5'**

### *Identifier Groups (a.k.a. PIAF id's)*

An identifier group is a single column that contains a number of hierarchically structured ids. Let's assume that column A contains *field-no/parcel-no/plant-no* values e.g. "123/7/10" for field 123, parcel 7, plant 10. In the id specification, one would write this a "A/3" where 3 denotes the number of parts in each id-value. The separator is always '/'. Each part will have to be mapped to a name. The parts are noted as "A", "/A" and "//A" for 1<sup>st</sup>, 2<sup>nd</sup> and 3<sup>rd</sup> part.

<b>Column A</b> - Name
Treatment,TRT
<b>Column /A</b> - Name
Repetition
<b>Column //A</b> - Name
Parcel

**A unique feature of identifier groups** is the following:

You can say: "parcel 123 7" or "plant 123 7 10" to directly navigate to a particular parcel or plant. Note that the numbers in this command **have to be** spoken as **natural numbers!** Digit-by-digit pronunciation can't be supported as this would introduce ambiguities (e.g. "parcel 1 2 3" could be field 12, parcel 3 or field 1 parcel 23).

### *Alternative identifiers*

Sometimes identifiers are alternatives for one another (two different identifiers for the same sample). Consider the following schema A (field), B (parcel in field), C (plant in parcel), D (plant code) where D values **uniquely identify** each plant.

The id specification should then be written as "A, B, C, D/1"

The notation "D/1" represents a single part identifier-group. Identifier-groups always function as independent groups of identifiers. Inside an identifier-group, hierarchical semantics are again valid.

#### **3.2.12 Mapping of the criteria**

After configuring the identifiers, it's time to configure the criteria being evaluated during the survey.

First, we define which columns contain criteria to be evaluated. We identify the column range for the criteria. A column range can be specified as:

C, E	columns C and E
F, Y-AA	columns F and the columns in range Y to AA
D, F, E	columns D, F and E

Again, the autoconfiguration feature of smatrix will propose a range for you based on the content of the table. Correct this proposal if needed.

Note that the sequence of the columns does not have to be the regular sequence. The specified sequence will influence **the sequence in which criteria are evaluated**.

Example:

	A	B	C	D	E
1		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
2	Parcel	0-100%	1,3,5,7,9	50-120cm	1-500
3	100				
4	101				
5	102				
6	103				

**Mapping Criteria**

Criteria column range

As for the identifiers, also here we need to tell smatrix what concept each selected column represents. Is it a disease, a trait, a growth length? On top of that, we need to specify what kind of data will be entered in the cells of each column. Will it be numbers? In what range? Or colors? Which ones?

The available concepts are configured in the headers.xml file. The available data ranges are configured in the values.xml file. You will find more details about how to setup those files later in this manual. In the smatrix user interface, you can only select from the pre-configured concepts and data ranges.

The autoconfiguration feature of smatrix will try to match the headers in the excel sheet to subjects in the headers.xml file. Data ranges however have to be selected manually.

Example:

	A	B	C	D	E
1		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
2	Parcel	0-100%	1,3,5,7,9	50-120cm	1-500
3	100				
4	101				
5	102				
6	103				

**Mapping Criteria**

Criteria column range

B-E

Please give for each column the evaluation criterion and the value range.

Column B

Category

Generic

Criterion

Characteristic 1

Value range

Autoconfiguration recognized the 'Characteristic 1' header in the table and matched it to the criterion with the same name in category 'Generic'. The value range however still has to be selected.

Column B

Category

Generic

Criterion

Characteristic 1

Value range

0%-100%

This configuration of column B allows you to say 'characteristic 1 50 percent' to enter the value 50 in column B of the current row.

Criterion and data range selection directly impact the way you will address a column and enter data into a column. They do not do any data validation on data that is already present or is entered by hand.

Remark also that each criterion has an associated checkbox. By checking the box, we make the criterion active. This means that this criterion will be evaluated when we run the survey. This feature allows you to deactivate certain configured criteria for a particular survey.

### 3.2.13 Mapping of the criteria with subsamples

Consider the following table:

	A	B	C	D	E
1	Subsamples	5	10	5	1
2					
3					
4		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
5	Parcel	0-100%	1,3,5,7,9	50-120cm	1,3,5,7,9
6	100 1				
7	100 2				
8	100 3				
9	100 4				
10	100 5				
11	100 6				
12	100 7				
13	100 8				
14	100 9				
15	100 10				
16	110 1				
17	110 2				
18	110 3				
19	110 4				

For each parcel, 4 criteria are evaluated: Characteristic 1-4.

Characteristic 1 and 3 are evaluated by taking 5 subsamples, characteristic 2 requires 10 subsamples and characteristic 4 only 1.

ID column range ⓘ

Subsample ID Column ⓘ

Due to the fact that a subsample id column was identified, smatrix knows that subsampling is required. Therefore, smatrix opens up additional configuration fields for each criterion:

Column C

Category  
Generic

Criterion  
Characteristic 2

Value range ⓘ  
1 - 9

Subsample Group  
[Empty]

# Subsamples  
10

In addition to the value range, you can now also select a subsample group and enter a number of subsamples.

Criteria within the same subsample group always have the same number of subsamples.

Grouped criteria are evaluated together during the survey.

Suppose we assign characteristic 1 and 3 to the same group.

Column B

Category  
Generic

Criterion  
Characteristic 1

Value range  
0%-100%

Subsample Group  
Subsample Group 1

# Subsamples  
5

Column D

Category  
Generic

Criterion  
Characteristic 3

Value range  
50-120 (cm)

Subsample Group  
Subsample Group 1

# Subsamples  
5

Then the evaluation sequence will be the following.

	A	B	C	D	E
1	Subsamples	5	10	5	1
2					
3					
4		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
5	Parcel	0-100%	1,3,5,7,9	50-120cm	1,3,5,7,9
6	100 1				
7	100 2				
8	100 3				
9	100 4				
10	100 5				
11	100 6				
12	100 7				
13	100 8				
14	100 9				
15	100 10				
16	110 1				
17	110 2				
18	110 3				
19	110 4				

First, Subsample Group 1 is evaluated, alternatively entering samples for Characteristic 1 and Characteristic 3.

	A	B	C	D	E
1	Subsamples	5	10	5	1
2					
3					
4		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
5	Parcel	0-100%	1,3,5,7,9	50-120cm	1,3,5,7,9
6	100 1				
7	100 2				
8	100 3				
9	100 4				
10	100 5				
11	100 6				
12	100 7				
13	100 8				
14	100 9				
15	100 10				
16	110 1				
17	110 2				
18	110 3				
19	110 4				

After completing all 5 samples for Subsample Group 1, Characteristic 2 samples are collected and finally Characteristic 4 is collected.

Without grouping, the data collection would have proceeded as follows:

	A	B	C	D	E
1	Subsamples	5	10	5	1
2					
3					
4		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
5	Parcel	0-100%	1,3,5,7,9	50-120cm	1,3,5,7,9
6	100 1				
7	100 2				
8	100 3				
9	100 4				
10	100 5				
11	100 6				
12	100 7				
13	100 8				
14	100 9				
15	100 10				
16	110 1				
17	110 2				
18	110 3				
19	110 4				

All subsamples for each criterium are collected before proceeding to the next one.

### 3.2.14 Mapping of multi-level criteria

For some surveys it is desired to define a column as a combined (“multi-level”) criterion. You might for example have a number of columns that all intend to measure the same “Symptom 1”, but on different leaf levels.

