

Chaîne éditoriale
des nouveaux *Cahiers GUTenberg*

Thierry Bouche

Cellule MathDoc & institut Fourier
Grenoble

Journée GUT, 9 octobre 2006, Paris

Sommaire

- 1 Introduction
- 2 Interface utilisateur
- 3 Automatisation
- 4 Diversité de mise en page
- 5 Cahiers GUTenberg nouvelle formule

cedram

- Un projet français de soutien aux revues académiques de maths.
- Un partenariat entre
 - CNRS, ministère de la recherche (moyens financiers),
 - La communauté mathématique et les sociétés savantes (orientations),
 - la cellule MathDoc (moyens humains, travail sur les métadonnées et les formats).
 - et des journaux indépendants de bonne renommée scientifique (AFST, AIF, AMBP, JTNB, ...).
- En ligne sur www.cedram.org depuis mars 2006.

Défis

- Interface utilisateur simple (formats d'entrée : fichiers standards \LaTeX / \BibTeX).
- Pas de duplication de métadonnées.
- Toutes les versions d'une publication (papier, électronique. . .) doivent toujours être cohérentes (*un process*).
- Pas de « style maison » (maquette, marche. . .) imposé aux journaux participants.

Principes

- 1 Une métadonnée est entrée au plus une fois dans le système, dans le fichier qu'elle concerne.
- 2 Tout ce qui n'est pas déterminé par un fichier donné — ne doit pas se trouver dans ce fichier.
- 3 Tout ce qui peut être calculé — *doit* l'être !
- 4 Ne pas tout réinventer, ne pas pondre des formats exotiques que personne ne pourra maîtriser.
- 5 Rester pragmatique mais éviter les goulots d'étranglement qui pourraient limiter la diversité et la qualité des produits de la chaîne en aval.

Concept

Un journal

est un ensemble de tomes,

composés de fascicules,

composés d'articles,

plus d'autres éléments, majoritairement constants.

Systeme commun

L'environnement de production associe à chacun de ces niveaux un ensemble de fichiers.

cedram.cls variante d'amsart.cls (avec des ajouts de smfart, smfthm, smfbib) : le cœur du système.
Définit la structure des sources, fournit l'architecture pour la gestion des métadonnées, et de nombreux paramètres pour permettre toutes les maquettes.

Trois modes d'opération :

- Volume (génère un fascicule).
- Article (pour chaque article).
- Special (pour les autres éléments).

Journal

Un journal est défini par un fichier `.clo` (chargé depuis `cedram.cls`) et un ensemble de fichiers `.tex` :

cedram-CG.clo Les constantes du journal (titre, ISSN, maquette).

CG-front.tex Éléments préliminaires (page de titre, de copyright, adresses, comité de rédaction. . .).

CG-Sommaire.tex Canevas pour la table des matières : un « special » appelé par le fichier précédent.

CG-back.tex Éléments finaux (formulaire d'abonnement, pubs. . .).

CG-couverture.tex Le canevas pour la couverture d'un fascicule.

Fascicule

Un fascicule est défini par un fichier \LaTeX de classe cedram en mode « volume » (options JACRO,Volume).

CG_46-47.tex Le fichier fascicule définit :

- La tomainson (année, mois, tome, numéro).
- La première page du premier article.
- Le titre si nécessaire.
- La liste ordonnée des articles.
- Des extras (éditos, encarts, pubs. . .).

Article

Un « article » est un répertoire contenant tout ce qu'il faut pour l'inclure dans le volume papier mais aussi sur le site Web (images, inputs, bibliographie), en particulier le fichier maître `.tex` dont la compilation produit l'article.

Les métadonnées du fascicule, les numéros de page ne sont pas déclarés du tout : ils seront héritées du fascicule dans lequel l'article paraît.

`devroye/devroye.tex` Fichier maître article (langue, titre, auteur, résumés, texte, biblio, etc.).

`devroye/input.tex` Inputs \TeX .

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

`devroye/img1.pdf` Illustration.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — articles

Les fonctions de base de la classe cedram en mode article sont :

- Un paquet d'options. La version écran (screen) est imposée quand on prépare la publication définitive.
- Chargement automatique d'un fichier .cfg au `\begin{document}`.
- Le truc *lastpage*.
- Écriture d'une ligne de \TeX dans un fichier auxiliaire, comportant toutes les données de l'article qui pourraient servir à produire la ligne de sommaire correspondante.
- Écriture dans un fichier auxiliaire XML de toutes les métadonnées relatives à l'article.

La compilation du fichier article produit le PDF de cet article qui pourra être mis en ligne.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Automatisation — fascicules

Les fonctions de base de la classe cedram en mode fascicule sont :

- Un paquet d'options (couverture, tirés à part. . .).
- Écriture d'un fichier .cfg pour chacun des articles.
- Compilation de chaque article ou spécial dans un sous-process (write18), et inclusion dans le PDF.
- Mémorisation de la liste des articles traités, production à la volée de la table des matières du volume à partir des lignes de chaque article.
- Écriture de l'entête et production du XML du volume à partir des XML de chaque article.

La compilation du fichier fascicule produit un PDF de toutes les pages intérieures de ce volume.

Folios

Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomainson et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.
- 5 \cleardoublepage.
- 6 Recommencer pour le suivant.

Folios

Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomaison et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.
- 5 \cleardoublepage.
- 6 Recommencer pour le suivant.

Folios

Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomaison et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.
- 5 \cleardoublepage.
- 6 Recommencer pour le suivant.

Folios

Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomaison et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.

⑤ \cleardoublepage.

⑥ Recommencer pour le suivant.

Folios

Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomaison et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.
- 5 \cleardoublepage.

6 Recommencer pour le suivant.

Folios

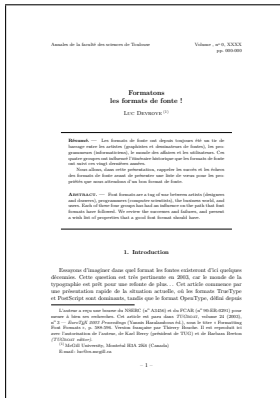
Un folio est en essence un sous-produit de la finalisation d'un fascicule : à part la première page du premier article d'un fascicule, il est toujours calculé.

L'astuce est aussi simple que ça :

- 1 Initialiser le compteur page du volume avant de traiter le premier article.
- 2 Écrire la tomainson et la page actuelle au sein du volume dans le fichier .cfg du premier article.
- 3 Compiler deux fois le fichier article.
- 4 Inclure les pages de cet article (sauf la page de garde pour le web) dans le PDF du volume en cours de production.
- 5 \cleardoublepage.
- 6 Recommencer pour le suivant.

Diversité de mise en page

Même source, présentation variable (1).



Annales de l'Institut Fourier

Annales de la faculté des sciences
de Toulouse (mathématiques)

Diversité de mise en page

Même source, présentation variable (2).



Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .

Nouvelle maquette pour les Cahiers

- Même format.
- Fonte de base : Utopia et son complément Fourier (maths et vignettes).
- Corps assez petit, mais interligne généreux pour une lecture confortable.
- Police « machine à écrire » : LMTT, pas de linéale.
- Maquette modérément déjantée : quelques éléments graphiques comme des hirondelles et des folios surdimensionnés. Les entêtes sont en bas.
- Marge généreuses avec un second module en réserve pour les flottants encombrants (code, hors-texte).
- Placement dynamique des flottants pour faire des double pages un peu vivantes. . .