

# How do I structure my IT documentation?

Dr. Richard Heigl





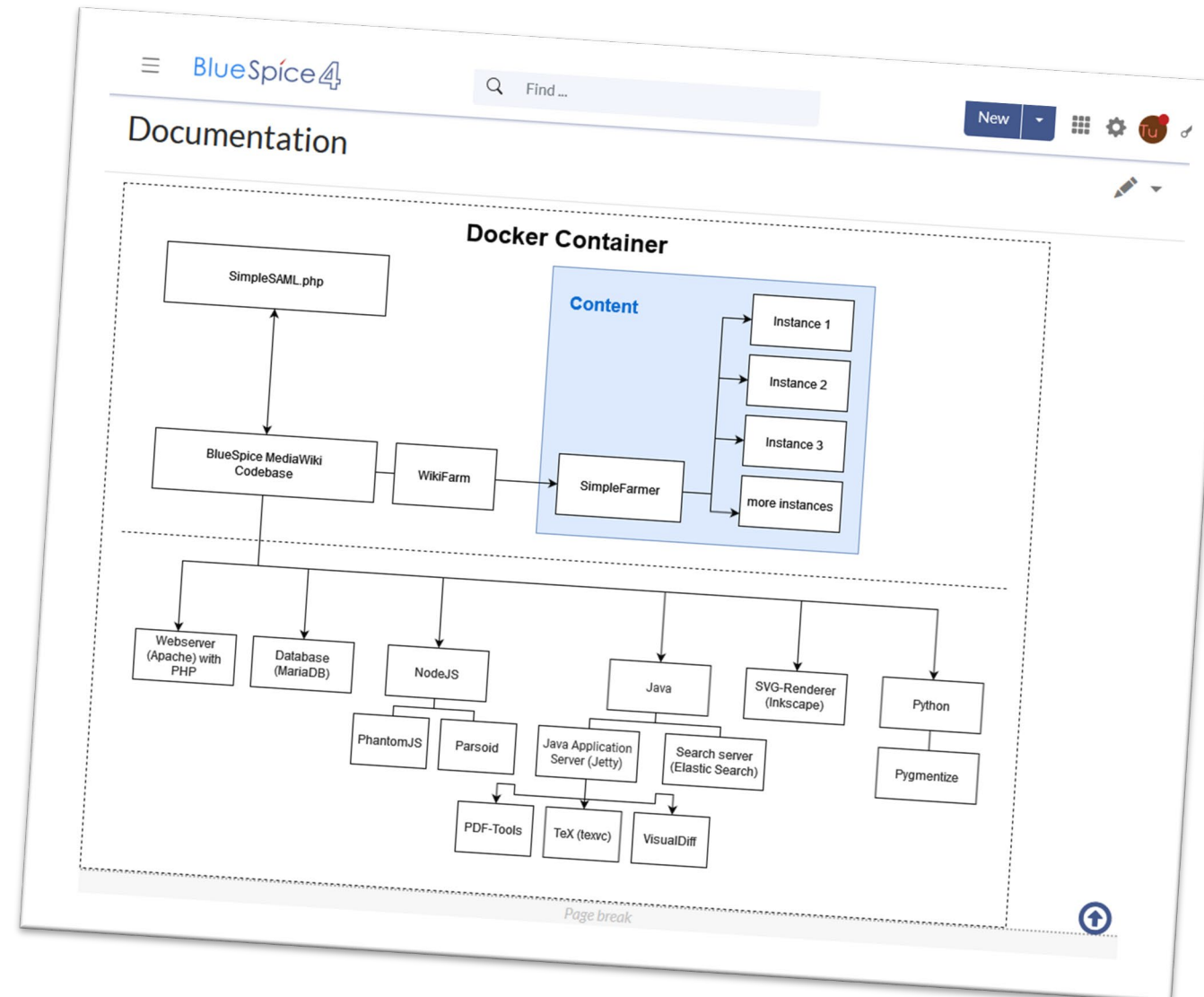
**Are there best practices?**

**Yes.**



# 1. Define your Use Case

- Software Documentation
- Online help, FAQ and how-to's
- IT process management (e.g. IT emergency manual)
- Customer and installation documentation
- IT project documentation (e.g. introduction of new services)
- Cataloging or inventory





**Start with one!**

## My example

- Online documentation of a complex enterprise software:  
[cargo support](#)
- Some typical requirements
  - Documentation of the various modules with multiple versions
  - Release notes
  - Optionally multilingual expandable
  - Space for internal information
  - Information should be easy to find
  - ... many others ...