	A	B	C	D	E	F	G	H
1	Plot	Symptom 1	Symptom 1	Symptom 1	Symptom 2	Symptom 2	Symptom 2	Weed 1
2	Plot	Leaf level F	Leaf level F-1	Leaf level F-2	Leaf level F	Leaf level F-1	Leaf level F-2	
3	1							
4	2							
5	3							
6	4							
7	5							

To allow such configuration, the “Multiple criterion levels” option needs to be checked:

Mapping Criteria

Criteria column range

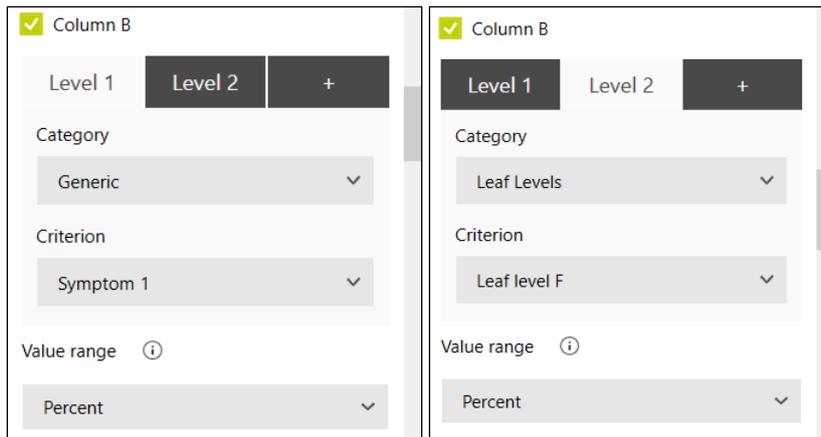
B-H

Multiple criterion levels ⓘ

If the autoconfiguration detected multi-level criteria, this will already be checked.

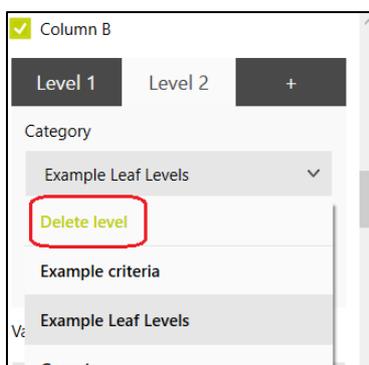
Otherwise you can configure this manually as well.

When this box is checked, each column is configurable on up to three levels. In the above example, "Symptom 1" is the highest level. To configure the leaf level, you can switch to the tab "Level 2" and select the correct criterion.



Note that the value range is independent of currently shown level. Modifying it will change it for the column.

If autoconfiguration did not detect the correct number of levels (maximum 3), you can also add another level using the "+" tab or remove a level by selecting the "delete" option in the drop-down menu of the category. This deletes the currently active tab (here "Level 2") and all levels below it.



**Note on navigation:**

The navigation can be now be used in a hierarchical manor similar as for the identifiers. If succeeding columns have the same criteria chosen on "Level 1", only the name of "Level 2" can be used to navigate between them. Using the above example:

User> "Leaf level F-1"

=> If current column is B or D it navigates to C

=> If current column is E or G it navigates to F

=> If current column is H, this is invalid as there is no column with "Weed 1" and "Leaf level F-1"

It is possible to say the name of "Level 1" of any multi-level Criterion. If this is not unique, it will navigate to the first column that has the matching name in "Level 1"

User> "Symptom 2"

=> If current column is B,C,D or H it navigates to E (first column with Symptom 2)

=> If current column is E,F or G it is ignored (Already in Symptom 2)

You can also concatenate the names of the levels to address a column in a unique way.

User> "Symptom 2, Leaf level F-1"

=> Always navigates to column F

### 3.2.15 Rows selection

Just as you may not desire to evaluate all columns (criteria) during a particular survey, you may wish to evaluate only certain rows (parcels, plants, pots).

So, you can specify a row range. One or multiple row ranges can be specified.

Examples:

11-52	From Excel row number 11 till Excel row number 52
11-20, 31-40	2 ranges, from 11 till 20 and from 31 till 40
11, 17, 19	rows 11, 17 and 19

Only the selected rows will be editable by voice during the survey.

Note: To address a certain row by voice you can speak its identifier.

8	008 Ass-Code:	A1	A1
9	009 Date:		
10	009 Plot (Sub)		
11	4 101		
12	10 102		
13	7 103		
14	5 104		
15	 105		

User> "Plot one hundred and four"

### 3.2.16 Parameters

This configuration section groups a number of parameters that will influence the survey process for this sheet. They are stored in the configuration file for this sheet (or in the template).

- Start cell

If possible, smatrix will guess the start cell. The cursor will be put in this start position the first time editing is done. When reopening a previously edited file, the cursor position will not be changed as long as it is within a valid row and column range.

- Unavailable value

A cell containing the unavailable value will be considered by smatrix as deliberately left open. Every data range contains the unavailable value. Therefore, you can always enter this value by saying 'unavailable' (or the equivalent in your language)

- Evaluation procedure
  - Evaluation (without subsampling)

8	008	Ass-Code:	A1	A1
9	009	Date:		
10	009	Plot (Sub)		
11	4	101		
12	10	102		
13	7	103		
14	5	104		
15		105		



In **single criterion mode**, the next cell is always in the same column. The cursor skips to the next in-range row. An explicit navigation command is needed to switch between columns.

In **multiple criteria mode**, the next cell is the one in the same row but in the next in-range column. When the last in-range column has been reached, the cursor skips to the next in-range row.

- Plot evaluation in subsample mode

9	009	ZeitpCod.:	A3	A3
10	009	Plot (Sub)		
11	6	101 1		
12	6	101 2		
13	6	101 3		
14	6	101 4		
15	6	101 5		
16	1	102 1		
17	1	102 2		
18	1	102 3		
19	1	102 4		
20	1	102 5		
21	7	103 1		
22	7	103 2		
23	7	103 3		
24	7	103 4		
25	7	103 5		
26	7	103 6		
27	7	103 7		
28	7	103 8		
29	7	103 9		
30	7	103 10		
31	7	103 11		
32	7	103 12		
33	7	103 13		

Single criterion mode functions as before.

In **multiple criteria mode**, the next cell is the next subsample for the current criterion. When all subsamples are evaluated, the next cell is the first subsample of the next in-range criterion. When all in-range criteria are evaluated, the next cell is the first subsample of the first in-range criterion of the next in-range object.

When criteria are grouped, the logic is the same except that the group is treated as one single criterium (see the example in 3.2.13).

- Advancement

In command mode, the cursor remains in the same cell until an explicit navigation command is given. Successive value-commands will overwrite the value in the current cell.

User> "Plot one hundred and four"

User> "15"

User> "50"                      50 overwrites the original value 15

User> Next                      Only now the cursor moves

In auto mode, the cursor moves to the next cell after a value is set with a value-command.

User> "Plot one hundred and four"

User> "15"                      After entering the value, the cursor moves to the next cell

User> "50"                      50 is written in the next cell

User> Next                      One cell is skipped

- Skipping cells

When Smatrix is instructed to jump to the next or previous cell, already filled cells can be skipped. Setting this switch to ON will enable the skipping of filled cells.

At run time, you can switch between the two modes with a voice command.

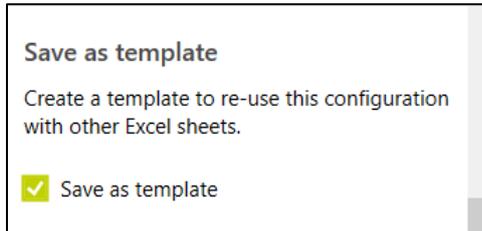
- Optional checks

Smatrix can verify that no cells are left empty. The check can be made at different occasions during the survey.

