JACoW Proceedings Scripts

Installation and Execution

Installation for Windows OS

Volker RW Schaa

GSI, Darmstadt

Version 4.2 — July 2006

Contents

Changes			
Date	Version	Name	Comment
October 05	1.0	Jeff Patton	base version
March 06	1.1	John Poole	minor changes
May 06	2.0	VRW Schaa	new version: revive old TeX version which ended in 2005 leaving only fragments of code. Now incorporating several handwritten notes
June 06	2.1	VRW Schaa	completely new and updated version (added: Autorun.inf, files delivered with script, Production of abstract booklet, etc.)
July 2006	4.1	VRW Schaa	changed version number to get in synch with Matt's versioning of the »Proceedings Scripts« on http://www-esh.fnal.gov/SPMS/.

1 Documentation Guideline

The PDF file you are reading is a self-contained document: it contains all JACoW scripts and files you will need for the production of a web site, abstract booklet, proceedings volume, and a CD-ROM together with the documentation how to use them.

This documentation is organized in the following way:

❶ Setup of your system with the required software to run the JACoW scripts (↪ section 2)

❷ Description of JACoW scripts, what they do, in which sequence to run, etc. (↪ section 3)

❸ Description of additional scripts to check for consistency and correctness (↪ section 5)

❹ Additional hints for handling error conditions, etc. (↪ section 7)

❺ List of scripts and documentation with versions and change history (↪ section 8)

In each of these sections the description of tools and scripts is as detailed as necessary to understand the basic functionality. A more extensive description is given in later sections.

Note: All references of products and packages contain links to web pages. Links in »blue« are external ones, links in »red« are document internal ones. Warnings and remarks are in this color: »warning«. Due to an error in Acrobat Reader, the name of an attached file is not printed, to warn you a tag is printed in the margin.

# 2 Software needed by the JACoW proceedings scripts

Before the JACoW proceedings scripts can be executed you will need to install the following software packages.

2.1 Perl

The Perl install package from ActivePerl is recommended, it is easy to install and always up to date. It can be found at the ActivePerl web site. Select the MS windows package with MSI installer from the download page (the size is ≈13 MB, but you need ≈60 MB hard disk space for a typical installation with documentation).

When starting the Perl Setup you can select which features to install. If you want to use Perl just for the JACoW scripts, you can deselect the Documentation and Examples installation, thus saving ≈22 MB.

You can select the destination directory where Perl will be installed with Browse.
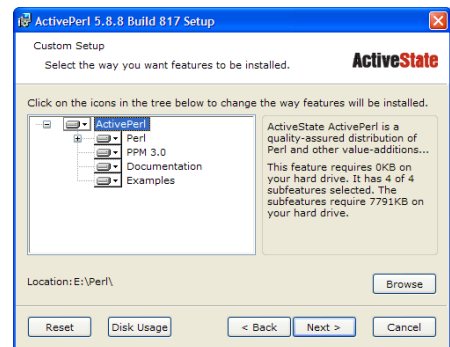
Installation Notes

Old Perl version: If you have a different Perl version or installation already on your computer, you should try first that the scripts work with your Perl version/installation. If not, make sure that you remove the installation and unset any Perl Environment Variables (like PERLLIB, PERL5LIB, or PERL5OPT) before starting the installation of ActivePerl. Otherwise, these variables may cause incompatible versions of Perl modules to be used during the installation process.

System read permission: Do not launch the installation package from a directory for which the "System" account does not have read permission. If you do, the Windows Installer Service will not be able to access the MSI file in order to perform the installation. You may get an error message or it may fail mysteriously.

Non privileged user: On Windows NT/2000/XP/2003 systems ActivePerl installations must be performed by a privileged user. If you install ActivePerl without full administrator privileges, the following problems occur:
- Environment variables are set only for the current user.
- Registry entries are created under HKEY_CURRENT_USER and not under HKEY_LOCAL_MACHINE.
- ActivePerl applications will not write messages to the event log.
- ActivePerl only appears in the Add/Remove Programs list for the current user.

In the proceedings script the special perl package »`Sort-ArbBiLex`« is needed, which is not automatically installed by ActivePerl. It is used for lexical sorting on the author names (containing accented characters and Umlauts). Loading of additional or updated perl packages is done using the Perl Package Manager (»`ppm`« or the newer version »`ppm3`«).

You do this from a command prompt window by starting »`ppm3`«.

```
E:>ppm3
PPM - Programmer's Package Manager version 3.4.
Copyright (c) 2001 ActiveState SRL. All Rights Reserved.

Entering interactive shell. Using Term::ReadLine::Stub as readline library.

Type 'help' to get started.

ppm>
```

You can search for the »`Sort-ArbBiLex`« package (while being online) by using the command »`search sort-a*`« which will show you a list of installable packages containing »`sort-a`« in the name:

```
ppm> search sort-a*
Searching in Active Repositories
  1. Sort-ArbBiLex     [4.01] Sort-ArbBiLex
  2. Sort-Array        [0.26] Sort-Array
  3. Sort-ArrayOfArrays [1.00] Sort-ArrayOfArrays
```

Package number 1 is the newest version of »`Sort-ArbBiLex`« which you can load and install with

```
ppm> install 1
Package 1:
===================
Install 'Sort-ArbBiLex' version 4.01 in ActivePerl 5.8.8.817.
===================
Downloaded 15358 bytes.
Extracting 5/5: blib/arch/auto/Sort/ArbBiLex/.exists
Installing E:\Perl\html\site\lib\Sort\ArbBiLex.html
Installing E:\Perl\site\lib\Sort\ArbBiLex.pm
Successfully installed Sort-ArbBiLex version 4.01 in ActivePerl 5.8.8.817.
ppm>
```

Now you are done with the Perl installation.

You should check that everything is setup correctly in a command prompt window by starting »`perl`«:

```
E:\EPAC06>perl -v

This is perl, v5.8.8 built for MSWin32-x86-multi-thread
(with 25 registered patches, see perl -V for more detail)

Copyright 1987-2006, Larry Wall

Binary build 817 [257965] provided by ActiveState http://www.ActiveState.com
Built Mar 20 2006 17:54:25
<...>
E:\EPAC06>
```

2.2 LaTeX

There are several TeX installation packages, the easiest to use is ProTeXt. The installation is cleverly driven by the pdf installation document (available in English, French, German, Polish, and Spanish). While reading the installation guide in Acrobat Reader you select various components, along with explanations, from a list of installable items.

You can download ProTeXt from http://www.tug.org/protext/ as a self-extracting image (size ≈410 MB). In the 2005 version ProTeXt can install:

MiKTeX   a TeX implementation for Windows (http://www.miktex.org/).

Ghostscript   an Open Source PostScript (PS) and Portable Document Format (PDF) previewer. Ghostscript itself only offers a command line interface, as GUI you need GSview (next item). See http://www.ghostscript.com/.

GSview   is a graphical user interface (GUI) for Ghostscript (see above). GSview allows selected pages to be viewed, printed, or converted to bitmap, PostScript or PDF formats. It is free to use, if you want to disable the GSview nag screen (which requires to hit return), consider registering GSview, see http://www.ghostgum.com.au/.

WinEdt   is a powerful and versatile ASCII editor and shell for MS Windows with a strong predisposition towards the creation of LaTeX/TeX documents. It integrate seamlessly with a TeX system, such as MiKTeX or TeXLive (fpTeX) (shareware, see http://www.winedt.com/).

                Note: Due to the shareware status of WinEdt, a free trial period of 31+ days starts directly after installation (try-before-you-buy software). The unregistered trial copy is fully functional and stays that way. After the trial period a nag screen to register will appear increasingly often.

TeXnicCenter is a feature rich integrated development environment (IDE) for developing LaTeX documents on Microsoft Windows (freely available under GPL, see http://www.toolscenter.org/)

MiKTeX and Ghostscript/GSview have to be installed, but you are free to take any command center (IDE) or editor you like.

You should test the installation of LaTeX by typing the following command in a command prompt window:

```
pdflatex sample2e.tex
```

The correct output using this command, version numbers and directory names may vary, important is the number of generated pages (=3) and the size (71274 bytes):

```
This is pdfeTeX, Version 3.141592-1.21a-2.2 (MiKTeX 2.4) entering extended mode
(D:\MiKTeX\texmf\tex\latex\base\sample2e.tex LaTeX2e <2005/12/01> Babel <v3.8g>
<...>
Output written on sample2e.pdf (3 pages, 71274 bytes).
Transcript written on sample2e.log.
```

A nice feature of the MiKTeX system is that it will prompt you to install that package the proceedings scripts reference which are not part of the base install. So you can install a minimum set of packages:
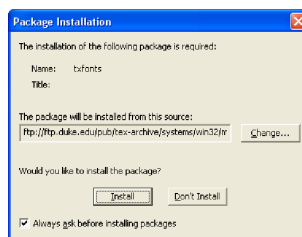

Figure 1: Loading of a required package in MiKTeX

2.3 File Transfer Utility: `wget`

You need a file transfer utility to download the XML output from SPMS in an encoding save way. Starting the SPMS script (`http://<your_spms_address>/spms.xml`, e.g. `http://oraweb.cern.ch/pls/epac06/spms.xml`) in a browser window could lead to obscured character sequences and really big files which might even crash Internet Explorer.

There are more candidates for file transfer than `wget`, but in this readme only `wget` is described. If you are more familiar with a different file transfer utility you savely can use it.

GNU `wget` is a free software package for retrieving files using `HTTP`, `HTTPS` and `FTP`, the most widely-used Internet protocols. It is a non-interactive command line tool, so it can easily be called from scripts. GNU `wget` has many features to make retrieving large files or mirroring entire web or FTP sites easy. See the documentation on the GNU `wget` site for more details.
You can download a version for MS windows from `http://xoomer.virgilio.it/hherold/`.

Starting a download of the conference XML file from SPMS using `wget` is done in the following way:

```
wget http://<your_spms_address>/spms.xml
```

Every download of the XML file »`spms.xml`« using the above notation will not overwrite an existing file but create a new one »`spms.xml.1`«, »`spms.xml.2`«, and so on. So make sure that the downloaded file is used by the config and control the download destination:

```
wget -O <file_name> http://<your_spms_address>/spms.xml
wget --output-document=<file_name> http://<your_spms_address>/spms.xml
```

You can use the command option »`-r`« (or »`--recursive`«) for recursive retrieving which will let you overwrite an existing »`spms.xml`«:

```
wget -r http://<your_spms_address>/spms.xml
wget --recursive http://<your_spms_address>/spms.xml
```

Using `wget` on a production SPMS like EPAC'06 you will get something similar to the following:

```
wget --output-document=epac06.xml http://oraweb.cern.ch/pls/epac06/spms.xml
--16:24:03--  http://oraweb.cern.ch/pls/epac06/spms.xml
           => 'epac06.xml'
Resolving oraweb.cern.ch... done.
Connecting to oraweb.cern.ch[137.138.203.145]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9,044,569 [text/html]

100%[====================================>] 9,044,569    711.27K/s    ETA 00:00

16:26:21 (711.27 KB/s) - 'epac06.xml' saved [9044569/9044569]
```

2.4 PDF utilities `pdfinfo`, `pdftotext`, `pdffonts`, ...

The following PDF utilities are needed by some of the scripts: `pdfinfo`, `pdftotext`, and `pdffonts`. These utilities are part of `xpdf` an open source viewer for PDF files and can be downloaded from the xpdf site. Look for precompiled binaries for **x86, DOS/Win32**. There should be an entry with *pdftops, pdftotext, pdfimages, pdfinfo, and pdffonts only* with a link to a file like `xpdf-<version>-win32.zip` (size is ≈1.6 MB): http://www.foolabs.com/xpdf/download.html.