## 2. Determine key points of your documentation

- **Document types:** software descriptions, minutes, checklists, planning documents
- **Topic groups:** teams, departments, projects
- **Access** permissions
- Pay special attention to **versions and releases** of software

- **Assistance systems:** workflows, approvals, reading confirmation
- Supported **languages**

Read more in the BlueSpice Blog:

- [Structuring heterogeneous-content](#)
- [Categories, document types and namespaces](#)



**Create a distribution and  
structuring plan!**





**But keep it simple.**

## 3. Distribute and structure content

Classification system	Recommended for...	In our example ...
Namespaces	<p><b>Document types</b> Topic / product cluster</p> <p><b>Assistant systems and access restriction</b> needed: workflows, approvals and other functions</p>	<p>Software modules, Release notes, Space for internal knowledge base</p>
Subwikis	<p>Secure workspace for <b>teams</b> (e.g. HR) or documentation in other languages</p>	<p>Not needed in the first step, optional for internal wiki or English edition</p>
Categories	<p><b>Topics for teams and projects</b> Topic / product cluster</p>	<p>Clustering of topics for teams and working groups Search support</p>
Semantic	<p>Detailed <b>metadata</b> and in-depth information</p>	<p>In our case not needed in the first step</p>

## 3. Distribute and structure content

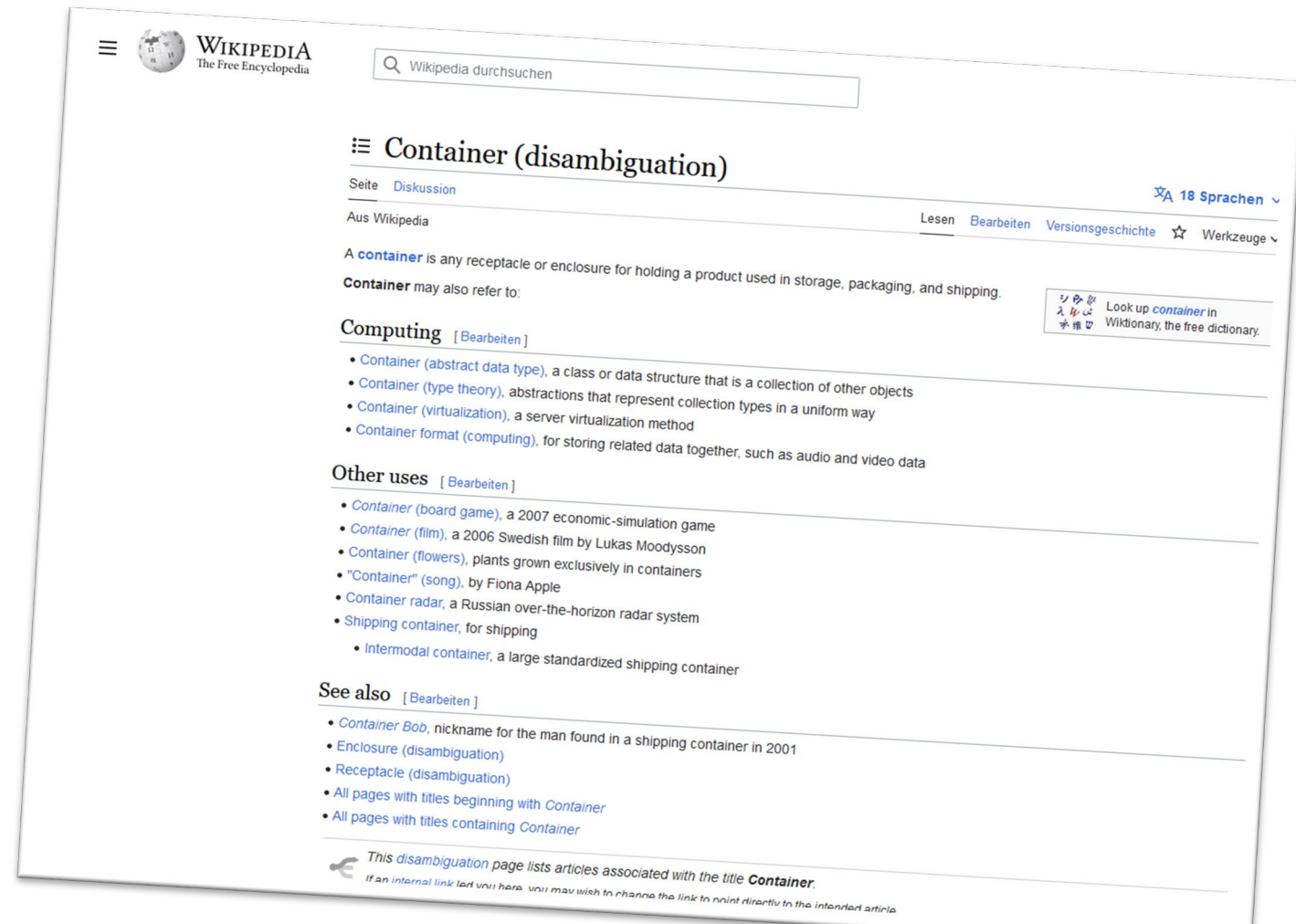
Navigation system	Recommended for...	In our example ...
Subpages	<b>Nested information</b> on a single topic Easy support of multilingualism Creation of simple alphabetical navigation	Releases: Detailed descriptions of new features
Book	<b>Training materials and manuals</b> Documentation Creation of a simple editorially maintained navigation	Used to display an article hierarchy (one "book" per module) Books can be exported
Left navigation	<b>Link to functions</b> (main page, latest changes, books), <b>document types</b> (guidelines, minutes, how-tos, manuals), topics	Quicklinks to modules, Release documentation, Internal pages
Mega menu	Links to <b>topics</b>	Not used



