

L_YX の高度な機能

L_YX プロジェクトチーム^{*1}

2011 年 5 月 15 日

目次

第 1 章 はじめに

第 2 章 L_AT_EX と L_AT_EX

- 2.1 L_AT_EX は L_AT_EX をどう使っているか
- 2.2 L_AT_EX ファイルを L_AT_EX に読み込む
- 2.3 L_AT_EX コードを挿入する
- 2.4 L_AT_EX と L_AT_EX プリアンブル
 - 2.4.1 L_AT_EX プリアンブルについて
 - 2.4.2 プリアンブルの変更
 - 2.4.3 例
 - 2.4.3.1 例 1: オフセット
 - 2.4.3.2 例 2: ラベル
 - 2.4.3.3 例 3: 段落の字下げ
 - 2.4.3.4 例 4: この文書
- 2.5 L_AT_EX および L_AT_EX のエラー

第 3 章 補完ツール

- 3.1 Bib_T_EX による参照文献のカスタマイズ
 - 3.1.1 他の引用様式
 - 3.1.2 節分けされた参考文献
 - 3.1.3 複数の参考文献
- 3.2 複合文書
 - 3.2.1 一般的な操作

	3.4.1	はじめに
	3.4.2	インタフェース
	3.4.3	使用法
第 4 章		L _Y X とインターネット網
	4.1	Math Output in XHTML
	4.2	Bibliography and Citations
	4.3	Indexes
	4.4	Nomenclature and Glossary
第 5 章		L _Y X サーバー
	5.1	はじめに
	5.2	L _Y X サーバーを開始するには
	5.3	通常の通信
	5.4	通知
	5.5	単純な L _Y X サーバプロトコル
	5.6	DVI と PDF の遡及検索
	5.6.1	自動有効化
	5.6.2	手動有効化
	5.6.3	特定のビューアを設定して使うには
	5.7	前方検索
第 6 章		特別な文書クラス
	6.1	A&A の論文
	6.1.1	はじめに
	6.1.2	最初の一步
	6.1.3	ヘッダ部
	6.1.4	概要
	6.1.5	サポートされる環境
	6.1.6	L _Y X でサポートされていないコマンド
	6.1.7	図フロートおよび表フロート
	6.1.8	Referee レイアウト

6.2.3	論文を書き終わったら	
6.2.4	特定のコマンドに関するコメント	
6.2.4.1	期待通りに動作するもの	
6.2.4.2	動作するが、若干説明が必要なもの	
6.2.4.3	実装されていないため、 \TeX コードを使用するもの	
6.2.4.4	実装できないもの	
6.2.5	FAQ とヒントと技とその他の考察	
6.2.5.1	\LyX を \AASTeX と協調させる	
6.2.5.2	表を処理する際の \LaTeX エラー	
6.2.5.3	参考文献	
6.2.5.4	EPS ファイルの読み込み	
6.2.5.5	可能だったが、実装しなかったもの	
6.2.6	最後に	
6.3	AMS \LaTeX	
6.3.1	これらのレイアウトが提供するもの	
6.4	AGU 各誌 (aguplus)	
6.4.1	概要	
6.4.2	新しいスタイル	
6.4.3	新しいフロート	
6.4.4	サポートされている学術誌	
6.4.5	バグと注意すべきこと	
6.5	Broadway	
6.5.1	はじめに	
6.5.2	特別な問題	
6.5.3	特別の機能	
6.5.4	用紙寸法と余白	
6.5.5	環境	
6.6	Dinbrief	
6.7	EGS 各誌 (egs)	
6.7.1	概要	
6.7.2	新しいスタイル	

	6.9.2.1	任意設定クラスオプション
6.9.3		サポートされる環境
6.9.4		スライドの組を作る
	6.9.4.1	タイトルページを作る
	6.9.4.2	新しいスライドを開始する
	6.9.4.3	定理・補題・証明など
	6.9.4.4	箇条書き
	6.9.4.5	図と表
	6.9.4.6	ページヘッダおよびページフッタ
6.9.5		サポートされない Foil _{TEX} 機能
	6.9.5.1	長さ
	6.9.5.2	ヘッダとフッタ
6.10		Hollywood (Hollywood スペック・スクリプト)
	6.10.1	はじめに
	6.10.2	特別な問題
	6.10.3	特別な機能
	6.10.4	用紙寸法と余白
	6.10.5	環境
	6.10.6	脚本で使われる特別な用語
6.11		ijmpc および ijmpd
	6.11.1	概観
	6.11.2	論文を書く
	6.11.3	原稿を投稿するための準備
	6.11.4	TEX コードの使用
6.12		iopart
	6.12.1	概要
	6.12.2	論文を書く
6.13		Kluwer
	6.13.1	概要
	6.13.2	論文を書く
	6.13.3	投稿の準備

6.14.3	letter (koma-script)
6.14.4	新しい letter クラス : letter (koma-script v.2)
6.14.5	問題点
6.15	Latex8 (IEEE 学会論文)
6.15.1	はじめに
6.15.2	はじめに
6.15.3	サポートされている環境
6.15.4	画面と印字の違い
6.16	Memoir
6.16.1	概観
6.16.2	基本的な機能と制約
6.16.3	追加された機能
6.17	mw クラス
6.18	Paper
6.19	RevTeX4
6.19.1	導入
6.19.2	プリアンブルに関すること
6.19.3	レイアウト
6.19.4	重要な注意
6.19.5	問題点
6.20	Springer 学術誌 (svjour)
6.20.1	説明
6.20.2	新スタイル
6.20.3	サポートしているジャーナル
6.20.4	Credits
6.20.5	バグ
6.21	Slides (別名 SLiTeX)
6.21.1	はじめに
6.21.2	使い始めるには
6.21.3	段落環境
6.21.3.1	サポートされている環境

	6.21.4.3 スライドで注釈を使う	
	6.21.5 slides クラスのひな型ファイル	
第 7 章	ソフトウェアを追加する必要がある LyX 機能	
7.1	TeX のチェック	
7.1.1	はじめに	
7.1.2	使いかた	
7.1.3	微調整のしかた	
7.2	LyX でのバージョン管理	
7.2.1	はじめに	
7.2.2	LyX における RCS コマンド	
7.2.2.1	登録	
7.2.2.2	変更をチェックインする	
7.2.2.3	チェックアウトして編集	
7.2.2.4	リポジトリのバージョンに戻す	
7.2.2.5	最後のチェックインを取り消す	
7.2.2.6	履歴を表示	
7.2.2.7	改訂情報	
7.2.3	LyX における CVS コマンド	
7.2.3.1	CVS の一般的な利用法	
7.2.3.2	登録	
7.2.3.3	変更のチェックイン	
7.2.3.4	変更のチェックアウト	
7.2.3.5	レポジトリのバージョンに戻す	
7.2.3.6	ローカルディレクトリチェックアウトのレポジトリから	
7.2.3.7	履歴を表示	
7.2.4	LyX における SVN コマンド	
7.2.4.1	登録	
7.2.4.2	変更をチェックイン	
7.2.4.3	チェックアウトして編集	
7.2.4.4	保存されたバージョンに戻す	

	7.2.4.9	文書の改訂情報	1
7.2.5		SVN と Windows 環境	1
	7.2.5.1	準備	1
	7.2.5.2	文書を Subversion 管理下に置くには	1
	7.2.5.3	Windows 下で SVN と共に SSH トンネルを使うには	1
	7.2.5.4	行末変換	1
7.2.6		さらなる調整	1
7.2.7		バージョン管理と文書比較	1
7.3		文芸的プログラミング	1
	7.3.1	はじめに	1
	7.3.2	文芸的プログラミング	1
	7.3.2.1	参考文献	1
	7.3.3	LyX と文芸的プログラミング	1
	7.3.3.1	ドキュメントとコードの生成 (weaving および tangling)	1
	7.3.3.2	LyX の設定	1
	7.3.3.3	デバッグ機能の拡張	1
	7.3.3.4	ツールバーの拡張	1
	7.3.3.5	色を変える	1
第 8 章		LaTeX の達人の秘密	1
8.1		多段組み	1
	8.1.1	目的	1
	8.1.2	制限事項	1
	8.1.3	例	1
	8.1.3.1	二段組み	1
	8.1.3.2	多段組み	1
	8.1.3.3	カラムの中のカラム	1
8.2		箇条書き (連番) の番号振り	1
8.3		Dropped Capitals	1
8.4		標準的でない形の段落	1
8.5		まとめ	1

第 1 章

はじめに

いまお読みの取扱説明書『 L_YX 拡張機能篇』は、基本的には『ユーザーの手引き』第二部です。この文書を別部とした理由は、単純に『ユーザーの手引き』がすでに大部なっていることと、『ユーザーの手引き』が、ほとんどの文書を作成する上で必要な機能をすべて説明してしまっているためです。一方で、 L_YX プロジェクトチームは、 L_Y を各設定ファイルと外部パッケージによって拡張可能な状態にしておくことを長期的な標にしています。つまり、ある $L_A\TeX$ パッケージをサポートしたい場合に、 L_YX 自身変更を加えることなく、当該パッケージ用のレイアウトファイル（ないしモジュール）作成すればいいようにしているのです。すでにこのような形で、いくつもの新機能がユーザーによって追加されています。この取扱説明書は、そのような形での貢献を文書化する冊子として位置付けられています。

この取扱説明書では、ファックスのサポート・バージョン管理・SGML サポートのうな、正しく機能するためには追加的なソフトウェアを必要とする特殊機能についても明しています。さらに、 $L_A\TeX$ の強力な機能を直接使って文書を美しく仕上げるための $L_A\TeX$ ツールやテクニックを述べた章も収録しています。つきつめれば、 L_YX とはあくで WYSIWYM たることを企図しているため、 $L_A\TeX$ の全機能の一部へのインタフェースは用意するものの、すべてを用意することはないでしょう。

もし、まだ『はじめの一步』をお読みでなかったら、この取扱説明書から読みはじめのは誤りです。『はじめの一步』には、全取扱説明書に共通の表記法と書式が説明されていますので、まずそれからお読みください。さらに、この取扱説明書を読もうとするに、『ユーザーの手引き』をよくお読みになって、 L_YX の全基本機能を習得してください。

第 2 章

LyX と L^AT_EX

2.1 LyX は L^AT_EX をどう使っているか

本章は、T_EX 熟練者と L^AT_EX に興味のある方向けの章です。ここでは、LyX と L^AT_EX が、印刷可能な出力を生成するのに、どのように協働しているのかを説明します。全取説明書中で本章だけは、L^AT_EX のことを多少なりともご存知であることを前提とします。

かつて LyX は、「L^AT_EX の WYSIWYM なフロントエンド」と呼ばれていましたがいまではこれは正しくありません。L^AT_EX のフロントエンドは、いくつも出回っています*1。これらは、基本的に L^AT_EX を走らせる機能を有し、編集中のファイルのエラーを示す機能を持ったテキストエディタです。LyX も、エディタであり、L^AT_EX を走らせる機能を有し、ファイル中のエラーを指摘しますが、それ以上にもっともっと多くのことをしてくれます。ひとつには、LyX を効果的に使うのに、L^AT_EX の知識が必要とされないことが挙げられます。また LyX は、それ自身、いくつかの拡張を L^AT_EX に加えています。時間のあるときにでも、いちど以下の手順を試してみてください。まずファイルメニューからエクスポート▷L^AT_EX（あるいは表示▷ソースを閲覧）を選択し、そこから出力される .tex ファイルのプリアンブルをご覧ください。LyX によって特別に定義された新しいマクロが複数定義されていることに気づかれるはずです。これらのマクロは、文書中で使用する機能に応じて、自動的に定義されます。

自動的に L^AT_EX を実行するコマンドが、以下のようにいくつかあります。

- 表示▷書式
- 表示▷更新▷書式

• ファイル▷ファックス

これらは、最後に L^AT_EX を実行してからファイルに変更が加えられた場合のみ、L_YX が実行します。

L_YX が編集中のファイルを L^AT_EX にかけるとき、L_YX は以下の段階をふみます。

1. 文書を L^AT_EX に変換し、ファイルの拡張子を .lyx から .tex に変えて保存。
2. この .tex ファイルに対して（必要に応じて複数回）L^AT_EX を実行し、L^AT_EX ファイルをコンパイルするのに必要な他のコマンド（bibtex や makeindex など）を実行。。
3. もしエラーがあれば、エラーログを表示。

L^AT_EX を表示▷DVI を使用して実行した場合には、L_YX はこの後、DVI ビューアを使用して DVI ファイルを表示します。表示▷PostScript を使用した場合には、L_YX は以下の段階をふみます。

- dvips を実行して、DVI ファイルを PostScript® に変換。
- ghostview などの PostScript® ビューアを実行して、PostScript® ファイルを表示。

L_YX は、他の書式を表示したりエクスポートしたりする際にも、同様のことを行い

2.2 L^AT_EX ファイルを L_YX に読み込む

L_YX でファイル▷インポート▷L^AT_EX コマンドを使用すると、L^AT_EX ファイルを L_YX にインポートすることができます。このコマンドは、tex2lyx というプログラムで実行し、foo.tex などというファイルから foo.lyx を生成します。L_YX は、この変換されたファイルを開きます*2。

tex2lyx は、文法的に正しい L^AT_EX ファイルのほとんどを変換することができます。すべてを変換できるわけではありません。tex2lyx は、理解不能な部分は T_EX コードボックスの形で残しますので、ファイルを tex2lyx で変換したのち、T_EX コードボックスを探して、手動で正しく編集してください。

T_EX コードボックスがなにかご存じなければ、次節をお読みください。

2.3 L^AT_EX コードを挿入する

どの L_YX 文書にもつねに T_EX コードを挿入することが可能であるというきわめて単なる理由によって、L^AT_EX にできることは、つねに L_YX でも行うことができます。一方 L_YX は、すべての L^AT_EX 構成物を L_YX 内に表示できるわけではなく、将来的にもそうならないでしょう。かりに L^AT_EX コマンドを L_YX 文書に挿入する必要が出てきた場合は、T_EX コードボックスを使用してください。これは、挿入▷T_EX コードから、文書中挿入することができます。

以下は、L^AT_EX コマンドを L_YX 文書中に挿入する例です。コードは以下のようになっているとしましょう。

```
\begin{tabular}{ll}
\begin{minipage}{5cm}
これは、ミニページ環境の例です。
これには、
(フLOATでない) 図表を
含め、ほとんど何でも
入れることができます。
\end{minipage}
&
\begin{minipage}{5cm}
\begin{verbatim}
\begin{minipage}{5cm}
これ...
\end{minipage}
\end{verbatim}
\end{minipage}
\end{tabular}
```

上記のテキストを含んだ T_EX コードボックスが、この段落の直後にあります。この取扱明書を L_YX 上でご覧になっていれば、T_EX コード差込枠のみが見えるはずです。印刷

これは、ミニページ環境の例です。

これには、(フロートでない)図表 `\begin{minipage}{5cm}`
を含め、ほとんど何でも入れるこ `これ... \end{minipage}`
とができます。

TeX コードを使用する方法に加えて、複雑な LaTeX コードを含む部分を別ファイルで作成しておき、そのファイルをインクルードするのに挿入▷ファイル▷子文書する方法もあります(取り込みの型として Input を選択しなくてはなりません)。法をとる場合には、.tex ファイルが正しく動作することが確認できている場合のみの方がよいでしょう。さもないと、LaTeX エラーを追跡するたいへんな作業を行わはなりません。

最後に、いくつか強調しておくべきことがあります。

- LyX は、あなたの LaTeX コードが正しいかどうかをチェックしません。
- 車輪の再発明をしないようにご注意ください。

二番目の点に関して、LyX はひじょうにたくさんの機能を装備しており、その数はづけています。ご自身で LaTeX コードを書かなくてはならないと思ひ込む前に、これこれの機能をもっていないことを、取扱説明書で確実に確かめてください。さの中には、封筒のラベルから美しい複数ページにわたる表を作るものまで、ありとることを行う膨大な数の LaTeX パッケージが存在します。詳細については、[CTAN](#) 8節をご覧ください。

もし、文書中で例外的に手の込んだことをなさりたいければ、かならず良い LaTeX 書籍を参考とするようにしてください。『ユーザーの手引き』の参考文献に多くの LaTeX 関連書籍が引用されています。

2.4 LyX と LaTeX プリアンブル

2.4.1 LaTeX プリアンブルについて

すでに LaTeX についてご存知であれば、ここでプリアンブル(前置文)が何に向るか説明する必要はないでしょう。もしご存じなければ、以下の説明で大まかな理られるはずです。詳しくは LaTeX 関連書籍をご参照ください。いずれにしても、LyX 文書の LaTeX プリアンブルで何ができるか、または何を必要がないか説

2.4 L_AT_EX と L_AT_EX プリアンブル

- 文書クラス（ドキュメントクラス）の宣言。

これは L_AT_EX がすでに行ってくれています。もしあなたが熟達した L_AT_EX 使いであり、自作の文書クラスを使いたい場合には、取扱説明書『カスタマイズ篇』をご覧になれば、L_AT_EX に自作文書クラスへの仲介を行わせるにはどうすればよいかの報があります。

- パッケージ使用の宣言。

L_AT_EX パッケージは、パッケージがプリアンブルで宣言された場合のみ、文書中利用可能となる、追加コマンド群を提供します。例えば、`indentfirst` パッケージは、すべての段落の最初の行を字下げします。他にも、ラベルや宛名書き、余設定等の様々なパッケージがあります。

- カウンタ・変数・長さ・幅の設定。

L_AT_EX のカウンタと変数の中には、プリアンブルでグローバルに設定しないかぎり、望んだ効果を持たないものがあります。（文書中で設定したり再設定したりすることのできる変数もあります。）余白は、プリアンブルで設定しなくてはならない変数の代表例です。他には、箇条書きに付けるラベル書式が挙げられます。じは、この設定は文書中どこでもできるのですが、プリアンブルでいちどだけ設定するのが望ましい方法です。

- ユーザー定義コマンドの宣言（`\newcommand` または `\renewcommand` を使用）。
ユーザー定義コマンドは、文書中で頻出する L_AT_EX コマンドの短縮形です。このようなコマンドを宣言するには、プリアンブルが適した場所ですが、どこで宣言することも可能です（もちろんコマンドが初めて使用される場所よりも前です）。まりないことではありますが、お使いの文書中に生の L_AT_EX コードがあふれる場合には便利でしょう。

L_AT_EX は、その生成する `.tex` ファイルのプリアンブルに、それ自身の定義を追加します。これによって、L_AT_EX の生成する L_AT_EX ファイルの可搬性を高めています。

2.4.2 プリアンブルの変更

L_AT_EX が L_AT_EX ファイルのプリアンブルに追加するコマンドは、つねに同じです。これは、L_AT_EX 自体にパッチを当てないかぎり変更できないようになっています。しかしながら、文書へ設定ダイアログの L_AT_EX プリアンブルの項から、プリアンブルに追加をする

の再発明をしないように、とはすでに申しあげたことです)。また、プリアンブルが正しいことを確認してください。L_AT_EX は、プリアンブルをチェックしません。エラーがあれば、「Missing `\begin{document}`」といったエラーが出ることでしょ。のようなエラーが出たときは、プリアンブルを再確認してください。

2.4.3 例

以下は、プリアンブルに追加できるものとその効果を示す例です。

2.4.3.1 例1：オフセット

L_AT_EX には、`\hoffset` と `\voffset` というページ位置を制御する 2 つの変数があります。その機能は名前から明らかなとおりです。これらの変数は、ラベルを印刷するに便利です。往々にして、印刷用紙の寸法とそのうち実際に印字可能な範囲は同一ありません。ここに `\hoffset` と `\voffset` が役立つわけです。

`\hoffset` と `\voffset` の既定値はともに 0 ポイント、つまりページをシフトしません。残念ながら、DVI ドライバによってはつねにページをシフトすることがあるようです。我々には、どうしてそのようなことをするのか、そしてシスアドがどうしてそれを行わないのか理解できませんが、もし個人的に管理できないシステムで L_AT_EX を使用してシスアドがお馬鹿さんである場合には、`\hoffset` と `\voffset` が役に立ちます。たとえば上余白と左余白がつねに 0.5 インチずつ大きすぎるものとしましょう。その場合はプリアンブルに追加するとよいでしょう。

```
\setlength{\hoffset}{-0.5 in}
\setlength{\voffset}{-0.5 in}
```

これで余白が修正されます。

2.4.3.2 例2：ラベル

いま、たくさんの宛名ラベルを印刷をしたいものとしましょう。ラベルシートを作るために便利な `labels.sty` という L_AT_EX パッケージが、お近くの CTAN アーカイブで入手可能です。お使いのシステムには、既定ではこのパッケージは導入されていないかもしれません。ご自身でお確かめください。また、このパッケージの取扱説明書を読

2.4 L^AT_EX と L^AT_EX プリアンプル

```
\usepackage{labels}  
\LabelCols=3  
\LabelRows=7  
\LeftBorder=8mm  
\RightBorder=8mm  
\TopBorder=9mm  
\BottomBorder=2mm
```

これは Avery® ラベルシートのストック 5360 番を使用する場合の設定です。これで、宛名を `\begin{labels}` と `\end{labels}` で囲めば、宛名ラベルの印刷をすることができます。本機能およびその他の機能に関しては、`labels.sty` の取扱説明書に説明されています。

そのうち誰かが、このパッケージを直接サポートする L^AT_EX レイアウトファイルを作ってくれるかもしれません。その誰かはあなたかもしれません。

2.4.3.3 例 3：段落の字下げ

アメリカ人は、すべての段落の最初の行を字下げするように教育されています。そのために彼らの他の変な習慣と同じく、いつものようにすべての段落の最初の行を下げるので、アメリカ人はごちゃごちゃと文句を言いつづけるのです。(これはユーモアです。(ほんと?) ユーモアですってば。)

もちろん、これは標準的な組版ルールではありません。書籍では、典型的には、他の段落のあとにくる段落のみ字下げを行ないます。一行めを字下げをするそもそもの目的は隣接する段落から区別するためです。したがって、図のあとや節の最初の段落など、先する段落がない場合には、字下げをする必要はないのです。

けれどもあなたが典型的なアメリカ人なら(ユーモアですって!) そのような小さいことはかまわずに、いつもどおりの字下げをせずにはいられないでしょう! プリアンプルに次の行を加えて下さい。

```
\usepackage{indentfirst}
```

もしお使いの T_EX 頒布版が脳死状態でなければ、このパッケージが入っているはずなので、建国の父たちが意図したように、すべての段落が字下げされます。

ます。

2.5 L_YX および L^AT_EX のエラー

L_YX が L^AT_EX を呼び出す際、L_YX は L^AT_EX に、かりにエラーがあったとしても無視して先に進むように指示します。その後 L_YX は、L^AT_EX の実行ログファイル後処理用に使います。ログファイルを分析したのち、L_YX はエラーを一覧表示するアラログを表示します。エラーのうちどれかをクリックすると、L_YX ファイル中、エラーの発生した場所に表示を移動します*³。

人によっては、ログファイルを直接ご覧になりたいかもしれません。これは、L^AT_EX ログで見ることができます。エラーメッセージや警告には、よく見かけるものもつかありますので、それらをここに収録しておきます。ここにはないエラーや警告では、L^AT_EX 関連の良書を参照してください。

- L^AT_EX Warning

これで始まるメッセージは、L^AT_EX コード自体を「デバッグ」するための警告メッセージです。相互参照や参考文献を追加したり変更した場合に、もう一度を実行する必要があることを知らせるために、このようなメッセージが表示されます。このメッセージは、だいたいにおいて無視しても大丈夫です。

- L^AT_EX Font Warning

L^AT_EX がフォントを見つけられなかったという警告メッセージです。多くの場合、このメッセージの後に、L^AT_EX が使用した代替フォントに関する情報が続きます。これも無視して構いません。

- Overfull \hbox

L^AT_EX は、ほぼどんなときでもこの警告を吐き出します。これは、一行が長すぎて右余白にはみ出してしまったという警告ですが、ほとんどの場合、最終出力でも判別できないくらい(せいぜい1ポイントか2ポイント)です。あるいは、か2文字右余白に出ているくらいです。L^AT_EX は、どんな文書を書いても、とも一つはこのメッセージを出すようです。

このメッセージは無視しても大丈夫です。何か行が長すぎるような問題があ

2.5 L_YX および L^AT_EX のエラー

目で見ればわかります。出力を目で確認してください*4。

- Underfull \hbox

これは上の警告と違ってあまり出てきません。なぜだかは分かりませんが、L^AT_EX は 1 行を短めにする印字するよりは、長めにする方が好きなようです。

- Overfull \vbox と Underfull \vbox

ページ区切りに問題があったという警告です。この場合も、出力を目で確認してください。何か問題があれば、見てすぐわかります。

- L^AT_EX Error: File 'Xxxx' not found

ファイル「Xxxx」が、システムに導入されていません。これは通常、文書が必要している L^AT_EX パッケージに、導入済でないものがある場合に現れます。プリントルに変更を加えておらず、`\usepackage{}` コマンドも使用していないとすれば、L_YX が読み込もうしているパッケージに存在しないものがあるということです。ヘルプ▷L^AT_EX の設定を使用して、L_YX が認識しているパッケージの一覧をてください。このファイルは、(ツール▷再設定を使用して) L_YX の再設定を行たびに更新され、どのパッケージが検出されたか、それらのパッケージが何をすものかを表示します。

`\usepackage{}` コマンドを使っていて、肝心のパッケージが導入されていない場には、ご自身で導入する必要があります。

- L^AT_EX Error: Unknown option

このように始まるエラーメッセージは、パッケージに誤ったあるいは未定義のオプションが指定されていることを伝えようとするものです。パッケージの取扱説明をご確認ください。

- Undefined control sequence

お使いの文書に L^AT_EX コードを挿入した際、打ち間違いをしてしまうと、このようなメッセージが表示されます。あるいは、パッケージを読み込ませるのを忘れたかもしれません。いずれにしても、このメッセージは、あなたが未定義のコマンドを使用したことを示しています。

これ以外にも、エラーメッセージや警告メッセージがあります。一般に、読んで意味がぐにわかるものは、L^AT_EX のメッセージです。何を言っているのかさっぱりわからないものもありますが、通常これらは、T_EX のメッセージであり、それが何を意味している

