

# **SiSU - Manual**

Ralph Amissah

copy @ [www.jus.uio.no/sisu/](http://www.jus.uio.no/sisu/)

---

Copyright © Ralph Amissah 2007, part of SiSU documentation, License GPL 3;

## Contents

<b>SiSU - Manual, Ralph Amissah</b>	<b>1</b>
<b>What is SiSU?</b>	<b>1</b>
1. Introduction - What is SiSU? . . . . .	2
2. How does sisu work? . . . . .	5
3. Summary of features . . . . .	6
4. Help . . . . .	9
4.1 SiSU Manual . . . . .	9
4.2 SiSU man pages . . . . .	9
4.3 SiSU built-in interactive help . . . . .	9
5. Commands Summary . . . . .	11
5.1 Description . . . . .	11
5.2 Document Processing Command Flags . . . . .	11
6. command line modifiers . . . . .	18
7. database commands . . . . .	19
8. Shortcuts, Shorthand for multiple flags . . . . .	20
8.1 Command Line with Flags - Batch Processing . . . . .	20
9. Introduction to SiSU Markup . . . . .	21
9.1 Summary . . . . .	21
9.2 Markup Examples . . . . .	21
9.2.1 Online . . . . .	21
9.2.2 Installed . . . . .	22
10. Markup of Headers . . . . .	23
10.1 Sample Header . . . . .	23
10.2 Available Headers . . . . .	24
11. Markup of Substantive Text . . . . .	26
11.1 Heading Levels . . . . .	26
11.2 Font Attributes . . . . .	26
11.3 Indentation and bullets . . . . .	27
11.4 Footnotes / Endnotes . . . . .	28
11.5 Links . . . . .	29
11.5.1 Naked URLs within text, dealing with urls . . . . .	29
11.5.2 Linking Text . . . . .	30
11.5.3 Linking Images . . . . .	31
11.6 Grouped Text . . . . .	32
11.6.1 Tables . . . . .	32
11.6.2 Poem . . . . .	33
11.6.3 Group . . . . .	35
11.6.4 Code . . . . .	38
11.7 Book index . . . . .	40
12. Composite documents markup . . . . .	41

<b>Markup Syntax History</b>	<b>42</b>
13. Notes related to Files-types and Markup Syntax . . . . .	43
14. SiSU filetypes . . . . .	46
14.1 .sst .ssm .ssi marked up plain text . . . . .	46
14.1.1 sisu text - regular files (.sst) . . . . .	46
14.1.2 sisu master files (.ssm) . . . . .	46
14.1.3 sisu insert files (.ssi) . . . . .	47
14.2 sisupod, zipped binary container (sisupod.zip, .ssp) . . . . .	47
15. Experimental Alternative Input Representations . . . . .	48
15.1 Alternative XML . . . . .	48
15.1.1 XML SAX representation . . . . .	48
15.1.2 XML DOM representation . . . . .	48
15.1.3 XML Node representation . . . . .	49
16. Configuration . . . . .	50
16.1 Determining the Current Configuration . . . . .	50
16.2 Configuration files (config.yml) . . . . .	50
17. Skins . . . . .	51
17.1 Document Skin . . . . .	51
17.2 Directory Skin . . . . .	51
17.3 Site Skin . . . . .	51
17.4 Sample Skins . . . . .	51
18. CSS - Cascading Style Sheets (for html, XHTML and XML) . . . . .	53
19. Organising Content . . . . .	54
19.1 Directory Structure and Mapping . . . . .	54
19.1.1 General Directories . . . . .	54
19.1.2 Remote Directories . . . . .	55
19.1.3 Sisupod . . . . .	55
19.2 Organising Content . . . . .	56
20. Homepages . . . . .	57
20.1 Home page and other custom built pages in a sub-directory . . . . .	57
20.2 Home page within a skin . . . . .	57
21. Markup and Output Examples . . . . .	58
21.1 Markup examples . . . . .	58
21.2 A few book (and other) examples . . . . .	59
21.2.1 “ <i>Viral Spiral</i> ”, David Bollier . . . . .	59
“ <i>The Wealth of Networks</i> ”, Yochai Benkler . . . . .	60
“ <i>Two Bits</i> ”, Christopher Kelty . . . . .	60
“ <i>Free Culture</i> ”, Lawrence Lessig . . . . .	60
“ <i>CONTENT</i> ”, Cory Doctorow . . . . .	61
“ <i>Democratizing Innovation</i> ”, by Eric von Hippel . . . . .	61
“ <i>Free as in Freedom: Richard Stallman’s Crusade for Free Software</i> ”, by Sam Williams . . . . .	62
“ <i>Free For All: How Linux and the Free Software Movement Undercut         the High Tech Titans</i> ”, by Peter Wayner . . . . .	62

## Contents

---

“ <i>The Cathedral and the Bazaar</i> ”, by Eric S. Raymond	63
“ <i>Down and out in the Magic Kingdom</i> ”, Cory Doctorow	63
“ <i>Little Brother</i> ”, Cory Doctorow	64
“ <i>For the Win</i> ”, Cory Doctorow	64
“ <i>Accelerando</i> ”, Charles Stross	65
“ <i>Tainaron</i> ”, Leena Krohn	65
“ <i>Sphinx or Robot</i> ”, Leena Krohn	66
“ <i>War and Peace</i> ”, Leo Tolstoy, PG Etext 2600	66
“ <i>Don Quixote</i> ”, Miguel de Cervantes [Saavedra], translated by John Ormsby, PG Etext 996	67
“ <i>Gulliver’s Travels</i> ”, Jonathan Swift, transcribed from the 1892 George Bell and Sons edition by David Price, PG Etext 829	68
“ <i>Alice’s Adventures in Wonderland</i> ”, Lewis Carroll, PG Etext 11	68
“ <i>Through The Looking-Glass</i> ”, Lewis Carroll, PG Etext 12	69
“ <i>Alice’s Adventures in Wonderland</i> ” and “ <i>Through The Looking-Glass</i> ”, Lewis Carroll, PG Etexts 11 and 12	69
“ <i>Gnu Public License 2</i> ”, (GPL 2) Free Software Foundation	70
“ <i>Gnu Public License v3 - Third discussion draft</i> ”, (GPLv3) Free Software Foundation	70
“ <i>Debian Social Contract</i> ”	70
“ <i>Debian Constitution v1.3</i> ”, (simple/default markup)	71
“ <i>Debian Constitution v1.3</i> ”, (markup adjusted for output to more closely match the original)	71
“ <i>Debian Constitution v1.2</i> ”, (simple/default markup)	72
“ <i>Debian Constitution v1.2</i> ”, (markup adjusted for output to more closely match the original)	72
“ <i>A Uniform Sales Terminology</i> ”, Vikki Rogers and Albert Kritzer	73
“ <i>The Autonomous Contract</i> ” 1997 - markup sample	73
“ <i>The Autonomous Contract Revisited</i> ” - markup sample	74
“ <i>United Nations Convention on Contracts for the International Sale of Goods</i> ”	75
/PECL/ the “Principles of European Contract Law”	75
21.3 SQL - PostgreSQL, SQLite	76
21.4 Lex Mercatoria as an example	76
21.5 For good measure the markup for a document with lots of (simple) tables	77
21.6 And a link to the output of a reported case	77
22. A Checklist of Output Features	78
23. SiSU Search - Introduction	81
24. SQL	82
24.1 populating SQL type databases	82
25. Postgresql	83
25.1 Name	83
25.2 Description	83

## Contents

---

25.3 Synopsis	83
25.4 Commands	83
25.4.1 create and destroy database	83
25.4.2 import and remove documents	84
26. Sqlite	85
26.1 Name	85
26.2 Description	85
26.3 Synopsis	85
26.4 Commands	85
26.4.1 create and destroy database	85
26.4.2 import and remove documents	86
27. Introduction	87
27.1 Search - database frontend sample, utilising database and SiSU features, including object citation numbering (backend currently PostgreSQL)	87
27.2 Search Form	88
28. Hyperestraier	89
29. sisu_webrick	91
29.1 Name	91
29.2 Synopsis	91
29.3 Description	91
29.4 Summary of man page	91
29.5 Document processing command flags	91
29.6 Further information	91
29.7 Author	91
29.8 SEE ALSO	92
30. Remote Source Documents	93
<b>Remote Document Output</b>	<b>94</b>
31. Remote Output	95
31.1 commands	95
31.2 configuration	95
32. Remote Servers	96
<b>Download information</b>	<b>97</b>
33. Download SiSU - Linux/Unix	98
SiSU Current Version - Linux/Unix	98
Source (tarball tar.gz)	98
Git (source control management)	98
Debian	99
RPM	100
<b>Installation</b>	<b>102</b>
34. Installation	103
34.1 Debian	103

## Contents

---

34.2 Other Unix / Linux	104
34.2.1 source tarball	104
35. SiSU Components, Dependencies and Notes	106
35.1 sisu	106
35.2 sisu-complete	107
35.3 sisu-examples	108
35.4 sisu-pdf	108
35.5 sisu-postgresql	108
35.6 sisu-remote	109
35.7 sisu-sqlite	109
36. Quickstart - Getting Started Howto	111
36.1 Installation	111
36.1.1 Debian Installation	111
36.1.2 RPM Installation	111
36.1.3 Installation from source	112
36.2 Testing SiSU, generating output	113
36.2.1 basic text, plaintext, html, XML, ODF, EPUB	113
36.2.2 LaTeX / pdf	113
36.2.3 relational database - postgresql, sqlite	114
36.3 Getting Help	114
36.3.1 The man pages	114
36.3.2 Built in help	114
36.3.3 The home page	114
36.4 Markup Samples	115
<b>HowTo</b>	<b>116</b>
37. Getting Help	117
37.1 SiSU “man” pages	117
37.2 SiSU built-in help	117
37.3 Command Line with Flags - Batch Processing	117
38. Setup, initialisation	119
38.1 initialise output directory	119
38.1.1 Use of search functionality, an example using sqlite	119
38.2 misc	121
38.2.1 url for output files -u -U	121
38.2.2 toggle screen color	121
38.2.3 verbose mode	121
38.2.4 quiet mode	121
38.2.5 maintenance mode intermediate files kept -M	121
38.2.6 start the webrick server	121
38.3 remote placement of output	121
39. Configuration Files	123
40. Markup	124
40.1 Headers	124
40.2 Font Face	124

## Contents

---

40.2.1 Bold	125
40.2.2 Italics	125
40.2.3 Underscore	125
40.2.4 Strikethrough	125
40.3 Endnotes	125
40.4 Links	126
40.5 Number Titles	126
40.6 Line operations	127
40.7 Tables	127
40.8 Grouped Text	128
40.9 Composite Document	128
41. Change Appearance	129
41.1 Skins	129
41.2 CSS	129
<b>Extracts from the README</b>	<b>131</b>
42. README	132
42.1 Online Information, places to look	133
42.2 Installation	133
42.2.1 Debian	133
42.2.2 RPM	134
42.2.3 Source package .tgz	134
42.2.4 to use setup.rb	135
42.2.5 to use install (prepared with “Rake”)	135
42.2.6 to use install (prepared with “Rant”)	135
42.3 Dependencies	136
42.4 Quick start	140
42.5 Configuration files	141
42.6 Use General Overview	141
42.7 Help	141
42.8 Directory Structure	142
42.9 Configuration File	142
42.10 Markup	142
42.11 Additional Things	143
42.12 License	143
42.13 SiSU Standard	143
<b>Extracts from man 8 sisu</b>	<b>145</b>
43. Post Installation Setup	146
43.1 Post Installation Setup - Quick start	146
43.2 Document markup directory	147
43.2.1 Configuration files	147
43.2.2 Debian INSTALLATION Note	147
43.2.3 Document Resource Configuration	148
43.2.4 Skins	149

## Contents

---

44. FAQ - Frequently Asked/Answered Questions . . . . .	150
44.1 Why are urls produced with the -v (and -u) flag that point to a web server on port 8081 ? . . . . .	150
44.2 I cannot find my output, where is it? . . . . .	150
44.3 I do not get any pdf output, why? . . . . .	150
44.4 Where is the latex (or some other interim) output? . . . . .	150
44.5 Why isn't SiSU markup XML . . . . .	151
44.6 LaTeX claims to be a document preparation system for high-quality typesetting. Can the same be said about SiSU? . . . . .	151
44.7 Can the SiSU markup be used to prepare for a LaTeX automatic building of an index to the work? . . . . .	152
44.8 Can the conversion from SiSU to LaTeX be modified if we have special needs for the LaTeX, or do we need to modify the LaTeX manually? . . . . .	152
44.9 How do I create GIN or GiST index in Postgresql for use in SiSU . . . . .	153
44.10 Are there some examples of using Ferret Search with a SiSU reposi- tory? . . . . .	153
Have you had any reports of building SiSU from tar on Mac OS 10.4? . . . . .	154
44.12 Where is version 1? . . . . .	154
44.13 What is the difference between version 1 and 2? . . . . .	154
45. Who might be interested in the SiSU feature set? . . . . .	155
46. Work Needed . . . . .	156
47. Wishlist . . . . .	157
48. Editor Files, Syntax Highlighting . . . . .	158
49. Help Sources . . . . .	160
49.1 man pages . . . . .	160
49.2 sisu generated output - links to html . . . . .	160
49.2.1 www.sisudoc.org . . . . .	160
49.3 man2html . . . . .	161
49.3.1 locally installed . . . . .	161
49.3.2 www.jus.uio.no/sisu . . . . .	161
<b>Metadata</b> . . . . .	<b>162</b>
SiSU Metadata, document information . . . . .	162

## Contents

---

SiSU - Manual, Ralph Amissah	1
What is SiSU?	2

## 1. Introduction - What is SiSU?

**SiSU** is a framework for document structuring, publishing (in multiple open standard formats) and search, comprising of: (a) a lightweight document structure and presentation markup syntax; and (b) an accompanying engine for generating standard document format outputs from documents prepared in `sisu` markup syntax, which is able to produce multiple standard outputs (including the population of sql databases) that (can) share a common numbering system for the citation of text within a document.

**SiSU** is developed under an open source, software libre license (GPL3). Its use case for development is work with medium to large document sets and cope with evolving document formats/ representation technologies. Documents are prepared once, and generated as need be to update the technical presentation or add additional output formats. Various output formats (including search related output) share a common mechanism for cross-output-format citation.

**SiSU** both defines a markup syntax and provides an engine that produces open standards format outputs from documents prepared with **SiSU** markup. From a single lightly prepared document `sisu` custom builds several standard output formats which share a common (text object) numbering system for citation of content within a document (that also has implications for search). The `sisu` engine works with an abstraction of the document's structure and content from which it is possible to generate different forms of representation of the document. Significantly **SiSU** markup is more sparse than html and outputs which include html, EPUB, LaTeX, landscape and portrait pdfs, Open Document Format (ODF), all of which can be added to and updated. **SiSU** is also able to populate SQL type databases at an object level, which means that searches can be made with that degree of granularity.

Source document preparation and output generation is a two step process: (i) document source is prepared, that is, marked up in `sisu` markup syntax and (ii) the desired output subsequently generated by running the `sisu` engine against document source. Output representations if updated (in the `sisu` engine) can be generated by re-running the engine against the prepared source. Using **SiSU** markup applied to a document, **SiSU** custom builds (to take advantage of the strengths of different ways of representing documents) various standard open output formats including plain text, HTML, XHTML, XML, EPUB, OpenDocument, LaTeX or PDF files, and populate an SQL database with objects<sup>1</sup> (equating generally to paragraph-sized chunks) so searches may be performed and matches returned with that degree of granularity ( e.g. your search criteria is met by these documents and at these locations within each document). Document output formats share a common object numbering system for locating content. This is particularly suitable for “published” works (finalized texts as opposed to works that are frequently changed or updated) for which it provides a fixed means of reference of content.

In preparing a **SiSU** document you optionally provide semantic information related to the document in a document header, and in marking up the substantive text provide

---

<sup>1</sup>objects include: headings, paragraphs, verse, tables, images, but not footnotes/endnotes which are numbered separately and tied to the object from which they are referenced.

information on the structure of the document, primarily indicating heading levels and footnotes. You also provide information on basic text attributes where used. The rest is automatic, `sisu` from this information custom builds<sup>2</sup> the different forms of output requested.

**SiSU** works with an abstraction of the document based on its structure which is comprised of its headings<sup>3</sup> and objects<sup>4</sup>, which enables **SiSU** to represent the document in many different ways, and to take advantage of the strengths of different ways of presenting documents. The objects are numbered, and these numbers can be used to provide a common basis for citing material within a document across the different output format types. This is significant as page numbers are not well suited to the digital age, in web publishing, changing a browser's default font or using a different browser can mean that text will appear on a different page; and publishing in different formats, html, landscape and portrait pdf etc. again page numbers are not useful to cite text. Dealing with documents at an object level together with object numbering also has implications for search that **SiSU** is able to take advantage of.

One of the challenges of maintaining documents is to keep them in a format that allows use of them independently of proprietary platforms. Consider issues related to dealing with legacy proprietary formats today and what guarantee you have that old proprietary formats will remain (or can be read without proprietary software/equipment) in 15 years time, or the way the way in which html has evolved over its relatively short span of existence. **SiSU** provides the flexibility of producing documents in multiple non-proprietary open formats including html, pdf<sup>5</sup> ODF,<sup>6</sup> and EPUB.<sup>7</sup> Whilst **SiSU** relies on software, the markup is uncomplicated and minimalistic which guarantees that future engines can be written to run against it. It is also easily converted to other formats, which means documents prepared in **SiSU** can be migrated to other document formats. Further security is provided by the fact that the software itself, **SiSU** is available under GPL3 a licence that guarantees that the source code will always be open, and free as in libre, which means that that code base can be used, updated and further developed as required under the terms of its license. Another challenge is to keep up with a moving target. **SiSU** permits new forms of output to be added as they become important, (Open Document Format text was added in 2006 when it became an ISO standard for office applications and the archival of documents), EPUB was introduced in 2009; and allows the technical representations existing output to be updated (html has evolved and the related module has been updated repeatedly over the years, presumably when the World Wide Web Consortium (w3c) finalises html 5 which is currently under development, the html module will again be updated allowing all existing documents to be regenerated as html 5).

---

<sup>2</sup>i.e. the html, pdf, epub, odf outputs are each built individually and optimised for that form of presentation, rather than for example the html being a saved version of the odf, or the pdf being a saved version of the html.

<sup>3</sup>the different heading levels

<sup>4</sup>units of text, primarily paragraphs and headings, also any tables, poems, code-blocks

<sup>5</sup>Specification submitted by Adobe to ISO to become a full open ISO specification

<<http://www.linux-watch.com/news/NS7542722606.html>>

<sup>6</sup>ISO standard ISO/IEC 26300:2006

<sup>7</sup>An open standard format for e-books

The document formats are written to the file-system and available for indexing by independent indexing tools, whether off the web like Google and Yahoo or on the site like Lucene and Hyperestraier. 11

**SiSU** also provides other features such as concordance files and document content certificates, and the working against an abstraction of document structure has further possibilities for the research and development of other document representations, the availability of objects is useful for example for topic maps and thesauri, together with the flexibility of **SiSU** offers great possibilities. 12

**SiSU** is primarily for published works, which can take advantage of the citation system to reliably reference its documents. **SiSU** works well in a complementary manner with such collaborative technologies as Wikis, which can take advantage of and be used to discuss the substance of content prepared in **SiSU** . 13

<<http://www.sisudoc.org/>> 14

<<http://www.jus.uio.no/sisu/>> 15

## 2. How does sisu work?

16

**SiSU** markup is fairly minimalistic, it consists of: a (largely optional) document header, made up of information about the document (such as when it was published, who authored it, and granting what rights) and any processing instructions; and markup within the substantive text of the document, which is related to document structure and typeface. **SiSU** must be able to discern the structure of a document, (text headings and their levels in relation to each other), either from information provided in the document header or from markup within the text (or from a combination of both). Processing is done against an abstraction of the document comprising of information on the document's structure and its objects,[2] which the program serializes (providing the object numbers) and which are assigned hash sum values based on their content. This abstraction of information about document structure, objects, (and hash sums), provides considerable flexibility in representing documents different ways and for different purposes (e.g. search, document layout, publishing, content certification, concordance etc.), and makes it possible to take advantage of some of the strengths of established ways of representing documents, (or indeed to create new ones).

17

### 3. Summary of features

Γ sparse/minimal markup (clean utf-8 source texts). Documents are prepared in a single UTF-8 file using a minimalistic mnemonic syntax. Typical literature, documents like “War and Peace” require almost no markup, and most of the headers are optional.

Γ markup is easily readable/parsable by the human eye, (basic markup is simpler and more sparse than the most basic HTML), [this may also be converted to XML representations of the same input/source document].

Γ markup defines document structure (this may be done once in a header pattern-match description, or for heading levels individually); basic text attributes (bold, italics, underscore, strike-through etc.) as required; and semantic information related to the document (header information, extended beyond the Dublin core and easily further extended as required); the headers may also contain processing instructions. SiSU markup is primarily an abstraction of document structure and document metadata to permit taking advantage of the basic strengths of existing alternative practical standard ways of representing documents [be that browser viewing, paper publication, sql search etc.] (html, epub, xml, odf, latex, pdf, sql)

Γ for output produces reasonably elegant output of established industry and institutionally accepted open standard formats.[3] takes advantage of the different strengths of various standard formats for representing documents, amongst the output formats currently supported are:

Γ html - both as a single scrollable text and a segmented document

Γ xhtml

Γ epub

Γ XML - both in sax and dom style xml structures for further development as required

Γ ODF - open document format, the iso standard for document storage

Γ LaTeX - used to generate pdf

Γ pdf (via LaTeX)

Γ sql - population of an sql database, (at the same object level that is used to cite text within a document)

Also produces: concordance files; document content certificates (md5 or sha256 digests of headings, paragraphs, images etc.) and html manifests (and sitemaps of content). (b) takes advantage of the strengths implicit in these very different output types, (e.g. PDFs produced using typesetting of LaTeX, databases populated with documents at an individual object/paragraph level, making possible granular search (and related possibilities))

Γ ensuring content can be cited in a meaningful way regardless of selected output format. Online publishing (and publishing in multiple document formats) lacks a useful way of citing text internally within documents (important to academics generally and to lawyers) as page numbers are meaningless across browsers and formats. sisu seeks

to provide a common way of pinpoint the text within a document, (which can be utilized for citation and by search engines). The outputs share a common numbering system that is meaningful (to man and machine) across all digital outputs whether paper, screen, or database oriented, (pdf, HTML, EPUB, xml, sqlite, postgresql), this numbering system can be used to reference content.

Γ Granular search within documents. SQL databases are populated at an object level (roughly headings, paragraphs, verse, tables) and become searchable with that degree of granularity, the output information provides the object/paragraph numbers which are relevant across all generated outputs; it is also possible to look at just the matching paragraphs of the documents in the database; [output indexing also work well with search indexing tools like hyperestraier]. 33

Γ long term maintainability of document collections in a world of changing formats, having a very sparsely marked-up source document base. there is a considerable degree of future-proofing, output representations are “upgradeable”, and new document formats may be added. e.g. addition of odf (open document text) module in 2006, epub in 2009 and in future html5 output sometime in future, without modification of existing prepared texts 34

Γ SQL search aside, documents are generated as required and static once generated. 35

Γ documents produced are static files, and may be batch processed, this needs to be done only once but may be repeated for various reasons as desired (updated content, addition of new output formats, updated technology document presentations/representations) 36

Γ document source (plaintext utf-8) if shared on the net may be used as input and processed locally to produce the different document outputs 37

Γ document source may be bundled together (automatically) with associated documents (multiple language versions or master document with inclusions) and images and sent as a zip file called a sisupod, if shared on the net these too may be processed locally to produce the desired document outputs 38

Γ generated document outputs may automatically be posted to remote sites. 39

Γ for basic document generation, the only software dependency is **Ruby** , and a few standard Unix tools (this covers plaintext, HTML, EPUB, XML, ODF, LaTeX). To use a database you of course need that, and to convert the LaTeX generated to pdf, a latex processor like tetex or texlive. 40

Γ as a developers tool it is flexible and extensible 41

Syntax highlighting for **SiSU** markup is available for a number of text editors. 42

**SiSU** is less about document layout than about finding a way with little markup to be able to construct an abstract representation of a document that makes it possible to produce multiple representations of it which may be rather different from each other and used for different purposes, whether layout and publishing, or search of content 43

i.e. to be able to take advantage from this minimal preparation starting point of some of 44

the strengths of rather different established ways of representing documents for different purposes, whether for search (relational database, or indexed flat files generated for that purpose whether of complete documents, or say of files made up of objects), online viewing (e.g. html, xml, pdf), or paper publication (e.g. pdf)...

the solution arrived at is by extracting structural information about the document (about headings within the document) and by tracking objects (which are serialized and also given hash values) in the manner described. It makes possible representations that are quite different from those offered at present. For example objects could be saved individually and identified by their hashes, with an index of how the objects relate to each other to form a document. 45

## 4. Help 46

### 4.1 SiSU Manual 47

The most up to date information on sisu should be contained in the sisu\_manual, available at: 48

```
<http://sisudoc.org/sisu/sisu\_manual/> 49
```

The manual can be generated from source, found respectively, either within the **SiSU** tarball or installed locally at: 50

```
./data/doc/sisu/markup-samples/sisu_manual 51
```

```
/usr/share/doc/sisu/markup-samples/sisu_manual 52
```

move to the respective directory and type e.g.: 53

```
sisu sisu_manual.ssm 54
```

### 4.2 SiSU man pages 55

If **SiSU** is installed on your system usual man commands should be available, try: 56

```
man sisu 57
```

Most **SiSU** man pages are generated directly from sisu documents that are used to prepare the sisu manual, the sources files for which are located within the **SiSU** tarball at: 58

```
./data/doc/sisu/markup-samples/sisu_manual 59
```

Once installed, directory equivalent to: 60

```
/usr/share/doc/sisu/markup-samples/sisu_manual 61
```

Available man pages are converted back to html using man2html: 62

```
/usr/share/doc/sisu/html/ 63
```

```
./data/doc/sisu/html 64
```

An online version of the sisu man page is available here: 65

```
Γ <various sisu man pages> 8 66
```

```
Γ <sisu.1> 9 67
```

### 4.3 SiSU built-in interactive help 68

This is particularly useful for getting the current sisu setup/environment information: 69

```
sisu -help 70
```

```
sisu -help [subject] 71
```

sisu -help commands	72
sisu -help markup	73
sisu -help env [for feedback on the way your system is setup with regard to sisu]	74
sisu -V [environment information, same as above command]	75
sisu (on its own provides version and some help information)	76
Apart from real-time information on your current configuration the <b>SiSU</b> manual and man pages are likely to contain more up-to-date information than the sisu interactive help (for example on commands and markup).	77
NOTE: Running the command sisu (alone without any flags, filenames or wildcards) brings up the interactive help, as does any sisu command that is not recognised. Enter to escape.	78

---

<sup>8</sup> <<http://www.jus.uio.no/sisu/man/>>

<sup>9</sup> <<http://www.jus.uio.no/sisu/man/sisu.1.html>>

## 5. Commands Summary 79

### 5.1 Description 80

**SiSU** SiSU is a document publishing system, that from a simple single marked-up document, produces multiple of output formats including: plaintext, html, xhtml, XML, epub, odt (odf text), LaTeX, pdf, info, and SQL (PostgreSQL and SQLite), which share numbered text objects (“object citation numbering”) and the same document structure information. For more see: <http://www.jus.uio.no/sisu>

### 5.2 Document Processing Command Flags 82

#### **-a [filename/wildcard]** 83

produces plaintext with Unix linefeeds and without markup, (object numbers are omitted), has footnotes at end of each paragraph that contains them [ -A for equivalent dos (linefeed) output file] [see -e for endnotes]. (Options include: -endnotes for endnotes -footnotes for footnotes at the end of each paragraph -unix for unix linefeed (default) -msdos for msdos linefeed)

#### **-b [filename/wildcard]** 84

see -xhtml

#### **-color-toggle [filename/wildcard]** 85

screen toggle ansi screen colour on or off depending on default set (unless -c flag is used: if sisurc colour default is set to ‘true’, output to screen will be with colour, if sisurc colour default is set to ‘false’ or is undefined screen output will be without colour). Alias -c

#### **-concordance [filename/wildcard]** 86

produces concordance (wordmap) a rudimentary index of all the words in a document. (Concordance files are not generated for documents of over 260,000 words unless this limit is increased in the file sisurc.yml). Alias -w

#### **-C [-init-site]** 87

configure/initialise shared output directory files initialize shared output directory (config files such as css and dtd files are not updated if they already exist unless modifier is used). -C -init-site configure/initialise site more extensive than -C on its own, shared output directory files/force update, existing shared output config files such as css and dtd files are updated if this modifier is used.