When one or more empty cells are detected, the location of the first one is read to the user.

### 3.2.17 Save as template

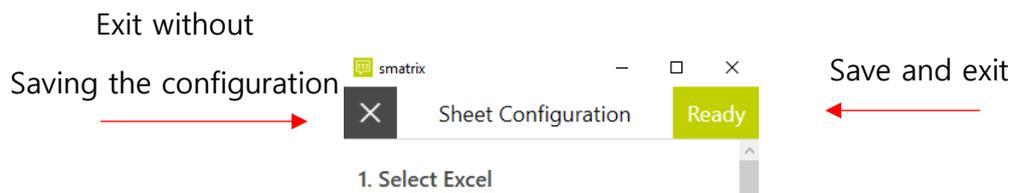
If you want to re-use the same configuration for other sheets you can store the current configuration as a template. This can be done by checking the option "save as template" at the end of the configuration:



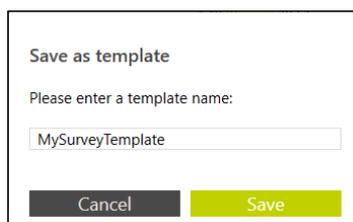
The actual storing will not be done when checking this box, but when saving the configuration using the “READY” button. A popup will be shown which asks you for the desired template name (see 3.2.18 “Save the configuration”).

### 3.2.18 Save the configuration

When the configuration is complete the Ready button changes from  to  and is now enabled. You can now:

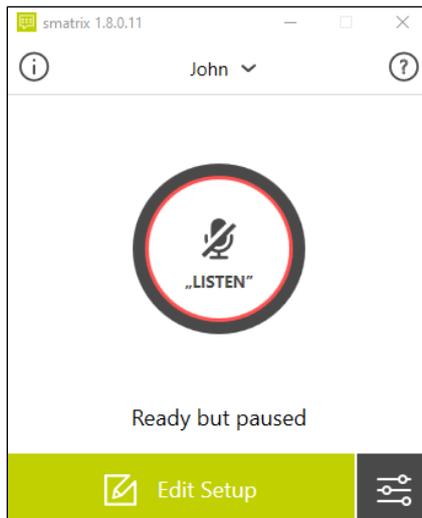


The “Ready” button saves the configuration data and brings you back to the start screen, ready for inputting data.



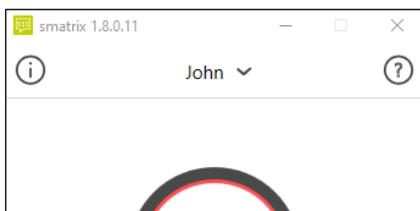
If the “save as template” box was checked in the configuration, a popup asks for the template name you want to use for storing the template.

Smatrix is now ready for data input. The configuration can be changed at any moment by using the “Edit Setup” button. Also, the “Edit options” button is still available for changing application options e.g. audio settings.



### 3.2.19 Input Data in Sheet

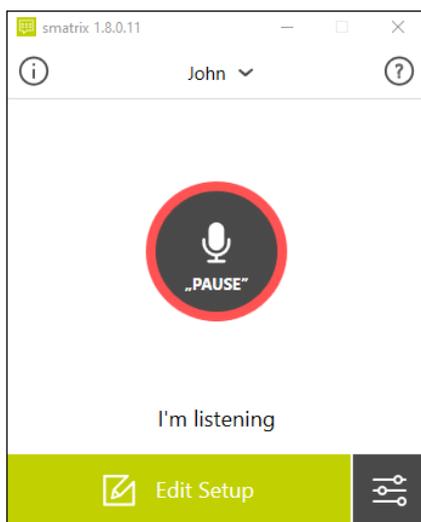
After leaving the sheet setup, smatrix is ready for speech input but still paused. We will now give a quick overview of the speech commands that are available.



This overview is also available through the help function. ←

### 3.2.20 Ready <-> Pause

**User> "Listen"** The application changes to the listening state, the cursor in the Excel sheet moves to the last saved position (if in range) or to the start cell otherwise.



	A	B	C	D
5	002	Target:	FUMOF	PAPRH
6	004	Crop:	BEAVA	BEAVA
7	007	Ass-Type.:	CONTRO	CONTRO
8	008	Ass-Code:	A1	A1
9	009	Date:		
10	009	Plot (Sub)		
11	4	101		
12	10	102		
13	7	103		
14	5	104		
15	1	105		
16	3	106		
17	12	107		

**User> "Pause"** The application changes to the paused state, no speech commands are executed until the user says "Listen" or pushes the "Listen" button.

### 3.2.21 Entering data

- User>** "twenty four"                      The user pronounces a number in the natural way.
- User>** "two four"                              The user spells a number digit by digit.
- User>** "characteristic 1, two four"      The cursor moves to the designated column and enters the value there.
- User>** "zero point nine"                      A number with decimals. Decimals are ALWAYS spelled!
- User>** "twenty three point two five"      A number with decimals. Decimals are ALWAYS spelled!

**User>** "correction, twenty-five"              When plot advancement is in command mode, correction is done by repeating the correct value before navigating to another cell. When plot **advancement is automatic**, the cursor has already moved on when the user hears the mistaken data entry. The correction prefix will make the cursor backtrack to the previous cell, then correct the value and move forward again.

When plot advancement is automatic, the equivalent command sequence of "correction, twenty-five" is:

**User>** "back" or "correction"

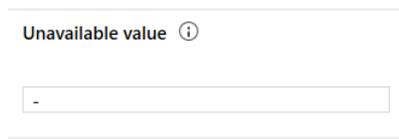
**User>** "twenty-five"

### 3.2.22 Entering the 'not available' value

User> "unavailable"

User> "not available"

All configured data ranges contain one extra value i.e. the 'unavailable' value. This value is used to indicate that a correct value cannot be given. The actual value entered in the cell can be configured in the 'Sheet configuration' section in the field 'Unavailable value'.



### 3.2.23 Entering multiple values

This command is the equivalent of entering several times the same value.

User> "ten times five"                      The next N (10 in the example) cells in the evaluation order are filled with the same value (5 in the example). This command is the equivalent of repeatedly entering the same value. Data entry stops:

- When the value is not valid for a criterion
- At the end of the row-range

User> "five times 'unavailable'"              The next 5 cells will receive the 'unavailable' value.

If the "skipping cells" option is on, only empty cells will receive a value. Otherwise, this command **may overwrite existing values**.





**System**> Fill all empty values for criterion 'characteristic 2' with '20'? For filling a criterion, a confirmation is asked!

**User**> "no" Oops, a misunderstanding. All empty cells (all subsamples) for the named criterion would have been filled with 20.

**User**> "fill characteristic 2 with 12"

**System**> Fill all empty values for criterion 'characteristic 2' with '12'? This is better!

**User**> "yes" Now, all empty cells for this criterion are filled with the value 12.

### 3.2.26 Undo and Redo

When you realize that your last command was a mistake, you can use the 'undo' command to go back in time. 'redo' repeats the action of the last undone command.

