Table of Contents

1 Category:GettingStarted

1.1 FlowchartWiki - Wiki-based Process Modelling and Documentation

Simple Navigation: click on an Icon in the auto-generated graph on the left hand side.
See for the diagram each page (including diagrams on the left) are automatically created as pages in the wiki.
Just click on a Process in the diagram to go directly to a page.

FlowchartWiki is an extension to the well known Mediawiki Software - both are **free and open source**.

Download Documentation as .pdf
To obtain a copy of this GettingStarted guide please click on the `[export pdf]` tab at the top of the screen. A pdf document is created "on the fly" by the PDFbook extension - this can be downloaded as well.

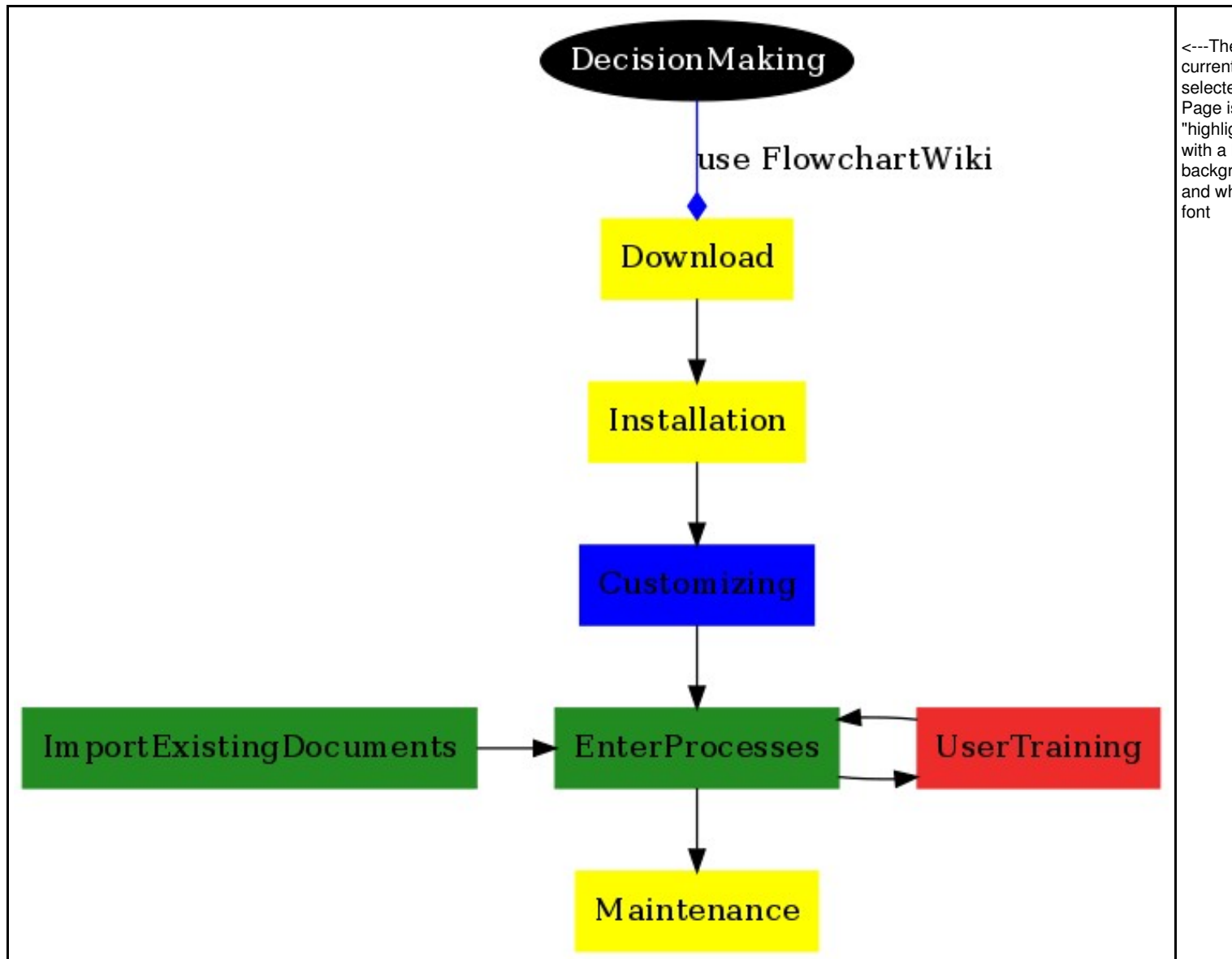ModelType::Draw Type::Process

1.2 Make it Sub-Process of ProcessList

FlowchartWiki supports hierarchical processes: Processes inside processes. See Category:ProcessList for a hierarchical diagram.
Type::ProcessType::Rect_Red Level::1020

2 DecisionMaking

2.1 Free and Open Source

- Both, MediaWiki and the FlowchartWiki Extension, are free and Open Source and licensed under the GPL General Public License
- Download for free, install it and use it for free
- Modify the source if you wish

2.2 5 Minute Introduction to FlowchartWiki - Business

- Use a Wiki to simplify the self-organizing of Teams and Business Processes
- Simple and easy to use - very low training requirements
- make the relevant information available to the person performing the task when and where it is needed - instead of hiding it in rarely updated binders somewhere hidden on a shelf
- Team-Members can immediately follow up and contribute which results in
- ongoing and up-to-date documentation of business processes and procedures
- Integrate the process documentation into the daily work - by adding checklists and tips & tricks to the documentation
- Process improvement (Kaizen) is simplified: current documentation is available and incremental changes are easily documented and implemented
- Proven MediaWiki Software - used by Wikipedia with thousands of pages and millions of users
- just access it with your browser from any PC - no Software to install or distribute
- simultaneous access - no files to share or distribute, no versions to monitor
- export complete process documentation as a .pdf Document for archiving or other purposes

- access to the documentation may be shared with "extended Valuestream Partners" - Customers or Suppliers to streamline the extended valuestream
- easily convert your existing .doc based documentation and import it into the wiki

From a consultants perspective:

- enable the client employees to "think process"
- the work of the consultant in defining and documenting processes is immediately visible and transparent to the client
- due to low training requirements, client employees can be easily integrated into the project and can contribute early on by adding information or giving feedback
- easy and early integration of client employees results in client buy-in and quick wins

2.3 5 Minute Introduction to FlowchartWiki - Technical

- Based on Mediawiki
- Built on the popular "LAMP" Stack: Linux/Windows/Unix, Apache, mySQL/PostgreSQL, PHP
- FlowchartWiki is implemented as an Extension to MediaWiki and adds just one table to the MediaWiki database
- uses GraphViz Graph creation Software to create the graphs
- uses htmldoc and an extended pdfbook extension to create .pdf documentation of processes
- implements a set of custom tags to create/display the graphs (see "How is it done, exactly?" below)
- supports hierarchical categories - Processes of Processes (of Processes of...)
- MediaWiki User-Management can be integrated with LDAP or other tools

2.4 How is it done, exactly?

After installing the Extension and adding a few tags to Category and Process pages, you are ready to run. This is a brief explanation on how it is done. Please see the full documentation for all the details.)
A **category page** represents a Process. Three Tags are added to a category page:

```
<CategoryBrowser />
[[ModelType::Draw]]
[[Type::Process]]
```

- CategoryBrowser: The CategoryBrowser tag displays the process diagram on the current page.
- ModelType: The "ModelType" tag selects a Process Type and defines the shapes and colors used for the diagrams. used. (This could be EPK, FlowChart, etc. and is fully customizable).
- Type::Process tags this page as being a Process, when added to another category as a sub-process.

A **Wiki Page** represents a step within a process. Only a few tags are added to a single wiki page:

```
[[NextStep::Customizing]]
<Dependencies />
[[Type::Rect_Green]]
[[Level::0995]]
[[Category:GettingStarted]]
```

- NextStep::Customizing tags a Link to a "next process step". NextStep could also be "performedBy", "uses", etc. and is freely assignable. This describes the type of link to another wikipage. There is no limit in linking to other pages.
- Dependencies: This tag creates a table inside the current wikipage that shows "who links here" and "where do I link to" including the types of links. Using this tag is optional.
- Type::Rect_Green: Describes the type of this page and determines via the customizing what type of shape and color is used for displaying this process step in the diagram. Here we used a Modeltype::Draw tag in the process definition that is customized with having types like Rect_Green (green rectangle), Rect_Red, Rect_Blue. The EPK customizing includes types like Event, Decision, Function, Datasource, Person, Department, Product, each with its own shape and color settings.
- Level::0995: The automatic flowcharting needs some hints on where to place the process step into the diagram. We are using a line number system. All Process steps with the same line number will show up on the same line in the diagram. Higher line numbers will be displayed on a lower level. That´s the only option you have to modify the diagram - sorry, no more hours spent with beefing up your slideshows...
- Category:GettingStarted: This links the process step to the process, by linking it to the category page. This tag is standard MediaWiki.

That´s it.

