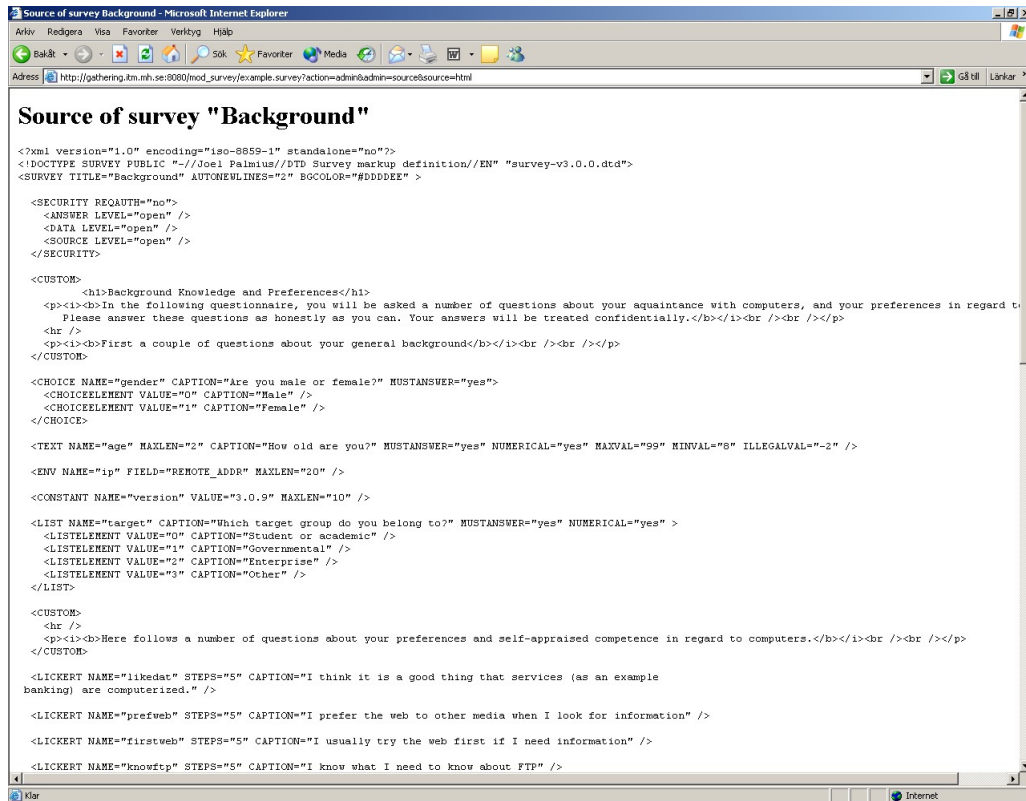


# Mod\_Survey Syntax Reference

## 3.2.x branch



**MSc. Joel Palmius**  
**Mid Sweden University**  
**Department of Information Technology and Media**

**Mod\_Survey Syntax Reference: 3.2.x branch**  
by MSc. Joel Palmius

DocBook Version Edition

Published 2004

Copyright © 2004 Joel Palmius



This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/><sup>1</sup> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

# Table of Contents

<b>1. Introduction.....</b>	<b>1</b>
1.1. Tags .....	1
1.2. Markup.....	1
1.3. Includes .....	1
1.4. Snippets.....	1
<b>2. Configuration tags.....</b>	<b>3</b>
2.1. SURVEY .....	3
2.2. MAILCOPY.....	8
2.3. SECURITY .....	10
<b>3. Visible variable-carrying tags .....</b>	<b>19</b>
3.1. BOOLEAN .....	19
3.2. CHOICE.....	20
3.3. CUSTOM.....	23
3.4. LICKERT .....	25
3.5. LIST .....	27
3.6. MATRIX .....	30
3.7. MEMO .....	36
3.8. TEXT .....	37
<b>4. Invisible variable-carrying tags .....</b>	<b>41</b>
4.1. CONSTANT.....	41
4.2. DATETIME.....	41
4.3. ENV .....	43
4.4. IMPORT.....	44
4.5. TIMER .....	48
<b>5. Tags related to multipaging.....</b>	<b>49</b>
5.1. CASEROUTE .....	49
5.2. IFROUTE.....	50
5.3. RANDOMROUTE.....	52
5.4. ROUTE .....	53
5.5. SEQUENCE.....	54
<b>6. Layout-related and other tags.....</b>	<b>57</b>
6.1. NEWLINE .....	57
6.2. COMMENT .....	57
<b>7. Extra HTML markup.....</b>	<b>59</b>
7.1. Color .....	59
7.2. Font styles .....	59
7.3. Graphics .....	59
7.4. Special.....	60
7.5. Custom HTML markup.....	60

<b>8. Dynamic content.....</b>	<b>61</b>
8.1. Including external files.....	61
8.2. Referencing previous variables .....	61
8.2.1. Session variables .....	62
8.3. Perl snippets .....	62

# List of Tables

2-1. SURVEY parameter reference.....	3
2-2. MAILCOPY parameter reference .....	8
2-3. SECURITY parameter reference.....	10
2-4. DBIAUTH parameter reference (Subtag of SECURITY).....	10
2-5. FILEAUTH parameter reference (Subtag of SECURITY).....	11
2-6. TOKENAUTH parameter reference (Subtag of SECURITY) .....	11
2-7. Subtags of SECURITY ("security domains") .....	13
2-8. LEVEL settings .....	13
2-9. ANSWER parameter reference .....	14
2-10. DATA parameter reference .....	14
2-11. DEBUG parameter reference.....	15
2-12. FLUSH parameter reference.....	15
2-13. SOURCE parameter reference.....	15
3-1. BOOLEAN parameter reference .....	19
3-2. CHOICE parameter reference .....	20
3-3. CHOICEELEMENT parameter reference.....	21
3-4. DBCHOICEELEMENT parameter reference .....	22
3-5. CUSTOM parameter reference.....	23
3-6. LICKERT parameter reference.....	25
3-7. LIST parameter reference .....	27
3-8. LISTELEMENT parameter reference .....	29
3-9. DBLISTELEMENT parameter reference.....	29
3-10. MATRIX parameter reference .....	30
3-11. MATRIXCOLUMN parameter reference.....	33
3-12. MATRIXROW parameter reference .....	34
3-13. MEMO parameter reference .....	36
3-14. TEXT parameter reference .....	38
4-1. CONSTANT parameter reference .....	41
4-2. DATETIME parameter reference .....	42
4-3. FORMAT codes.....	42
4-4. ENV parameter reference .....	43
4-5. Examples of environment variables.....	44
4-6. IMPORT parameter reference .....	44
4-7. VARIABLE parameter reference.....	45
4-8. FILTER parameter reference .....	46
4-9. ARRAY parameter reference.....	47
4-10. TIMER parameter reference .....	48
5-1. CASEROUTE parameter reference .....	49
5-2. CASE parameter reference .....	49
5-3. IFROUTE parameter reference.....	50
5-4. IF parameter reference.....	50

5-5. Boolean conditions .....	51
5-6. Boolean condition parameter reference.....	51
5-7. RANDOMROUTE parameter reference .....	52
5-8. ALTERNATIVE parameter reference .....	53
5-9. ROUTE parameter reference .....	53
5-10. SEQUENCE parameter reference .....	54
5-11. FILE parameter reference .....	55
6-1. NEWLINE parameter reference .....	57
6-2. COMMENT parameter reference .....	57

# Chapter 1. Introduction

This manual summarizes the syntax of the survey language in the 3.2.x branch. For an overview of what has changed since the 3.0.x branch, please refer to "Mod\_Survey for Beginners".

Note that the following material presupposes an understanding of the basics of Mod\_Survey. For an introduction on how survey files are written, and a step-by-step tutorial on the survey language, refer to "Mod\_Survey for Beginners".

