



# Selected Variable Data Entry

## *Functional Specification*

Version	Date	Autor	Changes
0.1	2013-02-07	Peter Pavek	Initial version
0.2	2013-02-11	Peter Pavek	Feedback from Assar. Added “unit” to variables.
0.3	2013-02-14	Peter Pavek	Changed “is visible” to “is available”. Changed unit “m” in pictures to ve “dm”.
0.4	2013-02-26	Peter Pavek	Adjusted after feedback, switching top panels, changing functionality and cleaning up text w r t list entries and variables.

## Introduction

### Background

Teraim develops software for mobile data collection. The key aspects of this software are its configurability and embedded data value validation.

### Scope

This document describes the functionality of a generic data entry page where a subset of variables is selected and assigned zero or more values. This page is always used when the variables that are to be filled out are not fixed ahead of time.

The foundations described in [WFE] and [Rules] apply here and provide a context.

## Definitions

Page	Used in this document to mean the contents of the screen – both the visible parts and the parts that may be hidden, but can be scrolled to.
Variable	Data structure for storing a number of data values. A variable has an associated entry field. The graphical layout of the field depends on the type of the variable. Each variable has an “is available” flag. By default, it is TRUE. The type of the variable defines the associated “keyboard”.

## Selected Data Entry

### Requirements

At times, there is a need to select a number of variables from a list and assign them values. In a typical situation, the number of variables in the list is much bigger than the number of eventually selected variables.



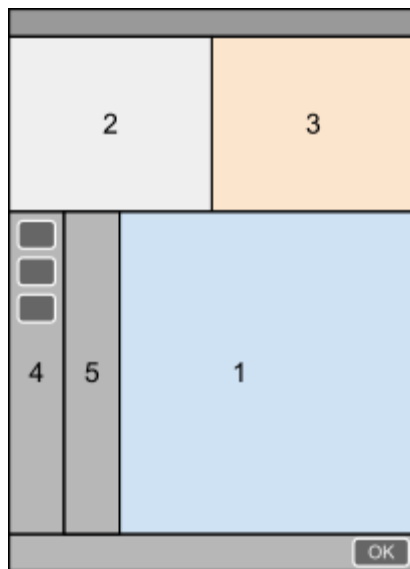
More specifically, the requirements are;

- Function for displaying the list of available variables.
- Functions for sorting the displayed variable list need to be available. Basic functions include sorting in an alphabetic order per variable name or per other variable attribute. Custom functions may be added at a later time.
- When a variable in the list is selected, per user gesture, it is added to the “selected variables.”
- A selected variable can be deselected, per user gesture, removing it from the “selected variables” and making it again be selectable in the variable list.
- The selected variables need to be presented in a clear way, separately from the variable list.
- The entered values need to be verified.
- Function for processing all selected variables and displaying the results needs to be present. Basic functions include summing number of variables and summing their values. Custom functions may be added later.

## Page Display Layout

The following figure shows the required page layout structure. The page has a number of sections – panels – numbered 1 to 5. The functionality of these is as follows;





- Panel 1 is used to display the list of available variables.
- Panel 2 is used to display the results of the aggregation operations on the selected variables.
- Panel 3 holds details or other depictions.
- Panel 4 is holds the defined variable list sorting functions, each depicted as a button.
- Panel 5 displays an index into the available variables in panel 1. This panel may or may not exist, depending on the sorting function that is selected in panel 4.



## Page Component Functionality

### Panel 1 – Field List

This panel displays a vertical list of fields. The texts are left-aligned. At the far right, each list element shall have an info button (“i”).

Gran	<input type="checkbox"/>	
Tall	<input checked="" type="checkbox"/>	
Björk	<input type="checkbox"/>	
Asp	<input type="checkbox"/>	

In the case that there are more list entries than what fits vertically in the panel, a scroll bar appears on the right side of the panel.

Each list entry represents one or more variables. Depending on the number of associated variables with each list entry and their types, the display of it will vary. The figure below shows list elements with one numeric variable. If there would be

Gran, täckning	<input type="text"/>	m2	<a href="#">i</a>
Tall, täckning	<input type="text"/>	m2	<a href="#">i</a>
Björk, täckning	<input type="text"/>	m2	<a href="#">i</a>
Asp, täckning	<input type="text"/>	m2	<a href="#">i</a>

Tapping on a list element selects it. A selected entry is marked as “selected” by color change. Depending on the type of variable, selection may imply that it should be assigned one or several values. An appropriate editor pops up allowing the user to enter the value(s).