2.5 Features

- Creates Diagrams from the links between WikiPages.
- All Pages in one category that are tagged will be shown in the diagram
- Hierarchical Processes: One Process can contain other processes, so you can drill down to lower levels.
- Each WikiPage shows the whole process and the current step is marked.
- configurable display of the process:
    - whole process with marked step
    - whole process plus extract of process steps "around" the currently selected one

- customizable display: Instead of "drawing" diagrams, a Type is assigned to the process step which determines the shape and color used for drawing. This allows a standardisation of the diagrams and displays. (Think of EPK/Aris Diagrams or other types of diagrams.)

2.6 Optional Features

- Export the whole process documentation to a .pdf by using the PDFbook Extension. (Including the graphs) (We currently provide an extended version of the original).

2.7 Limitations

- Charting / GraphViz
    - FlowchartWiki is using GraphViz to automagically create the charts.
    - GraphViz offers very limited control of chart appearance or layout. (We try to give GraphViz some hints by using "line-numbers")
    - Swimlane charts or manual placement of elements is not supported.
- Cacheing / Performance
    - FlowchartWiki currently does not support cacheing of pages. It will test on each pageload, if the diagram needs to be recreated.

2.8 Continue with Installation

- Test-Drive FlowchartWiki on your own system or your internal network: you can freely download and install the software.
- Download now: Install::Download

Type of page 'DecisionMaking': Elli_Blue
Where do I link to: **Who links here:**
  Download (Install)
Type::Elli_Blue Level::0990
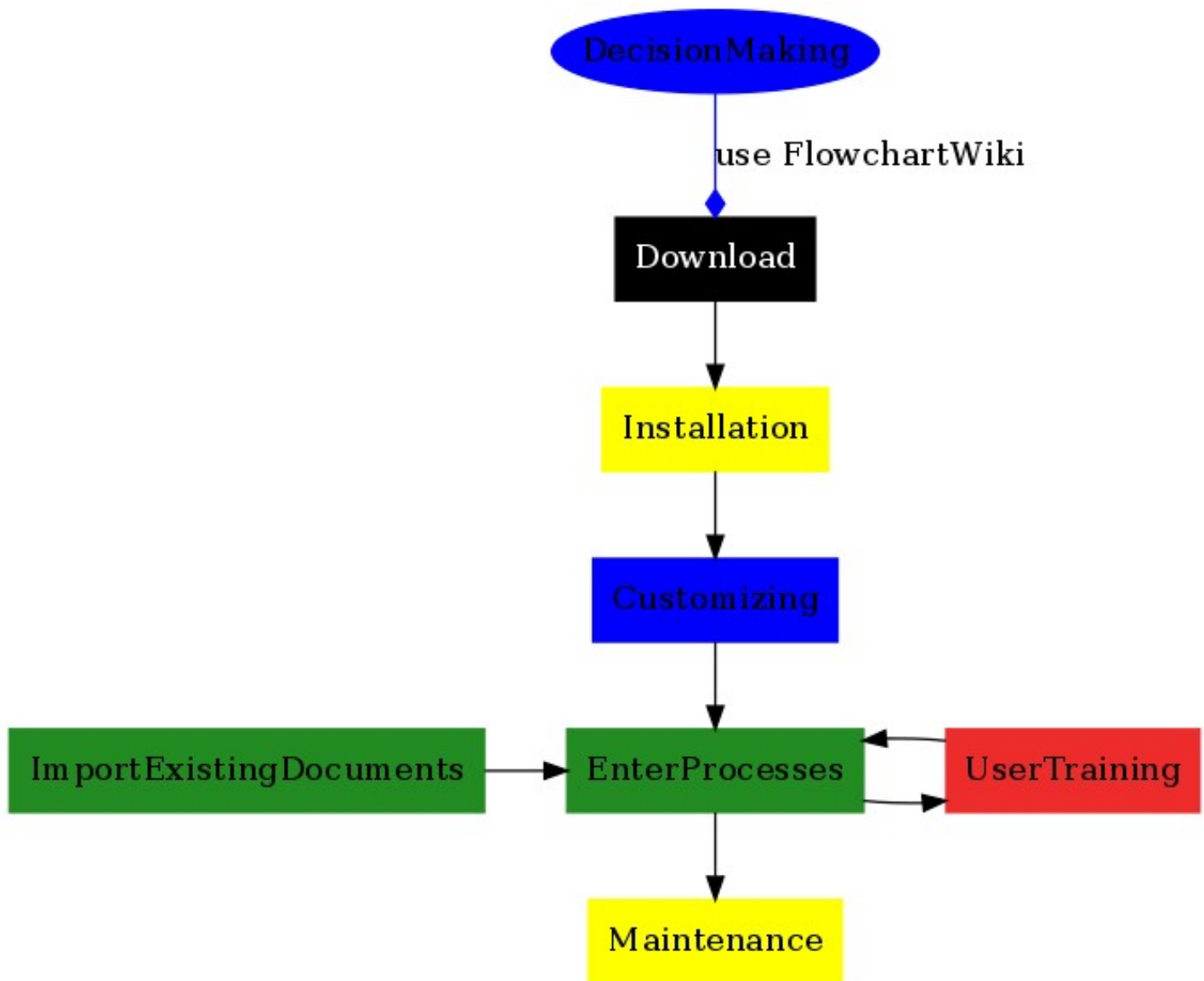
3 Download

3.1 Download

3.1.1 Current Version FlowchartWiki 1.2.0, PdfBook 1.1.1

(If you are running an older version, please see the upgrade info in the release-notes below.)

There are two versions of the files available:

- a .zip file for Windows installations
- a .tar.gz file for Linux/Unix installations

The contents of both files are the same, just the packaging is different.

- FlowchartWiki
    - Please download the files from Sourceforge
- PDFbook Extension (extended version of [[1]]) (License: LGPL [[2]]
    - Please download the files from Sourceforge

3.1.2 Installation prerequisites

- FlowchartWiki and MediaWiki are based on the "LAMP" Stack: Linux/Unix/Windows, Apache, mySQL/Postgres, PHP. This platform should be available on your system.
- FlowchartWiki is an Extension for MediaWiki. Please install MediaWiki first, then add the FlowchartWiki Extension. (See also Installation)
- The standard installation of FlowchartWiki will NOT work together with Semantic MediaWiki (SMW) installed in the same wiki. (See SMW+FlowchartWiki for a workaround.)

3.2 Unpacking and installing

Please see Installation

3.3 Release and upgrade Notes - Current Release FlowchartWiki

3.3.1 FlowchartWiki 1.2.0

- Bugfix by Hiroyuki S.: long pagenames in fchw_LoadPages() in lib.php
- catching some php-Notice "undefined index" errors in lib.php with isset()
- **Upgrade from 1.1.x:** Replace the extension directory with the new version and delete the images in ./images/flowchartwiki

3.3.1.1 FlowchartWiki 1.2.0-beta2

- Bugfix by Gerrit I. - double entries in Database when tags are twice in a wikipage
- PHP 5.3 compatibility - tested together with Peter v.L.
- Changed field length in schema_mysql.sql for from_title and relation from 255 to 120 due to problems with UTF8 databases and indices being longer than 1024.
- added $wgExtensionCredits

3.3.1.2 FlowchartWiki 1.2.0-beta1

- major rebuild of .dot file creation, reduced Database-accesses and moved data-structure into internal object hierarchy (see fchwobjects.php for the objects)
- This was a prerequsite for adding other features to the graph:
- Include a Link to the Category-Page on top of the graph, if Customizing has a Page-Type "Category".
- Position all Pages that have no Level::xxxx assigned at the bottom of the graph, sorted in alphabetical order.
- These pages CAN have a Type::xxxx assigned to use a shape and Color from the customizing. Links to these pages may work, but will probably screw up the graph :-)
- add Customizing-Function fchw['zLevels'] for the number or items per row for the "Unassigned" Pages.
- This can be set in LocalSettings.php. If not set, it defaults to 4 and is set in flowchartwiki.php.
- Calculate the Height of the graph based on the approximate number of rows. (This may not work properly for CategoryBrowser2 which displays two images.)

3.3.2 FlowchartWiki 1.1.1

- fixed bug with ' (Apostroph) and " " (Blank) in Category Names. (Many thanks to Martin from Tübingen, Germany to report this.)

Upgrading from 1.1.0 to 1.1.1

- Replace the existing /extensions/flowchartwiki installation with the new files. (You may want to backup your existing files first.)
- delete the contents of the `./images/flowchartwiki` directory and run `php ./extensions/flowchartwiki/maintenance/fchw_RefreshPages.php`

Upgrading from previous versions to FlowchartWiki 1.1.1:

- replace the existing /extensions/flowchartwiki installation with the new files. (You may want to backup your existing files first.)
- change the name of the `./images/fchw` directory to `./images/flowchartwiki`. Please ensure you keep the proper read/write/create/delete authorizations.
- test your installation with the Special Pages:Check_Fchw Page.
- change the customizing pages for the process models to the new Version 1.1 format shown in Customizing
- delete the contents of the `./images/flowchartwiki` directory and run `php ./extensions/flowchartwiki/maintenance/fchw_RefreshPages.php`

3.3.3 FlowchartWiki 1.1.0

- updated Special Pages:Check_Fchw
- changed calls to GraphViz to platform-dependant calls for Unix/Linux, Windows-Apache and Windows-IIS platforms
- fixed calls to GraphViz when there are " "(<space>) characters in the path to the `./images/flowchartwiki` directory.
- added $wgDbPrefix to hash for filenames in ./images/flowchartwiki to avoid clashes in multi-instance installations
- added "concentrate=true" to .dot files to combine arrows to/from boxes into one arrow.
- experimental support of Windows-IIS platform.