エラーメッセージが表示された場合は、つねに以下のチェックリストにしたがうよいでしょう。

1. ご自分で挿入した L^AT_EX コードに打ち間違いがないかどうか確かめる。
2. 打ち間違いがなければ、コマンドを正しく使ったかどうか確かめる。
3. 文書のもっともはじめの部分にエラーボックスが多数積み上がった場合には `Missing \begin{document}` が表示されているならば、プリアンブルに `\begin{document}` があることを意味します。プリアンブルのデバックを行ってください。
4. プリアンブルに何も追加せず、文書に L^AT_EX コードをまったく挿入しない場合には、L^AT_EX 頒布版自体をまず疑ってください。導入されていないパッケージを探しだして、導入してください。
5. 導入されていないパッケージはなかったものとしましょう。L_YX を微調整オプションを使用していますか。特に、手動で Protected Blank や Linebreak、Page などたくさん挿入するなど、微調整オプションを悪用していませんか。段落環境を使用する代わりに、これらを使用してその場をしのごうとしていませんか。
6. 微調整オプションはまったく使用しておらず、規則に乗っ取って作業していませんか。何か曲芸のようなことをしようとはしていませんか。数式で、たとえば表のセルに図を入れるなど、おかしいことをしようとしていますか。
7. L^AT_EX が改行位置を見つけられないほど長い一節がありませんか。既定では、改行位置を決定するために単語間に空白をたくさん追加することを、L^AT_EX はかなり嫌いかもしれません。可能であれば、問題が生じないように段落を修正するのが望ましいでしょう。難しければ、L^AT_EX が改行しやすく (sloppy) なるように、その部分を `\sloppy` で囲ってください。
8. 入れ子の階層を深くしすぎていませんか。L_YX は、(今のところ) 環境の入れ子の階層が制限内にあるかチェックしません。かりに、環境を 17 段の入れ子にしていたら、それが原因です (L^AT_EX の上限は 5 段です)。
9. エラーメッセージは発生しないけれども、出力がひどいものとしましょう。
 - 幅が広すぎたり長すぎてページに収まらない場合には、
 - (a) 図がページに合うように大きさを変える、
 - (b) 表がページに合うように削る、

2.5 L_YX および L^AT_EX のエラー

これらすべてが役に立たないとすれば、ひょっとしたら L_YX のバグを発見したのかもしれませんが。

第 3 章

補完ツール

3.1 BibTeX による参考文献のカスタマイズ

BibTeX の基本的な使用法は、『ユーザーの手引き』の参考文献データベース (*BibTeX* の節に説明があります。以下の各小節では、*LYX* でサポートされている参考文献の特別な機能について説明することとします。

3.1.1 他の引用様式

標準の BibTeX は、引用された著作を参照するのに、([12] のように) 数字を使用します。しかし、多くの科学分野で、これとは異なる引用様式が用いられています。もっとよく見られるのは、著者-出版年様式 (例: Knuth 1984a) です。*LYX* は、この様式を提供する 2 つのパッケージ *natbib* と *jurabib* をサポートしています。これらのパッケージには、それぞれ長所短所がありますが、ここでは詳細に立ち入る紙幅がありません。純な著者-出版年様式 (あるいは著者-番号) 様式が欲しい場合や、無数にある *natbib* スタイルファイルのうちどれかを使用したい場合には、定評のある *natbib* パッケージおそらくよい選択でしょう。短縮タイトルによる参照や *ibid.* など、特別な機能が必要な場合には、*jurabib* パッケージの利用を考えるとよいかもしれません。

LYX で の両パッケージの取扱い方は、ほぼ同じです。文書▷設定メニューを開き、参考文献項目から *Natbib* オプションか *Jurabib* オプションを選択してください。これらパッケージによって、引用ダイアログに機能が追加され、参照の様式が選択できるようになります ([Knuth, 1984], [Knuth, 1984], [Knuth, 1984], [1984, など])。両パッケ

在します*¹。

3.1.2 節分けされた参考文献

場合によっては、参考文献をいくつかの節に分ける必要があることがあります。ば、あなたが歴史家であるとすれば、出典と学術文献を分ける機能は、おそらく「必須」でさえあります。残念ながら、BibTeX 自身ではこのようなことができません。適切な L^AT_EX パッケージの助けを借りると、BibTeX をこのような必要性を満たすに拡張することができます。

L^AT_EX は、このようなパッケージの一つである bibtopic*²をそのままサポートします。このパッケージの長所は、(multibib のような他のパッケージと比較すると)引用コマンドを定義する必要がないことです。そのかわりに、参考文献の違う節にの節向けの項目を収録した別の参考文献データベースを用意する必要があります。ば、参考文献を「出典」の節と「学術文献」の節に分けたい場合には、それぞれのデータベース、たとえば sources.bib と scientific.bib とを作成する必要があります。

文書▷設定を開いて参考文献の下節分けされた参考文献をチェックしてください。すると、参考文献の各節に一つずつ、複数の参考文献を挿入することができます。上記は、BibTeX 参考文献 sources.bib と二つめの参考文献データベース scientific.bib を挿入します。各節に同じスタイルを用いるか異なるスタイルを用いるかは自由で、わえて、参考文献の節に、指定したデータベースの「引用されたすべての参照」を載せる(これが既定です)か、「引用されなかったすべての参照」を載せるか、あるいは「すべての参照」を載せるかを選択することができます。これは、もし参考文献を「本用された出典」・「本文に引用されていない出典」・「学術文献」の三つの節に分割した場合に、便利でありましょう。各節の見出しは、通常の節や小節として入れることができます。bibtopic は、参考文献用の見出しを取り除いてしまいますので、手動でこれしなくてはなりません(たとえば章*や節*などをお使いください)。

3.1.3 複数の参考文献

文書の節ごとあるいは章ごとの参考文献など、複数の参考文献は BibTeX 自身サポートされていません。しかしながら、もし若干の T_EX コード(第 2.3 節を参照)を実行することを厭わなければ、L^AT_EX で節分けされた参考文献を作るのに使用されています。

3.2 複合文書

を参照) bibtopic パッケージが、この課題を解決する簡単な方法を提供します*3。

まず文書▷設定に行って参考文献の下節分けされた参考文献を有効にしてください。つぎに文書中において、参考文献 (BibT_EX 参考文献本体を含む) の入った節 `\begin{btUnit}` と `\end{btUnit}` のあいだにはさまなくてはなりません (これらのコンドは T_EX コードとして挿入する必要があります)。参考文献には、現在位置の btUn において引用されたすべての参照が含まれることになります。(註) この方法を用いる場合には、すべての参照はいずれかの btUnit 内部になくなくてはなりません。また、btUn は入れ子構造にすることができません。

3.2 複合文書

3.2.1 一般的な操作

たくさんの節を含んだ大きなファイルで作業をする場合、文書をいくつかのファイルに分割した方が往々にして便利です。場合によっては、表部分にはしょっちゅう変が加えられるけれども、その前の文章はつねに同じであるようなケースもあるでしょう。このような場合には、複合文書を利用することを、まずご検討ください。たとえば、学論文は通常、導入部・所見・結果・議論・結論の五つの節から成ります。これら各節別々の L_YX ファイルにして、タイトル・著者・概要・参考文献等と、これら五つのインクルード*4 ファイルを持つ、もう一つ別の「マスター」ファイルを作成することができます。これらの各ファイルは、単にマスターファイルにインクルードされだけでなく、単独でも出力したり印刷したりすることのできる完全な L_YX ファイルのように注意することが重要です。しかも、これらの各ファイルは、同じ文書クラスなくてはなりません。article クラスと book クラスを混在させるようなことはなさいでください。L_AT_EX ファイルをインクルードすることもできますが、このファイルはプリアンブルがあってはなりません (`\begin{document}` 行とその前のすべて、および `\end{document}` 行は削除しておかなければなりません)。さもないと、DVI ファイル作成する際にエラーが発生するでしょう。

L_YX では、挿入▷ファイル▷子文書で簡単にファイルをインクルードすることができます。このメニューを選んでクリックすると、カーソルの現在位置に小さな枠が挿入されます。この枠をクリックすると、ダイアログが現れるので、そこでインクルードするファ

ファイル選択ボックスは、いまや自明でしょう。インクルード方法には、「include」、「input」、「verbatim」の三つがあります。「include」と「input」の違いは、正確にはの達人にしか意味のないものですが、ほとんどの人にとって意味のある違いといえます。「include」されたファイルは新規頁から組版されるのに対し、「input」されたファイルは現在のページから組版される点です。

一般に、マスターファイルは、組版前に完全な L^AT_EX ファイルに変換されますが、インクルードされるファイルは、プリアンブル情報をまったく持たない L^AT_EX ファイルに変換されます。

「verbatim」インクルードファイルでは、ファイルの内容がそのまま(つまり「verbatim」モードで)固定幅タイプライタフォントで組版されます。通常、このファイルの空白は通常の L^AT_EX での空白の取扱いとは違って、連続した複数の空白がそのまま出力されるので、単なる見えない空白です。ただし、出力の空白に印付けをするチェックボックスを有効にすると、空白が存在することを明確にするために空白の印が出力されます。

3.2.2 ファイル間の相互参照

この節の内容は、すでに古くなっています。既定のマスター文書についてと、マスター文書を開いているときに子文書をどう開くかについて記述する必要があります。[[要修]

異なるファイル間に相互参照を設定することも可能です。まず、すべての関係するファイルを開いてください。例として、2 つのファイルを扱う場合を考え、各ファイルを A と呼びます。B が A にインクルードされるファイルだとします。A にラベルを付けて B で参照したいとしましょう。文書 B で相互参照ダイアログを開いて、使用する「ファイル」を選んでください。

3.2.3 すべての子文書での参考文献

この節も更新が必要です。この点に関する内容が wiki にあるので、それをこの節に更新することも可か。

すべての子文書の全文献項目を含んだ参考文献一覧をコピーし、それをコメントアウトしてください。このようにすることで、L^AT_EX は .bib ファイルを見つけることができ、参考文献一覧を見えるようにせずとも簡単に参照を挿入することができるようになります。

3.3 装飾的なヘッダ・フッタ

さい。

```
\usepackage{comment}
\includecomment{comment}
```

<http://wiki.lyx.org/FAQ/Unsorted#toc31>も参照してください。

3.3 装飾的なヘッダ・フッタ

既定のページレイアウトは、どちらかといえば素っ気ないものです。article 文書クラスの場合には、ページの下の方真ん中にページ番号がつくだけです。この取扱説明書 KOMA-script の book クラスなので、すこしは凝ってますが、ほんとうに着飾るには『ユーザーの手引き』に述べられているように、文書ページスタイルを「装飾的 (fancy)」に設定する必要があります。この節では、望んだとおりの効果を得るために、 \LaTeX プリアンブルに挿入する必要のある \LaTeX コードについて解説します。

ページヘッダは、「left (左)」、「center (中央)」、「right (右)」と命名された 3 つのフィールドに分けられています。フッタも同様に 3 つのフィールドに分類されています。これらのフィールドをもっとも簡単に設定する \LaTeX コマンドには、`\lhead`・`\chead`・`\rhead`・`\lfoot` などがあります。かりに、各ページの左上端に自分の名前を入れたいのとしましょう。次のようなコマンドをプリアンブルに入れるだけです。

```
\lhead{物書 はじめ}
```

これで自分の名前が左上に現れます。フィールドに取り除きたい既定の内容がある場合 (往々にして中央のフッタにはページ番号が入ります) には、空の引数を持つコマンドを入れるだけです。すなわち、

```
\cfoot{}
```

とします。

もっと美しくしてみましょう。節番号を「第 節」という言葉といっしょに (第 3 節のように) 左上に入れ、ページ番号を (4 ページのように) 右上に、自分の名前を左下に、付を右下に入れます。以下のコマンドをプリアンブルに入れます。

```
\lhead{第\thesection 節}
```

```
\chead{}
```

```
\rhead{\thepage ページ}
```

ウンタを取得するもので、これによって現在の節番号とページ番号を印字します。`\today` は、今日の日付を印字するものです。

ヘッダ下またはフッタ上に描かれる罫線の太さも変更することができます。もし必要なければ、太さを 0 にしてください。ヘッダ罫線は、既定では 0.4pt の太さであり、フッタ罫線は 0pt です。太さを変更するには、`\renewcommand{\headrulewidth}{0.4pt}`、`\renewcommand{\footrulewidth}{0.4pt}` のようなコマンドを使用してください。

`\thispagestyle{empty}`・`\thispagestyle{plain}`・`\thispagestyle{fancy}` のようなコマンドを使えば、各ページごとにヘッダ・フッタ設定を有効にしたり無効にすることができます。設定を変更したいページに、これらのコマンドを \TeX コードで挿入してください。実は、タイトルページは、その後のページは大域的に装飾的 (fancy) に設定されていれば装飾的 (fancy) になるにもかかわらず、既定でプレーン (plain) なるように設定されているのです。

奇数ページの左上だけに何かを挿入するなど、もっと複雑なコマンドもあるので、詳しい解説は fancyhdr パッケージの取扱説明書をご参照ください (fancyhdr.dvips の `README` を見てください)。

最後の例として、ヘッダやフッタに画像を挿入できることを取り上げましょう。ロゴをページ左上に入れたいものとします。たとえば、

```
\lhead{\resizebox{1in}{!}{\includegraphics{logo.eps}}}
```

とすれば、これを実現できます (文書中に他に画像を挿入していない場合には、`\usepackage{graphics}` をプリアンプルに宣言する必要があります)。

3.4 箇条書き (記号) ブリットの指定

ALLAN RAE 文

3.4.1 はじめに

\LaTeX には、ブリットの形が 216 個準備されていて、簡単なダイアログから呼び出すことができます。このダイアログを使えば、箇条書き (記号) の各階層で使うブリットを変更することができます。この設定は文書全体に共通なので、段落ごとに別のブリットを指定することはできません^{*5}。

3.4 箇条書き（記号）ブリットの指定

3.4.2 インタフェース

メニュー項目の文書▷設定を選択してダイアログを開き、ブリットタブを選択してください。

そのダイアログに、ブリットの形の一覧表があります。フォントと書いてあるドロップダウンメニューで、6つの各ブリット形パネルを切り替えることができます。また、一表左の階層パネルは、変更したい階層を選択するものです。任意のブリットでは、指定したい \LaTeX 表現を入力することができます。任意のブリットを指定した場合には、そのコマンドが必要とするパッケージを \LaTeX プリアンブルで指定する必要があります。

6つのパネルは、それぞれが必要とするパッケージによって分類されています。下記はフォント名と \LaTeX パッケージの対応表です。

フォント	必要とするパッケージ
標準	\LaTeX 本体
数式	amssymb.sty
絵文字 1	pifont.sty
絵文字 2	pifont.sty
絵文字 3	pifont.sty
絵文字 4	pifont.sty

\LaTeX は、ユーザーが導入されていないパッケージのブリットを使用したとしても、警告を発しません。ファイルを閲覧したり印刷したりしようとしたときに \LaTeX エラーが出るようならば、パッケージが導入されていない可能性が大です*⁶。

3.4.3 使用法

変更したいブリット階層を選んで、ブリットの形と大きさを選んでください。何らかの変更を加えたとしても、 \LaTeX 上には反映されませんが、文書の表示を行えば確認されます*⁷。

*⁶ \LaTeX が、パッケージがなくてもブリットを使用することを許すのは、編集をここで行ったとしても、のどこへエクスポートする可能性があるからです。

ここで使っている簡条書きでは、第 1 階層のプリットを変更するために、以下の \LaTeX コー
されています。

```
\let\savelabelitemi=\labelitemi
\renewcommand\labelitemi[0]{\small\(\sharp\)}
```

＃ 簡条書き（記号）の階層は、`\labelitem` コマンドの一部にローマ数字で指定されていること
てください。

第 4 章

LyX とインターネット網

LyX has long supported the export of documents to various web-friendly formats such as HTML. Before version 2.0, however, HTML export was always accomplished through the use of external converters.^{*1} These fall into two large groups: there are converters that use L^AT_EX as an intermediate format, such as `htlatex`, `html2latex`, and `plastex`, relying upon LyX to produce the L^AT_EX; and there is a converter that works directly on LyX files, `eLyXer`. All of these have different advantages and disadvantages.

The L^AT_EX-based converters have the advantage that, in principle, they know everything about the exported document that L^AT_EX does. Such converters do not care, for example, if a certain block of code was produced by LyX or was Evil Red Text. These converters know about `aux` files, counters, and references, and can often make use of the `bb1` files generated by BibT_EX. On the other hand, L^AT_EX is a very hard language to parse—it is sometimes said that only T_EX itself understands T_EX—and L^AT_EX-based converters will often choke on what seem to be quite simple constructions. Complex ones can throw them completely, and as of this writing, for example, the *Embedded Objects* manual will not export with `htlatex`, though the others will. The other issue concerns how math is handled. These converters typically convert the formulae into little pictures that are then linked from the HTML document. Since they are actually generated by L^AT_EX, they are accurate. But they do not scale well, and just getting them to look as if they are actually meant to be in your document—

Alex Fernandez's **eLyXer** (<http://elyxer.nongnu.org/>) solves some of these problems. In particular, the formulae it produces scale perfectly, since it renders using a combination of HTML and CSS rather than converting the formulae to images. But even moderately complex formula are rendered less well than with **htlatex**; this reflects the limitations of HTML.*² More importantly (as of version 1.0.1 at least), **eLyXer** has limited support for math macros and no support for user-defined paragraph or character styles. These limitations make **eLyXer** unsuitable for rendering the documents LyX users produce. In principle, of course, these problems could be solved, but the LyX developers have decided to follow a different path and have made LyX itself capable of writing XHTML, just as it is capable of writing DocBook, and plaintext.

As of this writing, XHTML output remains under development and should probably be regarded as “experimental”.*³ Still, the developers have chosen this approach because it has several potential advantages over the other two.

These advantages are primarily due to the fact that the XHTML output routines, since they are part of LyX, know everything LyX knows about the document being exported.*⁴ So they know about the table of contents (as displayed in the output), about the counters associated with different paragraph styles, and about user-defined styles. The XHTML output routines know what LyX knows about internationalization, too, so they will output “Chapter 1” or “Kapitel 1”, depending upon the language in effect at the time.

Quite generally, the output routines know what LyX knows about document layout, that is, about how the document is to be rendered on screen. We use this information when we output the document as XHTML. In particular, LyX *automatically* generates CSS style information corresponding to the layout information it uses to render the document on screen: if section headings are supposed to be sans-serif and bold when seen in LyX, then (by default) they will be sans-serif and bold when viewed in a web browser, too. And this is true not just for pre-defined styles, like Section, but for any style, including user-defined styles. Indeed, the XHTML output routines

*² That said, **eLyXer** can also use **isMath** and **MathJax** for equations, but this setting

4.1 Math Output in XHTML

no distinction between user-defined paragraph and character styles and LyX's own pre-defined styles: in each case, everything LyX knows about the styles is contained in the layout files. And much the same is true as regards pre-defined textual insets such as footnotes, and various custom insets.

The result is that XHTML output can be customized and extended in exactly the same way L^AT_EX output can be customized and extended: through layout files and modules. See chapter five of the *Customization* manual for the details.

The remainder of this chapter contains more detailed information on XHTML output, its limitations, and ways to work around those limitations.

4.1 Math Output in XHTML

LyX offers four choices for how math is rendered. These have various advantages and disadvantages:

- MathML

MathML is a dialect of XML designed specifically for mathematics on the web, and it typically renders very well in browsers that support it. The disadvantage is that not all browsers support MathML, and support is not complete even in the Gecko-based browsers, such as Firefox.

If LyX is unable to render a formula as MathML—for example, if the formula uses the `xymatrix` package or ERT—then it will instead output the formula as an image.

- HTML

As mentioned above, `elyxer` outputs math as HTML, styled by CSS.*⁵ For simple formulae, this can work quite well, though with more complicated formulae it tends to break down. Still, this method has the advantage that it is very widely supported and so it may be appropriate for documents that contain only a little, fairly simple math.

If LyX is unable to render a formula as HTML—for example, if the formula uses the `xymatrix` package or ERT—then it will instead output the formula

in fact, that are used for instant preview.*⁶ The advantage to this method is that the images are simply generated by L^AT_EX, so they are very accurate. The disadvantage, as mentioned earlier, is that these are bitmapped images, so they do not scale terribly well, and one cannot copy them, etc. (As of this writing, there are also problems about equation numbering. See [bug 6894](#).)

The size of the images can be controlled by setting the “Math Images Size” parameter under Document▷Settings▷Output.*⁷

If LyX for some reason fails to create an image for a formula (e.g., if a required L^AT_EX package is not installed), then it will fall back to outputting the formula as L^AT_EX.

- L^AT_EX

Finally, LyX will happily output math as L^AT_EX. As well as being the method of last resort, this method can be used with such tools as [jsMath](#), which uses JavaScript to render L^AT_EX embedded in HTML documents. LyX wraps L^AT_EX in either a `span` (for inline formulas) or `div` (for displayed formulas) with `class='math'`, as is required for jsMath.

One of these output methods must be selected under Document▷Settings▷Output. By default, LyX outputs MathML. This is a document-wide setting, therefore

Eventually, LyX will offer the user the option to select an alternate output method for a particular inset, say, one that isn’t being rendered very well by MathML.

4.2 Bibliography and Citations

XHTML output fully supports bibliographies and citations.

Citation labels are generated by the same machinery that generates LyX’s on-screen labels, so the labels will look in the output much as they do in LyX, though they will not be clickable. If you are using numerical citations, then LyX will output numerical labels such as [1] or [17], rather than simply showing the citation key in square brackets as Bib_TE_X does on-screen. If you are using author-year citations, then LyX will add lowercase letters to the years, just as Bib_TE_X does, if it finds more than one citation for a

4.3 Indexes

author-year combination. The labels will be printed with the bibliography entries. Note that there is, at present, no way to customize the appearance of the labels, for example, to choose between square brackets and parentheses.

Bibliography output is handled by the same machinery that handles the presentation of reference information in the citation dialog, so you will see in the XHTML output pretty much what you would see if you were to look at a given entry in the citation dialog. The formatting can be customized in your layout file or, preferably, in a module. See the *Customization* manual for the details.

The main defect at present is that cross-referenced information is printed with every entry with which it is associated. So you can see things like this:

Jason Stanley, “Context and Logical Form”, in *Language in Context: Selected Essays* (Oxford: Oxford University Press, 2007), pp. 30–68.

Jason Stanley, “Semantics in Context”, in *Language in Context: Selected Essays* (Oxford: Oxford University Press, 2007), pp. 201–30.

This should be fixed before long.

There is no support at present for sectioned bibliographies. If you have multiple bibliographies, then `LyX` will print the same bibliography over and over.

4.3 Indexes

`LyX` will happily export indexes as XHTML, but with certain limitations at the moment.

Index export will be most reliable when you do not attempt to use the fancy constructs that are described in the section on indexes in the *User’s Guide*.^{*9} We describe how they are handled using the subsection headings from that section.

- Grouping Index Entries (aka, sub-entries): `LyX` makes an effort to support these, but the entries must be separated by “! ”, that is, there must be space around the exclamation point. This is because it is otherwise too difficult to check for escaped exclamation points, ones in math, and so forth.

end.

- Cross-referencing: There is no support for cross-referencing. If L^AT_EX finds an entry containing the “|see...” construct, it is just dropped, and the entry is treated as an ordinary index entry.
- Index Entry Order: L^AT_EX does support attempts to fix the sorting order. It takes what is before the first ‘@’ it finds and use that for sorting, taking what follows the first ‘@’ to be the actual entry. At present, L^AT_EX does not check for escaped ‘@’, so do not try to index email addresses.
- Index Entry Layout: You can format entries by using the text style dia... by using any other method available within L^AT_EX itself. There is no support for constructs like: “entry|textbf”. Indeed, if L^AT_EX finds a pipe symbol, ‘|’, in an entry, it will delete it and everything that follows it.
- Multiple Indexes: There is no support for multiple indexes. Rather, all index entries will be printed as one large index. To avoid our printing several volumes of the index, we print only the main index, so make sure you have one.

4.4 Nomenclature and Glossary

There is at present no support for glossaries. Adding it would be fairly trivial and welcome.

