

9/18/2013

# MITK-Doku & Doxygen

Do's and Don'ts

1. MITK-Doku
2. Documenting Code
3. How To Doxygen

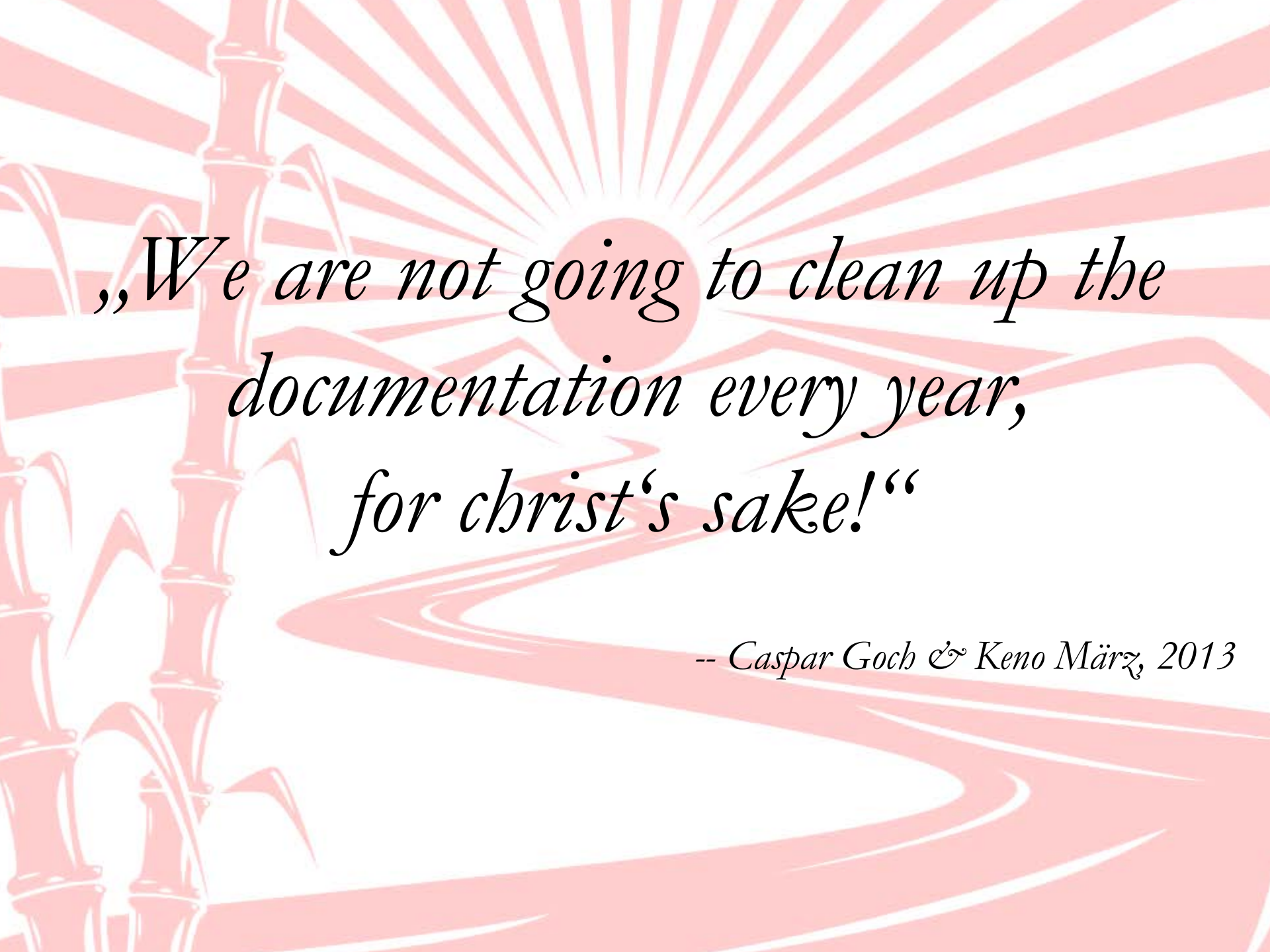
# MITK Doku – Neue Struktur

- 1. Overview** ← Target Group: „Soon to be MITK Users“
  - Auswahl verschiedener Themen
  - Erklärung der Grundbegriffe
- 2. User Manual** ← Target Group: Non-Developers
  - User Manual
  - Plugin-Dokumentation
- 3. Developer Manual** ← Target Group: Developers
  - Quickstart
  - MITK Concepts
  - Module Manuals
  - Developing with the MITK Application Framework
  - Deployment

The background features a stylized sunburst with radiating lines in shades of pink and red. On the left side, there are vertical bamboo stalks with curved leaves, also rendered in pink and red tones. The overall aesthetic is clean and modern with a warm color palette.

*„Thou shalt add your new Doxygen pages to the matching subgroup instead of creating a new group“*

*-- Caspar Goch & Keno März, 2013*



*„We are not going to clean up the  
documentation every year,  
for christ's sake!“*

*-- Caspar Goch & Keno März, 2013*

# I'm writing a Plugin

- **What:** Write a plugin manual!
- **Where:** Your plugin has a dedicated doxygen folder.
- **How:**
  - User-Centric (What is YOUR user?)
  - Clinical / Research staff must understand it
  - Link to other plugins that are useful in typical tasks.
  - But only to those with equal dependencies!
  - Avoid subpages (Think PDF)