Install the binaries where they could be found from a command prompt window (set `path`), a test reveals:

```
D:>pdfinfo
pdfinfo version 3.01
Copyright 1996-2005 Glyph & Cog, LLC
Usage: pdfinfo [options] <PDF-file>
  -f <int>        : first page to convert
  -l <int>        : last page to convert
  -box            : print the page bounding boxes
  -meta           : print the document metadata (XML)
  -enc <string>   : output text encoding name
  -opw <string>   : owner password (for encrypted files)
  -upw <string>   : user password (for encrypted files)
  -cfg <string>   : configuration file to use in place of .xpdfrc
  -v              : print copyright and version info
  -h              : print usage information
  -help           : print usage information
  --help          : print usage information
  -?              : print usage information
```

A real live example from a conference pdf file will show:

```
D:>pdfinfo D:\DIPAC2005\PAPERS\CTMMO2.PDF
Title:          Micro-strip Metal Foil Detectors for the Beam Profile Monitoring
Subject:        Proceedings of DIPAC 2005
Keywords:       monitoring, electron, radiation, photon, ion
Author:         V.M. Pugatch, V.E. Aushev, NASU/INR, Kiev; C. Bauer, MPI-K, Heidelberg;
                M. Braeuer, DESY, Hamburg; O.A. Fedorovitch, NASU/INR, Kiev; H. Franz,
                DESY, Hamburg; V.G. Karengin, NASU/IM, Kiev; A.V. Mikhailenko, NASU/INR,
                Kiev; V.L. Perevertailo, NASU/IM, Kiev; S.V. Prystupa, Y.V. Pylypchenko,
                NASU/INR, Kiev; M.T. Schmelling, MPI-K, Heidelberg; K. Wittenburg, DESY,
                Hamburg
Creator:        LaTeX with hyperref package
Producer:       pdfeTeX-1.21a
CreationDate:   02/24/06 15:45:01
Tagged:         no
Pages:          3
Encrypted:      no
Page size:      592.777 x 789.041 pts
File size:      233533 bytes
Optimized:      no
PDF version:    1.4
```

2.5 Autorun Inf Editor

You will need this editor if you are planning to produce a CD-ROM with »autoplay« characteristics. When a CD is put into the drive, the operating system looks for the »autorun.inf« file. If the file exists, it reads it and then knows how to »autoplay« the CD.

The Autorun.inf Editor is a free utility which makes it easy to create own »autorun.inf« files. With the editor you can create an autorun.inf file that can:

- Set a label for the CD-ROM (this label will be displayed in »My Computer« and »Explorer« on a Windows computer)

- Set an icon for the CD-ROM

- Specify a program on the CD-ROM (or user's computer) to run when the CD-ROM is inserted

- Add entries and commands to the context menu for when a user right-clicks on the CD-ROM

You can get the zip file from Moon Valley Software. After a registration with name, email address, zip code, and country information the download link will be emailed to you. For easy access I have included this zip file (AIF10.ZIPPED) in the documentation PDF.[1] This file is named with the extension ».ZIPPED« because Acrobat Reader 7 won't let you access the file if it is named ».ZIP« (see the remark in section 3.1 on page 11).

As an example for the »autorun.inf« file generated by the AIF Editor see the file below (example for the LINAC2004 conference), the user interface to AIF you can see on the right:
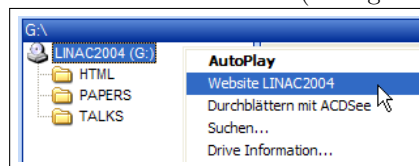
```
1 [autorun]
2 open=Opener.exe index.htm
3 Shell\Option1=Website LINAC2004
4 Shell\Option1\Command=opener.exe http://bel.gsi.de/linac2004/index.html
```

Line 2 defines that »index.htm« will be opened with the helper program »Opener.exe« which is an utility you will find in the file AIF10.ZIPPED. If you do not specify an relative path, »Opener.exe« has to reside in the root directory (SCS2006_CD in our case).
Due to problems in Windows XP where »Icon« and »Label« cannot be shown, these entries are not used.
Line 3 shows the text which will appear in the context menu when you right click on the CD-ROM and the little menu pops up. Line 4 specifies the action that will happen when the user selects it (see figure below).

Figure 2: Autorun.inf Editor

CD Context menu with additional option

Please read the step by step instructions in »AutorunInfEditor.chm« to understand the possibilities you have with AIF.
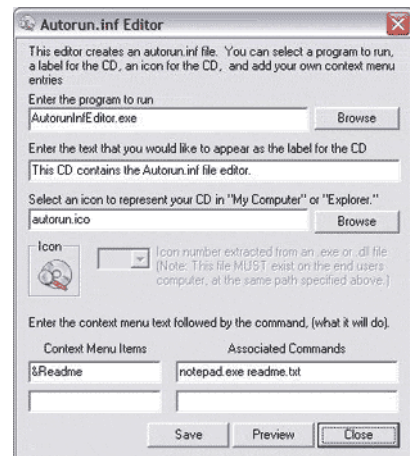
---

[1] From the Moon Valley Software web site: This program may be freely distributed as-is.

2.6 Ruby Scripting Language

The installation of Ruby is not mandatory. Ruby is only needed if you want to generate an Abstract Booklet ($\hookrightarrow$ section 4.2).

A Ruby installation for Windows can be found at the One-Click Installer – Windows on RubyForge. There you will find lots of versions and always the last stable one. The actual stable release is »ruby-1.8.2-15«. Select the
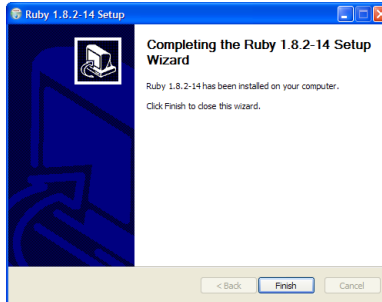

Figure 3: Ruby Setup Wizard

windows package with MSI installer for download (the size is $\approx 13\,$MB). The installation is very easy, you only have to add the directory where the executables are found (e. g. `E:\Ruby\bin`) to the actual »PATH« setting.

You should check that everything is setup correctly in a command prompt window by starting »ruby«:

```
e:epac06>ruby --version
ruby 1.8.2 (2004-12-25) [i386-mswin32]
```

This is important because the command »texexec« will just die without any output or warning if Ruby is not installed or the executable not found.

3 JACoW Scripts

The general setup and the execution of the JACoW scripts are first listed in sequence and described in detail later.

❶ Setup of local site for JACoW scripts

   ① Setup scripts and base directories with SCS2006.ZIPPED (↪ section 3.1)

   ② Adapt the configuration file »`conference.config`« to your needs (↪ section 3.2)

❷ Production run (↪ section 4.1)

   ① Download the conference XML file from SPMS (↪ section 2.3)

   If you want to generate an Abstract booklet then your sequence will be

   ⑤ Run the script »`spmsbatch.pl`« to generate the source input for the Abstract booklet (↪ section 3.3)

   ❍ Start ConTEXt (it's a variant of TEX like LATEX) to compile the generated files and produce an Abstract booklet (↪ section 4.2)

   ② Download the PDF files of the edited papers from SPMS and place them in »`./papers-final`«

   ③ Run the script »`scan-keywords.pl`« to generate the keyword list (↪ section 3.5)

   ④ After final QA make sure that you generate paper TOC values (↪ section 3.7)

   ⑤ Run the script »`spmsbatch.pl`« to generate web pages, the TEX files for the abstract booklet, the proceedings volume and to package the papers from »`./papers-final`« (↪ section 3.3)

   ⑥ Run the command file »`gen_texpdf.bat`« in »`./PAPERS`« to package the raw files (add page numbers, header and footer lines, fill hidden fields with keywords, etc.) (↪ section 3.4)

   ⑦ Copy all files from `../PAPERS/*.pdf`, `./HTML/*.*`, and `./TALKS/*.pdf` to the webserver directories

   ⑧ Copy all files from `./PAPERS/*.pdf`, `./HTML/*.*`, `./TALKS/*.pdf`, the final `INDEX.HTM`, and all other linked files to a separate directory tree to produce a CD-ROM (↪ section 4.4)

❸ Check of consistency (↪ section 5)

   ① Check the generated pdf files for completeness (↪ section 5.1)

   ② Run the script »`pagecheck.pl`« to compare the number of pages a pdf file has in the SPMS database against the real pdf file (↪ section 5.2)

   ③ Use Xenu to check the script generated web pages for broken links and orphan files (↪ section 5.3)

❹ Compare all errors that have occurred and could not be explained to the ones documented in section »`Known Bugs`« (↪ section 7).

3.1 Script Setup

The JACoW scripts and an example setup are included as a zip file in this PDF. You can extract it and unzip it in the main directory of your conference proceedings. It is called SCS2006.ZIPPED which stands for Sample Conference Setup 2006. Right click on the file name to get more infos and open or save the file to disk. This file is named with the extension ».ZIPPED« because Acrobat Reader 7 won't let you access the file if it is named ».ZIP«. There is a security note in the Acrobat Support Knowledgebase (Modify the default behavior of file attachments in Acrobat and Adobe Reader (7.x and 3D on Windows and Mac OS)) but I do not recommend the use of InstallShield Tuner 7 to solve this problem the Acrobat way. So please save the file and rename the extension to one which is accepted by your extraction software.

The Conference Sample Setup file creates a directory structure in accordance with the basic configuration file, and places the Perl scripts and TeX files in the correct directories (see contents of »conference.config«).
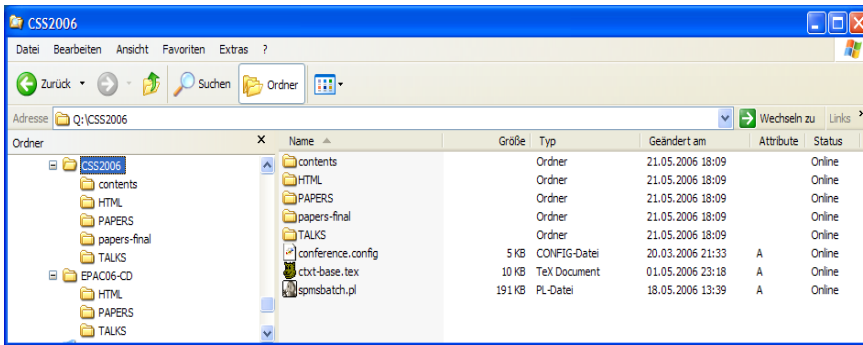

Figure 4: Standard layout for Sample Conference Working Directories
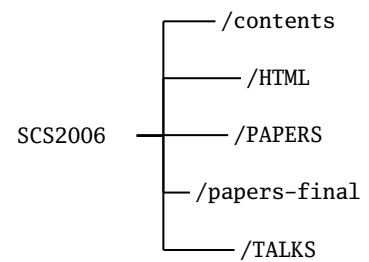
Figure 5: Basic layout

The purpose of the created subdirectories is as follows (listed in alphabetical order):

- The »contents« directory will hold photos, dividing sheets, etc. and the scripts will write intermediate .tex files to this location.
- The »HTML« directory will hold all the generated HTML files, cascading style sheet, and logos for the JACoW web site.
- The »PAPERS« directory will hold the final conference pdf files, including the proceedings volume and abstract booklet (if requested in the config file).
- The »papers_final« directory will hold the conference pdf papers in the *raw* state (without any markup of the conference, like page numbers, classification or conference name) as they come from SPMS.
- The »TALKS« directory will hold the conference slide files (pdf).