第 5 章

LyX サーバー

5.1 はじめに

「LyX サーバー」を使用すると、他のプログラムが、LyX と通信したり、LyX コマンドを実行させたり、LyX の内部状態に関する情報を取得したりすることができるようになります。これは熟練したユーザー向けの機能ですが、便利ははずです。たとえば、LyX サーバーに書き込むことによって、JabRef のような参考文献管理ソフトが、LyX に文引用を「押し込む」ことが可能となります。

ただし、現在のところ、LyX サーバーは *Windows* では機能しないことに注意してください^{*1}。

5.2 LyX サーバーを開始するには

LyX サーバーは、名前をつけたパイプの対を利用することで機能します。パイプは通常 `UserDir` の下に作られ (Windows を除く。Windows では、名前を付けたローカルのパイプは `\\.\pipe` に特別なオブジェクトとして置かれます) 「`.lyxpipe.in`」および「`.lyxpipe.out`」という名前がついています。外部プログラムは、`.lyxpipe.` に書き込みを行い、戻りデータを `.lyxpipe.out` から読み出します。パイプ名の語は、ツール▷設定から、たとえば「`/home/myhome/lyxpipe`」あるいは Windows で「`\\.\pipe\lyxpipe`」などに設定することができます。サーバーを開始するためにはこれを手動で設定しなくてはなりません。

サーバーを開始しません。POSIX (Unix 型) システムでは、なんらかの理由で、終了する際に使用されていない「古い」パイプが削除されずに残ったりすると、L_YX を削除しようとして、何らかの理由でこれが失敗した場合、手動でパイプを削除のちに、L_YX を再起動する必要があります。Windows では、プログラムが終了しラッシュしたりすると、OS がパイプを削除するので、「古い」パイプが残ることはありません。

サーバーを使用する L_YX プロセスを複数起動するには、それぞれのプロセスにユーザーディレクトリを使用させたり、別々の preferences ファイルを使用させる別々の設定を使用する必要があります。

クライアントプログラムを開発しようとしているのであれば、L_YX サーバのデバッグ情報を有効にすると便利かもしれません。そのためには、L_YX を `lyxserver` として起動してください。

C 言語で書かれた完全なサンプルクライアントが、ソース頒布版の `development/` にあります。

もう一つの便利なツールとして、`src/client/lyxclient` にあるコマンドラインのクライアントがあります。

5.3 通常の通信

L_YX コールを発行するには、クライアントは ASCII テキスト行を入力パイプにみまなくてはなりません。この行は、以下の書式を持ちます。

LYXCMD:クライアント名:関数:引数

クライアント名 は、クライアントが自由に選べる名前です。これが用いられるクライアントが他の要求クライアントから結果を見分けることができるように返答をする際、クライアント名をエコーするときだけです。

関数 は、L_YX に実行させたい関数です。これは、ミニバッファで使用するコマンドです。

引数 は、一部の関数のみに効力を持つオプションの引数です (たとえば、「自己 LFUN は、現在のカーソル位置に引数を文章として挿入します」)。

5.4 通知

の書式をしています。ここでクライアント名と関数は、コマンド要求からそのままエコーされますが、データはコマンド実行の結果に依存した、多少なりとも有用な情報です。「font-state」のような一部のコマンドは、L_YX の内部状態に関する情報を返しますが、のデータ応答を返すものもあります。空のデータ返答は、コマンド実行が無事終わったことを意味します。

エラーが発生した場合の L_YX の返答は

ERROR:クライアント名:関数:エラーメッセージ

の書式をしており、エラーメッセージには、なぜコマンドが失敗したかの説明が入っているはずです。

(例)

```
echo "LYXCMD:test:beginning-of-buffer:" > ~/.lyxpipe.in
echo "LYXCMD:test:get-xy:" > ~/.lyxpipe.in
read a < ~/.lyxpipe.out
echo $a
```

5.4 通知

L_YX は、発生しているイベントを、非同期的にクライアントに通知することが可能です。いまのところ、ユーザーがあるキー列を関数「notify」に割り当てているときのみに L_YX はこの通知を行います。L_YX が送る文字列の書式は

NOTIFY:キー列

となっていて、キー列は、ユーザーが実際に打鍵したキー列の印字表現になります。

この機構は、L_YX のコマンド群を拡張してマクロを実装するのに使用することができます。なんらかのキー列を「notify」に割り当ててください。それから出力パイプで待つクライアントを起動し、キー列に応じたコマンドを読み込み、一つないし複数の L_YX コマンドを発行する L_YX 要求と L_YX コールを使用する関数を開始するのです。

5.5 単純な L_YX サーバプロトコル

LYXSRV:クライアント名:プロトコルメッセージ

の書式をしており、プロトコルメッセージは「hello」ないし「bye」です。クライアントから「hello」を受け取ると、L_YX はメッセージを聴いていることをクライアントのために返答をし、L_YX が終了することをクライアントに伝えるのには、L_YX が送信します。

5.6 DVI と PDF の遡及検索

DVI/PDF ビューア^{*2}には、遡及検索 (reverse search ないし inverse search とします) 機能を提供しているものがあります。これを使えば、DVI/PDF 出力の特定をクリックすることによって、L_YX の文書中で対応する行にカーソルを移動させることができます。これを実現するためには、ビューアは L_YX と情報交換を行う必要があります。これは、命名パイプ (*lyxpipe*) を使用するか、L_YX が作業ディレクトリに生 UNIX ドメインソケット (*lyxsocket*) を使用する (これは *lyxclient* プログラムと通信するのに使用している方法です) ことで、L_YX サーバを経由して実現することができます。場合によっては、ビューアと L_YX を橋渡しする補助スクリプトを必要とすることがあるかもしれませんが、それ以外は、ビューアは L_YX と直接通信することができます。これは、選んだビューアと基本ソフトに依存します。また、必要とされるビューア設定法と遡及検索が実際に行われる方法とも依存します。したがって、下記ではビューア毎の遡及検索の設定の仕方を説明することとします。その前に、DVI/PDF で遡及検索を有効にするためには、一般に何が必要とされるかを説明します。

5.6.1 自動有効化

ほとんどの場合、文書▷出力▷出力と同期を有効にすれば、L_YX がすべて自動的に行ってくれます。その場合、L_YX は自動的に、PDF の場合は SyncT_EX マクロの場合には *srcltx* パッケージを挿入します。このオプションは、表示/更新ツールバ簡単にアクセスすることができます。

srcltx パッケージや *src-specials* を読み込むと、時折、組版に影響を与えますので、仕上がりの組版においてはこれらを無効にしてください。

また、設定で自製マクロを指定することもできるので、何か特別な調整が必要な

5.6 DVI と PDF の遡及検索

5.6.2 手動有効化

L^AT_EX は、遡及検索のために複数の方法を提供しています。ある方法 `latex/pdflatex` プログラムに組み込まれており、他の方法は外部パッケージとして提供されています。どちらを選択するかは、お使いの L^AT_EX 頒布版が既に特定の法を提供しているか（組み込みによる方法は比較的新しいものです）と、お使いのビューアがそれに対応しているかに依存します。使用可能な方法を以下に述べます。

src-special 経由の DVI 組み込み検索（DVI のみ）

この方法は、DVI ファイルに遡及検索に必要な情報を追加するものです。これは、以前の L^AT_EX から利用可能となっており（比較的最近の L^AT_EX 頒布版はすべて対応しています）、安定して動作します。これを有効にするには、設定▷ファイル処理▷変換子 LaTeX (plain)->DVI ないし LaTeX (plain)->DraftDVI 変換子を `latex -src-special $i` に変更します。もしこれが動作しないときは、お使いの L^AT_EX エンジンが違うオプションを必要としていないか確認してください（頒布版によっては文法の異なることがあります）。

外部パッケージ（PDFSync および srcltx）

`pdfsync` および `srcltx` パッケージは、PDF 出力（`pdflatex` 経由）および DVI 出力で遡及検索を提供します。これを有効にするには、L_YX プリアンブルで以下のようにパッケージを読み込んでください。

- PDF の遡及検索には `\usepackage{pdfsync}`
- DVI の遡及検索には `\usepackage[active]{srcltx}`

と書きます。DVI と PDF の両方の遡及検索を実行できるようにするためには、プリアンブルに以下の行を挿入する方法があります。

```
\usepackage{ifpdf}
\ifpdf
  \usepackage{pdfsync}
\else
```

このようにすれば、DVI でも PDF (pdf_latex) でも正しいパッケージを使いながら、ドキュメントを閲覧することができます。

なお、PDFSync は、文書の出力レイアウトに干渉する可能性があるので注意してください。したがって、仕上がり文書では PDFSync を無効にすることをお勧めします。

SyncT_EX を使った組み込み遡及検索 (DVI および PDF)

(pdf)latex の最近のバージョンは、PDF および DVI 両方の遡及検索を組み込みサポートしています。このいわゆる *SyncT_EX* 機能は、主に、PDFSync パッケージと pdf_latex プログラムへ統合したことと、それを src-specials アプローチへ取り込んだ成果です。これを使用するためには、少なくとも T_EXLive 2008 か最近の MikT_EX 版が必要です。また、既に SyncT_EX サポートを提供している PDF ビューアは少ありません (Mac 上の Skim と Windows 上の SumatraPDF)。

DVI 出力で SyncT_EX を有効にするには、設定▷ファイル処理▷変換子の LaT_EX (plain)→DVI 変換子を latex -synctex=1 \$i に変更してください。PDF 出力用には、LaT_EX (pdf_latex)→PDF (pdf_latex) 変換子を pdf_latex -synctex=1 \$i に変更してください。また、ビューアを SyncT_EX で使うのに設定を行う必要かどうか、お使いのビューアの取扱説明書をご確認ください^{*3}。

5.6.3 特定のビューアを設定して使うには

Xdvi (全プラットフォーム)

L_YX は、lyxclient プログラムを自動的に使うのに必要なフックを、既に提供しているので、xdvi を使用する場合には、DVI 遡及検索を実行するのに他にすべきことはありません。使用中の文書を上記で述べたように設定するだけです (遡及検索は、Linux 上では Ctrl-Click、Mac OS X 上では Alt-クリックで始動します)。

しかし、何らかの理由で、L_YX と通信するのにソケットではなく命名パイプを使う場合には、設定▷ファイル処理▷ファイル書式の DVI ビューアを xdvi -e 'lyxeditor.sh %f %l' (lyxeditor.sh は適切なスクリプト名) に変更してください。たとえば、最低限のシェルスクリプトは

```
#!/bin/sh
```

```
LYXPipe="/path/to/lyxpip"
```

5.6 DVI と PDF の遡及検索

```
echo "$COMMAND" > "${LYXPIPE}".in || exit
read < "${LYXPIPE}".out || exit
```

のようになります。ここで/path/to/lyxpipe は、設定▷パスで指定した LyXServer
タイプのパスを指定してください^{*4}。

MacDviX (Mac OS X)

/Applications/MacDviX_Folder/calleditor.script の終わりに

```
/Applications/LyX.app/Contents/MacOS/lyxeditor "$2" $1
exit 1
```

という行を付け加えてください。Applications フォルダ以外の場所に LyX を導入した
合には、それに従って上記の行を修正してください。

遡及検索は、Alt-クリック (OPTION-クリック) で始動します。

Skim (Mac OS X)

設定▷ファイル処理▷ファイル書式▷PDF (pdflatex) のビューア設定に open
Skim.app \$\$i と入力し、Skim▷設定▷Sync で LyX を選択してください。

遡及検索は、COMMAND-SHIFT-クリックで始動します。

Okular (KDE)

設定▷Okular を設定...▷編集で「カスタムテキストエディタ」を選択し、コマンド
lyxclient -g %f %l を加えてください。

遡及検索は、SHIFT-クリックで始動します。

YAP (Windows)

yap を起動し、View▷Options メニューを選択し、「Inverse DVI Search」タブを選択
します。「New...」ボタンをクリックして開くウィンドウの「Name:」フィールドに「Ly
Editor」(または自分の好きな名称)と入力してください。「...」というラベルのボタン
をクリックしてファイルダイアログを開き、バッチファイル lyxeditor.bat (以下参照)
あるディレクトリを見つけます。lyxeditor.bat を選択してプログラムの引数を %f

```
@echo off
echo LYXCMD:revdvi:server-goto-file-row:%1 %2> \\.\pipe\lyxp
type \\.\pipe\lyxpipe.out
```

LyX で指定した LyXServer パイプパスが、\\.\pipe\lyxpipe となっていること
するか、lyxeditor.bat ラッパの方を適切に修正してください。

yap では、遡及検索はダブルクリックで始動します。

SumatraPDF (Windows)

SumatraPDF を遡及検索で使用するためには、設定▷ファイル処理▷ファイル
PDF (pdflatex) のビューア設定で SumatraPDF -inverse-search “lyxedito
%f %1” (lyxeditor.bat は上記のラッパ) と入力してください。SumatraPDF
コマンド PATH がない場合、フルファイル名を使用してください。

遡及検索はダブルクリックで始動します。

YAP (Cygwin)

まずはじめに、yap が Windows 環境での既定の DVI ビューアになっているこ
認してから、yap を起動し、View▷Options メニューを選択し、「Inverse DVI Sear
ブを選択します。「New...」ボタンをクリックして開くウィンドウの「Name:」フ
ードに「LyX Editor」(または自分の好きな名称) と入力してください。「...」とい
ルのボタンをクリックしてファイルダイアログを開き、バッチファイル lyxedito
プログラム (これは、Cygwin では既定で LyX 実行ファイルとともに導入されま
あるディレクトリを見つけます。lyxeditor.exe を選択してプログラムの引数を
%1 と指定してください。これによって、LyX との通信に *lyxsocket* が使用されるよ
ります。何らかの理由で *lyxpipe* を使用したい場合には、-g オプションを削除して
設定で LyXServer パイプパスを指定することを忘れないでください。

yap では、遡及検索はダブルクリックで始動します。

SumatraPDF (Cygwin)

SumatraPDF を遡及検索で使用するためには、設定▷ファイル処理▷ファ
式▷PDF (pdflatex) のビューア設定で SumatraPDF.sh と入力してください。

5.7 前方検索

```
SumatraPDF -inverse-search "lyxeditor -g %f %l" $(basename $1)
```

SumatraPDF は、Windows ネイティブのアプリケーションであり、L^AT_EX の Cygwin が使用する POSIX パスを理解しないので、このラップスクリプトが必要となります。SumatraPDF.exe がお使いのコマンドパスにない場合は、上記のスクリプトで POSIX のフルパスを使用してください。-g は *lyxsocket* 経由の通信を有効にします。ここでも *lyxpipe* を使用したい場合には、-g オプションを削除して、L^AT_EX 設定で L^AT_EX Server パスパスを指定することを忘れないでください。

遡及検索はダブルクリックで始動します。

5.7 前方検索

前方検索を有効にするためには、生成された PDF/DVI 出力ファイルに、T_EX ソースに関する情報を追加する必要があります。これは、第 5.6.1 節に述べられている方法で行うことができます。

次に、設定▷出力▷一般で自分の使うビューアを利用するように、L^AT_EX を設定する必要があります。ただし、yap・xdvi・okular・Sumatra PDF など、一部のビューアしかこの操作は利用できません^{*5}。Evince や xpdf などの他のビューアでは、少なくとも、ファイルの特定頁を読み込み直したり、指し示したりすることができますが、これらの場合に *synctex* の外部コールを使用することができます。設定ダイアログに用意されている用をよく研究してください。設定中で置き換えられる値は以下の通りです。\$\$n は行番号指し、\$\$t は書き出された一時.tex ファイル、\$\$o は書き出された出力ファイルを指し示す（一時ディレクトリにどちらが存在するかに依存して dvi あるいは pdf）。

閲覧時には、一時ディレクトリに .dvi ファイルがあるか .pdf ファイルがあるか（特定文書を閲覧するのにどの書式を使用したか）によって、適当なプレビュー書式を L^AT_EX 自動的に選択します。

第 6 章

特別な文書クラス

6.1 A&A の論文

PETER SÜTTERLIN 文

6.1.1 はじめに

この節では、文書クラス `aa.cls` 第 5.01 版を使用して科学雑誌 *Astronomy and Astrophysics* (『天文学と宇宙物理学』: www.edpsciences.fr/aa/ <http://www.edpsciences.fr/aa/>) に投稿する論文を執筆するための L^AT_EX の使用法を説明します。このパッケージは、FT ไซต์

[ftp://ftp.edpsciences.org/pub/aa/readme.html](http://ftp.edpsciences.org/pub/aa/readme.html)

からダウンロードすることができます。

このパッケージには取扱説明書がついていますが、本節は、本来の取扱説明書の代わりとなることを意図したものではなく、あなたの論文を正しい様式に整えるための小さな内書であるに過ぎません。

この学術誌の出版元は、2001 年 1 月 1 日以降、Springer 社から EDP Science 社に変更されていることにご注意ください。この出版社の変更によって、`thesaurus` コマンド廃止など、スタイルファイルに若干の変更が加わっています。L^AT_EX の `aa` クラスは、スタイルファイルの最新版 (第 5.01 版) をサポートしています。もし古い版が装備されて

6.1.2 最初の一步

L_AT_EX とともに頒布されている例示用のひな型を土台として、書き始めることをします。ひな型を使用しない場合には、以下の設定にしてください。

- 文書▷設定ダイアログで article (A&A) を選択してください (まあ、これはすね)。
- ページレイアウトオプションは変更しないで、既定値のままに設定しておいてください。全体のレイアウトはマクロが行うので、何も変更してはいけません。

6.1.3 ヘッダ部

最初に入力すべきものは、ヘッダ情報です。ヘッダ情報には、以下の 7 つの項目、いくつかは必須ではありません。

- タイトル : [必須]
- サブタイトル : [非必須]
- 著者 : [必須]
- 住所 : [必須]
- 抜き刷り : [非必須] 著者が複数いる場合、抜き刷りの送付を依頼するのに必要な人を記す。
- メール : [非必須] 連絡用の電子メールアドレス。
- 日付 : [必須] 推奨されるフォーマットは、Received: <日付>; Accepted: <日付>です。

\maketitle コマンドは、ヘッダが終了した場所に、L_AT_EX が自動的に入れてくれる。手動で挿入する必要はありません。各項目の順序は関係ありませんが、見た目のために実際の文書のレイアウトに合わせるために、上記の順序を維持しておくといいでしょう。

現住所の付記などのために、ヘッダ部に脚注を挿入したい場合には、挿入▷脚注として標準の脚注を挿入してください。この場合には、L_AT_EX は自動的に \thanks{} を挿入します。

これらのトピックの他に、このマクロは、L_AT_EX が対応していない、以下の 2 つの

6.1 A&A の論文

れる順に番号が付けられるので、各著者に番号マーカーを置く必要があります。

- `\email{アドレス}`は、至急連絡用に電子メールアドレスを付記します。

いずれの場合にも、 \LaTeX 中に対応したコマンドを入力し、 \LaTeX コードとしてマークしてください。`$\text{\LaTeX}Dir/examples` にある用例を参照してください。

6.1.4 概要

概要は、ヘッダ部の直後に置きます。第 5 版から、概要は環境からコマンドに変更され、一段落に限るという制限が加えられました。さらに、ここにはキーワードを含む項を入れなくてはなりません。これは、まだ \LaTeX では実装されていないため、 \LaTeX コマンド `\keywords{}` を手動で入力し、 \LaTeX コードとしてマークする必要があります。例にある論文を参照してください。

6.1.5 サポートされる環境

A&A 論文用レイアウトでは、文章を構造化するために、以下の環境がサポートされています。

- 標準
- 節
- 小節
- 小々節
- 箇条書き（記号）
- 箇条書き（連番）
- 箇条書き（記述）
- キャプション
- 概要
- 謝辞（Acknowledgment）
- 参考文献
- \LaTeX

きます。

- `\and`
- `\email`
- `\appendix`
- `\authorrunning`
- `\inst{}`
- `\keywords{}`
- `\object{}`
- `\titlerunning{}`

これらのコマンドのいずれかを使いたい場合には、自分で入力する必要があります。これを `LATEX` コマンドとしてマークすることを忘れないで下さい。

6.1.7 図フロートおよび表フロート

`LATEX` は、必要なフロート環境 `figure`・`figure*`・`table`・`table*`をサポートしています。ここでは詳しい説明はしません。ユーザーの手引きをご覧ください。ただ、表は左しなくてはならないことに注意して下さい。そうするためには、表を選択して、編落設定で配置を変更してください。

一つだけ、特別なものがあります。それは、図横にキャプションを配置する図での図を作成するには、以下のようにしなくてはなりません。

1. 挿入▷フロート▷図を使って、幅広の図フロートを作成します。それからフロートのラベルを右クリックして設定...を選択し、それで開かれるフロート設定ダイアログで段をまたぐを有効にして、OK ボタンを押します。
2. キャプション文を入力します。
3. キャプションの前にカーソルを移動し、Return を押して、カーソルをキャプションの上に移動します。
4. 図を挿入します。
5. カーソルを図の後ろに置いて、挿入▷整形▷水平方向の空白を選択して水平空白の設定ダイアログを開き、空白ドロップダウンから水平フィルを選択し、水平フィルを挿入します。

6.2 AAST_EX

8. カーソルをキャプション文の後ろに移動し、挿入▷T_EX コードを選択して、別T_EX 差込枠を挿入し、その中に閉じ括弧を入力します。

例示用論文中の図もご参照ください。

6.1.8 Referee レイアウト

投稿用には、論文を特別なダブルスペースレイアウトに整形しなくてはなりません。のためには、文書クラスに referee オプションを与える必要があります。これは、文書設定ダイアログの任意設定クラスオプションフィールドを使って行うことができます。の欄に referee と云う文字列を入力してください。

6.1.9 例示用論文

Examples ディレクトリには、L_YX で書かれた例示用論文が収録されています。これは元のマクロパッケージの例示用論文を L_YX 用に直したものです。これはヒントを得るために使用したり、元の L^AT_EX コードと L_YX で の書き方の比較に使ったりしてください。

6.2 AAST_EX

MIKE RESSLER 文

6.2.1 はじめに

AAST_EX は、American Astronomical Society (アメリカ天文学会) が発行する Astronomical Journal (宇宙物理学雑誌) 同誌の Letters と Supplement を含む)・Astronomical Journal (天文学雑誌)・Publications of the Astronomical Society of the Pacific (太平洋天文学会誌) 三誌への電子投稿を可能にするために、同学会が作成したマクロ集です。L_YX は、これらの文書を生成する上で、なかでも数式・引用・画像処理能力において、優れたツールであることが認識されています。L_YX は、同マクロ第 5.0 版以降を必要としており、できればここで説明されている第 5.2 版以降が望まれます。第 5.0 版未満のは、L^AT_EX 2.09 で使用するよう企図されているため、根本的に L_YX とは互換性がありません。AAST_EX パッケージは、以下の AAST_EX ウェブサイトからダウンロードでき

パッケージ中には、完備したユーザーの手引きが収録されていますので、 L_YX で書き始める前に、その中身を完全に理解しておく必要があります。 L_YX は、 $\text{AAST}_\text{E}\text{X}$ 全コマンドを理解する必然性をなくすものではなく、すべてを手入力する苦労を減くだけです。仕上がりの $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ 文書が、論文を投稿しようとしているジャーナルの要に完全に合致していることを確認するのは、あなたの責任です。

6.2.2 新しい論文を書き始める

論文を書き始めるには、 $\text{AAST}_\text{E}\text{X}$ ひな型ファイルを土台に始めることを強くお薦めします。ファイル▷新規(ひな型使用)をクリックして、新しいファイル名を入力し、型 `aastex.lyx` を選んでください。これには、論文原稿のほとんどに共通して使われるフィールドが書き込まれています。書き込まれている文字列(括弧<>を含む)を正しく報告で上書きしてください。 $\text{AAST}_\text{E}\text{X}$ コマンド・ $\text{AAST}_\text{E}\text{X}$ 環境の多くは、 L_YX 中で実行することができますが、特に`\altaffilmark` や `\altaffiltext` のようにのももあります。この二つのコマンドは、ひな型ファイルを開いたとき、赤色にされているのですぐ目に入ることでしょう。このようなコマンドは、直接 $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ コマンドを入力して、それを $\text{T}_\text{E}\text{X}$ コードとしてマークしなければなりません。このようなコードは、 $\text{T}_\text{E}\text{X}$ コードあるいは Evil Red Text (邪悪な赤い文)と呼ばれます。 $\text{AAST}_\text{E}\text{X}$ に必要とされる $\text{T}_\text{E}\text{X}$ コードを最小にするよう努力はしていますが、まだ望まれるはなっていないかもしれません。

6.2.3 論文を書き終わったら

論文を満足する水準で書き終え、プレビューや印刷が正しく為されているとして、だてに学術誌に投稿する前にしなくてはならない「後処理」が残されています。

1. 論文を $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ ファイルとして書き出します(ファイル▷書き出し▷ $\text{L}_\text{A}\text{T}_\text{E}\text{X}$)。
2. 出力した `.tex` ファイルを好きなエディタで編集します。
 - (a) `\documentclass` コマンドよりも前のコメント行を削除します。
 - (b) `\usepackage...{fontenc}` という行があれば(通常は `\documentclass` の後にあります)削除します。また、`\secnumdepth` 行があれば、これも削除します。

6.2 AAST_EX

3. 編集後のファイルを L^AT_EX にかけて、正しく処理されることを確認します。
4. 学術誌の要求事項を再読して、ファイル名と書式が正しいことを確認します。
5. 投稿してください。

6.2.4 特定のコマンドに関するコメント

AAST_EX ユーザーの手引き (aasguide.tex) に、各 AAST_EX コマンドの詳しい説明があるので、ここでは、各コマンドの使用法の詳細な説明は行いません。そこで、ここはユーザーの手引きに現れるとおりに一覧を列挙して、必要なところだけコメントを加えることにしましょう。

6.2.4.1 期待通りに動作するもの

これらは期待通りに動作するので、ここでは列挙するだけにして、コマンドの説明されていない節を挙げておきます。 \documentclass (2.1.1)・\begin{document} (2.2)・\title (2.3)・\author (2.3)・\affil (2.3)・\abstract (2.4)・\keywords (2.5)・\section (2.7)・\subsection (2.7)・\subsubsection (2.7)・\paragraph (2.7)・\facilities (2.10)・\begin{displaymath} (2.12)・\begin{equation} (2.12)・\begin{eqnarray} (2.12)・\begin{mathletters} (2.12)・\begin{thebibliography} (2.13.1)・\bibitem (2.13.2)・全引用コマンドとその派生 (2.13.2)・一般的な graphicx 図コマンド (2.14.1)・\begin{table} (2.15.4)・\begin{tabular} (2.15.4)・\caption (2.15.4)・\label (他にもあるが特に 2.15.4)・\tablerefs (2.15.5)・\tablecomments (2.15.5)・\url (2.17.4)・\end{document} (2.18)。

以下のスタイルオプションも正しく動作します。 longabstract (2.4)・preprint (3.2.1)・preprint2 (3.2.2)・eqsecnum (3.3)・flushrt (3.4) these are in the text box of the class options in the journal's style file.

