Implementation of the Documentation Tools

Gerrit Muller

Philips Research
IST-SWA-AME
Prof Holstlaan 4 (WL01)
5656 AA Eindhoven
The Netherlands
gerrit.muller@philips.com
http://www.extra.research.philips.com/natlab/sysarch/

version: 1.0

22nd December 1999

Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

1 Introduction

The Gaudí project will produce a large number of articles and presentations about System Architecture. The first 3 months of the project were used for exploration of reader requirements, potential contents, distribution media and support tools.

This article describes the implementation of the tools to fulfil the quest of making the art of system architecting more accessible and understandable.

2 Naming Conventions

All files belonging to a single subject are stored in a subdirectory with the name **<SubjectName>**. For every type of output a rootfile exists, with the following names:

Article or report <Subjectname>Paper.tex

HTML <Subjectname>HTML.tex

Presentation <Subjectname>Slides.tex

Figures have descriptive names, without spaces, every word starting with a capital, for instance GenerationFlow.vsd. The file-extensions follow the standard meaning, vsd for ViSio Drawings and eps for Encapsulated PostScript.

Labels are prefixed with the type of reference:

fig: figures

tab: tables

sec: sections

req: requirements

Bibliography labels are prefixed with the type of publication:

www: internet publications

book: books

The files **Introduction.tex** and **history.tex** are always present and read via \input to save paper. Sections are stored in a file, with a descriptive name without spaces, words starting with a capital, for instance NamingConventions.tex. Those files are read via \include to enable partial generation. Finer granularity than one file per section is possible.

3 New LaTeX commands

3.1 Include Figure Commands

The commands to include a figure generate a floating picture, with a preferred positioning of *thb*. A label is generated with the name fig:<FileNameWithoutExtension>. Figures are resized depending on the command. For many figures fitting to the width of the text is the most convenient.

/LatexMyCommands/FigureCommand.tex

command	parameters
\FigureTextwidth	FileNameWithoutExtension, Caption
\FigureImposeWidth	FileNameWithoutExtension, Caption, Width
\SharedFigureTextwidth	Pathname, FileNameWithoutExtension, Caption
\SharedFigureImposeWidth	Pathname, FileNameWithoutExtension, Caption, Width

3.2 Slide Commands

The slide commands generate a single slide with a header, a body and a footer. The frontpage slide is generated without header and footer, because the meta information of the footer is shown more prominently in the body part of the frontpage. The body is centered in the vertical direction. Two commands are provided to include a graphics picture. The picture is either forced to fit horizontal or vertical. Two other commands can be used to put normal Latex in the body either direct or indirect from file.

/LatexMyCommand/SlideCommand.tex

command	parameters
\mySlideFrontPage	Abstract
\NoteOnDistribution	fine print shown on frontpage f.i. Confidential or Gaudí note
\mySlideFitHeight	SlideHeader, FileName
\mySlideFitWidth	SlideHeader, FileName
\mySlideInput	SlideHeader, FileName
\mySlide	SlideHeader, Text

The information in the footer is controlled predefined commands, which can be redefined by means of the renewcommand. These commands are defined in table 1

In the current implementation a poor mans option is used for the slidecounter:

command	default	comment
\authorname		
\BedrijfsUnit	Research	
\department		for instance: IST-SWA-AME
\version	-1	
\presentation		title of the presentation
\slidecount		text adjacent to the current slide number,
\Logo	Philips Logo	filename of the logo to be used
\LeftFooterWidth	140mm	
\RightFooterWidth	25mm	

Table 1: Commands which define the fields within the footer

the text behind the slidecounter total number of slides is explicitly defined, somehow I did not yet get \pageref to work correctly in the slide package.

The Slide commands internally use 2 commands to define the header and the footer:

myHeader {Text}

myFooter {FileName}

3.3 Article Commands

A number of additional commands is defined to generate a standard frontpage, especially the header of the frontpage.

/LatexMyCommands/ArticleCommand.tex

command	default	comment
\ArticleTitle		
\authorname		
\coauthor		
\address	address of building WL	
\email		
\homepage	URL Gaudi homepage	
\version		
\MyTitle		generates entire title

3.4 Requirement Commands

/LatexMyCommand/RequirementCommand

A special kind of list is defined to support numbered requirements. The numbering of requirements continues throughout the entire document. A label is defined explicitly for every requirement to allow for symbolic referencing.

begin{requirements}

req {label}{text}

end{requirements}

3.5 Specific Gaudí Commands

A few more specialized commands are defined for convinience of the Gaudí project:

/LatexMyCommands/GaudiCommand.tex

command	comment
\Gaudi	Prints Gaudí
\GaudiNoteOnDistribution	Text of Gaudí distribution note

3.6 History Commands

/LatexMyCommand/HistoryCommand

The histiry section at the end of every document is build up by a general header, an header per published version and a list of changes of such version relative to the previous version. This list is made by a specialized list environment which minimizes the vertical space. The new commands are:

/LatexMyCommands/HistoryCommand.tex

command	parameters	comment
\History		Generates the header
\CurrentVersion	version, date, author	Defines the current version; to be printed in normalsize characters
\OldVersion	version, date, author	Defines an old version; to be printed in tiny characters

\begin{changelog} ...\item ...\end{changelog} A list environment with minimized vertical spacing, to minimize paper use

4 TeXShell extensions

The TeXShell is a convenient combination of a simple editor with syntax highlighting and a simple extendable toolbar. The following buttons are added to the toolbar:

- BibTeX
- dvips
- GenPdf
- GenPdfL
- tex4ht
- GenHTML (not yet realized)

The existence of landscape specific buttons is an unwanted feature. It is a workaround for the fact that dvips has to be told explicitly to convert to landscape, even in teh case of landscape input (sic).

Most buttons perform a single step in the chain of transformations. The Gen* buttons automate the backend of the generation process and open the viewer with the final result.

It is expected that some of the buttons for intermediate results will be removed, when the generation process is running stable.

History

Version: 1.0, date: december 22 1999 changed by: Gerrit Muller

- homepage command added
- ArticleTitle command added
- SharedFigure commands added (additional pathname parameter)
- update of frontpage and footer layout

Version: 0, date: october 25 1999 changed by: Gerrit Muller

- Initial Version.