A distinction is drawn between the uppercase and lowercase directory names, the uppercase ones will later go to the JACoW web site and on the CD-ROM, the lowercase ones are just working directories which are only used during the generation of the web site and the proceedings volume, etc.

3.2 Config Setup

What you need to check is the configuration file. The file »conference.config« contains all directory settings, default tags and switches for the generation of abstract booklet and proceedings.

All configuration parameters concerning the conference setup are set in lines 4...10. The »conference_xmlfile« points to the XML file downloaded from SPMS, if no directory specification is given, it refers to the directory where »spmsbatch.pl« is run. The »conference_logo« is used on the banner page[2] as you can see on the left in figure 6. The directory where the logo is placed is defined by »img_directory« on line 31. The parameter

---

[2]and on the generated index file »indexloc.htm« which makes the web site more easily accessible before you have produced your own adapted web index page.

»conference_logo_size« is used to provide enough space for the conference name (»conference_name« and »conference_site«) which are displayed in the banner. The size is given in pixel format »Hor x Vert«. The word »Proceedings« is automatically attached to the »conference_name«. The »conference_url« is linked with the conference logo[3].
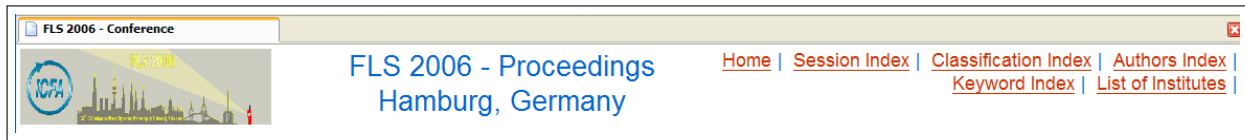

Figure 6: Banner page with conference logo and conference name and site

```
1  #*******************
2  # Conference Setup
3  #
4  conference_xmlfile   = spms.xml
5  conference_logo      = ../HTML/logo.jpg
6  conference_logo_size = 240x72
7  conference_url       = http://fls2006.desy.de/index_eng.html
8  conference_respm     = v.r.w.schaa&#064;gsi.de
9  conference_name      = FLS 2006
10 conference_site      = Hamburg, Germany
```

The »debug_file« is used for debugging output by the Perl script »spmsbatch.pl«. All print statements of the format »print DBG« go to this file.

```
11 #*******************
12 # other files
13 #
14 debug_file = protocol.txt
```

The lines 15...33 contain the parameters for several directories and the base url. The »base_url« is used on the banner page (see fig. 6) for the URL entry »Home«. The »content_directory« is a place where files for the proceedings volume are coming from or are written to, i.e. the lists for authors and institutes (authtexidx.tex, insttex.tex). The »html_directory« is used for all generated web pages for authors list, keyword index, institutes list, session list, and classification index. The style sheet for these web pages (confproc.css) has to be placed here too. In the »slides_directory« all PDF files with slides from talks are placed. From the »img_directory« the pictograms to designate links for slides (▣) or videos (🎥) are taken. Normally this directory is the same as the »html_directory«. The »conference_logo« (line 5) uses a direct link to the conference logo image.

The »paper_directory« will be the destination of all »packaged« PDF files. »packaged« means that the »raw« PDF files taken from »raw_paper_directory« have received their correct page numbers, header and footer lines. In the »raw_paper_directory« only PDF files may be placed which had been uploaded to SPMS after the JACoW editing process including final Quality Assurance (QA) etc.

```
15 #********* Proceedings of FLS 2006, Hamburg, Germany
16 # directories
17 #    base_       base for reference links from docs, papers, pages [index.html]
18 #    content_    this is where the time-schedule, dividing sheets, pictures, etc. are coming front
19 #    html_       for all html-files [.html]  (author, keyword, institute, session, ...)
20 #    slide_      for all slide- or transparency-files [.pdf]
21 #    img_        for all images [.png, .jpg, .gif, ...]
22 #    paper_      for all papers *with* embedded infos [.pdf]
23 #                     (keyword/author/..., conference tag, page numbers, etc.)  [.pdf]
24 #    raw_paper_  for all papers *without* embedded infos [.pdf]
25 #                     (just the processed files from JACoW editors)
26 #
```

---

[3]And on »indexloc.htm« too.

```
27 base_url            = http://adweb.desy.de/mpy/FLS2006/proceedings/index.htm
28 content_directory   = ./contents/
29 html_directory      = ./HTML/
30 slides_directory    = ../TALKS/
31 img_directory       = ../HTML/
32 paper_directory     = ./PAPERS/
33 raw_paper_directory = ../papers-final/
```

The »`paper_not_received_text`« parameter is used for defining the text which will be put in papers when the final text has not been sent by the author. Before the final deadline mostly the text »`PAPER NOT YET RECEIVED`« is used, after the deadline this is changed to »`PAPER NOT RECEIVED`«.

```
34 #*******************
35 # text to be put in papers when the final text has not been sent by the author
36 #   DIPAC2003 used "PAPER NOT YET RECEIVED" before finalizing
37 #             and "PAPER NOT RECEIVED" after submission stop
38 #
39 paper_not_received_text = PAPER NOT YET RECEIVED
```

The »`session_div_pages`« parameter defines the number of pages which will be put before a new session starts. These pages act as divider between sessions and intro into the next session in the proceedings volume. If »`session_div_pages`« is »`1`«, there must be a PDF file in »`content_directory`« with the name »`<session>.pdf`«. This parameter is only used if a proceedings volume (»`proceedings.pdf`«) is generated (see parameter »`proceedings_volume_switch`«).

```
40 #*******************
41 # number of pages between each session (these pages act as divider between sessions and
42 #                                  intro into the next session in the proceedings volume)
43 session_div_pages = 0                # dividing pages between sessions
```

With »`use_arbilex_sorting`« you define that a lexical sorting of authors names should be used. This is mostly important for names with accented characters (»öüéèãçïôûø« and so on). On page 4 the installation of the Perl package »**Sort-ArbBiLex**« was shown.

```
44 #*******************
45 # sort_authors_lexically "yes|no"
46 #         for lexical sorting (of authors names). Using the package "ArbiLex"
47 #         offers sort functions for arbitrary sort orders
48 #
49 use_arbilex_sorting = yes
```


The last four line in the file »`conference.config`« are switches to control the generation of notes on web pages or paper volumes for the conference. Switch »`paper_ftn_switch`« controls (»`1`« for footnotes, »`0`« for no footnotes) the appearance of footnotes on the web pages (see the figure to the left in the lower part *[1]* and *[2]*). The switch »`paper_agy_switch`« controls (»`1`« for print funding note, »`0`« for no funding notes) the output of funding notes (see text **Funding:** in upper part of figure) directly under the authors' list.

With »`proceedings_volume_switch`« you control the generation of a proceedings volume (»`1`« yes, »`0`« no), »`context_switch`« determines whether an abstract booklet is produced.

```
50 #*******************
51 paper_ftn_switch = 1              # footnotes will (not=0) be included
52 paper_agy_switch = 0              # funding notes will (not=0) be included
53 proceedings_volume_switch = 0    # Proceedings volume will (not=0) be produced
54 context_switch = 1               # (do not=0) generate a ConTeXt abstract booklet
```

3.3 Script `spmsbatch`

This script is the first one you run after downloading the SPMS' XML file ($\hookrightarrow$ section 2.3). It is started on the directory where »conference.config« resides:

```
1 E:\EPAC2006>spmsbatch.pl
```

You will get a number of messages on the screen during the script's run: about the XML file, the logo and all sessions found in the XML file.

```
1  config file 'conference.config' found!
2
3  config file points to 'spms.xml'
4  logo: 70 x 70
5
6  Conference XML 'EPAC 2006' opened
7   going to close session '0': Linear Colliders, Lepton Accelerators and New Acceleration Techniques
8   going to close session '2': Circular Colliders
9   going to close session '3': Hadron Accelerators
10  going to close session '4': Synchrotron Light Sources and FELs
11 --> Papercode: MOPCH121 ### Documentname: MOPCH119.DOC
12 --> Papercode: MOPCH144 ### Documentname: MOPCH119.DOC
13  --> Affiliation Request Pending for Godet
14  --> Affiliation Request Pending for Radovanov
15  going to close session '5': Poster Session
16 <...>
17  going to close session '16': Circular Colliders
18  --> Affiliation Request Pending for Walter
19  --> Affiliation Request Pending for Larijani
20 <...>
21 --> Papercode: TUPLS027 ### Documentname: TUPLS026.DOC
22 --> Papercode: TUPLS027 ### Documentname: TUPLS026.PDF
23  --> Affiliation Request Pending for Malinowski
24  --> Affiliation Request Pending for Massidda
25  going to close session '18': Poster Session
26  going to close session '19': Accelerator Technology
27  going to close session '20': Linear Colliders, Lepton Accelerators and New Acceleration Techniques
28  going to close session '21': Linear Colliders, Lepton Accelerators and New Acceleration Techniques
29  going to close session '22': Hadron Accelerators
30  going to close session '23': Beam Dynamics and Electromagnetic Fields
31 <...>
32 --> Papercode: THPLS028 ### Documentname: THPLS028.DOC
33  going to close session '45': Beam Instrumentation and Feedback
34
35 Conference XML 'EPAC 2006' closed
```

Line 1 signals that the »conference.config« is found and processing will continue. Line 3 tells that the XML file which will be read is »spms.xml«. Line 4 gives the dimensions of the logo to be used in the banner. Line 6 and 35 shows that reading the XML file has begun/ended. All sessions founds in the XML file are numbered and their name is given at the end of reading the appropriate session (lines 7–10, 15–17, 28–30, 33). The lines 11–12, 21–22, and 32 show papercode ids under which files have been uploaded which do not agree to the naming scheme of paper source files (line 32 shows an »o« instead of a »0«). This can most likely happen when Javascript is disabled during file upload and the necessary checks on file names can not be made.