**User**> "ten times five" Oops, I just overwrote valuable data

**User**> "undo" Fine! What did I do again?

**User**> "redo" That was bad!

**User**> "undo" Back where we started

### 3.2.27 Navigating

- User>** "parcel one hundred and four"  
with parcel number (ending in) 104
- The cursor is advanced to the **first** row
- User>** "next"
- The cursor is moved to the next in-range cell. The behavior depends on the evaluation mode: single criterion / multiple criteria.
- User>** "previous" or "back"
- The cursor is moved to the preceding in-range cell. The behavior depends on the evaluation mode: single criterion / multiple criteria.
- User>** "next parcel"
- The cursor is advanced to the first row with a different parcel number than the current one.
- User>** "next parcel 4"
- The cursor is advanced to the **next** row with a parcel number ending in 4.
- User>** "next subsample"
- The cursor is advanced to the next subsample for the current criterion.
- User>** "characteristic 1"
- The cursor is positioned in the column corresponding to "characteristic 1".
- User>** "subsample 19"
- The cursor is moved to subsample 19 for the current criterion.
- User>** "parcel 10 characteristic 1"
- The cursor is moved to the first row with a plot number 10 and positioned in the column corresponding to "characteristic 1".



### 3.2.30 Setting options

**User>** "barge-in on / off"                      When barge-in is on, the user can interrupt a spoken message by giving the next speech command. If barge-in is off, the user has to wait until the message is read before starting to talk.

**User>** "always on top on / off"                      When always on top is on, the smatrix window will position itself always in front of other windows, hence remaining visible at all times.

**User>** "single criterion"                      In single criterion mode, the next cell is the next in-range cell in the same column. **Only one criterion is evaluated.**

	A	B	C	D	E	F
1						
2				Yellow Rust	Brown Rust	Black Rust
3	Treatment	Tray	Plant			
4	1	100	1			
5			2			
6			3			
7			4			
8			5			
9	2	101	1			
10			2			
11			3			
12			4			
13			5			
14	3	102	1			
15			2			
16			3			
17			4			
18			5			

**User>** "multiple criteria"                      In multiple criteria mode, the next cell is the next in-range cell in the same row or the next row if the last column was reached.

In the example, all criteria for a plant are evaluated before moving to the next plant. This command is equivalent to 'multiple criteria per plant'

	A	B	C	D	E	F
1						
2				Yellow Rust	Brown Rust	Black Rust
3	Treatment	Tray	Plant			
4	1	100	1			
5			2			
6			3			
7			4			
8			5			
9	2	101	1			
10			2			
11			3			
12			4			
13			5			
14	3	102	1			
15			2			

**User>** “multiple criteria per tray”                      In multiple criteria mode per id-level, one criterium is handled for all samples within the same id-level before moving to the next criterium or the following sample with a different id at the given level.

In the example, all plants in the same tray are evaluated for each criterion before moving to the next tray.

	A	B	C	D	E	F
1						
2				Yellow Rust	Brown Rust	Black Rust
3	Treatment	Tray	Plant			
4	1	100	1			
5			2			
6			3			
7			4			
8			5			
9	2	101	1			
10			2			
11			3			
12			4			
13			5			
14	3	102	1			
15			2			

**User>** “automatic progress on / off”                      When automatic progress is on, the cursor moves to the next cell after a value is entered. When off, the cursor will only move after an explicit navigation command.

**User>** "skipping cells on / off"                      When skipping cells is on, the next/previous commands will skip filled cells. When off, the cursor will not skip cells. This is the old (pre-1.5) and also the default behavior.

Note: Options set by speech commands are not saved between sessions. To set saved options, use the "Edit Options" button.

### 3.2.31 Inserting a new column

**User>** "insert column after 'yellow rust'"      This command inserts a new column in the excel sheet after the column associated with criterion 'yellow rust'. You are put back in configuration mode to associate the new column with a criterion and a data range.

**User>** "insert 'black rust' before 'yellow rust'"      This command inserts a new column in the excel sheet after the column associated with criterion 'yellow rust'. This column is associated with the criterion 'black rust' and its default data range, if that is configured in `headers_lang.xml`. You are put back in configuration mode to verify the setup.

**User>** "insert 'black rust' with levels before 'yellow rust'"      This command is only valid when using multi-level criteria. It inserts multiple columns that have same levels as the mentioned 'yellow rust', just with 'black rust'. Consider an example where two adjacent columns have 'yellow rust' on level 1 and level 2 is once "level F" and once "level F-1".

	A	B	C	D	E
1	Plot	Symptom 1	Symptom 2	Yellow Rust	Yellow Rust
2	Plot			Leaf level F	Leaf level F-1
3	1				
4	2				
5	3				
6	4				
7	5				

The above command would not introduce one but rather two new columns. Both have level 1 as 'black rust'. Level 2 would be once "level F" and once "level F-1".

	A	B	C	D	E	F	G
1	Plot	Symptom 1	Symptom 2	Black Rust	Black Rust	Yellow Rust	Yellow Rust
2	Plot			Leaf level F	Leaf level F-1	Leaf level F	Leaf level F-1
3	1						
4	2						
5	3						
6	4						
7	5						

Everything else is like the above command without "with levels". You are put back in configuration mode to verify the setup.

### 3.3 Counting Survey

#### 3.3.1 What is it?

During a normal survey with subsampling, for each object a number of subsample values is stored for each evaluated trait.

Example: One wants to evaluate the wireworm/slugs damage of potatoes. From each identified lot of potatoes, a number (100 e.g.) of potatoes are randomly selected and for each potato the number of wireworm/slug holes is counted and recorded as a subsample.

	A	B	C	D	E	F
1	<b>Hoogstraten 2019 Potato Survey</b>				Date: 10.10.2109	
2						
3						
4				<b>Wireworm #holes</b>	<b>Slug #holes</b>	
5	<b>Lot Number</b>	<b>Subsample</b>				
6						
7	1a	1		0	0	
8		2		3	0	
9		3		0	0	
10		4		1	1	
11		5		0	0	
12		6		5	0	
13		7		0	1	
14		8		1	0	
15		9		1	0	
16		...		1	2	
17		99		2	0	
18		100		0	0	
19	1b	1		0	1	
20		2		0	2	
21		3		0	2	

From those subsamples, a distribution can be derived i.e. what percentage of potatoes have a number of damages in particular ranges, say 0, 1 or 2, 3 to 5 and more than 5. This requires a post-processing step on the subsample data.

A counting survey established this distribution directly by counting the number of subsamples with trait values in each of a number of pre-defined ranges. The actual values are not stored. In the Excel sheet, one column is assigned to each trait-value range combination.

	A	B	C	D	E	F	G	H	I	J	K
1	# Subsamples	100	100	100	100		100	100	100		100
2		Wireworm damage					slug damage				
3	Plot	class A	class B	class C	class D	Total	class A	class B	class C	class D	Total
4		0	1-2	3-5	>5		0	1-2	3-5	>5	
5	1	33	26	26	15	100	62	20	10	8	100
6	2	22	24	28	26	100					0
7	3					0					0
8	4					0					0
9	5					0					0
10	6					0					0

When judging a potato, the surveyor will either say the exact number of damages detected or a range identifier (depending on the range configuration). The corresponding count will then be incremented.

### 3.3.2 Configuration

The following data items have to be defined:

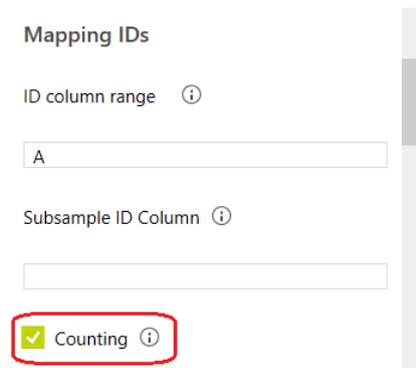
- Which traits are counted?
- How many subsamples are evaluated for a particular trait?
- Which columns are used to represent the counting classes?
- What utterances will classify a subsample in a class and hence increment the counter for that class?