## 1.1. Tags

The majority of the survey language consists of an XML-based set of "tags". These are written in the style common to HTML, XML and SGML. Most of the tags define variables (usually questions), but there are also tags related to page routing and configuration.

In the reference, a name of an optional parameter is written as `PARAMETER`, while a required parameter is written as *PARAMETER*.

In the following text an "open" tag will denote a tag which have contents apart from its parameters, and is written in the form `<TAG PARAMETERS...>(contents)</TAG>`. A "closed" tag does not have any contents apart from its parameters, and must be immediately terminated in the form `<TAG PARAMETERS... />`.

## 1.2. Markup

Apart from the tags, there is also a set of specific codes used to modify the HTML output. These are in the form `{code}...{/code}`. They are used, for example, to boldface a word in a caption. The extra markup can be used in all places in the code that end up as visible output. Examples are in the `CAPTION` parameter of a question or inside a `CUSTOM` block.

## 1.3. Includes

The `SURVEY` markup allows linking to external files during different phases of the survey's life span: they can be included before or after syntax parsing. This can be used, for example, to include one common set of questions into many surveys, or to keep large blocks of HTML code outside the definition of questions.

## 1.4. Snippets

Finally, there are perl "snippets". This is perl code which can be inserted in the survey and execute during various instances of the survey life span. Examples could be for producing a large MATRIX from an array of values. The perl snippets are fed through the "Safe" module and will not allow any interaction with the file system.



# Chapter 2. Configuration tags

## 2.1. SURVEY

The SURVEY tag is the top-level container of the survey. All other data is contained within the SURVEY tag. The survey tag is always open.

**Table 2-1. SURVEY parameter reference**

Name	Values	Default	Description
ASCIIFILE	A filename to an existing file		This parameter is used to override the default place for saving data when a respondent submits answers. Most user will never need this, as the functionality has been outdated by the central repository which is used per default.
AUTONEWLINES	An integer	0	If set to a value higher than 0, this many line breaks will be inserted between every visible question.
BACKGROUND	An URL to an image		If set, this will be specified in the resulting HTML page's body tag in the "background" parameter. This will cause the referenced image to be placed as the page background.
BGCOLOR	A HTML color code	#FFFFFF	The background color of the survey page.

Name	Values	Default	Description
CAPTWIDTH	An integer	300	With the default themes, this specifies the width allocated to captions to the left in the page (ie, usually the space for the question texts).
CHECKKEY	yes/no	yes	Whether a check should be made for if the respondent submits a valid "key" for a second page in a multipage sequence. Set this to no if you want respondents to go back to check/change their answers. OBSERVE that this should never be set to "no" on the last page in a survey chain.
CLEARTEXT	Any string	Clear	The text on the "clear" button in the bottom of the survey page.
CONTINUE	An URL		When not in a multipage sequence, this will send the user on to another page when answers have been submitted. See also REDIRECT.
DBITABLE	A database table name		If set, this will override the default saving of data in the central repository, and instead save data in the specified (pre-existing) database table. See also DBIDSN, DBIUSER and DBIPASSWD.

Name	Values	Default	Description
DBIDSN	A database DSN		The connection string to use when saving data in a database table. For PostgreSQL users this could for example be "DBI:Pg:dbname=mydb".
DBIUSER	A database user		The user string to use when connecting to a database.
DBIPASSWD	A database password		The password string to use when connecting to a database.
DELIMITER	Any character	;	(OBSOLETE) delimiter to use when saving data in ASCIIFILES. Most users will want to leave this unchanged.
DESCRIPTION	Any string		If set, this will be inserted in the resulting HTML page's "meta description", something which is often used for indexing by search engines.
DOSBR	yes/no	no	In data exports, use 0x0d0x0a rather than 0x0a as line break. Most users will want to leave this unchanged unless they have really old statistics software run in a DOS environment.

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
KEYWORDS	Any string		If set, this will be inserted in the resulting HTML page's "meta keywords", something which is often used for indexing by search engines.
LANGUAGE	A two-letter language code		Override system-wide setting of language. If set to for example "de", all error messages on this survey page will be displayed in german. Available languages can be found in the Lang directory of the installation.
LINKCOLOR	A HTML color code	#0044FF	Color to use for links in the survey pages.
NOTDISPLAYEDVAL	An integer	999	This is the value question that were never displayed in a multipage sequence will get when data is finally submitted. Advanced users may want to change this to their statistics software's "missing" value.
PERSIST	yes/no	no	Whether to allow respondent to temporarily save answer in order to return later for finishing the survey.
PERSISTTEXT	Any string	"Complete later"	Text to show on persistence button.

Name	Values	Default	Description
PROGRES	"n1/n2"		Determine the progress in a survey chain where n1 is number of questions left to answer and n2 is the width (in pixels) of a progress bar. This parameter does not do anything in itself, it is used in conjunction with the progress bar, see the html markup section. (The number of questions answered so far is automatically calculated)
RANDOM	yes/no	no	Whether to randomize display order of tags on page.
REDIRECT	yes/no	no	If in multipage or when CONTINUE is set, this will automatically send the user to the next page (redirecting with 0 seconds delay). If switched off, the user will instead be presented with a link "click here to continue".
RETRIVETEXT	Any string	"Finish your pre-answered survey"	Text to show on continue persisted survey button.
SHOWCLEAR	yes/no	yes	Whether to show the "clear" button at all in the bottom of the survey page.

Name	Values	Default	Description
STYLESHEET	An URL to a stylesheet		If set, this will be specified as the HTML page's "link rel stylesheet". Users keen on design may want to use this to specify extra CSS styles for their survey page.
SUBMITTEXT	Any string	Submit	Text to use on the "submit" button.
SUBSET	An integer	0	Randomly pick this many tags for display. Disabled if set to 0.
TEXTCOLOR	A HTML color code.	#000000	The color to use on the text in the survey.
THEME	A valid theme name (see online examples)	cloud	Name of system style to use.
<i>TITLE</i>	Any string		The string to set in the browser's title bar.
VLINKCOLOR	A HTML color code	#0044FF	The color to use for visited links in the survey.

## 2.2. MAILCOPY

The MAILCOPY tag specifies that an email containing the data submitted by the latest respondent should be sent to a specified email address. While this tag is possible to place anywhere in a multipage chain, it makes most sense to place it in the last page.

Note that configuration of default email address and SMTP host takes place in survey.conf and thus isn't possible to change by the user.

**Table 2-2. MAILCOPY parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

Name	Values	Default	Description
CAPTIONS	yes/no	no	Display captions corresponding to values rather than the values themselves.
CC	An email address		The email address to use for the "CC:" field
FROM	An email address	[specified in survey.conf]	The email address to use for the "From:" field
MSGAFTERDATA	yes/no	no	Place the message part of the email after the listed data rather than before.
REPLYTO	An email address		The email address to use for the "Reply-To:" field
SIGNATURE	Any string		Text (usually a name) to add in the bottom of the email
SUBJECT	Any string	"[Mod_Survey] autogenerated message"	The text to use for the "Subject:" field
TO	An email address	[specified in survey.conf]	The email address to use for the "To:" field

Example:

```

<SURVEY TITLE="mailcopy_demo">

<CHOICE NAME="choice" CAPTION="Do you like...">
<CHOICEELEMENT CAPTION="yes" VALUE="1" />
<CHOICEELEMENT CAPTION="no" VALUE="2" />
</CHOICE>

<MAILCOPY FROM="testperson@web.de" SIGNATURE="testperson"
SUBJECT="question_results" TO="receptor@test.com" CAPTIONS="yes">
</MAILCOPY>

</SURVEY>

```

## 2.3. SECURITY

The SECURITY tag and its subtags specifies how access should be permitted (or not permitted) to security domains of the survey.