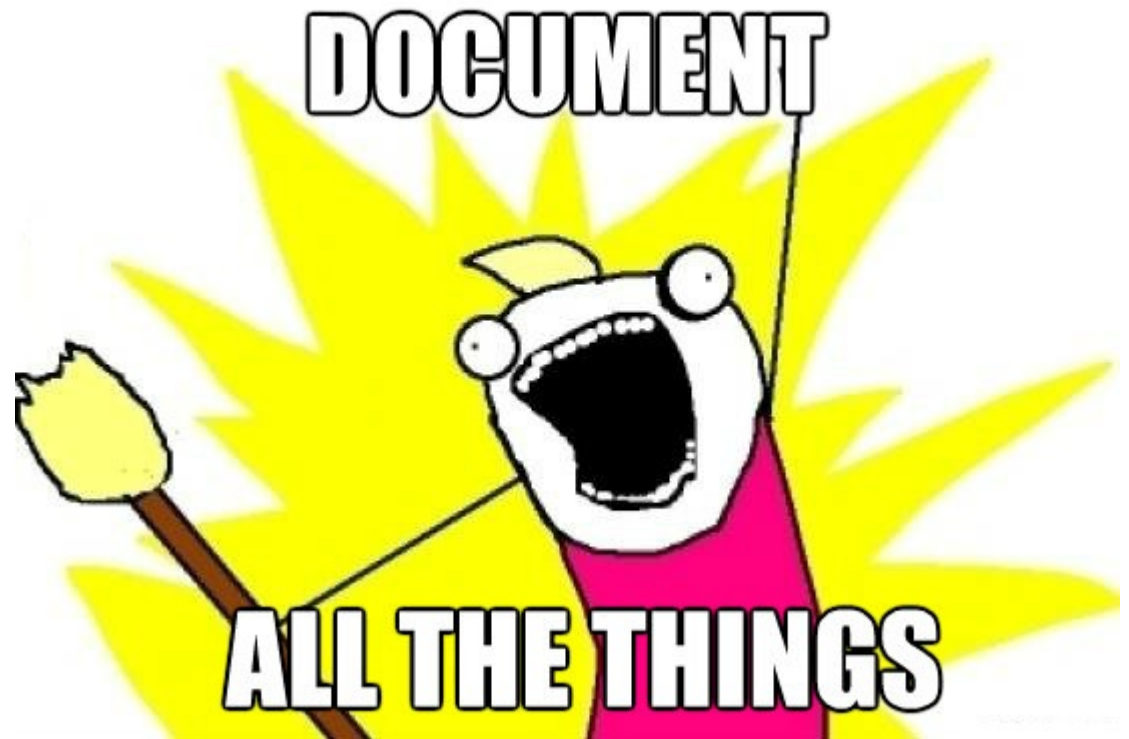
# I'm writing a Module

- **What:** Write a Module Documentation!
- **Where:** Documentation\Doxygen\DeveloperManual\Toolkit\ModuleManuals
- **How:**
  - Developer-Centric
  - Explain the intended workflow
  - Explain constraints
  - Explain module concepts
  - Name dependencies
  - Avoid subpages (Think PDF)



## I'm on my Platform Project

- **What:** Document the new Concept
- **Where:** Core\Documentation\Doxygen\Concepts
- **How:**
  - Developer-Centric
  - Explain the intended workflow
  - Explain constraints
  - Avoid subpages (!)





## I'm working with MITK

- **What:** Document your sodding code!
- **Where:** In your header files
- **How:**
  - Follow this styleguide:
    - Examples\Documentation\Example.h

- Document NOW not LATER
  - or else →



## Undocumented Modules

# of Modules: **38** (/wo submodules)

# of documented Modules: **6**

Percentage of documented Modules:

**15,8%**

*Give Modules their own Doxygen  
Folders (Like Plugins)*

*To be discussed*

I don't know how  
to Doxygen!



## Document your code

```
/**  
 * \brief A short (!) description of your class / method.  
 *  
 * The more detailed description is needed for some more  
 * elaborate description. Here you can describe  
 * anything anyone might ever want to know about your class. Of  
 * especial interest might be to mention  
 * what it can be used for or what its main purpose is. If you  
 * want you can even use pictures (you might want  
 * take a look at the doxygen documentation for that).  
 */
```

# How to Doxygen

- HTML + Latex: Use both!
- E.g. Lists

```
<ul>  
  <li>  
  <li>  
</ul>
```

## Sections

Use names hierarchically!  
Names must be globally unique!