Gran	<input type="text" value="17"/>	<input type="text" value="87"/>	<a href="#">i</a>
Tall	<input type="text"/>	<input type="text"/>	<a href="#">i</a>
Björk	<input type="text"/>	<input type="text"/>	<a href="#">i</a>
Asp	<input type="text"/>	<input type="text"/>	<a href="#">i</a>

As an example, assuming that the list entry “Tall” would have two variables – “Täckning” and “Höjd” – to fill out, selecting Tall should bring up the following editor.

Tall

Täckning

%

Höjd (85)

dm

OK

Tapping outside the window cancels the operation, hiding the editor.

Tapping the info button of a list entry displays a new panel with variable details, e.g., its properties. A link (or links) may be provided leading to local (and/or central) variable descriptions. Tapping on this link opens a web browser to the specified URI.

## Panel 2 – Variable Aggregation Function

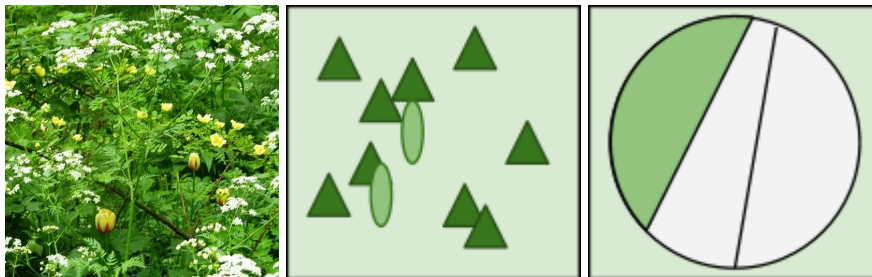
This panel holds one or several displays of values. The displays may be graphical or textual and may be combined in style. The basic concept is that this panel is used to show values computed from the selected variables and their values. The panel may, however, show as diverse things as a photograph, an animation, bars, or text.

Antal arter	3	Antal arter	3
Total täckning	27	Total täckning	27
Max höjd	125	<div><div></div></div>	
Snitt höjd	78	Höjd	
		61   75   127	

Note that the function generating the contents must have access both to the selected variables as well as to global variables, as some functionality may require global data.

### Panel 3 – Detail or Aggregation Display

This panel displays a further detail, additional aggregation display, information. The contents may be sensitive to what field entry is selected in panel 1 or what aggregation function is selected in panel 2..



### Panel 4 – Variable Sorting and Filtering Functions

This panel holds buttons for sorting and filtering functions. These functions control the order of the entries in the variable list in panel 1. One sorting function is always active – initially, the first one. The active sorting function is indicated by color change of its button.

Each sorting function determines its own secondary sorting function, i.e., how entries are sorted when they have the same “position” in the list. Note that the sorting function must have access to global variables.

It must be possible to add new sorting functions and corresponding buttons beyond those defined in the initial release.

Examples of sorting functions are alphabetic by displayed text, alphabetic by other variable properties (e.g., by class or type of species), or by frequency in a certain zone. Examples of filtering functions are conifers, lichen, etc.

### Panel 5 – Index Function

This panel holds relevant index entries into the list of entries in panel 1. The contents of this panel are controlled by the sorting and filtering functions. The width of this panel (and consequently that of panel 1) is also controlled by the filtering functions. Some functions may decide to omit this, setting panel 5 width to 0. Others may set it to be quite wide, to accommodate longer words written horizontally.

The A-Ö sorting function may have a narrow index panel containing the alphabet. The species sorting function may have a wide index panel listing categories.

Selecting an index entry scrolls the panel 1 list to the first entry belonging to that index. Thus, selecting “B” would jump to “Björk” (assuming that that entry was the first one starting with a B.)

# Data Entry Page Configuration

A sequence of actions is used to define a page with a random variable data entry function. The actions define the list of variables, the sorting functions, and aggregation functions. The sequence is implemented as a workflow. The generic concept is described in [WFE] and [WDE]. The specifics for this context consist of special workflow blocks and definition of execution run-time data. The following workflow blocks are used to implement the required actions.