**Set a few design rules.**

## 4. Ensure basic orientation and support for the search

- Browsing along user journeys
  - e.g. Mainpage → Module portal (with entry points for manuals, functions and technology, “Recent changes”, highlighted topic) → Function description
- Search optimization
  - Provide overview articles (e.g. Container) and disambiguation pages
  - Work with redirects



## 4. Ensure basic orientation and support for the search

- Title
  - As simple and meaningful a title as possible
  - Avoidance of special characters
  - If the name is the same, the title should be extended by the respective topic in brackets
- Avoiding subpages



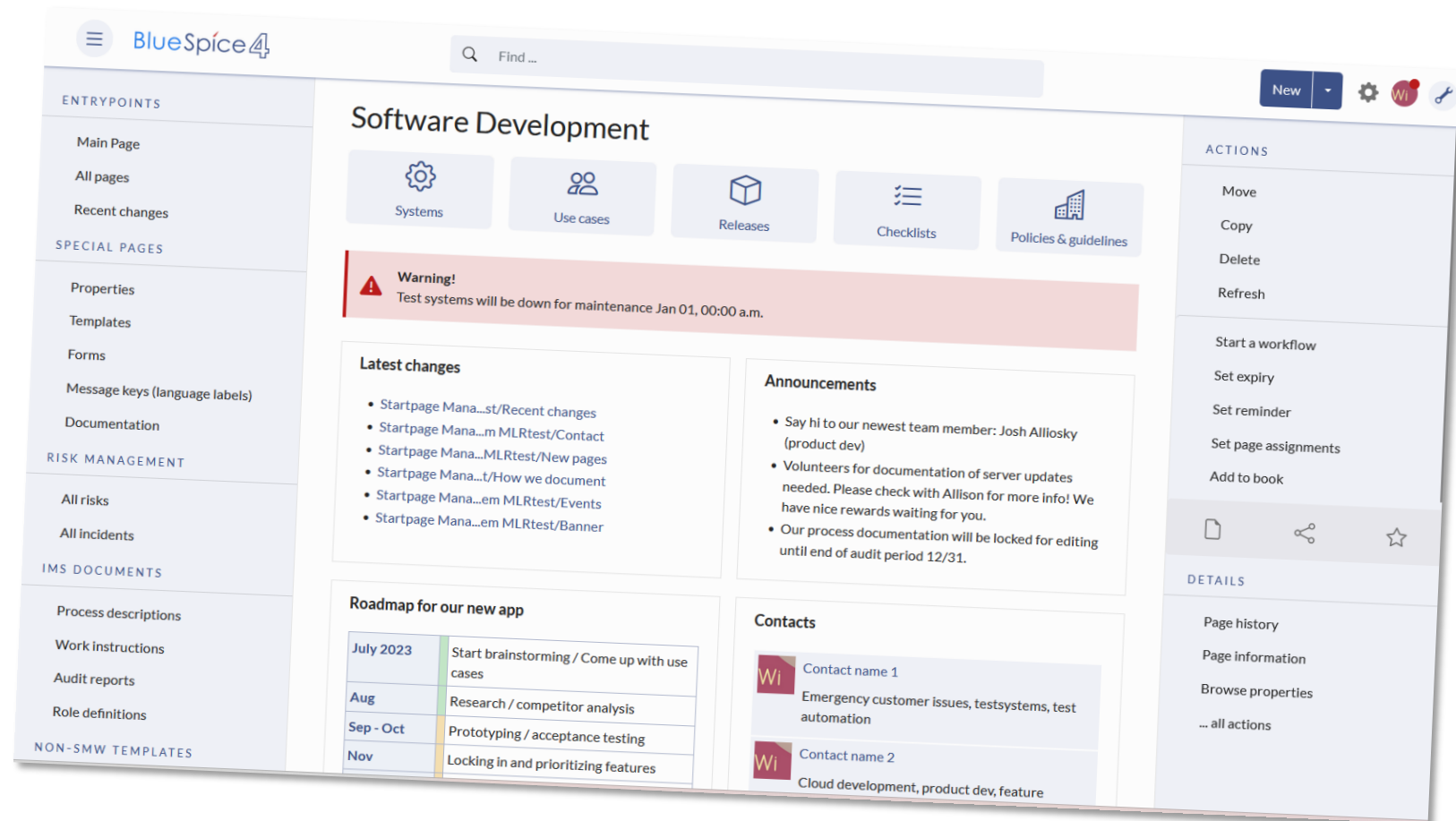
# 5. Working out with templates for portals and pages

## ■ Portals

- Internal Helpdesk
- Software development
- Team page
- ...

## ■ Search in topics

## ■ Page templates





**Thank you for your attention!**



# Contact us!

E-mail: [sales@bluespice.com](mailto:sales@bluespice.com)

Telephone: +49 (0)941 660 80 185

[www.bluespice.com](http://www.bluespice.com)

[www.hallowelt.com](http://www.hallowelt.com)