The lines 13–14, 18–19, and 23–24 show that an author is referenced whose affiliation and/or profile entry is still pending, that means that a request has been submitted, but the profile is still not complete. So expect to see entries on the web pages and in the abstract booklet like:
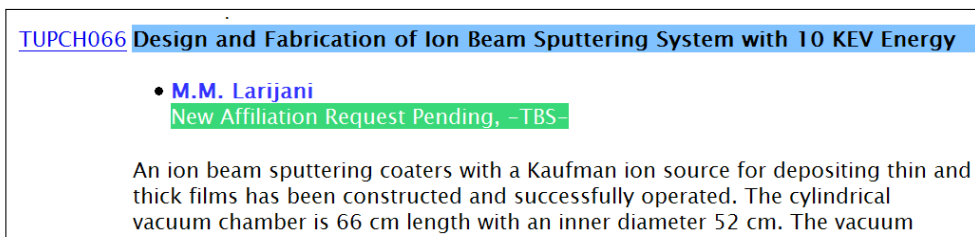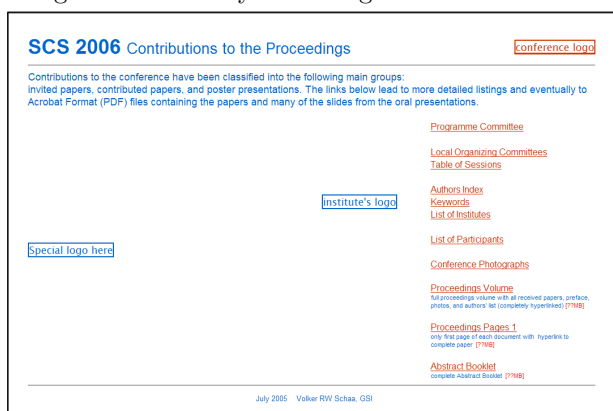

Figure 7: Affiliation Request Pending for Mr. Larijani

After the output in line 35 the script run is complete and you should have a complete web site. You can test the generated site by launching »`INDEXLOC.HTM`« in the root directory.

After each run of »`spmsbatch`« a default »`INDEXLOC.HTM`« is generated in the root directory which should ease the launch of the generated web site. This file functions as an »`INDEX.HTM`« skeleton and has all necessary links prepared. It could be easily adapted to the *real* thing. Model for this template were the EPAC web presentations in 2004 and 2002.[4] The necessary html source attributes for the conference and institute's logos are there and have to be filled (the conference logo is preset from the config file, only width and height have to be adjusted: `<img alt="logo" src="HTML/logo.jpg" width="xx" height="xx">`). The positions for logos are shown in the template to the left (see rectangles with logo designation). Links for committees (»`Programme`« and »`Local Organizing`«), »`Proceedings Volume`« (full proceedings volume with all received papers, preface, photos, and authors' list and completely hyperlinked), »`Proceedings Pages 1`« (contains only the first page of each document with hyperlinks to the complete paper) and »`Abstract Booklet`« are configured. After customizing »`INDEXLOC.HTM`« you should rename it to »`INDEX.HTM`«, otherwise the next script run will destroy your changes.

Selecting one of the links to »`Table of Sessions`«, »`Authors Index`«, »`Keywords`« or »`List of Institutes`« will bring you to the banner page (See figure 6 on page 12) which will show the web pages generated from the information in the XML file.
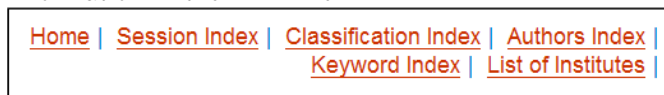
The figure to the left shows the link section of the banner page with all links to the generated pages. »`Session Index`« works only, if sessions have already been assigned in the SPMS. So be warned if you run the script and generate the web pages in an early stage of the conferences (i.e. before the uploaded abstracts are assigned to specific sessions) this link will lead you to a (mostly) empty page. The same is true for the link to »`Keyword Index`« in a stage before any paper have been run through the process of keyword generation (↪ section 3.5 on page 17). The links which surely work from the earliest stage of a conference are the »`Authors Index`« and »`List of Institutes`«.

Running the script before any `paper_code`s have been assigned, the `abstract_id` will be used instead. This `abstract_id` is named »`Contribution ID`« in the SPMS search panel.

---

[4]See http://accelconf.web.cern.ch/accelconf/e04/default.htm for the EPAC 2004 JACoW site

3.4 Command File `gen_texpdf`

The command file `gen_texpdf.bat` in the directory »`./PAPERS`« is automatically generated by a run of `spmsbatch.pl`. Apart from the introduction remarks in line 2...5, for each paper in the SPMS' XML file a sequence of commands (lines 6...15 or 16...23) is generated. The first three lines are just comments stating the paper which is going to be processed. Then a pdfLaTeX run is processing »`<paper_code>.tex`« (which has also been generated by `spmsbatch.pl`. The resulting PDF file is processed by `thumbpdf.exe` to generate thumbnails which are included into the PDF file by the next run of pdfLaTeX (line 11). If the thumbnails are already generated by a previous run, `thumbpdf` and a second pdfLaTeX run are skipped (line 20). An optimized version of the PDF is created using `pdfopt.exe` (lines 12 and 20), deleting and renaming of not needed files follows.

Please be aware that the thumbnail produced by a run of `thumbpdf.exe` is used as long as it is found on the directory. It will not be reproduced if the underlying PDF file changes due to the automatic skipping of this step in the command precedure. If you want to reprocess the thumbnail you have to delete the file »`<paper_code>.tpt`« before running `spmsbatch.pl`.

```
1  @echo off
2  REM Command file to generate final pdfs from raw pdfs
3  REM (raw i.e. without Title, Author, Keyword fields)
4  REM --------------------------- 20 Mar 2003 vrwSchaa
5  REM                            mailto:v.r.w.schaa@gsi.de
6  echo.
7  echo generating "MOXPA01.pdf"
8  echo.
9  pdflatex MOXPA01.tex
10 thumbpdf MOXPA01.pdf
11 pdflatex MOXPA01.tex
12 pdfopt MOXPA01.pdf MOXPA01-opt.pdf
13 del MOXPA01.pdf
14 rename MOXPA01-opt.pdf MOXPA01.pdf
15 del MOXPA01.aux
16 echo.
17 echo generating "MOXPA02.pdf"
18 echo.
19 pdflatex MOXPA02.tex
20 pdfopt MOXPA02.pdf MOXPA02-opt.pdf
21 del MOXPA02.pdf
22 rename MOXPA02-opt.pdf MOXPA02.pdf
23 del MOXPA02.aux
24 ...
```

Purpose	Packaging the papers and providing thumb-nails for PDF backwards compatibility
Directory	`$paper_directory` ↪ `./PAPERS`

needs	`pdflatex.exe`
	`thumbpdf.exe`
	`pdfopt.exe`

reads	`<paper_code>.tpt`

writes	`<paper_code>.pdf`
	`<paper_code>.tpt`
	PROCEEDINGS.PDF
	PROCEEDINGS-PAGES1.PDF

3.5 Script scan–keywords

The Perl script »`scan-keywords.pl`« is used to generate a list of the five topmost used keywords of each paper. The papers are checked for JACoW specific keywords, this list is available on the JACoW web site. You may want to add additional keywords to the list (e. g. PAC05 added »`SNS`« and »`Spallation Neutron Source`« to the master list).

The way »`scan-keywords.pl`« operates:

❶ it reads the keyword file »`keywords.list`« and stores the keywords and their regular expressions in an internal structure,

❷ it searches the current directory for pdf files (`<paper_code>.pdf`),

❸ it checks whether there is an already converted text file for the found pdf file (`<paper_code>.txt`),

❹ if **no**, it converts the pdf file into a text file (`<paper_code>.txt`) using »`pdftotext`«,

❺ it scans the text files and compares the keyword list to the file's contents,

❻ it writes the file (`keyword-count.txt`) with entries for the five topmost used keywords, like

```
<paper_code>=keyword1;keyword2;keyword3;keyword4;keyword5;
```

which should read

```
POM031=single-bunch;acceleration;beam-losses;linear-collider;simulation;
```

After running »`scan-keywords.pl`« you should check the new generated »`keyword-count.txt`« file for lines like »`<paper_code>=`«. If you encounter such an entry you should check »`<paper_code>.txt`« (in the same directory) to see if it contains readable ASCII text.

Each »`<paper code>.pdf`« is translated to ASCII which is written to a file named »`<paper_code>.txt`«. Sometimes (mostly with Macintosh papers and papers with special character set encodings) you will only see garbage there. Try to produce a »`<paper_code>.txt`« file manually with the ASCII contents of the paper, and rerun »`scan-keywords.pl`«. It then should fill the above mentioned line with the correct keywords. Sometimes you may have to cut and paste from the Word file into »`<paper code>.txt`« to get real ASCII as even cut and paste from AcrobatReader will not work.

Purpose	This script identifies the five topmost used keywords of a given list in each paper.
Directory	`$raw_paper_directory` ↪ `./papers-final`
needs	`pdftotext.exe`

reads	`keywords.list`
	`<paper_code>.pdf`
	`<paper_code>.txt`

writes	`keyword-count.txt`
output is used by	`spmsbatch.pl`

3.6 Script `keyword-sql`

The Perl script »`keyword-sql.pl`« is used to generate a procedure file (»`keywords_plsql.txt`«) for the SPMS system to load the keywords via `<paper_id>` into the appropriate table for repatriation of keyword data. For this the script reads the generated file »`keyword-count.txt`« ($\hookrightarrow$ section 3.5) and writes exec records for SPMS. Matt[5] has written the corresponding procedure »`spms.abstract_keywords`« which is used to input the data into the keyword data tables.

Each generated line in »`keywords_plsql.txt`« looks like

```
exec spms.abstract_keywords ('<paper_code>', '<keyword>');
```

Purpose	Loading of keywords into appropriate table for repatriation
Directory	`$raw_paper_directory` $\hookrightarrow$ `./papers-final`
reads	`keyword-count.txt`
writes	`keywords_plsql.txt`
output is used by	SPMS (»`spms.abstract_keywords`«)

3.7 SPMS `Generate TOC Values`

The Administrator Pages of the SPMS web interface (you have to log in with administrator privileges) provide an option to generate table of contents (TOC) entries for all papers. This is necessary to generate correct page numbers for the PDF files, web pages, and proceedings.