#### **-CC** 88

configure/initialise shared output directory files initialize shared output directory (config files such as css and dtd files are not updated if they already exist unless modifier is used). The equivalent of: -C -init-site configure/initialise site, more extensive than -C on its own, shared output directory files/force update, existing shared output config files such as css and dtd files are updated if -CC is used.

#### **-c [filename/wildcard]** 89

see -color-toggle

- 
- dal** [filename/wildcard/url] 90  
 assumed for most other flags, creates new intermediate files for processing (document abstraction) that is used in all subsequent processing of other output. This step is assumed for most processing flags. To skip it see -n. Alias -m
- delete** [filename/wildcard] 91  
 see -zap
- D** [instruction] [filename] 92  
 see -pg
- d** [-db-[database type (sqlite|pg)]] [-instruction] [filename] 93  
 see -sqlite
- epub** [filename/wildcard] 94  
 produces an epub document, [sisu version 2 only] (filename.epub). Alias -e
- e** [filename/wildcard] 95  
 see -epub
- F** [-webserv=webrick] 96  
 see -sample-search-form
- git** [filename/wildcard] 97  
 produces or updates markup source file structure in a git repo (experimental and subject to change). Alias -g
- g** [filename/wildcard] 98  
 see -git
- harvest \*.ss[tm]** 99  
 makes two lists of sisu output based on the sisu markup documents in a directory: list of author and authors works (year and titles), and; list by topic with titles and author. Makes use of header metadata fields (author, title, date, topic\_register). Can be used with maintenance (-M) and remote placement (-R) flags.
- help** [topic] 100  
 provides help on the selected topic, where topics (keywords) include: list, (com)mands, short(cuts), (mod)ifiers, (env)ironment, markup, syntax, headers, headings, endnotes, tables, example, customise, skin, (dir)ectories, path, (lang)uage, db, install, setup, (conf)igure, convert, termsheet, search, sql, features, license
- html** [filename/wildcard] 101  
 produces html output, segmented text with table of contents (toc.html and index.html) and the document in a single file (scroll.html). Alias -h
- h** [filename/wildcard] 102  
 see -html
- I** [filename/wildcard] 103  
 see -texinfo
- i** [filename/wildcard] 104  
 see -manpage

- L** 105  
prints license information.
- machine [filename/wildcard/url]** 106  
see `-dal` (document abstraction level/layer)
- maintenance [filename/wildcard/url]** 107  
maintenance mode files created for processing preserved and their locations indicated.  
(also see `-V`). Alias `-M`
- manpage [filename/wildcard]** 108  
produces man page of file, not suitable for all outputs. Alias `-i`
- M [filename/wildcard/url]** 109  
see `-maintenance`
- m [filename/wildcard/url]** 110  
see `-dal` (document abstraction level/layer)
- no-ocn** 111  
[with `-html` `-pdf` or `-epub`] switches off object citation numbering. Produce output without identifying numbers in margins of html or LaTeX/pdf output.
- N [filename/wildcard/url]** 112  
document digest or document content certificate ( DCC ) as md5 digest tree of the document: the digest for the document, and digests for each object contained within the document (together with information on software versions that produced it) (digest.txt).  
`-NV` for verbose digest output to screen.
- n [filename/wildcard/url]** 113  
skip the creation of intermediate processing files (document abstraction) if they already exist, this skips the equivalent of `-m` which is otherwise assumed by most processing flags.
- odf [filename/wildcard/url]** 114  
see `-odt`
- odt [filename/wildcard/url]** 115  
output basic document in opendocument file format (opendocument.odt). Alias `-o`
- o [filename/wildcard/url]** 116  
see `-odt`
- pdf [filename/wildcard]** 117  
produces LaTeX pdf (portrait.pdf & landscape.pdf). Default paper size is set in config file, or document header, or provided with additional command line parameter, e.g. `-papersize-a4` preset sizes include: 'A4', U.S. 'letter' and 'legal' and book sizes 'A5' and 'B5' (system defaults to A4). Alias `-p`
- pg [instruction] [filename]** 118  
database postgresql ( `-pgsql` may be used instead) possible instructions, include: `-createdb`; `-create`; `-dropall`; `-import [filename]`; `-update [filename]`; `-remove [filename]`; see database section below. Alias `-D`

- po** [**language\_directory/filename language\_directory**] 119  
see -po4a
- po4a** [**language\_directory/filename language\_directory**] 120  
produces .pot and po files for the file in the languages specified by the language directory. **SiSU** markup is placed in subdirectories named with the language code, e.g. en/ fr/ es/. The sisu config file must set the output directory structure to multilingual. v3, experimental
- P** [**language\_directory/filename language\_directory**] 121  
see -po4a
- p** [**filename/wildcard**] 122  
see -pdf
- quiet** [**filename/wildcard**] 123  
quiet less output to screen.
- q** [**filename/wildcard**] 124  
see -quiet
- rsync** [**filename/wildcard**] 125  
copies sisu output files to remote host using rsync. This requires that sisurc.yml has been provided with information on hostname and username, and that you have your “keys” and ssh agent in place. Note the behavior of rsync different if -R is used with other flags from if used alone. Alone the rsync -delete parameter is sent, useful for cleaning the remote directory (when -R is used together with other flags, it is not). Also see -scp. Alias -R
- R** [**filename/wildcard**] 126  
see -rsync
- r** [**filename/wildcard**] 127  
see -scp
- sample-search-form** [**-webserv=webrick**] 128  
generate examples of (naive) cgi search form for sqlite and postgresql depends on your already having used sisu to populate an sqlite and/or postgresql database, (the sqlite version scans the output directories for existing sisu\_sqlite databases, so it is first necessary to create them, before generating the search form) see -d -D and the database section below. If the optional parameter -webserv=webrick is passed, the cgi examples created will be set up to use the default port set for use by the webrick server, (otherwise the port is left blank and the system setting used, usually 80). The samples are dumped in the present work directory which must be writable, (with screen instructions given that they be copied to the cgi-bin directory). -Fv (in addition to the above) provides some information on setting up hyperestraier for sisu. Alias -F
- scp** [**filename/wildcard**] 129  
copies sisu output files to remote host using scp. This requires that sisurc.yml has been provided with information on hostname and username, and that you have your “keys” and ssh agent in place. Also see -rsync. Alias -r

- sqlite** **-[instruction]** **[filename]** 130  
 database type default set to sqlite, (for which `-sqlite` may be used instead) or to specify another database `-db-[pgsql, sqlite]` (however see `-D`) possible instructions include: `-createdb`; `-create`; `-dropall`; `-import [filename]`; `-update [filename]`; `-remove [filename]`; see database section below. Alias `-d`
- sisupod** 131  
 produces a `sisupod` a zipped `sisu` directory of markup files including `sisu` markup source files and the directories local configuration file, images and skins. Note: this only includes the configuration files or skins contained in `./_sisu` not those in `~/_sisu` `-S [filename/wildcard]` option. Note: (this option is tested only with `zsh`). Alias `-S`
- sisupod** **[filename/wildcard]** 132  
 produces a zipped file of the prepared document specified along with associated images, by default named `sisupod.zip` they may alternatively be named with the filename extension `.ssp` This provides a quick way of gathering the relevant parts of a `sisu` document which can then for example be emailed. A `sisupod` includes `sisu` markup source file, (along with associated documents if a master file, or available in multilingual versions), together with related images and skin. **SiSU** commands can be run directly against a `sisupod` contained in a local directory, or provided as a url on a remote site. As there is a security issue with skins provided by other users, they are not applied unless the flag `-trust` or `-trusted` is added to the command instruction, it is recommended that file that are not your own are treated as untrusted. The directory structure of the unzipped file is understood by `sisu`, and `sisu` commands can be run within it. Note: if you wish to send multiple files, it quickly becomes more space efficient to zip the `sisu` markup directory, rather than the individual files for sending). See the `-S` option without `[filename/wildcard]`. Alias `-S`
- source** **[filename/wildcard]** 133  
 copies `sisu` markup file to output directory. Alias `-s`
- S** 134  
 see `-sisupod`
- S** **[filename/wildcard]** 135  
 see `-sisupod`
- s** **[filename/wildcard]** 136  
 see `-source`
- texinfo** **[filename/wildcard]** 137  
 produces `texinfo` and `info` file, (view with `pinfo`). Alias `-I`
- txt** **[filename/wildcard]** 138  
 produces plaintext with Unix linefeeds and without markup, (object numbers are omitted), has footnotes at end of each paragraph that contains them [ `-A` for equivalent dos (linefeed) output file] [see `-e` for endnotes]. (Options include: `-endnotes` for endnotes `-footnotes` for footnotes at the end of each paragraph `-unix` for unix linefeed (default) `-msdos` for msdos linefeed). Alias `-t`
- T** **[filename/wildcard (\*.termsheet.rb)]** 139

standard form document builder, preprocessing feature

**-t [filename/wildcard]** 140  
see `-txt`

**-urls [filename/wildcard]** 141  
prints url output list/map for the available processing flags options and resulting files that could be requested, (can be used to get a list of processing options in relation to a file, together with information on the output that would be produced), `-u` provides url output mapping for those flags requested for processing. The default assumes `sisu_webrick` is running and provides webrick url mappings where appropriate, but these can be switched to file system paths in `sisurc.yml`. Alias `-U`

**-U [filename/wildcard]** 142  
see `-urls`

**-u [filename/wildcard]** 143  
provides url mapping of output files for the flags requested for processing, also see `-U`

**-v2 [filename/wildcard]** 144  
invokes the `sisu v2` document parser/generator. This is the default and is normally omitted.

**-v3 [filename/wildcard]** 145  
invokes the `sisu v3` document parser/generator. Currently under development and incomplete, `v3` requires `ruby1.9.2p180`. You may run `sisu3` instead.

**-verbose [filename/wildcard]** 146  
provides verbose output of what is being generated, where output is placed (and error messages if any), as with `-u` flag provides a url mapping of files created for each of the processing flag requests. Alias `-v`

**-V** 147  
on its own, provides **SiSU** version and environment information (`sisu -help env`)

**-V [filename/wildcard]** 148  
even more verbose than the `-v` flag.

**-v** 149  
on its own, provides **SiSU** version information

**-v [filename/wildcard]** 150  
see `-verbose`

**-webrick** 151  
starts ruby's webrick webserver points at `sisu` output directories, the default port is set to 8081 and can be changed in the resource configuration files. [tip: the webrick server requires link suffixes, so html output should be created using the `-h` option rather than `-H` ; also, note `-F webrick` ]. Alias `-W`

**-W** 152  
see `-webrick`

**-wordmap [filename/wildcard]** 153

see `-concordance`

**-w [filename/wildcard]** 154

see `-concordance`

**-xhtml [filename/wildcard]** 155

produces xhtml/XML output for browser viewing (sax parsing). Alias `-b`

**-xml-dom [filename/wildcard]** 156

produces XML output with deep document structure, in the nature of dom. Alias `-X`

**-xml-sax [filename/wildcard]** 157

produces XML output shallow structure (sax parsing). Alias `-x`

**-X [filename/wildcard]** 158

see `-xml-dom`

**-x [filename/wildcard]** 159

see `-xml-sax`

**-Y [filename/wildcard]** 160

produces a short sitemap entry for the document, based on html output and the `sisu_manifest`.

`-sitemaps` generates/updates the sitemap index of existing sitemaps. (Experimental, [g,y,m announcement this week])

**-y [filename/wildcard]** 161

produces an html summary of output generated (hyperlinked to content) and document specific metadata (`sisu_manifest.html`). This step is assumed for most processing flags.

**-zap [filename/wildcard]** 162

Zap, if used with other processing flags deletes output files of the type about to be processed, prior to processing. If `-Z` is used as the lone processing related flag (or in conjunction with a combination of `-[mMvVq]`), will remove the related document output directory. Alias `-Z`

**-Z [filename/wildcard]** 163

see `-zap`

<b>6. command line modifiers</b>	164
<b>-no-ocn</b>	165
[with -html -pdf or -epub] switches off object citation numbering. Produce output without identifying numbers in margins of html or LaTeX/pdf output.	
<b>-no-annotate</b>	166
strips output text of editor endnotes <sup>*1</sup> denoted by asterisk or dagger/plus sign	
<b>-no-asterisk</b>	167
strips output text of editor endnotes <sup>*2</sup> denoted by asterisk sign	
<b>-no-dagger</b>	168
strips output text of editor endnotes <sup>+1</sup> denoted by dagger/plus sign	

---

<sup>\*1</sup>square brackets

<sup>\*2</sup>square brackets

<sup>+1</sup>square brackets

<b>7. database commands</b>	169
dbi - database interface	170
-D or -pgsql set for postgresql -d or -sqlite default set for sqlite -d is modifiable with -db=[database type (pgsql or sqlite)]	171
<b>-pg -v -createall</b>	172
initial step, creates required relations (tables, indexes) in existing postgresql database (a database should be created manually and given the same name as working directory, as requested) (rb.dbi) [ -dv -createall sqlite equivalent] it may be necessary to run sisu -Dv -createdb initially NOTE: at the present time for postgresql it may be necessary to manually create the database. The command would be 'createdb [database name]' where database name would be SiSU_[present working directory name (without path)]. Please use only alphanumerics and underscores.	
<b>-pg -v -import</b>	173
[filename/wildcard] imports data specified to postgresql db (rb.dbi) [ -dv -import sqlite equivalent]	
<b>-pg -v -update</b>	174
[filename/wildcard] updates/imports specified data to postgresql db (rb.dbi) [ -dv -update sqlite equivalent]	
<b>-pg -remove</b>	175
[filename/wildcard] removes specified data to postgresql db (rb.dbi) [ -d -remove sqlite equivalent]	
<b>-pg -dropall</b>	176
kills data" and drops (postgresql or sqlite) db, tables & indexes [ -d -dropall sqlite equivalent]	
The -v is for verbose output.	177

## 8. Shortcuts, Shorthand for multiple flags 178

### **-update [filename/wildcard]** 179

Checks existing file output and runs the flags required to update this output. This means that if only html and pdf output was requested on previous runs, only the -hp files will be applied, and only these will be generated this time, together with the summary. This can be very convenient, if you offer different outputs of different files, and just want to do the same again.

### **-0 to -5 [filename or wildcard]** 180

Default shorthand mappings (note that the defaults can be changed/configured in the sisurc.yml file):

#### **-0** 181

-mNhwpAobxXyYv [this is the default action run when no options are give, i.e. on ‘sisu [filename]’]

#### **-1** 182

-mhewpy

#### **-2** 183

-mhewpaoy

#### **-3** 184

-mhewpAobxXyY

#### **-4** 185

-mhewpAobxXDyY -import

#### **-5** 186

-mhewpAobxXDyY -update

add -v for verbose mode and -c for color, e.g. sisu -2vc [filename or wildcard] 187

consider -u for appended url info or -v for verbose output 188

### 8.1 Command Line with Flags - Batch Processing 189

In the data directory run sisu -mh filename or wildcard eg. “sisu -h cisg.sst” or “sisu -h \*.{sst,ssm}” to produce html version of all documents. 190

Running sisu (alone without any flags, filenames or wildcards) brings up the interactive help, as does any sisu command that is not recognised. Enter to escape. 191

<b>9. Introduction to SiSU Markup<sup>10</sup></b>	192
<b>9.1 Summary</b>	193
SiSU source documents are plaintext (UTF-8) <sup>11</sup> files	194
All paragraphs are separated by an empty line.	195
Markup is comprised of:	196
Γ at the top of a document, the document header made up of semantic meta-data about the document and if desired additional processing instructions (such an instruction to automatically number headings from a particular level down)	197
Γ followed by the prepared substantive text of which the most important single characteristic is the markup of different heading levels, which define the primary outline of the document structure. Markup of substantive text includes:	198
Γ heading levels defines document structure	199
Γ text basic attributes, italics, bold etc.	200
Γ grouped text (objects), which are to be treated differently, such as code blocks or poems.	201
Γ footnotes/endnotes	202
Γ linked text and images	203
Γ paragraph actions, such as indent, bulleted, numbered-lists, etc.	204
Some interactive help on markup is available, by typing <code>sisu</code> and selecting <code>markup</code> or <code>sisu -help markup</code>	205
To check the markup in a file:	206
<code>sisu -identify [filename].sst</code>	207
For brief descriptive summary of markup history	208
<code>sisu -query-history</code>	209
or if for a particular version:	210
<code>sisu -query-0.38</code>	211

## 9.2 Markup Examples 212

### 9.2.1 Online 213

Online markup examples are available together with the respective outputs produced from <http://www.jus.uio.no/sisu/SiSU/examples.html> OR FROM [http://www.jus.uio.no/sisu/sisu\\_examples/](http://www.jus.uio.no/sisu/sisu_examples/) 214

There is of course this document, which provides a cursory overview of `sisu` markup and 215

<sup>10</sup>From sometime after SiSU 0.58 it should be possible to describe SiSU markup using SiSU, which though not an original design goal is useful.

<sup>11</sup>files should be prepared using UTF-8 character encoding

the respective output produced: `<http://www.jus.uio.no/sisu/sisu_markup/>`

an alternative presentation of markup syntax: `/usr/share/doc/sisu/on_markup.txt.gz` 216

### 9.2.2 Installed 217

With **SiSU** installed sample skins may be found in: `/usr/share/doc/sisu/markup-samples` (or equivalent directory) and if `sisu-markup-samples` is installed also under: `/usr/share/doc/sisu/markup-samples-non-free` 218

## 10. Markup of Headers

219

Headers contain either: semantic meta-data about a document, which can be used by any output module of the program, or; processing instructions.

220

Note: the first line of a document may include information on the markup version used in the form of a comment. Comments are a percentage mark at the start of a paragraph (and as the first character in a line of text) followed by a space and the comment:

221

```
% this would be a comment
```

222

### 10.1 Sample Header

223

This current document is loaded by a master document that has a header similar to this one:

224

```
% SiSU master 2.0
```

225

```
@title: SiSU
:subtitle: Manual
```

```
@creator: :author: Amissah, Ralph
```

```
@rights: Copyright (C) Ralph Amissah 2007, part of SiSU documentation,
License GPL 3
```

```
@classify:
:type: information
:topic_register: SiSU>manual;electronic documents:SiSU>manual
:subject: ebook, epubublishing, electronic book, electronic publishing,
electronic document, electronic citation, data structure,
citation systems, search
```

```
% used.by: manual
```

```
@date:
:published: 2008-05-22
:created: 2002-08-28
:issued: 2002-08-28
:available: 2002-08-28
:modified: 2010-03-03
```

```
@make:
:num_top: 1
:breaks: new=C; break=1
:skin: skin_sisu_manual
:bold: /Gnu|Debian|Ruby|SiSU/
:manpage: name=sisu - documents: markup, structuring, publishing in
multiple standard formats, and search;
```

```

        synopsis=sisu [-abcDdeFhIiMmNnopqRrSsTtUuVvwXxYyZz0-9]
[filename/wildcard ]
    . sisu [-Ddcv] [instruction]
    . sisu [-CcFLSVvW]
    . sisu --v2 [operations]
    . sisu --v3 [operations]

```

**@links:**

```

{ SiSU Homepage }http://www.sisudoc.org/
{ SiSU Manual }http://www.sisudoc.org/sisu/sisu_manual/
{ Book Samples & Markup Examples }http://www.jus.uio.no/sisu/SiSU/examples.html
{ SiSU Download }http://www.jus.uio.no/sisu/SiSU/download.html
{ SiSU Changelog }http://www.jus.uio.no/sisu/SiSU/changelog.html
{ SiSU Git repo }http://git.sisudoc.org/?p=code/sisu.git;a=summary
{ SiSU List Archives }http://lists.sisudoc.org/pipermail/sisu/
{ SiSU @ Debian }http://packages.qa.debian.org/s/sisu.html
{ SiSU Project @ Debian
}http://qa.debian.org/developer.php?login=sisu@lists.sisudoc.org
{ SiSU @ Wikipedia }http://en.wikipedia.org/wiki/SiSU

```

**10.2 Available Headers**

226

Header tags appear at the beginning of a document and provide meta information on the document (such as the Dublin Core), or information as to how the document as a whole is to be processed. All header instructions take the form @headername: or on the next line and indented by once space :subheadername: All Dublin Core meta tags are available

227

**@identifier:** information or instructions

228

where the “identifier” is a tag recognised by the program, and the “information” or “instructions” belong to the tag/identifier specified

229

Note: a header where used should only be used once; all headers apart from @title: are optional; the @structure: header is used to describe document structure, and can be useful to know.

230

This is a sample header

231

```

% SiSU 2.0 [declared file-type identifier with markup version]
@title: [title text] [this header is the only one that is mandatory]
    :subtitle: [subtitle if any]
    :language: English
@creator:
    :author: [Lastname, First names]
    :illustrator: [Lastname, First names]
    :translator: [Lastname, First names]
    :prepared_by: [Lastname, First names]
@date:

```

232

233

234

235

234

235

235

235

235

235

```

:published: [year or yyyy-mm-dd]
:created: [year or yyyy-mm-dd]
:issued: [year or yyyy-mm-dd]
:available: [year or yyyy-mm-dd]
:modified: [year or yyyy-mm-dd]
:valid: [year or yyyy-mm-dd]
:added_to_site: [year or yyyy-mm-dd]
:translated: [year or yyyy-mm-dd]

@rights: 236
:copyright: Copyright (C) [Year and Holder]
:license: [Use License granted]
:text: [Year and Holder]
:translation: [Name, Year]
:illustrations: [Name, Year]

@classify: 237
:topic_register: SiSU:markup sample:book;book:novel:fantasy
:type:
:subject:
:description:
:keywords:
:abstract:
:isbn: [ISBN]
:loc: [Library of Congress classification]
:dewey: [Dewey classification]
:pg: [Project Gutenberg text number]

@links: { SiSU }http://www.sisudoc.org 238
        { FSF }http://www.fsf.org

@make: 239
:skin: skin_name [skins change default settings related to the appearance
of documents generated]
:num_top: 1
:headings: [text to match for each level
(e.g. PART; Chapter; Section; Article; or another: none;
BOOK|FIRST|SECOND; none; CHAPTER;)]
:breaks: new=:C; break=1
:promo: sisu, ruby, sisu_search_libre, open_society
:bold: [regular expression of words/phrases to be made bold]
:italics: [regular expression of words/phrases to italicise]

@original: 240
:language: [language]

@notes: 241
:comment:
:prefix: [prefix is placed just after table of contents]

```

## 11. Markup of Substantive Text 242

### 11.1 Heading Levels 243

Heading levels are :A~ ,:B~ ,:C~ ,1~ ,2~ ,3~ ... :A - :C being part / section headings, followed by other heading levels, and 1 -6 being headings followed by substantive text or sub-headings. :A~ usually the title :A~? conditional level 1 heading (used where a stand-alone document may be imported into another) 244

**:A~ [heading text]** Top level heading [this usually has similar content to the title @title: ] NOTE: the heading levels described here are in 0.38 notation, see heading 245

**:B~ [heading text]** Second level heading [this is a heading level divider] 246

**:C~ [heading text]** Third level heading [this is a heading level divider] 247

**1~ [heading text]** Top level heading preceding substantive text of document or sub-heading 2, the heading level that would normally be marked 1. or 2. or 3. etc. in a document, and the level on which sisu by default would break html output into named segments, names are provided automatically if none are given (a number), otherwise takes the form 1~my\_filename\_for\_this\_segment 248

**2~ [heading text]** Second level heading preceding substantive text of document or sub-heading 3 , the heading level that would normally be marked 1.1 or 1.2 or 1.3 or 2.1 etc. in a document. 249

**3~ [heading text]** Third level heading preceding substantive text of document, that would normally be marked 1.1.1 or 1.1.2 or 1.2.1 or 2.1.1 etc. in a document 250

1~filename level 1 heading, 251

% the primary division such as Chapter that is followed by substantive text, and may be further subdivided (this is the level on which by default html segments are made)

### 11.2 Font Attributes 252

markup example: 253

```
normal text, {emphasis}, !{bold text}!, /{italics}/, _{underscore}_,
"{citation}",
^ {superscript}^ , ,{subscript},, +{inserted text}+, -{strikethrough}-,
#{monospace}#
```

254

normal text

*{emphasis}*\* [note: can be configured to be represented by bold, italics or underscore]

**!{bold text}!**

`/{italics}/`

`_{underscore}_`

`"{citation}"`

`^ {superscript}^`

`,{subscript},`

`+{inserted text}+`

`-{strikethrough}-`

`#{monospace}#`

**resulting output:**

normal text, **emphasis** , **bold text** , *italics*, underline, “citation”, superscript, subscript, inserted text, ~~strikethrough~~, monospace

normal text 257

**emphasis** [note: can be configured to be represented by bold, italics or underscore] 258

**bold text** 259

*italics* 260

underline 261

“citation” 262

superscript 263

subscript 264

inserted text 265

~~strikethrough~~ 266

monospace 267

**11.3 Indentation and bullets** 268

**markup example:** 269

ordinary paragraph 270

`_1` indent paragraph one step

`_2` indent paragraph two steps

`_9` indent paragraph nine steps

**resulting output:** 271



asterisks if required }~ continues	
normal text ~{** another unnumbered asterisk footnote/endnote }~ continues	
<b>resulting output:</b>	297
normal text * continues	298
normal text ** continues	299
<b>markup example:</b>	300
normal text ~[* editors notes, numbered asterisk footnote/endnote series ]~ continues	301
normal text ~[+ editors notes, numbered asterisk footnote/endnote series ]~ continues	
<b>resulting output:</b>	302
normal text * <sup>3</sup> continues	303
normal text + <sup>2</sup> continues	304
<b>Alternative endnote pair notation for footnotes/endnotes:</b>	305
% note the endnote marker "~^ "	306
normal text~^ continues	
^ ~ endnote text following the paragraph in which the marker occurs the standard and pair notation cannot be mixed in the same document	307
<b>11.5 Links</b>	308
<b>11.5.1 Naked URLs within text, dealing with urls</b>	309
urls found within text are marked up automatically. A url within text is automatically hyperlinked to itself and by default decorated with angled braces, unless they are contained within a code block (in which case they are passed as normal text), or escaped by a preceding underscore (in which case the decoration is omitted).	310
<b>markup example:</b>	311
normal text <code>http://www.sisudoc.org/</code> continues	312
<b>resulting output:</b>	313
normal text <code>&lt;http://www.sisudoc.org/&gt;</code> continues	314
An escaped url without decoration	315
*unnumbered asterisk footnote/endnote, insert multiple asterisks if required	
**another unnumbered asterisk footnote/endnote	
* <sup>3</sup> editors notes, numbered asterisk footnote/endnote series	
+ <sup>2</sup> editors notes, numbered asterisk footnote/endnote series	

**markup example:** 316

```
normal text _http://www.sisudoc.org/ continues 317
```

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free
```

**resulting output:** 318

```
normal text http://www.sisudoc.org/ continues 319
```

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free 320
```

where a code block is used there is neither decoration nor hyperlinking, code blocks are 321  
discussed later in this document

**resulting output:** 322

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free 323
```

```
deb-src http://www.jus.uio.no/sisu/archive unstable main non-free
```

## 11.5.2 Linking Text 324

To link text or an image to a url the markup is as follows 325

**markup example:** 326

```
about { SiSU }http://url.org markup 327
```

**resulting output:** 328

```
about <SiSU> markup 329
```

A shortcut notation is available so the url link may also be provided automatically as a 330  
footnote

**markup example:** 331

```
about {~ SiSU }http://url.org markup 332
```

**resulting output:** 333

```
about <SiSU>14 markup 334
```

Internal document links to a tagged location, including an ocn 335

**markup example:** 336

```
about { text links }#link_text 337
```

**resulting output:** 338

```
about <text links> 339
```

Shared document collection link 340

**markup example:** 341

```
about { SiSU book markup examples }:SiSU/examples.html 342
```

**resulting output:** 343

