

The cedram journal production system

Thierry Bouche

Cellule MathDoc & institut Fourier
Grenoble

EuroT_EX 2006, July 8th 2006, Debrecen

Outline

- 1 Introduction
- 2 User interface
- 3 Automatisation
- 4 Layout versatility
- 5 *Cahiers GUTenberg* nouvelle formule
- 6 Demo



cedram

- A French project to develop tools to help the production of academic math journals.
- A partnership between
 - CNRS, Ministry of research (funds),
 - the mathematical community and learned societies (orientations),
 - Cellule MathDoc (staff, metadata & Web expertise),
 - independant high quality research journals (AFST, AIF, AMBP, JTNB, ...).
- Up and running at www.cedram.org since March 2006.



Challenges

- Simple user interface (standard \LaTeX / \BibTeX input files).
- No metadata duplication.
- Paper, electronic, whatever view of a publication always in sync (*one process*).
- No 'house style' imposed to the participating journals.

Principles

- 1 Any metadata is input at most once in the system, in the relevant file.
- 2 Anything that is not deterministically determined by a given file —should stay away from that file.
- 3 Anything that can be computed —*should* be computed!
- 4 Do not reinvent the wheel, do not invent exotic formats that no one will master.
- 5 Stay pragmatic but avoid bottle-necks that would impact versatility of future use or quality of the output.

Concept

A journal
is a set of volumes,
made of issues,
made of articles,
plus various other material, mostly constant.

Core system

The production environment attaches to each level a set of files.

cedram.cls variant of `amsart.cls` (with add-ons from `smfart`, `smfthm`, `smfbib`): the core of the system.

Defines the input format, provides the general architecture for storing metadata, and many hooks to allow variable layouts.

Three modes of operations:

- Volume (builds a whole issue).
- Article (for each article).
- Special (for non-article material like TOCs).

Journal

A journal is defined by a `.clo` file (loaded at the end of `cedram.cls`) and a set of `.tex` inputs or specials:

cedram-CG.clo Journal-wide constants (title, ISSN, layout).

CG-front.tex Front matter inputs (title page, copyright page, addresses, scientific committee. . .).

CG-Sommaire.tex TOC template: a special loaded by the previous file.

CG-back.tex Back matter inputs (subscription info. . .).

CG-couverture.tex The cover template of an issue.

Issue

A journal issue is defined by a \LaTeX file with class `cedram` and options `JACRO,Volume`.

CG_46-47.tex The issue file defines:

- Numbers (year, month, volume, issue).
- First article's page number.
- Special issue title, if required.
- The ordered list of articles.
- Extras (editorial statements, ads. . .).

Article

An article is a directory containing all necessary data (images, inputs, bibliography) called from one master file bearing the same basename as the directory.

The article master file declares all article metadata, the journal and main language as options to `cedram.cls`.

Volume info and page numbers are *not* declared: they will be inherited from the issue where the article ultimately appears.

`devroye/devroye.tex` Article master file (language, title, author, abstracts, text, biblio, etc.).

`devroye/input.tex` \TeX inputs.

`devroye/devroye.bib` Bib \TeX file.

`devroye/img1.pdf` Illustration.

Automatisation devices—articles

The core functionalities provided by `cedram.cls` in article mode are:

- A bunch of options for various outputs. Screen version (with first page added, `hyperref` activated) is forced when preparing the final version.
- Automatic load of a `.cfg` file at `\begin{document}`.
- The `lastpage` trick.
- Write a \TeX line to an auxiliary file with all data pertaining to the article that could be used in a TOC.
- Write to an XML auxiliary file all metadata pertaining to the article that could be used elsewhere.

Compiling the article file produces the article PDF for the Web.

Automatisation devices—volumes

The core functionalities provided by `cedram.cls` in volume mode are:

- A bunch of options for various outputs (Cover, offprints. . .).
- Automatic write of a `.cfg` file for each included article.
- Compile each included article or special in a subprocess (`write18`) and include it in the volume PDF.
- Store the list of articles and assemble a volume TOC file from all article generated lines.
- Write a volume XML header and assemble a volume XML file from all article generated snippets.

Compiling the volume file produces a single PDF made of all inner pages of that volume.

Page numbers

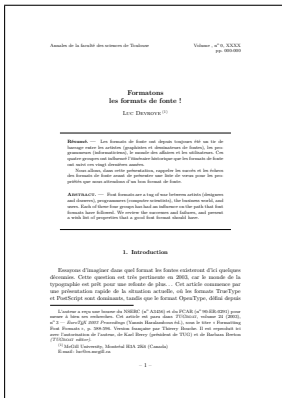
A page number is essentially a by-product of a completed issue, except the first page of an issue, it is set nowhere.