**Table 2-3. SECURITY parameter reference**

Name	Values	Default	Description
REQAUTH	yes/no	no	Authentication mechanism to use if yes

The SECURITY tag is open and may contain the following subtags:

**Table 2-4. DBIAUTH parameter reference (Subtag of SECURITY)**

Name	Values	Default	Description
DBIDSN	A DSN connection string		DSN to use when connecting for DBI auth mechanism
ENCRYPTED	yes/no	yes	If using dbi auth, are passwords stored in encrypted form?
DBIPASS	A valid password		Password to use when connecting for DBI auth mechanism
DBIUSER	A valid username		Username to use when connecting for DBI auth mechanism
RESP-TABLE	A database table name		Database table where a list of respondents are found
RESP-NAMES-COL	A table column name		Column in the respondent table, containing the names of the respondents



<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
RESP-PWD-COL	A table column name		Column in the respondent table, containing the passwords of the respondents
USER-TABLE	A database table name		Database table where a list of users are found
USER-NAMES-COL	A table column name		Column in the user table, containing the names of the users
USER-PWD-COL	A table column name		Column in the user table, containing the passwords of the users

**Table 2-5. FILEAUTH parameter reference (Subtag of SECURITY)**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
RESPFILE	A file name		If present, this is used as a list of respondents. The file is expected to be in the "htpasswd" format (see man/info on apache's "htpasswd")
USERFILE	A file name		If present, this is used as a list of users not expected to answer the survey. This is usually a set of people who are going to administrate the survey and/or download data. The file is expected to be in the "htpasswd" format (see man/info on apache's "htpasswd")

**Table 2-6. TOKENAUTH parameter reference (Subtag of SECURITY)**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
APPEND	yes/no		If set to yes, the token will be appended automatically when routing to the last survey page.
ARG	Any string		The notation of the token
DBIDSN	A DSN connection string		DSN to use when connecting for DBI auth mechanism
DBIPASS	A valid password		Password to use when connecting for DBI auth mechanism
DBIUSER	A valid username		Username to use when connecting for DBI auth mechanism
RESPCOL	A table column name		Column in the table where the responents are found
RESPFILE	A file name		If present, this is used as a list of respondents. The file is expected to be in the "htpasswd" format (see man/info on apache's "htpasswd")
RESP-TABLE	A database table name		Database table where a list of respondents are found
USERCOL	A table column name		Column in the table where the users are found

Name	Values	Default	Description
USERFILE	A file name		If present, this is used as a list of users not expected to answer the survey. This is usually a set of people who are going to administrate the survey and/or download data. The file is expected to be in the "htpasswd" format (see man/info on apache's "htpasswd")
USER-TABLE	A database table name		Database table where a list of users are found

**Table 2-7. Subtags of SECURITY ("security domains")**

Name	Function
ANSWER	Security settings for respondents (ie who are allowed to answer the survey)
DATA	Security settings for data module (ie who are allowed to download data)
DEBUG	Security settings for debug info (who are allowed to see debug output) (OUTDATED/OBSOLETE)
FLUSH	Security setting for flush (who are allowed to remove all data)
SOURCE	Security settings for source (who are allowed to view the source of the survey)

All security tags have a LEVEL parameter used to restrict or open access to the security domain.

**Table 2-8. LEVEL settings**

Name	Function
open	Access is unrestricted, everyone may do this
closed	Access is denied, no-one may do this

<b>Name</b>	<b>Function</b>
user	Access is allowed for anyone present in USERFILE
respondent	Access is allowed for anyone present in RESPFILE
any	Access is allowed for anyone present either in RESPFILE or in USERFILE

**Table 2-9. ANSWER parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
HOSTS	A comma-separated list of IP numbers		Allow access only from these hosts
LEVEL	A level setting	open	See table above
PASSWORD	Any string		Allow access to anyone who gives this password
UNIQUE	yes/no	no	Respondent may only take survey once
USERS	A comma-separated list of users		Allow access to only these users, who must be listed in USERFILE

**Table 2-10. DATA parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
HOSTS	A comma-separated list of IP numbers		Allow access only from these hosts
LEVEL	A level setting	closed	See table above
PASSWORD	Any string		Allow access to anyone who gives this password
USERS	A comma-separated list of users		Allow access to only these users, who must be listed in USERFILE

**Table 2-11. DEBUG parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
HOSTS	A comma-separated list of IP numbers		Allow access only from these hosts
LEVEL	A level setting	closed	See table above
PASSWORD	Any string		Allow access to anyone who gives this password
USERS	A comma-separated list of users		Allow access to only these users, who must be listed in USERFILE

**Table 2-12. FLUSH parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
HOSTS	A comma-separated list of IP numbers		Allow access only from these hosts
LEVEL	A level setting	closed	See table above
PASSWORD	Any string		Allow access to anyone who gives this password
USERS	A comma-separated list of users		Allow access to only these users, who must be listed in USERFILE

**Table 2-13. SOURCE parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
HOSTS	A comma-separated list of IP numbers		Allow access only from these hosts
LEVEL	A level setting	closed	See table above

Name	Values	Default	Description
PASSWORD	Any string		Allow access to anyone who gives this password
USERS	A comma-separated list of users		Allow access to only these users, who must be listed in USERFILE

Example (Authentication via database):

```
<SURVEY TITLE="Security_demo">
<SECURITY REQAUTH="yes">
<DBIAUTH DBIDSN="dbi:mysql:database=test"
DBIUSER="somebody" RESP-NAMES-COL="name"
DBIPASS="secret" RESP-TABLE="testsurvey"
RESP-PWD-COL="password"
ENCRYPTED="no"/>
<ANSWER LEVEL="respondent"/>
<DATA PASSWORD="random" LEVEL="open"/>
<FLUSH LEVEL="closed"/>
<SOURCE LEVEL="closed"/>
</SECURITY>
</SURVEY>
```

Example (Authentication via file):

```
<SURVEY TITLE="Security_demo">
<SECURITY REQAUTH="yes">
<FILEAUTH RESPFILE="\home\sombody\respfile"/>
<ANSWER LEVEL="respondent"/>
<DATA PASSWORD="random" LEVEL="open"/>
<FLUSH LEVEL="closed"/>
<SOURCE LEVEL="closed"/>
</SECURITY>
</SURVEY>
```

Example (Authentication via token):

```

<SURVEY TITLE="tokentest">
<SECURITY REQAUTH="yes">
  <ANSWER LEVEL="respondent"/>
  <DATA PASSWORD="random" LEVEL="open"/>
  <SOURCE LEVEL="closed"/>
  <FLUSH LEVEL="closed"/>
  <DEBUG LEVEL="closed"/>
    <TOKENAUTH
      DBIDSN="dbi:mysql:database=test" DBIUSER="somebody"
      DBIPASS="secret" RESP-TABLE="testsurvey"
      RESPCOL="name"
      APPEND="no"
      ARG="token"/>

</SECURITY>
</SURVEY>

```

**Note:** When using authentication via token, the token must be appended on the url that routes to the survey. For example:

*http://...../testsurvey.survey?action=display&token="respondent"*





# Chapter 3. Visible variable-carrying tags

## 3.1. BOOLEAN

The BOOLEAN tag specifies an on/off checkbox kind of style question. The BOOLEAN tag is always closed.

**Table 3-1. BOOLEAN parameter reference**

Name	Values	Default	Description
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	BOOLEANcap	Which style (as specified in a stylesheet) to use for the caption of the object.
CHECKED	yes/no	no	Whether the checkbox should be checked or not.
NAME	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
STYLE	A style name	BOOLEANelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.

Example:

```

<SURVEY TITLE="boolean_test">
<BOOLEAN NAME="mail" CAPTION="Yes, I want to be notified via e-mail"
CHECKED="no" />
</SURVEY>

```

## 3.2. CHOICE

The CHOICE tag specifies an array of possible options. The user may be single choice or multiple choice depending on the state of the MULTIPLE parameter (see below). The CHOICE tag is always open and must contain at least one CHOICEELEMENT.