We will treat each of those questions now.

- For each trait to be counted a subsample group is selected and all counting columns for that trait are assigned to the same subsample group.
- The subsample group count determines the number of subsamples.
- For each of the columns in the subsample group, the associated data range determines the utterances that classify a subsample in a particular class.

For the above example:

One has to select the subsample counting option:



Mapping IDs

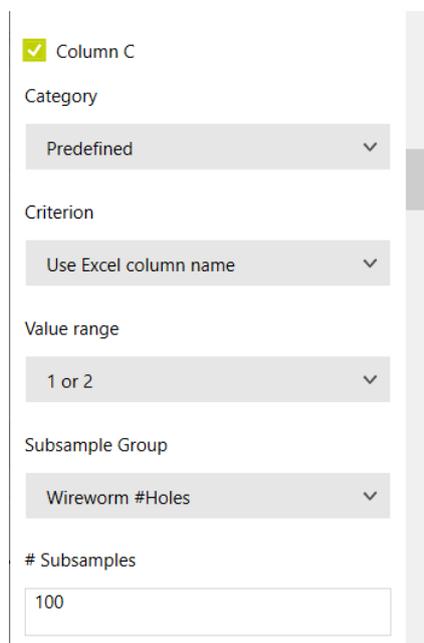
ID column range ⓘ

A

Subsample ID Column ⓘ

Counting ⓘ

Then for each column belonging to the wireworm damage counting, an appropriate data range is selected.



Column C

Category

Predefined ▼

Criterion

Use Excel column name ▼

Value range

1 or 2 ▼

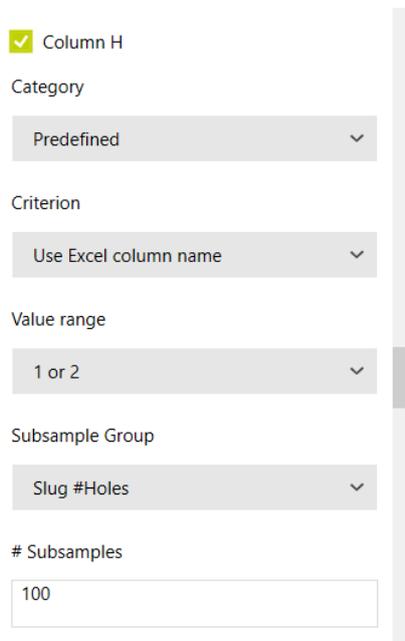
Subsample Group

Wireworm #Holes ▼

# Subsamples

100

Similar for the slug damage counting columns:



Column H  
 Category  
 Predefined  
 Criterion  
 Use Excel column name  
 Value range  
 1 or 2  
 Subsample Group  
 Slug #Holes  
 # Subsamples  
 100

Of course, appropriate entries have to be created in `values_lang.xml` and `headers_lang.xml` for the particular data ranges and subsample groups to be selectable. See section 4.2 for configuration information.

**Important:** The data ranges that are used within the same subsample group may not overlap, because they are recognizable in parallel! If they would overlap, recognition results would be ambiguous leading to unpredictable behavior.

Note that subsample group names can be spoken for navigation. The wording depends on the definition of the subsample group. To be able to say “slug damage” for the “Slug #Holes” subsample group, you need to configure a pronounce statement for it like

```
<topic id="{SUBSAMPLEGROUP}" name="Subsamplegroups">
```

...

```
<subject id="SSGSLUG" name="Slug #Holes">
```

```
<pronounce>slug damage</pronounce>
```

</subject>

</topic>

### 3.3.3 Voice commands for counting

**User>** "slug damage"                      To select the slug damage counting group.

**User>** "2"                                      A value ('2') of one of the group's data ranges. The value in the associated column will be incremented by 1.

**User>** "10 times class A"                      A repetition of a value ('class A') of one of the group's data ranges. The value in the associated column will be incremented by the amount given (10 in this example).

**User>** "fill with no damage"                      A repetition of a value ('no damage') of one of the group's data ranges. The value in the associated column will be incremented by the amount needed to complete the number of subsamples for this group.

### 3.4 Creating a user profile

From version 1.8 onwards, training of the recognition engine is not required any longer. User profiles only store user preferences configured via the System tab reachable via the "Edit Options" button.

## 4 Customizing smatrix

This section of the manual explains the aspects of the application that can be changed for all users of the application. They are not part of the user profile.

### 4.1 Editing smatrix defaults

Note: Smatrix defaults can only be edited by a person with administrator privileges.

The following defaults can be edited in the file: “C:\Program Files (x86)\Smatrix\smatrix.exe.config”

Parameter	Description
Language	Default smatrix language (“en-GB” or “en-US” for English, “de-DE” for German, “fr-FR” for French, “nl-NL” for Dutch, “es-ES” for Spanish). A user can set a different preferred language in the options.
WaveLogging	“True” or “False”. This options switches recordings by default on or off. In some countries, it may not be legal to set this to “True”.
Bargeln	“True” or “False”. Default value for the Barge-In option.
AlwaysOnTop	“True” or “False”. Default value for the Always On Top option.

## 4.2 Customizing criteria names and data ranges

During data collection, one or more criteria are evaluated for a particular object (plot, plant, pot or for that matter any surveyed object).

Smatrix is hence agnostic about what type of object you are surveying. During the configuration process, you select which type of object is identified by the identifiers embedded in the sheet. Therefore, the list of available object names can be configured.

A criterion can be the presence of a disease, the presence of insects, the effect of a treatment, presence of weeds ... The actual criteria depend on the line of business of the smatrix user. Therefore, the list of criteria names can be customized.

A criterion evaluation results in a count, a percentage, a code, a text ... Therefore, it is clear that also the available data ranges should be customizable for a particular business.

Obviously, item type names, criteria names and also data ranges can be language dependent.

Templates for both criteria names, object names and data ranges are available in “C:\ProgramData\smatrix\templates”

Note: Do not change any files in this directory because they will be overwritten when installing newer versions of smatrix or by de- / re-installing the current version

Customized criteria and data files have to be stored in “C:\ProgramData\smatrix\data”

**Note:** The folder “C:\ProgramData\smatrix\data\configuration templates” will be used to store user defined templates i.e. parameter files are stored here when a user saves a configuration as a template.

Files stored under “C:\ProgramData\smatrix\data” will not be removed or overwritten when smatrix is de- / re-installed.

#### 4.2.1 Object names

Object names are stored in the headers.xml file. Language specific versions can be stored in headers\_*lang*.xml (*lang* being 'de' for German, 'en' for English, 'fr' for French, 'nl' for Dutch, 'sp' for Spanish). Headers.xml is the default file if no language specific file is found.

Object names are represented by a <subject> xml-element. They are grouped under a <topic> xml-element with the reserved id “{SAMPLE}”.

```
<topic id="{SAMPLE}" name="Sample">
```

```
    ... here all object name <subject> entries
```

```
</topic>
```

Each <subject> entry represents an object name that can be chosen by the user when configuring the sheet.

```
<subject id="S2" name="Parcel,Plot,Plot (Sub)">
```

```
    <pronounce>parcel</pronounce>
```

```
    <pronounce>plot</pronounce>
```

```
</subject>
```

Subjects have an id. This id must be unique within the enclosing topic. One should carefully choose these and keep them unchanged. Once a sheet configuration has been made based on the content of a headers file, the ids are used to link identifier columns to the corresponding object name. Changing them will make the configuration incorrect or unusable. Also, subject ids should be consistent between languages. If you want to be able to share the same configured sheet between operators speaking a different language, language specific header files should be consistent!