The trick is as simple as:

- 1 Initiate the volume's page counter before loading the first article.
- 2 Write the issue info and current volume's page number to the article's .cfg file.
- 3 Compile the article file twice so that all metadata, including issue info and page numbers are in final form.
- 4 Include the pages of the just generated article PDF (omitting the Web first page) in the volume PDF.
- 5 `\cleardoublepage`.
- 6 Start again.

Layout versatility

Same input, varying presentation (1).



Annales de l'institut Fourier

Annales de la faculté des sciences
de Toulouse (mathématiques)

Layout versatility

Same input, varying presentation (2).

Annales mathématiques Blaise-Pascal, 36(1), 2006-2007 (XXXXX)
**Formations
les formats de fonte !**
 LUC DEVROYE

Résumé

Les formats de fonte ont depuis toujours été un tir de langage entre les artistes (graphistes et designers) et les techniciens (les programmeurs informatiques), le monde des affaires et les utilisateurs. Ces quatre groupes ont influencé l'évolution historique que les formats de fonte ont eue ces vingt dernières années.

Mais alors, dans cette présentation, regardez les succès et les échecs des formats de fonte ainsi de présenter une liste de ceux pour les propriétaires que nous attendons d'un bon format de fonte.

Abstract

Font formats are a tug of war between artists (designers and users), programmers (computer scientists), the business world, and users. Each of these four groups has had an influence on the path that font formats have followed. We review the successes and failures, and present a wish list of properties that a good font format should have.

1. Introduction

Essayer d'imaginer dans quel format les fontes existaient d'ici quelques décennies. Cette question est très pertinente en 2003, car le monde de la typographie est prêt pour une révolution de plus... Cet article commence par une présentation rapide de la situation actuelle, où les formats TrueType et PostScript sont dominants, tandis que le format OpenType, défini depuis plus de huit ans, est en cours de présentation. Nous passons ensuite à une

Contenu à lire sur le serveur du NSERC (nr 4340) et du FCAR (nr 90-ER-4291) pour savoir à quoi ressemblent ces articles. Cet article est publié dans l'édition, volume 32 (2003), nr 3 - Revue GUTenberg (Nouvelles Recherches en IA), sous le titre « Formatting Font Formats », p. 309-336. Version française par Thierry Bouche. Il est reproduit ici avec l'autorisation de l'auteur, de Karl Berry (premier de TUG) et de Barbara Beeton (TUGboat editor).

1



 **FORMATONS
LES FORMATS DE FONTE !**
 © Luc Devroye

Résumé. Les formats de fonte ont depuis toujours été un tir de langage entre les artistes (graphistes et designers de fontes), les programmeurs informatiques, le monde des affaires et les utilisateurs. Ces quatre groupes ont influencé l'évolution historique que les formats de fonte ont eue ces vingt dernières années.

Mais alors, dans cette présentation, regardez les succès et les échecs des formats de fonte ainsi de présenter une liste de ceux pour les propriétaires que nous attendons d'un bon format de fonte.

Abstract. Font formats are a tug of war between artists (designers and users), programmers (computer scientists), the business world, and users. Each of these four groups has had an influence on the path that font formats have followed. We review the successes and failures, and present a wish list of properties that a good font format should have.

Contenu à lire sur le serveur du NSERC (nr 4340) et du FCAR (nr 90-ER-4291) pour savoir à quoi ressemblent ces articles. Cet article est publié dans l'édition, volume 32 (2003), nr 3 - Revue GUTenberg (Nouvelles Recherches en IA), sous le titre « Formatting Font Formats », p. 309-336. Version française par Thierry Bouche. Il est reproduit ici avec l'autorisation de l'auteur, de Karl Berry (premier de TUG) et de Barbara Beeton (TUGboat editor).

1. Introduction

Essayer d'imaginer dans quel format les fontes existaient d'ici quelques décennies. Cette question est très pertinente en 2003, car le monde de la typographie est prêt pour une révolution de plus... Cet article commence par une présentation rapide de la situation actuelle, où les formats TrueType et PostScript sont dominants, tandis que le format OpenType, défini depuis plus de huit ans, est en cours de présentation.

Cahiers GUTenberg nr 0 - XXXX, p. 000-000

Cahiers GUTenberg layout features

- Same page format.
- Utopia with Fourier (maths and typo extras) as main font.
- Small body size, but generously leaded for good running text experience.
- LMTT for typewriter style, no sans serif.
- Some graphical design elements like crop marks and oversized page numbers (typographer's private jokes?). Also headings downstairs.
- Wide margins with room for a second alignment for oversized floats.
- Dynamical placement of floats in the quest for lively double pages!

Demo

Everything I said can be demonstrated!

Thanks!