**Table 3-2. CHOICE parameter reference**

Name	Values	Default	Description
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	CHOICEcap	Which style (as specified in a stylesheet) to use for the caption of the object.
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
LABELCAPTSTYLE	A style name	CHOICElabelcap	Which style (as specified in a stylesheet) to use for the labels of the object.
MAXLEN	An integer	80	When in multiple mode, how much space should be allocated in data exports for the resulting string.

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
MULTI	yes/no	no	Whether the respondent should be allowed to choose more than one of the options.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
OTHERFIELD	Any string		Specifies the caption of an extra option with a free-text field. If blank, the free-text field will not be displayed.
RANDOM	yes/no	no	Whether the options should be listed in random order. If no, then they will be listed in the order they appear in the survey file.
STYLE	A style name	CHOICEelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.

The CHOICEELEMENT subtag of the CHOICE tag represent one respondent-selectable answer to the object's question.

**Table 3-3. CHOICEELEMENT parameter reference**

Name	Values	Default	Description
<i>CAPTION</i>	Any string		The option's description.
CHECKED	yes/no	no	Whether the option should be pre-selected.
<i>VALUE</i>	An integer		The value this option represents.

The DBCHOICEELEMENT subtag of the CHOICE tag represents an array of possible answers. The array(s) must have been either imported via the IMPORT tag, or created in a perl snippet.

**Table 3-4. DBCHOICEELEMENT parameter reference**

Name	Values	Default	Description
<i>CAPTIONARRAY</i>	An array name		Array to use for the captions.
VALUEARRAY	An array name		Array to use for the values.

Example:

```
<SURVEY TITLE="CHOICE">
```

```

  <CHOICE NAME="ch1" CAPTION="Plain choice">
    <CHOICEELEMENT CAPTION="Option 1" VALUE="1" />
    <CHOICEELEMENT CAPTION="Option 2" VALUE="2" />
    <CHOICEELEMENT CAPTION="Option 3" VALUE="3" />
    <CHOICEELEMENT CAPTION="Option 4" VALUE="4" />
  </CHOICE>

```

```

  <CHOICE NAME="ch2" MULTI="yes" CAPTION="Multiple answers">
    <CHOICEELEMENT CAPTION="Option 1" VALUE="1" />
    <CHOICEELEMENT CAPTION="Option 2" VALUE="2" />
    <CHOICEELEMENT CAPTION="Option 3" VALUE="3" />
    <CHOICEELEMENT CAPTION="Option 4" VALUE="4" />
  </CHOICE>

```

```

  <CHOICE NAME="ch3" CAPTION="Other field" OTHERFIELD="Other option">
    <CHOICEELEMENT CAPTION="Option 1" VALUE="1" />

```

```

    <CHOICEELEMENT CAPTION="Option 2" VALUE="2" />
    <CHOICEELEMENT CAPTION="Option 3" VALUE="3" />
    <CHOICEELEMENT CAPTION="Option 4" VALUE="4" />
  </CHOICE>

```

```

</SURVEY>

```

### 3.3. CUSTOM

The CUSTOM tag specifies one or more objects, visually designed by the author of the survey. The CUSTOM tag is always open and contains HTML code.

Note that the CUSTOM tag can also be used to insert any HTML data into the survey. It does not necessarily have to specify a variable. Whether it does is specified by the VARIABLES parameter.

**Table 3-5. CUSTOM parameter reference**

Name	Values	Default	Description
ESCAPED	yes/no	yes	Whether the HTML code contained in the tag should be considered "escaped". See explanation below table.
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.

Name	Values	Default	Description
NUMERICAL	yes/no	yes	Whether the variables specified by the tag should be considered numerical when exported.
VARIABLES	A comma-separated list of unique object names		A name is a string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive. If this parameter is left empty, the tag will not be considered as a variable.

Example:

```
<SURVEY TITLE="CUSTOM">
```

```

<CUSTOM ESCAPED="no">
  <h6>No-variable custom content</h6>
  <h5>No-variable custom content</h5>
  <h4>No-variable custom content</h4>
  <h3>No-variable custom content</h3>
  <h2>No-variable custom content</h2>
  <h1>No-variable custom content</h1>
</CUSTOM>

```

```

<CUSTOM ESCAPED="no">
  <br /><br />
  The following is a variable-carrying question designed from scratch:
  <br />
  <br />
</CUSTOM>

```

```
<CUSTOM VARIABLES="cu1" ESCAPED="no">
```

```

<b>I want to go...</b><br /><br />
<table width="300" height="300" cols="3" rows="3" border="1">
  <tr>
    <td> </td>
    <td><center><input type="radio" name="cul"
      value="n" /><br />North</center></td>
    <td> </td>
  </tr>
  <tr>
    <td><input type="radio" name="cul" value="e" /> West</td>
    <td> </td>
    <td>East <input type="radio" name="cul" value="e" /></td>
  </tr>
  <tr>
    <td> </td>
    <td><center>South<br /><input type="radio" name="cul"
      value="s" /></center></td>
    <td> </td>
  </tr>
</table>
</CUSTOM>

</SURVEY>

```

### 3.4. LICKERT

The LICKERT tag specifies a scale with discreet values ranging between two opposites. The default function is to range with five steps between "true" and "false". The LICKERT tag is always closed.

**Table 3-6. LICKERT parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
ACCESS	yes/no	no	If set to yes, an access key to the radiobuttons of the lickert is set, so e.g "1" can be pressed for the fist radio button of the lickert, "2" for the second and so on
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	LICKERTcap	Which style (as specified in a stylesheet) to use for the caption of the object.
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
LEFTCAPSTYLE	A style name	LICKERTleftcap	Which style (as specified in a stylesheet) to use for the label to the left in the object.
LEFTTAG	Any string	True	The left extreme of the scale.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.



Name	Values	Default	Description
REVERSED	yes/no	no	Whether the scale should be counted as having its highest value to the left rather than to the right.
RIGHTCAPTSTYLE	A style name	LICKERTrightcap	Which style (as specified in a stylesheet) to use for the label to the right in the object.
RIGHTTAG	Any string	False	The right extreme of the scale.
STEPS	An Integer	5	The number of scale elements.
STYLE	A style name	LICKERTelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.

Example:

```

<SURVEY TITLE="LICKERT">
<LICKERT NAME="lc1" CAPTION="Plain lickert" />
<LICKERT NAME="lc2" STEPS="15" CAPTION="Large lickert" />
<LICKERT NAME="lc3" CAPTION="Lickert with other tags"
LEFTTAG="Positive" RIGHTTAG="Negative" />
</SURVEY>

```

## 3.5. LIST

The LIST tag specifies a list of string from which the respondent can choose one. The LIST tag may be presented so that only a few of the strings are displayed at a time (so that the respondent has to scroll to view more of the list). A LIST tag is always open and must contain at least one LISTELEMENT.

**Table 3-7. LIST parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	LISTcap	Which style (as specified in a stylesheet) to use for the caption of the object.
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
MAXLEN	An integer	80	How much space should be allocated in data exports for the resulting string.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
NUMERICAL	yes/no		Whether the result should be consider numerical. If yes, then the VALUE parameters of all LISTELEMENTS must be set to an integer value.

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
RANDOM	yes/no	no	Whether the options should be listed in random order. If no, then they will be listed in the order they appear in the survey file.
STYLE	A style name	LISTelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.
VISIBLELEN	An integer	5	How many strings the respondent should be able to see at a time. If set to 1, the object will appear as a drop-down box rather than as a list box.

The LISTELEMENT subtag of the LIST tag represent one respondent-selectable answer to the object's question.

**Table 3-8. LISTELEMENT parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
<i>CAPTION</i>	Any string		The option's description.
<i>VALUE</i>	Any string	The caption of the element	The value this option represents. If NUMERICAL is yes in the CHOICE, then this must be an integer.
SELECTED	yes/no	no	Whether the option should be pre-selected.