Subjects have a name attribute. The value of the name attribute is used to display this object name in the drop-down lists in the smatrix user interface.

In the example "Parcel,Plot,Plot (Sub)", there are 3 names in here separated by a comma. You can consider this as one name "Parcel" and two aliases "Plot" and "Plot (Sub)". The name and the aliases will be used during autoconfiguration.

Object names have to be pronounceable because they are used to navigate by voice to a particular object. When aliases were added or names are not pronounceable, <pronounce> elements must be added. When you add <pronounce> elements, **only those** will be valid pronunciations. So in the above example, the only valid pronunciations for the object name "Parcel,Plot,Plot (Sub)" are "parcel" and "plot".

Multiple equivalent definitions could have been:

```
<subject id="S2" name="Parcel"/>
<subject id="S3" name="Plot"/>
<subject id="S4" name="Plot (Sub)">
    <pronounce>parcel</pronounce>
    <pronounce>plot</pronounce>
</subject>
```

However, the first definition is preferable because it is shorter and expresses better that it is all about the same type of object and it is easier to maintain.

Subsample groups can be named as well using <subject> xml-elements. They are grouped under a <topic> xml-element with the reserved id "{SUBSAMPLEGROUP}".

```
<topic id="{SUBSAMPLEGROUP}" name="Sample groups">
```

... here all subsample group name <subject> entries. In case a subsample group is supposed to be used in a counting survey (see section 3.3) it makes sense to specify pronounce-statements within each subject, as the subsample group name is recognizable.

```
</topic>
```

#### 4.2.2 Criteria names

Criteria names are stored in the headers.xml file. Language specific versions can be stored in headers\_*lang*.xml (*lang* being 'de' for German, 'en' for English, 'fr' for French, 'nl' for Dutch, 'sp' for Spanish). Headers.xml is the default file if no language specific file is found.

Criteria are represented by a <subject> xml-element. They are grouped in categories. Each category is represented by a <topic> xml-element.

```
<headers>
```

```
<topic id="GENERIC" name="Generic">
```

```
<subject id="CHAR1" name="Characteristic 1" range="NUMBER"/>
```

```
<subject id="CHAR2" name="Characteristic 2"/>
```

... other subjects ...

```
</topic>
```

```
<topic id="SPECIAL" name="Special">
```

```
<subject id="SPEC1" name="SZFG6">
```

```
<pronounce>Special 6</pronounce>
```

```
</subject>
```

```
<subject id="SPEC2" name="SZFG7">
```

```
    <pronounce>Special 7</pronounce>
```

```
</subject>
```

... other subjects ...

```
</topic>
```

```
</headers>
```

**Note on id's:** various elements have id's. One should carefully choose these and keep them unchanged. Once a configuration has been made based on a headers file, the id's are used to link sheet columns to the corresponding criterion. Changing them will make the configuration incorrect or unusable. Topic ids have to be unique within the file. Subject ids have to be unique within the particular topic.

**Note on names:** Both topics and subjects have a name attribute. The value of the name attribute is used to display this category (topic) or criterion (subject) in the drop-down lists in the smatrix user interface.

**Note on pronunciations:** Criteria names (not so for category names) have to be pronounceable because they are used to navigate by voice to a particular column. In case they are not or if alternative pronunciations are required, <pronounce> elements can be added. When you add <pronounce> elements, **only those** will be valid pronunciations

**Note on range attribute:** Criteria subjects can have an optional range attribute. The value of the range attribute should be an id associated with a value range topic (see following heading). This value range topic will be the default value range for the given criterium.

### 4.2.3 Value Ranges

Value ranges are stored in the values.xml file. Language specific versions can be stored in values\_*lang*.xml (*lang* being 'de' for German, 'en' for English, 'fr' for French, 'nl' for Dutch, 'es' for Spanish, 'pt' for Portuguese). Values.xml is the default file if no language specific file is found.

Value Ranges are represented by a <topic> xml-element. Values in a value range are represented either by generators (<generator> xml-element) or by <subject> xml-elements or by a mixture of both.

A special class of value ranges are the built-in ones. They have no explicit values defined because these are built-in.

The only currently available built-in value range is the date range. It is identified by the id attribute value "DATE". The name attribute value can be freely chosen.

```
<topic id="DATE" name="Datum"/>
```

**Note:** a more powerful way of specifying Date ranges is to use a Generator of type Date.

#### Numeric Data Range

A user defined value range starts with a <topic> xml-element. The id and name attributes can be freely chosen as long as the id is unique within the file.

```
<topic id="MyDataRange" name="0.0-9.9, 10, 20, ... 100 % + Perfect">
```

... here all <subject> and <generator> entries

```
</topic>
```

The values from 0.0% to 9.9% are defined with a generator element. A generator element is like an instruction to generate values without having to define them one by one.

```
<generator type="NUMBER" from="0" to="9" by="1" decimals="1">  
  <suffix name="percent" optional="true" include="no"/>  
</generator>
```

We specified that the generator is of type NUMBER, that the whole number part goes from 0 to 9 with intervals of 1. This would generate the sequence 0,1,2...9. However, we also want 1 decimal after the comma, therefore the decimals attribute.

**Note:** From version 1.7.1.7 a new notation is supported:

The parameters `from`, `to` and `by` can now have decimals after the point/comma (depending on your locale)

```
<generator type="NUMBER" from="0,0" to="9,9" by="0,1"/>
```

The old parameter `decimals` (re)defines `by` when `by` is not defined or has no decimals after the point/comma.

```
<generator type="NUMBER" from="90" to="100" decimals="2"/>
```

 is equivalent to

```
<generator type="NUMBER" from="90" to="100" by="0,01"/>
```

```
<generator type="NUMBER" from="90" to="100" by="1" decimals="2"/>
```

 is equivalent to

```
<generator type="NUMBER" from="90" to="100" by="0,01"/>
```

The number of decimals allowed is the maximum of the decimals in `from`, `to` and `by`.

```
<generator type="NUMBER" from="0,3" to="1" by="0,01"/>
```

 will use 2 decimals.

We also want to allow the user to say the word 'percent' after the numeric value.

Therefore, we add a `<suffix>` xml-element inside the generator. We specify that the percent addition is optional. The user can say it but is not obliged. The `include`-attribute is set to 'no' to indicate that the word 'percent' does not have to be added in the result that will be stored in Excel.

Suppose we wanted to add the "%" -sign concatenated to the numeric value. Then we could define the generator either as:

```
<generator type="NUMBER" from="0" to="9" by="1" decimals="1">
```

```
<suffix id="%" name="percent" optional="true" include="yes" action="concatenate"/>
```

```
</generator>
```

Or:

```
<generator type="NUMBER" from="0" to="9" by="1" decimals="1">
```

```
<suffix name="%" optional="true" include="yes" action="concatenate">
```

```
<pronounce>percent</pronounce>
```

```
</ suffix>
```

```
</generator>
```

When the user now says "9 comma 3", the text string "9.3%" will be send to Excel. How excel displays this depends on formatting rules defined in the sheet.

**Note:** smatrix always sends numeric values to Excel **except** if there is a prefix or suffix. Then a text string is sent to Excel.

To generate the 10,20 ... 100% values, we define another generator.

```
<generator type="NUMBER" from="10" to="100" by="10">
```

```
<suffix name="percent" optional="true"/>
```

```
</generator>
```

Remark that we did not add the `include="no"` on the suffix. That is because this is the default.

As a last value, we wanted to add the word "Perfect" as a synonym for the value 100.