```
about <SiSU book markup examples> 344
```

---

<sup>14</sup><<http://www.sisudoc.org/>>

### 11.5.3 Linking Images

345

#### markup example:

346

```
{ tux.png 64x80 }image
```

347

```
% various url linked images
```

```
{tux.png 64x80 "a better way" }http://www.sisudoc.org/  
{GnuDebianLinuxRubyBetterWay.png 100x101 "Way Better - with Gnu/Linux, Debian  
and Ruby" }http://www.sisudoc.org/  
{^^ ruby_logo.png "Ruby" }http://www.ruby-lang.org/en/
```

#### resulting output:

348



349



350



351



15

352

#### linked url footnote shortcut

353

```
{^^ [text to link] }http://url.org  
% maps to: { [text to link] }http://url.org ~{ http://url.org }~
```

354

---

<sup>15</sup> <<http://www.ruby-lang.org/en/>>

% which produces hyper-linked text within a document/paragraph, with an endnote providing the url for the text location used in the hyperlink

text marker 355

note at a heading level the same is automatically achieved by providing names to headings 1, 2 and 3 i.e. 2~[name] and 3~[name] or in the case of auto-heading numbering, without further intervention. 356

## 11.6 Grouped Text 357

### 11.6.1 Tables 358

Tables may be prepared in two either of two forms 359

**markup example:** 360

```
table{ c3; 40; 30; 30; 361
```

```
This is a table
this would become column two of row one
column three of row one is here
```

```
And here begins another row
column two of row two
column three of row two, and so on
```

```
}table
```

**resulting output:** 362

```
This is a table                this would become column two of row one   column three of row one is here
And here begins another row   column two of row two                       column three of row two, and so on 363
```

a second form may be easier to work with in cases where there is not much information in each column 364

**markup example:** <sup>16</sup> 365

```
!_ Table 3.1: Contributors to Wikipedia, January 2001 - June 2005 366
```

```
{table~h 24; 12; 12; 12; 12; 12; 12;}
|Jan. 2001|Jan. 2002|Jan. 2003|Jan.
2004|July 2004|June 2006
Contributors* | 10| 472| 2,188| 9,653| 25,011| 48,721
Active
contributors** | 9| 212| 846| 3,228| 8,442| 16,945
```

---

<sup>16</sup>Table from the Wealth of Networks by Yochai Benkler  
[http://www.jus.uio.no/sisu/the\\_wealth\\_of\\_networks.yochai\\_benkler](http://www.jus.uio.no/sisu/the_wealth_of_networks.yochai_benkler)

Very active contributors\*\*\* | 0| 31| 190| 692| 1,639| 3,016  
 No. of English language articles| 25| 16,000| 101,000| 190,000| 320,000| 630,000  
 No. of articles, all languages | 25| 19,000| 138,000| 490,000| 862,000|1,600,000

Contributed at least ten times; \* at least 5 times in last month; \* more than 100 times in last month.

resulting output:

**Table 3.1: Contributors to Wikipedia, January 2001 - June 2005**

	Jan. 2001	Jan. 2002	Jan. 2003	Jan. 2004	July 2004	June 2006
Contributors*	10	472	2,188	9,653	25,011	48,721
Active contributors**	9	212	846	3,228	8,442	16,945
Very active contributors***	0	31	190	692	1,639	3,016
No. of English language articles	25	16,000	101,000	190,000	320,000	630,000
No. of articles, all languages	25	19,000	138,000	490,000	862,000	1,600,000

Contributed at least ten times; \* at least 5 times in last month; \* more than 100 times in last month.

### 11.6.2 Poem

basic markup:

```
poem{
```

```
    Your poem here
```

```
}poem
```

Each verse in a poem is given an object number.

markup example:

```
poem{
```

```
        'Fury said to a
          mouse, That he
            met in the
              house,
"Let us
  both go to
    law: I will
      prosecute
```

```

        YOU. --Come,
            I'll take no
            denial; We
            must have a
        trial: For
        really this
        morning I've
        nothing
        to do."
        Said the
        mouse to the
        cur, "Such
        a trial,
        dear Sir,
            With
            no jury
            or judge,
        would be
        wasting
        our
        breath."
        "I'll be
        judge, I'll
        be jury,"
            Said
        cunning
        old Fury:
        "I'll
        try the
            whole
            cause,
            and
            condemn
        you
        to
        death.'"
    
```

}poem

resulting output:

```

        'Fury said to a
        mouse, That he
        met in the
        house,
        "Let us
        both go to
        law: I will
        prosecute
    
```

376

377

YOU. –Come,  
 I'll take no  
 denial; We  
 must have a  
 trial: For  
 really this  
 morning I've  
 nothing  
 to do.”  
 Said the  
 mouse to the  
 cur, ”Such  
 a trial,  
 dear Sir,  
 With  
 no jury  
 or judge,  
 would be  
 wasting  
 our  
 breath.”  
 ”I'll be  
 judge, I'll  
 be jury,”  
 Said  
 cunning  
 old Fury:  
 ”I'll  
 try the  
 whole  
 cause,  
 and  
 condemn  
 you  
 to  
 death.”

**11.6.3 Group** 378

**basic markup:** 379

`group{` 380

Your grouped text here

`}group`

A group is treated as an object and given a single object number.

**markup example:** 381

`group{` 382

'Fury said to a  
        mouse, That he  
        met in the  
        house,  
"Let us  
both go to  
law: I will  
prosecute  
YOU. --Come,  
        I'll take no  
        denial; We  
        must have a  
        trial: For  
        really this  
        morning I've  
nothing  
to do."  
Said the  
mouse to the  
cur, "Such  
a trial,  
dear Sir,  
        With  
        no jury  
or judge,  
would be  
wasting  
our  
breath."  
"I'll be  
judge, I'll  
be jury,"  
        Said  
        cunning  
        old Fury:  
"I'll  
try the  
        whole  
        cause,  
        and  
        condemn  
you  
to  
death."'

}group

**resulting output:**

383

‘Fury said to a  
mouse, That he  
met in the  
house,  
”Let us  
both go to  
law: I will  
prosecute  
YOU. –Come,  
I’ll take no  
denial; We  
must have a  
trial: For  
really this  
morning I’ve  
nothing  
to do.”  
Said the  
mouse to the  
cur, ”Such  
a trial,  
dear Sir,  
With  
no jury  
or judge,  
would be  
wasting  
our  
breath.”  
”I’ll be  
judge, I’ll  
be jury,”  
Said  
cunning  
old Fury:  
”I’ll  
try the  
whole  
cause,  
and  
condemn  
you  
to  
death.”’

384

## 11.6.4 Code

385

Code tags `{ ... }` (used as with other group tags described above) are used to escape regular sisu markup, and have been used extensively within this document to provide examples of SiSU markup. You cannot however use code tags to escape code tags. They are however used in the same way as group or poem tags.

386

A code-block is treated as an object and given a single object number. [an option to number each line of code may be considered at some later time]

387

**use of code tags instead of poem compared, resulting output:**

388

```

        'Fury said to a
          mouse, That he
            met in the
              house,
"Let us
  both go to
    law: I will
      prosecute
        YOU. --Come,
          I'll take no
            denial; We
              must have a
                trial: For
                  really this
                    morning I've
                      nothing
to do."
  Said the
    mouse to the
      cur, "Such
        a trial,
          dear Sir,
            With
              no jury
                or judge,
                  would be
                    wasting
our
  breath."
    "I'll be
      judge, I'll
        be jury,"
          Said
            cunning
              old Fury:
                "I'll
                  try the

```

389

```

        whole
        cause,
        and
        condemn
        you
        to
        death."

```

From SiSU 2.7.7 on you can number codeblocks by placing a hash after the opening code tag `code{#` as demonstrated here: 390

```

1 |           'Fury said to a 391
2 |           mouse, That he
3 |           met in the
4 |           house,
5 |           "Let us
6 |           both go to
7 |           law: I will
8 |           prosecute
9 |           YOU. --Come,
10 |           I'll take no
11 |           denial; We
12 |           must have a
13 |           trial: For
14 |           really this
15 |           morning I've
16 |           nothing
17 |           to do."
18 |           Said the
19 |           mouse to the
20 |           cur, "Such
21 |           a trial,
22 |           dear Sir,
23 |           With
24 |           no jury
25 |           or judge,
26 |           would be
27 |           wasting
28 |           our
29 |           breath."
30 |           "I'll be
31 |           judge, I'll
32 |           be jury,"
33 |           Said
34 |           cunning
35 |           old Fury:
36 |           "I'll
37 |           try the
38 |           whole

```

```

39 |           cause,
40 |             and
41 |           condemn
42 |           you
43 |           to
44 |           death."'
```

## 11.7 Book index

392

To make an index append to paragraph the book index term relates to it, using an equal sign and curly braces. 393

Currently two levels are provided, a main term and if needed a sub-term. Sub-terms are separated from the main term by a colon. 394

```

Paragraph containing main term and sub-term.
={Main term:sub-term}
```

395

The index syntax starts on a new line, but there should not be an empty line between paragraph and index markup. 396

The structure of the resulting index would be: 397

```

Main term, 1
  sub-term, 1
```

398

Several terms may relate to a paragraph, they are separated by a semicolon. If the term refers to more than one paragraph, indicate the number of paragraphs. 399

```

Paragraph containing main term, second term and sub-term.
={first term; second term: sub-term}
```

400

The structure of the resulting index would be: 401

```

First term, 1,
Second term, 1,
  sub-term, 1
```

402

If multiple sub-terms appear under one paragraph, they are separated under the main term heading from each other by a pipe symbol. 403

```

Paragraph containing main term, second term and sub-term.
={Main term:sub-term+1|second sub-term}
```

404

```

A paragraph that continues discussion of the first sub-term
```

The plus one in the example provided indicates the first sub-term spans one additional paragraph. The logical structure of the resulting index would be: 405

```

Main term, 1,
  sub-term, 1-3,
  second sub-term, 1,
```

406

## 12. Composite documents markup

407

It is possible to build a document by creating a master document that requires other documents. The documents required may be complete documents that could be generated independently, or they could be markup snippets, prepared so as to be easily available to be placed within another text. If the calling document is a master document (built from other documents), it should be named with the suffix **.ssm**. Within this document you would provide information on the other documents that should be included within the text. These may be other documents that would be processed in a regular way, or markup bits prepared only for inclusion within a master document **.sst** regular markup file, or **.ssi** (insert/information). A secondary file of the composite document is built prior to processing with the same prefix and the suffix **.\_sst**.

408

basic markup for importing a document into a master document

409

```
<< filename1.sst
```

410

```
<< filename2.ssi
```

The form described above should be relied on. Within the Vim editor it results in the text thus linked becoming hyperlinked to the document it is calling in which is convenient for editing. Alternative markup for importation of documents under consideration, and occasionally supported have been.

411

```
<< filename.ssi
```

412

```
<<{filename.ssi}
```

% using textlink alternatives

```
<< |filename.ssi|@|^ |
```

## Markup Syntax History

413

### 13. Notes related to Files-types and Markup Syntax 414

2.0 introduced new headers and is therefore incompatible with 1.0 though otherwise the same with the addition of a couple of tags (i.e. a superset) 415

0.38 is substantially current for version 1.0 416

depreciated 0.16 supported, though file names were changed at 0.37 417

```
Γ sisu -query=[sisu version [0.38] or 'history'] 418
```

provides a short history of changes to **SiSU** markup 419

**SiSU 2.0** (2010-03-06:09/6) same as 1.0, apart from the changing of headers and the addition of a monospace tag related headers now grouped, e.g. 420

```
@title: 421
:subtitle:
```

```
@creator:
:author:
:translator:
:illustrator:
```

```
@rights:
:text:
:illustrations:
```

see document markup samples, and `sisu -help headers` 422

the monospace tag takes the form of a hash `#` 423

```
#{ this enclosed text would be monospaced }# 424
```

**1.0** (2009-12-19:50/6) same as 0.69 425

**0.69** (2008-09-16:37/2) (same as 1.0) and as previous (0.57) with the addition of book index tags 426

```
/^ ={.+?}$ / 427
```

e.g. appended to a paragraph, on a new-line (without a blank line in between) logical structure produced assuming this is the first text “object” 428

```
={GNU/Linux community distribution:Debian+2|Fedora|Gentoo;Free Software
Foundation+5} 429
```

```
Free Software Foundation, 1-6 430
```

```
GNU/Linux community distribution, 1
Debian, 1-3
Fedora, 1
Gentoo,
```

**0.66** (2008-02-24:07/7) same as previous, adds semantic tags, [experimental and not-used] 431

```
/[:;]{.+?}[:;][a-z+]/ 432
```

**0.57** (2007w34/4) **SiSU** 0.57 is the same as 0.42 with the introduction of some a shortcut 433

to use the headers @title and @creator in the first heading [expanded using the contents of the headers @title: and @author:]

`:A~ @title by @author` 434

**0.52** (2007w14/6) declared document type identifier at start of text/document: 435

`SiSU 0.52` 436

or, backward compatible using the comment marker: 437

`% SiSU 0.38` 438

variations include ‘**SiSU** (text|master|insert) [version]’ and ‘sisu-[version]’ 439

**0.51** (2007w13/6) skins changed (simplified), markup unchanged 440

**0.42** (2006w27/4) \* (asterisk) type endnotes, used e.g. in relation to author 441

**SiSU 0.42** is the same as 0.38 with the introduction of some additional endnote types, 442

Introduces some variations on endnotes, in particular the use of the asterisk 443

`~{* for example for describing an author }~` and `~{** for describing a second author }~` 444

\* for example for describing an author 445

\*\* for describing a second author 446

and 447

`~[* my note ]~` or `~[+ another note ]~` 448

which numerically increments an asterisk and plus respectively 449

\*1 my note +1 another note 450

**0.38** (2006w15/7) introduced new/alternative notation for headers, e.g. @title: (instead of 0~title), and accompanying document structure markup, :A,:B,:C,1,2,3 (maps to previous 1,2,3,4,5,6) 451

**SiSU 0.38** introduced alternative experimental header and heading/structure markers, 452

`@headername: and headers :A~ :B~ :C~ 1~ 2~ 3~` 453

as the equivalent of: 454

`0~headername and headers 1~ 2~ 3~ 4~ 5~ 6~` 455

The internal document markup of **SiSU 0.16** remains valid and standard Though note that **SiSU 0.37** introduced a new file naming convention 456

**SiSU** has in effect two sets of levels to be considered, using 0.38 notation A-C headings/levels, pre-ordinary paragraphs /pre-substantive text, and 1-3 headings/levels, levels which are followed by ordinary text. This may be conceptualised as levels A,B,C, 1,2,3, and using such letter number notation, in effect: A must exist, optional B and C may follow in sequence (not strict) 1 must exist, optional 2 and 3 may follow in sequence i.e. there are two independent heading level sequences A,B,C and 1,2,3 (using the 0.16 457

standard notation 1,2,3 and 4,5,6) on the positive side: the 0.38 A,B,C,1,2,3 alternative makes explicit an aspect of structuring documents in **SiSU** that is not otherwise obvious to the newcomer (though it appears more complicated, is more in your face and likely to be understood fairly quickly); the substantive text follows levels 1,2,3 and it is ‘nice’ to do most work in those levels

**0.37** (2006w09/7) introduced new file naming convention, .sst (text), .ssm (master), .ssi (insert), markup syntax unchanged 458

**SiSU** 0.37 introduced new file naming convention, using the file extensions .sst .ssm and .ssi to replace .s1 .s2 .s3 .r1 .r2 .r3 and .si 459

this is captured by the following file ‘rename’ instruction: 460

```
rename 's/_s[123]$/_sst/' *.s{1,2,3} 461
rename 's/_r[123]$/_ssm/' *.r{1,2,3}
rename 's/_si$/_ssi/' *.si
```

The internal document markup remains unchanged, from **SiSU** 0.16 462

**0.35** (2005w52/3) sisupod, zipped content file introduced 463

**0.23** (2005w36/2) utf-8 for markup file 464

**0.22** (2005w35/3) image dimensions may be omitted if rmagick is available to be relied upon 465

**0.20.4** (2005w33/4) header 0~links 466

**0.16** (2005w25/2) substantial changes introduced to make markup cleaner, header 0~title type, and headings [1-6]~ introduced, also percentage sign (%) at start of a text line as comment marker 467

**SiSU** 0.16 (0.15 development branch) introduced the use of 468

the header 0~ and headings/structure 1~ 2~ 3~ 4~ 5~ 6~ 469

in place of the 0.1 header, heading/structure notation 470

**SiSU** 0.1 headers and headings structure represented by header 0{~ and headings/structure 1{ 2{ 3{ 4{~ 5{ 6{ 471

## 14. SiSU filetypes 472

SiSU has plaintext and binary filetypes, and can process either type of document. 473

### 14.1 .sst .ssm .ssi marked up plain text 474

SiSU documents are prepared as plain-text (utf-8) files with SiSU markup. They may 475  
make reference to and contain images (for example), which are stored in the directory  
beneath them `._sisu/image`. SiSU plaintext markup files are of three types that may be  
distinguished by the file extension used: regular text `.sst`; master documents, composite  
documents that incorporate other text, which can be any regular text or text insert; and  
inserts the contents of which are like regular text except these are marked `.ssi` and are not  
processed.

SiSU processing can be done directly against a `sisu` documents; which may be located 476  
locally or on a remote server for which a url is provided.

SiSU source markup can be shared with the command: 477

```
sisu -s [filename] 478
```

#### 14.1.1 `sisu` text - regular files (`.sst`) 479

The most common form of document in SiSU, see the section on SiSU markup. 480

```
<http://www.sisudoc.org/sisu/sisu\_markup> 481
```

```
<http://www.sisudoc.org/sisu/sisu\_manual> 482
```

#### 14.1.2 `sisu` master files (`.ssm`) 483

Composite documents which incorporate other SiSU documents which may be either 484  
regular SiSU text `.sst` which may be generated independently, or inserts prepared solely  
for the purpose of being incorporated into one or more master documents.

The mechanism by which master files incorporate other documents is described as one of 485  
the headings under under SiSU markup in the SiSU manual.

Note: Master documents may be prepared in a similar way to regular documents, and 486  
processing will occur normally if a `.sst` file is renamed `.ssm` without requiring any other  
documents; the `.ssm` marker flags that the document may contain other documents.

Note: a secondary file of the composite document is built prior to processing with the 487  
same prefix and the suffix `._sst` <sup>17</sup>

```
<http://www.sisudoc.org/sisu/sisu\_markup> 488
```

```
<http://www.sisudoc.org/sisu/sisu\_manual> 489
```

---

<sup>17</sup>.`ssc` (for composite) is under consideration but `._sst` makes clear that this is not a regular file to be  
worked on, and thus less likely that people will have “accidents”, working on a `.ssc` file that is  
overwritten by subsequent processing. It may be however that when the resulting file is shared `.ssc` is an  
appropriate suffix to use.

### 14.1.3 sisu insert files (.ssi)

490

Inserts are documents prepared solely for the purpose of being incorporated into one or more master documents. They resemble regular **SiSU** text files except they are ignored by the **SiSU** processor. Making a file a .ssi file is a quick and convenient way of flagging that it is not intended that the file should be processed on its own.

491

### 14.2 sisupod, zipped binary container (sisupod.zip, .ssp)

492

A sisupod is a zipped **SiSU** text file or set of **SiSU** text files and any associated images that they contain (this will be extended to include sound and multimedia-files)

493

**SiSU** plaintext files rely on a recognised directory structure to find contents such as images associated with documents, but all images for example for all documents contained in a directory are located in the sub-directory `_sisu/image`. Without the ability to create a sisupod it can be inconvenient to manually identify all other files associated with a document. A sisupod automatically bundles all associated files with the document that is turned into a pod.

494

The structure of the sisupod is such that it may for example contain a single document and its associated images; a master document and its associated documents and anything else; or the zipped contents of a whole directory of prepared **SiSU** documents.

495

The command to create a sisupod is:

496

```
sisu -S [filename]
```

497

Alternatively, make a pod of the contents of a whole directory:

498

```
sisu -S
```

499

**SiSU** processing can be done directly against a sisupod; which may be located locally or on a remote server for which a url is provided.

500

<[http://www.sisudoc.org/sisu/sisu\\_commands](http://www.sisudoc.org/sisu/sisu_commands)>

501

<[http://www.sisudoc.org/sisu/sisu\\_manual](http://www.sisudoc.org/sisu/sisu_manual)>

502

## 15. Experimental Alternative Input Representations 503

### 15.1 Alternative XML 504

SiSU offers alternative XML input representations of documents as a proof of concept, experimental feature. They are however not strictly maintained, and incomplete and should be handled with care. 505

#### convert from sst to simple xml representations (sax, dom and node): 506

sisu -to-sax [filename/wildcard] or sisu -to-sxs [filename/wildcard] 507

sisu -to-dom [filename/wildcard] or sisu -to-sxd [filename/wildcard] 508

sisu -to-node [filename/wildcard] or sisu -to-sxn [filename/wildcard] 509

#### convert to sst from any sisu xml representation (sax, dom and node): 510

sisu -from-xml2sst [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 511

or the same: 512

sisu -from-sxml [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 513

#### 15.1.1 XML SAX representation 514

To convert from sst to simple xml (sax) representation: 515

sisu -to-sax [filename/wildcard] or sisu -to-sxs [filename/wildcard] 516

To convert from any sisu xml representation back to sst 517

sisu -from-xml2sst [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 518

or the same: 519

sisu -from-sxml [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 520

#### 15.1.2 XML DOM representation 521

To convert from sst to simple xml (dom) representation: 522

sisu -to-dom [filename/wildcard] or sisu -to-sxd [filename/wildcard] 523

To convert from any sisu xml representation back to sst 524

sisu -from-xml2sst [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 525

or the same: 526

sisu -from-sxml [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]] 527

<b>15.1.3 XML Node representation</b>	528
To convert from sst to simple xml (node) representation:	529
sisu -to-node [filename/wildcard] or sisu -to-sxn [filename/wildcard]	530
To convert from any sisu xml representation back to sst	531
sisu -from-xml2sst [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]]	532
or the same:	533
sisu -from-sxml [filename/wildcard [.sxs.xml,.sxd.xml,sxn.xml]]	534

<b>16. Configuration</b>	535
<b>16.1 Determining the Current Configuration</b>	536
Information on the current configuration of <b>SiSU</b> should be available with the help command:	537
<code>sisu -v</code>	538
which is an alias for:	539
<code>sisu --help env</code>	540
Either of these should be executed from within a directory that contains sisu markup source documents.	541
<b>16.2 Configuration files (config.yml)</b>	542
<b>SiSU</b> configuration parameters are adjusted in the configuration file, which can be used to override the defaults set. This includes such things as which directory interim processing should be done in and where the generated output should be placed.	543
The <b>SiSU</b> configuration file is a yaml file, which means indentation is significant.	544
<b>SiSU</b> resource configuration is determined by looking at the following files if they exist:	545
<code>./_sisu/sisurc.yml</code>	546
<code>~/.sisu/sisurc.yml</code>	547
<code>/etc/sisu/sisurc.yml</code>	548
The search is in the order listed, and the first one found is used.	549
In the absence of instructions in any of these it falls back to the internal program defaults.	550
Configuration determines the output and processing directories and the database access details.	551
If <b>SiSU</b> is installed a sample <code>sisurc.yml</code> may be found in <code>/etc/sisu/sisurc.yml</code>	552

<b>17. Skins</b>	553
Skins modify the default appearance of document output on a document, directory, or site wide basis. Skins are looked for in the following locations:	554
./_sisu/skin	555
~/_sisu/skin	556
/etc/sisu/skin	557
<b>Within the skin directory</b> are the following the default sub-directories for document skins:	558
./skin/doc	559
./skin/dir	560
./skin/site	561
A skin is placed in the appropriate directory and the file named skin_[name].rb	562
The skin itself is a ruby file which modifies the default appearances set in the program.	563
<b>17.1 Document Skin</b>	564
Documents take on a document skin, if the header of the document specifies a skin to be used.	565
@skin: skin_united_nations	566
<b>17.2 Directory Skin</b>	567
A directory may be mapped on to a particular skin, so all documents within that directory take on a particular appearance. If a skin exists in the skin/dir with the same name as the document directory, it will automatically be used for each of the documents in that directory, (except where a document specifies the use of another skin, in the skin/doc directory).	568
A personal habit is to place all skins within the doc directory, and symbolic links as needed from the site, or dir directories as required.	569
<b>17.3 Site Skin</b>	570
A site skin, modifies the program default skin.	571
<b>17.4 Sample Skins</b>	572
With SiSU installed sample skins may be found in:	573

- `/etc/sisu/skin/doc` and `/usr/share/doc/sisu/markup-samples/samples/_sisu/skin/-doc` 574
- (or equivalent directory) and if `sisu-markup-samples` is installed also under: 575
- `/usr/share/doc/sisu/markup-samples-non-free/samples/_sisu/skin/doc` 576
- Samples of `list.yml` and `promo.yml` (which are used to create the right column list) may be found in: 577
- `/usr/share/doc/sisu/markup-samples-non-free/samples/_sisu/skin/yml` (or equivalent directory) 578

18. CSS - Cascading Style Sheets (for html, XHTML and XML) 579
- CSS files to modify the appearance of **SiSU** html, XHTML or XML may be placed in the configuration directory: `./_sisu/css` ; `~/_sisu/css` or; `/etc/sisu/css` and these will be copied to the output directories with the command `sisu -CC`. 580
- The basic CSS file for html output is `html.css`, placing a file of that name in directory `_sisu/css` or equivalent will result in the default file of that name being overwritten. 581
- HTML: `html.css` 582
- XML DOM: `dom.css` 583
- XML SAX: `sax.css` 584
- XHTML: `xhtml.css` 585
- The default homepage may use `homepage.css` or `html.css` 586
- Under consideration is to permit the placement of a CSS file with a different name in directory `_sisu/css` directory or equivalent, and change the default CSS file that is looked for in a skin.<sup>18</sup> 587

---

<sup>18</sup>**SiSU** has worked this way in the past, though this was dropped as it was thought the complexity outweighed the flexibility, however, the balance was rather fine and this behaviour could be reinstated.

## 19. Organising Content 588

### 19.1 Directory Structure and Mapping 589

The output directory root can be set in the `sisurc.yml` file. Under the root, subdirectories are made for each directory in which a document set resides. If you have a directory named `poems` or `conventions`, that directory will be created under the output directory root and the output for all documents contained in the directory of a particular name will be generated to subdirectories beneath that directory (`poem` or `conventions`). A document will be placed in a subdirectory of the same name as the document with the filetype identifier stripped (`.sst` `.ssm`) 590

The last part of a directory path, representing the sub-directory in which a document set resides, is the directory name that will be used for the output directory. This has implications for the organisation of document collections as it could make sense to place documents of a particular subject, or type within a directory identifying them. This grouping as suggested could be by subject (`sales_law`, `english_literature`); or just as conveniently by some other classification (X University). The mapping means it is also possible to place in the same output directory documents that are for organisational purposes kept separately, for example documents on a given subject of two different institutions may be kept in two different directories of the same name, under a directory named after each institution, and these would be output to the same output directory. Skins could be associated with each institution on a directory basis and resulting documents will take on the appropriate different appearance. 591

#### 19.1.1 General Directories 592

`./subject_name/` 593

% files stored at this level e.g. `sisu_manual.sst`

`./subject_name/_sisu`

% configuration file e.g. `sisurc.yml`

`./subject_name/_sisu/skin`

% skins in various skin directories `doc`, `dir`, `site`, `yaml`

`./subject_name/_sisu/css`

`./subject_name/_sisu/image`

% images for documents contained in this directory

`./subject_name/_sisu/mm`

### 19.1.2 Remote Directories

594

`./subject_name/`

595

% containing sub\_directories named after the generated files from which they are made

`./subject_name/src`

% contains shared source files text and binary e.g. `sisu_manual.sst` and `sisu_manual.sst.zip`

`./subject_name/_sisu`

% configuration file e.g. `sisurc.yml`

`./subject_name/_sisu/skin`

% skins in various skin directories `doc`, `dir`, `site`, `yml`

`./subject_name/_sisu/css`

`./subject_name/_sisu/image`

% images for documents contained in this directory

`./subject_name/_sisu/mm`

### 19.1.3 Sisupod

596

`./sisupod/`

597

% files stored at this level e.g. `sisu_manual.sst`

`./sisupod/_sisu`

% configuration file e.g. `sisurc.yml`

`./sisupod/_sisu/skin`

% skins in various skin directories `doc`, `dir`, `site`, `yml`

`./sisupod/_sisu/css`

`./sisupod/_sisu/image`

% images for documents contained in this directory

./sisupod/\_sisu/mm

## 19.2 Organising Content

598

## 20. Homepages

599

**SiSU** is about the ability to auto-generate documents. Home pages are regarded as custom built items, and are not created by **SiSU**. More accurately, **SiSU** has a default home page, which will not be appropriate for use with other sites, and the means to provide your own home page instead in one of two ways as part of a site's configuration, these being:

1. through placing your home page and other custom built documents in the subdirectory `_sisu/home/` (this probably being the easier and more convenient option)
2. through providing what you want as the home page in a skin,

Document sets are contained in directories, usually organised by site or subject. Each directory can/should have its own homepage. See the section on directory structure and organisation of content.

### 20.1 Home page and other custom built pages in a sub-directory

604

Custom built pages, including the home page `index.html` may be placed within the configuration directory `_sisu/home/` in any of the locations that is searched for the configuration directory, namely `./_sisu` ; `~/_sisu` ; `/etc/sisu` From there they are copied to the root of the output directory with the command:

```
sisu -CC
```

606

### 20.2 Home page within a skin

607

Skins are described in a separate section, but basically are a file written in the programming language **Ruby** that may be provided to change the defaults that are provided with `sisu` with respect to individual documents, a directories contents or for a site.

If you wish to provide a homepage within a skin the skin should be in the directory `_sisu/skin/dir` and have the name of the directory for which it is to become the home page. Documents in the directory `commercial_law` would have the homepage modified in `skin_commercial_law.rb` ; or the directory `poems` in `skin_poems.rb`

```
class Home
  def homepage
    # place the html content of your homepage here, this will become
index.html
    <<HOME <html>
    <head></head>
    <doc>
    <p>this is my new homepage.</p>
    </doc>
    </html>
    HOME
  end
end
```

610

**21. Markup and Output Examples** 611

**21.1 Markup examples** 612

Current markup examples and document output samples are provided at `<http://www.jus.uio.no/sisu/SiSU/examples.html>` 613

For some documents hardly any markup at all is required at all, other than a header, and an indication that the levels to be taken into account by the program in generating its output are. 614

## 21.2 A few book (and other) examples

615



19

616

### 21.2.1 “Viral Spiral”, David Bollier

617

618

```

<"Viral Spiral", David Bollier>
  <document manifest> 20
  < html, segmented text>
  < html, scroll, document in one>
  < epub>
  < pdf, landscape>
  < pdf, portrait>
  < odf:odt, open document text>
  < xhtml scroll>
  < xml, sax>
  < xml, dom>
  < plain text utf-8>
  < concordance>
  < dcc, document content certificate (digests)>
  <markup source text>
  <markup source (zipped) pod>

```

<sup>19</sup>Reproduced with the kind permission of author and artist Leena Krohn, <http://www.kaapeli.fi/krohn> “Aukio” is from the work “*Sphinx or Robot*” [http://www.jus.uio.no/sisu/sphinx\\_or\\_robot.leena\\_krohn.1996](http://www.jus.uio.no/sisu/sphinx_or_robot.leena_krohn.1996) which is included as a book example in this section, together with another of the author’s works, “*Tainaron*” [http://www.jus.uio.no/sisu/tainaron.leena\\_krohn.1998](http://www.jus.uio.no/sisu/tainaron.leena_krohn.1998)

<sup>20</sup>[http://www.sisudoc.org/sisu/viral\\_spiral.david\\_bollier/sisu\\_manifest.html](http://www.sisudoc.org/sisu/viral_spiral.david_bollier/sisu_manifest.html)

**“The Wealth of Networks”, Yochai Benkler**

619

&lt;”The Wealth of Networks”, Yochai Benkler&gt;

620

<document manifest> <sup>21</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

**“Two Bits”, Christopher Kelty**

621

&lt;”Two Bits”, Christopher Kelty&gt;

622

<document manifest> <sup>22</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

**“Free Culture”, Lawrence Lessig**

623

---

<sup>21</sup> <[http://www.sisudoc.org/sisu/the\\_wealth\\_of\\_networks.yochai\\_benkler/sisu\\_manifest.html](http://www.sisudoc.org/sisu/the_wealth_of_networks.yochai_benkler/sisu_manifest.html)>

<sup>22</sup> <[http://www.sisudoc.org/sisu/two\\_bits.christopher\\_kelty/sisu\\_manifest.html](http://www.sisudoc.org/sisu/two_bits.christopher_kelty/sisu_manifest.html)>

624

<"Free Culture", Lawrence Lessig>  
<document manifest> <sup>23</sup>  
< html, segmented text>  
< html, scroll, document in one>  
< epub>  
< pdf, landscape>  
< pdf, portrait>  
< odf:odt, open document text>  
< xhtml scroll>  
< xml, sax>  
< xml, dom>  
< plain text utf-8>  
< concordance>  
< dcc, document content certificate (digests)>  
<markup source text>  
<markup source (zipped) pod>

## ***"CONTENT", Cory Doctorow***

625

<"CONTENT", Cory Doctorow>  
<document manifest> <sup>24</sup>  
< html, segmented text>  
< html, scroll, document in one>  
< epub>  
< pdf, landscape>  
< pdf, portrait>  
< odf:odt, open document text>  
< xhtml scroll>  
< xml, sax>  
< xml, dom>  
< plain text utf-8>  
< concordance>  
< dcc, document content certificate (digests)>  
<markup source text>  
<markup source (zipped) pod>

626

## ***"Democratizing Innovation", by Eric von Hippel***

627

<"Democratizing Innovation", by Eric von Hippel>  
<document manifest> <sup>25</sup>

628

---

<sup>23</sup> <[http://www.sisudoc.org/sisu/free\\_culture.lawrence\\_lessig/sisu\\_manifest.html](http://www.sisudoc.org/sisu/free_culture.lawrence_lessig/sisu_manifest.html)>

<sup>24</sup> <[http://www.sisudoc.org/sisu/content.cory\\_doctorow/sisu\\_manifest.html](http://www.sisudoc.org/sisu/content.cory_doctorow/sisu_manifest.html)>

<sup>25</sup> <[http://www.sisudoc.org/sisu/democratizing\\_innovation.eric\\_von\\_hippel/sisu\\_manifest.html](http://www.sisudoc.org/sisu/democratizing_innovation.eric_von_hippel/sisu_manifest.html)>

< html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

***“Free as in Freedom: Richard Stallman’s Crusade for Free Software”, by Sam Williams***

629

<”Free as in Freedom: Richard Stallman’s Crusade for Free Software”, by Sam Williams>

630

<document manifest> <sup>26</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

***“Free For All: How Linux and the Free Software Movement Undercut the High Tech Titans”, by Peter Wayner***

631

<”Free For All: How Linux and the Free Software Movement Undercut the High Tech Titans”, by Peter Wayner>

632

---

<sup>26</sup> <[http://www.sisudoc.org/sisu/free\\_as\\_in\\_freedom.richard\\_stallman\\_crusade\\_for\\_free\\_software.sam\\_williams/sisu\\_manifest.html](http://www.sisudoc.org/sisu/free_as_in_freedom.richard_stallman_crusade_for_free_software.sam_williams/sisu_manifest.html)>

<document manifest> <sup>27</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

***“The Cathedral and the Bazaar”, by Eric S. Raymond***

633

<“The Cathedral and the Bazaar”, by Eric S. Raymond>

634

<document manifest> <sup>28</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

***“Down and out in the Magic Kingdom”, Cory Doctorow***

635

<“Down and out in the Magic Kingdom”, Cory Doctorow>

636

<document manifest> <sup>29</sup>

<sup>27</sup> <[http://www.sisudoc.org/sisu/free\\_for\\_all.peter\\_wayner/sisu\\_manifest.html](http://www.sisudoc.org/sisu/free_for_all.peter_wayner/sisu_manifest.html)>

<sup>28</sup> <[http://www.sisudoc.org/sisu/the\\_cathedral\\_and\\_the\\_bazaar.eric\\_s\\_raymond/sisu\\_manifest.html](http://www.sisudoc.org/sisu/the_cathedral_and_the_bazaar.eric_s_raymond/sisu_manifest.html)>

<sup>29</sup> <[http://www.sisudoc.org/sisu/down\\_and\\_out\\_in\\_the\\_magic\\_kingdom.cory\\_doctorow/sisu\\_manifest.html](http://www.sisudoc.org/sisu/down_and_out_in_the_magic_kingdom.cory_doctorow/sisu_manifest.html)>

< html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“Little Brother”, Cory Doctorow***

637

<”Little Brother”, Cory Doctorow>

638

<document manifest> <sup>30</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“For the Win”, Cory Doctorow***

639

<”For the Win”, Cory Doctorow>

640

<document manifest> <sup>31</sup>  
 < html, segmented text>  
 < html, scroll, document in one>

<sup>30</sup> <[http://www.sisudoc.org/sisu/little\\_brother.cory\\_doctorow/sisu\\_manifest.html](http://www.sisudoc.org/sisu/little_brother.cory_doctorow/sisu_manifest.html)>

<sup>31</sup> <[http://www.sisudoc.org/sisu/for\\_the\\_win.cory\\_doctorow/sisu\\_manifest.html](http://www.sisudoc.org/sisu/for_the_win.cory_doctorow/sisu_manifest.html)>

- < epub>
- < pdf, landscape>
- < pdf, portrait>
- < odf:odt, open document text>
- < xhtml scroll>
- < xml, sax>
- < xml, dom>
- < plain text utf-8>
- < concordance>
- < dcc, document content certificate (digests)>
- <markup source text>
- <markup source (zipped) pod>

### **“Accelerando”, Charles Stross**

641

<”Accelerando”, Charles Stross>

642

- <document manifest> <sup>32</sup>
- < html, segmented text>
- < html, scroll, document in one>
- < epub>
- < pdf, landscape>
- < pdf, portrait>
- < odf:odt, open document text>
- < xhtml scroll>
- < xml, sax>
- < xml, dom>
- < plain text utf-8>
- < concordance>
- < dcc, document content certificate (digests)>
- <markup source text>
- <markup source (zipped) pod>

### **“Tainaron”, Leena Krohn**

643

<”Tainaron”, Leena Krohn>

644

- <document manifest> <sup>33</sup>
- < html, segmented text>
- < html, scroll, document in one>
- < epub>
- < pdf, landscape>

---

<sup>32</sup> <[http://www.sisudoc.org/sisu/accelerando.charles\\_stross/sisu\\_manifest.html](http://www.sisudoc.org/sisu/accelerando.charles_stross/sisu_manifest.html)>

<sup>33</sup> <[http://www.sisudoc.org/sisu/tainaron.leena\\_krohn.1998/sisu\\_manifest.html](http://www.sisudoc.org/sisu/tainaron.leena_krohn.1998/sisu_manifest.html)>

< pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“Sphinx or Robot”, Leena Krohn***

645



646

647

<”Sphinx or Robot”, Leena Krohn>  
 <document manifest> <sup>34</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“War and Peace”, Leo Tolstoy, PG Etext 2600***

648

649

<”War and Peace”, Leo Tolstoy> <sup>35</sup>

<sup>34</sup><[http://www.sisudoc.org/sisu/sphinx\\_or\\_robot.leena\\_krohn.1996/sisu\\_manifest.html](http://www.sisudoc.org/sisu/sphinx_or_robot.leena_krohn.1996/sisu_manifest.html)>

<sup>35</sup><[http://www.jus.uio.no/sisu/war\\_and\\_peace.leo\\_tolstoy/toc.html](http://www.jus.uio.no/sisu/war_and_peace.leo_tolstoy/toc.html)> [U+258C] The ascii text was taken from Project Gutenberg. The markup transforms required are trivial. Of interest, in this instance I am saved by having alternative syntaxes/(structural modes) for marking up endnotes... as it was possible to do a simple search and replace to make the Project Gutenberg ascii presentation suitable for

```

<document manifest> 36
< html, segmented text>
< html, scroll, document in one>
< epub>
< pdf, landscape>
< pdf, portrait>
< odf:odt, open document text>
< xhtml scroll>
< xml, sax>
< xml, dom>
< plain text utf-8>
< concordance>
< dcc, document content certificate (digests)>
<markup source text>
<markup source (zipped) pod>

```

**“Don Quixote”, Miguel de Cervantes [Saavedra], translated by John Ormsby, PG Etext 996**

650

```

<"Don Quixote", Miguel de Cervantes [Saavedra]>

```

651

```

<document manifest> 37
< html, segmented text>
< html, scroll, document in one>
< epub>
< pdf, landscape>
< pdf, portrait>
< odf:odt, open document text>
< xhtml scroll>
< xml, sax>
< xml, dom>
< plain text utf-8>
< concordance>

```

[U+3014]bSiSUb[U+3015], using the older endnote markup style. This example instructs the program to use regular expressions, in this example the words: none; none; BOOK—FIRST—SECOND; CHAPTER; occurring at the beginning of a line, to identify what should be treated as different levels of heading in a document (and used to make the table of contents). Note that there was [U+3014]uvery little markup requiredu[U+3015] after the document headers and Project Gutenberg legal notices. As I presume the legal notices are similar in Project Gutenberg documents, (and I could not bear to think of preparing the same legal notices twice), I moved those to the "skin" for the Project, and these are now represented in the markup by [U+3014]lt[U+3015]:insert1[U+3014]gt[U+3015] and [U+3014]lt[U+3015]:insert2[U+3014]gt[U+3015] and the legal notices are available for similar insertion into the next Project Gutenberg text prepared for [U+3014]bSiSUb[U+3015], should there be one. [U+258C] I did a stylesheet/skin for the Gutenberg Project, ... I may have to remove.

<sup>36</sup> <[http://www.sisudoc.org/sisu/war\\_and\\_peace.leo\\_tolstoy/sisu\\_manifest.html](http://www.sisudoc.org/sisu/war_and_peace.leo_tolstoy/sisu_manifest.html)>

<sup>37</sup> <[http://www.sisudoc.org/sisu/don\\_quixote.miguel\\_de\\_cervantes/sisu\\_manifest.html](http://www.sisudoc.org/sisu/don_quixote.miguel_de_cervantes/sisu_manifest.html)>

< dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

**“*Gulliver’s Travels*”, Jonathan Swift, transcribed from the 1892 George Bell and Sons edition by David Price, PG Etext 829**

652

<”Gulliver’s Travels”, Jonathan Swift>  
 <document manifest> <sup>38</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

653

**“*Alice’s Adventures in Wonderland*”, Lewis Carroll, PG Etext 11**

654

<”Alice’s Adventures in Wonderland”, Lewis Carroll>  
 <document manifest> <sup>39</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>

655

<sup>38</sup> <[http://www.sisudoc.org/sisu/gullivers\\_travels.jonathan\\_swift/sisu\\_manifest.html](http://www.sisudoc.org/sisu/gullivers_travels.jonathan_swift/sisu_manifest.html)>

<sup>39</sup> <[http://www.sisudoc.org/sisu/alices\\_adventures\\_in\\_wonderland.lewis\\_carroll/sisu\\_manifest.html](http://www.sisudoc.org/sisu/alices_adventures_in_wonderland.lewis_carroll/sisu_manifest.html)>

<markup source text>  
 <markup source (zipped) pod>

***“Through The Looking-Glass”, Lewis Carroll, PG Etext 12***

656

<”Through The Looking-Glass”, Lewis Carroll>  
 <document manifest> <sup>40</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

657

***“Alice’s Adventures in Wonderland” and “Through The Looking-Glass”, Lewis Carroll, PG Etexts 11 and 12***

658

<”Alice’s Adventures in Wonderland” and ”Through The Looking-Glass”, Lewis Carroll>  
 <document manifest> <sup>41</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

659

<sup>40</sup> <[http://www.sisudoc.org/sisu/through\\_the\\_looking\\_glass.lewis\\_carroll/sisu\\_manifest.html](http://www.sisudoc.org/sisu/through_the_looking_glass.lewis_carroll/sisu_manifest.html)>

<sup>41</sup> <[http://www.sisudoc.org/sisu/alices\\_adventures\\_in\\_wonderland\\_and\\_through\\_the\\_looking\\_glass.lewis\\_carroll/sisu\\_manifest.html](http://www.sisudoc.org/sisu/alices_adventures_in_wonderland_and_through_the_looking_glass.lewis_carroll/sisu_manifest.html)>

**“Gnu Public License 2”, (GPL 2) Free Software Foundation**

660

&lt;”Gnu Public License 2”, (GPL 2) Free Software Foundation&gt;

<document manifest> <sup>42</sup>

- < html, segmented text>
- < html, scroll, document in one>
- < epub>
- < pdf, landscape>
- < pdf, portrait>
- < odf:odt, open document text>
- < xhtml scroll>
- < xml, sax>
- < xml, dom>
- < plain text utf-8>
- < concordance>
- < dcc, document content certificate (digests)>
- <markup source text>
- <markup source (zipped) pod>

661

**“Gnu Public License v3 - Third discussion draft”, (GPLv3) Free Software Foundation**

662

&lt;”Gnu Public License 3 - Third discussion draft”, (GPL v3 draft3) Free Software Foundation&gt;

<document manifest> <sup>43</sup>

- < html, segmented text>
- < html, scroll, document in one>
- < epub>
- < pdf, landscape>
- < pdf, portrait>
- < odf:odt, open document text>
- < xhtml scroll>
- < xml, sax>
- < xml, dom>
- < plain text utf-8>
- < concordance>
- < dcc, document content certificate (digests)>
- <markup source text>
- <markup source (zipped) pod>

663

**“Debian Social Contract”**

664

<sup>42</sup> <[http://www.sisudoc.org/sisu/gpl2.fsf/sisu\\_manifest.html](http://www.sisudoc.org/sisu/gpl2.fsf/sisu_manifest.html)><sup>43</sup> <[http://www.sisudoc.org/sisu/gpl3\\_draft3.fsf/sisu\\_manifest.html](http://www.sisudoc.org/sisu/gpl3_draft3.fsf/sisu_manifest.html)>

665

<"Debian Social Contract">  
 <document manifest> <sup>44</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

***"Debian Constitution v1.3"*, (simple/default markup)**

666

<"Debian Constitution v1.3">  
 <document manifest> <sup>45</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

667

***"Debian Constitution v1.3"*, (markup adjusted for output to more closely match the original)**

668

<"Debian Constitution v1.3", (markup adjusted for output to more closely match the original)>

669

<sup>44</sup> <[http://www.sisudoc.org/sisu/debian\\_social\\_contract\\_v1.1/sisu\\_manifest.html](http://www.sisudoc.org/sisu/debian_social_contract_v1.1/sisu_manifest.html)>

<sup>45</sup> <[http://www.sisudoc.org/sisu/debian\\_constitution\\_v1.3/sisu\\_manifest.html](http://www.sisudoc.org/sisu/debian_constitution_v1.3/sisu_manifest.html)>

<document manifest> <sup>46</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“Debian Constitution v1.2”, (simple/default markup)***

670

<”Debian Constitution v1.2 (more translations)”>

671

<document manifest> <sup>47</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### ***“Debian Constitution v1.2”, (markup adjusted for output to more closely match the original)***

672

<”Debian Constitution (more translations)”, (markup adjusted for output to more closely match the original)>

673

<sup>46</sup> <[http://www.sisudoc.org/sisu/debian\\_constitution\\_v1.3.adjusted/sisu\\_manifest.html](http://www.sisudoc.org/sisu/debian_constitution_v1.3.adjusted/sisu_manifest.html)>

<sup>47</sup> <[http://www.sisudoc.org/sisu/debian\\_constitution\\_v1.2/sisu\\_manifest.html](http://www.sisudoc.org/sisu/debian_constitution_v1.2/sisu_manifest.html)>

<document manifest> <sup>48</sup>  
< html, segmented text>  
< html, scroll, document in one>  
< epub>  
< pdf, landscape>  
< pdf, portrait>  
< odf:odt, open document text>  
< xhtml scroll>  
< xml, sax>  
< xml, dom>  
< plain text utf-8>  
< concordance>  
< dcc, document content certificate (digests)>  
<markup source text>  
<markup source (zipped) pod>

***“A Uniform Sales Terminology”, Vikki Rogers and Albert Kritzer***

674

<”A Uniform Sales Terminology”, Vikki Rogers and Albert Kritzer>

675

<document manifest> <sup>49</sup>  
< html, segmented text>  
< html, scroll, document in one>  
< epub>  
< pdf, landscape>  
< pdf, portrait>  
< odf:odt, open document text>  
< xhtml scroll>  
< xml, sax>  
< xml, dom>  
< plain text utf-8>  
< concordance>  
< dcc, document content certificate (digests)>  
<markup source text>  
<markup source (zipped) pod>

***“The Autonomous Contract” 1997 - markup sample***

676

<”The Autonomous Contract” 1997 - markup sample>

677

<document manifest> <sup>50</sup>

---

<sup>48</sup> <[http://www.sisudoc.org/sisu/debian\\_constitution\\_v1.2.adjusted/sisu\\_manifest.html](http://www.sisudoc.org/sisu/debian_constitution_v1.2.adjusted/sisu_manifest.html)>

<sup>49</sup> <[http://www.sisudoc.org/sisu/a\\_uniform\\_international\\_sales\\_terminology.vikki\\_rogers.and.albert\\_kritzer/sisu\\_manifest.html](http://www.sisudoc.org/sisu/a_uniform_international_sales_terminology.vikki_rogers.and.albert_kritzer/sisu_manifest.html)>

<sup>50</sup> <[http://www.sisudoc.org/sisu/the\\_autonomous\\_contract.amissah.19970710/sisu\\_manifest.html](http://www.sisudoc.org/sisu/the_autonomous_contract.amissah.19970710/sisu_manifest.html)>

< html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### *“The Autonomous Contract Revisited”* - markup sample

678

679

<“The Autonomous Contract Revisited” - markup sample> <sup>51</sup>

<document manifest> <sup>52</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>

<sup>51</sup>[http://www.jus.uio.no/sisu/autonomy\\_markup0/toc.html](http://www.jus.uio.no/sisu/autonomy_markup0/toc.html) [U+258C] alternative markup variations revolving around endnotes [U+258C] (i) as above, markup with embedded endnotes, and header list of words/phrases to emphasise [U+258C]

[http://www.jus.uio.no/sisu/sample/syntax/autonomy\\_markup0.sst.html](http://www.jus.uio.no/sisu/sample/syntax/autonomy_markup0.sst.html) [U+258C]

[http://www.jus.uio.no/sisu/sample/markup/autonomy\\_markup0.sst](http://www.jus.uio.no/sisu/sample/markup/autonomy_markup0.sst) [U+258C] (ii) Again markup with embedded endnotes, but font faces changed within paragraphs rather than in header as in i [U+258C]

[http://www.jus.uio.no/sisu/sample/syntax/autonomy\\_markup1.sst.html](http://www.jus.uio.no/sisu/sample/syntax/autonomy_markup1.sst.html) [U+258C]

[http://www.jus.uio.no/sisu/sample/markup/autonomy\\_markup1.sst](http://www.jus.uio.no/sisu/sample/markup/autonomy_markup1.sst) [U+258C] (iii) Markup with endnote placemarks within paragraphs, the endnotes following the paragraph that contains them

[http://www.jus.uio.no/sisu/sample/syntax/autonomy\\_markup2.sst.html](http://www.jus.uio.no/sisu/sample/syntax/autonomy_markup2.sst.html) [U+258C]

[http://www.jus.uio.no/sisu/sample/markup/autonomy\\_markup2.sst](http://www.jus.uio.no/sisu/sample/markup/autonomy_markup2.sst) [U+258C] (iv) Another alternative is to place the marked up endnotes sequentially and at the end of the text. This also works. The paragraph variant iii is perhaps easier to visually check should there be missing endnotes; but this variant iv may better suit the conversion of alternatively pre-prepared documents.

<sup>52</sup>[http://www.sisudoc.org/sisu/autonomy\\_markup0/sisu\\_manifest.html](http://www.sisudoc.org/sisu/autonomy_markup0/sisu_manifest.html)

< dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

## ***“United Nations Convention on Contracts for the International Sale of Goods”***

680

<“United Nations Convention on Contracts for the International Sale of Goods”><sup>53</sup>

681

<document manifest><sup>54</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>  
 < odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

## **/PECL/ the “Principles of European Contract Law”**

682

<“Principles of European Contract Law”>

683

<document manifest><sup>55</sup>  
 < html, segmented text>  
 < html, scroll, document in one>  
 < epub>  
 < pdf, landscape>  
 < pdf, portrait>

---

<sup>53</sup><http:

//www.jus.uio.no/sisu/un\_contracts\_international\_sale\_of\_goods\_convention\_1980/toc.html> [U+258C] This example instructs the program to use regular expressions, in this example the words: Part, Chapter, Section, Article occurring at the beginning of a line, to identify what should be treated as different levels of heading in a document (and used to make the table of contents). [U+258C] This example instructs the program to use regular expressions, in this example the words: Part, Chapter, Section, Article occurring at the beginning of a line, to identify what should be treated as different levels of heading in a document (and used to make the table of contents).

<sup>54</sup><http:

//www.sisudoc.org/sisu/un\_contracts\_international\_sale\_of\_goods\_convention\_1980/sisu\_manifest.html>

<sup>55</sup><http://www.sisudoc.org/sisu/eu\_contract\_principles\_parts\_1\_to\_3\_2002/sisu\_manifest.html>

< odf:odt, open document text>  
 < xhtml scroll>  
 < xml, sax>  
 < xml, dom>  
 < plain text utf-8>  
 < concordance>  
 < dcc, document content certificate (digests)>  
 <markup source text>  
 <markup source (zipped) pod>

### 21.3 SQL - PostgreSQL, SQLite

A Sample search form is available at <<http://search.sisudoc.org>>

A few canned searches, showing object numbers. Search for:

<English documents matching Linux OR Debian>

<GPL OR Stallman>

<invention OR innovation>

<copyright in English language documents>

Note that the searches done in this form are case sensitive.

Expand those same searches, showing the matching text in each document:

<Linux OR Debian>

<GPL OR Stallman>

<invention OR innovation in English language>

<copyright in English language documents>

Note you may set results either for documents matched and object number locations within each matched document meeting the search criteria; or display the names of the documents matched along with the objects (paragraphs) that meet the search criteria.<sup>56</sup>

### 21.4 Lex Mercatoria as an example

There is quite a bit to peruse if you explore the site Lex Mercatoria:

<<http://www.lexmercatoria.org/>><sup>57</sup>

or perhaps:

<<http://lexmercatoria.org/treaties.and.organisations/lm.chronological>><sup>58</sup>

<sup>56</sup>of this feature when demonstrated to an IBM software innovations evaluator in 2004 he said to paraphrase: this could be of interest to us. We have large document management systems, you can search hundreds of thousands of documents and we can tell you which documents meet your search criteria, but there is no way we can tell you without opening each document where within each your matches are found.

<sup>57</sup><<http://www.jus.uio.no/lm/index>>

**21.5 For good measure the markup for a document with lots of (simple) tables** 703

SiSU is not optimised for table making, but does handle simple tables. 704

Γ <Output of table file example> <sup>59</sup> 705

**21.6 And a link to the output of a reported case** 706

<<http://www.jus.uio.no/lm/england.fothergill.v.monarch.airlines.hl.1980/toc.html>> 707

---

<sup>58</sup> <<http://www.jus.uio.no/lm/treaties.and.organisations/lm.chronological>>

<sup>59</sup> <<http://www.jus.uio.no/lm/un.conventions.membership.status/toc.html>>

## 22. A Checklist of Output Features

708

This table gives an indication of the features that are available for various forms of output of SiSU .

709

## sisu-2.0.0 on 2010-03-06

710

711

feature	txt	ltx/pdf	HTML	EPUB	XML/s	XML/d	ODF	SQLite	pgSQL
headings	*	*	*	*	*	*	*	*	*
footnotes	*	*	*	*	*	*	*	*	*
bold, underscore, italics	.	*	*	*	*	*	*	*	*
strikethrough	.	*	*	*	*	*	*		
superscript, subscript	.	*	*	*	*	*	*		
extended ascii set (utf-8)	*	*	*	*	*	*	*		*
indents	*	*	*	*	*	*	*		
bullets	.	*	*	*	*	*	.		
groups									
tables		*	*	*	.	.	.	.	.
poem	*	*	*	*	.	.	*	.	.
code	*	*	*	*	.	.	*	.	.
url	*	*	*	*	*	*	*	.	.
links	*	*	*	*	*	*	*	.	.
images	-	*	*	*	T	T	*	T	T
image caption	-	*	*	*					
table of contents		*	*	*	*	*	.		
page header/footer?	-	*	*	*	*	*	t		
line break	*	*	*	*	*	*	*		
page break		*					*		
segments			*	*					
skins	*	*	*	*	*	*			
ocn	.	*	*	*	*	*	-?	*	*
auto-heading numbers	*	*	*	*	*	*	*	*	*
minor list numbering	*	*	*	*	*	*	*	*	*
special characters	.	.	.	.					

## sisu-1.0.0 on 2009-10-28

712

713

feature	txt	ltx/pdf	HTML	XML/s	XML/d	ODF	SQLite	pgSQL
headings	*	*	*	*	*	*	*	*
footnotes	*	*	*	*	*	*	*	*
bold, underscore, italics	.	*	*	*	*	*	*	*
strikethrough	.	*	*	*	*	*		
superscript, subscript	.	*	*	*	*	*		
extended ascii set (utf-8)	*	*	*	*	*	*		*
indents	*	*	*	*	*	*		

## SiSU - Manual

---

feature	txt	ltx/pdf	HTML	XML/s	XML/d	ODF	SQLite	pgSQL
bullets	.	*	*	*	*	.		
groups								
tables		*	*	.	.	.	.	.
poem	*	*	*	.	.	*	.	.
code	*	*	*	.	.	*	.	.
url	*	*	*	*	*	*	.	.
links	*	*	*	*	*	*	.	.
images	-	*	*	T	T	*	T	T
image caption	-	*	*					
table of contents		*	*	*	*	.		
page header/footer?	-	*	*	*	*	t		
line break	*	*	*	*	*	*		
page break		*				*		
segments			*					
skins	*	*	*	*	*			
ocn	.	*	*	*	*	._?	*	*
auto-heading numbers	*	*	*	*	*	*	*	*
minor list numbering	*	*	*	*	*	*	*	*
special characters	.	.	.					

### sisu-0.36.6 on 2006-01-23

714

715

feature	txt	ltx/pdf	HTML	XHTML	XML/s	XML/d	ODF	SQLite	pgSQL
headings	*	*	*	*	*	*	*	*	*
footnotes	*	*	*	*	*	*	*	*	*
bold, underscore, italics	.	*	*	*	*	*	*	*	*
strikethrough	.	*	*	*	*	*	*		
superscript, subscript	.	*	*	*	*	*	*		
extended ascii set (utf-8)	*	*	*	*	*	*	*		*
indents	*	*	*	*	*	*	*		
bullets	.	*	*	*	*	*	.		
groups									
tables		*	*	.	.	.	.	.	.
poem	*	*	*	.	.	.	*	.	.
code	*	*	*	.	.	.	*	.	.
url	*	*	*	*	*	*	*	.	.
links	*	*	*	*	*	*	*	.	.
images	-	*	*	T	T	T	*	T	T
image caption	-	*	*						
table of contents		*	*	*	*	*	.		
page header/footer?	-	*	*	*	*	*	t		
line break	*	*	*	*	*	*	*		
page break		*					*		
segments			*						

feature	txt	ltx/pdf	HTML	XHTML	XML/s	XML/d	ODF	SQLite	pgSQL
skins	*	*	*	*	*	*			
ocn	.	*	*	*	*	*	-?	*	*
auto-heading numbers	*	*	*	*	*	*	*	*	*
minor list numbering	*	*	*	*	*	*	*	*	*
special characters	.	.	.						

716

Done

\* yes/done

. partial

- not available/appropriate

Not Done

T task todo

t lesser task/todo

not done

### 23. SiSU Search - Introduction

717

**SiSU** output can easily and conveniently be indexed by a number of standalone indexing tools, such as Lucene, Hyperestraier. 718

Because the document structure of sites created is clearly defined, and the text object citation system is available hypothetically at least, for all forms of output, it is possible to search the sql database, and either read results from that database, or just as simply map the results to the html output, which has richer text markup. 719

In addition to this **SiSU** has the ability to populate a relational sql type database with documents at an object level, with objects numbers that are shared across different output types, which make them searchable with that degree of granularity. Basically, your match criteria is met by these documents and at these locations within each document, which can be viewed within the database directly or in various output formats. 720

## 24. SQL 721

### 24.1 populating SQL type databases 722

**SiSU** feeds sisu markup documents into sql type databases PostgreSQL<sup>60</sup> and/or SQLite<sup>61</sup> database together with information related to document structure. 723

This is one of the more interesting output forms, as all the structural data of the documents are retained (though can be ignored by the user of the database should they so choose). All site texts/documents are (currently) streamed to four tables: 724

Γ one containing semantic (and other) headers, including, title, author, subject, (the Dublin Core...); 725

Γ another the substantive texts by individual “paragraph” (or object) - along with structural information, each paragraph being identifiable by its paragraph number (if it has one which almost all of them do), and the substantive text of each paragraph quite naturally being searchable (both in formatted and clean text versions for searching); and 726

Γ a third containing endnotes cross-referenced back to the paragraph from which they are referenced (both in formatted and clean text versions for searching). 727

Γ a fourth table with a one to one relation with the headers table contains full text versions of output, eg. pdf, html, xml, and ascii. 728

There is of course the possibility to add further structures. 729

At this level **SiSU** loads a relational database with documents chunked into objects, their smallest logical structurally constituent parts, as text objects, with their object citation number and all other structural information needed to construct the document. Text is stored (at this text object level) with and without elementary markup tagging, the stripped version being so as to facilitate ease of searching. 730

Being able to search a relational database at an object level with the **SiSU** citation system is an effective way of locating content generated by **SiSU**. As individual text objects of a document stored (and indexed) together with object numbers, and all versions of the document have the same numbering, complex searches can be tailored to return just the locations of the search results relevant for all available output formats, with live links to the precise locations in the database or in html/xml documents; or, the structural information provided makes it possible to search the full contents of the database and have headings in which search content appears, or to search only headings etc. (as the Dublin Core is incorporated it is easy to make use of that as well). 731

---

<sup>60</sup> <<http://www.postgresql.org/>>  
<<http://advocacy.postgresql.org/>>  
<<http://en.wikipedia.org/wiki/Postgresql>>

<sup>61</sup> <<http://www.hwaci.com/sw/sqlite/>>  
<<http://en.wikipedia.org/wiki/Sqlite>>

<b>25. Postgresql</b>	732
<b>25.1 Name</b>	733
<b>SiSU</b> - Structured information, Serialized Units - a document publishing system, postgresql dependency package	734
<b>25.2 Description</b>	735
Information related to using postgresql with sisu (and related to the sisu_postgresql dependency package, which is a dummy package to install dependencies needed for <b>SiSU</b> to populate a postgresql database, this being part of <b>SiSU</b> - man sisu).	736
<b>25.3 Synopsis</b>	737
sisu -D [instruction] [filename/wildcard if required]	738
sisu -D -pg -[instruction] [filename/wildcard if required]	739
<b>25.4 Commands</b>	740
Mappings to two databases are provided by default, postgresql and sqlite, the same commands are used within sisu to construct and populate databases however -d (lowercase) denotes sqlite and -D (uppercase) denotes postgresql, alternatively -sqlite or -pgsql may be used	741
<b>-D</b> or <b>-pgsql</b> may be used interchangeably.	742
<b>25.4.1 create and destroy database</b>	743
<b>-pgsql -createall</b>	744
initial step, creates required relations (tables, indexes) in existing (postgresql) database (a database should be created manually and given the same name as working directory, as requested) (rb.dbi)	
<b>sisu -D -createdb</b>	745
creates database where no database existed before	
<b>sisu -D -create</b>	746
creates database tables where no database tables existed before	
<b>sisu -D -Dropall</b>	747
destroys database (including all its content)! kills data and drops tables, indexes and database associated with a given directory (and directories of the same name).	
<b>sisu -D -recreate</b>	748
destroys existing database and builds a new empty database structure	

- 25.4.2 import and remove documents** 749
- sisu -D -import -v [filename/wildcard]** 750  
populates database with the contents of the file. Imports documents(s) specified to a postgresql database (at an object level).
- sisu -D -update -v [filename/wildcard]** 751  
updates file contents in database
- sisu -D -remove -v [filename/wildcard]** 752  
removes specified document from postgresql database.

**26. Sqlite** 753**26.1 Name** 754

**SiSU** - Structured information, Serialized Units - a document publishing system. 755

**26.2 Description** 756

Information related to using sqlite with sisu (and related to the sisu\_sqlite dependency package, which is a dummy package to install dependencies needed for **SiSU** to populate an sqlite database, this being part of **SiSU** - man sisu). 757

**26.3 Synopsis** 758

`sisu -d [instruction] [filename/wildcard if required]` 759

`sisu -d -(sqlite|pg) -[instruction] [filename/wildcard if required]` 760

**26.4 Commands** 761

Mappings to two databases are provided by default, postgresql and sqlite, the same commands are used within sisu to construct and populate databases however -d (lowercase) denotes sqlite and -D (uppercase) denotes postgresql, alternatively `-sqlite` or `-pgsql` may be used 762

`-d` or `-sqlite` may be used interchangeably. 763

**26.4.1 create and destroy database** 764

`-sqlite -createall` 765

initial step, creates required relations (tables, indexes) in existing (sqlite) database (a database should be created manually and given the same name as working directory, as requested) (rb.dbi)

`sisu -d -createdb` 766

creates database where no database existed before

`sisu -d -create` 767

creates database tables where no database tables existed before

`sisu -d -dropall` 768

destroys database (including all its content)! kills data and drops tables, indexes and database associated with a given directory (and directories of the same name).

`sisu -d -recreate` 769

destroys existing database and builds a new empty database structure

- 26.4.2 import and remove documents** 770
- sisu -d -import -v [filename/wildcard]** 771  
populates database with the contents of the file. Imports documents(s) specified to an sqlite database (at an object level).
- sisu -d -update -v [filename/wildcard]** 772  
updates file contents in database
- sisu -d -remove -v [filename/wildcard]** 773  
removes specified document from sqlite database.

## 27. Introduction 774

### 27.1 Search - database frontend sample, utilising database and SiSU features, including object citation numbering (backend currently PostgreSQL) 775

<Sample search frontend> <sup>62</sup> A small database and sample query front-end (search from) that makes use of the citation system, object citation numbering to demonstrates functionality.<sup>63</sup> 776

**SiSU** can provide information on which documents are matched and at what locations within each document the matches are found. These results are relevant across all outputs using object citation numbering, which includes html, XML, EPUB, LaTeX, PDF and indeed the SQL database. You can then refer to one of the other outputs or in the SQL database expand the text within the matched objects (paragraphs) in the documents matched. 777

Note you may set results either for documents matched and object number locations within each matched document meeting the search criteria; or display the names of the documents matched along with the objects (paragraphs) that meet the search criteria.<sup>64</sup> 778

#### **sisu -F -webserv-webrick** 779

builds a cgi web search frontend for the database created

The following is feedback on the setup on a machine provided by the help command: 780