The DBLISTELEMENT subtag of the LIST tag represents an array of possible answers. The array(s) must have been either imported via the IMPORT tag, or created in a perl snippet.

**Table 3-9. DBLISTELEMENT parameter reference**

Name	Values	Default	Description
<i>CAPTIONARRAY</i>	An array name		Array to use for the captions.
VALUEARRAY	An array name		Array to use for the values.

Example:

```

<SURVEY TITLE="LIST">
  <LIST NAME="lst1" CAPTION="Plain list">
    <LISTELEMENT CAPTION="Option 1" />
    <LISTELEMENT CAPTION="Option 2" />
    <LISTELEMENT CAPTION="Option 3" />
    <LISTELEMENT CAPTION="Option 4" />
  </LIST>

  <LIST NAME="lst2" CAPTION="Drop-down list" VISIBLELEN="1">
    <LISTELEMENT CAPTION="Option 1" />
    <LISTELEMENT CAPTION="Option 2" />
    <LISTELEMENT CAPTION="Option 3" />
    <LISTELEMENT CAPTION="Option 4" />
  </LIST>
</SURVEY>

```

## 3.6. MATRIX

The MATRIX tag specifies an array of variables, each of which can take a single- or multiselect answer. The MATRIX consists of MATRIXCOLUMN and MATRIXROW tags, which defines the possible answers to all the questions in the array, and the questions themselves. Depending on the settings of the UNIQUECOLUMNS and UNIQUEROWS parameters, either the rows or the columns will be chosen as being the axis carrying the variables.

**Table 3-10. MATRIX parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
BORDER	yes/no	no	Whether a visible grid should be printed, framing the questions and columns.
CAPTALIGN	top/middle/bottom	top	How row texts should be aligned vertically.
CAPTION	Any string		The lead-in text of the MATRIX, printed above the row/column grid.
COLCAPTSTYLE	A style name	MATRIXcolcap	Which style (as specified in a stylesheet) to use for the column captions in the object.
COLWIDTH	An integer	80	The width (in pixels) to allocate for each column, excepting the row text.
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
MULTI	yes/no	no	Whether the respondent should be allowed to choose more than one of the options.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
<i>NAME</i>	A unique object name		A string of no more than 6 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive. Note that a MATRIX's base name can only be six characters long, since the actual variables will be named "name01".. "nameNN" where NN is the number of questions.
RANDOM	none/column/row/both	none	Whether the order of columns and/or rows should be randomized.
ROWCAPTSTYLE	A style name	MATRIXrowcap	Which style (as specified in a stylesheet) to use for the row captions in the object.
STYLE	A style name	MATRIXelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.
TITLESTYLE	A style name	MATRIXtitle	Which style (as specified in a stylesheet) to use for the title of the object.

Name	Values	Default	Description
UNIQUECOLUMNS	yes/no	no	Whether only one answer should allowed per column. If this is yes, and UNIQUEROWS is no, then the columns rather than the rows will define variables. Note that MULTI takes precedence over this.
UNIQUEROWS	yes/no	yes	Whether only one answer should be allowed per row. Note that MULTI takes precedence over this.

The MATRIXCOLUMN defines a column in the MATRIX. In MATRIX's default mode of operation, this is a possible answer to a question defined in a MATRIXROW. A MATRIX block must contain at least one MATRIXCOLUMN.

**Table 3-11. MATRIXCOLUMN parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

Name	Values	Default	Description
<i>CAPTION</i>	Any string		The lead text of this column. In MATRIX's default mode this is a selectable answer to a question, but if UNIQUECOLUMNS is "yes" (in MATRIX), this will be a question lead text whereas the rows will be selectable answers. If both UNIQUECOLUMNS and UNiquerows are "yes", then the ROWS are chosen as carrying the variables, but uniqueness checks will be done on both axes.
VALUE	An integer		The value this column represents in MATRIX's default mode of operation. This will do nothing if in column mode.

The MATRIXROW defines a row in the MATRIX. In MATRIX's default mode of operation, this is a question, with its possible answers defined by MATRIXCOLUMNS. A MATRIX block must contain at least one MATRIXROW.

**Table 3-12. MATRIXROW parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------



Name	Values	Default	Description
<i>CAPTION</i>	Any string		The lead text of this row. In MATRIX's default mode this is a question but if UNIQUECOLUMNS is "yes" (in MATRIX), this will be a selectable answer whereas the columns will define questions. If both UNIQUECOLUMNS and UNIQUEROWS are "yes", then the ROWS are chosen as carrying the variables, but uniqueness checks will be done on both axes.
VALUE	An integer		The value this row represents if in column mode. This will do nothing if in MATRIX's default mode of operation.

Example:

```
<SURVEY TITLE="MATRIX">
```

```

<MATRIX NAME="ma1" BORDER="0" CAPTION="Plain matrix">
  <MATRIXCOLUMN CAPTION="Col 1" VALUE="1" />
  <MATRIXCOLUMN CAPTION="Col 2" VALUE="2" />
  <MATRIXCOLUMN CAPTION="Col 3" VALUE="3" />
  <MATRIXCOLUMN CAPTION="Col 4" VALUE="4" />
  <MATRIXCOLUMN CAPTION="Col 5" VALUE="5" />
  < MATRIXROW CAPTION="Question 1" />
  <MATRIXROW CAPTION="Question 2" //>
  <MATRIXROW CAPTION="Question 3" />
  <MATRIXROW CAPTION="Question 4" />
  <MATRIXROW CAPTION="Question 5" />
</MATRIX>

```

```

<MATRIX NAME="ma2" BORDER="0" MULTI="yes"
CAPTION="Multiple-answer matrix">
  <MATRIXCOLUMN CAPTION="Col 1" VALUE="1" />
  <MATRIXCOLUMN CAPTION="Col 2" VALUE="2" />
  <MATRIXCOLUMN CAPTION="Col 2" VALUE="2" />
  <MATRIXCOLUMN CAPTION="Col 2" VALUE="2" />
  < MATRIXCOLUMN CAPTION="Col 3" VALUE="3" />
  <MATRIXCOLUMN CAPTION="Col 4" VALUE="4" />
  <MATRIXCOLUMN CAPTION="Col 5" VALUE="5" />
  <MATRIXROW CAPTION="Question 1" />
  <MATRIXROW CAPTION="Question 2" />
  <MATRIXROW CAPTION="Question 3" />
  <MATRIXROW CAPTION="Question 4" />
  <MATRIXROW CAPTION="Question 5" />
</MATRIX>

<MATRIX NAME="ma3" BORDER="0" RANDOM="row"
CAPTION="Randomized question order (reload to see change)">
  <MATRIXCOLUMN CAPTION="Col 1" VALUE="1" />
  <MATRIXCOLUMN CAPTION="Col 2" VALUE="2" />
  <MATRIXCOLUMN CAPTION="Col 3" VALUE="3" />
  <MATRIXCOLUMN CAPTION="Col 4" VALUE="4" />
  <MATRIXCOLUMN CAPTION="Col 5" VALUE="5" />
  < MATRIXROW CAPTION="Question 1" />
  <MATRIXROW CAPTION="Question 2" />
  <MATRIXROW CAPTION="Question 3" />
  <MATRIXROW CAPTION="Question 4" />
  <MATRIXROW CAPTION="Question 5" />
</MATRIX>

</SURVEY>

```

## 3.7. MEMO

The MEMO tag specifies an open answer multi-line text field.

**Table 3-13. MEMO parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	MEMOcap	Which style (as specified in a stylesheet) to use for the caption of the object.
COLS	An integer	50	How many columns wide the text field should be.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
ROWS	An integer	15	How many rows high the text field should be.
STYLE	A style name	MEMOelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.

Example:

```
<SURVEY TITLE="MEMO">
  <MEMO NAME="me1" CAPTION="Plain memo" />
</SURVEY>
```

## 3.8. TEXT

The TEXT tag specifies an open free-text one line field. The field can be limited to only accepting numerical values within a range. The TEXT tag is always closed.