3.3.4 FlowchartWiki 1.1.0-RC6

- updated cacheing of images, added "?Timestamp=<timestamp>" to image-tag of graph images in order to remove the requirement to press "shift-reload" in the browser to see the updated image.
- Database table access now uses Mediawiki DB-Prefix. (Thanks to J.R.M.)
- Updated special page CheckFchw to show a graph created by graphviz for testing graphviz.

3.3.5 FlowchartWiki 1.1.0-RC5

- Fixed call to GraphViz.exe on Windows installations (fixed problems on W2K and W2K3 servers), Unix installs are not affected
- Added optional `PageName::DifferentNameForPage` Tag to override the labeling of the box in the Graph (Defaults to Name of WikiPage)
- BugFix: Quote in PageName
- BugFix: "_" are replaced with " " in PageName

3.3.6 FlowchartWiki 1.1.0-RC1

- Added special page Special:CheckFchw to check correct installation Special:CheckFchw
- Fixed bug: Redirected pages causes timeout in some cases
- New customization format allows to set color, label, shape on arrows in graph

3.4 Release and upgrade Notes - Current Release PdfBook

3.4.1 PdfBook 1.1.1

- fixed bug with ' (Apostroph) and " " (Blank) in Category names.

Upgrading from previous versions to pdfbook-1.1.1:

- Replace the existing files with the files of the new version. (You may want to backup your existing installation first.)

3.4.2 PdfBook 1.1.0

- updated Special Pages:Check_Fchw
- changed calls to htmldoc to platform-dependant calls for Unix/Linux, Windows-Apache and Windows-IIS platforms
- changed unique filename to pdf-book-<uniqueID>
- deletes temporary files after delivery to user, so `./images/pdfbook` directory should be mostly empty.
- fixed calls to htmldoc when there are " "(<space>) characters in the path to the `./images/flowchartwiki` directory.

3.4.3 PdfBook 1.1.0-RC3

- minor update to special page Special:CheckPdfBook

3.4.4 PdfBook 1.1.0-RC1

- Added special page Special:CheckPdfBook to check correct installation Special:CheckPdfBook
- PdfBook now should also work when the FlowchartWiki extension is not installed or used in a category.

3.5 Release 1.0.x

3.5.1 FlowchartWiki 1.0.1

- Fixed bug: Broken graph if name of pages contains " " space

3.5.2 FlowchartWiki 1.0.0-RC3

- Added simple cache for graphs
- Replacement for Hash function (if not exists - on some Solaris hosts)

3.5.3 PdfBook 1.0.0-RC3

- Sorted book by Level
- Added support for codepage ISO-8859-2

3.5.4 FlowchartWiki 1.0.0-RC2

- Fixed for Windows hosts (flowchartwiki and pdfbook too)
- Fixed duplicate entries in dependencies

3.5.5 FlowchartWiki 1.0.0-RC1

- First Release of FlowchartWiki. Please send feedback to thomas dot kock at gmx dot de.

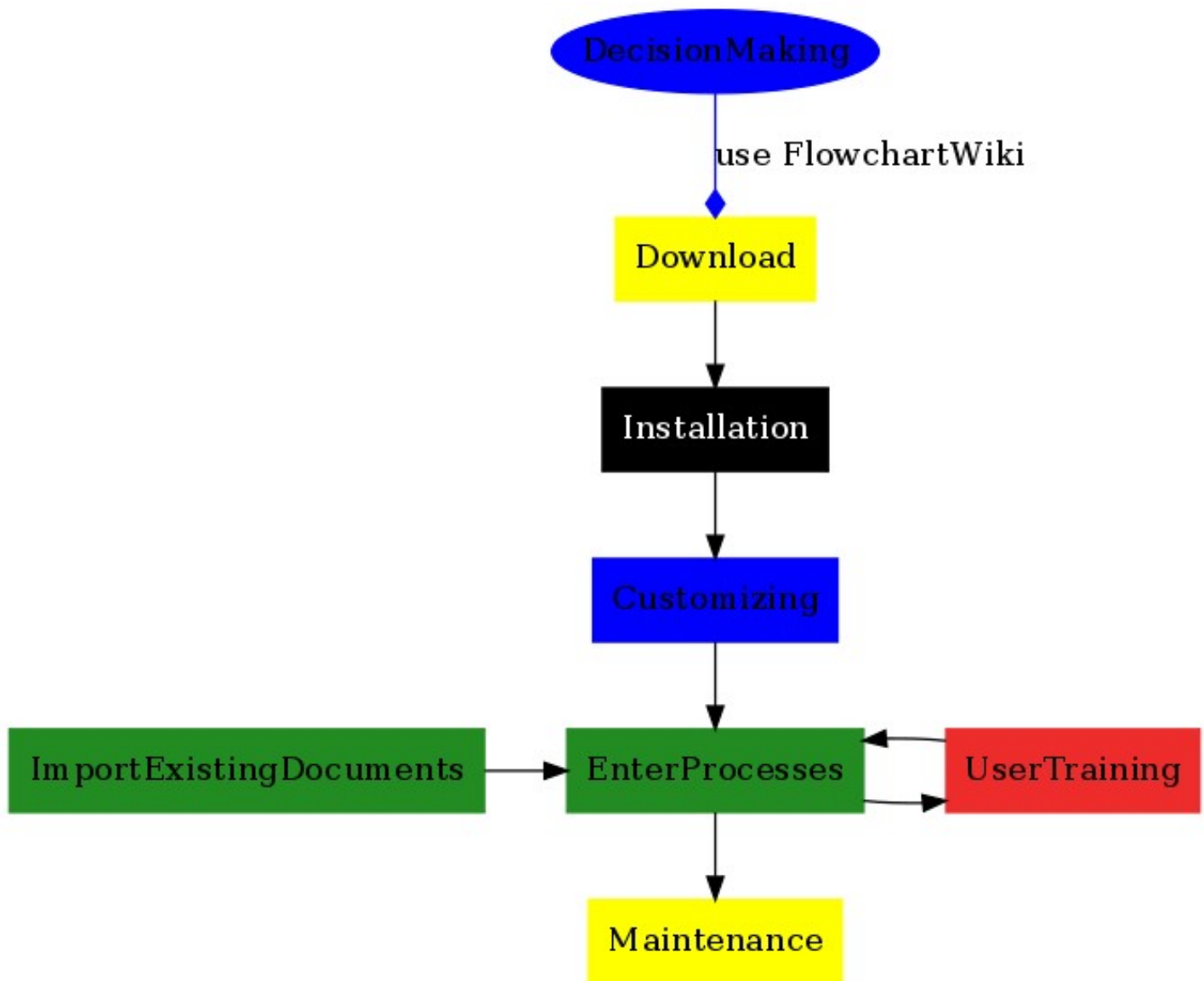**Type of page 'Download':** Rect_Yellow
Where do I link to: **Who links here:**
 Installation (NextStep)   DecisionMaking (Install)
NextStep::Installation Type::Rect_Yellow Level::0995

4 Installation

4.1 System requirements

- Mediawiki system requirements as listed here [[1]] with PostgreSQL or mySQL database.
- Parallel usage of Semantik MediaWiki with FlowchartWiki is not supported. (Contributions welcome.)

4.2 Overview of tested and supported platforms

On these platforms we have installed and tested MediaWiki together with FlowchartWiki. If you use it on other platforms, please let us know so we can update this table.

OS / Database	Windows-Apache	Windows-IIS	Linux
Postgresql	untested	untested	tested OK
mySQL	tested OK	experimental, see notes	tested OK

4.3 Preparation (Windows)

- Install Apache, PHP and mySQL or Postgresql Database.

- ◆ WAMP (Windows - Apache - mySQL) is pretty easy to set up.
- ◆ You could use one of the prepacked installers like WinLamp
- ◆ Please see the documentation of these installers for more details

Possible issues on Windows:

- Depending on your environment, GraphViz and HtmlDoc may need addional .dll's: vcredist_x86.exe, msvcr71.dll, libssl (libeay32.dll, ssleay32.dll)
- Calls from PHP to external programs may be restricted by the OS (This is primarily an issue with IIS, see below.)
- On Windows (XP?) with Apache there seem to be problems in calling the htmldoc and graphviz executables when the the DocumentRoot in httpd.conf contains " " (Blanks/Spaces). Please move the DocumentRoot to a directory path without spaces - like "c:\htdocs" and update DocumentRoot accordingly - to "c:/htdocs". (until 1.1.0-RC6, fixed in 1.1.0)
- Please let us know about your success / problems on Windows platforms.

Windows - IIS: (experimental, you have been warned!)