6.2.4.2 動作するが、若干説明が必要なもの

以下のコマンドは動作しますが、若干の説明が必要です。

- 以下の項目は、学術誌の編集者向けに取ってあるものですが、どうしても必要ならば、L^AT_EX プリアンブルに入れることができます。 \received・\revised

- `\email` (2.3) 「単独で」使うことはできますが、段落の途中で使うことはできません。埋め込む必要がある場合には、`TEX` コードを使用してください。
- `\and` (2.3) 後ろに`{}`が付記されます。これはエラーの原因にはなりません。
- `\notetoeditor` (2.6) 「単独で」使うことはできますが、段落の途中で使うことはできません。埋め込む必要がある場合には、`TEX` コードを使用してください。
- `\placetable` (2.8) 相互参照用タグを挿入することはできません。手動で名前を入力する必要があります。
- `\placefigure` (2.8) —`\placetable` と同様。
- `\acknowledgements` (2.9) 後ろに`{}`が付記されます。これはエラーの原因にはなりません。
- `\appendix` (2.11) 後ろに`{}`が付記されます。これはエラーの原因にはなりません。
- `\figcaption` (2.14.2) 文章のはじめにカーソルを置いて挿入▷短縮タイプを選択すれば、非必須のファイル名引数を挿入することができます。「短縮タイプ」とありますが、`\figcaption` が必要とする型の非必須引数を挿入します。これはいつの日か改名されることでしょう。
- `\objectname` (2.17.1) 非必須パラメータとしてカタログ ID を取ることで `\figcaption` と同様。
- `\dataset` (2.17.1) 非必須パラメータとしてカタログ ID を取ることで `\figcaption` と同様。

6.2.4.3 実装されていないため、`TEX` コードを使用するもの

`\altaffilmark` (2.3)・`\altaffiltext` (2.3)・`\eqnum` (2.12)・`\setcounter{equation}` (2.12)・学術誌名略語 (2.12.4)・`\figurenum` (2.12.4)・`\epsscale` (2.14.1)・`\plotone` (2.14.1)・`\plottwo` (2.14.1)・`\tablenum` (2.15.4)・`\tableline` (2.15.4: 入れたい場所の後ろに、左側セルの最初の要素として使用する。表中では、`LyX` の方法は使わないでください)・`\tablenotemark` (2.15.4)・`\tablenotetext` (2.15.5)・補遺 (2.17) の大半 (`\objectname`・`\dataset`・`\email` を除く。上記参照)・`\singlespace` (3.1)・`\doublespace` (3.1)・`\onecolumn` (3.2)・`\twocolumn` (3.2)

6.2 AAST_EX

deluxetable を使う必要があるならば、テキストエディタで別ファイルとして編集し、入▷ファイル▷子文書を使用して、L_YX ファイルに読み込むことを推奨します。この例は、aas_sample.lyx ファイルをご参照ください。

6.2.5 FAQ とヒントと技とその他の考察

6.2.5.1 L_YX を AAST_EX と協調させる

L_YX に、新しいレイアウトと文書クラスを認識させるには若干コツが要ります。もうまく行かなかつたら、次のようにしてみてください。

1. L^AT_EX が、AAST_EX を検出できることを確認してください。AAST_EX 頒布版にまれる sample.tex (あるいは table.tex でも可) を、L^AT_EX や AAST_EX とまったく関係のないディレクトリにコピーして、sample.tex を L^AT_EX にかけてください。
2. aastex.layout が、L_YX の layouts フォルダにあることを確認してください。
3. L_YX 中からツール▷再初期設定を再度実行し、L_YX を再起動してください。
4. ひな型ではなく、通常の新規ファイルを開いてください。AAST_EX が、文書▷定のクラス一覧に表示されていますか？

既存の AAST_EX 文書进行处理する際に、AAST_EX レイアウトを見つけられないとか、「イトルレイアウトと通常のレイアウトを混在させないでください」といった警告が出るうであれば、正しい導入が行われていません。

6.2.5.2 表进行处理する際の L^AT_EX エラー

既定では、L_YX は、表のキャプションやタイトルを中央揃えにしようと試みます。これは AAST_EX と悪い相互干渉を発生させるようですので、キャプションやタイトルのどこをクリックして、編集▷段落設定を選択し、配置を両端揃えに設定してください。私ところではこれで解決しました。

6.2.5.3 参考文献

若干の注意があります。(1) 文献項目の本文中の空白取りが少しおかしくなることあるようです。書誌情報を入力する際には、最後の著者と、著作年を入れる括弧の

クスが現れます。このボックスをクリックして、残りの情報を入力してください。参考文献を追加するには、既存の参考文献の後ろでリターンを押します。すると、ボックスの付いた新規行が出てくるので、後は同様にしてください。

6.2.5.4 EPS ファイルの読み込み

AAST_EX には、自前の図関連コマンド（たとえば `\plotone`）がありますが、私定の `graphicx` を使った） \LaTeX の標準図コマンドの方がずっと好きです。望めば、ロートボックスに、 \TeX コードで `\plotone` 等のコマンドを挿入することはできません。私には正しいレイアウトを得ることができた試しがありません。標準の画像コマンド、たとえば、 \LaTeX は、`\usepackage{graphics}` コマンドを \LaTeX プリアンプルに挿入し、 \LaTeX 2_ε の標準的な方法で取り扱って、文章の中にちりばめます。現在、ApJ は、うどこのようにして作成した図を受け入れています、AJ はまだ「すべてを末尾に上げる」方法を使用しているかもしれません。

6.2.5.5 可能だったが、実装しなかったもの

「素敵な」ものの中には、実装することも可能だったけれども、あえて実装しなかったものがあります。たとえば、紙の原稿ではダブルスペースであったとしても、 \LaTeX ウィンドウ中で、文章をダブルスペースにする意味はないと考えました。また、`preprint` 式と `preprint2` 様式を、独立したレイアウトにすることもしませんでした。いずれも、執筆時間のほとんどを、平の原稿モードで使うのがほとんどですから、これらでディスクスペースを消費しないことにしました。

6.2.6 最後に

これまでに進んできた道のりは、長い人も短い人もいることでしょう。私には、と AJ に発表した論文がありますが、その努力の 98% は \LaTeX 上で為されました。2% が、 \LaTeX での後処理と若干の後片付けです。投稿の過程では何のトラブルもなかった。これらの学術誌が通常の投稿原稿と違う点に気づくことはなかったでしょう。です。がんばって発表してください！

6.3 AMS \LaTeX

6.3 AMS L^AT_EX

ができます。各学術誌毎の指示については、AMS の説明書をご参照ください（通常 L^AT_EX 出力中の一行を変更する必要があるだけです）。AMS の説明書は、ウェブ Web の <http://www.ams.org> か、FTP で <ftp://ftp.ams.org/pub/tex/amslatex/> から入手することができます。これらのレイアウトは、数学的な文章を書くのであれば、どんなときでも適切かつ有用であるはずです。

基本となる AMS L^AT_EX レイアウトには、以下の二つがあります。

- amsart : 標準の AMS article 書式。
- amsbook : 標準の AMS book（実際はモノグラフ）書式。

これらのレイアウトには、それ自身では、AMS クラスを使用するのに必要最低限のものしか含まれていません。特に、定理や補題などを設定するのに使用される「theorem」環境は、何一つ含まれていません。これらは、代わりに定理（AMS）モジュールに収録されており、AMS クラスのいずれかを選択すると、このモジュールが既定で読み込まれるようになっていきます（これは別のクラスでも使用することができ、別のものに変えたいときは取り除くこともできます）。少し使用頻度の低いものとしては、定理（AMS 拡張）モジュールがあり、これは手動で読み込む必要があります。

既定では、定理類は、文書を通して連番が振られますが、これは定理（節毎連番）モジュールを読み込めば変更することができ、book（AMS）を使用している場合には定理（章毎連番）を用いても変更することができます。これらを用いると、出力は $n.m$ のようになります、最初の数字は節番号（または章番号）、次の数字はそれまで同節内（章内）にわたった出力の全数を表します。大半の環境は、連番なしでも用いることができます。連番なしのものは、環境名の後ろにアスタリスクを付けたもので表されます。連番なしの出力が欲しい場合には、定理（連番なし）モジュールを用います。

これらのモジュールは、AMS クラスで使用する必要はないことに注意してください。定理（AMS）モジュールや上述の他のモジュールは、他のクラス、たとえば article・report・book（KOMA-script）などで問題なく使用することができます。

6.3.1 これらのレイアウトが提供するもの

これらのレイアウトが提供している環境を列挙すると、長い一覧となります。AMS L^AT_EX では、実のところ、「theorem」環境の派生型を際限なく定義していくことが可能

住所 これは著者の恒久的な住所です。

現在の住所 住所と別に指定されているときには、著者の投稿時の仮初めの住所で

電子メール 著者の電子メールアドレス

URL 著者のウェブアドレス。任意。

キーワード 論文中で論じられているトピックを識別するためのキーワードや
レース。

分野分類 *Mathematical Reviews* に説明が公表されている AMS Subject Classification
の分類番号を指します。上述した AMS のウェブサイトでも、オンラインで
とができます。

謝辞

献呈

翻訳者

以下の環境は、定理モジュールと定理 (AMS) モジュールの両方で提供され、後者
には、星付き (連番なし) 版と星なし (連番) 版の両方が提供されます。定理 (連番
モジュールでは、同環境の星付き版だけが提供されます。

定理 1. これは、通常、得られた主要な結果を叙述するのに使用されます。

系. これは直前の叙述から比較的直接に導き出せる結果を叙述するのに使います。
主要な結果であることがあります。

補題 2. 他の叙述を証明するために必要とされる、比較的小さな結果です。

命題 3. 議論されている一般的な理論に (希望的には) 何かを付け加えるような、
はない結果です。

推論 4. 正当化を与えずに述べられた叙述で、著者には証明法がわからないけれど
しいと (少なくとも著者にとっては) 思われるものです。

定義. これは自明でしょう。この環境のフォントは、前述の環境とは異なるフォン
用されます。

問題 5. これが何を指すかは必ずしも特定されていません。ご自分で解明してくだ

6.4 AGU 各誌 (aguplus)

ケース 1. 一般的に、これらはある条件の特定の場合を取り上げて、長い議論を区切るに用いられます。

ケース 2. ケースの連番の枠組みは、それ自身で独立していて、他の連番の叙述とは区別されます。

Proof. この環境の最後には、証了記号（通常は四角ですがスタイルによって変わります）が置かれます。この環境の中に別の環境を入れたい——たとえばケース環境など——場合で、証了記号は証明環境の終わりにのみ現れるようにしたい場合には、他の環境は証明環境の中に入れ子にしなくてはなりません。入れ子に関する情報については、ユーザー手引きの環境を入れ子にするの節をご覧ください。

そして、以下は定理（AMS 拡張）で提供されるものです。

基準. 必要とされる条件です。

アルゴリズム. 使用する一般的手順です。

公理. 議論されている系内部で真と仮定される性質や命題です。

条件. 議論の現在の文脈で仮定されている条件を述べるのに、よく用いられます。

注釈. 注記と同様です。

記法. 表記法の説明に用いられます。

要約 7. 本当に説明する必要がありますか？

謝辞. 謝辞（acknowledgement）です。

結論. 長い議論の終わりに時々用いられます。

事実 8. 命題と同様の用いられ方をしますが、だいたい長くないものに使われるようです。

さらに、AMS クラスは、AMS L^AT_EX パッケージと AMS フォントパッケージを自動的に提供します。これらの環境を使用するためには、お使いのシステム上で利用可能となっている必要があります。

6.4 AGU 各誌 (aguplus)

6.4.1 概要

これは American Geophysical Society (アメリカ地球物理学会) の各学術誌レイアウトファイルです。AGU 発行のクラスファイルおよび AGUplus の両方 (は[ftp://ftp.agu.org/journals/latex/journals](http://ftp.agu.org/journals/latex/journals)にあります) が導入されています。が必要です。

6.4.2 新しいスタイル

段落および段落*は、定義し直されています。LyX の画面では、依然として段落*と表示されますが、AGU クラスにおける L^AT_EX の対応物は、Subsubsubsection*です。

新しく定義されたスタイルには、左ヘッダ・右ヘッダ・受理日・改訂・採択日・C 文 ID・著者住所・廃棄用コメントがあります。これらは、ほとんど草稿用属性で、クラス説明書で説明されています。

おそらく、これはまだ恐ろしく不完全です。

6.4.3 新しいフロート

平面表 (Planotable) と挿絵 (Plate) が新しいフロートです。また、表キャプション付け加えられています。

6.4.4 サポートされている学術誌

- *Journal of Geophysical Research*: jrga.layout — Martin Vermeer

ここにあなたの作ったものを加えてください。難しくはありません。jrga.layout aguplus.inc を用例として見てください。

6.4.5 バグと注意すべきこと

新しいレイアウトを使うためには、新規文書の作成時に以下の手順を行う必要

6.5 Broadway

これはバグです)。

3. BibT_EX 差込枠の 2 番目のフィールドに `agu` と入力して、参考文献のスタイル `agu.bst` を使用していることを確認してください。標準スタイルはうまく動作しません。

6.5 Broadway

GARST REESE 文

6.5.1 はじめに

Broadway は演劇を書くためのものです。書式は Hollywood よりも装飾的で、それほど標準化されていません。この書式はワークショップに向いているでしょう。

6.5.2 特別な問題

Hollywood と同様です。

6.5.3 特別の機能

話者の名前をラベルとして挿入し、名前を挿入する際には、このラベルを相互参照してください。相互参照ダイアログには、登場人物の現在の配役が表示されます。

6.5.4 用紙寸法と余白

US レター寸法で、余白は左 1.6 インチ、右 0.75 インチ、上 0.5 インチ、下 0.75 インチです。

6.5.5 環境

以下の環境を使うことができます。右に記してあるキー割当ては、`broadway.bind` 使った場合のものです。

舞台設定と動作の説明に用います。最初に話者名をすべて大文字で書きます。

- 幕

自動的に連番が振られます。連番は、画面上ではアラビア数字ですが、出口ローマ数字です。
- 幕*

幕の副題です。単に中央揃えの文になります。
- 場面

自動的に連番は振られませんので、手動で番号を付けてください。これは私すればいいか分からなかったためです。
- 幕開きに :

ト書きの特殊な場合で、カーテンが上がる時の舞台設定と動作を説明します。
- 話者

話者（役者）の配役です。すべて大文字で中央揃えに置かれます。
- 括弧付き

話者への指示です。括弧は自動的に挿入されます。画面上には左括弧（のみ）されますが、印字出力には両方の括弧が表示されます。この環境は、対話ので使用されます。
- 対話

話者の言う内容。
- 幕切れ

幕が降ります。
- タイトル
- 著者
- 右寄せ住所

6.6 Dinbrief

文書クラス `dinbrief` は、ドイツの慣習に則った書簡の組版を行うのに用いられるような型ファイルが `.../lyx/share/templates` にあるので、それを土台として使用

6.7 EGS 各誌 (egs)

MARTIN VERMEER 文

6.7.1 概要

これは European Geophysical Society (欧州地球物理学会) の学術誌向けのレイアウトファイルです。必要とされる `egs.cls` は、EGS のウェブサイト <http://www.copernicus.org/> からダウンロードできます。

6.7.2 新しいスタイル

新しいスタイルとして、右寄せ住所・L^AT_EX タイトル・所属・学術誌名・原稿番号・頭著者・受理日・採択日・オフセットがあります。現在のレイアウトファイルは、残念ながらほとんどモジュール化されておらず、さまざまな `std*.inc` ファイルを読み込む形変えた方が便利になると思われます。

6.8 Elsevier 各誌

ROD PINNA 文

Elsevier Science Publishers B.V. は、同社出版の様々な学術誌に論文を投稿するための、標準的 L^AT_EX 文書クラス (`elsart.cls`) を提供しています。スタイルファイルは直接同社のウェブサイト <http://authors.elsevier.com/> からダウンロードすることができます。使用法の解説は、クラスファイルとともに提供されており、出版社の要求事項が詳細に述べられています。L^yX には、この文書クラスを使うための、レイアウトと型ファイルのパッケージが同梱されています。このクラスファイルの導入法は、他 L^AT_EX パッケージと同様であり、Elsevier の説明書に解説があります。

`elsart.cls` を使用するために、レイアウトファイル `elsart.layout` が提供されます。Elsevier のクラスファイルは、基本的に標準 `article` クラスに基づいているため、通常の機能はほとんど使用することができます。また、Elsevier クラスには、AMS 環境に類似の数式環境がたくさん定義されています。これらのコマンドは、すべて Elsevier

め、使用しないのが最上です。理想的には、Elsevier の説明書で言及されているもののパッケージは、使用しない方が良いでしょう。Elsevier は、提出されたファイルのまま使いながら、クラスファイルを、論文を投稿した特定の学術誌用のクラスファイル置き換えることを意図しているため、基本的にできるだけ「クリーンな」 \LaTeX フォイルを要求しています。これは同時に、文書の整形に時間をかけすぎるべきではない、ことを意味しています。論文が出版される時には、書式はいずれにしても変わってからです。このレイアウトのその他の使用法は、基本的に標準 `article` クラスと同様です。Elsevier が何を許容し、何を許容しないかに関する詳細は、Elsevier の説明書をご覧ください。

6.9 Foils (別名 FoilTeX)

ALLAN RAE 文

6.9.1 はじめに

本節では、プレゼンテーション用のスライドを作るための、 \LaTeX の使用法を解説します。スライドを作ることのできる文書クラスには、`slides`・`FoilTeX`・`seminar`・`beamer`・`powerdot` といったクラスがありますが、本節では、そのうち `slides` の解説を行います。誤解のないように、わかりやすく明確に、もう一度言っておきます。

本節は「`presentation (FoilTeX)`」クラスのみの解説を行います。

「`transparencies (slides)`」の解説をお探しであれば、第 6.21 節をご覧ください。このコンピュータに `foils` クラス (`presentation (FoilTeX)`) が導入されていない場合は、それほど良くはありませんが、おそらく `slides` クラスを使わざるを得ないでしょう。

`foils` クラスは、 \LaTeX 2_ε の一部として含まれている \LaTeX クラスファイル `foils.cls` の第 2.1 版とともに使われるように設計されています。

6.9.2 最初の一步

6.9 Foils (別名 FoilT_EX)

- 文書設定ダイアログ中の、両面文書オプションと二段組文書オプションは変更しないでください。これらは foils クラスでは無視されます。
- 既定のフォント寸法は 20pt であり、17pt・25pt・30pt を選択することも可能です。
- 既定のフォントは sans serif ですが、数式ではすべて、通常のローマン体が使用されます。
- FoilT_EX は、35 mm スライド用の特殊寸法の外、A4 とレター大の用紙寸法をサポートします。A5・B5・リーガル・エグゼクティブ大の用紙寸法はサポートしていません。
- フロートの配置の設定は無視されますので、わざわざ変更する必要はありません。フロートはすべて、本文中で定義された位置に表示されます。
- このクラスのページレイアウトの設定は、若干異なった挙動を示します。FoilT_EX では、ユーザ定義のロゴをはじめ、フッタとヘッダに広範な設定ができます。詳細については第 6.9.4.6 節をご覧ください。タイトルページは、文書中の他のページとは異なった取扱がなされ、ページ番号はつねに付けられず、(ロゴが定義されない場合) ロゴは、つねにページ下部中央に表示されます。使用できるページ様式とその挙動は、以下の通りです。

empty	最終出力には、ページ番号をはじめ、ヘッダ・フッタが出力されません (もちろん脚注は除きます) 。
plain	最終出力には、ページ番号がページ下部の中央に出力されますが、他のヘッダ・フッタは出力されません (脚注除く) 。
foilheadings	ページ番号が右下隅に出力されます。追記したヘッダ・フッタも出力されます。これが既定値です。
fancy	これは fancyheadings パッケージを使えるようにします。ただし FoilT _E X パッケージの作者は、ページレイアウトが崩れる可能性があるので、これを使用することは勧めないと言及しています。

6.9.2.1 任意設定クラスオプション

以下のオプションは文書設定ダイアログの任意設定クラスオプションボックスで使うことができます。

35mmSlide これは、ページレイアウトを縦 7.33 インチ横 11 インチに設定します。

dvips	これは、新規 foils 文書を作成する度に自動的に設定されます。このオプションは、横向きに設定されたページを回転させるのに、dvips ドを使うよう FoilT _E X に指示します。
landscape	このオプションは、用紙寸法を横向きページ大に変更するだけで、行いません。したがって、このオプションを使う場合は、各ページさせる外部プログラムを使用する、プリンタに用紙を横向きに給紙必要があります。また、このオプションは、Foilhead 環境と Rotatef 環境の役割を逆転させることに注意してください（これらの環境には次節で説明します）。
leqno	数式番号を左側に付けます。
fleqn	数式を左揃えにします。

6.9.3 サポートされる環境

他のクラスで標準的にサポートされている環境は、foils クラスでもサポートされます。それ以外に、FoilT_EX が提供している環境と、L_yX が付け加えた環境があります。下の環境は、他のクラスと共通の環境です。

- | | |
|------------------------|----------|
| • 標準 | • タイトル |
| • 箇条書き（記号） | • 著者 |
| • 箇条書き（連番） | • 日付 |
| • 箇条書き（記述） | • 概要 |
| • 箇条書き（リスト） | • 参考文献 |
| • L _y X コード | • 住所 |
| • 詩句 | • 右寄せ住所 |
| • 引用（字下げあり） | • キャプション |
| • 引用（字下げなし） | • コメント |

つまり、節区切りを除けば、主な環境のすべてです。一枚のスライドは、基本的タイトルと本文を備えた自己完結した節なので、FoilT_EX は新しいスライドを始めるようなコマンドを備えています。

6.9 Foils (別名 FoilT_EX)

- スライドタイトル (小) (ShortFoilhead)
- 横置きスライドタイトル (小) (ShortRotatefoilhead)

これらの違いは次節で説明します。

スライドは、アイデアを説明したり、新しい定理を説明したりするのにしばしば使われるので、FoilT_EX はそれらを説明するのに便利な環境を幅広く用意しています。

- | | |
|------|-------|
| • 定理 | • 定理* |
| • 補題 | • 補題* |
| • 系 | • 系* |
| • 命題 | • 命題* |
| • 定義 | • 定義* |
| • 証明 | |

星付き版は連番が振られず、星なし版は連番が振られます。他に、L_YX で追加された下の 2 つの箇条書き環境があります。

- 箇条書き (チック)
- 箇条書き (クロス)

FoilT_EX には、強力なヘッダ機能とフッタ機能があり、これらはプリアンブルで設定するのが最良ですが、文章中のどこでも設定することができます。もし文書中でこれらの設定を変更したい場合には、スライドの一番上、foilhead の直後で行うのが良いでしょう。

この目的のためには、以下のコマンドスタイルがあります (MARTIN VERMEER)。

- | | |
|-------------|--------|
| • グラフィックスロゴ | • 右ヘッダ |
| • 配布制限 | • 左ヘッダ |
| • 右フッタ | |

FoilT_EX が提供しているコマンドで、L_YX が直接サポートしていないコマンドもありますが、これらの説明と使い方については、第 6.9.5 節で行います。

6.9.4 スライドの組を作る

6.9.4.1 タイトルページを作る

タイトル・著者・日付・概要環境のある他のクラスとは異なり、foils は独立したタイトルを作成します。日付環境を使用しなかった場合には、 \LaTeX は（出力を直したときにはいつも）その時点の日付を挿入します。

6.9.4.2 新しいスライドを開始する

前述のように、新しいスライドを開始するには、4 つの方法があります。縦向きのスライドを開始するには、スライドタイトル (Foilhead) やスライドタイトル (小) (ShortFoilhead) を使用します。両環境の違いは、スライドタイトル (foilhead) とスライド本文の余白の違いです。

横向きのスライドは、横向きスライドタイトル (Rotatefoilhead) か横向きスライドタイトル (小) (ShortRotatefoilhead) 環境を使用して生成します。ここでも両環境は、タイトルと本文の間の余白のみです。どちらの場合も、(小) と記された版はスライドタイトルと本文の間の余白が、0.5 インチ (約 1.27cm) 小さくなります。

横向きのスライドをサポートする際、一つ問題となるのは、スライドを回転させること、PostScript 出力を生成するのに dvips ドライバを使用することを要求することです。もしスライドをプリンタに横に給紙することができれば、dvips ドライバを持っていなくても、横向きのスライドをつくることができます;-)

6.9.4.3 定理・補題・証明など

\LaTeX 中の小さなバグによって、これらの環境の同じ型をお互いに直接続けて置くことはできません。これらは何かによって分離されなくてはなりません。仮に二つ続けても、2 つの環境をマージしたかのように、前の環境が延長されるだけです。この問題を回避するにはどうすればよいのでしょうか。最も簡単な方法は、二つの環境の間に章を置くか、あるいは二つの間に「%」だけから成る \LaTeX 環境を置くことです。よって、 \LaTeX は 2 つの別々の環境を生成するように強制され、正しい \LaTeX 出力を生成することができます。 \LaTeX 頒布版に含まれる用例ファイルに例があります。この問題の同じ定理型環境を立て続けに置こうとした場合にのみ発生することに注意してください。

6.9 Foils (別名 FoilT_EX)

い。記号による箇条書きを使用したい場合には、上記第 3.4 節の箇条書き (記号) プリットの指定の節も、読まれることが望ましいでしょう。

箇条書き (チック) と箇条書き (クロス) の二つの新しい様式は、チェック印や × 印箇条書きのラベルに使用する専用の環境を提供することで、予定一覧や正誤一覧を作りやすくするように設計されたものです。これらの箇条書きは、箇条書き (記号) の派生型専用を用意したものに過ぎません。ただし、これらは、psnfss パッケージが導入済であることを要求します。

6.9.4.5 図と表

FoilT_EX では、図表フロートは、本文中の挿入箇所にこれらが正確に表示されるよう再定義されており、ページの上部やユーザー指定場所に押しのけたりはしないようになっています。たとえフロート配置の設定に変更を加えたとしても、単に無視されます。

6.9.4.6 ページヘッダおよびページフッタ

グラフィックスロゴと配布制限の二つは、左フッタ文を制御するものです。前者は、グラフィックスのロゴをスライドに入れるためのもので、既定値は「-Typeset by FoilT_EX」となっています。後者は、「機密事項」などのように、聴衆に秘密区分を提示すること意図しています。既定値では、何も設定されていません。

スライドの残り三隅には、右フッタ (既定ではページ番号) ・右ヘッダ (右上) ・左ヘッダ (左上) を使って文字を入れることができます。

6.9.5 サポートされない FoilT_EX 機能

以下で言及されているコマンドは、L^AT_EX 環境内で設定されるか、他の環境中の T_EX コードとして設定される必要があります。

6.9.5.1 長さ

長さは、すべて `\setlength{長さ名}{新規長さ}` コマンドで調節されます。ここで長さ名は、変更したい長さに付けられている名前置き換え、新規長さは、その長さの値で置き換えます。長さはすべて、インチ (in) ・ミリメートル (mm) ・ポイント (pt) のような長さ単位か、`\textwidth` のように文書比あるいはフォント基準の長さで指定する必

`\setlength{\foilheadskip}{-0.5in}`と書き込みます。

フロート周囲の余白は、以下の長さを設定することで調整することができます。

<code>\abovefloatskip</code>	本文とフロート上辺との間隔
<code>\abovecaptionskip</code>	フロートとキャプションとの間隔
<code>\belowcaptionskip</code>	キャプションとそれに続く本文との間隔
<code>\captionwidth</code>	この長さを調整することでキャプションを、周囲の本文くすることができます。 <code>\textwidth</code> に対する相対的な調整するのが最も良いでしょう。

以下のようなタイトルページ関連の長さもあり、タイトルが長い場合や、著者が複数場合に使うと便利かもしれません。

<code>\abovetitleskip</code>	ヘッダとタイトルとの間隔
<code>\titleauthorskip</code>	タイトル環境と著者環境との間隔
<code>\authorauthorskip</code>	著者が複数いる場合の著者行の間隔
<code>\authordateskip</code>	著者と日付との間隔
<code>\dateabstractskip</code>	日付と概要との間隔

最後に紹介する長さ関連コマンドは、箇条書き環境すべてに使用できるもの。`\zerolistvertdimens` を箇条書き環境の内側に置くと、項目間の垂直余白がすべて除かれます。これは長さではなくコマンドなので、上述のもののように`\setlength` する必要がないことに注意してください。

6.9.5.2 ヘッダとフッタ

`\LogoOn` および `\LogoOff` は、`MyLogo` で定義したロゴを特定のページに入れるかを制御します。`\LogoOff` をプリアンプルに入れると、全スライドからロゴが取れます。特定のスライドにのみロゴを付けたくない場合は、当該ページのスライドの直後に直接 `\LogoOff` を置き、次のスライドタイトルの直後に直接 `\LogoOn` をください。

文書レイアウトダイアログで fancy ページ様式を使用することを選択した場合、横置きページのヘッダとフッタが回転後も正しく配置されるように、プリアンプルに `\let\headwidth\textwidth` を加える必要があるでしょう。これは `fancyh`

6.10 Hollywood (Hollywood スペック・スクリプト)

GARST REESE 文

6.10.1 はじめに

脚本を正しい書式で書くことは、ハリウッドにおける「通過儀礼」です。ハリウッド脚本は、読み手が内容に集中できるようにし、俳優にとっても読みやすくかつ親しみやすいように設計されています。脚本の 1 ページは、映画の 1 分でなければならず、銀幕上見えたり聞こえたりしないものは、脚本に混入してはなりません。また、全体を通して Courier 12 pt フォントを使わなければならず、イタリック体は用いてはいけません。

6.10.2 特別な問題

話者の行は、文の途中で改行することは決してあってはなりません。もし話者行のが、次頁にまたがる場合には、話者名を再度書き、その後に (Cont'd) と加えます。

6.10.3 特別な機能

話者名はラベルとして挿入し、話者名を挿入するには、そのラベルを相互参照してください。相互参照ダイアログは、登場人物の現在の配役を表示することになります。これは、ト書きに話者名を挿入するのにも使用することができます。

6.10.4 用紙寸法と余白

US レター寸法で、左余白 1.6 インチ・右余白 0.75 インチ・上余白 0.5 インチ・下余白 0.75 インチです。

6.10.5 環境

以下の環境が利用可能です。右に示されているキー割当てを使用するには、hollywood.bind を使用してください。

- フェイドイン:
通常、この後には「Sally は目を覚ますと」のように続きます。
- 屋内:
新しい屋内 (INTERIOR) カメラの設定を指定します。この後には、か
DAY や NIGHT など指定して、必要な照明を定義します。この行はすべ
字で書きます。
- 屋外:
屋外 (EXTERIOR) カメラの設定を指定します。この行はすべて大文字
ます。
- 話者
話す役。
- 括弧付き
話者への指示。自動的に括弧 () が挿入されますが、L_YX 上に表示されるの
括弧 (のみです。印刷出力には右括弧も表示されます。
- 対話
話者が話す内容。
- 画面遷移
カメラの動きの指示。例 : CUT TO:。
- フェイドアウト:
- 著者
- タイトル
- 右寄せ住所