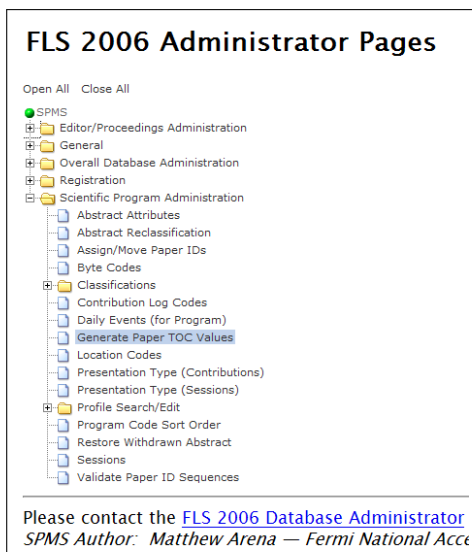You can find this option in the »`Scientific Program Administration`«:

Figure 8: Generate TOC Values

Figure 9: TOC Numbers Generated

---

[5]Matt Arena mailto:arenam@fnal.gov

18

4 Production of Web Site, Abstract Booklet, Proceedings Volume, CD-ROM

4.1 Production Run for the Web Site

The whole process of creating a web site update is as follows:

❶ Download a new XML file which reflects the actual SPMS status like:
   »`wget http://<your spms server>/spms.xml`«
   Make sure that the downloaded file is used by the config!
   A new file download will not overwrite »`spms.xml`« but create a
   new file named »`spms.xml.1`«, »`spms.xml.2`«, etc. (↪ section 2.3)

❷ Copy all (new) papers from SPMS to `./papers-final`

❸ Run `scan_keywords.pl` to update the keywords (↪ section 3.5)

❹ Copy all talks (transparencies) from SPMS to `./TALKS` (make sure that the implied rules have to be followed—SPMS gives a link to the file only if it has been uploaded via SPMS, if you later update this directory with new files, make sure that the name is »`<paper_code>_TALK.PDF`«).

❺ Run `spmsbatch.pl` (↪ section 3.3)

❻ Run the command file `gen_texpdf.bat` in »`./PAPERS`« to package the raw files (↪ section 3.4)

❼ Copy all files from `./PAPERS/*.pdf`, `./HTML/*.*`, and `./TALKS/*.pdf` to the webserver directories

❽ Run »`Xenu`« (↪ section 5.3) to check the directory structure for broken links and orphan files.

4.2 Production of an Abstract Booklet

A run of »`spmsbatch.pl`« will produce the content for an abstract booklet if this option is selected in the config file[6]. The resultant file is placed in the »`paper_directory`« with the name »`ctxt-info.tex`« and included by »`abstract-base.tex`«. This file could not be compiled with the standard »pdfTEX« because the generated source uses the ConTEXt TEX-engine to provide automatically generated bleeding boxes around fields for authors, title and papercode (see an example page from the abstract booklet for the EPAC'06 conference to the right).

If you have installed Ruby (↪ section 2.6) and the executables are reached from the command line—you should try it first in a command prompt window:

```
D:\SCS2006\PAPERS>ruby --version
ruby 1.8.2 (2004-12-25) [i386-mswin32]
```

you can start with the standard command sequence as follows after having set up the environment for ConTEXt:

```
D:\SCS2006\PAPERS>texexec --pdf --autopdf abstract-base.tex
D:\SCS2006\PAPERS>pdfopt abstract-base.pdf abstract.pdf
```

»`texexec`« is the base command for ConTEXt; see »`texexec --help`« what it can do for you. »`--pdf`« is used to produce PDF output, and »`--autopdf`« will tell Acrobat Reader to close the old PDF file and reopen the new one after compiling.

---

[6] only if the ConTEXt generating switch (`context_switch`) in the configuration file is set to »`1`« (↪ section 3.2)

You can download a minimal setup for ConTEXt from the PRAGMA Advanced Document Engineering web site. There are two packages to download: minimal ConTEXt distribution (no binaries, platform independent) with 62 MB, minimal ConTEXt binaries (windows) with 5 MB. Unzip both ZIP files in a new directory:

```
C:\Context>unzip justtex.zip
Archive:  justtex.zip
   creating: tex/texmf/
   creating: tex/texmf/fonts/
   creating: tex/texmf/fonts/afm/
   creating: tex/texmf/fonts/afm/adobe/
   creating: tex/texmf/fonts/afm/adobe/bookman/
  inflating: tex/texmf/fonts/afm/adobe/bookman/pbkd8a.afm
<...>
C:\Context>unzip mswintex.zip
Archive:  mswintex.zip
   creating: tex/texmf-mswin/
   creating: tex/texmf-mswin/bin/
  inflating: tex/texmf-mswin/bin/afm2pl.exe
  inflating: tex/texmf-mswin/bin/afm2tfm.exe
  inflating: tex/texmf-mswin/bin/aleph.exe
  inflating: tex/texmf-mswin/bin/bibtex.exe
  inflating: tex/texmf-mswin/bin/bzip2.exe
<...>
```

ConTEXt comes with a setup command procedure (`setuptex.bat`) which you have to call first before you can execute any other ConTEXt command from a command prompt window (DOS box).

```
E:\EPAC06\PAPERS>c:\Context\tex\setuptex
```

After installation (unpacking of the ZIP files) you first have to generate TEX formats which is done with the command »`texexec --make`«[7]:

```
C:\Context\tex>texexec --make
eXExec | using search method 'kpsefast'
eXExec | updating file database
ktexlsr: Updating C:/Context2006/tex/texmf-mswin/ls-R...
ktexlsr: c:/Context2006/tex/texmf-project is not a directory. Skipping...
ktexlsr: Updating C:/Context2006/tex/texmf-fonts/ls-R...
ktexlsr: Updating C:/Context2006/tex/texmf-local/ls-R...
ktexlsr: Updating C:/Context2006/tex/texmf-extra/ls-R...
ktexlsr: Updating C:/Context2006/tex/texmf/ls-R...
ktexlsr: Done.
<...~1600 lines...>
7688 memory locations dumped; current usage is 12466&35035
723 symbolic tokens
ranscript written on metafun.log.
eXExec | tex engine path: C:/Context2006/tex/texmf-mswin/web2c/pdfetex
eXExec | mps engine path: C:/Context2006/tex/texmf-mswin/web2c
eXExec | texformat: 14/06/2006 23:32:00 > C:/Context2006/tex/texmf-mswin/web2c/pdfetex/cont-en.fmt
eXExec | texformat: 14/06/2006 23:32:08 > C:/Context2006/tex/texmf-mswin/web2c/pdfetex/cont-nl.fmt
eXExec | texformat: 14/06/2006 23:32:08 > C:/Context2006/tex/texmf-mswin/web2c/pdfetex/mptopdf.fmt
eXExec | mpsformat: 14/06/2006 23:32:12 > C:/Context2006/tex/texmf-mswin/web2c/metafun.mem
eXExec | runtime: 22.312
```

---

[7]If this command sequence just dies showing no output whatsoever, refer to section 2.6 on page 9.

Now you have a working ConTeXt setup which has to be initialized with »setuptex« each time you open a new command prompt window. After opening a new command prompt window you can now use the command sequence to compile the abstract booklet:

```
E:\EPAC2006\PAPERS>c:\Context\tex\setuptex
E:\EPAC2006\PAPERS>texexec --pdf --autopdf abstract-base.tex
```

## 4.3 Production of a Proceedings Volume

To Be Extended

## 4.4 Production of a CD-ROM Image

The process of producing a CD-ROM image is as follows:

❶ Create a basic directory structure like the layout in the figure following this topic list

❷ Copy the final »INDEX.HTM« to the »SCS2006_CD/« root directory

❸ Copy all packaged PDF files from the production directory to the corresponding directory of the CD-ROM (SCS2006/PAPERS/*.pdf ↝ SCS2006_CD/PAPERS/)

❹ Copy all talk PDFs to the corresponding directory (SCS2006/TALKS/*.pdf ↝ SCS2006_CD/TALKS/)

❺ Copy all web files (»htm«, »css«) and logos (»gif/png/jpg«) to the corresponding directory of the CD-ROM (SCS2006/HTML/[*.htm|*.css|*.gif|*.png|*.jpg] ↝ SCS2006_CD/HTML/)

❻ You might want to add special directories later for placing conference pictures and additional material which will be linked through the final »INDEX.HTM«.

❼ Run »Xenu« (↪ section 5.3) to check the directory structure for broken links and orphan files.

The basic directory structure for a CD-ROM image:

```
                      ┌── /HTML
                      │
    SCS2006_CD ───────┼── /PAPERS
                      │
                      └── /TALKS
```
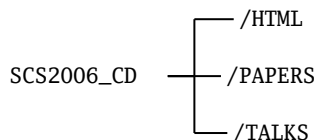
4.5 Special Characters in Abstracts

The script »spmsbatch.pl« is equipped to recognized an expanding set of characters and notations. It is able to transport these correctly into the abstract booklet, the proceedings volume, and the web site. Abstract authors can use these notations and special characters to structure the text, write small math formulas and use the correct spelling for accented words.

4.5.1 Set of accented characters

The characters in this table can be entered as codes in ISO-8859-1 (without Euro currency character), ISO-8859-15 (with Euro), UTF-8 (these are the characters in column »In«) or as HTML entities (column »HTML«). The character shown in the browser window will look like the character in column »Out« (depending on chosen font).

In	HTML	Out	In	HTML	Out	In	HTML	Out	In	HTML	Out	In	HTML	Out
Â	Â	Â	Ô	Ô	Ô	Û	Û	Û	Ê	Ê	Ê	Î	Î	Î
â	â	â	ô	ô	ô	û	û	û	ê	ê	ê	î	î	î
Á	Á	Á	Ó	Ó	Ó	Ú	Ú	Ú	É	É	É	Í	Í	Í
á	á	á	ó	ó	ó	ú	ú	ú	é	é	é	í	í	í
À	À	À	Ò	Ò	Ò	Ù	Ù	Ù	È	È	È	Ì	Ì	Ì
à	à	à	ò	ò	ò	ù	ù	ù	è	è	è	ì	ì	ì
Ä	Ä	Ä	Ö	Ö	Ö	Ü	Ü	Ü	Ë	Ë	Ë	Ï	Ï	Ï
ä	ä	ä	ö	ö	ö	ü	ü	ü	ë	ë	ë	ï	ï	ï
Ã	Ã	Ã	Õ	Õ	Õ	Ñ	Ñ	Ñ	Ð	Ð	Ð			
ã	ã	ã	õ	õ	õ	ñ	ñ	ñ	ð	ð	ð			
Å	Å	Å	Ø	Ø	Ø	ß	ß	ß	Þ	Þ	Þ			
å	å	å	ø	ø	ø	Ý	Ý	Ý	þ	þ	þ			
Æ	Æ	Æ				ý	ý	ý	Ç	Ç	Ç			
æ	æ	æ				ÿ	ÿ	ÿ	ç	ç	ç			

4.5.2 Set of recognized math characters (LaTeX mode)

The characters in this table can be entered as TeX/LaTeX codes (these are the characters in column »TeX«) or as HTML entities (column »HTML«). The character shown in the browser window will look like the character in column »Out« (depending on chosen font).