We can do that by adding a `<subject>` xml-element to the definition.

```
<subject name="100">
```

```
<pronounce>perfect</pronounce>
```

```
<pronounce>gorgeous</pronounce>
```

```
</subject>
```

We defined both perfect and gorgeous as synonyms for the value 100.

The complete data range definition then looks like this:

```
<topic id=" MyDataRange " name="0.0-9.9, 10, 20, ... 100 % + Perfect">
```

```
  <generator type="NUMBER" from="0" to="9" by="1" decimals="1">
```

```
    <suffix name="percent" optional="true"/>
```

```
  </generator>
```

```
  <generator type="NUMBER" from="10" to="100" by="10">
```

```
    <suffix name="percent" optional="true"/>
```

```
  </generator>
```

```
  <subject name="100">
```

```
    <pronounce>perfect</pronounce>
```

```
    <pronounce>gorgeous</pronounce>
```

```
  </subject>
```

```
</topic>
```

**Note on id's:** various elements have ids. One should carefully choose these and keep them unchanged. Once a configuration has been made based on a values file, the ids are used to link sheet columns to the corresponding value range. Changing them will make the configuration incorrect or unusable. Topic ids have to be unique within the file.

**Note on topic names:** The data ranges (topics) have a name attribute. The value of the topic name attribute is used to display this data range in the drop-down lists in the smatrix user interface. To avoid confusing the user, it is best to make the topic names self-explanatory and unique

<generator> elements generate a series of values. Generators of type NUMBER generate numbers within a range bounded (inclusively) by the 'from' and 'to' attributes. An optional 'step' value can be specified. An optional number of 'decimals' can be added. Also, a 'suffix' or 'prefix' can be specified. The suffix or prefix will by default not be included in the value. A suffix can have alternative pronunciations.

### Text Data Range

A range does not have to be numeric. It can be a mix as in the example above or only text values as in the example below.

```
<topic id="DAMAGE" name="animal damage">
```

```
  <subject name="rodents">
```

```
    <pronounce>rats</pronounce>
```

```
    <pronounce>mice</pronounce>
```

```
  </subject>
```

```
  <subject name="pets">
```

```
    <pronounce>dogs</pronounce>
```

```
    <pronounce>cats</pronounce>
```

```
  </subject>
```

```
</topic>
```

### Note on pronunciations:

Numeric values have their default pronunciations.

A subject representing a data value has to have a pronounceable name. If not, or if alternative pronunciations are required, <pronounce> elements have to be added. When you add <pronounce> elements, **only those** will be valid pronunciations. In all cases the id attribute value (or in absence of the id attribute the name attribute value) is the value to be entered in the sheet.

In the example, when saying "dogs", the value "pets" will be entered in the sheet.

### Date data range

A range for entering dates can be defined as said above with the built-in DATE topic

```
<topic id="DATE" name="Datum"/>
```

However, there is another way to define the entering of dates in a more flexible way.

Suppose we want to allow the user either to say the date in the usual way or the day number in the year.

This involves the use of a <generator> of type 'DATE'.

As always, we first add a topic:

```
<topic id="MyDates" name="Date as DayOfYear">
```

... here all <subject> and <generator> entries

```
</topic>
```

Remember that the name of the topic will be shown in the drop-down list in the smatrix user interface.

Now we add a generator for the dates:

```
<generator type="DATE" format="DayOfYear"/>
```

With this definition, the user can say 'Tomorrow' or '4<sup>th</sup> of July' or 'January 1<sup>st</sup> 2018' and the corresponding day of the year will be entered in Excel.

If we want to allow the user to say the day number directly, we can add a generator for that purpose.

```
<generator type="NUMBER" from="1" to="366" by="1"/>
```

So, our data range is defined as:

```
<topic id="MyDates" name="Date as DayOfYear">
  <generator type="DATE" format="DayOfYear"/>
  <generator type="NUMBER" from="1" to="366" by="1"/>
</topic>
```

A generator of type DATE can also have a prefix and or a suffix.

Consider the following definition:

```
<topic id="MyDates" name="Date as DayOfYear">
  <generator type="DATE" format="DayOfYear">
    <prefix name="Eli" optional="true" include="ifsaïd" action="word">
      <pronounce>eliminated</pronounce>
    <pronounce>eliminated on</pronounce>
```

```

    </prefix>

</generator>

<generator type="NUMBER" from="1" to="366" by="1">

    <prefix name="Eli" optional="true" include="ifsaid" action="word">

        <pronounce>eliminated</pronounce>

        <pronounce>eliminated on</pronounce>

    </prefix>

</generator>

</topic>

```

With this definition:

User says> 1 <sup>st</sup> of January	Excel receives: 1
User says> eliminated on 1 <sup>st</sup> of January	Excel receives: 'Eli 1'
User says> eliminated 255	Excel receives: 'Eli 255'
User says> 255	Excel receives: 255

In this case, the value 'Eli' and the value '1' are not concatenated. Instead because of the `include="word"`, a space is added between the 2 values.

### **Code data range**

The objective of this data range is to enter a code. A code is a sequence of symbols (letters, numbers, other characters) that is entered by voice by pronouncing a sequence of words, each representing one of the symbols.

When somebody shouts : “Papa Bravo Tango”, we know that it stands for the code “PBT” using the NATO alphabet.

To define such a range we use a generator element with type “CODE”.

```
<generator type="CODE" format="A,AA,AAA" A="N,H,D,K,G,U,C,B,F,S" action="word">
```

...

```
</generator>
```

The format attribute lists 3 code templates : A, AA, AAA

A template is a sequence of letters or symbols. A letter stands either for itself or represents a set of symbols. A symbol that is not a letter always represents only itself.

When a letter represents a set, that set is described by an attribute of that name.

In our example, the symbol set A is detailed by the attribute “A” as being the set of symbols: N,H,D,K,G,U,C,B,F,S.

By this prescription, valid codes examples are “N” , “DSF” but “X” and “CBHD” are not valid.

The symbol set can either be enumerated as in the example above but also ranges can be used. For example: N=“3-5,7,9” is equivalent to N=“3,4,5,7,9”. Z=“A-Z” means Z represents all uppercase letters.

If you want to include the **comma** symbol in a format or code set, you have to precede it with a backslash.

Next, we can add pronunciations for the different symbols:

```
<generator type="CODE" format="A,AA,AAA" A="N,H,D,K,G,U,C,B,F,S" action="word">
```

```
  <subject id="N">
```

```

    <pronounce>necrosis</pronounce>

    <pronounce>N</pronounce>

</subject>

<subject id="H">

    <pronounce>inhibition</pronounce>

    <pronounce>H</pronounce>

</subject>

...

</generator>

```

When no specific pronunciation is provided, the regular pronunciation for letters and numbers is used.

As for other generators, a prefix and/or suffix can be added.

#### 4.2.4 Customizing Autoconfiguration

When an Excel sheet was never configured before (no parameter file found in the directory of the Excel file), smatrix will try to autoconfigure the sheet. To make this autoconfiguration more successful, the *headers.xml* file can be enhanced with additional information.

- Add identifier names and header aliases

To enable the identification of identifier columns in a sheet, commonly used headers for those columns can be entered in the headers.xml file. They are configured as `<subject>` xml-elements under the `<topic>` with id "{SAMPLE}"

```
<topic id="{SAMPLE}" name="Sample">
```

... here all object name <subject> entries

```
</topic>
```

Each <subject> entry has a double purpose:

- To allow the user to select this entry as object name
- To add aliases to enhance the identification of identifier columns

Example:

```
<subject id="S4" name="treatment"/>
```

This entry allows the user to select 'treatment' as the name of a surveyed object.

