Using Templates and Formatting Macros in a Knowledge Base

This page is part of the guide to developing a knowledge base on Confluence Wiki. We have already shown you how to create your knowledge base space. Now we offer an introduction to the templates that Confluence provides.

Quick guide to templates in Confluence
- A template is a page with predefined content that can be used as a prototype when creating other pages.
- Templates are available across the Confluence site (global templates) or per space (space templates).
- Both 'global templates' and 'space templates', as described on this page, define the content of a page. They do not define the content of an entire space.
- You can import predefined templates, including those shipped with Confluence and additional templates from the Atlassian Plugin Exchange.
- To create a template for an entire space, see our guide to creating your documentation space.

Space Information Design

One question that comes up is whether a knowledge base should be implemented as a decision matrix, with a multi-select or other list. We knew that most of our users visit our documentation directly from search engines. Early on, we learned that at least for us, Google Analytics was showing that most traffic was coming from search engines, not from within our site. See Metrics for a discussion on that. So, we knew to focus on optimising page titles and tags rather than focus on a decision matrix.

Confluence's Built-in Feature Set

Page Templates

Confluence's Page Templates are an easy place to start. Begin by choosing how you want your templates to look. Here's how we made our page template:

```
# Symptoms
FILL IN SYMPTOMS HERE

# Cause
FILL IN CAUSE HERE

# Resolution
FILL IN RESOLUTION HERE

{htmlcomment}
ENTER SUPPORT TICKET LINKS
{htmlcomment}
```

The {htmlcomment} macro is part of Adaptavist's Content Formatting Macros, a handy plugin.

You might consider the scaffolding plugin or Form Field Markup for filling out forms. Our support engineers are quite used to using wiki markup (we live and breathe Confluence!), so we left this template as is.

Construct a Home Page: Section, Panels, RSS, and Notifications

Check out the Confluence Knowledge Base Home. The colored content in the middle of the page is an amalgamation of panel, rss and attachment macros. Go to Tools >> Wiki Markup to see.

Wondering how we just made Tools >> Wiki Markup in a different font in the sentence above? Use the {{ }} notation to get the courier font, which you can use for breaking out text. And this would be, of course, a {tip}. Remember, you can visit Tools >> Wiki Markup on this page as well!

The wiki markup there is a little confusing, but it breaks down like this:
1. A **Section** macro, defining the top and bottom of the page
2. **Column** macros, dividing the panels into three
3. **Panel macros**, with hot, hot, hot html colours
4. **Content By Label** Macros
5. An RSS Icon, embedded as an attachment, with a link to the RSS feed. Notice the magic on this one:

   ![rss20.gif](http://confluence.atlassian.com/createrssfeed.action?types=page&sort=created&spaces=CONFKB&labelString=conf32&rssType=rss1&maxResults=50&timeSpan=120&publicFeed=true&title=Confluence+3.2+Knowledge+Base+Articles&showDiff=false)

   *That's a link, with the attachment as the alias, an alignment parameter, and the URL to the RSS feed. Cool, right?*

**Exporting Pages**

These are built-in Confluence features. Check Page Exports. You can learn how to customise the PDF export as well.

**Content Macros**

Besides the ones listed above, some favourite macros for formatting content around the site are the Tip, Info, Warning, and Note Macros, ad the Code macro. They make things look great. We also use Adaptavist’s Content Formatting Macros when we really want to bring our A-game.

Every once in a while we get mileage out of the *(nomarkup)* user macro, when we want to discuss using a macro itself. I suppose that's because Confluence is one of the products we support! It might be useful for other reasons...

**JIRA Issues Macro**

Using JIRA for bug tracking? You can't expect your users to necessarily have exactly the right JQL filters to show the appropriate topics. We have some nice JIRA Issues Macros on display. One of my favourites is in the JIRA KB's Causes for OutOfMemory Errors:

   
   ![jiraissues](http://jira.atlassian.com/sr/jira.issueviews:searchrequest-xml/temp/SearchRequest.xml?jqlQuery=project+%3D+JRA+AND+issuetype+in+%28Bug%2C+Improvement%2C+subTaskIssueTypes%28%29%29+AND+component+%3D+%22Performance+-+Memory+leaks%22+OR+BY+key+DESC&tempMax=200|columns=fixversion;summary;status;key|anonymous=true)

Notice the JQL there including components and sorting, plus the specific columns.

**Labels**

Sometimes, an article needs to be cross listed. For that we label our articles. Some great macros to use for this are *(contentbylabel)*, *(listlabels)* and the *(dynamiccontentbylabels)* macro in the Content Survey Plugin.

**Next Steps**

You now have a good idea of how Confluence templates work. What next? Take a look at Proactive Communications in a Knowledge Base.