### Page:

`\page` PageName Human Readable Name

### Section:

`\section` PageNameSectionName Human Readable Name

### Subsection:

`\subsection` PageNameSectionNameSubsectionName Human Readable Name

# Linking

## Link by Reference:

`\ref Name :`

Add a link to the Section or Page. Display [Human Readable Name](#)

## Reference as Subpage:

`\subpage Name:`

Add a link to the Section or Page. Display [Human Readable Name](#).  
Also make the linked page a subpage of this page.

## External Link:

`<a href=„http://www.link.address.de“> Human Readable Link Name</a>`

Link to static external site



## Table of contents

- Don't write TOC manually!
- `\tableofcontents`

### Table of Contents

↓ What are Applications?

↓ Which Application am I using?

↓ List of Applications

# Referencing Classes

## Reference Class

For further information, see class `mitk::MagicFilter`.

Generates a link to the file documentation.

# Markdown

- Doxygen supports Markdown
- Designed to make documentation readable as plain text
- Reference: <http://www.stack.nl/~dimitri/doxygen/manual/markdown.html>

# Markdown Examples

## Headers

This is an level 1 header  
=====

This is an level 2 header  
-----

# Markdown Examples

## Lists

- Item 1  
More text for this item.
- Item 2
  - + nested list item.
  - + another nested item.
- Item 3

# Markdown Examples

## Lists

- Item 1  
More text for this item.
- Item 2
  - + nested list item.
  - + another nested item.
- Item 3

# Markdown Examples

## Code Blocks

Preformatted verbatim blocks can be created by indenting each line in a block of text by at least 4 extra spaces

This a normal paragraph

```
This is a code block
```

We continue with a normal paragraph again.