6.10.6 脚本で使われる特別な用語

- (O.S) — off screen
- (V.O) — voice over
- b.g. — background
- C.U. — close-up
- PAN — camera movement
- INSERT — cut to close up of

6.11 ijmpc および ijmpd

PANAYOTIS PAPASOTIRIOU 文

6.11.1 概観

ijpmc パッケージは、*International Journal of Modern Physics C* への電
投稿を可能にするためのマクロ集です。同様に、ijpmd パッケージは、*Intern
tional Journal of Modern Physics D* へ投稿する現行を作成するためのもの
です。両誌とも World Scientific 社の刊行です。対応する文書クラス名は、それぞ
ws-ijmpc.cls および ws-ijmpd.cls です。これらのファイルは、著者のための
針とともに、<http://www.worldscinet.com/ijmpc/mkt/guidelines.shtml>お
び<http://www.worldscinet.com/ijmpd/mkt/guidelines.shtml>の各サイトから
ダウンロードすることができます。両パッケージは、標準「article」パッケージの修正版
あり、両者は（一部を除き）ほぼ同一です。LyX は、これらのパッケージのほとんどの
能をサポートしています。私は両誌へ投稿する論文を LyX で書き上げましたが、何ら
問題も生じませんでした。

6.11.2 論文を書く

通常と同じく、論文を書く上で最も易しい方法は、ひな型を使って書き始めることです。
ファイル▷新規 (ひな形使用) をクリックして、ひな型 ijmpc.lyx もしくは ijmpd.l
を選択してください。すると、原稿で良く使用されるフィールドを含んだ、ほぼ空の文
ができますので、既存のフィールド（鍵括弧<>を含む）を正しい情報で置き換えてくだ
い。以下の各点にご留意ください。

1. 両パッケージでは、フォント寸法や文書のページ様式を変更することが禁じられ
るので、LyX 中でこれらを変更することはできません。
2. 文書言語は変更してはいけません。論文をプレビューする前に、babel パッケー
が使用されていないことを確認してください。これは、ツール▷設定をクリック
て、言語オプションタブを選択し、言語設定で Babel を使うチェックボックスを

ドを定義するのに使用することができます。この機能は、ijmpd パッケージでサポートされていないので、ご注意ください。

5. 「定義」「ステップ」「例」「注記」「記法」「定理」「証明」「系」「補題」「命題 (Prop)」「問題」「主張」「予想」といった新しい環境を使用することができます。これらの使い方は、大凡自明でしょう。L_YX は、これらの環境をすべてサポート。それぞれについて、適切なラベルと文字様式および連番スキームを使用します。
6. 両パッケージは、単純な引用を使用しますので、natbib パッケージは使用しません。L_YX 中では、引用参照は通常通り表示されますが、出力できない文字として表示されます。引用に通常の本文を使用したい場合には、「`\refcite{キー}`」のように `refcite` コマンドを使用してください。
7. 両パッケージには、「謝辞」環境がありません。謝辞を置くには、「節*」環境を使用してください。
8. 付録は、論文中、謝辞の後かつ参考文献の前であれば、入れることができます。L_YX 上では、「付録領域」という特別な環境があり、これを使って付録の開頭を指定することができます。この環境は空のままにしておかなくてはなりません。この環境は、L^AT_EX コマンドを書き出すだけで、実際に出力は行いません。L_YX 中では、青字で「Appendix」という単語が表示され、この箇所より後の節は付録であることを明示します。「付録領域」は、最初の付録の前になくても構いません。ことに注意してください。さもなくば、出力中において、付録はすべて通として連番が振られてしまいます。
9. ijmpc および jmpd パッケージは、表キャプションを実装するのに `tbl` コマンドを使用します。そのため、L_YX が作成した表は正しく印字されますが、そのキャプションは無視されます。しかし、T_EX コードを少し使えば、この問題を回避して、キャプションを望み通り印字することができるようになります。そうするには、表フロートを通常通り作成して、キャプションを削除し、代わりに T_EX で `\tbl{表キャプション}{(ママ)}` と入れます。そこに表の内容を入れて、T_EX コードで閉じてください。このからくりがどのように実装されているかは、ひな型ファイルに入っている表の例を精査してみてください。別として、表キャプションが欲しい場合に、表フロート全体を `.tex` ファイルに保存し、このファイルを L_YX 文書に取り込んでください (挿入▷ファイル▷子文書)。フロートの作り方についての詳細は、各パッケージに含まれている `ws-ijmp`

6.11.3 原稿を投稿するための準備

論文を投稿する前には、 L_X 文書を $\text{L}_\text{A}\text{T}_\text{E}_\text{X}$ ファイルに書き出し (ファイル▷書き出し $\text{L}_\text{A}\text{T}_\text{E}_\text{X}$)^{*1}、その `.tex` ファイルに以下の変更を加える必要があります。

1. `\documentclass` コマンドより前にあるコメント行を削除する。
2. 自身で明示的に $\text{L}_\text{A}\text{T}_\text{E}_\text{X}$ プリアンプルに置いたコマンドを除き、`\makeatletter` コマンドと `\makeatother` コマンドの間にあるものをすべて削除する。

こうして修正を加えた `.tex` ファイルを保存し、必要な回数 $\text{L}_\text{A}\text{T}_\text{E}_\text{X}$ で処理してください。それによって得られた `.dvi` 文書も確認された方がよろしいでしょう。

6.11.4 $\text{T}_\text{E}_\text{X}$ コードの使用

$\text{T}_\text{E}_\text{X}$ コードを使用するのは、2 つのコマンドに限られ、これらは文書の最初に置かれてはなりません。論文をひな型 `ijmpc.lyx` または `ijmpd.lyx` を使用して書き始め場合には、これら必要な $\text{T}_\text{E}_\text{X}$ コードは、すでに書き込まれています。通常、これらを除する必要はありません。2 つのうち 1 番めの $\text{T}_\text{E}_\text{X}$ コードのみが、奇数ページおよび偶数ページの上部に印字する情報 (それぞれ著者名と論文の短縮タイトル) を指定するために、修正する必要があります。この $\text{T}_\text{E}_\text{X}$ コードは、`\markboth{著者名}{論文の短縮タイトル}` という形である必要があります。

6.12 iopart

UWE STÖHR 文

6.12.1 概要

`iopart` パッケージは、Institute of Physics (イギリス物理学会) 発行の術誌へ電子的に投稿する原稿を作成するための文書クラスを提供しています。`iopart` クラスを使用した論文作成方法の著者用説明書は、ウェブサイト <http://ftp.iop.org/pub/journals/latex2e> から `iopart` パッケージとともに

6.12.2 論文を書く

論文を書くもっとも簡単な方法は、 $\text{L}_\text{Y}\text{X}$ の用例ファイルフォルダにある *IOP-art* を土台にすることです。このファイルを開き、新しい名前で保存して、書き始めてください。この用例ファイルは、特定の環境の使用法についても説明しています。もっともな助言として、以下の点にご注意ください。

- 文書が PDF・PS・DVI にコンパイルできるようにするためには、文書設定オプションで、二つの AMS math パッケージを使う関連オプションが有効でないことを確認してください！
- タイトル環境の選択如何によって、論文の種類が定義されます。種類毎に以下をタイトルに使用してください。
 - 論文用には `title`
 - Review 用には `review`
 - Topical review 用には `topical`
 - Comment 用には `comment`
 - Note 用には `note`
 - 論文用には `paper` (タイトルと同じ)
 - Preliminary communication 用には `prelim`
 - Rapid communication 用には `rapid`
 - 編集者への Letter には `letter`
- 書簡以外のタイトル環境には、すべて短縮タイトルをつけることもできます
- $\text{L}_\text{Y}\text{X}$ では直接にサポートされていませんが、一般的なタイトル環境としてがあります。執筆する文書が、他のタイトル型にそぐわない場合には、 $\text{T}_\text{E}\text{X}$ でこれを指定することができます。

特別な表や数式組版に関するヒントなど、詳しい情報については、IOP 著者用ガイドを参照してください。

6.13 Kluwer

Kluwer パッケージは、Kluwer Academic Publishers 社が発行する学術誌への電子投
を可能にするために、同社が作成したマクロ集です。同社の発行誌のうち、(少なくとも
の専門内で)もっともよく知られているものは、*Astrophysics and Space Science* (『宇
物理学と宇宙科学』)と *Solar Physics* (『太陽系物理学』)ですが、他にも多くの学術誌が
ります (<http://www.wkap.nl/jrnl1list.htm/JRNLHOME>に全覧があります)。Kluw
パッケージは、<http://www.wkap.nl/kaphtml.htm/STYLEFILES>からダウンロードで
ます。完全なユーザーの手引きも同パッケージ内に収録されています(個別にダウンロ
ドすることも可能です)。

L_YX は、このパッケージの機能の多くをサポートしていますが、すべてではありません。しかし、T_EX コードを必要とするのは、パッケージ中のいくつかの「変わった」コマンドだけになりました (6.13.4 参照)。最近、私は L_YX を使って書いた論文を *Astrophysics and Space Science* に投稿しましたが、何らの問題も発生しませんでした。

論文を書き始めるには、ひな型ファイル Kluwer を使うのが最も簡単です。ファイル新規 (ひな型使用) をクリックし、ひな型 `kluwer.lyx` を選択してください。これは、稿に通常必要となる各フィールドと、それらの使用法が短く書かれただけの、(ほとんど何も書かれていない文書です。他のテンプレートと同様に、既存の文章を (括弧<>をむ) 正しい情報で上書きしてください。

AAS_TE_X パッケージと同様、論文を投稿する前に、以下のような「後処理」を行う必要があります。

1. ファイル▷書き出し▷ LaTeX をクリックして、論文を LaTeX ファイルに書き出す。
2. 生成された `.tex` ファイルに、テキストエディタを使用して、以下のような変更を加える。

3. この .tex ファイルを必要な回数 (通常 3 回以内) だけ L^AT_EX にかける。
4. 生成された .dvi ファイルを xdvi などを使って開き、問題がないか確認する (いをしていない限り問題ないはずです)。

6.13.4 Kluwer パッケージの「風変わり」な特徴

Kluwer パッケージには、以下のような「風変わりな」特徴があります。

1. 複数の論文を 1 つの L^AT_EX ファイルへ入れることができます^{*2}。各論文は、`article` 環境で囲まなくてはなりません。残念ながら、1 つの論文だけを書く場合でもこの環境を省略することはできません。つまり、各論文は `\begin{article}` から始まり、`\end{article}` で終わります。この環境を L_YX に実装することも可能ですが、醜い上に初心者混乱させる可能性があるため、実装することはしませんでした。したがって、これらの命令 (`\begin{article}` と `\end{article}`) は直接本文中に入力して、L^AT_EX コード (「T_EX コード」のことです) としてマージする必要があります。
2. 論文の冒頭に置く情報 (タイトル・サブタイトル・著者名・所属・ヘッダ用著者名・ヘッダ用著者名・概要・キーワード) は、「opening」とよばれる環境で囲まなくてはなりません。これは L_YX では実装されていないので、タイトルやサブタイトルなどは、2 つの T_EX コード行 (`\begin{opening}` および `\end{opening}`) の間に入れる必要があります。
3. ユーザーの手引きによれば、参考文献の各項目のラベルは、`\protect\citeauthor{著者名}{年}` と書く必要があります。

ひな型 `kluwer.lyx` は、これらの風変わりな点がすべて考慮されています。このひな型を使用して、論文を新たに書きはじめる場合には、特別なことを行う必要はありません。以下の点にのみ留意してください。

1. ひな型にある T_EX コードを削除しないでください。
2. 新しい参考文献項目を入力する場合には、ひな型に例示されている参考文献をコピーして、必要に応じて修正してください。

6.14 Koma-Script

BERND RELLERMEYER 文

6.14.1 概要

LyX の文書クラス *article (koma-script)*・*report (koma-script)*・*book (koma-script)*・*letter (koma-script)* は、それぞれ Koma-Script ファミリーの L^AT_EX 文書クラス *scrartcl.cls*・*scrreprt.cls*・*scrbook.cls*・*scrletter.cls* に対応しています。これらは、標準文書クラス *article.cls*・*report.cls*・*book.cls*・*letter.cls* の代物であり、これらよりも多くの点でヨーロッパに於ける活版印刷の慣例に合致します。

- 標準文字寸法は、*article (koma-script)*・*report (koma-script)*・*book (koma-script)* で 11pt、*letter (koma-script)* で 12pt です。
- 見出しや箇条書き（記述）環境のラベル、および *letter (koma-script)* 文書クラスの要素の多くでは、ボールド体サンセリフフォントが使われます*³。章見出しの番号は、節見出しの連番と同様に行われ、別行立てで「Chapter...」という行がきません。さらに、多くのオプションによって、見出しの外観を修正することができます（LyX では、文書▷設定ダイアログのクラスオプションフィールドで指定します）。これらのオプションの独語での詳細な説明が、Koma-Script の取扱説明書 *scrguide* にあります。
- Koma-Script 文書クラスで、印字部分をデザインする主な方法は、BCOR オプションまたは DIV オプションを使う方法です（LyX では、文書▷設定ダイアログのクラスオプションフィールドで指定します）。これらのオプションは文書▷設定ダイアログのオプションと同様に、ページ余白を明示的に変更することができます。これらとその他印字領域オプションのドイツ語での詳細な説明が、Koma-Script 取扱説明書 *scrguide* にあります。
- Koma-Script ファミリーの L^AT_EX 文書クラスには、多くの追加コマンドが定義されています。それらのうち、LyX で意味を持つものは、対応する段落環境として装されています。

Koma-Script ファミリーの L^AT_EX 文書クラスに関する独語の詳細な説明が、Koma-Script 取扱説明書 *scrguide* にあります*4。以下の各節では、L_YX に関係する側面のみ説明します。

6.14.2 article (koma-script) ・ report (koma-script) ・ book (koma-script)

文書クラス *article (koma-script)* ・ *report (koma-script)* ・ *book (koma-script)* ぞれぞれレイアウトファイル `scrartcl.layout` ・ `scrreprt.layout` ・ `crbook.layout` によって実装されています。これらは、対応する標準文書クラス *article* ・ *report* ・ *book* の段落環境を、若干の修正を加えた上ですべて含んでいますが、L_YX 固有の箇条書き環境は例外であり、同じ機能を持つ新しいラベリング環境で置き換えられています。このラベリング環境以外にも、新しい段落環境が多く追加されています。これらは、*(koma-script)* には含まれていないのでご注意ください。

- 部 (addpart) ・ 章 (addchap) ・ 節 (addsec) : これらは、それぞれ部* ・ 章* ・ 節* の働きをしますが、加えて目次に項目が表示されるようになります。部 (addpart) と章 (addchap) は、*article (koma-script)* には収録されていません。
- 章 (addchap*) ・ 節 (addsec*) : これらは、章 (addchap) と節 (addsec) とまったく同等に動作しますが、加えてランニングヘッダを取り除きます。章 (addchap) は *article (koma-script)* には収録されていません*5。
- 小見出し (minisec) : これは文書構造に現れない形で、続く段落の直上に標準的な見出しを付けます。
- 上部キャプションと下部キャプションは、ある要素の上部ないし下部にキャプションを配置する際、位置に応じて正しく余白取りを行う特別なキャプションで版規則に厳密に従うならば、表キャプションは、つねに表の上に置くべきです (図 2.8a)。また、クラスオプション `tablecaptionsabove` を用いれば、キャプション指定すると、表には上部キャプションが、図には下部キャプションが適用されるようになります。これを使用するためには、Koma-Script 第 2.8q 版以上が必要です。
- 格言 (dictum) : これは、章頭などに警句を置くのに用いることができます。オプション引数 (挿入▷短縮タイトル) を用いれば、格言の作者を入れることができます。格言と作者の間は、線で区切られます。これを使用するためには、Koma-Script 第 2.8q 版以上が必要です。格言 (dictum) は、*article (koma-script)* ・ *report (koma-script)* ・ *book (koma-script)* には収録されていません。

以下の各環境は、標準のタイトル・著者・日付とともに文書のタイトル領域を形成します。これらは、最初の「通常の」段落よりも前に置かれる必要があります*⁶。これらの環境は2度以上使用された場合には、後に使用されたものが前のものを上書きします。つまり、最後に使用したもののみが有効になります。しかしながら、タイトルや著者、日付のよう異なる環境の順序を変更しても、生成された文書には何の効果も生じません。

- 主題：通常のタイトル部（タイトル・著者・日付）の上に、文書の主題用の中央揃えの段落を生成します。
- 出版社：通常のタイトル部（タイトル・著者・日付）の下に、出版社名用の中央揃えの段落を生成します。
- 献呈：*report (koma-script)* と *book (koma-script)* では、タイトル頁の次に、献呈用の中央揃えの段落を独立した頁として生成します。*article (koma-script)* の場合は、通常のタイトル部（タイトル・著者・日付）の下に献辞用の中央揃えの段落を生成します。
- タイトル頭書き：通常のタイトル部（タイトル・著者・日付）の上に、文書頭書き用の左揃えの段落を生成します。
- 扉裏上部：*report (koma-script)* や *book (koma-script)* の両面印刷において、扉裏上部に左揃え段落を生成します。片面印刷や、*article (koma-script)* では何の効果も生じません。
- 扉裏下部：*report (koma-script)* や *book (koma-script)* の両面印刷において、扉裏下部に左揃え段落を生成します。片面印刷や、*article (koma-script)* では何の効果も生じません。
- 追加タイトル：実際の文書の前に、整形を行わない段落から成る、特殊な「汚い」ページを生成します。

文書クラス *article (koma-script)* ・ *report (koma-script)* ・ *book (koma-script)* の各レイアウトファイルは、`scrmacros.inc` ファイルを読み込みます。このファイルは、ユーザ環境を定義する場所として想定されています。`scrmacros.inc` を、個人用レイアウトディレクトリにコピーして編集してください！

6.14.3 letter (koma-script)

が、部分的に変更を加えた形で収録されていますが、 $\text{L}_\text{Y}\text{X}$ 固有環境の $\text{L}_\text{Y}\text{X}$ コード・ト・箇条書き (一覧) は、新しくラベリング環境に置き換えられています。それ以外標準文書クラスとは異なり、標準環境の $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ ・引用 (字下げあり)・引用 (字下げなし) 詩句が収録されています。さらに、新しい letter 固有環境がたくさん追加されています。

この文書クラスで作成した書簡の外観は、様々な $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ コマンドを $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ プリントアウトに置くことで制御することができます^{*7}。これらの $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ コマンドのドイツ語的な説明は、Koma-Script の取扱説明書 *scrguide* にあります。この方法で、書簡は、個人用書簡レイアウトを作ることができます。

書簡環境および頭語環境は、書簡の冒頭部を定義するもので、すべての書簡で使ってはなりません。 $\text{L}_\text{Y}\text{X}$ 文書クラスでは、これらの環境を強調するために、左余白それぞれ L と O という文字が表示されます。同一ファイル中には、複数の書簡を書くことができます。頭語環境は、同じ受取人への書簡を新しく書き出すものであり、書簡環境は、新しい受取人を生成するものです。結語・PS・CC・Encl は、通常の段落環境なので、一の書簡中でも何度も使用することができます。

- 書簡：受取人のための段落をつくり、暗黙に書簡の開始を定義します。
- 頭語：敬称のための段落をつくり、暗黙に新しい書簡を開始します。
- 結語：結語のための段落をつくります。
- PS：追伸のための段落をつくります。
- CC：同報一覧のための段落をつくります。
- Encl：同封物のための段落をつくります。

名前・署名・住所・電話・場所 (Place)・返送先住所・配送手段 (Specialmail)・タイトル・主題は、情報入力を促すラベルの付いた入力環境で、文書クラスが適切に行います^{*8}。これらの環境は、対応する頭語環境の前に置く必要があります。

生成された書簡の実際の外観は、特定の環境の使い方だけではなく、他の要素に

^{*7} 例えば、名前と住所の入った書簡ヘッダの標準的外観は、きわめて頑固です。「普通の」ヘッダアンブル中に以下のようなコマンドを入れることで生成することができます。

```
\firsthead{\parbox[b]{\textwidth}
  {\ignorespaces \fromname\\ \ignorespaces \fromaddress}}
\nexthead{\parbox[b]{\textwidth}
  {\ignorespaces \fromname \hfill \ignorespaces \pagename\}}
```

するため、これらの環境を WYSIWYG として実装することには、あまり意味がありません。たとえば、署名環境で入力した署名は、標準的な挙動では、同じ書簡に結語環境も用されているときのみ、書簡の出力に現れます。電話環境に入力された値は、標準的な挙動では、書簡の出力には全く表示されません。書簡のヘッダを自由にデザインできることは、前出の脚注で示してあります。

入力環境は、空の段落として使うことができます。これは、たとえば署名環境の場合有効です。署名環境を全く使用しない場合は、標準的な挙動では、名前環境の値が署名として使用されますが、空の署名環境を作成しておくと、署名の値は無定義となりません。

入力環境を使用すると、個人データ（名前や住所など）が既に入力された入力環境と後から入力するデータの入力環境を備えた、書簡のひな型を作成することができます。

- 名前：送り主の名前。標準的な挙動では、書簡のヘッダ部にスモールキャップ体中央揃えの段落として表示されます。
- 署名：送り主の署名。標準的な挙動では、結語環境の下に表示されます。署名環境を使用しないと、代わりに名前環境の値が表示されます。
- 住所：送り主の住所。標準的な挙動では、書簡ヘッダの送り主の名前の下に中央揃えの段落として表示されます。
- 電話：送り主の電話番号。標準的な挙動では、 \LaTeX 変数 `\telephonenumber` を設定するだけです。
- 場所 (Place)：書簡を書いた場所。
- 日付：書簡を書いた日付。場所 (Place) と日付は、標準的な挙動では、受取人フィールドの下に、単一の右揃え行として場所と日付を生成します。空の日付環境を作成しておくと、場所 (Place) の値如何に関わらず、場所も日付も表示されません。日付環境を使用しない場合には、書簡を出力した日付が使用されます。
- 返送先住所：送り主の返送先住所。標準的な挙動では、受取人の上に小文字のサンセリフ体フォントで表示されます。
- 特別便：特別便の情報。標準的な挙動では、受取人フィールドの上かつ返送先住所の下に、下線付きで表示されます。
- 場所：追加情報。標準的な挙動では、受取人フィールドの下の右側に表示されます。
- タイトル：書簡の題名。標準的な挙動では、主題の上に大きなボールド体サンセリフフォントで表示されます。

使用されます。しかしながら、これらの「ビジネス書簡環境」の一つを使用した場合、場所 (Place) の値は書簡には表示されなくなり、 \LaTeX 変数 `\fromplace` が設定されるようになります。通常時に、受取人フィールドの下に右揃えで出力される場所と日付は、抑制されます。これらのビジネス環境は、ラベル付きの入力環境として実装されており、対応する頭語環境よりも前に置く必要があります。

- Yourref: Your ref.
- Yourmail: Your letter of.
- Myref: Our ref.
- Customer: Customer no.
- Invoice: Invoice no.