TeX	HTML	Out	TeX	HTML	Out	TeX	HTML	Out	TeX	HTML	Out
\alpha	α	α	\beta	β	β	\gamma	γ	γ	\Gamma	Γ	Γ
\delta	δ	δ	\Delta	Δ	Δ	\epsilon	ε	ϵ	\zeta	ζ	ζ
\eta	η	η	\theta	θ	θ	\Theta	Θ	Θ	\iota	ι	ι
\kappa	κ	κ	\lambda	λ	λ	\Lambda	Λ	Λ	\mu	μ	μ
\micro	--	μ	\nu	ν	ν	\xi	ξ	ξ	\Xi	Ξ	Ξ
\pi	π	π	\Pi	Π	Π	<pi>	--	π	\rho	ρ	ρ
\sigma	σ	σ	\Sigma	Σ	Σ	\varsigma	ς	ς	\tau	τ	τ
\upsilon	υ	υ	\phi	φ	ϕ	\Phi	Φ	Φ	\chi	χ	χ
\psi	ψ	ψ	\Psi	Ψ	Ψ	\omega	ω	ω	\Omega	Ω	Ω

4.5.3 Math characters available in LaTeX notation

In	Out	In	Out	In	Out	In	Out	In	Out
\deg	°	\sum	Σ	\prod	∏	\int	∫	\oint	∮
\bigcap	∩	\bigcup	∪	\bigvee	⋁	\bigwedge	⋀	\bigotimes	⊗
\bigoplus	⊕	\hbar	ℏ	\imath	ı	\jmath	ȷ	\ell	ℓ
\wp	℘	\Re	ℜ	\Im	ℑ	\prime	′	\emptyset	∅
\angle	∠	\infty	∞	\partial	∂	\nabla	∇	\forall	∀
\exists	∃	\neg	¬	\surd	√	\top	⊤	\bot	⊥
\backslash	\	\clubsuit	♣	\diamondsuit	◇	\heartsuit	♡	\spadesuit	♠
\dag	†	\ddag	‡	\S	§	\P	¶	\copyright	©
\pounds	£	\diamond	◇	\Box	□	\cdot	·	\ldots	...
\cdots	⋯	\vdots	⋮	\ddots	⋱	\lfloor	⌊	\lceil	⌈
\langle	⟨	\rfloor	⌋	\rceil	⌉	\rangle	⟩	\uparrow	↑
\downarrow	↓	\Uparrow	⇑	\Downarrow	⇓	\leq	≤	\geq	≥
\ll	≪	\gg	≫	\subset	⊂	\supset	⊃	\subseteq	⊆
\supseteq	⊇	\in	∈	\ni	∋	\equiv	≡	\sim	~
\perp	⊥	\simeq	≃	\parallel	∥	\approx	≈	\cong	≅
\neq	≠	\propto	∝	\times	×	\times	×	\div	÷
\ast	∗	\star	★	\circ	∘	\bullet	●	\cap	∩
\cup	∪	\vee	∨	\wedge	∧	\diamond	◇	\oplus	⊕
\otimes	⊗	\oslash	⊘	\dagger	†	\ddagger	‡	\longmapsto	⟼
\Longrightarrow	⟹	\Longleftarrow	⟸	\leftarrow	←	\Leftarrow	⇐	\rightarrow	→
\Rightarrow	⇒	\leftrightarrow	↔	\Leftrightarrow	⇔	\mapsto	↦	\lesssim	≲
\hbar	ℏ								

A web page showing Greek and Math characters:

4.5.4 Set of recognized "writings"

The script will interpret the following character sequences as shorthand for superscripts, subscripts or highlighting. The character sequence in this table can be entered as written, some characters are necessary to recognize the shorthand: the visible space »␣« stands for a required space, the red question mark »?« stands for any character except numeric.

Characters entered as in column »In« will be formatted like what you find in column »Out« (depending on chosen font).

In	Out	In	Out	In	Out
e+e-	e^+e^-	e-␣	e^-␣	e+␣	e^+␣
H+	H^+	H2+	H^{2+}	H-␣	H^-␣
D+	D^+			microsec	μs
+-	$\pm$	+/-	$\pm$	\sqrt{123}	$\sqrt{123}$
ab^{abc}	ab^{abc}	ab\$^{abc}\$	ab^{abc}	ab^123?	ab^{123}?
ab_{abc}	ab_{abc}	ab\$_{abc}\$	ab_{abc}		
10e123	10^{123}	10^123	10^{123}	10**123␣	10^{123}␣
10e-123	10^{-123}	10^-123	10^{-123}	_abcdef_	**abcdef**
10-123␣	10^{-123}␣	12.3e12␣		12.3×10^{12}␣	

4.5.5 List Formatting

A different case of recognized "writing" is shown next. Formatting a list in an abstract is not possible, because hard line breaks and other formatting rules are not allowed. With a list in a grouping character sequence (start list with »[+« and end list »+]«) it is possible to use a low level formatting. Each item of the list has to be preceded by either »n)« (for numeric itemize lists: 1), 2), 3), . . .) or »a)« (for alpha itemize lists: a), b), . . .).

A web page showing an itemized list:

The ROXIE program has been devised for the design and optimization of superconducting magnets. The 2–D electromagnetic model of a fast–ramping magnet in ROXIE consists of

1. a representation of strands by line currents,
2. a coupling of the finite element method and the boundary element method to take into account the field contribution of the magnet yoke, as well as eddy–current effects in conductive bulk material,
3. a model for persistent currents,
4. a model for inter–filament coupling currents, and
5. a model for inter–strand coupling currents in Rutherford–type cables.

We will present the coupling of all these effects in the mathematical framework of the theory of discrete electromagnetism.

The 2–D electromagnetic model of a fast–ramping magnet in ROXIE consists of

a. a representation of strands by line currents,
b. a coupling of the finite element method and the boundary element method to take into account the field contribution of the magnet yoke, as well as eddy–current effects in conductive bulk material,
c. a model for persistent currents,
d. a model for inter–filament coupling currents, and
e. a model for inter–strand coupling currents in Rutherford–type cables.

We will present the coupling of all these effects in the mathematical framework of the theory of discrete electromagnetism.

The above formatting was achieved by introducing the start/stop codes ([+, +]) into the text as could be seen in the following listing:

```
The 2-D electromagnetic model of a fast-ramping magnet in ROXIE consists of [+ 1) a representation of strands
by line currents, 2) a coupling of the finite element method and the boundary element method to take into
account the field contribution of the magnet yoke, as well as eddy-current effects in conductive bulk
material, 3) a model for persistent currents, 4) a model for inter-filament coupling currents, and 5) a model
for inter-strand coupling currents in Rutherford-type cables. +] We will present the coupling of all these
effects in the mathematical framework of the theory of discrete electromagnetism.
```

5 Check Utilities and Procedures

5.1 Check for Completeness

Check the generated PDF files for completeness:

Page numbers

Check the generated »`<paper_code>.pdf`« files in »`./PAPERS`« for their page number(s). These should be different from »`1`« (ok—the first paper will have a starting page number of »`1`«, but that's the only one!). If you find more, check the generated paper TOC values ($\hookrightarrow$ section 3.7).

Keywords

Check some of the generated »`<paper_code>.tex`« files in »`./PAPERS`« to verify that the »`pdfkeywords`« field in the LaTeX files is populated correctly. You should search in the setup command »`\hypersetup`« for the »`pdfkeywords`« entry. In line 6 you see a correctly filled entry.

```
1  \hypersetup{pdfpagemode=UseThumbs,%
2             pdfstartview=FitBH,
3             pdftitle={New Technology in Hydrogen Absorbers for Muon Cooling Channels},
4             pdfauthor={M.A.C. Cummings, Northern Illinois University, DeKalb, Illinois},
5             pdfsubject={Proceedings of PAC 2005},
6             pdfkeywords={emittance, simulation, linac, proton, scattering}%
7             }
8  \hypersetup{:
9             :
10            pdfkeywords={}%
11            }
```

If you find entries like line 11, you should check the output of the script »`scan-keywords.pl`« ($\hookrightarrow$ section 3.5).

Contents

Check the generated »`<paper_code>.pdf`« files in »`./PAPERS`« to verify that the contents of the *raw* PDF file has been placed correctly in the resulting PDF.

If there was a problem you will most likely see a PDF that looks like the one to the right (unless it was a session that did not submit a paper).

Another likely cause of this is the directories were not defined correctly in the »`conference.config`« file (look for »`raw_paper_directory`« in the description in section 3.2 on page 12).
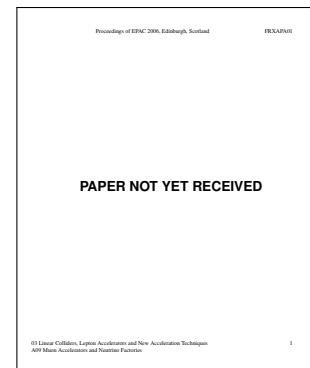
5.2 Script pagecheck

The Perl script »`pagecheck.pl`« is used to check the number of pages a pdf file has in the SPMS database against the real pdf file in the »`$raw_paper_directory`«. During a run of script »`spmsbatch.pl`« the file »`page_per_paper.txt`« is written containing the number of pages SPMS reported in the last version of »`spms.xml`«.

The script »`pagecheck.pl`« reads the file »`$raw_paper_directory/page_per_paper.txt`«, stores the reported pdf file names and number of pages in an internal data structure. Only these pdf files are checked on the local directory by extracting the number of pages (and the page sizes) using »`pdfinfo`«.

```
E:\DIPAC2005\papers-final>pagecheck.pl
reading pdfs from directory: ./
reading #pages from: './pages_per_paper.txt'
 [    1]   ITMM01 ==>  5
 [    2]   ITMM02 ==>  1
 [    3]   ITMM03 ==>  5
```

```
[   4]    ITMM04 ==>  5
[   5]    CTMM01 ==>  1
[   6]    CTMM02 ==>  3
[   7]    POM001 ==>  3
[   8]    POM002 ==>  3
<...>
[  99]    POW027 ==>  3
[ 100]    ITWA01 ==>  5
[ 101]    CTWA01 ==>  3
[ 102]    CTWA02 ==>  3
checking ITMM01.pdf
checking ITMM02.pdf
checking ITMM03.pdf
checking ITMM04.pdf
<...>
checking CTWA01.pdf
checking CTWA02.pdf
```

Running `pagecheck.pl` will show you first the number of reported pdf files (`<paper_code>`) and number of pages, then it starts to check each pdf file. The error message »`Error:  PDF version 1.6 - xpdf supports version 1.5 (continuing anyway)`« will result in a message about the used PDF version (JACoW now accepts PDFs with version 1.5). The results of the checks are written to »`pagecheck-result.txt`«, a sample output is shown next:

```
ITMM01 page count ok
CTWW99 page count ok; possible problems with non JACoW page size: '1258 x 836'
ITMM02 page count ok; possible problems with non JACoW page size: '612 x 792'
ITMM03 page count ok ==> conflict with JACoW pdf version (v 1.6)
ITMM04 spms: 4, file: 5 pages
CTMM01  ======> empty pdf file
CTMM02 page count ok
POM001 page count ok
```

Purpose	Script checks for inconsistencies in number of pages between database and PDF file.
Directory to place	$raw_paper_directory ↪ ./papers-final
needs	pdfinfo.exe
reads	page_per_paper.txt <paper_code>.pdf
writes	pagecheck-result.txt
output is used by	*interactive*

5.3 Xenu Link Sleuth

Xenu's Link Sleuth checks web sites for broken links. Link verification is done on normal links, images, frames, plug-ins, backgrounds, local image maps, style sheets, scripts and java applets. It displays a continously updated list of URLs which you can sort by different criteria. A report can be produced at any time. You can find Xenu at http://home.snafu.de/tilman/xenulink.html, the downloadable file is ≈550 kB, the zip file includes a setup executable. You should follow the advice from the web site and check the software for viruses before starting it. The web site has several links for online documentation. Xenu is used to check the script generated web pages and the directory structure for broken links and orphan files **before** uploading this to the JACoW site or making a CD-ROM.