- Be prepared to do some serious tweaking with rights for IUSR and IIS_WPG. This MAY include rights to `cmd.exe`, the `c:/program files/graphviz` and `c:/program files/htmldoc` directories, the `./images/flowchartwiki` and `./images/pdfbook` directories - and possibly more.
- Granting enough rights, we were successfully running flowchartwiki/graphviz on W2K3 with IIS.
- We quit digging deeper into making htmldoc work due to these hints: """Enable auditing of 'object failures' in the local security policy, see if anything is in Security event log. Another couple of tools to use are Regmon and Filemon from Sysinternals. Auditing, Regmon, Filemon can narrow down if its a obvious permissions issue. If not, it's something in your local security policy preventing it.""" (Source: [2])
- also tweaking with the calls to the executables in `checkfchw_body.php`, `checkpdfbook_body.php` first and then applying those changes to `graphviz.php` and `pdfbook.php` may be necessary.
- IIS seems to handle URLs differently. Please check Manual:Short_URL and URL_rewrite_in_IIS and adjust your `LocalSettings.php` accordingly. What worked for us:

```
$wgScript           = "/mediawiki/index.php";
$wgArticlePath      = "$wgScript/$1";
$wgUsePathInfo      = true;
```

- The Special:CheckFchw page seems to have timing issues. It is testing for the existence of the generated .png file. The file may be there, but windows seems to be slow to recognize this. Please crosscheck the `./images/flowchartwiki` directory for a `FchwTest.png` file. This problem may also re-appear in graphviz.php when creating images during normal usage. The program already waits, but maybe windows is even slower...
- **Contributions with detailed instructions and fixes are very much appreciated.**

4.4 Preparation (Unix/Linux)

- usually Apache, PHP and mySQL or Postgresql are installed in the base installation. If not, please add these packages now.
- for running php-scripts outside of apache, you may need to add a package named php5-cli.
- For Solaris we recommend using Coolstack
- For Solaris: Graphviz needs gsfonts installed

```
/opt/csw/bin/pkg-get install gsfonts
```

Possible Issues on Linux:

- Graphviz: (Was reported for CentOS 64bit): apache-user needs access to "`/usr/lib64/graphviz/config6`" which can be set by granting "`chmod 755`" on that file. The Error Message in Apache-Log was "`Format: "png" not recognized. Use one of:`" (Thanks to Gero from Boeblingen, Germany)
- Graphviz: (Was reported for RHEL5): requires GD-support in Graphviz which is included in the package "graphviz-**gd**-2.12-8.el5" in addition to installing "graphviz-2.12-8.el5". (Thanks to Mangesh)
    - ◆ Test with '/usr/bin/dot -v' and check if the 'render' line contains 'png'. Press Ctrl-c to exit dot.

4.5 Installation steps

MediaWiki Installation

- download [[3]]
- install [[4]] Mediawiki as described in Mediawiki documentation
- Test your Mediawiki installation

Tip:

- If you want to run multiple (separate) wikis, use parallel installations of mediawiki. These installations can share the same database with different $dbprefix settings. FlowchartWiki is not compatible with running multiple (separate) wikis in a single wiki-installation like "Method One" documented on SteveRumberg.com, which switches $dbprefix based on a url-parameter.

GraphViz and Flowchartwiki Extension Installation

- Download and install Graphviz to your system [[5]] (The MediaWiki GraphViz Extension is not required.)
- Download and extract Flowchartwiki package to mediawiki/extensions
- rename the directory from flowchartwiki-x.y.z to flowchartwiki

Database configuration

- if you use a Database Prefix ($wgDbPrefix), please adjust the database scripts accordingly. (see notes in the .sql scripts).
- Postgres
    - import ./extensions/flowchartwiki/maintenance/schema_pg.sql to your Postgres database

```
psql db wikiname < schema_pg.sql
```

- mySQL
    - import ./extensions/flowchartwiki/maintenance/schema_mysql.sql to your mySQL Database

```
mysql --user=<dbuser> --password=<password> <dbName> < ./extensions/flowchartwiki/maintenance/schema_mysql.sql
```

Directory setup and Wiki Configuration

- create directory *flowchartwiki* in *images* subdirectory a set permissions on *flowchartwiki* directory, (Please note: The name of this directory was changed from `fchw` to `flowchartwiki` in version 1.1.0.)

```
chmod 777 ./images/flowchartwiki
```

- add these lines to mediawiki/LocalSettings.php
    - note: Set `<path-to-graphviz-Dot>` to the path on your system:
        ◊ Unix/Linux: could be "`/usr/bin/dot`"
        ◊ Windows: could be "`C:\\Program Files\\Graphviz 2.20\\bin\\dot.exe`"" (use double \\)

```
# Disable cache - otherwise graphs are not updated properly
$wgCachePages    = false;
$wgCacheEpoch = max( $wgCacheEpoch, gmdate( 'YmdHis' ) );
# Include libraries
require_once("$IP/extensions/flowchartwiki/flowchartwiki.php");
$fchw['GraphvizDot'] = "<path-to-graphviz-Dot>";
```

4.6 Optional installation

4.6.1 PDF Book - Export your Process Documentation as a .pdf Document

- Download HTMLDoc: Open Source: [6] Commercial: [7]
    - Unix/Linux: Prebuild packages should be available for your distribution.
    - Windows: Please use Google to locate a compiled open-source package or purchase a commercial license from the vendor.
- Install HTMLDoc
- Download the FlowchartWiki PdfBook Extension (See Download) (This Extension has been modified to work with FlowchartWiki).
- extract PdfBook mediawiki extension to ./extensions directory
- rename directory pdfbook-x.y.z to pdfbook
- create directory *pdfbook* in *images* subdirectory and set permissions on *pdfbook* directory,

```
chmod 777 ./images/pdfbook
```

- add these lines to mediawiki/LocalSettings.php and set the correct path to the htmldoc executable

```
require_once("$IP/extensions/pdfbook/pdfbook.php");
$PdfBookShowTab = true;
$PdfBookHtmlDoc = "c:\\program files\\htmldoc\\htmldoc.exe";
```

4.7 Additional settings

If you want to allow to view pages only for registered users, add these lines to mediawiki/LocalSettings.php

```
# Allow only authorized persons
$wgGroupPermissions['*']['read'] = false;
$wgGroupPermissions['*']['createaccount'] = false;
$wgShowExceptionDetails = true;
```

4.8 Parallel usage with Semantic MediaWiki

The standard installation of FlowchartWiki will NOT work together with Semantic MediaWiki, as the style used for tagging the links is the same as in SMW.

Daniel L. submitted a workaround that uses a different tagging. He is using "--" for FlowchartWiki instead of "::" (which is used by SMW). If you want to use FlowchartWiki in parallel with SMW, you may want to give his modification a try.

Linktypes.php: approx. line 58:

```
< if (strpos($LinkText, "--") > 0) {
< $Relation = substr($LinkText, 1, strpos($LinkText, "--")-1);
< $LinkTmp = substr($LinkText, strpos($LinkText, "--")+2, -1);
< $output .= "[$LinkTmp|$Relation--$LinkTmp]";
---
> if (strpos($LinkText, "::") > 0) {
> $Relation = substr($LinkText, 1, strpos($LinkText, "::")-1);
> $LinkTmp = substr($LinkText, strpos($LinkText, "::")+2, -1);
> $output .= "[$LinkTmp|$Relation::$LinkTmp]";
```

Linktypes.php approx. line 164:

```
< if (strpos($Link, "--") > 0) {
< $Relation = substr($Link, 0, strpos($Link, "--"));
< $To_title = substr($Link, strpos($Link, "--")+2);
---
> if (strpos($Link, "::") > 0) {
> $Relation = substr($Link, 0, strpos($Link, "::"));
> $To_title = substr($Link, strpos($Link, "::")+2);
```

4.9 Testing your installation

Special Pages
Two "special pages" will help you to test and troubleshoot your installation. (New in Version 1.1.0)

- Check FlowChartWiki extension (Special:CheckFchw)
- Check PDFBOOK extension (Special:CheckPdfBook)

These pages check, if everything is installed correctly, including executables, paths, permissions and will try to create a graph or .pdf document.

Set up a "Test Process"
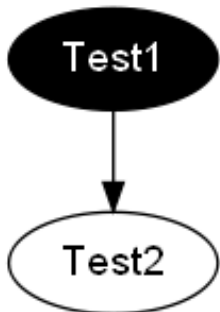
(After entering the process, this is the diagram you should see on the WikiPage Test1.)

- **Step1:** Create category `Category:Test` with content (Cut & Paste)

```
<CategoryBrowser />
[[ModelType::Draw]]
```

- **Step 2:** Create Page `Test1` with content (Cut & Paste)

```
<CategoryBrowser />
[[LinksTo::Test2]]
<Dependencies />
[[Type::Rect_Red]]
[[Level::1000]]
[[Category:Test]]
```

- **Step 3:** Create Page `Test2` with content (Cut & Paste)

```
<CategoryBrowser />
<Dependencies />
[[Type::Rect_Blue]]
[[Level::1010]]
[[Category:Test]]
```

- **Step 4:** Tests:
  - ♦ Open the Category Page `Category:Test`
    You should see a diagram with two process steps (Test1 and Test2) which is clickable and will link to the Pages Test1 and Test2.