**Table 3-14. TEXT parameter reference**

Name	Values	Default	Description
CAPTION	Any string		The lead-in text of the object, usually its question string.
CAPTSTYLE	A style name	TEXTcap	Which style (as specified in a stylesheet) to use for the caption of the object.
DEFAULT	Any string		The default String which is displayed in the field
ILLEGALVAL	An integer	-1	What the resulting value should be if the respondent did not answer the question.
MAXLEN	An integer	80	How much space should be allocated in data exports for the resulting string. Also the maximum number of characters possible to type in the field.
MAXVAL	An integer		If set, the answer will be checked so that it is not higher than the given integer. NUMERICAL must be yes for this to have any effect.

Name	Values	Default	Description
MINVAL	An integer		If set, the answer will be checked so that it is not lower than the given integer. NUMERICAL must be yes for this to have any effect.
MUSTANSWER	yes/no	no	Whether the respondent should be forced to give an answer.
NAME	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
NUMERICAL	yes/no/float		Whether the result should be consider numerical. If yes, then input string will be format checked as an integer. If float, it will be format checked as a float value.
STYLE	A style name	TEXTelm	Which style (as specified in a stylesheet) to use for the user-input part of the object.
UNIT	Any string		Label to use to the right of the text field.

Example:

```
<SURVEY TITLE="TEXT">
<TEXT NAME="tel" CAPTION="Plain text" />
```

```
<TEXT NAME="te2" MAXLEN="2" CAPTION="Numerical text"  
MUSTANSWER="yes" NUMERICAL="yes" MAXVAL="99" MINVAL="0" ILLEGALVAL="-2" />  
</SURVEY>
```

# Chapter 4. Invisible variable-carrying tags

## 4.1. CONSTANT

The CONSTANT tag specifies a constant value, which is not affected in any way by the user. This could, for example, be the revision of the questionnaire. The CONSTANT tag is always closed.

**Table 4-1. CONSTANT parameter reference**

Name	Values	Default	Description
MAXLEN	An integer	80	Space to allocate for this value in data exports.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.
<i>VALUE</i>	Any string		The value this constant represents.

Example:

```
<SURVEY TITLE="constant">  
  <CONSTANT NAME="modified" VALUE="Modified by Joel, 2004-06-20" />  
</SURVEY>
```

## 4.2. DATETIME

The DATETIME tag represents a time and/or datestamp of the moment when the respondent

submitted his answers. The actual contents of the stamp is set in the FORMAT parameter. The DATETIME tag is always closed.

**Table 4-2. DATETIME parameter reference**

Name	Values	Default	Description
FORMAT	Date/time format	#Y-#M-#D #h:#m#s	The format of the date/time stamp. See table below.
MAXLEN	An integer	80	Space to allocate for this value in data exports.
NAME	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.

The FORMAT parameter contains character codes representing different parts of a date/time stamp.

**Table 4-3. FORMAT codes**

Code	Represents	Example
#Y	four-digit year	2003
#M	two-digit month	04
#D	two-digit day of month	05
#d	three-digit day of year	290
#a	abbrev month name	"feb"
#A	full month name	"february"
#w	abbrev weekday name	"tue"
#W	full weekday name	"tuesday"
#h	two-digit hour (24)	23
#m	two-digit minute	15
#s	two-digit second	20



Code	Represents	Example
#e	seconds since epoch	1066464551

Example:

```
<SURVEY TITLE="timedemo">
<DATETIME NAME="time" />
</SURVEY>
```

### 4.3. ENV

The ENV tag fetches a value from the environment, such as the respondent's IP number, or the server's apache version. The ENV tag is always closed.

**Table 4-4. ENV parameter reference**

Name	Values	Default	Description
<i>FIELD</i>	An environment variable		Which environment variable should be read. See table below.
MAXLEN	An integer	80	Space to allocate for this value in data exports.
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.

There is no restriction in which environment variables that can be read. This is completely dependent on which variables are set by the server. Thus, there is no way to list all possible

variables here. However, there are a number of common variable names which are often used in web server environments.

**Table 4-5. Examples of environment variables**

Name	Function	Example output
REMOTE_HOST	The IP address of the respondent's client	10.251.0.42
REMOTE_ADDR	The name-resolved address of the respondent's client	gathering.itm.mh.se
REMOTE_USER	If the respondent is authenticated, then this is his name	Joel Palmius
HTTP_USER_CLIENT	The ID string of the respondent's client software	Mozilla/4; MSIE 5, Win98; Opera

Example:

```
<SURVEY TITLE="env_test">
<ENV NAME="ip" FIELD="REMOTE_ADDR" MAXLEN="50" />
<ENV NAME="host" FIELD="REMOTE_HOST" MAXLEN="50" />
<ENV NAME="user" FIELD="REMOTE_USER" MAXLEN="50" />
<ENV NAME="client" FIELD="HTTP_USER_CLIENT" MAXLEN="50" />
</SURVEY>
```

## 4.4. IMPORT

The IMPORT tag fetches one or more variables/values from a database table and inserts them into the survey as if they had been submitted by the user. The IMPORTed variables are treated as any other variables and are thus possible to use for routing and in dynamic markup.

The IMPORT tag can contain one FILTER subtag and must contain at least one VARIABLE subtag.

**Table 4-6. IMPORT parameter reference**

Name	Values	Default	Description
CAPTION	Any String	Import	The caption which is displayed if visible is set to yes
DBITABLE	A database table name		The database table from which we want to fetch data.
DBIDSN	A database DSN		The connection string to use for connecting to a database. For PostgreSQL users this could for example be "DBI:Pg:dbname=mydb".
DBIUSER	A database user		The user string to use when connecting to a database.
DBIPASS	A database password		The password string to use when connecting to a database.
VISIBLE	yes/no	no	Whether the imported variables should be displayed as a table in the survey.

Each VARIABLE subtag defines a variable fetched from a field in the database table specified in the IMPORT tag.

**Table 4-7. VARIABLE parameter reference**

Name	Values	Default	Description
<i>FIELD</i>	A table column name		Which database table column/field this variable should be read from.

Name	Values	Default	Description
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive. This is the <i>name</i> of the new variable, and is what you will refer to in the rest of the survey. It <i>can</i> be the same as the FIELD property, but does not have to be.
WRITABLE	yes/no	no	(NOT IMPLEMENTED) Whether it should be possible to give this variable a new value, which should then be saved back to the table field.

The FILTER subtag defines the criteria for narrowing the selection from the database table specified in the IMPORT tag. If no FILTER tag is specified, the first record in the table will be used.

Note that you can use the dynamic markup in the FILTER parameters. Thus, it's perfectly OK to say CRITERIA="userid={\$uid\$}" if "uid" was, say, a TEXT occurring earlier in the survey chain.

**Table 4-8. FILTER parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
CRITERIA	A SQL-compatible boolean statement		The filter string narrowing the selection from the table. For people knowing SQL, this is the part immediately after WHERE.
GROUP	A comma-separated list of field names		Group selection result based on the values in these fields. For people knowing SQL, this is the part immediately after GROUP BY.
GROUPCRITERIA	A SQL-compatible boolean statement		The filter string narrowing the selection to be applied to the group defined by the GROUP parameter. For people knowing SQL, this is the part immediately after the HAVING clause

Each ARRAY subtag constructs a new array from all the values of a table column.