## Define Variable Selection Page Block

This block defines a new page and directs the workflow engine to display it. The block has the following specific properties.

- *Page Name* (symbol) is the name of the page to display. The name must be unique in the system. When referring to this page, this name is used.
- *Page Label* (text) is the text that appears at the top of the page. If left empty, the page will have no header.
- *Is Available* (logical) is a flag that controls the visibility of the page. As long as the flag is true, the workflow engine will display the page.

## Create List Entry Block

This block defines an entry in the variable list in panel 1. Each entry represents one variable. If the variable does not already exist, this block creates the variable as well. Additional variable can be added to the list entry using the next block.

The block has the following properties:

- *Name* (symbol) is the name of this list entry. The name must be unique in the system. When referring to this list entry, this name is used.
- *Field Label* (text) is the label of this field, when displayed in the related page.
- *Variable Name* (symbol) is the name of the associated variable. The name must be unique to the system. When referring to this variable, this name is used.
- *Variable Type* (symbol) is the name of one of the predefined variable types. Examples are integer, number, text, truth-value, percentage, date, and currency. The value of this property directs the selection of the editor. A numeric type will bring up a number pad. A text type will bring up an alphanumeric keyboard. A truth-value type will not bring up an editor at all. Rather, tapping on the list entry selects it and sets the corresponding value to true.
- *Unit* (text) is the unit used for measuring the variable value. Examples are “m” and “dm”. Empty strings are allowed.
- *Page Name* (symbol) name of the page where the variable entry field should appear.
- *Is Available* (logical) is a flag that describes the availability for selection (visibility) of the list entry (not the variable!). Initially, this is true.
- *Show Previous Value* (logical) is a flag controlling whether to show the value from the previous sampling or not.

One or more blocks possible for each list entry.

### Add Variable to List Entry Block

This block adds a variable to a list entry. In this way, a list entry can represent more than one variable and the editor for the list entry allows editing one or more values, depending on the number of associated variables.

The block has the following properties:

- *List Entry Name* (symbol) is the name of the list entry where this variable is to be added. The name must exist.
- *Field Label* (text) is the label of this field, when displayed on the related page. Note that in the list entry, adding multiple variables results in the field label being an intersection of the labels, i.e., only the common parts are displayed. So, for example, labels “Björk, höjd” and “Björk, täckning” result in the label “Björk” (where the trailing separators comma space “,” are ignored.) In the editor, the individual field labels are used.
- *Variable Name* (symbol) is the name of this variable. The name must be unique to the system. When referring to this variable, this name is used.
- *Variable Type* (symbol) is the name of one of the predefined variable types. Note that if one of the variables associated to a list entry is a truth-value and another is not, the editor shows only an entry field for the other variable, i.e., the truth-value variable is set implicitly by the selection.
- *Unit* (text) is the unit used for measuring the variable value. Examples are “m” and “dm”.
- *Show Previous Value* (logical) is a flag controlling whether to show the value from the previous sampling or not.

### Add Rule Block

This block is described in [WDE] [Link](#) This block adds a rule to the referenced list entry variable.

### Add Aggregation Function Block

This block adds an aggregation function to panel 3. There may be a one or more aggregation functions added in this way. The block has the following properties:

- *Field Label* (text) is the label of this field, when displayed on the related page.
- *Function Name* (symbol) is the name of this function. The name must be one of the available aggregation functions. An example of an aggregation function is *no-of-selections*, that counts the number of selected data entries.
- *Page Name* (symbol) name of the page where the variable entry field should appear.

### Create Filter Block

This block adds a sorting and/or filtering function and associated button to panel 4. These functions affect the order in which the list entries in panel 1 are sorted. The first added function is the default selected function when the user opens the page. The block has the following properties:

- *Button Label* (text) is the label of the associated button, when displayed on the related page.

- *Function Name* (symbol) is the name of this function. The name must be one of the available sorting/field functions. Examples are:
  - A-Ö: sorting entries in alphanumeric order using their display texts.
  - Släkten: sort order based on a non-displayed variable property.
  - Vanligaste i regionen: predefined order using information on current region (as entered by user.)
- *Page Name* (symbol) name of the page where the variable entry field should appear.

## Condition Block

This block is described in [WDE] [Link](#). This block routes execution by choosing one of two next blocks depending on the value of an expression.