Figure 10: Xenu's report for all addresses found

Xenu can give you a good estimate about the total size you need on a CD-ROM, and can also be used to find files or images on your site which are no longer linked to at all—so called orphaned files.

Figure 11: Xenu's statistics for total and mime type dependent space needed
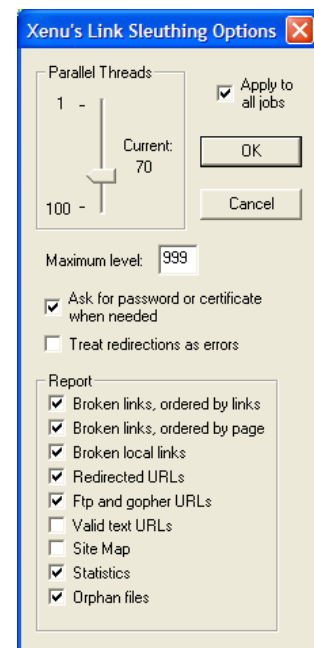

Figure 12: Xenu's options menu

5.4 Script `pdf-showfont`

The Perl script »`pdf-showfont.pl`« can help you to locate »`Type 3`« fonts and fonts which are not embedded (»`emb = no`«). The font usage is reported for every page, so it is easy to spot problematic pages.

```
>pdf-showfonts.pl Scripts-HowTo.pdf
        Title : untitled
     Producer : Acrobat Distiller 7.0.5 (Windows)
  CreationDate : 04/26/06 14
      ModDate : 04/26/06 14
        Pages : 7
    Page size : 595 x 792 pts
    File size : 408445 bytes
    Optimized : yes
   PDF version : 1.4
Seaching page:    1
<...>
Seaching page:    7
adding   page:    1 [  1   2   3   4 ]
adding   page:    2 [  1   2   3   4   5 ]
<...>
adding   page:    7 [  1   2   3   4 ]
output written on Scripts-HowTo.fonts
```

In the resulting »`<pdf-file>.fonts`« you will find all fonts used on each page (»`P00001...P00007`«), the type (check the listing for »`Type 3`« fonts), embedded fonts (»`emb = yes`«), subsetted fonts (»`sub = yes`«), and Unicode fonts (»`uni = yes`«). The object id is the same when a font is used more than once in the file.

```
H        :  name                                 type          emb sub uni object ID
P00001 :  --------------------------------- ------------ --- --- --- ---------
P00001 :  [none]                               Type 3        yes no  yes    56 0
P00001 :  Times-Roman                          Type 1        no  no  no     55 0
P00001 :  Times-Bold                           Type 1        no  no  no     60 0
P00002 :  --------------------------------- ------------ --- --- --- ---------
P00002 :  Times-Bold                           Type 1        no  no  no     60 0
P00002 :  Times-Roman                          Type 1        no  no  no     55 0
P00002 :  Symbol                               Type 1C       yes no  yes    40 0
P00002 :  Helvetica                            Type 1        no  no  no     41 0
P00003 :  --------------------------------- ------------ --- --- --- ---------
P00003 :  Times-Roman                          Type 1        no  no  no     55 0
P00003 :  Times-Bold                           Type 1        no  no  no     60 0
P00003 :  Courier                              Type 1C       yes no  no     44 0
P00004 :  --------------------------------- ------------ --- --- --- ---------
<...>
P00007 :  --------------------------------- ------------ --- --- --- ---------
P00007 :  [none]                               Type 3        yes no  yes    45 0
P00007 :  Times-Roman                          Type 1        no  no  no     55 0
P00007 :  Times-Bold                           Type 1        no  no  no     60 0
```

Purpose	Script to locate »Type 3« or non-embedded fonts in PDF files
Directory to place	*in pdf directories*
needs	pdfinfo.exe
reads	<pdf-file>.pdf
writes	<pdf-file>.fonts
output is used by	*interactive*

5.5 Script `pdfopt`

»`pdfopt`« uses Ghostscript to convert the Adobe Portable Document Format (PDF) file »`input.pdf`« to a so-called optimized form in »`output.pdf`«. Optimization puts the elements of the file into a more linear order and adds »hint« pointers, allowing Adobe's Acrobat(TM) products to display individual pages of the file more quickly when accessing the file through a network.

Purpose	Optimization of PDF files for fast web view
Directory to place	*tools directory in PATH*
needs	Ghostscript
reads	<pdf-file>.pdf
writes	<pdf-file>-opt.pdf
is used by	gen_texpdf.bat

6 Files Delivered with the Scripts

> To Be Extended

6.1 File »papers/abstract-base.tex«

Basic file for the generation of an abstract booklet using the ConTEXt TEX-engine ($\hookleftarrow$ section 4.2).

6.2 File »html/confproc.css«

The file »confproc.css« is a Cascading Style Sheet. CSS is a stylesheet language used to describe the presentation of a document. Its most common application is to style web pages in HTML and XHTML. The CSS specifications are maintained by the World Wide Web Consortium (W3C).
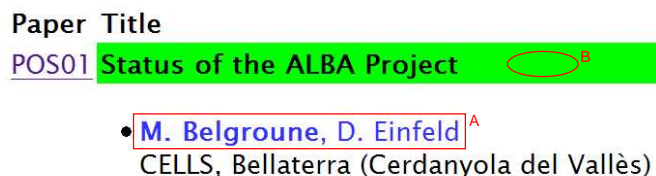
At the moment only very basic tags are provided to influence the presentation.

```
1 body { background-color: #FFFFFF; }
2 .author_cl { color: rgb(51, 51, 255); }
3 .author_se { color: rgb(255, 0, 0); }
4 .b { color:#000000; }
5 .w { color:#FFFFFF;}
6 sub, sup { font-size: 80%; }
7 td.paptitle {background-color: lime;  font-weight: bold; }
8 td.paptitle_key {background-color: lime;  font-weight: bold; text-align: left;  vertical-align: middle; }
```

Line 1 of the cascading style sheet defines the background of all pages to white.

The setting for ».author_cl« in line 2 determines all author names in »Session«, »Classification«, »Authors«, and »Keyword« pages (see A). Currently there is just the text color defined.

The background color and font weight for the paper title is set by »td.paptitle« in line 7. This setting is used for »Session«, »Classification«, and »Authors« pages (see B).

> **Paper Title**
> POS01 **Status of the ALBA Project**   B
>   • **M. Belgroune**, D. Einfeld A
>   CELLS, Bellaterra (Cerdanyola del Vallès)

The entry for »td.paptitle_key« in line 8 differs a bit from the entry for »td.paptitle« above. This entry is used for the background color of the title line in »Keyword« pages. On these pages the other keywords of a paper are listed to the right of the title bar in one to three lines. To keep a vertically centered text in the title bar, additional parameters are specified (vertical-align: middle).

The settings in lines 4 and 5 are solely for the appearance of classification and sub-classification names in the selector page for »`List of Classifications`«.

On the right you see three different ways of using classification and subclassification. The topmost figure left shows the standard EPAC style: Main classification without link and subclassifications with the ▶ symbol and link on white background.

The lower left figure shows only main classifications with links, no subclassifications present. On the right you see the PAC style (here PAC'05) with main classifications with or without subclassifications, all with links.

The setting for ».`author_se`« in line 3 effects the author name in »`Authors Index`« pages. Currently there is just the text color defined, highlighting the selected author.

Line 6 with »`sup`« and »`sub`« parameter settings will effect the typesetting of `superscript`s and `subscript`s. In the shown CSS file only the size is defined, but this parameter can easily be used to control the generated web pages for wrongly coded superscripts/subscripts by the script like the one on the right (»`10-50`« is correct).

In this case the entry was changed to:

```
sub, sup { font-size: 150%; color: red;}
```

6.3 File »`paper-finals/keywords.list`«

This file contains the keyword list which is used by the script »`scan-keywords`« (↪ section 3.5) to determine the keywords in conference papers.

The file is derived from the JACoW keyword list; it is available on the JACoW web site. You may want to add additional keywords to the list (e. g. PAC05 added »`SNS`« and »`Spallation Neutron Source`« to the master list).

If you add keywords to the (your) list, you have to obey to the list's syntax. Each keyword is given twice, once in its reference form, and then by a representation with regular expressions (Perl regex) in the following way:

```
electromagnetic-fields=electromagnetic.{0,1}field.{0,1}
```

so now `electromagnetic-fields` is found even when written as

```
electromagnetic_fields
electromagnetic_field_
electromagnetic-field_
electromagnetic-↵fields
electromagnetic-↵field_
electromagnetic-↵fields
electromagnetic-↵field_
```

This way the keyword hit rate on DIPAC2003 got more than 250% better.

The regular expression syntax of Perl is very extensive, for the keywords only two are actually used, the third one is an example to demonstrate the possible use:

.{0,1}   an arbitrarily character sequence of minimum 0 and maximum 1 character may appear here; an example is given above with »electromagnetic-fields«

(y|ies)   the character sequence »y« or »ies« must appear (one of the sequences may be empty). With this regular expression the keyword »factory« gets a hit when »factory« or »factories« is found.

(y|ies|) this will change the regular expression from above so that the keyword »factory« will get a hit when just »factor« is found.

7 Known Bugs and 'What to Do When...'

To Be Submitted

Compare all errors that have occurred and could not be explained, to the ones documented here. If you found anything new, please inform the author and send a zipped file with the following files:

- `spms.xml`
- `conference.config`
- `spmsbatch.pl`
- any other file that was involved in causing the error.

7.1 Selected Author not Highlighted in Affiliation

Drivotin, O.I.

Paper **Title**

1237   Simulation and Numerical Investigation of Self-Consistent Distributions for Charged Particle Beam

- **O.I. Drivotin**, V.V. Semenov
  St. Petersburg State University, Applied Mathematics & Control Processes Faculty, St. Petersburg

7.2 Selected Author Appears Twice in Affiliation and With Paper

van der Geer, S.B.

Paper   **Title**

WG421   Production of Ultra-Short, High-Brightness Waterbag Bunches

- **S.B. van der Geer**, O.J. Luiten, M.J. Van der Wiel, **S.B. van der Geer**
  TUE, Eindhoven

At Eindhoven University of Technology several closely related projects aim at the production of ultra-short high-brightness electron bunches. An overview of these efforts is given, with emphasis on micrometer precise rf-photogun manufacture and three dimensional compression of uniformly filled (waterbag) electron bunches.