The Diagrams will not be colored or using shapes because the customizing for the colors and shapes is not yet available, so default values are used. (If you want it pretty, see the Customizing documentation).

♦ Click on the diagram on the Pages `Category:Test` and `Test1, Test2` to test the navigation.
♦ you may want to have a look into `./images/flowchartwiki` where the files for the images are cached.
♦ Testing PdfBook export: Open the category page and click the "Export to PDF" tab. You should get an automatic download of a .pdf File. The .pdf Document contains all pages from the category, graphs included.

Next:

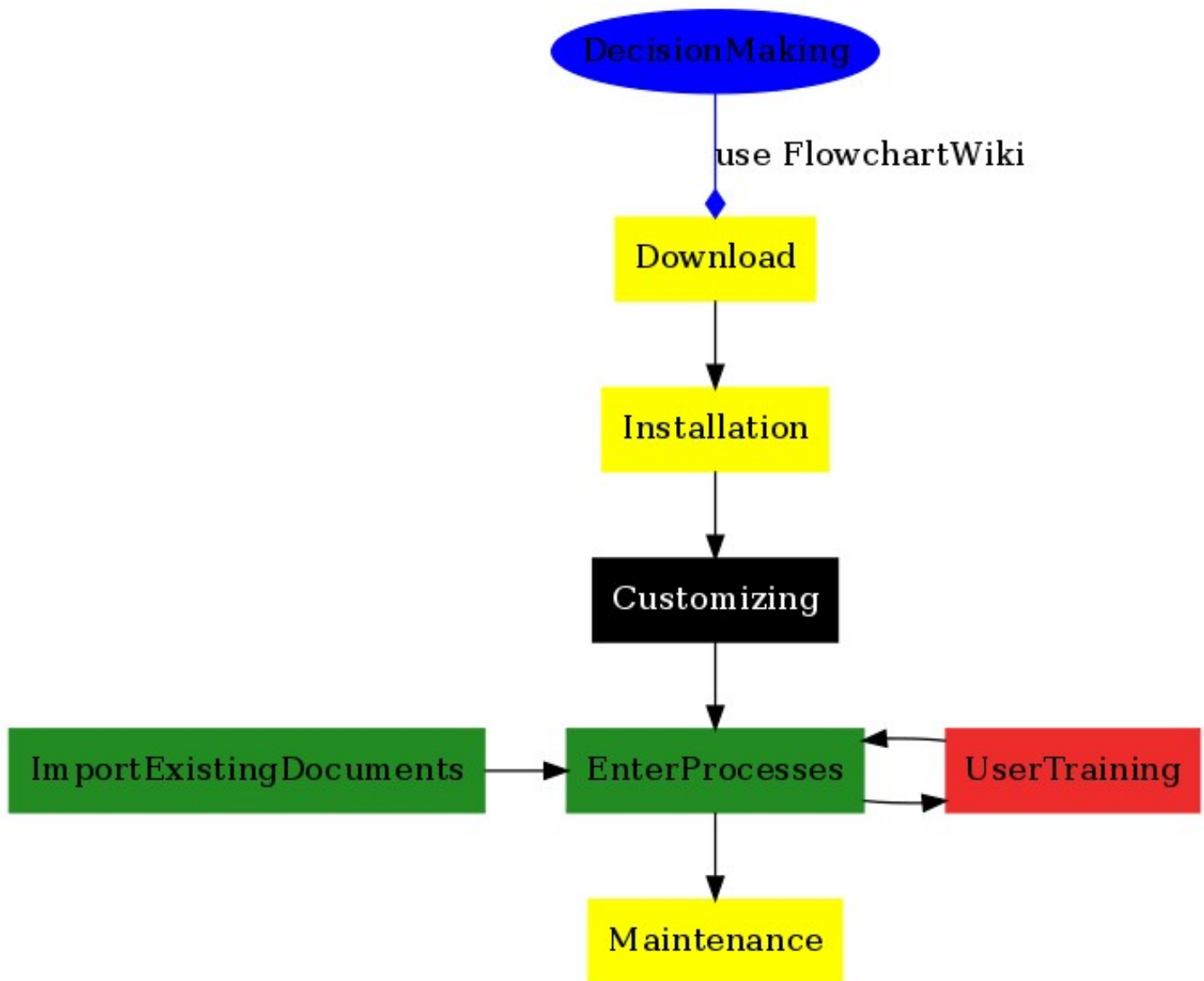- NextStep::Customizing

Type of page 'Installation': Rect_Yellow

Where do I link to:	**Who links here:**
Customizing (NextStep)	Download (NextStep)
Type::Rect_Yellow Level::1000	

5 Customizing

5.1 Customizing your Installation

Besides standard options to modify and customize your MediaWiki installation, FlowchartWiki has some additional customizing options.

Customizing is used to define the shapes and colors used in the diagrams created by FlowchartWiki. You can create customizing settings for different types of processes (EPK/Aris, "plain vanilla", ...) and use them in your process diagrams. Each Process needs to have one process-type assigned.

The standard distribution of FlowchartWiki includes a set of standard customizing pages that you can import (described below) and modify/adapt to your needs. The detailed documentation on how to set up the customizing is given below as well.

5.2 Import default customizing

5.2.1 Installation on a remote system

- Importing the customizing requires an upload of prepared wikipages from your local system to the remote system.
- The pages which will be uploaded are part of the flowchartwiki distribution and are located in the flowchartwiki/maintenance directory.
- Download flowchartwiki to your local system or download the flowchartwiki/maintenance directory from your server to your local system.

5.2.2 Installation on a local system

- locate the flowchartwiki/maintenance directory in your filesystem (it could be in ./apache2/htdocs/wiki/extensions/flowchartwiki/maintenance)

5.2.3 Importing the default Customizing

- In mediawiki choose Special pages/Import pages.
- Import the file import_customizing.xml (For sample contents see Customizing:Configure_Chart)

5.2.4 Optional imports

- Import the file import_ShapeTest.xml (For a Demo on this site see Category:ShapeTest)
- Import the file import_FlightBooking.xml (For a Demo on this site see Category:Flightbooking)

5.3 Customizing:Configure_Chart Page(s)

See Customizing:Configure_Chart for a Sample.

5.3.1 ModelType determines the Chart Type

Each Category Page that you use for documenting a Proces needs to contain one `ModelType` Tag like the one shown below.

```
[[ModelType::EPK]]
```

`EPK` defines that for this process the colors and shape types will be based on the EPK diagram definition in Customizing.

5.3.2 Configure the ModelType / Graphics

For each ModelType a customizing is required to define the shapes and colors used. To define the customizing of a ModelType you have two options:

- Use an individual Wiki Page (`Customizing:Configure_EPK`) to define the customizing for this ModelType.
- Use the Generic `Customizing:Configure_Chart` Wiki Page to define the customizing.

The first option has the advantage that this page can easily be exported and moved to other wikis or be distributed with your process documentation.

The lookup for Customizing is made in the order:

1. `Customizing:Configure_[ModelType]`
2. `Customizing:Configure_Chart`

If no customizing for the specified Modeltype is found, a standard round shape and Black&White color is used. (You may have seen this when doing your first install with the "Test" Process. After you have imported the Customizing, the Diagrams looked different.)

5.3.3 Available Shapes and Colors

Node-Shapes:
Please see Category:ShapeTest for a preview of all shapes supported by this GraphViz installation. As you can see from the Graph, not all shape types are supported currently. If you would like to test it or have it as a reference on your local installation, import the file `import_ShapeTest.xml` from the maintenance directory of your flowchartwiki distribution.

Colors:
Please see [1] for the list of available Colors.

Arrows:

- Please see [2] for the list of available arrows.
- Colors: as above
- Line-Types: solid, dashed (- - -), dotted (.....)
- Label: free text

5.4 Sample Customizing: Page

Sample `Configure_Chart` Page:

```
Some Text with Warning why this page should not be edited.
== Configuration ==
*Sample_Configure_ChartType
**Nodes
***PageType Shape Color_of_Shape [Color_of_Font Defaults to Black]
***PageType Shape Color_of_Shape Color_of_Font
```

The Configure_Chart Page is divided into two Section.

- The "Warning" Section
- The "Configuration" Section.

5.4.1 The Warning Section

The Warning Section is the initial Text in the page and is not parsed. It ends at `== Configuration ==`

5.4.2 The Configuration Section

The Configuration Section starts with `== Configuration ==` It contains the definition for one or more chart types.