```

sisu -help sql 781

Postgresql 782
  user:          ralph
  current db set: SiSU_sisu
  port:          5432
  dbi connect:   DBI:Pg:database=SiSU_sisu;port=5432

sqlite
  current db set: /home/ralph/sisu_www/sisu/sisu.sqlite.db
  dbi connect    DBI:SQLite:/home/ralph/sisu_www/sisu/sisu.sqlite.db
```

Note on databases built 783

By default, [unless otherwise specified] databases are built on a directory basis, from collections of documents within that directory. The name of the directory you choose to work from is used as the database name, i.e. if you are working in a directory called 784

---

<sup>62</sup><<http://search.sisudoc.org>>

<sup>63</sup>(which could be extended further with current back-end). As regards scaling of the database, it is as scalable as the database (here PostgreSQL) and hardware allow.

<sup>64</sup>of this feature when demonstrated to an IBM software innovations evaluator in 2004 he said to paraphrase: this could be of interest to us. We have large document management systems, you can search hundreds of thousands of documents and we can tell you which documents meet your search criteria, but there is no way we can tell you without opening each document where within each your matches are found.

/home/ralph/ebook the database SiSU\_ebook is used. [otherwise a manual mapping for the collection is necessary]

## 27.2 Search Form

### **sisu -F**

generates a sample search form, which must be copied to the web-server cgi directory

### **sisu -F --webserv-webrick**

generates a sample search form for use with the webrick server, which must be copied to the web-server cgi directory

### **sisu -Fv**

as above, and provides some information on setting up hyperestraier

### **sisu -W**

starts the webrick server which should be available wherever sisu is properly installed

The generated search form must be copied manually to the webserver directory as instructed

## 28. Hyperestraier

See the documentation for hyperestraier:

```
<http://hyperestraier.sourceforge.net/>
/usr/share/doc/hyperestraier/index.html
man estcmd
```

on `sisu_hyperestraier`:

```
man sisu_hyperestraier
/usr/share/doc/sisu/sisu-markup/sisu_hyperestraier/index.html
```

NOTE: the examples that follow assume that `sisu` output is placed in the directory `~/home/ralph/sisu_www`

(A) to generate the index within the webserver directory to be indexed:

```
estcmd gather -sd [index name] [directory path to index]
```

the following are examples that will need to be tailored according to your needs:

```
cd /home/ralph/sisu_www
estcmd gather -sd casket /home/ralph/sisu_www
```

you may use the ‘`find`’ command together with ‘`egrep`’ to limit indexing to particular document collection directories within the web server directory:

```
find /home/ralph/sisu_www -type f | egrep '/home/ralph/sisu_www/sisu/.+?.html$'
|estcmd gather -sd casket -
```

Check which directories in the `webserver/output` directory (`~/sisu_www` or elsewhere depending on configuration) you wish to include in the search index.

As `sisu` duplicates output in multiple file formats, it is probably preferable to limit the `estraier` index to `html` output, and as it may also be desirable to exclude files ‘`plain.txt`’, ‘`toc.html`’ and ‘`concordance.html`’, as these duplicate information held in other `html` output e.g.

```
find /home/ralph/sisu_www -type f | egrep '/sisu_www/(sisu|bookmarks)/.+?.html$'
| egrep -v '(doc|concordance).html$' |estcmd gather -sd casket -
```

from your current document preparation/markup directory, you would construct a `rune` along the following lines:

```
find /home/ralph/sisu_www -type f | egrep '/home/ralph/sisu_www/([specify first
directory for inclusion]|[specify second directory for inclusion]|[another directory for
inclusion? ...])/.+?.html$' | egrep -v '(doc|concordance).html$' |estcmd gather -sd
/home/ralph/sisu_www/casket -
```

(B) to set up the search form

(i) copy `estseek.cgi` to your `cgi` directory and set file permissions to 755:

```
sudo cp -vi /usr/lib/estraier/estseek.cgi /usr/lib/cgi-bin
```

sudo chmod -v 755 /usr/lib/cgi-bin/estseek.cgi	815
sudo cp -v /usr/share/hyperestraier/estseek.* /usr/lib/cgi-bin	816
[see estraier documentation for paths]	817
(ii) edit estseek.conf, with attention to the lines starting ‘indexname:’ and ‘replace:’:	818
indexname: /home/ralph/sisu_www/casket	819
replace: ^ file:///home/ralph/sisu_www<{!}>	820
replace: /index.html?\${!}/	821
(C) to test using webrick, start webrick:	822
sisu -W	823
and try open the url: <http://localhost:8081/cgi-bin/estseek.cgi>	824

<b>29. sisu_webrick</b>	825
<b>29.1 Name</b>	826
<b>SiSU</b> - Structured information, Serialized Units - a document publishing system	827
<b>29.2 Synopsis</b>	828
sisu_webrick [port]	829
or	830
sisu -W [port]	831
<b>29.3 Description</b>	832
sisu_webrick is part of <b>SiSU</b> (man sisu) sisu_webrick starts <b>Ruby</b> 's Webrick web-server and points it to the directories to which <b>SiSU</b> output is written, providing a list of these directories (assuming <b>SiSU</b> is in use and they exist).	833
The default port for sisu_webrick is set to 8081, this may be modified in the yaml file: ~/.sisu/sisurc.yml a sample of which is provided as /etc/sisu/sisurc.yml (or in the equivalent directory on your system).	834
<b>29.4 Summary of man page</b>	835
sisu_webrick, may be started on it's own with the command: sisu_webrick [port] or using the sisu command with the -W flag: sisu -W [port]	836
where no port is given and settings are unchanged the default port is 8081	837
<b>29.5 Document processing command flags</b>	838
sisu -W [port] starts <b>Ruby</b> Webrick web-server, serving <b>SiSU</b> output directories, on the port provided, or if no port is provided and the defaults have not been changed in ~/.sisu/sisurc.yml then on port 8081	839
<b>29.6 Further information</b>	840
For more information on <b>SiSU</b> see: < <a href="http://www.sisudoc.org/">http://www.sisudoc.org/</a> > OR < <a href="http://www.jus.uio.no/sisu">http://www.jus.uio.no/sisu</a> >	841
or man sisu	842
<b>29.7 Author</b>	843
Ralph Amissah < <a href="mailto:ralph@amissah.com">ralph@amissah.com</a> > OR < <a href="mailto:ralph.amissah@gmail.com">ralph.amissah@gmail.com</a> >	844

**29.8 SEE ALSO**

sisu(1)

sisu\_vim(7)

845

846

847

### 30. Remote Source Documents

848

**SiSU** processing instructions can be run against remote source documents by providing the url of the documents against which the processing instructions are to be carried out. The remote **SiSU** documents can either be sisu marked up files in plaintext .sst or .ssm or; zipped sisu files, sisupod.zip or filename.ssp

849

#### **.sst / .ssm - sisu text files**

850

**SiSU** can be run against source text files on a remote machine, provide the processing instruction and the url. The source file and any associated parts (such as images) will be downloaded and generated locally.

851

```
sisu -3 http://[provide url to valid .sst or .ssm file]
```

852

Any of the source documents in the sisu examples page can be used in this way, see <http://www.jus.uio.no/sisu/SiSU/examples.html> and use the url to the .sst for the desired document.

853

NOTE: to set up a remote machine to serve **SiSU** documents in this way, images should be in the directory relative to the document source ../\_sisu/image

854

#### **sisupod - zipped sisu files**

855

A sisupod is the zipped content of a sisu marked up text or texts and any other associated parts to the document such as images.

856

**SiSU** can be run against a sisupod on a (local or) remote machine, provide the processing instruction and the url, the sisupod will be downloaded and the documents it contains generated locally.

857

```
sisu -3 http://[provide url to valid sisupod.zip or .ssp file]
```

858

Any of the source documents in the sisu examples page can be used in this way, see <http://www.jus.uio.no/sisu/SiSU/examples.html> and use the url for the desired document.

859

## Remote Document Output

860

## 31. Remote Output

861

Once properly configured **SiSU** output can be automatically posted once generated to a designated remote machine using either `rsync`, or `scp`.

862

In order to do this some ssh authentication agent and keychain or similar tool will need to be configured. Once that is done the placement on a remote host can be done seamlessly with the `-r` (for `scp`) or `-R` (for `rsync`) flag, which may be used in conjunction with other processing flags, e.g.

863

```
sisu -3R sisu_remote.sst
```

864

### 31.1 commands

865

#### **-R [filename/wildcard]**

866

copies sisu output files to remote host using `rsync`. This requires that `sisurc.yml` has been provided with information on hostname and username, and that you have your “keys” and ssh agent in place. Note the behavior of `rsync` different if `-R` is used with other flags from if used alone. Alone the `rsync --delete` parameter is sent, useful for cleaning the remote directory (when `-R` is used together with other flags, it is not). Also see `-r`

#### **-r [filename/wildcard]**

867

copies sisu output files to remote host using `scp`. This requires that `sisurc.yml` has been provided with information on hostname and username, and that you have your “keys” and ssh agent in place. Also see `-R`

### 31.2 configuration

868

[expand on the setting up of an ssh-agent / keychain]

869

## 32. Remote Servers

870

As **SiSU** is generally operated using the command line, and works within a Unix type environment, **SiSU** the program and all documents can just as easily be on a remote server, to which you are logged on using a terminal, and commands and operations would be pretty much the same as they would be on your local machine.

871

## Download information

872

### 33. Download SiSU - Linux/Unix 873

#### SiSU Current Version - Linux/Unix 874

#### Source (tarball tar.gz) 875

Download the latest version of **SiSU** (and **SiSU** markup samples): 876

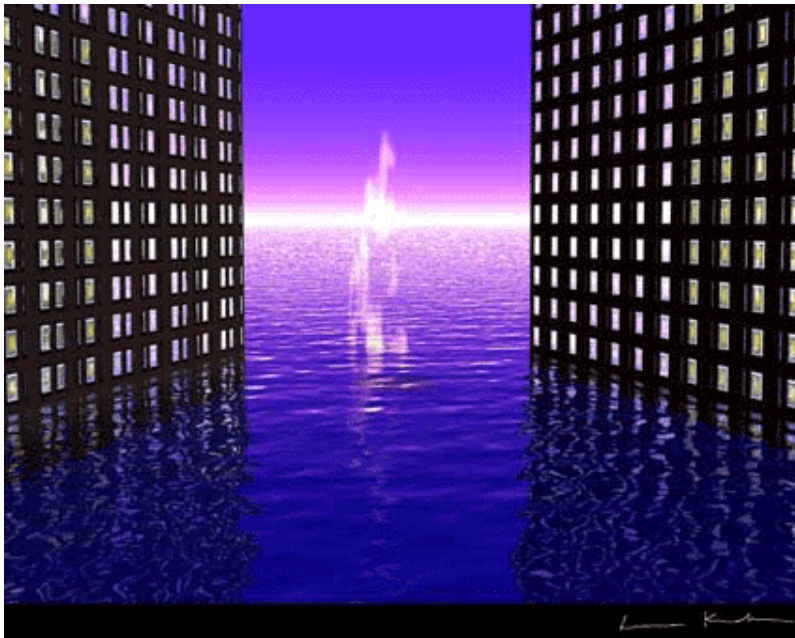
Γ `<sisu_3.0.4.orig.tar.gz (2011-03-11:10/5)>` <sup>65</sup> 877

Γ `<sisu-markup-samples_3.0.0.orig.tar.gz (of 2011-02-16:07/3 )>` <sup>66</sup> 878

For installation notes see `<http://sisudoc.org/sisu/sisu_manual/installation.html>` 879

For more general use see `<http://sisudoc.org/sisu/sisu_manual>` 880

For changelogs see `<http://www.jus.uio.no/sisu/SiSU/changelog.html>` 881



67

882

#### Git (source control management) 883

Git repository currently at: 884

Γ `git clone git://git.sisudoc.org/git/code/sisu.git` 885

Γ `<http://git.sisudoc.org/?p=code/sisu.git;a=summary>` 886

<sup>65</sup>`<http://www.jus.uio.no/sisu/pkg/src/sisu_3.0.4.orig.tar.gz>`

145c409526b26cb0a14b43f4c46219fb828dc41c8211d8f77bad486a98300678 1920526

<sup>66</sup>`<http://www.jus.uio.no/sisu/pkg/src/sisu-markup-samples_3.0.0.orig.tar.gz>`

999f3cc572d0558a6af4539db0c51691dcff3371d4f92e096cbf5835806aeed4 8446814

<sup>67</sup>Reproduced with the kind permission of author and artist Leena Krohn,

`<http://www.kaapeli.fi/krohn>` Tulva is from the work Sphinx or Robot

`<http://www.jus.uio.no/sisu/sphinx_or_robot.leena_krohn.1996>` other works available online include

Tainaron `<http://www.jus.uio.no/sisu/tainaron.leena_krohn.1998>`, these two works can be found in the

book sample section `<http://www.jus.uio.no/sisu/SiSU/examples.html#sample>`

On using git, see

887

**Git documentation:** <Git User's Manual>, <sup>68</sup> <Everyday GIT With 20 Commands Or So>, <sup>69</sup> <A tutorial introduction to git>, <sup>70</sup> <A tutorial introduction to git: part two> <sup>71</sup>

888

**User contributed texts:** <The Git Community Book>, <sup>72</sup> <Git Magic>, <sup>73</sup> <Git From the Bottom Up (pdf)> <sup>74</sup>

889

## Debian

890

This section contains information on the latest **SiSU** release. For installation notes see <[http://sisudoc.org/sisu/sisu\\_manual/installation.html](http://sisudoc.org/sisu/sisu_manual/installation.html)>

891

**SiSU** is updated fairly regularly in **Debian** testing and unstable, and should be available therefrom.

892

To add this archive, should you still choose to do so, add the following lines to your /etc/apt/sources.list

893

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free
deb-src http://www.jus.uio.no/sisu/archive unstable main non-free
```

894

895

[U+3014]b☐Source☐b [U+3015]

<sisu\_3.0.4.orig.tar.gz> <sup>75</sup>

<sisu\_3.0.4-1.debian.tar.gz> <sup>76</sup>

<sisu\_3.0.4-1.dsc> <sup>77</sup>

896

[U+3014]b☐Debs☐b [U+3015]

<sisu\_3.0.4-1\_all.deb> <sup>78</sup>

<sisu-complete\_3.0.4-1\_all.deb> <sup>79</sup>

<sup>68</sup> <<http://www.kernel.org/pub/software/scm/git/docs/user-manual.html>>

<sup>69</sup> <<http://www.kernel.org/pub/software/scm/git/docs/everyday.html>>

<sup>70</sup> <<http://www.kernel.org/pub/software/scm/git/docs/v1.4.4.4/tutorial.html>>

<sup>71</sup> <<http://www.kernel.org/pub/software/scm/git/docs/v1.4.4.4/tutorial-2.html>>

<sup>72</sup> <<http://book.git-scm.com/index.html>>

<sup>73</sup> <<http://www.cs-students.stanford.edu/~blynn/gitmagic/>>

<sup>74</sup> <[http://www.newartisans.com/blog\\_assets/git.from.bottom.up.pdf](http://www.newartisans.com/blog_assets/git.from.bottom.up.pdf)>

<sup>75</sup> <[http://www.jus.uio.no/sisu/pkg/src/sisu\\_3.0.4.orig.tar.gz](http://www.jus.uio.no/sisu/pkg/src/sisu_3.0.4.orig.tar.gz)>

<sup>76</sup> <[http://www.jus.uio.no/sisu/pkg/src/sisu\\_3.0.4-1.diff.gz](http://www.jus.uio.no/sisu/pkg/src/sisu_3.0.4-1.diff.gz)>

<sup>77</sup> <[http://www.jus.uio.no/sisu/pkg/src/sisu\\_3.0.4-1](http://www.jus.uio.no/sisu/pkg/src/sisu_3.0.4-1)>

dsc> [U+258C]4886196e536e3ad48b29d27b1f0664f77b58ee4b4b21e41c0351c6b3a33dd156 1216

<sup>78</sup> <[http://www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu\\_3.0.4-1\\_all.deb](http://www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu_3.0.4-1_all.deb)> [U+258C]sisu, the base code, (the main package on which the others depend), without any dependencies other than ruby (and for convenience the ruby webrick web server), this generates a number of types of output on its own, other packages provide additional functionality, and have their dependencies [U+258C]Depends: ruby (≥1.8.2), libwebrick-ruby [U+258C]Recommends: sisu-pdf, sisu-sqlite, sisu-postgresql, sisu-examples, vim-sisu, librmagick-ruby, trang, tidy, libtidy, librexml-ruby, zip, unzip, openssl

<sup>79</sup> <[http://www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu-complete\\_3.0.4-1\\_all.deb](http://www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu-complete_3.0.4-1_all.deb)> [U+258C]a package that pulls in other packages to build the whole of sisu (excluding sisu-examples)

<sisu-pdf\_3.0.4-1\_all.deb> <sup>80</sup>

<sisu-postgresql\_3.0.4-1\_all.deb> <sup>81</sup>

<sisu-sqlite\_3.0.4-1\_all.deb> <sup>82</sup>

For changelogs see:

<<http://www.jus.uio.no/sisu/SiSU/changelog.html>>

<[http://www.jus.uio.no/sisu/sisu\\_changelog/changelog.html](http://www.jus.uio.no/sisu/sisu_changelog/changelog.html)>

## non-free

Book markup samples have been moved to non-free as the substantive text of the documents are available under the author or original publisher's license, and usually do not comply with the **Debian** Free Software Guidelines.