WG421   Production of Ultra-Short, High-Brightness Waterbag Bunches

- **S.B. van der Geer**, O.J. Luiten, M.J. Van der Wiel, **S.B. van der Geer**
  TUE, Eindhoven

8 List of Scripts and Documentation with Versions and Change History

8.1 `keyword-sql.pl`

			Changes
Date	Version	Name	Comment
09. Dec 2005	0.9	volker rw schaa	• first try
12. Dec 2005	0.92	volker rw schaa	• minor changes
18. Dec 2005	0.95	volker rw schaa	• changed from SQL file to SMPS procedure (from »insert« to exec »spms.abstract_keywords«)
20. Mar 2006	0.99	volker rw schaa	• some updates to documentation
12. Jul 2006	1.0	volker rw schaa	• GPL license added

8.2 `pagecheck.pl`

			Changes
Date	Version	Name	Comment
03. Jun 2004	0.3	volker rw schaa	• first try
28. Apr 2006	0.4	volker rw schaa	•
May 2006	1.0	volker rw schaa	• small adjustments/final version
12. Jul 2006	1.1	volker rw schaa	• GPL license added
19. Jul 2006	1.2	john pool/vrws	• adjustment to paper size, which is sometimes 791/792. Now a ranges is accepted (592 > width > 597, 790 > height > 794)
23. Jul 2006	1.3	vrws	• pagecheck now reports a wrong pdf version (version > 1.5) and catches empty file errors

8.3 `scan-keywords.pl`

			Changes
Date	Version	Name	Comment
07. Jul 2003	0.9	volker rw schaa	• keywords in the list substituted by regular expressions in the following way: `electromagnetic-fields=electromagnetic.{0,1}field.{0,1}`, so now `electromagnetic-fields` is found when written as `electromagnetic_fields` `electromagnetic_field_` `electromagnetic-field_` `electromagnetic.↵fields` `electromagnetic.↵field_` `electromagnetic-↵fields` `electromagnetic-↵field_` Hit rate on DIPAC2003 more than 250% better
20. Dec 2003	1.0	volker rw schaa	• keyword list now loadable
	1.1	volker rw schaa	• script is now independent of settings in the spms script's config file
	1.2	volker rw schaa	• acronym list generated
07. Dec 2004	1.3	volker rw schaa	• read keyword list from current directory
19. May 2006	1.4	volker rw schaa	• acronym file not generated anymore
12. Jul 2006	1.5	volker rw schaa	• GPL license added

8.4 `spmsbatch.pl`

Changes			
Date	Version	Name	Comment
22. Apr 2003	0.6	volker rw schaa	
28. Apr 2003	0.8	volker rw schaa	
08. May 2003	0.9	volker rw schaa	
07. Jun 2003	0.92	volker rw schaa	• sort order corrected by using ArbBiLex
05. Oct 2003	0.95	volker rw schaa	• coloring of authors' index, • convertion of writing authors, abstracts, and institutes • using Unicode (&names; + &#entity;)
03. Nov 2003	0.96	volker rw schaa	• config file definitions refined, xml schema definition adapted
15. Nov 2003	1.0	volker rw schaa	• some minor nuisances taken care of (mailto, &# in config file, ...)
21. Jan 2004	1.1	volker rw schaa	• read additional file with production notes
03. Feb 2004	1.2	volker rw schaa	• abstracts will now be embedded into keyword and author list (in addition to session list)
09. Feb 2004	1.21	volker rw schaa	• change of some block related TEX commands into environments (\raggedright into flushleft)
10. Apr 2004	2.0	volker rw schaa	• adaption to matt's xml structure
22. Apr 2004	2.1	volker rw schaa	• nearly full adaption to matt's xml, • (re)start of documentation, • beautifying script after wild changes...
29. Apr 2004	2.2	volker rw schaa	• second stage of docu, • abstracts now internal instead of external, • script runs with Matt's test set (after some remedies: one line <keyword>-entries, supplying missing fields, etc.)
06. Jun 2004	3.0	volker rw schaa	• restart with very old version after disk crash, backup from old notebook not usable (???!), • files on new notebook weren't synchronized due to harddisk swap after initial tests, • status: loss of all adapted scripts • history: retyped from printed version's first and only page :-(
10. Jun 2004	3.1	volker rw schaa	• everything adapted, • docu missing, • scripts ugly, this is not a distributable version!! • authors are listed one per line in all displays (no combined entry per institute, tags <footnote> and <agency> are not honored)
20. Jul 2004	3.2	vrws	• extending the script for abstract booklet production with ConTEXt
30. Aug 2004	3.3	vrws	• change of hard-coded style info into style sheet for author and paper title, etc.
24. Sep 2004	3.8	vrws	• changed for CD production all file names to UPPERCASE
04. Oct 2004	3.9	vrws	• bugs fixed from introduction of UPPERCASE file names
11. Oct 2004	3.91	vrws	• all talks (the ones with a video = coupling) got transparencies to patch for missing files (edit key: #ä)

13. Oct 2004	3.92	vrws	• affiliation (contrib_abb/presenter_abb) is now composed of
			  • a) affiliation + town, if »Affiliation_abbrev« exists, otherwise
			  • b) affiliation.name1 + affiliation.name2 + Town.

17. Nov 2004	3.93	vrws	• keyword generation (search) from pdf files now in script »scan-keywords.pl« • output file will be read to get keywords ($raw_paper_directory/keyword-count.txt) • keyword generation from abstract text commented out (edit key: #ßß)
			• change history for version v3.94 – v4.03 lost
10. Oct 2005	4.04	vrws	• classification/subclassification PAC conferences have a different way of using class/subclass: a classification may not have a subclass, or may have entries under the classification and subclassification. This should be fixed now!
19. Oct 2005	4.05	vrws	• checking whether a slides file (<file_type abbrev="TRAN">Transparencies</file_type>) mentioned in XML is really there. A slides entry will only be written, when a file entry exists in $slides_directory
24. Oct 2005	4.06	vrws	• this version finally includes the (not so)special last name translation routine, which was already used in v 3.98/9, but somehow discarded. The translation to ISO-Latin 1 is surely not an good idea, we should decide on UTF8!
09. Nov 2005	4.07	vrws	• a bit more ISO 9660 in the scripts :-) all files now with .HTM

Date	Version	Name	Comment
			• clean-up of unused code
Nov 2005	4.08	vrws	• some new UTF8 code fragments added to translation table which are produced when file is stored directly (xml save, without cut/paste to file or `wget`)
10. Jan 2006	4.09	vrws	• lead/trailing spaces in all `<xname>` tags deleted • all code and comments deleted for keyword generation (now `scan-keyword.pl`)
11. Jan 2006	4.10	vrws	• more lead/trailing spaces in tags deleted, • sub classification string is now initialized (to »" "«) when classification is read • now a local default startup file »`INDEXLOC.HTM`« is created
14. Feb 2006	4.11	vrws	• Version 4.10 brought a new error by initialized sub classification string to "␣", the test still was on =""! • id for »@« from »@« changed to »﹫«
29. Apr 2006	4.12	vrws	• Reintroduction of provisions for page count checking (output file »`pages_per_paper.txt`«) • the new version of SPMS has a new tag structure `<coordinators>`...`</coordinators>` which contains an inconsistent use of the `<institute>` tag structure. The `<coordinators>`-code is skipped in reading
17. May 2006	4.13	vrws	• error corrected in transparency detection (uppercase/lowercase »`.pdf`«) • output of xml lines in debug log with »‖« • clean-up of config file (all unused variables deleted) • clean-up of debug lines
22. May 2006	5.0	vrws	• now optimized pdfs (for Fast Web View) will be produced by `gen_texpdf.bat`
28. May 2006	5.1	vrws	• some more variables initialized, so the number of error lines when processing non finalized xml files has been drastically reduced • in early stages of a conference 'New Affiliation Request Pending' might cause lots of error messages which are now reported, variables initialized • substitution of »$« and »%« were not line global but local, fixed now
12. Jun 2006	5.2	vrws	• in an early stage of a conference »`paper_code`«s are not present in the XML file while »`abstract_id`«s are. Code revised so that first »`abstract_id`« is taken for `$paper[]`, then in the next stage »`paper_code`« overwrites this setting, if present. Now (test) output can be generated without »`paper_code`« setting • the XML file is now opened as assumed »`UTF8`« input. So now the two and three byte »`UTF8`« codes (hopefully) are converted to the correct glyphs (still in test)
17. Jun 2006	5.3	vrws	• opening the XML as UTF8 helps in cases (FEL'06) but generates lots of errors in others (FLS'06); in addition the code is much slower using the utf8 option (0:10m compared to 2:07m for FEL'06). Now back to pre-version v5.2, code must be changed to handle more utf8 2 byte and 3 byte sequences
25. Jun 2006	5.4	vrws	• Matt has repaired the `<coordinators>`...`</coordinators>` tag structure, so now everything is read again
12. Jul 2006	5.5	vrws	• fixing error in xml when »chair« has no email (bug in generated XML) • fixed problem with ConTeXt output not being converted to TeX chars (some instances of author names and title lines escaped my notice) • changed substitution of »&« to »\&« • title entries in `KEYW*.HTM` and `AUTH*.HTM` extended to contain actual keyword and author. For `KEYW*.HTM` the actual keyword is added to the meta data »keywords« too. This is not directly visible on the web due to the framed setting, but indexing robots (Google, etc) which link them directly can show these infos now. • GPL license added

8.5 `readme-proceedings-script`

Changes			
Date	Version	Name	Comment
May 06	2.0	VRW Schaa	• new version: revive old TEX version which ended in 2005 leaving only fragments of code. Now incorporating several handwritten notes
June 06	2.1	VRW Schaa	• completely new and updated version (added: `Autorun.inf`, files delivered with script, Production of abstract booklet, etc.)
12. Jun 2006	2.1f	vrws	• •
17. Jun 2006	2.1g	vrws	• Instructions added how to download and install a ConTEXt stand-alone distribution (»CONTEXT minimal installation«)
25. Jun 2006	2.1h	vrws	• John Poole had problems running `texexec`; finally found that »Ruby« has to be installed. Installation description for »Ruby« added.
29. Jun 2006	2.1j	vrws	• tests reveiled that the template file »keyword-count.txt« was missing from »./paper-finals/« and therefore the script »spmsbatch« failed to complete successfully. File now added to SCS2006.
12. Jul 2006	4.1j	vrws	• changed version number to get in synch with Matt's versioning of the »Proceedings Scripts« on http://www-esh.fnal.gov/SPMS/.
23. Jul 2006	4.2	vrws	• Additions due to comments from Michael Abo-Bakr (Perl installation as non privileged user, WinEdt trial period, documentation for cURL deleted, error message from `spmsbatch.pl`) • documentation adapted to changed version of »pagecheck.pl« • LATEX description for use of special characters in Abstracts embedded and adapted (errors in script corrected)

9 GNU General Public License

```
#   <filename>              <$Version * #>
#   Copyright (C) 2002-2006 Gesellschaft fuer Schwerionenforschung mbH
#   <http://www.gsi.de> by Volker RW Schaa
#
#   ##### for the JACoW SPMS
#
#   This program is free software; you can redistribute it and/or modify
#   it under the terms of the GNU General Public License as published by
#   the Free Software Foundation; either version 2 of the License, or
#   (at your option) any later version.
#
#   This program is distributed in the hope that it will be useful,
#   but WITHOUT ANY WARRANTY; without even the implied warranty of
#   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#   GNU General Public License for more details.
#
#   You should have received a copy of the GNU General Public License
#   along with this program; if not, write to the Free Software
#   Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA
#   You can also get a copy of the license through the web at
#   <http://www.gnu.org/licenses/gpl.html>
```