Differences between configuration formats
1.2:

```
*Configure_<ChartType_1>
**Nodes
***[Category <Shape> <Color_of_Shape>]
***<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
**Arrows
***<LinkType> <Arrow-Shape> <Color_of_Arrow> [<Type_of_line>] [<Arrow_label>]
```

1.1:

```
*Configure_<ChartType_1>
**Nodes
***<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
**Arrows
***<LinkType> <Arrow-Shape> <Color_of_Arrow> [<Type_of_line>] [<Arrow_label>]
```

1.0:

```
*Configure_<ChartType_1>
**<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
```

Version 1.2 Configuration Format
The definition of a configuration is:

```
*Configure_<ChartType_1>
**Nodes
***[Category <Shape> <Color_of_Shape>]
***<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
*Configure_<ChartType_2>
**Nodes
***<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
**Arrows
***<LinkType> <Arrow-Shape> <Color_of_Arrow> [<Type_of_line>] [<Arrow_label>]
```

Sample:

```
*Configure_EPK
**Nodes
***Category    box           red
***Event       hexagon       azure3
***Decision    diamond       azure3
***Function    parallelogram azure3
***DataSource  rect          khaki1
***Person      box           chartreuse1
***Department  ellipse       chartreuse1
***Product     rect          yellow
**Arrows
***Yes         box           green         dashed      Yes
***No          diamond       #ffa0a0       solid       No
```

- First line: `*Configure_<ChartType>`
  - Only Groups starting with `*Configure` are parsed. If you want to disable a configuration, renaming it to something different (like `Disabled_Configure` will disable this configuration block.
  - `ChartType` defines the Type of the Chart e.g. "EPK" or "Flat". This is the type of the Diagram that is being used.
- Second line: `**Nodes`
  - starts the section where the PageTypes are defined.
- Definition of PageTypes on the next lines:
  - **New in 1.2.:** Optional: `Category`: When this is included, it will show a clickable category-name on top of the chart.
  - `PageType` defines the Pagetypes being used in this Diagram, for Example "Person", "Function", "Event", "Decision" etc.
  - `Shape` defines the Shape that is being used. The Definition of the shape is taken from the GraphViz Documentation at [[3]]. Please be aware that not all shapes are supported/working.
  - `Color_of_Shape` defines the color that is used to render the shape. The name of the colors and a color table can be seen at [[4]]
  - `Color_of_Font` defines that color of the font that is used for writing the name of the process-step into the shape. If not given, it defaults to black. The same table of colors applies as before.
- `**Arrows`
  - starts the definition block for the arrows.
  - If the Arrow is defined in the customizing, then the properties in the definition are applied. If it is not defined, it defaults to "normal" shape, "black" color, "solid" line and no label.
- Definition of the Arrows on the next lines:
  - `<LinkType>` This is the type of the link that is used in the Wikipage. Example links to wikipage `PageName` with the LinkType `Yes`.

    ```
    [[Yes::PageName]]
    ```
  - `<Arrow-Shape>` defines the "endpoint" of the arrow. See [[5]] for details. Defaults to "normal".
  - `<Color_of_Arrow>` the same definition as in the colors of shapes applies, defaults to black.
  - `[<Type_of_line>]`: solid (default), dashed (----), dotted (....)
  - `[<Arrow_label>]`: The label that is attached to the line in the graph. Currently only one word is supported (no blanks).

Version 1.0.x Configuration Format
The definition of a configuration is:

```
*Configure_<ChartType_1>
**<PageType> <Shape> <Color_of_Shape> [<Color_of_Font>, Defaults to Black]
*Configure_<ChartType_2>
**<PageType> <Shape> <Color_of_Shape> [<Color_of_Font> Defaults to Black]
```

Sample:

```
*Configure_EPK
**Event       hexagon         azure3
**Decision    diamond         azure3
**Function    parallelogram   azure3
**DataSource  rect            khaki1
**Person      box             chartreuse1
**Department  ellipse         chartreuse1
**Product     rect            yellow
```

5.5 Examples

- Customizing:Configure_Chart for General Customizing
- Customizing:Configure_EPK for EPK / Aris
- Customizing:Configure_ShapeTest for all the Shapes in the documentation of GraphViz

Active configuration Block:

```
*Configure_EPK
**Nodes
***Event       hexagon         azure3
***Decision    diamond         azure3
***Function    parallelogram   azure3
***DataSource  rect            khaki1
***Person      box             chartreuse1
***Department  ellipse         chartreuse1
***Product     rect            yellow
```

Disabled Configuration Block:

```
*Disabled_Configure_EPK
**Nodes
***Event       hexagon         azure3
***Decision    diamond         azure3
***Function    parallelogram   azure3
***DataSource  rect            khaki1
***Person      box             chartreuse1
***Department  ellipse         chartreuse1
```

```
***Product     rect          yellow
```

- NextStep::EnterProcesses

Type of page 'Customizing': Rect_Blue
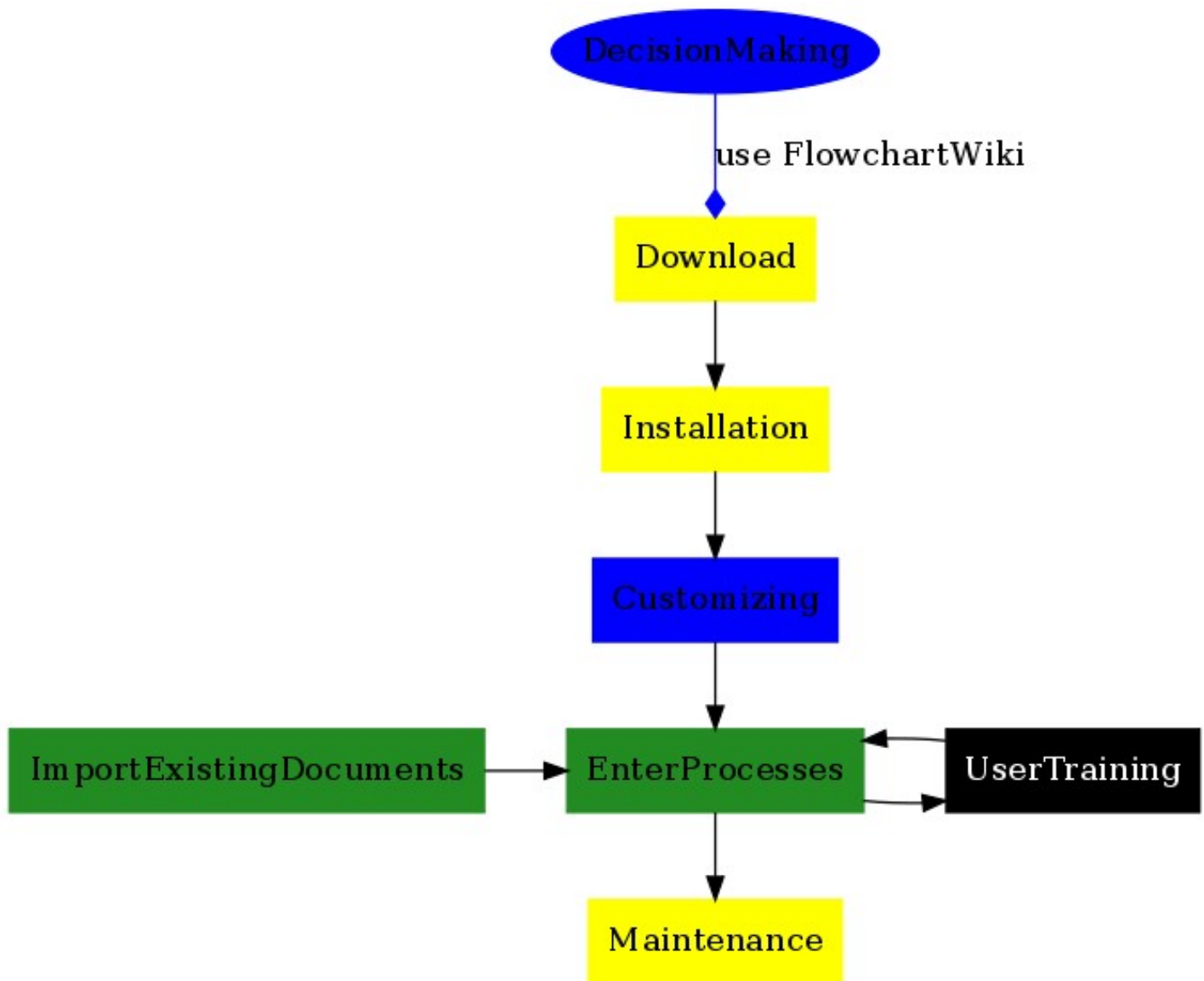**Where do I link to:**
  EnterProcesses (NextStep)
Type::Rect_Blue Level::1010

Who links here:
  Installation (NextStep)

6 UserTraining

ToDo: Document the small training effort required to get users hooked to FlowChartWiki.