Γ <sisu-markup-samples\_3.0.0-1\_all.deb> <sup>83</sup>

Γ <sisu-markup-samples\_3.0.0-1.dsc> <sup>84</sup>

For changelogs see:

<[http://www.jus.uio.no/sisu/SiSU/changelog\\_markup\\_samples.html](http://www.jus.uio.no/sisu/SiSU/changelog_markup_samples.html)>

<[http://www.jus.uio.no/sisu/sisu\\_markup\\_samples\\_changelog/changelog\\_markup\\_samples.html](http://www.jus.uio.no/sisu/sisu_markup_samples_changelog/changelog_markup_samples.html)>

## RPM

The RPM is generated from the source file using Alien.<sup>85</sup> Dependencies are not handled, not even that of the essential **Ruby**. In the Howto section see the note on installing **SiSU** on Fedora 12 (2010-01-12).

```
sudo rpm -i [package name]
```

```
[U+258C]Depends: ruby (≥1.8.2), sisu, sisu-pdf, sisu-postgresql, sisu-remote, sisu-sqlite, vim-sisu [U+258C]Recommends: sisu-examples
```

```
80<http://www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu-pdf\_3.0.4-1\_all.deb> [U+258C]dependencies used by sisu to produce pdf from LaTeX generated [U+258C]Depends: sisu, tetex-bin, tetex-extra, latex-ucs [U+258C]Suggests: evince, xpdf
```

```
81<http:
```

```
//www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu-postgresql_3.0.4-1_all.deb> [U+258C]dependencies used by sisu to populate postgresql database (further configuration is necessary) [U+258C]Depends: sisu, postgresql-8.1, libdbi-ruby, libdbm-ruby, libdbd-pg-ruby [U+258C]Suggests: pgaccess, libdbd-pgsql, postgresql-contrib-8.1
```

```
82<http:
```

```
//www.jus.uio.no/sisu/archive/pool/main/s/sisu/sisu-sqlite_3.0.4-1_all.deb> [U+258C]dependencies used by sisu to populate sqlite database [U+258C]Depends: sisu, sqlite, libdbi-ruby, libdbm-ruby, libdbd-sqlite-ruby [U+258C]Suggests: libdbd-sqlite
```

```
83<http:
```

```
//www.jus.uio.no/sisu/archive/pool/non-free/s/sisu-markup-samples/sisu-markup-samples_3.0.0-1_all.deb> marked up documents and other examples related to sisu, a larger package containing a number of texts Depends: sisu
```

```
84<http://www.jus.uio.no/sisu/pkg/src/sisu-markup-samples\_3.0.0-1.dsc>
```

```
7d9d434c74a1e96da3732e420d483466d7ca1266d4e6fba4bf7f21b9e3f73aad 1307
```

```
sisu-markup-samples_3.0.0-1.dsc
```

```
85<http://www.kitenet.net/programs/alien/>
```

Γ <sisu-1.0.3-2.noarch.rpm> <sup>86</sup>	910
Γ <sisu-markup-samples_2.0.3.orig-2.noarch.rpm> <sup>87</sup>	911
For changelogs see:	912
< <a href="http://www.jus.uio.no/sisu/SiSU/changelog_markup_samples.html">http://www.jus.uio.no/sisu/SiSU/changelog_markup_samples.html</a> >	913
< <a href="http://www.jus.uio.no/sisu/sisu_markup_samples_changelog/changelog_markup_samples.html">http://www.jus.uio.no/sisu/sisu_markup_samples_changelog/changelog_markup_samples.html</a> >	914

---

<sup>86</sup> <<http://www.jus.uio.no/sisu/pkg/rpm/sisu-1.0.3-2.noarch.rpm>>  
\*

created using alien

<sup>87</sup> <[http://www.jus.uio.no/sisu/pkg/rpm/sisu-markup-samples\\_2.0.3.orig-2.noarch.rpm](http://www.jus.uio.no/sisu/pkg/rpm/sisu-markup-samples_2.0.3.orig-2.noarch.rpm)>  
\*\*\*

<[http://www.jus.uio.no/sisu/archive/pool/non-free/s/sisu-markup-samples/sisu-markup-samples\\_2.0.3-1\\_all.deb](http://www.jus.uio.no/sisu/archive/pool/non-free/s/sisu-markup-samples/sisu-markup-samples_2.0.3-1_all.deb)>  
created using: alien -r sisu\_1.0.3-1\_all.deb

## Installation

915

## 34. Installation 916

See the <download pages> <sup>88</sup> for information related to installation. 917

### 34.1 Debian 918

**SiSU** is developed on **Debian** , and packages are available for **Debian** that take care of the dependencies encountered on installation. 919

The package is divided into the following components: 920

\*sisu\*, the base code, (the main package on which the others depend), without any dependencies other than ruby (and for convenience the ruby webrick web server), this generates a number of types of output on its own, other packages provide additional functionality, and have their dependencies 921

\*sisu-complete\*, a dummy package that installs the whole of greater sisu as described below, apart from sisu-examples 922

\*sisu-pdf\*, dependencies used by sisu to produce pdf from LaTeX generated 923

\*sisu-postgresql\*, dependencies used by sisu to populate postgresql database (further configuration is necessary) 924

\*sisu-remote\*, dependencies used to place sisu output on a remote server (further configuration is necessary) 925

\*sisu-sqlite\*, dependencies used by sisu to populate sqlite database 926

\*sisu-markup-samples\*, sisu markup samples and other miscellany (under **Debian** Free Software Guidelines non-free) 927

**SiSU** is available off <Debian Unstable and Testing> <sup>89</sup> install it using apt-get, aptitude or alternative **Debian** install tools. **SiSU** is currently comprised of eight packages. 928

Initial packaging is done here and to get the latest version of **SiSU** available you may add the following line(s) to your sources list: 929

```
#/etc/apt/sources.list 930
```

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free
deb-src http://www.jus.uio.no/sisu/archive unstable main non-free
```

The non-free section is for sisu markup samples provided, which contain authored works the substantive text of which cannot be changed, and which as a result do not meet the debian free software guidelines. 931

On **Debian** there is little more to know beyond how to install software on **Debian** using apt, aptitude or synaptic. 932

```
#Using aptitude: 933
```

---

<sup>88</sup> <<http://www.jus.uio.no/sisu/SiSU/download.html>>

<sup>89</sup> <<http://packages.qa.debian.org/s/sisu.html>>

```
aptitude update
```

```
aptitude install sisu-complete sisu-markup-samples
```

```
Using apt-get
```

```
apt-get update
```

```
apt get install sisu-complete sisu-examples
```

934

## 34.2 Other Unix / Linux

935

A source tarball or an rpms built using alien are available, (however dependencies have not been tested). **SiSU** is first packaged and tested with dependency handling for **Debian**.<sup>90</sup> Information on dependencies configured for **Debian** is provided as this may be of assistance.

936

### 34.2.1 source tarball

937

#### installation with provided install script

938

To install **SiSU**, in the root directory of the unpacked **SiSU** as root type.<sup>91</sup>

939

```
ruby install
```

940

Once installed see man 8 sisu for information on additional programs that sisu makes use of.

941

Further notes on install script.

942

The install script is prepared using Rant, and a Rantfile is provided,<sup>92</sup> with more comprehensive install options, and post install and setup configuration and generation of first test file, if you have installed Stefan Lang's <Rant><sup>93</sup> installed. While in the package directory, type: rant help, or rant -T, or to install **SiSU** as root, type:

943

\*install\* is an install script prepared using Stefan Lang's <Rant><sup>94</sup> It should work whether you have previously installed Rant or not. It has fairly comprehensive install options, and can do some post install and setup configuration and generation of first test file. For options type:

944

```
ruby install -T
```

945

To install as root type:

946

```
ruby install
```

947

<sup>90</sup>Notes on dependencies are provided in the section that follows

<sup>91</sup>This makes use of rant and the provided Rantfile. Note however, that additional external package dependencies, such as tetex-extra are not taken care of for you.

<sup>92</sup>a Rantfile has been configured to do post installation setup

<sup>93</sup><<http://make.rubyforge.org/>> <<http://make.rubyforge.org/>>

<[http://rubyforge.org/frs/?group\\_id=615](http://rubyforge.org/frs/?group_id=615)>

<sup>94</sup><<http://make.rubyforge.org/>> <<http://make.rubyforge.org/>>

<[http://rubyforge.org/frs/?group\\_id=615](http://rubyforge.org/frs/?group_id=615)>

For a minimal install type: 948

    ruby install base 949

**installation with setup.rb** 950

<setup.rb> <sup>95</sup> is provided the package and will install **SiSU** <sup>96</sup> installation is a 3 step process<sup>97</sup> the following string assumes you are in the package directory and that you have root as sudo: 951

    ruby setup.rb config && ruby setup.rb setup && sudo ruby setup.rb install 952

**installation of rpm** 953

The RPM is generated from the source file using Alien.<sup>98</sup> Dependencies are not handled, not even that of the essential **Ruby** . 954

---

<sup>95</sup> <<http://i.loveruby.net/en/projects/setup/>>

<sup>96</sup>Minero Aoki

<<http://i.loveruby.net/en/projects/setup/doc/>>

<sup>97</sup>Installation instructions

<<http://i.loveruby.net/en/projects/setup/doc/usage.html>>

<sup>98</sup> <<http://www.kitenet.net/programs/alien/>>

<b>35. SiSU Components, Dependencies and Notes</b>	955
The dependency lists are from the <b>Debian</b> control file for <b>SiSU</b> version 0.36, and may assist in building <b>SiSU</b> on other distributions.	956
<b>35.1 sisu</b>	957
Γ the base code, (the main package on which the others depend), without any dependencies other than ruby (and for convenience the ruby webrick web server), this generates a number of types of output on its own, other packages provide additional functionality, and have their dependencies	958
*Depends:* on ruby ( <i>i</i> =1.8.2), libwebrick-ruby	959
*Recommends:* sisu-pdf, sisu-sqlite, sisu-postgresql, sisu-examples, librmagick-ruby, trang, tidy, libtidy, librexml-ruby, zip, unzip, openssl	960
initialise directory	961
sisu -CC	962
html	963
sisu -hv [filename/wildcard]	964
sisu -Hv [filename/wildcard]	965
LaTeX (but sisu-pdf dependencies required to convert that to pdf)	966
sisu -pv [filename/wildcard]	967
plain text Unix with footnotes	968
sisu -av [filename/wildcard]	969
plain text Dos with footnotes	970
sisu -Av [filename/wildcard]	971
plain text Unix with endnotes	972
sisu -ev [filename/wildcard]	973
plain text Dos with endnotes	974
sisu -Ev [filename/wildcard]	975
openoffice odt	976
sisu -ov [filename/wildcard]	977
xhtml	978
sisu -bv [filename/wildcard]	979
XML SAX	980
sisu -xv [filename/wildcard]	981
XML DOM	982

sisu -Xv [filename/wildcard]	983
wordmap (a rudimentary index of content)	984
sisu -wv [filename/wildcard]	985
document content certificate	986
sisu -Nv [filename/wildcard]	987
placement of sourcefile in output directory	988
sisu -sv [filename/wildcard]	989
creation of source tarball with images, and placement of source tarball in ouput directory	990
sisu -Sv [filename/wildcard]	991
manifest of output produced (polls output directory and provides links to existing output)	992
sisu -yv [filename/wildcard]	993
url for output files -u -U	994
sisu -uv[and other flags] [filename/wildcard]	995
sisu -Uv [filename/wildcard]	996
toggle screen colour	997
sisu -cv[and processing flags] [filename/wildcard]	998
verbose mode	999
sisu -v[and processing flags] [filename/wildcard]	1000
sisu -V[and processing flags] [filename/wildcard]	1001
quiet mode	1002
sisu -q[and processing flags] [filename/wildcard]	1003
maintenance mode, intermediate files kept -M	1004
sisu -Mv[and other flags] [filename/wildcard]	1005
[the -v is for verbose]	1006
start the webrick server	1007
sisu -W	1008
<b>35.2 sisu-complete</b>	1009
Γ a dummy package that installs the whole <b>SiSU</b> , apart from sisu-examples	1010
*Depends:* ruby ( <i>i</i> =1.8.2), sisu, sisu-pdf, sisu-postgresql, sisu-remote, sisu-sqlite	1011
*Recommends:* sisu-examples	1012

<b>35.3 sisu-examples</b>	1013
Γ installs sisu markup samples and other miscelleny	1014
*Depends:* sisu	1015
<b>35.4 sisu-pdf</b>	1016
Γ dependencies used by sisu to produce pdf from LaTeX generated	1017
*Depends:* sisu, tetex-bin, tetex-extra, latex-ucs	1018
*Suggests:* evince, xpdf	1019
converts sisu LaTeX produced to pdf	1020
sisu -pv [filename/wildcard]	1021
[the -v is for verbose]	1022
<b>35.5 sisu-postgresql</b>	1023
Γ dependencies used by sisu to populate postgresql database (further configuration is necessary)	1024
*Depends:* sisu, postgresql-8.1, libdbi-ruby, libdbm-ruby, libdbd-pg-ruby	1025
*Suggests:* pgaccess, libdbd-pgsql, postgresql-contrib-8.1	1026
installs dependencies for sisu to work with and populate postgresql database	1027
create database	1028
sisu -Dv createall	1029
drop database	1030
sisu -Dv dropall	1031
import content	1032
sisu -Div [filename/wildcard]	1033
sisu -Dv import [filename/wildcard]	1034
update content	1035
sisu -Duv [filename/wildcard]	1036
sisu -Dv update [filename/wildcard]	1037
[the -v is for verbose]	1038
The following are available without installation of the sisu-postgresql component, but are of interest in this context	1039
generate a sample database query form for use with webserver on port 80	1040
sisu -F	1041

or for use with webrick server	1042
sisu -F webrick	1043
to start webrick server	1044
sisu -W	1045
<b>35.6 sisu-remote</b>	<b>1046</b>
Γ dependencies used to place sisu output on a remote server (further configuration is necessary)	1047
scp	1048
sisu -vr[and processing flags] [filename/wildcard]	1049
rsync	1050
sisu -vR[and processing flags] [filename/wildcard]	1051
[the -v is for verbose]	1052
*Depends:* sisu, rsync, openssh-client lsh-client, keychain	1053
<b>35.7 sisu-sqlite</b>	<b>1054</b>
Γ dependencies used by sisu to populate sqlite database	1055
*Depends:* sisu, sqlite, libdbi-ruby, libdbm-ruby, libdbd-sqlite-ruby	1056
*Suggests:* libdbd-sqlite	1057
installs dependencies for sisu to work with and populate sqlite database	1058
create database	1059
sisu -dv createall	1060
drop database	1061
sisu -dv dropall	1062
update content	1063
sisu -div [filename/wildcard]	1064
sisu -dv import [filename/wildcard]	1065
update content	1066
sisu -duv [filename/wildcard]	1067
sisu -dv update [filename/wildcard]	1068
[the -v is for verbose]	1069
The following are available without installation of the sisu-sqlite component, but are of interest in this context	1070
generate a sample database query form for use with webservice on port 80	1071

sisu -F	1072
or for use with webrick server	1073
sisu -F webrick	1074
to start webrick server	1075
sisu -W	1076

## 36. Quickstart - Getting Started Howto 1077

### 36.1 Installation 1078

Installation is currently most straightforward and tested on the **Debian** platform, as there are packages for the installation of **sisu** and all requirements for what it does. 1079

#### 36.1.1 Debian Installation 1080

**SiSU** is available directly from the **Debian** Sid and testing archives (and possibly Ubuntu), assuming your `/etc/apt/sources.list` is set accordingly: 1081

```
aptitude update 1082
aptitude install sisu-complete
```

The following `/etc/apt/sources.list` setting permits the download of additional markup samples: 1083

```
#/etc/apt/sources.list 1084
```

```
deb http://ftp.fi.debian.org/debian/ unstable main non-free contrib
deb-src http://ftp.fi.debian.org/debian/ unstable main non-free contrib
```

The aptitude commands become: 1085

```
aptitude update 1086
aptitude install sisu-complete sisu-markup-samples
```

If there are newer versions of **SiSU** upstream of the **Debian** archives, they will be available by adding the following to your `/etc/apt/sources.list` 1087

```
#/etc/apt/sources.list 1088
```

```
deb http://www.jus.uio.no/sisu/archive unstable main non-free
deb-src http://www.jus.uio.no/sisu/archive unstable main non-free
```

repeat the aptitude commands 1089

```
aptitude update 1090
aptitude install sisu-complete sisu-markup-samples
```

Note however that it is not necessary to install `sisu-complete` if not all components of `sisu` are to be used. Installing just the package `sisu` will provide basic functionality. 1091

#### 36.1.2 RPM Installation 1092

RPMs are provided though untested, they are prepared by running `alien` against the source package, and against the debs. 1093

They may be downloaded from: 1094

```
<http://www.jus.uio.no/sisu/SiSU/download.html#rpm> 1095
```

as root type: 1096

```
rpm -i [rpm package name] 1097
```

### 36.1.3 Installation from source 1098

To install **SiSU** from source check information at: 1099

`<http://www.jus.uio.no/sisu/SiSU/download.html#current>` 1100

Γ download the source package 1101

Γ Unpack the source 1102

Two alternative modes of installation from source are provided, `setup.rb` (by Minero Aoki) 1103  
and a `rant`(by Stefan Lang) built install file, in either case: the first steps are the same,  
download and unpack the source file:

For basic use **SiSU** is only dependent on the programming language in which it is written 1104  
**Ruby** , and **SiSU** will be able to generate html, EPUB, various XMLs, including ODF  
(and will also produce LaTeX). Dependencies required for further actions, though it relies  
on the installation of additional dependencies which the source tarball does not take  
care of, for things like using a database (postgresql or sqlite)<sup>99</sup> or converting LaTeX to  
pdf.

#### **setup.rb** 1105

This is a standard ruby installer, using `setup.rb` is a three step process. In the root 1106  
directory of the unpacked **SiSU** as root type:

```
ruby setup.rb config 1107
ruby setup.rb setup
#[and as root:]
ruby setup.rb install
```

further information on `setup.rb` is available from: 1108

`<http://i.loveruby.net/en/projects/setup/>` 1109

`<http://i.loveruby.net/en/projects/setup/doc/usage.html>` 1110

#### **“install”** 1111

The “install” file provided is an installer prepared using “rant”. In the root directory of 1112  
the unpacked **SiSU** as root type:

```
ruby install base 1113
```

or for a more complete installation: 1114

```
ruby install 1115
```

or 1116

```
ruby install base 1117
```

This makes use of Rant (by Stefan Lang) and the provided Rantfile. It has been configured 1118  
to do post installation setup setup configuration and generation of first test file. Note  
however, that additional external package dependencies, such as tetex-extra are not taken  
care of for you.

Further information on “rant” is available from: 1119

<sup>99</sup>There is nothing to stop MySQL support being added in future.

<code>&lt;http://make.rubyforge.org/&gt;</code>	1120
<code>&lt;http://rubyforge.org/frs/?group_id=615&gt;</code>	1121
For a list of alternative actions you may type:	1122
<code>ruby install help</code>	1123
<code>ruby install -T</code>	1124
<b>36.2 Testing SiSU, generating output</b>	1125
To check which version of sisu is installed:	1126
<code>sisu -v</code>	1127
Depending on your mode of installation one or a number of markup sample files may be found either in the directory:	1128
...	
or	1129
...	
change directory to the appropriate one:	1130
<code>cd /usr/share/doc/sisu/markup-samples/samples</code>	1131
<b>36.2.1 basic text, plaintext, html, XML, ODF, EPUB</b>	1132
Having moved to the directory that contains the markup samples (see instructions above if necessary), choose a file and run sisu against it	1133
<code>sisu -NhwoabxYyv free_as_in_freedom.rms_and_free_software.sam_williams.sst</code>	1134
this will generate html including a concordance file, opendocument text format, plaintext, XHTML and various forms of XML, and OpenDocument text	1135
<b>36.2.2 LaTeX / pdf</b>	1136
Assuming a LaTeX engine such as tetex or texlive is installed with the required modules (done automatically on selection of sisu-pdf in <b>Debian</b> )	1137
Having moved to the directory that contains the markup samples (see instructions above if necessary), choose a file and run sisu against it	1138
<code>sisu -pv free_as_in_freedom.rms_and_free_software.sam_williams.sst</code>	1139
<code>sisu -3 free_as_in_freedom.rms_and_free_software.sam_williams.sst</code>	1140
should generate most available output formats: html including a concordance file, opendocument text format, plaintext, XHTML and various forms of XML, and OpenDocument text and pdf	1141

### 36.2.3 relational database - postgresql, sqlite 1142

Relational databases need some setting up - you must have permission to create the database and write to it when you run sisu. 1143

Assuming you have the database installed and the requisite permissions 1144

```
sisu -sqlite -recreate 1145
```

```
sisu -sqlite -v -import free_as_in_freedom.rms_and_free_software.sam_williams.sst 1146
```

```
sisu -pgsql -recreate 1147
```

```
sisu -pgsql -v -import free_as_in_freedom.rms_and_free_software.sam_williams.sst 1148
```

## 36.3 Getting Help 1149

### 36.3.1 The man pages 1150

Type: 1151

```
man sisu 1152
```

The man pages are also available online, though not always kept as up to date as within the package itself: 1153