## Create Action Button Block

This block is described in [WDE] [Link](#). This block adds an action button to the bottom panel (no number) of the page. Additions are right-aligned, left to right.

# Examples

## Example 1 – Species Selection

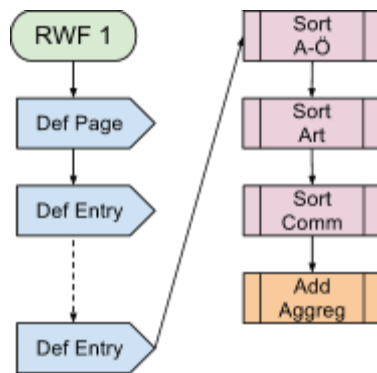
This example shows a workflow for creating a page where the existence of species is entered.

Antal arter 0	
<div>A - Ö</div> <div>Arter</div> <div>Vanlig</div>	<div>AI</div> <div>Asp</div> <div>Björk</div> <div>Fjällfibbla</div> <div>Fjällummer</div> <div>Gråfibbla</div> <div>Maskros</div> <div>Rödklöver</div> <div>Skogsklöver</div> <div>Örnbräken</div>
OK	

Thus, when a list entry is selected, it is added to the list of selected variables and the aggregation function – in this example only one – increments the number of selected species.

The workflow that created the page and defined its initial state is as follows. Note that the graphical appearance is for illustration purposes only.





The properties of the blocks are as follows:

#### **Block 1: Start Workflow**

Label: RWF 1

Name: RWF\_1

#### **Block 2: Define Variable Selection Page**

Label: Def Page

Name: Page\_1

Page Label: Existence of Species

is available: true

#### **Block 3: Create List Entry**

Label: Def Entry

Name: P1\_AI

Field Label: AI

Variable Name: AL

Variable Type: truth-value

Page Name: Page\_1

Unit:

is available: true

Show Previous Value: true

#### **Block 4: Create List Entry**

Label: Def Entry

Name: P1\_Asp

Field Label: Asp

Variable Name: ASP

Variable Type: truth-value

Unit:

Page Name: Page\_1

is available: true

Show Previous Value: true

#### **Block 5: Create List Entry**

Label: Def Entry

Name: P1\_Björk  
Field Label: Björk  
Variable Name: BJORK  
Variable Type: truth-value

:  
Page Name: Page\_1  
is available: true  
Show Previous Value: true  
:

#### **Block 7: Create Filter**

Label: Sort A-Ö  
Button Label: A - Ö  
Function Name: sort\_entries\_alphanurically\_by\_labels  
Page Name: Page\_1

#### **Block 8: Create Filter**

Label: Sort Art  
Button Label: Arter  
Function Name: sort\_entries\_alphanumerically\_by\_kind  
Page Name: Page\_1

#### **Block 9: Create Filter**

Label: Sort Comm  
Button Label: Vanlig  
Function Name: sort\_entries\_alphanumerically\_by\_frequency  
Page Name: Page\_1

#### **Block 9: Add Aggregation Function**

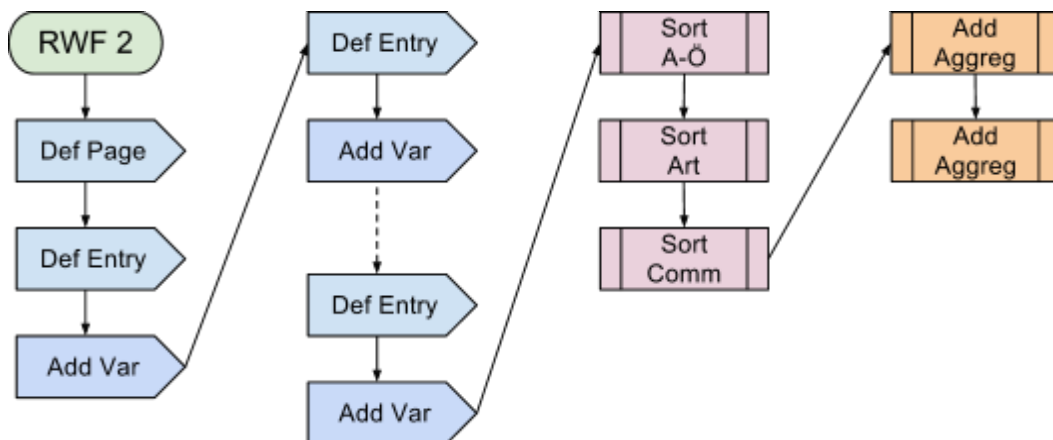
Label: Add Aggreg  
Field Label: Antal arter  
Function Name: number\_of\_selected\_entries  
Page Name: Page\_1