6.14.4 新しい letter クラス : letter (koma-script v.2)

by JÜRGEN SPITZMÜLLER

Koma-Script 第 2.8 版では、新しい letter クラス `scrlettr2` が導入されたため、`scrletter` は廃止されてサポート対象外となりました。これは— \LaTeX 側で—全く新しいイフェイスを持っており、旧クラスとは互換性がありません。したがって、 \LaTeX は両方サポートしていますが、新しい方のクラスを使用することを推奨します。

このクラスは、*letter (koma-script)* と同一の機能に加えて、新しい機能をいくつかサポートしています。基本となる項目は、住所（受取人の住所、旧レイアウトの書じ）・頭語・結語です。次の住所は別の書簡を新しく開始します（つまり一文書あ数の書簡を書くことができます）。新しく付け加えられた項目は、送り主の電子メ URL・ファックス・銀行、およびヘッダに（挿入▷図で）ロゴを入れることができた点です。

しかしながら、もっとも大きな改善は、書簡のレイアウトを、ほとんどどんなにも合わせて設定できるようになった点でしょう。これはプリアンブルか、クラクションとして読み込まれる特別なスタイルファイル（「レタークラスオプション」子*.lco）によって設定することができます*⁹。用例としては、 \LaTeX に収録されている型 *koma-letter2* をご覧ください。詳細な解説は、Koma-Script 取扱説明書（*scr*）にあります。

6.15 Latex8 (IEEE 学会論文)

6.14.5 問題点

Koma-Script 文書クラスを L_YX 上に表示する上で、L_YX の内部処理が若干の問題を
じめます。

- 章環境の章番号は、章見出しと同じ行ではなく、その上に表示されます。これはレイアウトファイル中の Counter_Chapter ラベル型に対する L_YX の内部処理が原因です。
- 章 (addchap) と節 (addsec) 環境の見出しは、「本当の」L^AT_EX の目次には追加されますが、L_YX の目次 (移動メニュー) には追加されません。
- *letter* 文書クラスでの段落は、垂直スペースによる段落区切りで表示され、字下げはされません。これが標準的な挙動となっていて、特別な L^AT_EX コマンドは必要ありません。しかし、文書▷設定ダイアログでは、対応するラジオボタンは字下げ設定されています。垂直スペースは、隙間を空けるために、文書に L^AT_EX コマンドをつねに追加するようになっていますが、この文書クラスの場合には、そのような挙動が望まれないためです。

6.15 Latex8 (IEEE 学会論文)

ALLAN RAE 文

6.15.1 はじめに

このクラスは、IEEE 提供の学会へ投稿する原稿を執筆する特定の目的のためのものですので、IEEE の Authors Kit を入手されることを強くお勧めします。このキットには latex.sty パッケージと付属の書誌情報スタイルファイルが含まれています。Authors Kit は、通常、初回投稿が受理されたときに、電子メールで送付されてきます。Authors Kit には、組版上の制約をはじめ、たくさんの役立つ情報が含まれていますので、以下ではあなたがこれを読まれたものと仮定して、同一の内容は省略するものとします。

6.15.2 はじめに

6.15.3 サポートされている環境

- 標準
- タイトル
- 著者
- 電子メール
- 所属
- 概要
- 節
- 小節
- キャプション

6.15.4 画面と印字の違い

おもに節番号の表示に関して、表示の仕方に小さな違いがあります。画面では、の後のピリオドが表示されませんが、出力には表示されますので、心配する必要はありません。

6.16 Memoir

JÜRGEN SPITZMÜLLER 文

6.16.1 概観

Memoir は、非常に強力で、着実に改良が加えられている文書クラスであり、フィクションとノンフィクションの文学作品用に設計されています。その目的は、ユーザーが文書の組版に関して最大限の裁量を得させることにあります。Memoir は、標準 `book` クラスに基づいていますが、`article` クラスをエミュレートすることもできます（以下

Memoir 開発者である Peter Wilson は、 \LaTeX の世界での、多くの便利なパッケージの作者として知られていますが、それらのほとんどは、Memoir に統合されました。そのため、章のデザインや、目次や付録などのレイアウトを、はるかに簡単に行うこと

しませんが^{*10}、 $\text{L}_\text{Y}\text{X}$ のフレームワークが持つ制約のために、おそらく決して追加することのないと考えられるものも多数あります。もちろん、ネイティブな $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ コマンド ($\text{T}_{\text{E}}\text{X}$ コード^{*11}) の助けを借りれば、全機能を使用できます。この節では、 $\text{L}_\text{Y}\text{X}$ がネイティブにサポートしている機能についてのみ、列挙するものとします。詳細な説明 (および他の機能) については、Memoir クラスの詳しい取扱説明書^{*12}をご覧ください。この取扱説明書は、本クラスのユーザーの手引きであるのみに留まらず、良組版に関する包括的な解説であり、それ自身が良い組版の傑出した用例となっています。

6.16.2 基本的な機能と制約

Memoir は、基本的に、標準 book クラスの全機能をサポートしています。しかしながら、以下に示すような若干の違いがあります。

フォント寸法: Memoir では、 $9 \cdot 10 \cdot 11 \cdot 12 \cdot 14 \cdot 17$ と、より広い範囲のフォント寸法が使用可能です。

ページ様式: Memoir パッケージと fancyhdr パッケージとのコマンドの衝突のため (パッケージは同じ名前のコマンドを定義しているため、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ が混乱させられます) fancy ページ様式は使用できません。その代わりに、Memoir には、多くの独自のページ様式が付属してきます (文書▷設定▷ページレイアウト参照)。これらを題ページに使用したい場合には、コマンド `\chapterstyle` を、本文またはプリントブルに記述する必要があります (例えば、`\chapterstyle{companion}` など)。

節分け: 標準クラスでは、節分けコマンド (章・節・小節等) は、非必須引数を取ることができます。これを用いて、目次やヘッダ用に別のタイトルを指定することができます (たとえば、タイトルが長すぎる場合などに用います)。 $\text{L}_\text{Y}\text{X}$ では、章節の始めで挿入▷短縮タイトルを使えば、これを指定することができます。一方 Memoir では、第 2 の非必須引数を取ることができ、目次用のタイトルとヘッダ用のタイトルの指定を分離しています。これを使えば、本文用と目次用とヘッダ用タイトルの 3 種類を指定することができます。この機能を使用したい場合には、2 つの非必須引数を入力して、最初のものに目次用、次のものにヘッダ用短縮タイトルを入れてください。

目次・図一覧・表一覧: 標準クラス (および他の多くのクラス) では、目次と図一覧および表一覧は、自動的に新担当頁に置かれます。Memoir は、これとは別の方針を採

タイトル頁: 理由は不明ですが、Memoir はタイトルページに丁付けを行います (標準クラスでは、タイトルページは「empty」様式、すなわち丁付けしません)。タイトル頁に丁付けを行いたくない場合には、プリアンブルに `\aliaspagestyle{title}{empty}` と入力してください。

Article: クラスオプション *article* を用いる (文書▷設定▷クラスオプションで入力) *article* 様式をエミュレートできます。すなわち、(脚注・図・表等の) カウンタは章毎にリセットされず、次の章は頁を改めず (ただし、「本物の」*article* クラスは違って、頁を改めることも可)。部は *book* クラスと同様に、独立の頁を扱います。

Oldfontcommands: L^AT_EX 旧 2.09 版で使用されていた古いフォントコマンド (`\it` など) は、Memoir においては、既定では使用することができません。Memoir がこれらのコマンドを検知すると、エラーを発して L^AT_EX を停止させます。しかし、クラスオプション *oldfontcommands* を指定すると、これらのコマンドが警告で置き換えられ、エラーの代わりに警告を発するのみとなります (これによって、少なからず L^AT_EX は停止しなくなります)。まだ、多くのパッケージと、就中多くの B^S スタイルファイルが、これらのコマンドを使用しているので、我々はこのオプションを既定で使うこととしました。

6.16.3 追加された機能

L^AT_EX でサポートされる機能 (今のところそれほど多くはありません) を簡単に説明しよう。詳しくは Memoir の取扱説明書^{*13} を見てください。

概要: 概要が、追加された機能として扱われているのを不思議に思われるかもしれませんが、*book* クラスとしては追加された機能なのです。通常 *book* クラスにはないのですが、*memoir* にはあります。これは、どこでも何回でも自由に行うことができます。

章要約: この古い組版様式はご覧になったことがあるかも知れません。章の内容の出しの下と目次に要約されているものです (たとえば、「我らが英雄がトロイに着する。彼は友人を失うが、新たな友人を得る。」など)。章要約 (chapter summary) はこれを行います。したがって、章要約は章の下でのみ意味を持ちます。

は、短い線で区切られます。この環境には二つの引数（本文と出典）が必要なのです。ここでも L_X を少しだます必要があります。ここでは、2 つの引数の間に、`<標語>{\<標語の著者>}` のように、（ T_E X モードで）波括弧を入れなくてはなりません。

詩題: Memoir には（複雑で隠喩的なものも含めて）詩を組版するための多くの機能が提供されていますが、 L_X は、そのうちいくつかしかサポートしていません。詩題はその一環です。これは詩用の中央揃えタイトルで、目次にも表示されます（詩句が詩の標準環境です。Memoir には、詩句の拡張版もありますが、これらを使うためには T_E X コードを使用しなくてはなりません。これらの環境は通常の詩句環境の中に入れなくてはならないのですが、 L_X ではそれができないためです）。

詩題*: 目次に現れない他は詩題と同じです。

6.17 Article (mwart)・book (mwbk)・report (mwrep)

TOMASZ LUCZAK 文

L_X 文書クラス *article* (mwart)・*report* (mwrep)・*book* (mwbk) は、それぞれ $\text{L}_\text{A}_\text{T}_\text{E}_\text{X}$ 文書クラス *mwart.cls*・*mwrep.cls*・*mwbk.cls* に対応しています。これらは、標準 *article.cls*・*report.cls*・*book.cls* を代替するものであり、多くの点で、ポーランド語の組版慣習によく適合しています。

基本的な違いは、次の点です。

- 付番のない見出し（節*などの星付き）も目次に加えられます。
- 以下のページスタイルの追加
 - `uheadings` 分離線付きヘッダ
 - `myheadings` `\markright` コマンドおよび `\markboth` コマンドで設定するカスタムヘッダ
 - `myuheadings` 分離線付きカスタムヘッダ
 - `outer` ページの外側にページ番号
- オプション
 - `rmheadings` セリフ体見出し — 既定
 - `sfheadings` サンセリフ体見出し
 - `authortitle` タイトルページは、著者名・タイトルの順で配置 — 既定

6.18 Paper

paper 文書クラスは標準 article クラスの代替となるものです。これは同様の機能しますが、サンセリフ体の節見出し・ヘディング等を持つこちらのレイアウトのに入るかも知れません。

6.19 RevTeX4

by AMIR KARGER

RevTeX 4 テキストクラスは、アメリカ物理学会の RevTeX 4.0 (1999 年 5 月の β ス) クラスで動作します。

L^AT_EX には、RevTeX 3.1 で動作する RevTeX テキストクラスもあります。しかし第 3.1 版は L^AT_EX 2.09 で動作するものであり、基本的に時代遅れなものになっています。L^AT_EX は L^AT_EX 2_ε を必要とすることから、第 3.1 版は L^AT_EX とうまく協調しないことしますが、動作するようには調整してあります。RevTeX 4.0 は、L^AT_EX 2_ε で素直するようにデザインされているので、L^AT_EX と RevTeX 4 テキストクラスの組で使のは、たいへん簡単です。

この取扱説明書は、RevTeX 4.0 の説明書の補完として使用されることを前提とします。したがって、ここでは RevTeX 固有マクロの説明はせず、必要な場合にはサンプルに何と書けば良いかを知っているものとして説明します。

6.19.1 導入

RevTeX 4 パッケージの README ファイルにあるとおり、とにかく RevTeX 4 を導入しなくてはなりません。パッケージは、The RevTeX 4 Website (<http://publish.aps.org/revtex4/>) から入手することができます。これを L^AT_EX 2_ε が出ることのできる場所に導入してください。それから、適当に選んだディレクトリ (このディレクトリは、標準的な LaTeX クラスファイルが導入されたのとは異なるディレクトリ) で簡単な RevTeX 4 テキストファイルを L^AT_EX でコンパイルできるかどうか確認してください。この後、L^AT_EX を再初期設定し、クラスファイルを検出し、RevTeX4 テキストクラスが使用できるようにする

6.19.2 プリアンブルに関すること

`\documentclass` に与える「preprint」や「aps」のような非必須引数は、他のクラス様、文書▷設定▷文書クラスのクラスオプションフィールドに入れます。RevTeX では少なくとも一つの非必須引数が必要なことに注意してください。

`\draft` など他のプリアンプルは、他のクラス同様、文書▷設定▷`LATEX` プリアンプルに入れてください。

6.19.3 レイアウト

レイアウトは、基本的に RevTeX 4.0 のコマンドに対応しています。たとえば、Email
レイアウトは、`\email{}`に対応しています。(少なくとも RevTeX 4.0 Beta 現在では)
住所と所属はまったく同じものなので、両方を使う必要はありません^{*14}。

6.19.4 重要な注意

RevTeX 4 には、バグを引き起こしがちなために、気をつけなくてはならない特殊な
がいくつかあるため、`LyX` ではさらに気をつけなくてはなりません。

RevTeX では、`\thanks` コマンドは、`\author` コマンドの外側に来なくてはなりません。LyX では、独立した感謝レイアウトがこの働きをします。直接、著者レイアウトに脚注を付けてしまうと、おかしいことが起こる可能性があります。詳細については RevTeX 4 の取扱説明書をご覧ください。

また、著者電子メール・著者 URL・感謝 (thanks) レイアウトは、著者レイアウトと所 (または同等の所属) レイアウトの間に置かなければなりません。感謝 (thanks) を所の後に置いてしまうと、 \LaTeX はコンパイルに失敗します。

6.19.5 問題点

このレイアウトの主な問題点は、電子メールやタイトルのようなレイアウトに非必須引数を用いることができないことです（これはこのレイアウトだけの問題ではありません。例えば、節レイアウトも非必須引数を使用することができません）。これは、たとえばページ番号が偶数の場合でも奇数の場合でも右側に配置されるべきであるというように、

ストエディタを使って L^AT_EX ファイルを編集し、非必須引数を書き加えなくてはいいということです。これらのレイアウトがないと、`\altaffiliation` (またそれな `\altaddress`) は使えませんが、対応するレイアウトは存在せず、手動で書き必要があります^{*15}。

6.20 Springer 学術誌 (svjour)

MARTIN VERMEER 文

6.20.1 説明

これは、Springer Verlag 社発行のいくつかの学術誌で使用されている書式のレイアウトファイルです。<http://www.springer.de/author/tex/help-journals.htm> 必要なクラスファイル (L^AT_EX 2_ε 用になりました) とそれに対応する学術誌の一覧があります。このレイアウトはモジュール式になっています。つまり、全ジャーナル共は `svjour.inc` に実装されており、各誌個別のレイアウトファイル (たとえば `Jou Geodesy` 用の `svjog.layout` など) から取り込むことができます。

このことは、上記の学術誌一覧にある他の Springer 誌のサポートを実装するため `svjog.layout` に書かれている指示に従って、自作の `sv<学術誌名>.layout` を付けて良いことを意味します。

このレイアウトは、*Journal of Geodesy* に対してのみ、十分な試験が為されます。svjour と svjog は、標準 L_YX 頒布版に収録済みです。関連するクラスファイル (Springer からダウンロードして) 適切なディレクトリに導入し、L^AT_EX を再設定 (te_TE_X の場合は、必要に応じて root 権限で `texhash` を実行します) L_YX を再定すれば、動作するはずです。

6.20.2 新スタイル

多くの定理環境 — 主張・推論・...・定理。

ヘッドノート・献呈・サブタイトル・ヘッダ用 L^AT_EX タイトル・ヘッダ用著者名・所属メール・オフプリント・キーワード・謝辞 (acknowledgements)・謝辞 (acknowledgements) 詳しくは Springer のクラスファイルの説明書を読んでください。

6.20.3 サポートしているジャーナル

- *Journal of Geodesy*: `svjog.layout` — Martin Vermeer
- *Probability Theory and Related Fields*: `svprobth.layout` — Jean-Marc Lasgouttes

あなた自身のレイアウトファイルをつくってください。難しくはありません。

6.20.4 Credits

これらのファイルは、以前の `ejour2.layout` をもとにしています。`ejour2.layout` は Springer の L^AT_EX 2.09 用のスタイルファイルを使った素人細工のようなレイアウトファイルです。これらのファイルはすでにありません。Jean-Marc Lasgouttesのおかげで L^AX のレイアウトファイルのメカニズムをよりよく理解することができました。

6.20.5 バグ

多分あります。以前の `ejour2` よりは少なくなっているとは思いますが。

制限事項：定理環境の環境の番号は表示されません。`#`が表示されるだけです。

6.21 Slides (別名 SLIT_EX)*¹⁶

JOHN WEISS 文

6.21.1 はじめに

本節では、オーバーヘッドプロジェクタ用のスライドをつくるのに、L^AX をどのように使えばよいかを説明します。これを行う文書クラスには、既定の `slide` クラスと `FoilTeX` スライドクラスの 2 つがあります。本節は前者について説明を行います。

誤解のないように、正しく明確に、もう一度言っておきます。

本節は文書クラス “`transparencies (slides)`” のみを説明しています。

(`FoilTEX`)」は、実際、本節で説明している既定 `slides` クラスよりも良いようです*。

このクラスは、旧 `SLITEX` パッケージを `LATEX 2ε` 用に改良したものです。`LATEX` 布版すべてが、このクラス（以下では単に「`slides`」と述べることにします）を含んでいるので、あなたもお持ちのはずです。前述の `foils` のように、同じくオーバーヘッドプロクタ用のスライドを生成しながら、このクラスよりも良い仕事をするクラスもあり、しかしながら、オーバーレイを生成するなど、他のクラスにはできないが `slides` なくともあります。詳しくは以下をお読みください。

6.21.2 使い始めるには

当然ながら、この文書クラスを使用するには、文書▷設定ダイアログのクラスリストから「`transparencies (slides)`」を選択しなくてはなりません。このクラスについて、ように、特に知っておいた方がよいことがいくつかあります。

- 用紙方向と二段組文書の各オプションは、あえて変更する必要はありません。これらのオプションは、`slides` クラスではサポートされていないので、効果ありません。
- ヘディング様式オプションは、このクラスでは少し異なった振る舞いをします。選択肢とその機能は、以下の通りです。
 プレーン (plain) 最終出力の右下隅にページ番号が付きます。
 設定 (headings) プレーン (plain) と同様ですが、それ以外にあなたが指定したタイムマーカーをすべて出力します。これが既定値です。
 空 最終出力には、ページ番号もタイムマーカーも位置揃えマーカーも出力しません。
- `slides` クラスには、もう一つ `clock` というオプションがあります。これを使用するには、クラスオプションに「`clock`」と入力してください。
 このオプションを使うと、注釈にタイムマーカーを加えることができます。は、第??節をご覧ください。

`slides` クラスを用いた文書を自動的にセットアップするには、ひな型ファイル「`slides.lyx`」を使用することもできます（新規ファイルを開く際にファイル▷新ひな型使用）を使用してください。ひな型ファイルには、このクラスが使用する特

6.21.3 段落環境

6.21.3.1 サポートされている環境

新しく slides 文書を開いたときにまず気付くのは、フォント寸法とフォント族の違いでしょう。ここでは、サンセリフ体フォントの極大寸法が用いられており、出力でもこのフォントが使用されます。これは、現在スライドを作成中であることの「目印」だと考えてください。出力スライドは大きなフォントを使用するため、書ける場所は少なくなります。もちろん、既定の大きな画面フォントは WYSIWYG ではなく、スライドであるこの目印に過ぎません。

次に気付くのは、(ツールバー左端の) 段落環境プルダウンボックスが変わっているでしょう。通常見かける環境の多くがなくなって、代わりに 5 つの新しい環境が入っています。これは、slides クラスが、標準段落環境のうち以下のものしかサポートしているためです。

- 標準
- 箇条書き (記号)
- 箇条書き (連番)
- 箇条書き (記述)
- 箇条書き (一覧)
- 引用 (字下げあり)
- 引用 (字下げなし)
- 詩句
- キャプション
- `LyX` コード
- コメント

これ以外の標準的な環境、例えば節見出し用の環境などは slides クラスでは使えません。一方、以下の新しい環境があります。

- スライド
- オーバーレイ
- 注釈

L_AT_EX では、空の段落環境に他の環境を入れることができません。それ自体は素晴らしいのですが、これは、スライドを平文以外では書き始めることができないことを意味します。これを取り扱うためには、ちょっとした「L_AT_EX マジック」が必要になります。

6.21.3.2 新環境の癖

5 つの新しい環境にはすべて、L_AT_EX 現行版の制約に由来する癖があります。いまいちよく、L_AT_EX では、環境を別の環境で開始することが禁止されています。これを回避するために、スライド環境は『ユーザーの手引き』で述べられているような段落環境でなっています。

スライド・オーバーレイ・注釈は「疑似環境」だとお考えください。これらは節や「キャプション」のように見えますが、実際には（必要なら前段落を終了して）段落環境を開始します。同様に、不可視文と可視文も「疑似コマンド」のお考えください。この 2 つは特定のアクションを行います。

スライド・オーバーレイ・注釈・不可視文・可視文の 5 環境に共通しているのは、このラベルを持つことです。このラベルに続く文章—通常は段落環境の内容になる文章—は、スライド・オーバーレイ・注釈・不可視文・可視文では、まったく意味がなくなります。L_AT_EX はこの部分を完全に無視します。これらの 5 つの環境の内容は、完全に空の環境としておいて構いません。

長めのラベルの後に文章を入れる必要はまったくないとはいえ、何か書いておくともあるかもしれません。たとえば、スライドの内容に関する短い説明などがそう。この場合、いつもどおり説明用のコメントを入力して Return を押してください。

一方、説明文を入れたくない場合には、L_AT_EX の制約に直面します。自然が真空のように、L_AT_EX では、古い環境に何かを置くまでは、その中に新しい環境を入れることができません。そこで以下のようにしてください。

- 新しいスライド・オーバーレイ・注釈・不可視文・可視文の中に入れる文章を入力してください。
- 段落の先頭に移動します。
- 次に Return を押します。
- 最後に、この新しい空の段落をスライド・オーバーレイ・注釈・不可視文・可視文に変更します。

6.21.4 スライド・オーバーレイ・注釈で発表用のスライドを作る

6.21.4.1 スライド環境を使う

もし本節が、実際のプレゼンテーションの作り方を教えてくれるものと期待されてい
とすれば、大いにがっかりされるかもしれません。当然ながら、プレゼンテーション用
資料を準備するうえで `slides` クラスを役立てる方法を詳しく説明はしますが、内容をど
するかはあなた次第です (そしてこれも `LATEX` の哲学です)。

スライド環境を選択すれば (方法は第 6.21.3.2 節参照のこと) `LATEX` は新規スライド
開始します。この環境ないし「疑似環境」のラベルは、鮮やかな青で「新規スライド:」
表示されます。この環境の後のテキスト環境や段落環境は、すべて新規スライドに入り
ます。簡単ですね。

`Slides` は、おそらく `LATEX` の中で強制的に改頁をしなくてはならない唯一のクラスで
(これは段落レイアウトダイアログで指定できます)。実際、一つのスライドの内容を入
し終わったら、改頁したくなるはずで。仮に一つのスライドに物理的に入りきらない
章を入力したとすると、入りきらない部分は、次の新規スライドにまわされます。しか
ながら、入りきらない部分のスライドにはページ番号が付きませんので、こうするのは
勧めしません。その上、一枚に収まらないスライドでは、オーバーレイを使用すると干
渉する恐れがあります。

オーバーレイ環境と注釈環境も、スライド環境と同様の動作を示します。これらは、
の代わりに鮮やかなマゼンタで、それぞれ「新規オーバーレイ:」ないし「新規注釈:」
というラベルを表示します。ラベルのフォントも、3 つの環境でそれぞれ異なるものが使
われます。

スライドと同様に、オーバーレイや注釈の内容が、一つのスライドないしページに物
理的に収まりきらない場合には、余分は新規スライドに回されます。この場合には、注釈
オーバーレイの本来の目的が達せられなくなりますので、ここでもこのような状態は避
けるべきでしょう。

6.21.4.2 スライドでオーバーレイを使用する

オーバーレイの背後にある考え方は、別のスライドの上にスライドを載せようとする
のです。メインのスライドに図があるとき、関連した文章を表示する前に、図について

ません。これらのような場合には、オーバーレイ環境を使えば、もっと楽になります。

各オーバーレイは、「親」スライドのページ番号を引継ぎ、その後に「-a」を付けます^{*19}。当然のことながら、スライドとオーバーレイの内容は両方とも、1枚のスライド物理的に収まるようになっていなければなりません。オーバーレイはスライドの一部分であると考えるのがよいでしょう。L_AT_EX では、オーバーレイのラベルをスライドのラベルよりも下げることで、このことを視覚的に表しています。また、印刷可能な出力すると、オーバーレイと親スライド両方の四隅に、揃えマーカーがあることに気づかずです。これは、2枚のスライドを物理的に重ね合わせるための補助となるものです。

2枚のスライドをオーバーレイさせる際の大きな問題は、2枚のスライドの内容を揃えることです。2枚目のスライドのグラフには、どれだけの余白を残しておくべきでしょうか？さらにややこしく、2枚目のスライドにグラフと文章を入れたいときに間に割り込んでしまうような文章がメインスライドにある場合には、どうでしょう。美しい寸法の垂直スペースを挿入してみることもできるでしょう。しかし、もっと良しとして、不可視文と可視文を使用する方法があります。

これらの名前が意味するとおり、不可視文と可視文は、それぞれ続く文章すべえなくしたり見えるようにしたりするコマンド型の段落環境です。しかしながら 6.21.3.2 節で述べたように、この 2つの環境の中には何も置くことができません。文を作成すると、「<以下不可視文>」と書かれた水色のラベルが、ページに中央揃え表示されます。このラベルに続く段落は、それが置かれるべきスライド（あるいはオーバーレイ；どちらでも同じことです）部分に、代わりに空白の部分が置かれます。

可視文の場合には、中央揃えのラベルは、鮮やかな緑の「<以下可視文>」となります。このラベル以降の段落は、通常通りの挙動を示します。新規のスライド・オーバーレイ注釈を開始すると、不可視文は自動的に終了されることに注意してください。そして、スライドの終わりでは、可視文を使用する必要はありません。

ここまでで、スライドやオーバーレイ上で不可視文と可視文を適切に使用して、ドキュメントをオーバーレイさせる方法が明らかになったことと思います。

1. メインのスライドかオーバーレイかに関わらず、スライド上に現れるすべてのスライドを作成します。
2. オーバーレイのみに表示するすべての図や段落の前に、不可視文環境を挿入します。必要ならば、オーバーレイのみの文章の後に可視文環境を挿入します。

5. オーバーレイ内部の不可視文行をすべて不可視文に変更し、可視文は不可視文に変更します。

これだけです。これでオーバーレイができあがりしました。

`LYX` slides クラスをデザインする上で、ひとつ残された問題があります。段落途中の章を不可視にしたり、不可視段落の途中の文章を再度可視にしたりすることはできないです。これを実現するためには、`LaTeX` コードを埋め込む必要があります^{*20}。

6.21.4.3 スライドで注釈を使う

オーバーレイと同様、注釈も「親」スライドに関連付けられています。ここでも `LYX` スライドクラスは視覚的な効果を与えています。注釈のラベルはスライドよりも短く（オーバーレイのラベルよりも長いですが）、オーバーレイのラベルと同様、派手な赤紫色です。加えて、印刷した注釈には、「親」スライドのページ番号に「-1」「-2」「-3」等を付け、ページ番号が付けられます。1つのスライドには、複数の注釈を関連付けることができるので、スライドとオーバーレイと同じように、長い注釈は、1枚の紙に収まるように分したいと思われるに違いありません。

注釈の目的は明らかです。スライドに関して付け加えたいことを書き留めておくことです。特定のスライドについての覚書として使うこともできます。後者の場合には、タイムマーカを利用したいこともあるかもしれません。現在のところ、`LYX` のスライドクラスは、`SLiTeX` の機能であるタイムマーカを「ネイティブには」サポートしていません。たがって、`LaTeX` コードを使用することに頼らなくてはなりません。

タイムマーカを使用するには、クラスオプション「`clock`」を追加指定する必要があります（第??節参照）。このオプションはタイムマーカ機能を有効にして、生成したすべての注釈の左下にタイムマーカを表示します。タイムマーカに表示するものを設定するには、`LaTeX` コマンド「`\settime{}`」および「`\addtime{}`」を使用します。両コマンドは、秒で測った時間を引数にとります。「`\settime{}`」は、タイムマーカを特定の時間設定し、「`\addtime{}`」は、タイムマーカを指定した分だけ進めます。このようにタイムマーカと注釈を用いれば、特定のスライドにどれだけの時間をかけるのか忘れないでいられます。

最後に、説明しておくべき機能がもう一つあります。当然ながら、スライドとオーバーレイ

レイはすべてスライドに、注釈はすべて普通の紙に印刷したいはずです。しかしながら注釈は関連付けられた親スライドの直後に来なくてはなりません。ユーザーはどうしたいのでしょうか。

幸いなことに、どれを印刷するか選択することを可能にする \LaTeX コマンドが複数あります。どちらも文書のプリアンブルに記述されなくてはなりません。コマンド「`\onlyslides{\slides}`」は、出力にスライドとオーバーレイのみが含まれるようになるものです。同様に、「`\onlynotes{\notes}`」は注釈以外の出力をすべて抑制します。このアドバイスとしては、両コマンドをプリアンブルに書いておいて、最初は両方ともコメントアウトしておくといいでしょう。そうすれば、執筆時にはプレゼンテーション全体をレビューすることができます。いったん書き終えたならば、二つのうち一方のコメントを外して、印刷したい方を選ぶことができます。私ならば、まず「`\onlyslides{\slides}`」のコメントを外して、現ファイル名に「*-slides」を付けた新たなファイルに出力します。いい、今度はこの行をふたたびコメントアウトするとともに、「`\onlynotes{\notes}`」のコメントを外して、これを「*-notes.ps」というファイルに出力します。そうすれば、必要に応じて、スライドまたは通常の紙をプリンタにセットし、一方のファイルをプリンタに送ればよいのです。

「`\onlyslides{}`」および「`\onlynotes{}`」コマンドには、他の引数を与えることができます。詳しくは、 \LaTeX の良書をご覧ください。

6.21.5 slides クラスのひな型ファイル

slides クラス用のひな型ファイルとして「slides.lyx」が提供されています。使用するには、新しいプレゼンテーションを作り始める際に、ファイル▷新規 (ひな型) メニューを選択してください。すると新規ファイルには、スライド オーバーレイ 注釈の組が、例として含まれており、スライドとオーバーレイには、さらに不可視注釈の使用例も含まれています。また、プリアンブルには、次のように書き込まれます。

```
% Uncomment to print out only slides and overlays
%
%\onlyslides{\slides}
```