- NextStep::EnterProcesses

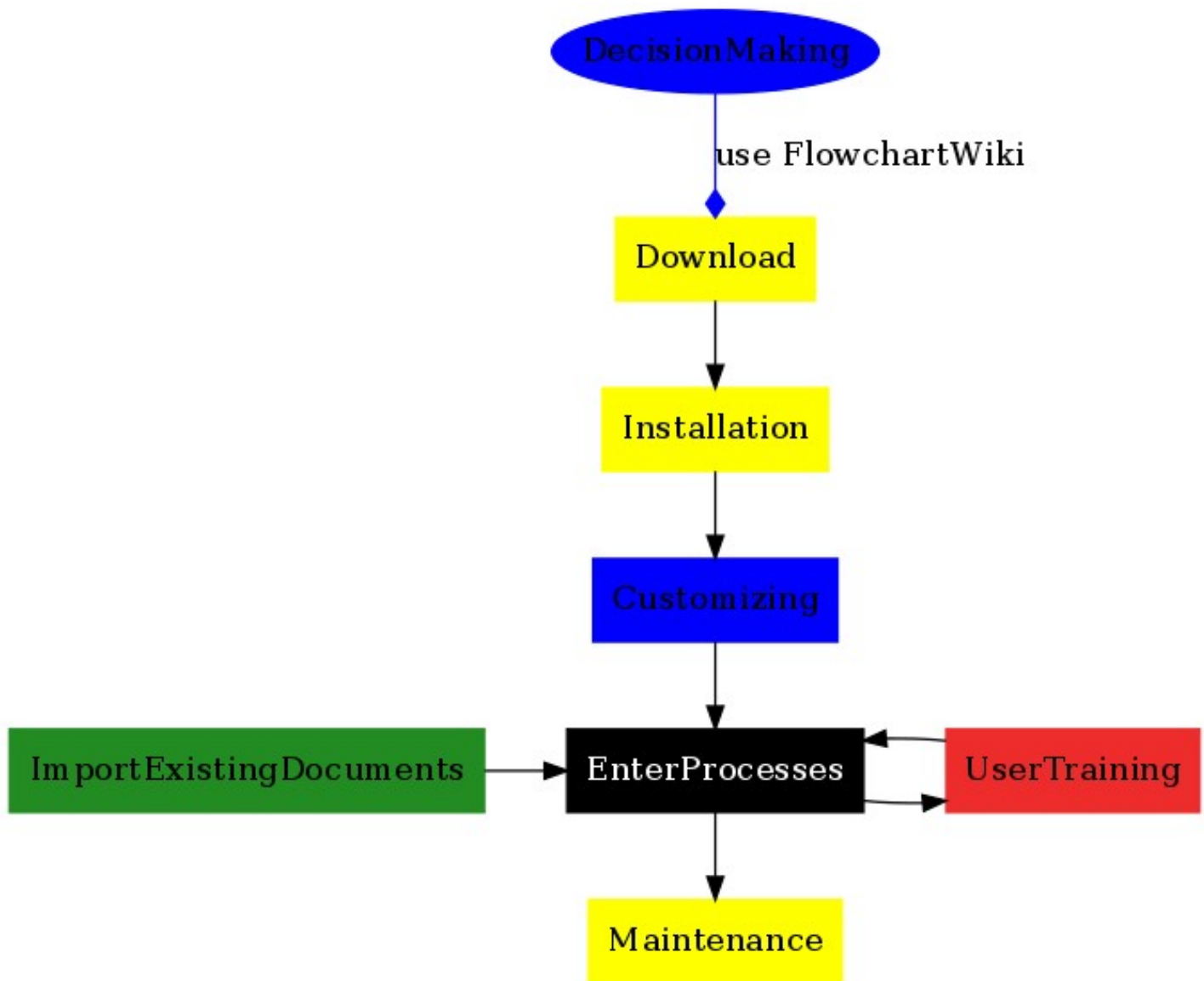**Type of page 'UserTraining':** Rect_Red
Where do I link to:
 EnterProcesses (NextStep)
Type::Rect_Red Level::1030

Who links here:
 EnterProcesses (NextStep)

7 EnterProcesses

7.1 How to create a process

Sample Category Page Content:

```
<CategoryBrowser />

Description of the process...

configure the type of the graphics
  '[ModelType::EPK]
  '[ModelType::Flat]
[[ModelType::EPK]]
```

- CategoryBrowser - Tag displays graph of current category. Items are process steps of this category.
- ModelType::EPK - Specifies graphics style for graph, definition from this page is used for all process step graphs.

7.2 How to create a process step

Sample page content:

```
<CategoryBrowser />
```

```
Description of the process step...

[[UsedBy::CheckBookingRequest]]

<Dependencies />

[[Type::Product]]
[[Level::1010]]
[[PageName::DifferentName]]
[[Category:Flightbooking]]
```

- `CategoryBrowser` tag displays graph of current category. Items are process steps of this category. Note: that you can use CategoryBrowser2 tag displays 2 graphs - left shows whole process of current step, and right only neighbours.
- `Dependencies` tag shows table with dependencies between process steps
- `Type::Product` - Type for current page (displays specified shape/color in graph)
- `Level::1010` - Steps with same level are in same line in graph. Increase the Level by 10 for the next line in the graph - to Level::1020, Level::1030 etc. (If you have done some BASIC Programming "in the good old days" - this is like line-numbers in Basic. - And yes, we are working on a renumbering function ;-))
- `PageName::DifferentName` (optional) - this overrides the usage of the name of the WikiPage in the Graph with "DifferentName". i.e. If your WikiPage is named "SomePage" - it will default to the box labeled "SomePage" in the Graph. If you use "PageName::Some_Other_Name_For_This_Page" it will be labeled "Some Other Name For This Page" in the Graph. ("_" are replaced with " "). To split the text into multiple lines, insert "\n" where a linebreak should be placed. Example: "PageName::Some_Other_Name\nFor_This_Page" would show the label in two lines: 1:"Some Other Name", 2: "For This Page".
- `Category:Flightbooking` - This is important. Describes participation in process.

7.3 Detail documentation

7.3.1 Tag CategoryBrowser

```
<CategoryBrowser />
```

Displays 1 graph.

```
<CategoryBrowser2 />
```

Displays 2 graphs. Left shows whole process and right only neighbours. CategoryBrowser2 on category page show only 1 graph.

```
<CategoryBrowser>Name_Of_Category</CategoryBrowser>
<CategoryBrowser2>Name_Of_Category</CategoryBrowser2>
```

Displays graph(s) for selected category (graph of another category process).

7.3.2 Tag Dependencies

```
<Dependencies />
```

Shows table with dependencies including type of links.

Type of page 'EnterProcesses': Rect_Green

Where do I link to:	**Who links here:**
UserTraining (NextStep)	UserTraining (NextStep)
Maintenance (NextStep)	ImportExistingDocuments (NextStep)
	Customizing (NextStep)

7.4 Tips & Tricks

FCKEditor:

If you use the FCKEditor with FlowchartWiki, you need to put `__NORICHEDITOR__` at the start of each page.
Otherwise FCKEditor may mangle the Tags like this: (Thanks to Andrew from New Zealand.)

```
  [[Type::Rect_Red]]                    # original Tag
  [[Rect Red|Type::Rect_Red]]           # 1st round
  [[Rect Red|Rect Red|Type::Rect_Red]]  # 2nd round
```
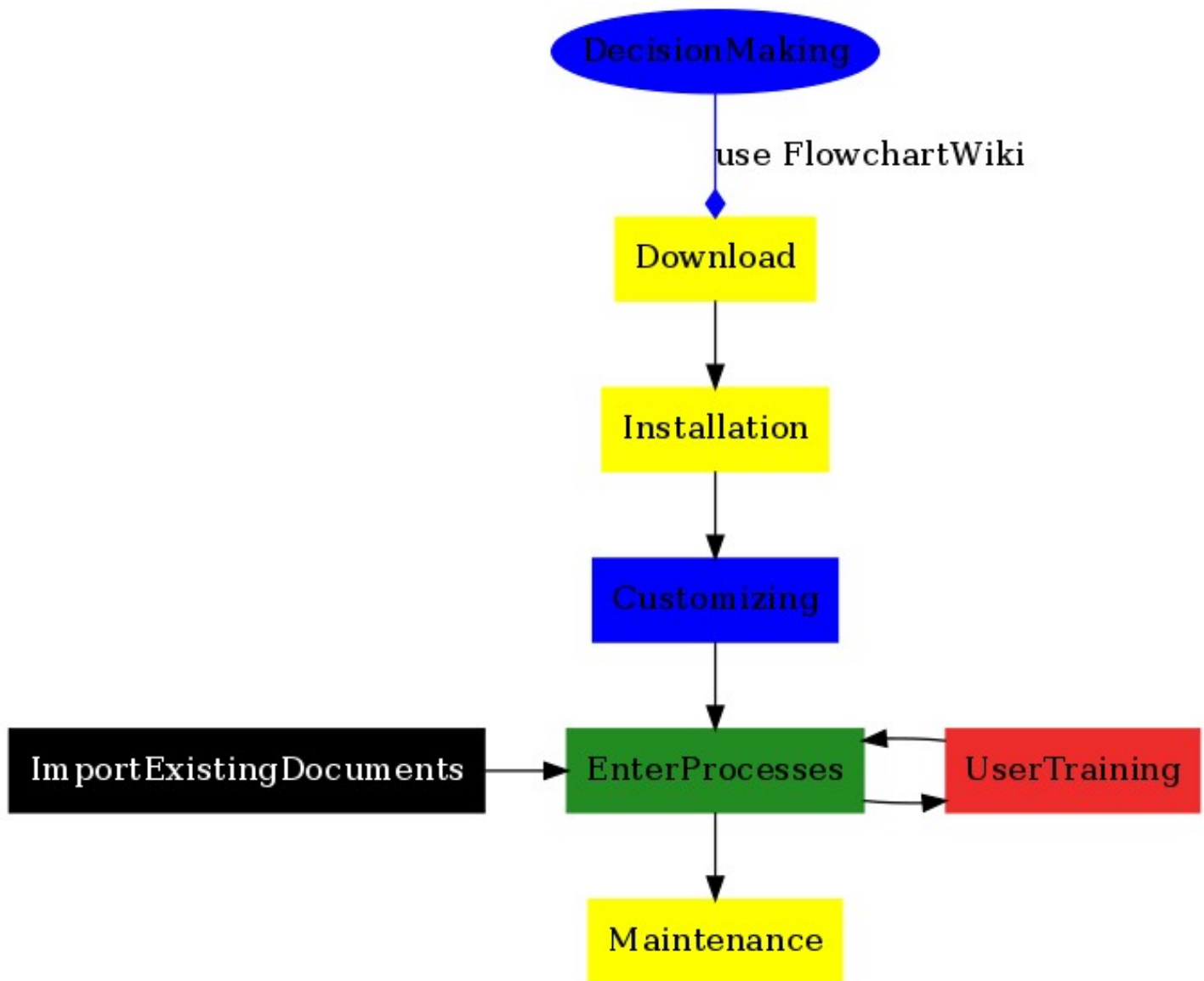
Positioning of the Tags:
You may want to place all the FlowchartWiki related tags (like "NextStep::Maintenance", "Type::Rect_Green", "Level::1030" etc.) to the end of the page, if your users are disturbed by the tags showing up in the middle of the text.