```
Γ <sisu.1> 100 1154
```

```
Γ <sisu.8> 101 1155
```

```
Γ <man directory> 102 1156
```

### 36.3.2 Built in help 1157

```
sisu -help 1158
```

```
sisu -help -env 1159
```

```
sisu -help -commands 1160
```

```
sisu -help -markup 1161
```

### 36.3.3 The home page 1162

```
<http://www.sisudoc.org/> 1163
```

```
<http://www.jus.uio.no/sisu/> 1164
```

```
<http://www.jus.uio.no/sisu/SiSU/> 1165
```

## 36.4 Markup Samples

1166

A number of markup samples (along with output) are available off:

1167

<<http://www.jus.uio.no/sisu/SiSU/examples.html>>

1168

Additional markup samples are packaged separately in the file:

1169

\*\*\*

On **Debian** they are available in non-free<sup>103</sup> to include them it is necessary to include non-free in your `/etc/apt/source.list` or obtain them from the sisu home site.

1170

---

<sup>100</sup><<http://www.jus.uio.no/sisu/man/sisu.1.html>>

<sup>101</sup><<http://www.jus.uio.no/sisu/man/sisu.8.html>>

<sup>102</sup><<http://www.jus.uio.no/sisu/man>>

<sup>103</sup>the **Debian** Free Software guidelines require that everything distributed within **Debian** can be changed - and the documents are authors' works that while freely distributable are not freely changeable.

## HowTo

1171

## 37. Getting Help 1172

An online manual of sorts should be available at: 1173

`<http://www.jus.uio.no/sisu\_manual/>` 1174

The manual pages provided with **SiSU** are also available online, and there is an interactive help, which is being superseded by the man page, and possibly some document which contains this component. 1175

### 37.1 SiSU “man” pages 1176

If **SiSU** is installed on your system usual man commands should be available, try: 1177

`man sisu` 1178

The **SiSU** man pages can be viewed online at:<sup>104</sup> 1179

An online version of the sisu man page is available here: 1180

`Γ <various sisu man pages>` <sup>105</sup> 1181

`Γ <sisu.1>` <sup>106</sup> 1182

`Γ <sisu.8>` <sup>107</sup> 1183

`Γ <sisu_webrick.1>` <sup>108</sup> 1184

### 37.2 SiSU built-in help 1185

`sisu -help` 1186

`sisu -help [subject]` 1187

`sisu -help env` [for feedback on the way your system is setup with regard to sisu] 1188

`sisu -V` [same as above command] 1189

`sisu -help commands` 1190

`sisu -help markup` 1191

### 37.3 Command Line with Flags - Batch Processing 1192

Running sisu (alone without any flags, filenames or wildcards) brings up the interactive 1193

---

<sup>104</sup>generated from source using rman

`<http://polyglotman.sourceforge.net/rman.html>`

With regard to **SiSU** man pages the formatting generated for markup syntax is not quite right, for that you might prefer the links under:

`<http://www.jus.uio.no/sample>`

<sup>105</sup>`<http://www.jus.uio.no/sisu/man/>`

<sup>106</sup>`<http://www.jus.uio.no/sisu/man/sisu.1.html>`

<sup>107</sup>`<http://www.jus.uio.no/sisu/man/sisu.8.html>`

<sup>108</sup>`<http://www.jus.uio.no/sisu/man/sisu\_webrick.1.html>`

help, as does any sisu command that is not recognised.

In the data directory run `sisu -mh filename` or wildcard eg. “`sisu -h cisg.sst`” or “`sisu -h` 1194  
\*.{sst,ssm}” to produce html version of all documents.

<b>38. Setup, initialisation</b>	1195
<b>38.1 initialise output directory</b>	1196
Images, css files for a document directory are copied to their respective locations in the output directory.	1197
while within your document markup/preparation directory, issue the following command	1198
sisu -CC	1199
<b>38.1.1 Use of search functionality, an example using sqlite</b>	1200
<b>SiSU</b> can populate PostgreSQL and Sqlite databases and provides a sample search form for querying these databases.	1201
This note provides an example to get you started and will use sqlite	1202
It is necessary to:	1203
(1) make sure the required dependencies have been installed	1204
(2) have a directory with sisu markup samples that is writable	1205
(3) use sisu to create a database	1206
(4) use sisu tp populate a database	1207
(5) use sisu to start the webrick (httpd) server	1208
(6) use sisu to create a search form	1209
(7) copy the search form to the cgi directory	1210
(8) open up the form in your browser	1211
(9) query the database using the search form	1212
<b>(1) make sure the required dependencies have been installed</b>	1213
if you use <b>Debian</b> , the following command will install the required dependencies	1214
aptitude install sisu-sqlite	1215
<b>(2) have a directory with sisu markup samples that is writable</b>	1216
ideally copy the sisu-examples directory to your home directory (because the directory in which you run this example should be writable)	1217
cp -rv /usr/share/doc/sisu/markup-samples/samples .	1218
you are better off installing the package sisu-markup-samples which will make the following available	1219
cp -rv /usr/share/doc/sisu/markup-samples-non-free/samples .	1220
<b>(3) use sisu to create an sqlite database</b>	1221
within the sisu-examples directory	1222

sisu -dv createall	1223
<b>(4) use sisu tp populate a database with some text</b>	1224
within the sisu-examples directory	1225
sisu -div free_*.sst	1226
or	1227
sisu -dv import free_*.sst debian_constitution_v1.2.sst debian_social_contract_v1.1.sst gpl2.fsfs.sst	
<b>(5) use sisu to start the webrick (httpd) server (if it has not already been started):</b>	1229
sisu -W	1230
<b>(6) use sisu to create a search form (for use with the webrick server, and your sample documents)</b>	1231
within the sisu-examples directory	1232
sisu -F webserv=webrick	1233
and follow the instructions provided	1234
#here i run into a problem, you are working from a read only #directory..., not my usual mode of operation, to complete the example #the following is necessary sudo touch sisu_sqlite.cgi sisu_pgsq.cgi sudo -P chown \$USER sisu_sqlite.cgi sisu_pgsq.cgi	1235
#now this should be possible: sisu -F webrick	1236
<b>(7) copy the search form to the cgi directory</b>	1237
sisu -F webserv=webrick	1238
and follow the instructions provided	1239
<b>(8) open up the form in your browser and query it</b>	1240
url:	1241
<http://localhost:8081/cgi-bin/sisu_sqlite.cgi>	1242
or as instructed by command sisu -F webrick	1243
<b>(9) query the database using the search form</b>	1244
if there are other options in the dropdown menu select	1245
document_samples_sisu_markup	1246
and search for some text, e.g.:	1247
aim OR project	1248
Γ selecting the *index* radio button gives an index of results using the object numbers	1249
Γ selecting the *text* radio button gives the content of the matched paragraphs with the match highlighted	1250

<b>(10) to start again with a new database</b>	1251
to start from scratch you can drop the database with the command	1252
<code>sisu -dv dropall</code>	1253
and go to step 3	1254
to get to step 3 in one step with a single command	1255
<code>sisu -dv recreate</code>	1256
continue subsequent steps	1257
<b>38.2 misc</b>	1258
<b>38.2.1 url for output files -u -U</b>	1259
<code>sisu -uv[and other flags] [filename/wildcard]</code>	1260
<code>sisu -Uv [filename/wildcard]</code>	1261
<b>38.2.2 toggle screen color</b>	1262
<code>sisu -cv[and processing flags] [filename/wildcard]</code>	1263
<b>38.2.3 verbose mode</b>	1264
<code>sisu -v[and processing flags] [filename/wildcard]</code>	1265
<code>sisu -V[and processing flags] [filename/wildcard]</code>	1266
<b>38.2.4 quiet mode</b>	1267
<code>sisu -q[and processing flags] [filename/wildcard]</code>	1268
<b>38.2.5 maintenance mode intermediate files kept -M</b>	1269
<code>sisu -Mv[and other flags] [filename/wildcard]</code>	1270
<b>38.2.6 start the webrick server</b>	1271
<code>sisu -W</code>	1272
<b>38.3 remote placement of output</b>	1273
configuration is necessary	1274
scp	1275

sisu -vr[and processing flags] [filename/wildcard]	1276
rsync	1277
sisu -vR[and processing flags] [filename/wildcard]	1278

<b>39. Configuration Files</b>	1279
Sample provided, on untarring the source tarball:	1280
conf/sisu/v2/sisurc.yml	1281
conf/sisu/v3/sisurc.yml	1282
and on installation under:	1283
/etc/sisu/v2/sisurc.yml	1284
/etc/sisu/v3/sisurc.yml	1285
The following paths are searched:	1286
./_sisu/v2/sisurc.yml or ./_sisu/v3/sisurc.yml	1287
./_sisu/sisurc.yml	1288
~/.sisu/v2/sisurc.yml or ~/.sisu/v3/sisurc.yml	1289
~/.sisu/sisurc.yml	1290
/etc/sisu/v2/sisurc.yml /etc/sisu/v3/sisurc.yml	1291
/etc/sisu/sisurc.yml	1292

<b>40. Markup</b>	1293
See sample markup provided on	1294
<code>&lt;http://www.sisudoc.org/&gt;</code>	1295
<code>&lt;http://www.jus.uio.no/sisu&gt;</code>	1296
<code>&lt;http://www.jus.uio.no/sisu_markup&gt;</code>	1297
<code>&lt;http://www.jus.uio.no/sisu/SiSU&gt;</code>	1298
in particular for each of the document output samples provided, the source document is provided as well	1299
<code>&lt;http://www.jus.uio.no/sisu/SiSU/examples.html&gt;</code>	1300
on untarring the source tarball:	1301
<code>data/doc/sisu/markup-samples</code>	1302
or the same once source is installed (or <code>sisu-examples</code> ) under:	1303
<code>/usr/share/doc/sisu/markup-samples/</code>	1304
and if you have <code>sisu-markup-samples</code> installed, under	1305
<code>data/doc/sisu/markup-samples-non-free/</code>	1306
<code>/usr/share/doc/sisu/markup-samples-non-free/</code>	1307
Some notes are contained within the <code>man</code> page, <b>man <code>sisu</code></b> and within <code>sisu help</code> via the commands <b><code>sisu help markup</code></b> and <b><code>sisu help headers</code></b>	1308
<b>SiSU</b> is for literary and legal text, also for some social science material. In particular it does not do formula, and is not particularly suited to technical documentation. Despite the latter caveat, some notes will be provided here and added to over time:	1309
<b>40.1 Headers</b>	1310
Headers <code>@headername:</code> provide information related to the document, this may relate to	1311
1. how it is to be processed, such as whether headings are to be numbered, what skin is to be used and markup instructions, such as the document structure, or words to be made bold within the document	1312
2. semantic information about the document including the dublin core	1313
<b>40.2 Font Face</b>	1314
Defaults are set. You may change the face to: bold, italics, underscore, strikethrough, ...	1315

<b>40.2.1 Bold</b>	1316
bold: [list of words that should be made bold within document]	1317
<b>bold line</b>	1318
!_ bold line	1319
<b>bold word or sentence</b>	1320
!{ bold word or sentence }!	1321
*{ bold word or sentence }*	1322
*boldword* or !boldword!	1323
boldword or !boldword!	1324
<b>40.2.2 Italics</b>	1325
italics: [list of words that should be italicised within document]	1326
<i>italicise word or sentence</i>	1327
/ { italicise word or sentence } /	1328
/italicisedword/	1329
/italicisedword/	1330
<b>40.2.3 Underscore</b>	1331
<u>underscore word or sentence</u>	1332
_{ underscore word or sentence }_	1333
_underscoreword_	1334
<b>40.2.4 Strikethrough</b>	1335
<del>strikethrough word or sentence</del>	1336
_{ strikethrough word or sentence }_	1337
-strikeword-	1338
-strikeword-	1339
<b>40.3 Endnotes</b>	1340
There are two forms of markup for endnotes, they cannot be mixed within the same document	1341
here <sup>109</sup>	1342

1. preferred endnote markup 1343  
 here~{ this is an endnote }~ 1344
2. alternative markup equivalent, kept because it is possible to search and replace to get markup in existing texts such as Project Gutenberg 1345  
 here~^ 1346  
 ^ ~ this is an endote 1347

#### 40.4 Links 1348

<SiSU> 1349

{ SiSU }http://www.sisudoc.org 1350



1351

{sisu.png }http://www.sisudoc.org 1352



1353

{ tux.png 64x80 }image 1354

<SiSU> <sup>110</sup> 1355

{^^ SiSU }http://www.sisudoc.org 1356

is equivalent to: 1357

{ SiSU }http://www.sisudoc.org ~{ http://www.sisudoc.org }~ 1358

the same can be done with an image: 1359



<sup>111</sup>

1360

{^^ sisu.png "SiSU" }http://www.sisudoc.org 1361

#### 40.5 Number Titles 1362

Set with the header @markup: 1363

<b>40.6 Line operations</b>	1364
Line Operations (marker placed at start of line)	1365
!_ bold line	1366
<b>bold line</b>	1367
_1 indent paragraph one level	1368
indent paragraph one level	1369
_2 indent paragraph two steps	1370
indent paragraph two steps	1371
_* bullet paragraph	1372
Γ bullet paragraph	1373
	1374
number paragraph (see headers for numbering document headings)	
1. number paragraph (see headers for numbering document headings)	1375
-# number paragraph level 2 (see headers for numbering document headings)	1376
a. number paragraph level 2 (see headers for numbering document headings)	1377
<b>40.7 Tables</b>	1378
Table markup sample	1379
<code>table{~h c3; 26; 32; 32;</code>	1380
This is a table, column1	
this would become row one of column two	
column three of row one is here	
column one row 2	
column two of row two	
column three of row two, and so on	
column one row three	
and so on	
here	
<code>}table</code>	
Alternative form of table markup	1381
<code>{t~h}</code>	1382

---

<sup>109</sup>this is an endnote

<sup>110</sup><<http://www.sisudoc.org>>

<sup>111</sup><<http://www.sisudoc.org>>

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
0	*	*	*	*	*	*	*
1	*	*	*	*	*	*	*
2	-	*	*	*	*	*	*
3	-	*	*	*	*	*	*
4	-	*	*	*	*	*	*
5	*	*	*	*	*	*	*

## 40.8 Grouped Text

1383

```

5.times { puts 'Ruby' }
code{
5.times { puts 'Ruby' }
}code

```

1384

1385

1386

1387

1388

A Limerick

1389

There was a young lady from Clyde,  
 who ate a green apple and died,  
 but the apple fermented inside the lamented,  
 and made cider inside her inside.

```
poem{
```

1390

```

There was a young lady from Clyde,
who ate a green apple and died,
but the apple fermented inside the lamented,
and made cider inside her inside.

```

```
}poem
```

## 40.9 Composite Document

1391

To import another document, the master document or importing document should be named filename.r3 (r for require)

1392

```
ii { filename.sst }
```

1393

```
ii { filename.ssi }
```

1394

## 41. Change Appearance 1395

### 41.1 Skins 1396

“Skins” may be used to change various aspects related to the output documents appearance, including such things as the url for the home page on which the material will be published, information on the credit band, and for html documents colours and icons used in navigation bars. Skins are ruby files which permit changing of the default values set within the program for **SiSU** output. 1397

There are a few examples provided, on untarring the source tarball: 1398

`conf/sisu/skin/doc/` 1399

`data/doc/sisu/markup-samples/samples/_sisu/skin/doc/` 1400

and on installation under: 1401

`/etc/sisu/skin/doc/` 1402

`/usr/share/doc/sisu-markup-samples/v2/samples/_sisu/skin/doc` 1403

The following paths are searched: 1404

`./_sisu/skin` 1405

`~/_sisu/skin` 1406

`/etc/sisu/skin` 1407

Skins under the searched paths in a per document directory, a per directory directory, or a site directory, named: 1408

`doc` [may be specified individually in each document] 1409

`dir` [used if identifier part of name matches markup directory name] 1410

`site` 1411

It is usual to place all skins in the document directory, with symbolic links as required from `dir` or `site` directories. 1412

### 41.2 CSS 1413

The appearance of html and XML related output can be changed for an output collection directory by preparing and placing a new css file in one of the `sisu css` directories searched in the `sisu` configuration path. These are located at: 1414

`./_sisu/css` 1415

`~/_sisu/css` 1416

and 1417

`/etc/sisu/css` 1418

The contents of the first directory found in the search path are copied to the corresponding `sisu` output directory with the command: 1419

sisu -CC	1420
The <b>SiSU</b> standard css files for <b>SiSU</b> output are:	1421
dom.css html.css html_tables.css index.css sax.css xhtml.css	1422
A document may specify its own/bespoke css file using the css header.	1423
css:	1424
[expand]	1425

## Extracts from the README

1426

<b>42. README</b>	1427
<b>SiSU 0.55 2007w27/6 2007-07-07</b>	1428
Homepage: < <a href="http://www.sisudoc.org">http://www.sisudoc.org</a> >	1429
old homepage: < <a href="http://www.jus.uio.no/sisu">http://www.jus.uio.no/sisu</a> >	1430
<b>Description</b>	1431
<b>SiSU</b> is lightweight markup based document creation and publishing framework that is controlled from the command line. Prepare documents for <b>SiSU</b> using your text editor of choice, then use <b>SiSU</b> to generate various output document formats.	1432
With minimal preparation of a plain-text (UTF-8) file using its native markup-syntax, <b>SiSU</b> produces: plain-text, HTML, XHTML, EPUB, XML, ODF:ODT (Opendocument), LaTeX, PDF, and populates an SQL database (PostgreSQL or SQLite) in paragraph sized chunks so that document searches are done at this “atomic” level of granularity.	1433
Outputs share a common citation numbering system, and any semantic meta-data provided about the document.	1434
<b>SiSU</b> also provides concordance files, document content certificates and manifests of generated output.	1435
<b>SiSU</b> takes advantage of well established open standard ways of representing text, and provides a bridge to take advantage of the strengths of each, while remaining simple. <b>SiSU</b> implements across document formats a “useful common feature set” [coming from a humanities, law, and possibly social sciences perspective, rather than technical or scientific writing] ... focus is primarily on content and data integrity rather than appearance, (though outputs in the various formats are respectable).	1436
A vim syntax highlighting file and an ftplugin with folds for sisu markup is provided. Vim 7 includes syntax highlighting for <b>SiSU</b> .	1437
man pages, and interactive help are provided.	1438
Dependencies for various features are taken care of in sisu related packages. The package sisu-complete installs the whole of <b>SiSU</b> .	1439
Additional document markup samples are provided in the package sisu-markup-samples which is found in the non-free archive the licenses for the substantive content of the marked up documents provided is that provided by the author or original publisher.	1440
Homepage: < <a href="http://www.sisudoc.org">http://www.sisudoc.org</a> >	1441
old homepage: < <a href="http://www.jus.uio.no/sisu">http://www.jus.uio.no/sisu</a> >	1442
<b>SiSU</b> - simple information structuring universe, is a publishing tool, document generation and management, (and search enabling) tool primarily for literary, academic and legal published works.	1443
<b>SiSU</b> can be used for Internet, Intranet, local filesystem or cd publishing.	1444
<b>SiSU</b> can be used directly off the filesystem, or from a database.	1445

SiSU 's scalability, is be dependent on your hardware, and filesystem (in my case Reiserfs), and/or database Postgresql. 1446

Amongst it's characteristics are: 1447

Γ simple mnemonic markup style, 1448

Γ the ability to produce multiple output formats, including html, structured XML, LaTeX, pdf (via LaTeX), stream to a relational database whilst retaining document structure - Postgresql and Sqlite, 1449

Γ that all share a common citation system (a simple idea from which much good), possibly most exciting, the following: if fed into a relational database (as it can be automatically), the document set is searchable, with results displayed at a paragraph level, or the possibility of an indexed display of documents in which the match is found together with a hyperlinked listing for each of each paragraph in which the match is found. In any event citations using this system (with or without the relational database) are relevant for all output formats. 1450

Γ it is command line driven, and can be set up on a remote server 1451

Γ Documents are marked up in SiSU syntax in your favourite editor. SiSU syntax may be regarded as a type of smart ascii - which in its basic form is simpler than the most elementary html. There is currently a syntax highlighter, and folding for Vim. Syntax highlighters for other editors are welcome. 1452

Input files should be UTF-8 1453

Once set up it is simple to use. 1454

## 42.1 Online Information, places to look 1455

<<http://www.sisudoc.org>> 1456

<<http://www.jus.uio.no/sisu>> 1457

Download Sources: 1458

<<http://www.jus.uio.no/sisu/SiSU/download.html#current>> 1459

<<http://www.jus.uio.no/sisu/SiSU/download.html#debian>> 1460

## 42.2 Installation 1461

NB. Platform is Unix / Linux. 1462

### 42.2.1 Debian 1463

If you use Debian use the Debian packages, check the information at: 1464

<<http://www.jus.uio.no/sisu/SiSU/download.html#debian>> 1465

(A) **SiSU** is available directly off the **Debian** archives for Sid and testing. It should necessary only to run as root: 1466

```
aptitude update 1467
```

```
aptitude install sisu-complete 1468
```

(B) If there are newer versions of **SiSU** upstream of the **Debian** archives, they will be available by adding the following to your `/etc/apt/sources.list` 1469

```
deb <http://www.jus.uio.no/sisu/archive> unstable main non-free 1470
```

```
deb-src <http://www.jus.uio.no/sisu/archive> unstable main non-free 1471
```

[the non-free line is for document markup samples, for which the substantive text is provided under the author or original publisher's license and which in most cases will not be debian free software guideline compliant] 1472

Then as root run: 1473

```
aptitude update 1474
```

```
aptitude install sisu-complete 1475
```

### 42.2.2 RPM 1476

RPMs are provided though untested, they are prepared by running `alien` against the source package, and against the debs. 1477

They may be downloaded from: 1478

```
<http://www.jus.uio.no/sisu/SiSU/download.html#rpm> 1479
```

### 42.2.3 Source package .tgz 1480

Otherwise to install **SiSU** from source, check information at: 1481

```
<http://www.jus.uio.no/sisu/SiSU/download.html#current> 1482
```

alternative modes of installation from source are provided, `setup.rb` (by Minero Aoki), `rake` (by Jim Weirich) built install file, `rant` (by Stefan Lang) built install file, 1483

**Ruby** is the essential dependency for the basic operation of **SiSU** 1484

1. Download the latest source (information available) from: 1485

```
<http://www.jus.uio.no/sisu/SiSU/download.html#current> 1486
```

2. Unpack the source 1487

Note however, that additional external package dependencies, such as `texlive` or `postgresql` should you desire to use it are not taken care of for you. 1488

- 42.2.4 to use setup.rb** 1489
- this is a three step process, in the root directory of the unpacked **SiSU** as root type: 1490
- ruby setup.rb config 1491
  - ruby setup.rb setup 1492
- as root: 1493
- ruby setup.rb install 1494
- further information: 1495
- <<http://i.loveruby.net/en/projects/setup/>> 1496
  - <<http://i.loveruby.net/en/projects/setup/doc/usage.html>> 1497
- 42.2.5 to use install (prpared with “Rake”)** 1498
- Rake must be installed on your system: 1499
- <<http://rake.rubyforge.org/>> 1500
  - <[http://rubyforge.org/frs/?group\\_id=50](http://rubyforge.org/frs/?group_id=50)> 1501
- in the root directory of the unpacked **SiSU** as root type: 1502
- rake 1503
- or 1504
- rake base 1505
- This makes use of Rake (by Jim Weirich) and the provided Rakefile 1506
- For a list of alternative actions you may type: 1507
- rake help 1508
  - rake -T 1509
- 42.2.6 to use install (prpared with “Rant”)** 1510
- (you may use the instructions above for rake substituting rant if rant is installed on your 1511  
system, or you may use an independent installer created using rant as follows:)
- in the root directory of the unpacked **SiSU** as root type: 1512
- ruby ./sisu-install 1513
- or 1514
- ruby ./sisu-install base 1515

This makes use of Rant (by Stefan Lang) and the provided Rantfile. It has been configured to do post installation setup configuration and generation of first test file. Note however, that additional external package dependencies, such as tetex-extra are not taken care of for you. 1516

further information: 1517

<<http://make.rubyforge.org/>> 1518

<[http://rubyforge.org/frs/?group\\_id=615](http://rubyforge.org/frs/?group_id=615)> 1519

For a list of alternative actions you may type: 1520

ruby ./sisu-install help 1521

ruby ./sisu-install -T 1522

### 42.3 Dependencies 1523

Once installed see ‘man 8 sisu’ for some information on additional programs that sisu makes use of, and that you may need or wish to install. (this will depend on such factors as whether you want to generate pdf, whether you will be using **SiSU** with or without a database, ...) ‘man sisu-markup-samples’ may also be of interest if the sisu-markup-samples package has also been installed. 1524

The information in man 8 may not be most up to date, and it is possible that more useful information can be gleaned from the following notes taken from the **Debian** control file (end edited), gives an idea of additional packages that **SiSU** can make use of if available, (the use/requirement of some of which are interdependent for specific actions by **SiSU** ) 1525

The following is from the debian/control file of sisu-3.0.2, which amongst other things provides the dependencies of sisu within **Debian** . 1526

```
Source: sisu 1527
Section: text
Priority: optional
Maintainer: SiSU Project <sisu@lists.sisudoc.org>
Uploaders: Ralph Amissah <ralph@amissah.com>
Build-Depends: debhelper (>= 8)
Standards-Version: 3.9.1
Homepage: http://www.sisudoc.org/
Vcs-Browser: http://git.sisudoc.org/?p=code/sisu.git
Vcs-Git: git://git.sisudoc.org/git/code/sisu.git
XS-Dm-Upload-Allowed: yes

Package: sisu 1528
Architecture: all
Depends: ${misc:Depends}, ruby (>= 1.8.2), libwebrick-ruby, rsync, unzip,
zip
Recommends:
```

sisu-pdf, sisu-sqlite, sisu-postgresql, imagemagick, keychain,  
librmagick-ruby,  
librexml-ruby, openssl, openssh-client | lsh-client, tidy,  
vim-addon-manager  
Suggests: lv, calibre, pinfo, texinfo, trang  
Conflicts: sisu-markup-samples (<= 1.0.11)  
Replaces: sisu-markup-samples (<= 1.0.11)  
Description: documents - structuring, publishing in multiple formats and  
search

SiSU is a lightweight markup based, command line oriented, document  
structuring, publishing and search framework for document collections.

.  
With minimal preparation of a plain-text, (UTF-8) file, using its  
native  
markup syntax in your text editor of choice, SiSU can generate various  
document formats (most of which share a common object numbering system  
for  
locating content), including plain text, HTML, XHTML, XML, EPUB,  
OpenDocument  
text (ODF:ODT), LaTeX, PDF files, and populate an SQL database with  
objects  
(roughly paragraph-sized chunks) so searches may be performed and  
matches  
returned with that degree of granularity: your search criteria is met by  
these  
documents and at these locations within each document. Object numbering  
is  
particularly suitable for "published" works (finalized texts as opposed  
to  
works that are frequently changed or updated) for which it provides a  
fixed  
means of reference of content. Document outputs also share semantic  
meta-data  
provided.

.  
SiSU also provides concordance files, document content certificates  
and  
manifests of generated output.  
.  
A vim syntax highlighting file and an ftplugin with folds for sisu markup  
is  
provided, as are syntax highlighting files for kate, kwrite, gedit and  
diakonos. Vim 7 includes syntax highlighting for SiSU.

.  
man pages, and interactive help are provided.

.  
Dependencies for various features are taken care of in sisu related

packages.

The package `sisu-complete` installs the whole of SiSU.

Additional document markup samples are provided in the package `sisu-markup-samples` which is found in the non-free archive the licenses for

the substantive content of the marked up documents provided is that provided

by the author or original publisher.

Package: `sisu-complete`

1529

Architecture: `all`

Depends:

`${misc:Depends}`, `ruby (>= 1.8.2)`, `sisu (= ${source:Version})`,

`sisu-pdf (= ${source:Version})`, `sisu-postgresql (= ${source:Version})`,

`sisu-sqlite (= ${source:Version})`

Description: `installs all SiSU related packages`

SiSU is a lightweight markup based document structuring, publishing and search

framework for document collections.

This package installs SiSU and related packages that enable `sisu` to produce

pdf and to populate PostgreSQL and sqlite databases.

See `sisu` for a description of the package.

Package: `sisu-pdf`

1530

Architecture: `all`

Depends:

`${misc:Depends}`, `sisu`, `texlive-latex-base`, `texlive-fonts-recommended`,  
`texlive-latex-recommended`, `texlive-latex-extra`, `texlive-xetex`,

`lmodern`,

`ttf-liberation`

Suggests: `evince | pdf-viewer`

Description: `dependencies to convert SiSU LaTeX output to pdf`

SiSU is a lightweight markup based document structuring, publishing and search

framework for document collections.

This package enables the conversion of SiSU LaTeX output to pdf.

Package: `sisu-pdf`

1531

Architecture: `all`

Depends:

`${misc:Depends}`, `sisu`, `texlive-latex-base`, `texlive-fonts-recommended`,  
`texlive-latex-recommended`, `texlive-latex-extra`, `texlive-xetex`,

`lmodern`,

ttf-liberation

Suggests: evince | pdf-viewer

Description: dependencies to convert SiSU LaTeX output to pdf

SiSU is a lightweight markup based document structuring, publishing and search

framework for document collections.

.

This package enables the conversion of SiSU LaTeX output to pdf.

Package: sisu-postgresql

1532

Architecture: all

Depends:

`{misc:Depends}`, sisu, libdbd-pg-ruby, libdbd-pg-ruby1.8, libdbi-ruby, libdbi-ruby1.8, libdbm-ruby, postgresql, libfcgi-ruby1.8 |

libfcgi-ruby1.9.1

Suggests: postgresql-contrib

Description: SiSU dependencies for use with PostgreSQL database

SiSU is a lightweight markup based document structuring, publishing and search

framework for document collections.

.

This package enables SiSU to populate a PostgreSQL database. This is done at

an object/paragraph level, making granular searches of documents possible.

.

This relational database feature of SiSU is not required but provides interesting possibilities, including that of granular searches of documents

for matching units of text, primarily paragraphs that can be displayed or

identified by object citation number, from which an index of documents matched and each matched paragraph within them can be displayed.

Package: sisu-sqlite

1533

Architecture: all

Depends:

`{misc:Depends}`, sisu, sqlite3, libsqlite3-ruby, libdbd-sqlite3-ruby, libdbd-sqlite3-ruby1.8, libdbi-ruby, libdbi-ruby1.8, libdbm-ruby, libfcgi-ruby1.8 | libfcgi-ruby1.9.1

Description: SiSU dependencies for use with SQLite database

SiSU is a lightweight markup based document structuring, publishing and search

framework for document collections.

.

This package enables SiSU to populate an SQLite database. This is done at an

object/paragraph level, making granular searches of documents possible.

This relational database feature of SiSU is not required but provides interesting possibilities, including that of granular searches of documents for matching units of text, primarily paragraphs that can be displayed or identified by object citation number, from which an index of documents matched and each matched paragraph within them can be displayed.

#### 42.4 Quick start

Most of the installation should be taken care of by the `aptitude` or `rant install`. (The `rant install` if run in full will also test run the generation of the first document).

After installation of `sisu-complete`, move to the document samples directory

```
cd /usr/share/doc/sisu/markup-samples/samples
```

and run

```
sisu -3 free_as_in_freedom.rms_and_free_software.sam_williams.sst
```

or the same:

```
sisu -NhwpoabxXyv free_as_in_freedom.rms_and_free_software.sam_williams.sst
```

look at output results, see the “`sisu_manifest`” page created for the document

or to generate an online document move to a writable directory, as the file will be downloaded there and e.g.

```
sisu -3 <http://www.jus.uio.no/sisu/src/free_culture.lawrence_lessig.sst>
```

the database stuff is extra perhaps, the latex stuff could be considered extra perhaps but neither needs to be installed for most of `sisu` output to work

examine source document, `vim` has syntax support

```
gvim free_as_in_freedom.rms_and_free_software.sam_williams.sst
```

additional markup samples in

```
<http://www.jus.uio.no/sisu/SiSU/examples.html>
```

For help

```
man sisu
```

or

```
sisu -help
```

e.g.

for the way `sisu` “sees/maps” your system

```
sisu -help env
```

for list of commands and so on

```
sisu -help commands
```

<b>42.5 Configuration files</b>	1559
The default configuration/setup is contained within the program and is altered by configuration settings in <code>/etc/[sisu version]/sisurc.yml</code> or in <code>~/.sisu/sisurc.yml</code>	1560
Γ configuration file - a yaml file	1561
<code>/etc/sisu/[sisu version]/sisurc.yml</code>	1562
<code>~/.sisu/sisurc.yml</code>	1563
Γ directory structure - setting up of output and working directory.	1564
* skins - changing the appearance of a project, directory or individual documents within <code>~/.sisu/skin</code>	1565
<code>~/.sisu/skin/doc</code> contains individual skins, with symbolic links from	1566
<code>~/.sisu/skin/dir</code> if the contents of a directory are to take a particular document skin.	1567
Γ additional software - eg. Tex and LaTeX (tetex, tetex-base, tetex-extra on <b>Debian</b> ), PostgreSQL, [sqlite], trang, tidy, makeinfo, ... none of which are required for basic html or XML processing.	1568
Γ if you use Vim as editor there is a syntax highlighter and fold resource config file for <b>SiSU</b> . I hope more syntax highlighters follow.	1569
There are post installation steps (which are really part of the overall installation)	1570
sisu -C in your marked up document directory, should do some auto-configuring provided you have the right permissions for the output directories. (and provided the output directories have already been specified if you are not using the defaults).	1571
<b>42.6 Use General Overview</b>	1572
Documents are marked up in <b>SiSU</b> syntax and kept in an ordinary text editable file, named with the suffix <code>.sst</code> , or <code>.ssm</code>	1573
Marked up <b>SiSU</b> documents are usually kept in a sub-directory of your choosing	1574
use the interactive help and man pages	1575
<code>sisu -help</code>	1576
<code>man sisu</code>	1577
<b>42.7 Help</b>	1578
interactive help described below, or man page:	1579
<code>man sisu</code>	1580
<code>man 8 sisu</code>	1581

‘man sisu\_markup-samples’ [if the sisu-markup-samples package is also installed] 1582

Once installed an interactive help is available typing ‘sisu’ (without) any flags, and select an option: 1583

sisu 1584

alternatively, you could type e.g. 1585

sisu -help commands 1586

sisu -help env 1587

sisu -help headers 1588

sisu -help markup 1589

sisu -help headings 1590

etc. 1591

for questions about mappings, output paths etc. 1592

sisu -help env 1593

sisu -help path 1594

sisu -help directory 1595

## 42.8 Directory Structure 1596

Once installed, type: 1597

sisu -help env 1598

or 1599

sisu -V 1600

## 42.9 Configuration File 1601

The defaults can be changed via **SiSU**’s configure file sisurc.yml which the program expects to find in ./\_sisu ~/\_sisu or /etc/sisu (searched in that order, stopping on the first one found) 1602

## 42.10 Markup 1603

See man pages. 1604

man sisu 1605

man 8 sisu 1606

Once installed there is some information on **SiSU** Markup in its help: 1607

sisu -help markup 1608

and 1609

sisu -help headers	1610
Sample marked up document are provided with the download tarball in the directory:	1611
./data/doc/sisu/markup-samples/	1612
These are installed on the system usually at:	1613
/usr/share/doc/sisu/markup-samples/	1614
More markup samples are available in the package sisu-markup-samples, which if installed should be available at:	1615
/usr/share/doc/sisu/markup-samples-non-free/	1616
Many more are available online off:	1617
< <a href="http://www.jus.uio.no/sisu/SiSU/examples.html">http://www.jus.uio.no/sisu/SiSU/examples.html</a> >	1618
<b>42.11 Additional Things</b>	1619
There is syntax support for some editors provided (together with a README file) in	1620
./data/sisu/v2/conf/editor-syntax-etc	1621
usually installed to:	1622
/usr/share/sisu/v2/conf/editor-syntax-etc	1623
<b>42.12 License</b>	1624
License: GPL 3 or later see the copyright file in	1625
./data/doc/sisu	1626
usually installed to:	1627
/usr/share/doc/sisu	1628
<b>42.13 SiSU Standard</b>	1629
<b>SiSU</b> uses:	1630
Γ Standard <b>SiSU</b> markup syntax,	1631
Γ Standard <b>SiSU</b> meta-markup syntax, and the	1632
Γ Standard <b>SiSU</b> object citation numbering and system	1633
© Ralph Amissah 1997, current 2006 All Rights Reserved.	1634
Γ however note the License section	1635
<b>CHANGELOG</b>	1636
./CHANGELOG	1637

and see	1638
< <a href="http://www.jus.uio.no/sisu/SiSU/changelog.html">http://www.jus.uio.no/sisu/SiSU/changelog.html</a> >	1639
< <a href="http://www.jus.uio.no/sisu/SiSU/changelog_markup_samples.html">http://www.jus.uio.no/sisu/SiSU/changelog_markup_samples.html</a> >	1640

## Extracts from man 8 sisu

1641

## 43. Post Installation Setup 1642

### 43.1 Post Installation Setup - Quick start 1643

After installation of sisu-complete, move to the document samples directory, 1644

```
cd /usr/share/doc/sisu/markup-samples/samples 1645
```

[this is not where you would normally work but provides sample documents for testing, 1646  
you may prefer instead to copy the contents of that directory to a local directory before  
proceeding]

and in that directory, initialise the output directory with the command 1647

```
sisu -CC 1648
```

then run: 1649

```
sisu -l free_as_in_freedom.rms_and_free_software.sam_williams.sst 1650
```

or the same: 1651

```
sisu -NhwpoabxXyv free_as_in_freedom.rms_and_free_software.sam_williams.sst 1652
```

look at output results, see the “sisu\_manifest” page created for the document 1653

for an overview of your current sisu setup, type: 1654

```
sisu -help env 1655
```

or 1656

```
sisu -V 1657
```

To generate a document from a remote url accessible location move to a writable directory, 1658  
(create a work directory and cd into it) as the file will be downloaded there and e.g.