**Table 4-9. ARRAY parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
<i>FIELD</i>	A table column name		Which database table column/field this array should be read from.
<i>NAME</i>	A array variable name		What to call the array. This is the string used for example in DB-CHOICEELEMENT later

Example:

```

<SURVEY TITLE="import_example">
  <IMPORT DBIDSN="dbi:mysql:database=testbase" DBIUSER="somebody"
  DBITABLE="testsurvey" VISIBLE="no" DBIPASS="secret">
    <ARRAY NAME="contname" FIELD="name_field" />
    <ARRAY NAME="contid" FIELD="id_field" />
  </IMPORT>

  <LIST NAME="acontact" CAPTION="Which contact are we talking about?" >
    <DBLISTELEMENT CAPTIONARRAY="contname" VALUEARRAY="contid" />
  </LIST>
</SURVEY>

```

## 4.5. TIMER

The TIMER tag represents the time (in seconds) it took for the respondent to answer the survey (in practise time from load to submit). The TIMER tag is always closed.

**Table 4-10. TIMER parameter reference**

Name	Values	Default	Description
<i>NAME</i>	A unique object name		A string of no more than 8 character, beginning with an alphabetical letter (a-z) and containing only alphanumerical characters (a-z, 0-9). The name is case-sensitive.

Example:

```

<SURVEY TITLE="timer_demo">
<TIMER NAME="timer" />
</SURVEY>

```

# Chapter 5. Tags related to multipaging

## 5.1. CASEROUTE

The CASEROUTE defines a conditional branching depending on the value of one previously submitted value. The CASROUTE basically consists of a SWITCH parameter telling which variable to evaluate, and a number of CASE subtags defining possible values and routes.

**Table 5-1. CASEROUTE parameter reference**

Name	Values	Default	Description
DEFAULT	An URL		Go to this location if none of the CASEs matched.
<i>SWITCH</i>	A variable name		Evaluate the CASEs against this variable.

The CASEROUTE tag must contain at least one CASE tag.

**Table 5-2. CASE parameter reference**

Name	Values	Default	Description
<i>CONTINUE</i>	An URL		Go to this location if VALUE matched the variable value.
<i>VALUE</i>	Any value		Evaluate this value against what was submitted in the SWITCH variable.

Example:

```
<SURVEY TITLE="Survey with CASE routing" CHECKKEY="no" >

  <SECURITY REQAUTH="no">
    <ANSWER LEVEL="open" />
    <DATA LEVEL="open" />
```

```

    <SOURCE LEVEL="open" />
</SECURITY>

<CHOICE CAPTION="Are you a worker or a boss" NAME="type">
    <CHOICEELEMENT CAPTION="worker" VALUE="0" />
    <CHOICEELEMENT CAPTION="boss" VALUE="1" />
</CHOICE>

<CASEROUTE SWITCH="type" DEFAULT="error.html">
    <CASE VALUE="0" CONTINUE="worker.survey" />
    <CASE VALUE="1" CONTINUE="boss.survey" />
</CASEROUTE>
</SURVEY>

```

## 5.2. IFROUTE

The IFROUTE defines a conditional branching dependent on one or more boolean conditions, such as whether a variable value is less than a certain constant. The IFROUTE tag contains IF subtags, each of which defines a condition by in its turn containing a list of boolean conditions (see below).

**Table 5-3. IFROUTE parameter reference**

Name	Values	Default	Description
DEFAULT	An URL		Go here if none of the IF statements matched.

The IFROUTE tag must contain at least one IF subtag.

**Table 5-4. IF parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------



Name	Values	Default	Description
BOOLOP	and/or	and	If there are several contained boolean conditions, combine them using this operator.
ELSE	An URL		If the IF <i>did not</i> match, then go to this URL.
<i>THEN</i>	An URL		If the IF matched, then go to this URL.

Each IF tag must contain at least one of the following subtags:

**Table 5-5. Boolean conditions**

Name	Function
EQUALS	The VARIABLE value is exactly VALUE
NOTEQUALS	The VARIABLE value is anything but VALUE
LESSTHAN	The VARIABLE value is less than VALUE
MORETHAN	The VARIABLE value is more then VALUE
LESSEQUALS	The VARIABLE value is less than or equal to VALUE
MOREEQUALS	The VARIABLE value is more then or equal to VALUE

All boolean condition tags have the following parameters.

**Table 5-6. Boolean condition parameter reference**

Name	Values	Default	Description
<i>VALUE</i>	An integer		Value to evaluate against.
<i>VARIABLE</i>	An variable name		Variable to evaluate against VALUE.

Example:

```

<SURVEY TITLE="Survey with XML correspondence to IF-THEN" CHECKKEY="no" >

<CUSTOM ESCAPED="no">
  <h1>IF-THEN</h1>
  Only answers for ages less than 26 are accepted. Enter a higher age to
  see an error page.<br /><br />
</CUSTOM>

<TEXT NAME="age" CAPTION="age" NUMERICAL="yes" MUSTANSWER="yes" />

<CHOICE CAPTION="gender" NAME="gender" MUSTANSWER="yes">
  <CHOICEELEMENT CAPTION="male" VALUE="0" />
  <CHOICEELEMENT CAPTION="female" VALUE="1" />
</CHOICE>

<IFROUTE DEFAULT="error.html">
  <IF THEN="male0-15.survey">
    <LESSTHAN VARIABLE="age" VALUE="16" />
    <EQUALS VARIABLE="gender" VALUE="0" />
  </IF>
  <IF THEN="female0-15.survey">
    <LESSTHAN VARIABLE="age" VALUE="16" />
    <EQUALS VARIABLE="gender" VALUE="1" />
  </IF>
  <IF THEN="male16-25.survey">
    <LESSTHAN VARIABLE="age" VALUE="26" />
    <MORETHAN VARIABLE="age" VALUE="15" />
    <EQUALS VARIABLE="gender" VALUE="0" />
  </IF>
  <IF THEN="female16-25.survey">
    <LESSTHAN VARIABLE="age" VALUE="26" />
    <MORETHAN VARIABLE="age" VALUE="15" />
    <EQUALS VARIABLE="gender" VALUE="1" />
  </IF>
</IFROUTE>
</SURVEY>

```

## 5.3. RANDOMROUTE

The RANDOMROUTE defines a conditional branching where the next page is randomly picked from a list.

**Table 5-7. RANDOMROUTE parameter reference**

Name	Values	Default	Description
DISTRIBUTION	square or normal	square	[not implemented].
MEAN	An integer	1	[not implemented].

The RANDOMROUTE tag must contain at least one ALTERNATIVE tag.

**Table 5-8. ALTERNATIVE parameter reference**

Name	Values	Default	Description
<i>CONTINUE</i>	An URL		Go to this location if this alternative is picked.

Example:

```
<SURVEY TITLE="Randomroute">
  <TEXT NAME="first" CAPTION="firstpage" />

  <RANDOMROUTE>
    <ALTERNATIVE CONTINUE="alt1.survey" />
    <ALTERNATIVE CONTINUE="alt2.survey" />
    <ALTERNATIVE CONTINUE="alt3.survey" />
    <ALTERNATIVE CONTINUE="alt4.survey" />
    <ALTERNATIVE CONTINUE="alt5.survey" />
  </RANDOMROUTE>
</SURVEY>
```

## 5.4. ROUTE

The ROUTE tag defines an unconditional route. This should not be confused with the CONTINUE parameter in the SURVEY tag. By using the ROUTE tag, one sends the respondent to the next page in a survey containing several pages. With SURVEY's CONTINUE, the respondent is sent to a page outside the survey.

**Table 5-9. ROUTE parameter reference**

Name	Values	Default	Description
<i>CONTINUE</i>	An URL		Go to this URL when respondent has submitted the answers on this page.

**Example**

```

<SURVEY TITLE="Plain (non-branching) sequence" >

  <CHOICE CAPTION="gender" NAME="gender">
    <CHOICEELEMENT CAPTION="male" VALUE="0" />
    <CHOICEELEMENT CAPTION="female" VALUE="1" />
  </CHOICE>

  <ROUTE CONTINUE="route-part2.survey" />
</SURVEY>

```

**5.5. SEQUENCE**

When having written a survey with more than one page, the *last* page must contain a SEQUENCE tag, listing all the pages that were contained in the survey. This should also include pages that the user did not see because of having been routed past them. The SEQUENCE tag must contain at least one FILE subtag.