Hence, the user can navigate by saying 'treatment 55'.

To enhance this entry for better autoconfiguration, we could write:

```
<subject id="S4" name="Treatment,TRT">
```

```
    <pronounce>treatment</pronounce>
```

```
</subject>
```

By specifying several aliases (Treatment, TRT) separated by comma, smatrix can easily detect columns with one of those headers as representing a treatment. Therefore, autoconfiguration will map that column to the object name treatment.

**Note:** Autoconfiguration will recognize a column header if the cell content, irrespective of case, **ends with** one of the aliases. A cell content of '009TRT' will match the 'TRT' alias.

Faced with multi-line headers in Excel, two meta-characters can be used to make the matching easier.

The 2-character sequence “\n” in an alias matches a newline in an Excel header.  
A **space character** in an alias matches one or more white space characters (including newline, tab, space ...) in an Excel header

So: “A\nB” will match only exactly “A*newline*B” in the header.

Better is to use: “A B” will match “A*newline*B” , “A B”, “A*tab*B”, “A B”, and many more.

- Add criterion names and header aliases

In exactly the same way, aliases can be added to the <subject> elements that represent criterion names. When smatrix recognizes such an alias in a sheet header, it can map that column to a particular criterion name

- Add subsample header aliases

Subsample numbers are easily detected when they are part of an object identifier (identifier + space + subsample-number). However, when the subsample numbers are in a column of their own, they cannot be distinguished from ordinary identifiers. To help autoconfiguration, typically used headers can be defined under a topic with id “{SUBSAMPLE}”.

```
<topic id="{SUBSAMPLE}" name="Subsample">
```

```
  <subject name="Subsample"/>
```

```
  <subject name="Sub"/>
```

```
</topic>
```

No pronunciation information is needed here because the entries are only used for recognizing sheet headers.

- Add #subsamples header aliases

When working with subsamples, it is sometimes necessary to have a different number of subsamples for some criteria.

This information is stored in the sheet as a separate header line. Example:

	A	B	C	D	E
1	Subsamples	5	10	5	1
2					
3					
4		Characteristic 1	Characteristic 2	Characteristic 3	Characteristic 4
5	Parcel	0-100%	1,3,5,7,9	50-120cm	1,3,5,7,9
6	100 1				
7	100 2				
8	100 3				
9	100 4				

Header line for #subsamples

In order for smatrix to recognize this header line, a specific topic with id "{#SUBSAMPLES}" can be created in the headers.xml file.

```
<topic id="{#SUBSAMPLES}" name="#Subsamples">
  <subject name="# Subsamples"/>
  <subject name="Subsamples"/>
</topic>
```

No pronunciation information is needed here because the entries are only used for recognizing locating the header line with the number of subsamples for each criterion.

#### 4.2.5 Smatrix configuration XML syntax

Smatrix is configured with two files: a headers file and a values file. Templates for these files are pre-installed in *C:\ProgramData\smatrix\templates*

Files should **never** be modified here because a re-installation of smatrix will overwrite the modified files. Instead, a template should be copied to `C:\ProgramData\smatrix\data` and be modified there.

If you operate in a mono-lingual environment, the files can be called headers.xml and values.xml. If you need to support multiple languages with the same installation, a language-suffix should be used (\_de for German, \_en for English, \_es for Spanish, \_fr for French, \_nl for Dutch, \_pt for Portuguese).

### The headers file

This file is used to configure the autoconfigure feature and to describe the various criteria to which columns can be mapped.

Xml-element	attribute	Content/Values	Comment
<headers>		<topic>	
<topic>		<subject>	A topic represents a subject category.
	id	<i>text</i>	Id, unique in file. The following topics are used for the autoconfiguration feature: {SAMPLE}, {SUBSAMPLE}, {#SUBSAMPLES}, {SUBSAMPLEGROUP} The other topics define criteria categories.
	name	<i>text</i>	Name to be displayed in drop-down list except for reserved topics "{.....}"
<subject>		<pronounce>	One or more pronunciations. Necessary if subject name is not pronounceable. If there are no <pronounce> elements, name will be used for pronunciation.
	id	<i>text</i>	Id, unique within the enclosing topic

	name	<i>text</i>	Name to be displayed in drop-down list except if in reserved topics "{...}" If a name constitutes a list of names separated by commas, all names are treated as aliases. This is used during autoconfiguration.
	range	<i>text</i>	Optional attribute. Id of a topic in the values file representing a value range. This value range will be the default range for this criterion.

### The values file

This file is used to define value ranges where each <topic> represents a range.

Xml-element	attribute	Content/Values	Comment
<values>		<topic>	Each topic represents a value range
<topic>		<subject>, <generator>	A <subject> represent a single text value. A <generator> represents a collection of values generated by the system.
	id	<i>text</i>	Id, unique in file
	name	<i>text</i>	Name to be displayed in drop-down list.
<subject>		<pronounce>	One or more pronunciations. Necessary if subject name is not pronounceable.
	id	<i>text</i>	Id, unique in topic. Represents the value of this subject.

	name	<i>text</i>	Name to be displayed in drop-down list. Represents the value of this subject when id is undefined.
<generator>		<prefix>, <suffix>, <subject>	prefix and suffix are optional, only 1 prefix and/or 1 suffix per generator can be defined. Subjects inside a CODE generator are used to define pronunciations for codes.
	type	"DATE" or "NUMBER" or "CODE"	
	from	<i>number with optional decimals</i>	<b>For type NUMBER</b> , begin (inclusive) of the range
	to	<i>number with optional decimals</i>	<b>For type NUMBER</b> , end (inclusive) of the range
	by	<i>number with optional decimals</i>	<b>For type NUMBER</b> , increment
	decimals	<i>Integer number</i>	<b>For type NUMBER</b> , number of digits after the comma
	format	"Date" "DayOfYear" "d/MMM/yy" "d" "dd.MM.yyyy"	<b>For type DATE</b> , one of these values: A date value The day of year numerical value Formatted: 7/Jun/18 Formatted: 07/06/2018 Formatted: 07.06.2018
	format	Example: "A,AA,AAA"	<b>For type CODE</b> , a list of code patterns separated by ','

			Each letter symbolizes either a set of code symbols or itself.
	A	Example: "1-3,7,9"	<b>For type CODE</b> , a list of letters, numbers or symbols that are part of the code set A. Either comma separated symbols or ranges x-y can be used.
	action	"word" or "concatenate"	<b>For type CODE</b> , is the code symbol included in the result as a separate word or concatenated to the other symbols?
<prefix>		<pronounce>	One or more pronunciations. Necessary if prefix name is not pronounceable.
	id	<i>text</i>	Prefix value
	name	<i>text</i>	Prefix value if no id present
	optional	"true" or "false"	Do you have to pronounce the prefix?
	include	"no","yes" or "ifsaid"	Is the prefix value included in the result? No = never, Yes = always, IfSaid = only if the suffix was pronounced?
	action	"word" or "concatenate"	Is the prefix value included in the result as a separate word or concatenated to the generator value?
<suffix>		<pronounce>	One or more pronunciations. Necessary if suffix name is not pronounceable.
	id	<i>text</i>	Suffix value
	name	<i>text</i>	Suffix value if no id present
	optional	"true" or "false"	Do you have to pronounce the suffix?
	include	"no","yes" or "ifsaid"	Is the suffix value included in the result? No = never, Yes = always, IfSaid = only if the suffix was pronounced?

	action	"word" or "concatenate"	Is the suffix value included in the result as a separate word or concatenated to the generator value?
<pronounce>		<i>text</i>	How to pronounce the containing subject, prefix or suffix