```
sisu -l <http://www.jus.uio.no/sisu/src/gpl.fsf.sst> 1659
```

```
sisu -3 <http://www.jus.uio.no/sisu/src/free_culture.lawrence_lessig.sst> 1660
```

examine source document, vim has syntax highlighting support 1661

```
gvim free_as_in_freedom.rms_and_free_software.sam_williams.sst 1662
```

additional markup samples in 1663

```
<http://www.jus.uio.no/sisu/SiSU/examples.html> 1664
```

it should also be possible to run sisu against sisupods (prepared zip files, created by 1665  
running the command `sisu -S [filename]`), whether stored locally or remotely.

```
sisu -3 <http://www.jus.uio.no/sisu/pod/free_culture.lawrence_lessig.sst.zip> 1666
```

there is a security issue associated with the running of document skins that are not your 1667  
own, so these are turned off by default, and the use of the following command, which  
switches on the associated skin is not recommended:

```
sisu -3 -trust <http://www.jus.uio.no/sisu/pod/free_culture.lawrence_lessig.sst.zip> 1668
```

For help 1669

man sisu 1670  
sisu -help 1671  
sisu -help env for the way sisu “sees/maps” your system 1672  
sisu -help commands for list of commands and so on 1673

## 43.2 Document markup directory 1674

Perhaps the easiest way to begin is to create a directory for sisu marked up documents within your home directory, and copy the file structure (and document samples) provided in the document sample directory: 1675

```
mkdir ~/sisu_test 1676  
cd ~/sisu_test 1677  
cp -a /usr/share/doc/sisu/markup-samples/samples/* ~/sisu_test/. 1678
```

better if you have installed sisu-markup-samples 1679

```
cp -a /usr/share/doc/sisu/markup-samples-non-free/samples/* ~/sisu_test/. 1680
```

**Tip:** 1681

```
sisu -U [sisu markup filename] 1682
```

should printout the different possible outputs and where sisu would place them. 1683

**Tip:** if you want to toggle ansi color add 1684

```
c 1685
```

to your flags. 1686

### 43.2.1 Configuration files 1687

**SiSU** configuration file search path is: 1688

```
./_sisu/sisurc.yaml 1689  
~/_sisu/sisurc.yaml 1690  
/etc/sisu/sisurc.yaml 1691
```

.% Debian Installation Note 1692

### 43.2.2 Debian INSTALLATION Note 1693

It is best you see 1694

```
<http://www.jus.uio.no/sisu/SiSU/download.html#debian> 1695
```

for up the most up to date information. 1696

notes taken from the **Debian** control file (end edited), gives an idea of additional packages that **SiSU** can make use of if available, (the use/requirement of some of which are interdependent for specific actions by **SiSU**) : 1697

Package: sisu 1698

**SiSU** is a lightweight markup based, command line oriented, document structuring, publishing and search framework for document collections. 1699

With minimal preparation of a plain-text, (UTF-8) file, using its native markup syntax in your text editor of choice, **SiSU** can generate various document formats (most of which share a common object numbering system for locating content), including plain text, HTML, XHTML, XML, OpenDocument text (ODF:ODT), EPUB, LaTeX, PDF files, and populate an SQL database with objects (roughly paragraph-sized chunks) so searches may be performed and matches returned with that degree of granularity: your search criteria is met by these documents and at these locations within each document. Object numbering is particularly suitable for “published” works (finalized texts as opposed to works that are frequently changed or updated) for which it provides a fixed means of reference of content. Document outputs also share semantic meta-data provided. 1700

**SiSU** also provides concordance files, document content certificates and manifests of generated output. 1701

A vim syntax highlighting file and an ftplugin with folds for sisu markup is provided, as are syntax highlighting files for kate, kwrite, gedit and diakonos. Vim 7 includes syntax highlighting for **SiSU** . 1702

man pages, and interactive help are provided. 1703

Dependencies for various features are taken care of in sisu related packages. The package sisu-complete installs the whole of **SiSU** . 1704

Additional document markup samples are provided in the package sisu-markup-samples which is found in the non-free archive the licenses for the substantive content of the marked up documents provided is that provided by the author or original publisher. 1705

Homepage: <<http://www.sisudoc.org>> 1706

old homepage: <<http://www.jus.uio.no/sisu>> 1707

### 43.2.3 Document Resource Configuration 1708

**sisu** resource configuration information is obtained from sources (where they exist): 1709

~/sisu/sisurc.yaml 1710

/etc/sisu/[sisu version]/sisurc.yaml 1711

sisu program defaults 1712

<b>43.2.4 Skins</b>	1713
<b>Skins</b> default document appearance may be modified using skins contained in sub-directories located at the following paths:	1714
./_sisu/skin	1715
~/_sisu/skin	1716
/etc/sisu/skin	1717
more specifically, the following locations (or their /etc/sisu equivalent) should be used:	1718
~/_sisu/skin/doc	1719
skins for individual documents;	1720
~/_sisu/skin/dir	1721
skins for directories of matching names;	1722
~/_sisu/skin/site	1723
site-wide skin modifying the site-wide appearance of documents.	1724
Usually all skin files are placed in the document skin directory:	1725
~/_sisu/skin/doc	1726
with softlinks being made to the skins contained there from other skin directories as required.	1727

<b>44. FAQ - Frequently Asked/Answered Questions</b>	1728
<b>44.1 Why are urls produced with the -v (and -u) flag that point to a web server on port 8081 ?</b>	1729
Try the following rune:	1730
Γ sisu -W	1731
This should start the ruby webserver. It should be done after having produced some output as it scans the output directory for what to serve.	1732
<b>44.2 I cannot find my output, where is it?</b>	1733
The following should provide help on output paths:	1734
Γ sisu -help env	1735
Γ sisu -V [same as the previous command]	1736
Γ sisu -help directory	1737
Γ sisu -help path	1738
Γ sisu -U [filename]	1739
Γ man sisu	1740
<b>44.3 I do not get any pdf output, why?</b>	1741
<b>SiSU</b> produces LaTeX and pdflatex is run against that to generate pdf files.	1742
If you use <b>Debian</b> the following will install the required dependencies	1743
Γ aptitude install sisu-pdf	1744
the following packages are required: tetex-bin, tetex-extra, latex-ucs	1745
<b>44.4 Where is the latex (or some other interim) output?</b>	1746
Try adding -M (for maintenance) to your command flags, e.g.:	1747
Γ sisu -HpMv [filename]	1748
this should result in the interim processing output being retained, and information being provided on where to find it.	1749
Γ sisu -help directory	1750
Γ sisu -help path	1751
should also provide some relevant information as to where it is placed.	1752

#### 44.5 Why isn't SiSU markup XML

1753

I worked with text and (though I find XML immensely valuable) disliked noise ... better to sidestep the question and say:

1754

**SiSU** currently “understands” three XML input representations - or more accurately, converts from three forms of XML to native **SiSU** markup for processing. The three types correspond to SAX (structure described), DOM (structure embedded, whole document must be read before structure is correctly discernable) and node based (a tree) forms of XML document structure representation. Problem is I use them very seldom and check that all is as it should be with them seldom, so I would not be surprised if something breaks there, but as far as I know they are working. I will check and add an XML markup help page before the next release. There already is a bit of information in the man page under the title **SiSU** VERSION CONVERSION

1755

```
sisu -to-sax [filename/wildcard]
```

1756

```
sisu -to-dom [filename/wildcard]
```

1757

```
sisu -to-node [filename/wildcard]
```

1758

The XML should be well formed... must check, but lacks sensible headers. Suggestions welcome as to what to make of them. [For the present time I am satisfied that I can convert (both ways) between 3 forms of XML representation and **SiSU** markup].

1759

```
sisu -from-xml2sst [filename/wildcard]
```

1760

#### 44.6 LaTeX claims to be a document preparation system for high-quality typesetting. Can the same be said about SiSU?

1761

**SiSU** is not really about type-setting.

1762

LaTeX is the ultimate computer instruction type-setting language for paper based publication.

1763

LaTeX is able to control just about everything that happens on page and pixel, position letters kerning, space variation between characters, words, paragraphs etc. formula.

1764

**SiSU** is not really about type-setting at all. It is about a lightweight markup instruction that provides enough information for an abstraction of the documents structure and objects, from which different forms of representation of the document can be generated.

1765

**SiSU** with very little markup instruction is able to produce relatively high quality pdf by virtue of being able to generate usable default LaTeX; it produces “quality” html by generating the html directly; likewise it populates an SQL database in a useful way with the document in object sized chunks and its meta-data. But **SiSU** works on an abstraction of the document's structure and content and custom builds suitable uniform output. The html for browser viewing and pdf for paper viewing/publishing are rather different things with different needs for layout - as indeed is what is needed to store information in a database in searchable objects.

1766

The pdfs or html produced for example by open office based on open document format

1767

and other office/word processor suits usually attempt to have similar looking outputs - your document rendered in html looks much the same, or in pdf... sisu is less this way, it seeks to have a starting point with as little information about appearance as possible, and to come up with the best possible appearance for each output that can be derived based on this minimal information.

Where there are large document sets, it provides consistency in appearance in each output format for the documents. 1768

The excuse for going this way is, it is a waste of time to think much about appearance when working on substantive content, it is the substantive content that is relevant, not the way it looks beyond the basic informational tags - and yet you want to be able to take advantage of as many useful different ways of representing documents as are available, and for various types of output to to be/look as good as it can for each medium/format in which it is presented, (with different mediums having different focuses) and **SiSU** tries to achieve this from minimal markup. 1769

#### **44.7 Can the SiSU markup be used to prepare for a LaTeX automatic building of an index to the work?** 1770

Has not been, is of interest though the question on introducing such possibilities is how to keep them as unobtrusive as possible, and as generically relevant as possible to other output formats (which is why the focus on object numbers). Unobtrusive refers both to the markup (where there is no big problem with introducing optional extras); and, more challengingly how to minimise impact on competing ideas/interests, such allowing the addition of semantic tags which could be tied to objects, mapped against the objects that contain them, (permitting mapping and mining of content in various ways that would be largely agnostic of output format - object numbering being an attempt to move beyond output format based content locators (such as page numbers). The desire being to (be a meta markup and) maintain agnosticism as to what is being generated and in development to favor solutions of that nature. Keep bridging LaTeX, XML, SQL ... make use of objects and serialisation for mapping whether against content or meta-content (such as semantic [or additional structural] markers). 1771

#### **44.8 Can the conversion from SiSU to LaTeX be modified if we have special needs for the LaTeX, or do we need to modify the LaTeX manually?** 1772

Should be possible to modify code, it is GPLv3, should be possible either to modify existing modules or write an independent module for generating bespoke latex. Generic improvements are welcome for inclusion/incorporation in the existing code base. 1773

If there are tools to generate mathematical/scientific formula from latex to images (jpg, png), the latex parser could conceivably be used to make these available to other output formats. 1774

#### 44.9 How do I create GIN or GiST index in Postgresql for use in SiSU 1775

This at present needs to be done “manually” and it is probably necessary to alter the sample search form. The following is a helpful response from one of the contributors of GiN to Postgresql Oleg Bartunov 2006-12-06: 1776

“I have tsearch2 slides which introduces tsearch2 <<http://www.sai.msu.su/~megeera/wiki/tsearch2slides>> 1777

FTS in PostgreSQL is provided by tsearch2, which should work without any indices (GiST or GIN) ! Indices provide performance, not functionality. 1778

In your example I'd do ( simple way, just for demo): 1779

0. compile, install tsearch2 and load tsearch2 into your database 1780

```
cd contrib/tsearch2; make&&make&&install&&make installcheck; psql DB j tsearch2.sql 1781
```

1. Add column fts, which holds tsvector 1782

```
alter table documents add column fts tsvector; 1783
```

2. Fill fts column 1784

```
update document set fts = to_tsvector(clean); 1785
```

3. create index - just for performance ! 1786

```
create index fts_gin_idx on document using gin(fts); 1787
```

4. Run vacuum 1788

```
vacuum analyze document; 1789
```

That's all. 1790

Now you can search: 1791

```
select lid, metadata_tid, rank_cd(fts, q,2)as rank from document, plainto_tsquery('markup syntax') q where q @@ fts order by rank desc limit 10; 1792
```

#### 44.10 Are there some examples of using Ferret Search with a SiSU repository? 1793

Heard good things about Ferret, but have not used it. The output directory structure and content produced by **SiSU** is very uniform. Have looked at a couple of other engines (hyperestraier, lucene). There it was enough to identify the files that needed to be indexed and pass them to the search indexing tool. Some Unix runde doing the job, such as: 1794

```
find /home/ralph/sisu_www -type f | \ 1795
egrep '/sisu_www/(sisu|document_archive)/.+?.html$' | \
egrep -v '(doc|concordance).html$' | \
estcmd gather -sd casket -
```

you would have to experiment with what gives the desired result, the file doc.html is the complete text in html (there are additional smaller html segments), and plain.txt the document as a text file. It may be possible to index the text file and return the html document. 1796

**Have you had any reports of building SiSU from tar on Mac OS 10.4?** 1797

None. In the early days of its release a Mac friend built and run the ruby code part that did not rely on system calls to bits like the latex engine. That is already some years back. He was not into writing or document markup, and did it as a favour at the time. I have not followed up that thread of development. 1798

It should however be possible, much of the output relies on plain ruby, and the system commands to latex etc. could be made appropriate for the underlying OS. 1799

**44.12 Where is version 1?** 1800

Version 1 was finally released on December 21, 2009, and largely to make a version 1 branch available as version 2 was in the pipeline with an imminent release. Most of **SiSU** was mature and stable long before the release of version 1. 1801

**44.13 What is the difference between version 1 and 2?** 1802

Input and output of version 1 and 2 are largely the same with the following significant exceptions. On the input side, document headers, that is metadata and processing instructions have changed in version 2. On the output side, version 2 was introduced with EPUB documents as a possible output, and over time is likely to be developed further. 1803

Version 2 introduces a new processing layer, which relies more on the programming language **Ruby** 's objects (and regular expressions), than version 1 does which relies on regular expressions. The thinking behind version 1 use of regular expressions was that it made it more straightforward to switch languages for processing, as many languages support regular expressions; the thinking behind version 2 was version 1 was more complicated than it need be, and since **Ruby** was the language used, why not make programming more straightforward, as it would be easier to develop further. 1804

Version 1 was removed with the introduction of the version 3 development branch 1805

**45. Who might be interested in the SiSU feature set?**

1806

**SiSU** is most likely to be of interest to people who are working with medium to large volumes of published texts that would like to have the presented in a uniform way that is searchable (either using sisu database integration or an appropriate indexing tool), with the possibility of multiple alternative output formats that may be added to and upgraded/updated over time. **SiSU** should be of interest to institutions/ organisations/ governments/ individuals with document collections and some technical knowhow that are interested in:

1807

Γ long term maintenance and reducing downstream/future costs of maintaining those document sets for which **SiSU** is suited.

1808

Γ the ability to output multiple standard format outputs for various purposes.

1809

Γ the implications for search offered

1810

## 46. Work Needed

1811

**SiSU** is fairly mature and for most purposes the syntax and what it is supposed to do is clear. For the most part additions and changes are minor and backward compatible, (in particular there may be things of interest that to be able to achieve will require additions to the syntax).

1812

Γ Amongst the most requested features is a way to represent and extract bibliographies from scholarly and other writings. This involves an extension of sisu markup syntax and a new module to extract the bibliography.

1813

Γ Integration of postgresql tsearch2 / gin indexing, (which currently needs to be done manually, and) which has been waiting for the integration of tsearch2 / gin into Postgresql main, which is supposed to occur in Postgresql 8.3

1814

Γ Internationalisation always. **SiSU** is utf-8 and for those parts that are utf-8 friendly will work out of the box - html and postgresql for example work out of the box (and for example comfortably represent Chinese text), LaTeX and odf do not work out of the box, they need additional work for extended language sets.

1815

Γ Refinements and improvements to output representations, some are fairly mature, others (such as manpages and info files (and even ODF) remain young.

1816

Γ Simple extension to contain, link and share included audio and multi-media files, (including sisupod.zip)

1817

## 47. Wishlist

1818

**SiSU** provides a lot of “plumbing” and is readily usable as a tool by those comfortable with marking up documents with an editor. The syntax is fairly easy to learn, especially the subset required to start using **SiSU** effectively.

1819

**SiSU** might also be of interest to developers interested in:

1820

Γ experimenting with the search implications offered

1821

Γ producing additional output formats

1822

Γ producing conversion tools

1823

Γ producing input interfaces, (experimenting with additional interfaces for producing sisu source documents)

1824

Several tools that are of interest would come under the heading interface and conversion. Amongst others, the following are of interest:

1825

Γ Converters from various document formats, such as Open Document Text (ODF), MS Word(TM) and Word Perfect(TM), even html. The problem here is one of the most important things for **SiSU** is to be able to recognise the structure of a document, and many documents prepared in other formats have not been prepared strictly with a view to representing structure, but appearance - so heading levels may be “painted” to look right rather than have the correct structural representation. Even if conversion is not perfect this may serve as a first step in assisting in conversion of documents to **SiSU** for those with legacy document sets that they would like to have in sisu format. (once in **SiSU** it is easier to get out in various other formats as this is what sisu does, within the constraints of the information that sisu uses to generate output)

1826

Γ The possibility to save directly from various word processors, and possibly templates within them to assist in making sure the document structure is “understood” by **SiSU** .

1827

Γ Web interface/front-end, a form like front end for the writing or submission of sisu documents to a server which uses **SiSU** to generate output. Headers could be made available as separate small entry forms with help provided to explain where they might be used. Apart from the most important headers such as title, author, date and possibly subject the remainder of the header forms could be placed after the form for substantive content. This would offer a more Web 2.0 like approach to the use of **SiSU** and the possibility of using it for collaborative editing of content (possibly for documents that are to be finalised/published as the citation system is most suited to published works). [Collaborative editing is currently possible through use of a collaborative editor such as Gobby which makes use of the Obby protocol].

1828

## 48. Editor Files, Syntax Highlighting 1829

The directory: 1830

`./data/sisu/v2/conf/editor-syntax-etc/` 1831

`./data/sisu/v3/conf/editor-syntax-etc/` 1832

`/usr/share/sisu/v2/conf/editor-syntax-etc` 1833

`/usr/share/sisu/v3/conf/editor-syntax-etc` 1834

contains rudimentary sisu syntax highlighting files for: 1835

Γ (g)vim <<http://www.vim.org>> 1836

package: sisu-vim 1837

status: largely done 1838

there is a vim syntax highlighting and folds component 1839

Γ gedit <<http://www.gnome.org/projects/gedit>> 1840

Γ gobby <<http://gobby.0x539.de/>> 1841

file: sisu.lang 1842

place in: 1843

`/usr/share/gtksourceview-1.0/language-specs` 1844

or 1845

`~/gnome2/gtksourceview-1.0/language-specs` 1846

status: very basic syntax highlighting 1847

comments: this editor features display line wrap and is used by Goby! 1848

Γ nano <<http://www.nano-editor.org>> 1849

file: nanorc 1850

save as: 1851

`~/nanorc` 1852

status: basic syntax highlighting 1853

comments: assumes dark background; no display line-wrap; does line breaks 1854

Γ diakonos (an editor written in ruby) <<http://purepistos.net/diakonos>> 1855

file: diakonos.conf 1856

save as: 1857

`~/diakonos/diakonos.conf` 1858

includes: 1859

status: basic syntax highlighting 1860

comments: assumes dark background; no display line-wrap 1861

- Γ kate & kwrite <<http://kate.kde.org>> 1862
- file: sisu.xml 1863
- place in: 1864
- /usr/share/apps/katepart/syntax 1865
- or 1866
- ~/kde/share/apps/katepart/syntax 1867
- [settings::configure kate::{highlighting,filetypes}] 1868
- [tools::highlighting::<markup,scripts>] 1869
- Γ nedit <<http://www.nedit.org>> 1870
- file: sisu\_nedit.pats 1871
- nedit -import sisu\_nedit.pats 1872
- status: a very clumsy first attempt [not really done] 1873
- comments: this editor features display line wrap 1874
- Γ emacs <<http://www.gnu.org/software/emacs/emacs.html>> 1875
- files: sisu-mode.el 1876
- to file ~/.emacs add the following 2 lines: 1877
- (add-to-list 'load-path "/usr/share/sisu/v2/conf/editor-syntax-etc/emacs") 1878
- (require 'sisu-mode.el) 1879
- [not done / not yet included] 1880
- Γ vim & gvim <<http://www.vim.org>> 1881
- files: 1882
- package is the most comprehensive sisu syntax highlighting and editor environment 1883  
provided to date (is for vim/ gvim, and is separate from the contents of this directory)
- status: this includes: syntax highlighting; vim folds; some error checking 1884
- comments: this editor features display line wrap 1885
- NOTE: 1886
- [SiSU parses files with long lines or line breaks, but, display linewidth (without line- 1887  
breaks) is a convenient editor feature to have for sisu markup]

---

<b>49. Help Sources</b>	1888
<b>49.1 man pages</b>	1889
man sisu	1890
man sisu-concordance	1891
man sisu-epub	1892
man sisu-git	1893
man sisu-harvest	1894
man sisu-html	1895
man sisu-odf	1896
man sisu-pdf	1897
man sisu-pg	1898
man sisu-po	1899
man sisu-sqlite	1900
man sisu-txt	1901
man 7 sisu_complete	1902
man 7 sisu_pdf	1903
man 7 sisu_postgresql	1904
man 7 sisu_sqlite	1905
man sisu_termsheet	1906
man sisu_webrick	1907
<b>49.2 sisu generated output - links to html</b>	1908
Note <b>SiSU</b> documentation is prepared in <b>SiSU</b> and output is available in multiple formats including amongst others html, pdf, odf and epub, which may be also be accessed via the html pages <sup>112</sup>	1909
<b>49.2.1 www.sisudoc.org</b>	1910
< <a href="http://sisudoc.org/sisu/sisu_manual/index.html">http://sisudoc.org/sisu/sisu_manual/index.html</a> >	1911
< <a href="http://sisudoc.org/sisu/sisu_manual/index.html">http://sisudoc.org/sisu/sisu_manual/index.html</a> >	1912

<b>49.3 man2html</b>	1913
<b>49.3.1 locally installed</b>	1914
file:///usr/share/doc/sisu/html/sisu.1.html	1915
file:///usr/share/doc/sisu/html/sisu.1.html	1916
/usr/share/doc/sisu/html/sisu_pdf.7.html	1917
/usr/share/doc/sisu/html/sisu_postgresql.7.html	1918
/usr/share/doc/sisu/html/sisu_sqlite.7.html	1919
/usr/share/doc/sisu/html/sisu_webrick.1.html	1920
<b>49.3.2 www.jus.uio.no/sisu</b>	1921
<http://www.jus.uio.no/sisu/man/sisu.1.html>	1922
<http://www.jus.uio.no/sisu/man/sisu.1.html>	1923
<http://www.jus.uio.no/sisu/man/sisu_complete.7.html>	1924
<http://www.jus.uio.no/sisu/man/sisu_pdf.7.html>	1925
<http://www.jus.uio.no/sisu/man/sisu_postgresql.7.html>	1926
<http://www.jus.uio.no/sisu/man/sisu_sqlite.7.html>	1927
<http://www.jus.uio.no/sisu/man/sisu_webrick.1.html>	1928

---

<sup>112</sup>named index.html or more extensively through sisu\_manifest.html

## Metadata

### SiSU Metadata, document information

**Document Manifest @:**

<[http://www.sisudoc.org/sisu/sisu\\_manual/sisu\\_manifest.html](http://www.sisudoc.org/sisu/sisu_manual/sisu_manifest.html)>

**Title:** SiSU - Manual

**Creator:** Ralph Amissah

**Rights:** Copyright (C) Ralph Amissah 2007, part of SiSU documentation, License GPL 3;

**Subject:** ebook, epubliishing, electronic book, electronic publishing, electronic document, electronic citation, data structure, citation systems, search

**Publisher:** SiSU <#60;text:a xlink:type='simple' xlink:href='http://www.jus.uio.no/sisu'62;http://www.jus.uio.no/sisu#60;/text:a#62;> (copy)

**Date created:** 2002-08-28

**Date issued:** 2002-08-28

**Date available:** 2002-08-28

**Date modified:** 2011-02-07

**Date:** 2008-07-21

**Topics Registered:** SiSU:manual;electronic documents:SiSU:manual

**Version Information**

**Sourcefile:** sisu\_manual.ssm.sst

**Filetype:** SiSU text insert 2.0

**Source Digest:** SHA256(sisu\_manual.ssm.sst)=6e9af9f4f6be91b82bddcaa3ed90e80f9a4a25d878cf4a64c168e41f223e246d-

**Skin Digest:** SHA256(skin\_sisu\_manual.rb)=689f79b53be3d51460af31dc82d6cbab4a253079db263cd715c8254a6af60eb8-

**Generated**

**Document (dal) last generated:** Fri Mar 11 23:48:55 +0100 2011

**Generated by:** SiSU 2.8.2 of 2011w10/5 (2011-03-11)

**Ruby version:** ruby 1.8.7 (2008-08-11 patchlevel 72) [i486-linux]