When finally submitting or downloading data, it is the SEQUENCE tag which tells the system which pages (and in consequence variables) to include.

**Table 5-10. SEQUENCE parameter reference**

Name	Values	Default	Description
------	--------	---------	-------------

Name	Values	Default	Description
SELFINCLUDE	yes/no	no	The page containing this SEQUENCE is listed in a FILE below. (CURRENTLY DOES NOTHING)

The SEQUENCE tag must contain at least one FILE subtag. Each FILE subtag points to one of the pages that were contained in the survey. Filenames may be absolute or relative, but may not be URLs.

**Table 5-11. FILE parameter reference**

Name	Values	Default	Description
<i>FILENAME</i>	A filename		One survey page to include in the SEQUENCE.

Example:

```
<SURVEY TITLE="sequence_selfinclude">
<CUSTOM ESCAPED="no">
<HTML>
<h1>This is the last page of the survey </h1>
<h2> The survey also includes example1.survey, example2.survey <br>
and example3.survey </h2>
</HTML>
</CUSTOM>
<SEQUENCE SELFINCLUDE="yes">
<FILE FILENAME="example1.survey"/>
<FILE FILENAME="example2.survey"/>
<FILE FILENAME="example3.survey"/>
<FILE FILENAME="last.survey"/>
</SEQUENCE>
</SURVEY>
```



# Chapter 6. Layout-related and other tags

## 6.1. NEWLINE

The NEWLINE tag inserts one or more linefeeds in the visible output. This is usually used to get more space between question objects. See also the SURVEY tag's AUTONEWLINES parameter. The NEWLINE tag is always closed.

**Table 6-1. NEWLINE parameter reference**

Name	Values	Default	Description
COUNT	An integer	1	Insert this many linefeeds (  HTML tags) in the visible output.

Example:

```
<SURVEY TITLE="newlinelisting">
<TEXT NAME="age" CAPTION="How old are you"/>
<NEWLINE COUNT="5"/>
<CHOICE NAME="gender" CAPTION="Are you male or female">
<CHOICEELEMENT CAPTION="male" VALUE="1"/>
<CHOICEELEMENT CAPTION="female" VALUE="2"/>
</CHOICE>
</SURVEY>
```

## 6.2. COMMENT

The COMMENT tag usually does nothing. It is available as a support for the author of the survey, so he can insert notes of his own in the survey file. The COMMENT tag is always open and may contain any text.

**Table 6-2. COMMENT parameter reference**

<b>Name</b>	<b>Values</b>	<b>Default</b>	<b>Description</b>
EMBED	yes/no	no	If yes, the comment will be included as a HTML comment in the HTML output.

Example:

```
<SURVEY TITLE="Comment_Example">  
  <COMMENT EMBED="yes">  
    This is a comment  
  </COMMENT>  
</SURVEY>
```



# Chapter 7. Extra HTML markup

This chapter specifies the codes used for inserting extra HTML code in the survey output. This can, for example, be used for changing colors or boldfacing text.

Note that this markup is in development and may change slightly during the versions to come. Further, it can be expected to be occasionally somewhat buggy.

## 7.1. Color

Color codes can be inserted in any visible output, such as in a question's CAPTION parameter. It is the used in the form "This is some {red}amusingly colored{/red} text". With this example "amusingly colored" would be red while the rest of the text would have the default text color (see the TEXTCOLOR parameter of the SURVEY tag).

The following color codes are defined:

{red}	{/red}
{green}	{/green}
{blue}	{/blue}
{yellow}	{/yellow}
{purple}	{/purple}
{cyan}	{/cyan}
{black}	{/black}
{white}	{/white}

Other colors can be defined manually using the custom HTML markup.

## 7.2. Font styles

Font styles can be inserted in any visible output, such as in a question's CAPTION parameter. It is the used in the form "This is some {i}emphasized{/i} text". With this example "emphasized" would be in italics while the rest of the text would be plain.

The following style codes are defined:

{b}	{/b}	(boldface)
{i}	{/i}	(italics)
{u}	{/u}	(underline)

## 7.3. Graphics

The currently only graphics-related special markup is one used for pasting images in visual output. By writing, for example, `CAPTION="{image}happyface.jpg{/image}"`, the image "happyface.jpg" would appear where the caption text would normally be.

## 7.4. Special

Special characters can be inserted in any visible output, such as in a question's `CAPTION` parameter. A special character is a character which does not mix well with XML syntax, and which therefore have to be coded. Examples of such characters are " and >.

An example would be having the following `CAPTION`: "I think x {[} y", which would be printed as "I think x < y".

The following special characters are defined:

{-}	=	(insert an equal mark)
{:}	;	(insert a semi-colon)
{'}	"	(insert a citation mark)
{[}	<	(insert a "less than")
{]}	>	(insert a "more than")
{A}	&	(insert an ampersand)

## 7.5. Custom HTML markup

By using the special character markup, it is possible to insert any HTML code in the output. It is possible, but somewhat cumbersome. For example, inserting a `<br>` in a `TEXT`'s `CAPTION` would look like this:

```
<TEXT NAME="test" CAPTION="Line 1{[}br{]}Line 2" />
```

In other words, one has to insert the special escape codes for the parts of the HTML markup.

# Chapter 8. Dynamic content

## 8.1. Including external files

External files may be included into the survey at two different stages during the survey' life cycle. The first is *before* parse. This causes the contents of the linked file to be parsed as if it was a part of the survey code. The second is *after* parse, in which case the contents will simply be pasted verbatim into the visual output.

```
{ @filename@ } (Include "filename" before parse)
{ ¶filename¶ } (Include "filename" after parse)
```

The before parse link can be inserted anywhere in the survey file. The after parse link can only be inserted in visual output, such as in a CUSTOM block or in a CAPTION.

## 8.2. Referencing previous variables

References to variable answers submitted on an earlier page can be inserted in all visible output, such as in a question's CAPTION. The types of references available are:

```
{ $variable$ } (Insert variable as-is)
{ %variable% } (Insert caption corresponding to variable value)
{ !variable/n1:output,n2:output...! } (Selection, choose output depending on variable value)
```

For example, an as-is insertion could be:

```
<TEXT NAME="test1" CAPTION="How old are you, {$respname$}?" />
```

.. which would result in, for example, the caption "How old are you, Joel?"

Suppose a previous page contained a CHOICE listing possible fruits. A caption insertion example could then be:

```
<LICKERT NAME="test2" CAPTION="Do you want a {%fruitsel%}?" />
```

.. which would result in, for example, the caption "Do you want a banana?"

Suppose a previous page contained a CHOICE for selecting respondent gender. A selection insertion could then be:

```
<TEXT NAME="test3" CAPTION="What is your name, {!gender/0:boy,1:girl!}?" />
```

.. which would result in, for example, the caption "What is your name, boy?"

### 8.2.1. Session variables

Session variables can be referenced via `{_[name]}`. A session variable do not necessarily have any connection to a question object, and some knowledge of the internals of `Mod_Survey` will be required to actually have some use of this feature.

## 8.3. Perl snippets

Perl snippets allows executable code to be inserted into the survey without voiding the system security. This is possible since all the included perl code is passed through the `Safe` module, which denies all forms of system access.

Snippet blocks are executed either before parse, after parse or submit time. Thus a snippet can be used to produce survey code. If printed as `{&..&}`, it will be executed before parse, if `{\..}` after parse and if `{|..}` at submit time. Snippets executed after parse can only be inserted into visible output, such as `CUSTOM` blocks or `CAPTIONS`, whereas snippets executed before parse can be put anywhere in the code. Submit-time perl can only be placed within the `SUBMIT` tag.

```
{& .. code to execute before parse .. &}
{\ .. code to execute after parse .. \}
{| .. code to execute submit time .. |}
```