- NextStep::Maintenance

- NextStep::UserTraining

Type::Rect_Green Level::1030

8 ImportExistingDocuments

8.1 Easy conversion of .doc Files with OpenOffice.org

The writer module of OpenOffice.org can open .doc documents and has a special export filter to export the document in MediaWiki markup. (=MediaWiki Formatting).

Detailed List of Features: [[1]]

8.2 Procedure

1. Download [OpenOffice.org]
2. Install the downloaded application
3. open the .doc file in the OpenOffice.org Writer module
4. Export the Document in MediaWiki markup as a .txt file
    1. Choose File -> Export
    2. Select File Format "MediaWiki (*.txt)" and enter a FileName.
5. Open the .txt File in Notepad or any other Text-Editor, mark the parts to be transferred or the whole document and copy it into the clipboard
6. Create or open the page in your Wiki and go to the Edit-Mode of the page
7. Paste the content from the clipboard into the Edit-Window of the Wikipage
8. Save the Wikipage

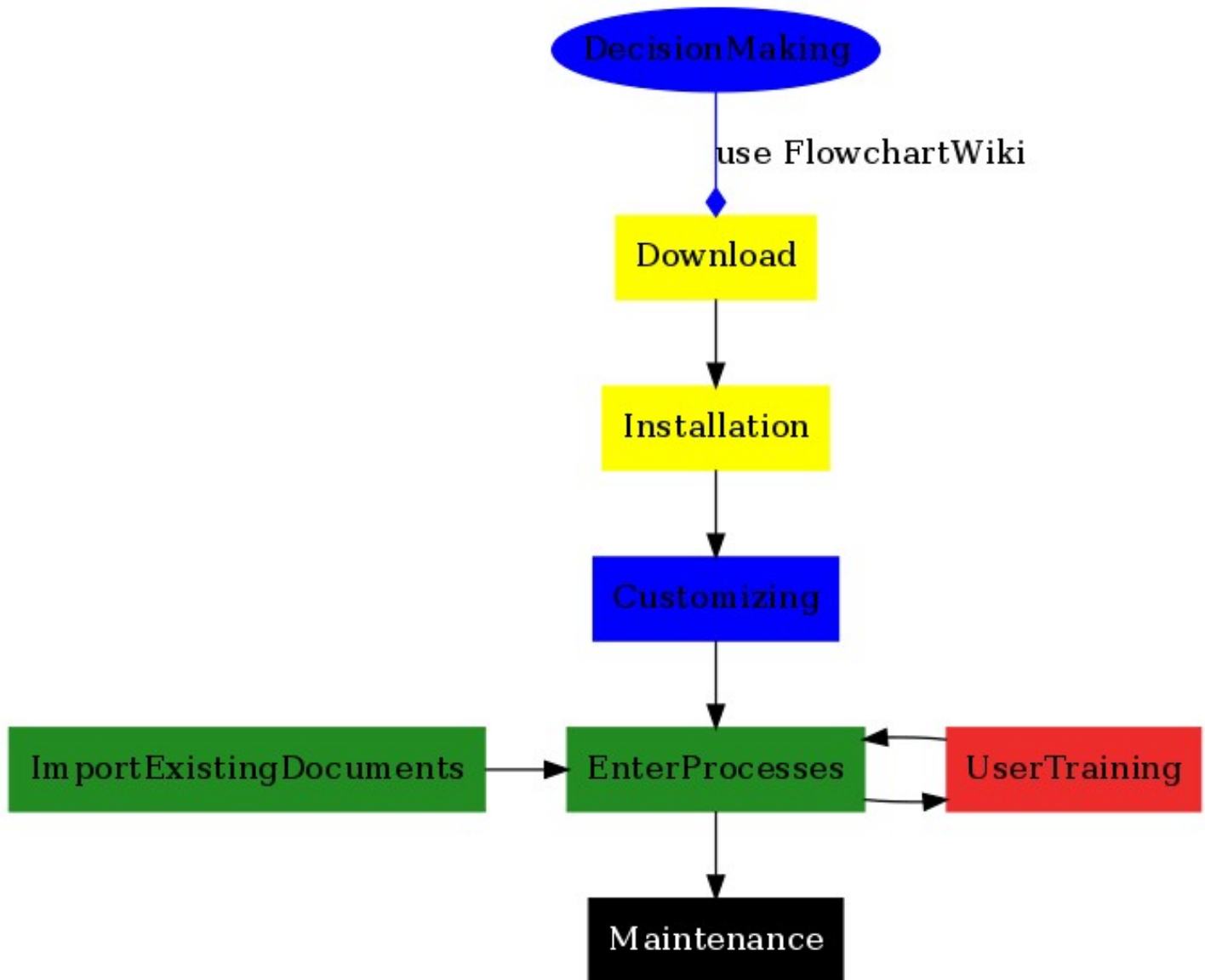**Type of page 'ImportExistingDocuments':** Rect_Green

Where do I link to:
EnterProcesses (NextStep)

- NextStep::EnterProcesses

Type::Rect_Green Level::1030

9 Maintenance

9.1 Re-Initializing the Database

If your FlowchartWiki Database is corrupted, you may re-generate the table that contains the links between the pages by running the fchw_RefreshPages.php script from the commandline.
cd to directory htdocs/<yourWiki>

```
php ./extensions/flowchartwiki/maintenance/fchw_RefreshPages.php
```

Windows Note:
PHP could be started by

```
c:\Program Files\Apache2\modules\php\php.exe
```

Unix Note:
You may need to have the PHP-cli package installed on your system.

9.2 Temporary Files

FlowchartWiki and PDFBook create temporary files in these directories:

```
./images/flowchartwiki
./images/pdfbook
```

FlowchartWiki Note:

- The filenames are created from a hash of the database-prefix and the name of the wikipage and are overwritten for each update.
- There are four files per page:
  - ♦ .dot source which was converted to .png and .map by graphviz.
  - ♦ .png with the graph,
  - ♦ .map with an ImageMap for HTML Display
  - ♦ .dot.md5 which contains the hash value of the dot file and is used to validate, if an image needs to be re-created.
- Deleting these files is safe, they will be recreated at the next access, but you will have to delete all 4 files per page.
- Stale files will only exist for pages that have been deleted, so a frequent purging of this directory is not required.

PDFBook Note:

- There are 2 files per .pdf document
  - ♦ .html source (no Extension) which gets converted to .pdf by htmldoc
  - ♦ .pdf with the .pdf document
- The filenames are 'pdf-book-' with a random number.
- These Files are created at each creation of a .pdf Document and are not re-used.
- In Pdfbook 1.1.0 the files are deleted after they have been delivered to the user, so the directory should be mostly empty.
- you may want to configure a cron-job to clean up this directory.

9.3 Moving to new empty server

- Export all pages from old server via page Special pages - Export pages
  - ♦ First see list of categories, and load pages for each one.
  - ♦ Don't forget to add category page like (Category:Test) and MainPage
  - ♦ Don't forget to add customizing pages (Customizing:Configure_Chart, Customizing:Configure_Chart_Documentation, Customizing:Configure_EPK)
- New server - choose Special pages / Import pages and select exported file
- Copy your logo to images/logo.png and this line to Localsettings.php

```
$wgLogo = "/wiki/images/logo.png";
```

- Copy Mediawiki:Sidebar page to new location

Type of page 'Maintenance': Rect_Yellow
Where do I link to:

Who links here:
EnterProcesses (NextStep)

Type::Rect_Yellow Level::1040