## **Example 2 – Species Data Entry**

This example shows a workflow for creating a page where two properties of selected variables receive values. The figure below illustrates three states of this page. Left is the initial state. Middle is when a list entry (Björk) is selected and an editor has appeared. Note that the list entries have two variables associated with them. Consequently, the editor for an entry shows two entry fields. Right is the state where three list entries have been selected and had values entered and the list has been re-sorted. Note also that the index panel, with the selected sort function, is not present.

Antal arter 0 Total täckning 0		Antal arter 0 Total täckning 0		Antal arter 3 Total täckning 44		Tall: täckning, höjd 12 110 Björk: täckning, höjd 25 73 Rönn: täckning, höjd 7 37	
A - Ö	Al Asp Björk Bok Ek Gran Rönn Tall	A - Ö	Al Asp Björk Bok Ek Gran Rönn	A - Ö	Tall Gran Björk Asp Al Rönn Ek Bok		
OK		Björk, täckning <input type="text"/> % Björk, höjd <input type="text"/> dm OK		OK			

The workflow that created the page and defined its initial state is as follows. Note that the graphical appearance is for illustration purposes only.



The properties of the blocks are as follows:

#### Block 1: Start Workflow

Label: RWF 2

Workflow Name: RWF\_2

#### Block 2: Define Variable Selection Page

Label: Def Page

Page Name: Page\_2

Page Label: Species

is available: true

#### Block 3: Create List Entry

Label: Def Entry

Name: P2\_Al

Field Label: Al, täckning

Variable Name: AL.TACKN

Variable Type: percentage  
Unit: %  
Page Name: Page\_2  
is available: true  
Show Previous Value: true

#### **Block 4: Add Variable to List Entry**

Label: Def Entry  
List Entry Name: P2\_Al  
Field Label: Al, höjd  
Variable Name: AL.HOJD  
Variable Type: integer  
Unit: dm  
is available: true  
Show Previous Value: true

#### **Block 5: Create List Entry**

Label: Def Entry  
Name: P2\_Asp  
Field Label: Asp, täckning  
Variable Name: ASP.TACKN  
Variable Type: percentage  
Unit: %  
Page Name: Page\_2  
is available: true  
Show Previous Value: true

#### **Block 6: Add Variable to List Entry**

Label: Def Entry  
List Entry Name: P2\_Asp  
Field Label: Asp, höjd  
Variable Name: ASP.HOJD  
Variable Type: integer  
Unit: dm  
is available: true  
Show Previous Value: true

:

#### **Block 7: Create Filter**

Label: Sort A-Ö  
Button Label: A - Ö  
Function Name: sort\_entries\_alphanurically\_by\_labels  
Page Name: Page\_2

#### **Block 8: Create Filter**

Label: Sort Art  
Button Label: Arter  
Function Name: sort\_entries\_alphanumerically\_by\_kind  
Page Name: Page\_2

#### **Block 9: Create Filter**

Label: Sort Comm  
Button Label: Vanligast  
Function Name: sort\_entries\_alphanumerically\_by\_frequency  
Page Name: Page\_2

#### **Block 10: Add Aggregation Function**

Label: Add Aggreg  
Field Label: Antal arter  
Function Name: number\_of\_selected\_entries  
Page Name: Page\_2

#### **Block 11: Add Aggregation Function**

Label: Add Aggreg  
Field Label: Total täckning  
Function Name: cumulative\_coverage  
Page Name: Page\_2

### **Example 3 – Data Entry Page with Rules**

This example shows a data entry page with variables containing validation rules. The variables in this example do not have old (previously collected) values. Thus, there are not shown in the entry field.

## **References**

- [WFE] Generic Workflow Engine, 2012, Pavek, P.
- [Rules] Rules for Validation, 2012, Pavek, P.
- [WDE] Workflow Engine for Data Entry, 2013, Pavek, P.