6.21 Slides (別名 S_{LI}T_EX)

最後に、私はこの文書クラスを、 \LaTeX 2 _{ϵ} の組み込みクラスである「S_{LI}T_EX エミュレーション」クラスをサポートする目的で作成しました。私も L_YX チームの他のメンバーもこの組み込みスライドクラスを使用することに賛成したり反対したりするものではありません。これは単に、ユーザーがこれを使いたかったり必要としたりしたときのためあるものです。プレゼンテーションを作成するためには、Foils クラス (第 6.9 節参照)「seminar」パッケージ (一部の \TeX 頒布版に含まれています) のように、他の \LaTeX クラスも存在します。私はこれらについては知らないのですが、他にどのようなことができるのか試してみてください。

第 7 章

ソフトウェアを追加する必要がある L_YX 機能

7.1 T_EX のチェック

ASGER ALSTRUP 文

7.1.1 はじめに

`chktex` プログラムが導入されていると^{*1}、ツールメニューの下に、T_EX 構文チェックという項目が表示されます。`chktex` は、CTAN<http://www.ctan.org/tex-archive/help/>から入手することができます。

ChkT_EX パッケージは、L^AT_EX の構成物には直感的でないものもあり、すぐ忘れてしがちであるのに不満を感じた JENS T. BERGER THIELEMANN が書いたプログラムです。このプログラムは、L^AT_EX ファイルを走査して、ファイルの整合性をチェックし、しがちな誤りを見つけ出します。別の技術用語では、L^AT_EX における `lint` であるとも言えます。

さて、L_YX はつねに正しい L^AT_EX ファイルを生成するはずなのに、文法チェッカー L_YX で何をするというのでしょうか。答えは簡単です。`lint` が C 言語プログラムの法をチェックするだけでなく、打ち間違いによる意味上のチェックも行うように、ChkT_EX は、文法エラーの他に犯しがちな写植上の誤りも検出してくれるからです。ChkT_EX は、以下のような報告を出力するようになります。

...ではなく...を使わなくてはなりません

- 括弧の前後にスペースを置かない：

(wrong spacing)：誤った空白

- よく使われる省略形の後ろは通常スペースにする：

e.g. is too wide spacing.：広すぎる空白

- 直前の文章が大文字で終わっているときに、文末スペースに直す：

This is a TEST. And this is wrong spacing.：誤った空白

- ラベルや類似コマンドの前のスペース：

ラベルが別のページに行ってしまうように、ラベルは文章に接していません。^{*2} このラベルは離れすぎです。

- 参照の前にハードスペースではなくスペース：

運が悪ければ、参照文字列と参照番号のあいだでちょうど改行されてしまい、望まぬ結果となります。第 7.1.1 節をご覧ください。

- 数字のあいだで × ではなく「x」が使われている：

2x2 は、2 × 2 よりも安っぽく見えます。

などなど...ChkT_EX は、印刷する前に文書を「仕上げる」上で、たいへん価値のあるツールです。必須のスペルチェックを終えた直後、組版の微調整をする前には、これをするべきです。

7.1.2 使いかた

このプログラムを導入したならば、使うのはツール▷T_EX 構文チェックを選択し、開けるので簡単です。こうすると、L_YX は使用中の文書の L^AT_EX ファイルを生成し、ChkT_EX を起動し、当該ファイルを検査しますが、ChkT_EX から警告があった場合、それを含んだ「エラーボックス」を L_YX 中に挿入します。警告は、誤りのある箇所に置かれるので、メニュー項目の移動▷エラーを選択するか、既定の cua キー設定ならば短絡キー C-g を押せば、簡単に警告の箇所を見出すことができます。ボックスを開けるには、マウスでボックスをクリックするか、cua キー設定の場合短絡キー C-i、emacs キー設定の場合は短絡キー C-o を押します。警告を読み、それを見示していれば修正します。警告が何を意味しているか理解しにくい場合には、無

7.1 T_EX のチェック

本文書自体、この機能を確認する良い材料になりますので、実行してみてください。すると、かなりの量の警告が出るはずです。これはコンピュータが敏感すぎるせいです。警告のほとんどは実際には警告に値しません。

7.1.3 微調整のしかた

場合によっては、ChkT_EX は、あなたが望むよりも煩すぎると思うかもしれません。のような場合には、必要と思うまで ChkT_EX を無効にするか、あなたにもう少し合うように ChkT_EX を調整することができます。どうしてもない場合には、表示▷すべてのラテックスを削除を選択して、すべての警告を即取り除いてしまうこともできます。

ChkT_EX は調整や拡張がひじょうに柔軟なのですが、L_YX における ChkT_EX 関連の話題が、それですべて解決すると思うべきではありません。L_YX は、ChkT_EX 出力*3 から行番号と内部構造を一致させるため、やや特殊な L^AT_EX ファイルを生成しなくてはならないので、警告の一部は正しく表示されていないように見えるでしょう。これに対処するためには、以下の 2 つの方法があります。

- 設定 (出力・L^AT_EX タブ) 中の ChkT_EX 起動コマンドラインを調整するか、グローバルな ChkT_EX の設定ファイル (通常は `chktextsrc` ファイル) を調整します。コマンドラインからどの警告を有効・無効にできるかは以下をご覧ください。
- ファイル▷書き出し▷L^AT_EX を使って文書を L^AT_EX ファイルに変換し、そのファイルに対して `chktext` を手動で実行します。このようにすると、L_YX 文書中の対応所を見つけるのがやや面倒ですが、少し我慢すれば見つけることができます。

以下は、設定で有効化・無効化できる警告メッセージです。警告を無効にするには `-n`、有効にするには `-w#` を用います。イタリック表示の項目は、既定で無効となっています。既定値は `"chktext -n1 -n3 -n6 -n9 -n22 -n25 -n30 -n38"` です。

L_YX は、`chktext` と通信するために、他のコマンドラインパラメータを特定の方法で用いているため、メッセージを有効化・無効化する以外のオプションを使うべきではありません。

1. *Command terminated with space.*
2. Non-breaking space (“~”) should have been used.
3. *You should enclose the previous parenthesis with “{ }”.*

7. Accent command “cmd” needs use of “cmd”.
8. Wrong length of dash may have been used.
9. “%s” *expected, found “%s”*.
10. Solo “%s” found.
11. You should use “%s” to achieve an ellipsis.
12. Inter-word spacing (“\ ”) should perhaps be used.
13. Inter-sentence spacing (“\@”) should perhaps be used.
14. Could not find argument for command.
15. No match found for “%s”.
16. Math mode still on at end of L^AT_EX file.
17. Number of “char” doesn’t match the number of “char”.
18. You should use either “ ” or “ ” as an alternative to “ ”.
19. You should use “ ” (ASCII 39) instead of “ ” (ASCII 180).
20. User-specified pattern found.
21. This command might not be intended.
22. *Comment displayed.*
23. Either “\,” or “\,” will look better.
24. Delete this space to maintain correct page references.
25. *You might wish to put this between a pair of “{ }”.*
26. You ought to remove spaces in front of punctuation.
27. Could not execute L^AT_EX command.
28. Don’t use \ / in front of small punctuation.
29. $\$ \times \$$ may look prettier here.
30. *Multiple spaces detected in output.*
31. This text may be ignored.
32. Use “ to begin quotation, not ’.
33. Use ’ to end quotation, not “.
34. Don’t mix quotes.
35. You should perhaps use “cmd” instead.
36. You should put a space in front of / after parenthesis.
37. You should avoid spaces in front of / after parenthesis.
38. *You should not use punctuation in front of / after quotes.*
39. Double space found.
40. You should put punctuation outside inner / inside display math mode.
41. You ought to not use primitive T_EX in L^AT_EX code.
42. You should remove spaces in front of “%s”
43. “%s” is normally not followed by “%c”.

7.2 L_YX でのバージョン管理

LARS BULLIK BJØNNES および PAVEL SANDA 文

7.2.1 はじめに

かつて私の友人が、L_YX をグループプロジェクトに使おうとしましたが、当時の L_YX には、バージョン管理機能やファイルロック機構が備わっていなかったために、採用見合わせました。この事実は少々私をいらだたせ、(将来的に CVS や SCCS のサポートできる拡張性を含め) 少なくとも RCS のサポートをするべきだと考えたのです。いまこれは実現しました。L_YX は、今ではもっとも基本的な RCS/CVS/SVN コマンドをサポートしています。もしもっと洗練されたことを行いたい場合には、ターミナル上から動で行う必要があります。

L_YX で実際にバージョン管理機能を使い始める前に、RCS/CVS/SVN の使用法に習っておかなくてはなりません。また、CVS のサポートは、subversion のサポートほどくはないので、CVS よりも SVN を使うことをお勧めします。Subversion を学び始めるは、SVN Book^{*4}が良いでしょう。RCS の場合は、「rsintro」をお読みください(これは man ファイルです。man rcsintro で読めます)。このファイルは、RCS の基本機能すべてを説明しています。特に、RCS ディレクトリに関するコメントと、マスター RCS ファイル(、v で終わるファイルです)の概念に注意を払ってください。

L_YX への実装においては、GNU RCS または CVS/SVN パッケージの最新版を仮定しています。旧版への保証はありません。ログメッセージのほとんどは、現在のところ操作後には表示されません。確認したい場合には、メッセージペインで確かめてください。

自製の外部コマンドを導入したい場合には、L_YX 関数説明書の vc コマンドを確認してください。

7.2.2 L_YX における RCS コマンド

以下の節では L_YX でサポートされている RCS コマンドを説明します。それらのコマンドはファイル▷バージョン管理のサブメニューにあります。L_YX は、RCS 5.7 に対して試験されています。

7.2.2.1 登録

文書がバージョン管理の下にない場合には、これがメニューにあらわれる唯一の。文書がバージョン管理下にあれば、項目登録は不可視になります。

このコマンドは、文書を RCS に登録します (CVS が管理しているディレクトリの場合)。すると対話的に、文書の最初の説明を入力するように尋ねられます。文書は読み込み専用モードになるので、変更を加えるためにはチェックアウトして編集する必要があります。バージョン管理下にある文書は、ミニバッファのファイル名に「[RCS:<バージョン> <ロック>]」という項目が表示されます。

実行される RCS コマンドは：`ci -q -u -i -t-"<初期説明文>" <ファイル名>`
スイッチについて知りたい場合は、`man ci` を読んでください。

7.2.2.2 変更をチェックインする

ファイルの編集が終わると、その変更をチェックインします。このとき、変更後の説明文を求められます。これは履歴ログに記録されます。バージョン番号は増え、変更がマスター RCS ファイルに適用され、文書がアンロックされて読み込みモードになります。

RCS コマンド：`ci -q -u -m"<説明文>" <ファイル名>`

7.2.2.3 チェックアウトして編集

これは、文書をロックして編集できるようにします。そしてあなたが文書を書きできるようにします。通常は、しばらくの間、文書を編集し、それが終わるとチェックインします。あなたがファイルをロックしたことを反映して、ステータスが変わります。

RCS コマンド：`co -q -l <ファイル名>`

7.2.2.4 リポジトリのバージョンに戻す

これは、最終チェックイン以降、文書に加えられた変更をすべて破棄します。変更される前に、警告が発せられます。

RCS コマンド：`co -f -u<バージョン> <ファイル名>`

7.2 LyX でのバージョン管理

RCS コマンド：`rcs -o<バージョン> <ファイル名>`

7.2.2.6 履歴を表示

RCS 文書の完全な履歴を表示します。ブラウザ中に、`rlog <ファイル名>`の出力が表示されます。詳細は `man rlog` をご覧ください。

7.2.2.7 改訂情報

LyX は、RCS バージョン番号情報（のみ）をサポートしています。詳細については 7.2.4.9を参照。

7.2.3 LyX における CVS コマンド

LyX は、現在 CVS を一部サポートしています。そのコマンドは、ファイル▷バージョン管理サブメニューにあります。バージョン管理システム SVN の方が強力なので、可能ならば CVS ではなく、SVN を使用してください。

7.2.3.1 CVS の一般的な利用法

CVS を初めて使いたい場合には、外部ツールを使用して自分のレポジトリを作成し、作業コピーをチェックアウトする必要があります。クライアント=サーバーモデルを使用している場合には、最初のレポジトリチェックアウトを行う前にログインする必要があります。かもしれません。

改訂管理下にある文書のレポジトリを他人も利用している場合に、同一文書の同一箇所に別々の変更を加えてしまうと問題が発生します。標準的な CVS レポジトリは、ファイルロック機構を有しません。これは驚くべきことかもしれませんが、コンフリクトは、文書の同じ箇所の特定の内容について、人々が意見を異にしている場合にのみ発生します。したがって、共著者同士がコンスタントにコミュニケーションを交わしていれば、このようなコンフリクトは滅多に発生しません。もしコミュニケーションがなければ、いずれにしても根本的な問題があるわけです。それにもかかわらず、いわゆる「予備付チェックアウト」を使用したいと思う人々もいます。この場合、最初にチェックアウトした際に、全ファイルの作業コピーは読み込み専用にされ、ユーザーは、作業コピーを書き込み可能にする特別なコマンドを使用した後、編集を開始することになります。変更

可能にしているとは限りません。

L_AT_EX は、あなたが予備付きチェックアウトを使用しているか、予備なしチェックアウトを使用しているか、推測しようと試みます。使用中の作業コピーが読み込み専用になっているか、あるいはそれが書き込み可能であって、CVS/Base サブディレクトリに文書のコピーが存在している場合には、予備付きチェックアウトを使用しているもよし、そうでない場合には予備なしチェックアウトを使用しているものと仮定します。予備付きチェックアウトが検出された場合には、作業コピーが読み込み専用の場合にのみ可能にするためには、チェックアウトメニューを使用しなくてはなりません。そのチェックインメニューが利用可能になるので、自分が加えた変更をリポジトリにアップロードした後、作業コピーを再び読み込み専用にする際には、これを使用します。

CVS の詳細な情報は、<http://www.nongnu.org/cvs>と<http://ximbiot.com>にあります。

以下に言及するサブコマンドとスイッチを理解するには、`man cvs` をお読みください。

7.2.3.2 登録

お使いの文書が改訂管理下に置かれていなければ、メニューにはこの項目だけが表示されません。もし改訂管理が有効ならば、登録項目は表示されません。

本コマンドは、既に文書ディレクトリを CVS 管理下に置いている場合にのみ（つまり、`CVS/Entries` ファイルが存在する場合にのみ）、お使いの文書を CVS に登録します。つまり、外部ツールを使用して、アーカイブを自分で作成しないしチェックアウトしませんが、これは必ずしも必要ではありません（このステップを忘れてしまうと、L_AT_EX は文書をローカルにのみ登録します）。

すると、文書の最初の説明文を入力するように促されます。登録したファイルはチェックインされたわけではないことに注意してください。

実行される CVS コマンドは、`cvs -q add -m"<入力した説明文>" "<ファイル名>"` です。

上記および他のすべての CVS コマンドにおける"<ファイル名>"は、「カレントディレクトリをファイルの場所に移動し、パスコンポーネントを除いたファイル名として使用する」という意味になります。

7.2.3.3 変更のチェックイン

7.2 L^AT_EX でのバージョン管理

CVS コマンド：

```
cvsv -q commit -m"<入力した説明文>" "<ファイル名>"あるいは  
cvsv -q unedit "<ファイル名>"
```

7.2.3.4 変更のチェックアウト

レポジトリを他人と共有している場合、他人の加えた変更を自らの作業コピーに取り
まなくてはならないことがあります。

CVS コマンド：`cvsv -q update "<ファイル名>"`

読み込み専用チェックアウトが検出された場合、作業コピーを書き込み可能にし予備
作成します。

CVS コマンド：`cvsv -q edit "<ファイル名>"`

7.2.3.5 レポジトリのバージョンに戻す

これは、最後のチェックイン以降、文書に加えられた変更をすべて廃棄します。変更
廃棄される前に警告が表示されます。最初にファイルが削除され、次に CVS 更新コマ
ンドが実行されます。

CVS コマンド：`cvsv -q update "<ファイル名>"`

予備付きチェックアウトが検出され、作業コピーに変更がない場合、予備のみが取り
されます。

CVS コマンド：`cvsv -q unedit "<ファイル名>"`

7.2.3.6 ローカルディレクトリチェックアウトのレポジトリからの更新

子文書や図を含んだり、外部 `.tex` ファイルを取り込むなど、文書が複雑になるにつれ
バージョン管理の利用は複雑になります。L^AT_EX は、文書が存在するツリー全体の更新を
サポートします。これは、CVS の使用法について詳しい知識のない人々や、レポジトリ
追加素材をコミットする野望のない人々と共同作業をするような場合に、特に便利です。
ファイル構造は、すべての外部ファイルが、文書と同じディレクトリかそのサブディレ
クトリに来るように組織する必要があります。いずれにしても、複合文書を新しいディレ
クトリを建てて収容するのは、望ましい習慣です。

コマンドレポジトリを使ってローカル辞書を更新は、ディレクトリ全体を更新します。

て表示されます。最初の部分は、文書名に続いて、更新操作前のバージョンが置きます。第 2 の部分は、>記号列の後にバージョン番号に続いて、レポジトリ版が置きます。

CVS コマンド：

```
cd $path; cvs diff ".( 変更が検出された場合には知らせる )
cd $path; cvs -q update ".
```

ここで\$path は文書パスを表します。

7.2.3.7 履歴を表示

これは、CVS 文書の全履歴を表示します。cvs log “<ファイル名>” の出力が、<ファイル名>に表示されます。

7.2.4 LyX における SVN コマンド

LyX は、現在 SVN を一部サポートしています。そのコマンドは、ファイル▷バージョン管理メニューにあります。レポジトリへのアクセスに、ssh 経由のパスワード保護を使用している場合には、端末ウィンドウでの入力を促されます。LyX は、SVN および 1.6 に対して試験されています*5。

7.2.4.1 登録

お使いの文書が改訂管理下に置かれていなければ、メニューにはこの項目だけが表示されます。もし改訂管理が有効ならば、登録項目は表示されません。

本コマンドは、既に文書のあるディレクトリを SVN 管理下に置いている場合（すなわち.svn/entries ファイルが存在する場合にのみ）お使いの文書を SVN します。つまり、アーカイブはご自分でチェックアウトしなくてはなりません。

すると、文書の最初の説明文を入力するように促されます。登録したファイルはコミットされたわけではないことに注意してください。

実行される SVN コマンドは、svn add -q “<ファイル名>” です。

スイッチの説明は man svn をお読みください。

7.2 L_YX でのバージョン管理

7.2.4.2 変更をチェックイン

ファイルの編集を終えたら、変更をコミットします。このとき、変更の説明文を入れよう促されます。その後、変更がコミットされます。

SVN コマンド^{*6} : `svn commit -q -m'<入力した説明文>' "<ファイル名>"`

7.2.4.3 チェックアウトして編集

このファイルの変更点をレポジトリから取得して更新します。すべての干渉点は、手で解決しなくてはならないので、この機能を使う前に SVN のマージ法と干渉解決法を解しておくようにしてください。

SVN コマンド^{*7} : `svn update -non-interactive "<ファイル名>"`

7.2.4.4 保存されたバージョンに戻す

これは、最後のチェックイン以降、文書に加えられた変更をすべて廃棄します。変更が廃棄される前に警告が表示されます。

SVN コマンド : `svn revert -q "<ファイル名>"`

7.2.4.5 レポジトリを使ったローカルディレクトリの同期^{*8}

上記の各コマンドはすべて、共通の短所を持っています。すなわち、これらのコマンドは現在の文書しか取り扱わないのです。もし、作業中の文書に図を含めたり、外部.tファイルを取り込んだりすれば、管理はずっと複雑になります。そこで L_YX は、文書においてあるツリー全体の更新をサポートします^{*9}。これは、subversion 管理を知らなかったりレポジトリに新しい素材をコミットする勇気のない人たちと共同作業をする羽目になったときに、ひじょうに役に立ちます。

ローカル辞書をレポジトリを使って更新コマンドは、ディレクトリ全体を更新し、マージに衝突が発生した場合には、ローカル版は保存されるので、意図しないデータロスが発生しません。ローカルな変更が発見された場合には、更新が開始される前にユーザに警告が与えられます。

SVN コマンド :

`svn diff $path(変更が発見された場合には通知します)`

```
svn update -accept mine-full $path
```

ここで\$path は文書のパスを示します。

7.2.4.6 履歴を表示

これは、SVN 文書の全履歴を表示します。svn log “<ファイル名>” の出力が、`svn log` コマンドの出力に一致して表示されます。

7.2.4.7 ファイルロック

別々の改訂管理システム間でファイルを交換しようとする、二人のユーザが同一（の一部）を編集している場合、マージが衝突する問題を引き起こします。その衝突が発生した場合、手動で解決しなくてはなりません。その合理的な解決策としては、与えられた時間に一人のユーザだけがファイルを編集することが許されることを保証するための、何らかのロック機構を導入することが考えられます。

SVN には、このように相互に排他的なファイルアクセスを提供する機構が二つあります。ロックと、svn:needs-locksvn ファイル特性に依拠した書き込み権限の自己管理（第 7.2.4.8 節を参照）です^{*10}。ある文書について、この特性が検出されたときには自動的に文書編集に SVN ロックを使用するようになり、チェックイン・チェックアウト機構は RCS と同じ枠組みに切り替わります。これは特に、LyX でのファイルの管理に以下の二つのモードがあることを意味します。

- 非ロック状態。読み込まれたファイルは、読み込み専用モードになります。この状態では、チェックアウトする必要があります。チェックアウトを行うと、リポジトリからの更新と、書き込みロックの獲得が行われます。ロックの獲得が失敗すると、非ロック状態から出ることができません。
- ロック状態。読み込まれたファイルは、「通常の」編集モードになります。他ユーザには、このファイルを編集する許可が与えられません。チェックインを行えば、変更のコミットと書き込みロックの解放が行われます。文書に変更が加えなかった場合には、コミットは実行されず^{*11}、書き込みロックの解放のみが行われます。

SVN コマンド：

7.2 LyX でのバージョン管理

```
svn unlock “<ファイル名>”
```

チェックアウト： `svn update “<ファイル名>”`

```
svn unlock “<ファイル名>”
```

7.2.4.8 自動ロック特性

上述したような `.lyx` ファイルに対する書き込み権限の自動設定は、ファイル▷バージョン管理▷ロック特性を使用で行うことができます。このコマンドは、SVN サーバファイルがロックされていないときのみ使用可能状態になります（つまり、使用するにチェックアウトが必要です）。

SVN コマンド：

チェックイン： `svn propset svn:needs-lock ON “<ファイル名>”`

チェックアウト： `svn propdel svn:needs-lock “<ファイル名>”`

7.2.4.9 文書の改訂情報

文書中で改訂情報を有効化するにはいくつかの方法があります。

- LyX による直接サポート

- ツリー改訂情報 (`vcs-tree-revision`) `svnversion` コマンドの出力を返します。下記の表が、結果の読み方のヒントを与えるでしょう。

出力	意味
4123:4168	改訂版の混合した作業コピー
4168M	修正された作業コピー
4123S	スイッチされた作業コピー
4123P	チェックアウト頻度が低かったことによる不完全な作業コピー
4123:4168MS	改訂版が混合し、修正され、スイッチされた作業コピー

- ファイル改訂情報。 `svn info -xml file.lyx` の出力を解析したものが返れます。サポートされるフラグは

- * 最終コミットのバージョン番号 (`vcs-revision`)

- * 最終コミットの著者 (`vcs-author`)

使用中文書の `.svn` ディレクトリがあるときのみ) 利用可能です。

- もう一つの—ハッカー的な—方法は、SVN キーワードを使用する方法です*。簡単に言えば、ファイルキーワード特性を設定して (例: `svn propset svn:keywords 'Rev' file.lyx`) 文書中にキーワード \TeX コード^{*13} タグ (例: *Rev*) を貼り付けます。このようにして、SVN クライアントは、各更新やコミットの後に自己改訂番号を代入することができるのです (例 *Rev : 59*)。このアプローチにつき問題があります。第一に、 \TeX の世界では「\$」文字は数式を表すになるので、LyX 中の *Rev* という数式表現が、すべて *Rev : 59* になってしまいます。その他の `Id`・`Date`・`Author` などといったキーワードも同様です。第二に、出力は使用中のロケールに依存するので、`Date` が用いられると、SVN は、引き起こすような文字列を容易に生成してしまいます。第三に、文書中にはだけでなく「*Rev:59*」という文字列全体が表示されてしまいます。Subversion ユーザ定義のキーワードを実装するまでは、このアプローチを信頼性を持つしたり、LyX がこれを直接サポートすることは、難しそうです。

7.2.5 SVN と Windows 環境

ユーザが、かなり早いうちに自分自身でコマンドライン操作を理解できないければ、TortoiseSVN を使った方が賢明でしょう。—P. A. ルービン

7.2.5.1 準備

LyX を導入して、Subversion レポジトリへのアクセス権を持っている他に、Subversion クライアントプログラムを導入する必要があります。クライアントプログラムの Windows 用インストーラは [CollabNet](#) にあります。ユーザは [TortoiseSVN](#) することを考えるでしょう。これは Subversion の操作を Windows Explorer のコンテキスト (右クリック) メニューに統合します。LyX 外部で行う操作は、一般的に `Ex` コンテキストメニューを使った方が便利です。TortoiseSVN は、LyX 自身が使用クライアントプログラムの代わりとは成りませんので、注意してください。

7.2.5.2 文書を Subversion 管理下に置くには

7.2 L^yX でのバージョン管理

トに追加する際には、新規文書が置かれることになるディレクトリへ、プロジェクトチェックアウトすることによって実現します。プロジェクト自体がバージョン管理下の場合（たとえばこの文書が新規プロジェクトの最初のファイルである場合）には、このディレクトリをレポジトリにインポートしなくてはなりません。これは、L^yX の外部で行います。インポートもチェックアウトも共に、TortoiseSVN を使えば、Explorer のコンテキストメニューから簡単に終わることができます。あるいは、DOS プロンプトでマンドライン・クライアントを使って成し遂げることもできます。TortoiseSVN を使ったプロジェクトのインポート手続きを下記で説明します。レポジトリは既に存在し、新規プロジェクトを C:\new project で開始するものとします。Subversion クライアントプログラムを使うための情報を得るには、DOS シェルで `svn -help` を実行してください。

1. Windows Explorer で C:\new project を見つけ出して右クリックし、TortoiseSVN▷Repo-browser を選択してください。必要であれば、レポジトリの URL を調整し、OK をクリックしてください。
2. 新規プロジェクトフォルダを置くレポジトリレベル（多くの場合最上位レベル）を右クリックし、Create folder... をクリックしてください。プロジェクトフォルダの名称を入力して OK をクリックしてください。必要であればログファイル用のメッセージを入力し、再び OK をクリックしてください。新規プロジェクトフォルダがレポジトリに現れるはずです。最後に OK を再度クリックして、レポジトリブラウザを終了してください。
3. 再度 C:\new project を右クリックし、今度は SVN Checkout... を選択してください。レポジトリにたった今作成した、プロジェクトフォルダの URL を選択し、チェックアウトディレクトリを C:\new project に設定してから、OK をクリックしてください。フォルダが空でないことを警告されますが、OK をクリックし進んでください。すると、C:\new project の下に .svn ディレクトリが作成されます。
4. L^yX で文書を新規作成するか開くかして、ファイル▷バージョン管理▷登録をクリックします。ログメッセージを入力して OK をクリックし、文書をバージョン管理にコミットしてください。

爾後、ファイル▷バージョン管理メニューのすべての機能が使用できるようになります。Windows Explorer の TortoiseSVN コンテキストメニューか、コマンドプロンプトで

Linux に較べて、Windows で SVN クライアントを SSH 経由で通信するようにするのは、かなり厄介な仕事です。どのようにクライアント側を設定すればよいか、少しも幾分かのヒントをお示しするつもりですが、ssh と Windows コマンドラインの前もって必要であり、それと共にかなりの量のフラストレーションが溜まることをおいてください。

1. 前節で述べたように、Windows 用の SVN クライアントを入手してください。が新規の導入である場合には、何らかの SVN コマンド（たとえば `svn -version`）を実行して、`config` ファイル群を生成させてください。これらは後で変更を必要があります。
2. Windows 用の `ssh` クライアントを選んでください。選択肢はたくさんありますがここでは Putty tools^{*14} のものを使用することとしましょう。クライアントの度にパスワードを訊いてこないように、接続を自動化する必要があります。するために、追加的なパスワード保護などを加えないキーのみを使用します。
 - (a) `puttygen` を使ってキーを生成します。`.ppk` プライベートキーファイルは保存し、パブリックキーファイルはサーバ側に置きます。SVN サーバが Linux で走っている場合、このパブリックキーの書式は Linux の `openssh` の形式がないために、「Public key for pasting into OpenSSH authorized file:」編集フィールドから、直接サーバの `~/.ssh/authorized_keys` または `authorized_keys2` ファイルにコピー＆ペーストを行わなくてはならぬことに注意してください。
 - (b) Putty の `plink` を入手してください。SVN の `config` ファイルの `[tunnels]` セクションに `ssh` コマンドを設定してください。

```
ssh=c:/path/plink.exe -i c:/path/private_key.ppk*16
```
3. SVN アーカイブをチェックアウトしてください。例:`svn co svn+ssh://u`

*14 <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

7.2 L_YX でのバージョン管理

7.2.5.4 行末変換

共著者同士が Linux と Windows を混在させているとき、L_YX は .lyx ファイル中での行末記号を使用することになります。これは、L_YX の機能に関する限り問題になりませんが、コミット差分は巨大になり、マージではコンフリクトを引き起こしがちです。幸いにして、SVN は、.lyx ファイルを native モードに切り替える際^{*17}、CR/LF 間をどのように取り扱えばいいかを知っているので大丈夫です^{*18}。

7.2.6 さらに調整

vc コマンド機構が最近になって付け加えられたおかげで、L_YX のパワーユーザは、訂管理のための自製コマンドを作ることができるようになりました。

たとえば、以下のようにして、二つの TortoiseSVN コマンドを直接統合することができます。

```
コミット:  vc-command DR "." "TortoiseProc /command:commit /path:$p"
元に戻す:  vc-command DR "." "TortoiseProc /command:revert /path:$p"
```

7.2.7 バージョン管理と文書比較

バージョン管理は、通常 diff ダンプを生成することによって、改訂版の間の変更点検査するのが、典型的な使用法です。これは平文ファイルには有効なのですが、複雑な造を持つ L_YX ファイルの場合には、あまり便利ではありません。そこで、L_YX は独自文書比較機能を提供しています。この機能呼び出すには 2 つの方法があります。1 は、L_YX の VC_COMPARE 関数を直接呼び出す方法であり（詳細は L_YX 関数説明書を参照）、もう 1 つは旧改訂と比較...メニューないしアイコンを使用する方法です。当該文書のうち、任意の 2 つの改訂版を比較することもできますし、編集集中の現行版を古い改訂と比較することもできます（このとき改訂版遡及数「0」は、編集集中のファイルと最後コミットした改訂版との比較を意味します）。

この機能は、SVN と RCS でサポートされていますが、RCS は、より複雑なバージョン付けスキームを持っているため、制約があります。ダイアログ中で改訂版を指定するの数字は、つねに RCS 改訂番号の最後の数を指し示します（例：「1.2.25」版のとき

7.3 文芸的プログラミング

KAYVAN SYLVAN (kayvan@sylvan.com) 改訂 EDMAR WIENSKOSKI JR. (edmar-w-jr@technologist.com) 原文

7.3.1 はじめに

この節の主な目的は、文芸的プログラミング (literate programming) のために何をどう使えばよいかをお見せすることです。ここでは、あなたがこのプログラミングをよくご存じで、「tangling」や「weaving」の意味するところを知っているものと明します。そうでない場合には、以下の各節にあるウェブリンクをたどってください。ここには、かつての開発の歴史から最新のツールチップまで、たくさんの良質なドキュメンテーションがあります。

また、あなたが、L_YX の設定を変更することを厭わない程度まで L_YX に習熟して L_YX リソースファイルにも親しんでいることを前提にしています。そうでない場合は、この必要とするトピックを扱っている、L_YX の他の説明書で調べてください。

7.3.2 文芸的プログラミング

以下は、文芸的プログラミング FAQ からの抜粋です。

文芸的プログラミングは、人間が読むのに適した形でのドキュメンテーションとソースの組み合わせです。実際、文芸的プログラムは楽しい読み物であるべきであり、さらには参加したいとさえ思わせるべきなのです！（ボブ、ごめん。言はいれなかった。）一般的に、文芸的プログラムは、ソースとドキュメンテーションを同一ファイル上に結合したものです。そして、文芸的プログラミングは、ファイルを解析して、読むことのできるドキュメンテーションか、あるいはコンパイル可能なソースを生成します。文芸的プログラミングの WEB スタイルは D. E. Knuth が、T_EX 組版ソフトウェアを開発中に作り出したものです。

他のエキスパートの発言です。

7.3 文芸的プログラミング

- 特に図式と数式を含んだドキュメンテーションの組版などがあります。

あなたの興味に火を付けたところで参考文献を見てみましょう。

7.3.2.1 参考文献

完全な文芸的プログラミング FAQ は

Literate Programming FAQ <http://shelob.ce.ttu.edu/daves/lpfaq/faq>

にあります。

この FAQ には、23 (二十三です!) の文芸的プログラミング用ツールが列挙されています。一部のツールは、特定のプログラミング言語用に特化したものですが、一般的な象を持つものもあります。私は自分で使うのに、いくつかの理由から NOWEB を選びました。

- ドキュメンテーションを \LaTeX でも HTML でも出力できる。
- 開放的なアーキテクチャに持っているので、新しいフィルタをプラグインとしてえたり、自分が必要な特別な処理を実行することが容易である。
- すでに利用可能となっているフィルタの良質なコレクションがある (HTML はの一つです)。
- 無料である。

Noweb のウェブページは

Noweb ホームページ <http://www.cs.virginia.edu/~nr/noweb/>

にあります。ここから始めれば、たくさんの興味深いリンクや、文芸的プログラミング用例にたどり着くことができます。

7.3.3 \LaTeX と文芸的プログラミング

\LaTeX における文芸的プログラミングのサポートは、 \LaTeX の汎用変換機構を使用して現しています。このサポートは「Noweb 非依存」的な方法で提供されているので、 \LaTeX の設定を変更するだけで、自分の好きな他の文芸的プログラミングツールを使って、

7.3.3.1 ドキュメントとコードの生成 (weaving および tangling)

文書クラスの選択 Noweb と LyX の導入に成功すると、新規文書を開いたり、書の文書クラスを変更しようとする際、以下の 3 つの新しい文書クラスが利用可能になっているはずだ。

- Article (Noweb)
- Book (Noweb)
- Report (Noweb)

文芸的プログラミング文書を作成する場合には、これらのうち一つを選んでください。

文芸的プログラミング文書は、これら 3 つのクラスに限定されないことに注意ください。letter のような他のスタイルから新しいクラスを作成することもできます。Article (AMS) のような他のクラスバリエーションと組み合わせて作成することもできます。既存のクラスではカバーできないような、特別なニーズがある場合には、LyX 用メーリングリスト (lyx-devel@lists.lyx.org) に連絡していただければ、新しいクラスを加えるように調整するか、そのしかたをお教えます^{*19}。また、Noweb 以外の文芸的プログラミングツールを使用する場合にも、そのための新しい文書クラスをつくる必要があります。

コードの入力 LyX では、SCRAP^{*20} という名前のレイアウトを使えば、コードを書くことができます。Noweb は次のように scrap を区切ります。

```
<<My scrap>>=
  code
  more code
  even more code
@
```

問題は<<と@の間にあるものを完全にそのままの形で扱わなければならないこと、つまり LyX はこの間に書かれたものへのどんな変更も禁止しなくてはなりません。これは Scrap と呼ぶ特別なレイアウトで扱われます。この環境は通常の段落のようであり、自由に空白を入れることができます。

7.3 文芸的プログラミング

Scrap 段落環境の欠点は、連続したコードの段落の間に空行がソースコードにも印刷したドキュメントにも入ることです。これを回避するためには、Scrap 環境の中の改行に (ctrl-return) を使ってください。上の例では次のようにします^{*21}。

```
<<My scrap>>=  
code  
more code  
even more code  
@
```

この環境はうまく働きます。不便な点は普通のリターンのかわりに ctrl-return を使わなければならないことです^{*22}。

また、Scrap の中に Noweb の %def を使うことができます。これを使うことで Noweb の identifier の相互参照に項目を加えることができます。次の例を見てください。

```
<<My scrap>>=  
def some_function(args):  
    "This is the doc string for this function."  
    print "My args: ", args  
@ %def some_function
```

これをどのように使って、その結果相互参照がどうなるかは *LIBDIR/examples/listerro* の中の文芸 python プログラムを見るとよくわかります。

ドキュメンテーションの生成 新しい文章ファイルを正しい文書クラスで作り、その中にコードとテキストも入れました。これをどうやって印刷したらいいのでしょうか。簡単です。表示▷ DVI 等を選ぶだけです。普通の文章の場合と同じで、特別なことは必要ありません。

LyX の内部で何が起っているかを説明しておきましょう。

1. 表示▷ DVI が選ばれると、L^AT_EX ファイルが作られます。

ドキュメントが文芸プログラミング用のクラスであった場合には、拡張子に (設定パネルで設定した) 文芸プログラム用のものがつけられます。それ以外は .tex が拡張子に使われます

りません。Scrap 環境でコードを入力するのだけがポイントです。

3. ドキュメントが文芸プログラムクラスなら、LyX は内部の LyX から Noweb コンバータを使い、それから Noweb から L^AT_EX のコンバータ^{*23}を使って L^AT_EX を生成します。

それ以外の場合はこのステップを飛ばします。

4. 最後に L^AT_EX が実行され、普通の文章と同じ後処理が行われます。

様々なコンバータから走らせるコマンドを変えることによって、特定の文芸プログラムのソフトウェアにしばらくならないようになっています。

コードの生成 ビルドメニューが選ばれた場合や、それに対応するツールバーンが押された場合、上のステップ 1 と同様に L^AT_EX ファイルが生成されます。その LyX は Noweb → Program のコンバータを起動します。通常は、このコンバータはコンバータと同じように 2 つの部分からできています。

1. コンバータプログラム。このプログラムは、一つのフォーマットから他のフォーマットへの変換を行います（この場合は、Noweb フォーマットからプログラム疑似フォーマットへ変換します）。
2. エラーログパーサー。これはエラーメッセージを LyX が理解できる形へ書きます。これによって、LyX はファイルバッファの正しい位置へエラーメッセージを配置することができます。

はじめのパートのために、「変換子」設定で、“build-script*\$i*”と指定します。基本的に、LyX が build-script（プログラムかスクリプト名）を（一般には LyX 時ファイル用のディレクトリにつくられる）Noweb ファイルの名前とともに起動することを意味しています。

以下に build-script の一例を書いておきます。パスが通っているディレクトリに使用します。

```
#!/bin/sh
#
notangle -Rbuild-script $1 | env NOWEB_SOURCE=$1 sh
```

コンバータの設定の次の部分は Flags です。これを“originaldir, parselog=li

7.3 文芸的プログラミング

コンバータコードが `listerrors` プログラムを探す順序は、はじめに `MYXDIR/scripts`、次に `LIBDIR/scripts`、最後にパスです。

文章中でのプログラムの構築操作の説明 `LyX` と `Noweb` の統合の最後の部分は “build-script” scrap です。一般に、あなたのプログラムの構築操作の説明それ自身 scrap に入れて文章中に含めなければなりません。上の例の `noweb` 用の build-script は、scrap を見つけるのに `notangle` コマンドを使い、`sh` を使って起動します。

以下に典型的な scrap がどのようなものかを書いておきます。

```
<<build-script>>=
#!/bin/sh

if [ -z "${NOWEB_SOURCE}" ]
then
    NOWEB_SOURCE=myfile.nw
fi

[... code to extract files ...]
[... code to compile files ...]
@
```

`LIBDIR/examples/listerrors.lyx` と `LIBDIR/examples/Literate.lyx` を見てください。この2つのファイルには異なる実装の `listerrors` プログラムが含まれていて、これまで明してきた各々の部分が全体としてどのようになるかをわかってもらうための例です。また、`LIBDIR/examples/noweb2lyx.lyx` も見てください。面白いことに、これらの3つファイルは、`Python`、`C`、`Perl` が各々使われていて、`LyX` の文芸プログラミングのサポートがプログラム言語に依存しないことを示しています。

7.3.3.2 `LyX` の設定

すべての文芸プログラムのための設定は、ツール▷設定パネルの「変換子」タブですることができます。重要な部分を見ておきましょう。

the “literate” format 書式タブで設定します。ここで、`Noweb` 特有の設定を行います。

`GUI` 名を `Noweb` に、拡張子を `nw` に設定してください。これによって、`LyX` は

マットからの変換を行えます)。

NoWeb->L^AT_EX このコンバータは、weaving の段階を行います。Noweb の場

“noweave -delay -index \$\$i > \$\$o” に設定してください。

NoWeb->Program これは、tangling を行います。以前に説明したよ

build-script \$\$i と設定して、Flags に“originaldir,parselog=liste
を入れてください。

7.3.3.3 デバッグ機能の拡張

ddd/gdb または他のデバッガで使える server-goto-file-row という関数が新たに L^AT_EX サーバに実装されました。

ddd/gdb でデバッグしているとき、一つのキーストロークで現在実行しているテキストエディタを起動できます。既定の ddd の設定では Shift-Ctrl-V になっています。この設定は ddd でツール▷設定▷ファイル書式ウインドウを表示して、編集プロダクトリを変えると変えることができます。

新しい L^AT_EX サーバの関数と、この ddd の機能を使うために編集プログラムを次に設定してください。

```
echo "LYXCMD:monitor:server-goto-file-row:@FILE@ @LINE@" >~/
```

こうしておくで ddd を使っていてプログラムの編集したい場所があったとき、(ddd のウインドウで) Shift-Ctrl-V を押すだけで ddd は L^AT_EX サーバに情報を送り、L^AT_EX ファイルを表示し、カーソルは ddd が示した場所へあらわれます。これでもうデバッグの最中に推測しながらスクロールして場所を探す必要がなくなります。

この機能が動くには L^AT_EX サーバが利用可能でなければなりません (既定では L^AT_EX サーバは使えないようになっています)。L^AT_EX サーバを使うようにするには、設定(ddd の)の L^AT_EX サーバパイプに “/home/<your-home-directory>/<your-home-directory>/.lyx/lyxpipe” 風にパスを指定してください。

詳しくは『取扱説明書カスタマイズ篇』の L^AT_EX サーバの説明を読んでください。

7.3.3.4 ツールバーの拡張

7.3 文芸的プログラミング

L^AT_EX はツールバーで使うためのいくつかのボタンを用意しています。私のツールバーはこの 6 つのショートカットに表示▷更新▷DVI と表示▷DVI を組み合わせたものになっています。

```
Toolbar
Layouts
Icon "layout Standard"
Icon "layout Section"
Icon "layout LATEX"
Icon "layout LATEX-Code"
Icon "layout Scrap"
Separator
Icon "buffer-view"
Icon "buffer-typeset"
Icon "build-program"
Separator
.
.
.
End
```

7.3.3.5 色を変える

L^AT_EX ではいくつかの色を設定で変えることができます。まず変えたいと思うのは L^AT_EX フォントカラーでしょう。既定では赤になっていますが、Scrap は L^AT_EX フォントをい、文芸的プログラミングでは多くの Scrap があるためにそれらがすべて赤では疲れてまいります。操作性▷色タブでこの色を変えることができます。

次にすることは画面の改行文字を見えなくすることです。改行文字の色を背景の色と同じにすることで改行文字を見えなくできます。私は改行文字を背景の色に似ているが少し違う色にすることを勧めます。こうすると、改行文字があるのはわかりますが、じゃまならないでしょう。

第 8 章

L^AT_EX の達人の秘密

L^AX は強力な道具ですが、T_EX/L^AT_EX でできることをすべてサポートしているわけではありません。しかし、ツールバーの T_EX ボタンを押したり、L^AT_EX プリアンブルに追加するのをためらわなければ、多くの T_EX や L^AT_EX のトリックを使うことができます。この節ではそのようなトリック、ヒント、アイデアを紹介します。まず家で、あまり重要でない文章で試してください。自分の博士論文でまず試そうとは思えない方がいいでしょう。

この節で紹介するものの多くは、通常の L^AT_EX ディストリビューションにはあまり入っていないファイルを使います。teT_EX を使っている場合には、ほとんどのファイルは含まれていますが、それでもいくつかは近くの CTAN アーカイブからダウンロードする必要があります。しばしば、同じことをするのにいくつかの方法があったり、同じことをするいくつかの L^AT_EX ファイルがあります。ここで紹介しているのは、いくつかの方法をすべてその中で我々が推薦するものではなく、単にあることをある特定のファイルでできるということの紹介です。

8.1 多段組み

LARS GULLIK BJØNNES 文

8.1.1 目的

この章^{*1}の目的は、L_YX の文章でどのように L^AT_EX パッケージ multicol を使
ができるのかを説明することです。L_YX は multicol パッケージをサポートして
るので、ちょっとしたトリックが必要です。この節を読むとどのようなトリックが明
なるはずです。

8.1.2 制限事項

multicol パッケージでは同じページ内で一段組みと多段組みを切り替えること
ます。脚注も（ほとんどの場合）正しく扱われますが、カラムの下ではなくページ
おかれます。しかし、L^AT_EX のフロートメカニズムは現在の実装では部分的に使う
できなくなっています。今のところこの環境の範囲では、ページ幅のフロートしか
せん。

8.1.3 例

8.1.3.1 二段組み

二段組みを使いたい場合には、二段組みのレイアウトを始めたい場所に L^AT_EX
ドで `\begin{multicols}{2}` を挿入してください。そして二段組みを終わりたい
`\end{multicols}` を入れます。次の例を見てください。

The Adventure of the Empty House
by SIR ARTHUR CONAN DOYLE

It was in the spring of the year 1894 that all London was interested, and the fashionable world dismayed, by the murder of the Honourable Ronald Adair under most unusual and inexplicable circumstances. The public has already learned those particulars of the crime which came out in the police investigation, but a good deal was suppressed upon that occasion, since the case for the

that it was not necessary to bring forward all the facts. Only now, at the end of ten years, am I allowed to supply those missing links which make up the whole remarkable chain. The crime was of interest in itself, but that interest was almost nothing to me compared to the inconceivable sequel, which afforded me the greatest joy and surprise of any event in my adventurous life. Even now, after this long interval, I find myself thrilling as I think

8.1 多段組み

joy, amazement, and incredulity which utterly submerged my mind. Let me say to that public, which has shown some interest in those glimpses which I have occasionally given them of the thoughts and actions of a very remarkable man, that they are not to

blame me if I have not shared my knowledge with them, for I should have considered my first duty to do so, had I not been barred by a positive prohibition from his own lips which was only withdrawn upon the third of last month.

8.1.3.2 多段組み

二段以上の多段組みをする場合も同じようにできます。

It can be imagined that my close intimacy with Sherlock Holmes had interested me deeply in crime, and that after his disappearance I never failed to read with care the various problems which came before the public. And I even attempted, more than once, for my own private satisfaction, to employ his methods in their solution, though with indifferent success. There was none, however, which appealed to me like this tragedy

of Ronald Adair. As I read the evidence at the inquest, which led up to a verdict of willful murder against some person or persons unknown, I realized more clearly than I had ever done the loss which the community had sustained by the death of Sherlock Holmes. There were points about this strange business which would, I was sure, have specially appealed to him, and the efforts of the police would have been supplemented, or more

probably anticipated, by the trained observation and the alert mind of the first criminal agent in Europe. All day as I drove upon my rounds I turned over the case in my mind and found no explanation which appeared to me to be adequate. At the risk of telling a twice-told tale, I will recapitulate the facts as they were known to the public and the conclusion of the inquest.

三段組み以上も可能ですが、読みにくくなり実用的ではないでしょう。

8.1.3.3 カラムの中のカラム

カラムの中にカラムを入れることもできます。

The Honourable Ronald Adair was the second son of the Earl of Maynooth, at that time governor of one of the Australian colonies. Adair's mother had returned from Australia to undergo the operation for cataract, and she, her son Ronald, and her daughter Hilda were living together at 427 Park Lane.

The youth moved in the best society—had, so far as was known, no enemies and no particular vices. He had been engaged to Miss

it. For the rest {s the man's life moved in a narrow and conventional circle, his habits were quiet and his nature unemotional. Yet it

Ronald Adair was fond of cards—playing continually, but never for such stakes as would hurt him. He was a member of the Baldwin, the Cavendish, and the Bagatelle card clubs. It was shown that, after dinner on the day of his death, he had played a rubber of whist at the latter club. He had also played there in the afternoon. The evidence of those who had played with him— Mr. Murray, Sir John Hardy, and Colonel Moran—showed that the game was whist, and that there was a fairly equal fall of the cards. Adair might have lost five pounds,

but not more. His fortune was a considerable one, and such a loss could not in any way affect him. He had played nearly every one club or other, but he was a cautious player and usually rose a winner. It came out in evidence that, in partnership with Colonel Moran, he had actually won as much as four hundred and twenty pounds in a sitting, some weeks before, from Godfrey Milner and Lord Basingham. So much for his recent history as it came out in the inquest.

より複雑な形でカラムの中にカラムを入れる例や、ヘッダのスペース、縦のカラムを区切る線の入力方などの例は、ファイル `examples/multicol.lyx` を読んでください。

8.2 箇条書き（連番）段落環境での番号振り

JOHN WEISS 文

既定での箇条書き（連番）段落環境での番号の付け方はアラビア数字で始まり、大文字のアルファベットで終わります。違う番号の付け方をしたい場合にはどうしたらいいのでしょうか。次の例を見てください。

```
\renewcommand{\labelenumi}{\Roman{enumi}.}
\renewcommand{\labelenumii}{\Alph{enumii}.}
\renewcommand{\labelenumiii}{\arabic{enumiii}.}
\renewcommand{\labelenumiv}{\alph{enumiv}.}
```

これは番号の付け方をまず大文字のローマ数字、次に大文字のアルファベット、次に小文字のアルファベットにします。

前の例にはもう少しおまけがついていて、番号の後に点が入ります。例えば初めのレベルは “I.” のようになります。読みやすいように、番号の付け方を説明する際には “I.”, “A.”, “1.”, “a.” のような表示方法を使うことにしましょう。

例を見て分かるように、各々のネストのレベルにラベルコマンド `\labelenumi`、`\labelenumii`、`\labelenumiii`、`\labelenumiv` とカウンタ `enumi` ... `enumiv` があります。使うことのできる番号

8.3 Dropped Capitals

```
\renewcommand{\labelenumi}{\#\Alph{enumi}\#}  
\renewcommand{\labelenumii}{\Alph{enumi}.\arabic{enumii}}  
\renewcommand{\labelenumiii}{\alph{enumiii}+}  
\renewcommand{\labelenumiv}{(\romannumeral{enumiv})}
```

この例は番号の付け方を <“#A#”, “A.1”, “a+”, “(i)”>のようにします。

8.3 Dropped Capitals

MIKE RESSLER 文

Those of you who like the style of old books probably also like “dropped capitals”—those large capital letters which begin each new chapter or section.

昔の本のスタイルが好きならこの例のように、dropped capitals も多分好きでしょう。これは新しい章、または節が大きな大文字で始まるものです。これを L_AT_EX/L^AT_EX で実装するのは (少しは plain T_EX を知る必要はありますが) 簡単ですが、この節の初めの T_EX モードを見てもらうと分かるように多くの仕事と繰り返しが必要になります。

`\bigdrop{-1em}{3}{ptmri}{T}` こちらの方が簡単でしょう。CTAN にある `dropcaps` (またはより新しい `dropping`) パッケージを使えば簡単にこのような文をあなたの文章に入れることができます。このパッケージは `teTEX` に含まれていないのでこの文章の中でお見せできませんが、この段落を新しい文章にコピーして、先頭 `\verb` と `+` を取り除き、プリアンプルに `\usepackage{dropcaps}` を加えると、Tim Roman Italic の T が 3 行文の高さで余白に 1em でた形で印刷されます (T_EX が探せディレクトリに `dropcaps.sty` があることを確認してください)。最初の引数は字下げ量です。この例では負ですから余白に突き出します。2 番目の引数は行数で表した文字高さです。3 番目の引数はフォント名です。tfm ファイルも持つ物なら何でも動きま (.../texmf/fonts/tfm ディレクトリで色々探してみてください)。私が気に入っている `yinit` で、これは dropped capital 用に特にデザインされたドイツ文字のフォントです。4 番目の引数が使う文字です。dropping パッケージは `\bigdrop` コマンド以外に少し簡単な `\dropping` コマンドも持っています。

8.4 標準的でない形の段落

う。そういう場合は
 楽しい T_EX コマンド
`\parshape` を使いまし
 ょう。この例で分かるよう
 に適当に行の長さを設定する
 ことで好きな形の段落が作れま
 す。馬鹿げているように見えるかも
 しれませんが、dropped capital を最終
 調整したり、長方形ではない図のま
 わりを文章で囲ったりするのに
 このような手作業が必要な
 です。

文 法 は `\parshape numlines #1indent #1length #2indent #2length #nindent #nlength` です。numlines は段落をつくる行数です。その段落の本文より短い場合はそこで終わり、長い場合には最後の行の定義がそれ以後も使われ、`#nindent` と `#nlength` エントリは左余白からの字下げの量と、字下げされたところの行の長さを指定します。この形は現在の段落にだけ適応されて、次の段落では通にリセットされます。

8.5 まとめ

見てきたように、この節には有用な例も馬鹿げた例も入っています。たとえば、節のように段落の形を変えることが必要な人がそれほどいるとは思えませんが、点は、T_EX や L^AT_EX でどのようにすればいいかを理解する気があれば、ほとんどを L_YX の中でできるということです。T_EX は、すばらしく強力な組版システムで、L_YX はそれをバックエンドとして使用しているので、そのすべての力があなたの利用可能なのです。Happy L_YXing!