

LyX の図表・フロート・注釈・ボックス・外部素材に関する詳細説明書

LyX プロジェクトチーム¹

バージョン 2.0.x

2011 年 5 月 15 日

¹コメントや間違いの指摘がありましたら、LyX 文書化メーリングリスト lyx-docs@lists.lyx.org までお送り下さい。

目次

第1章 図	1
1.1 画像ダイアログ	1
1.2 図フロート	3
1.3 画像書式	5
第2章 表	7
2.1 はじめに	7
2.2 表ダイアログ	7
2.3 表ツールバー	8
2.4 表編集メニュー	9
2.5 表フロート	10
2.6 長尺表	10
2.6.1 長尺表での脚注	13
2.6.2 長尺表の縦横揃え	13
2.6.3 長尺表のキャプション	14
2.6.3.1 長尺表への参照	14
2.6.3.2 キャプションの幅	15
2.6.3.3 表の各ページごとに異なるキャプション	16
2.7 長尺表にまつわる諸点	18
2.7.1 長尺表の計算	18
2.7.2 フロートと長尺表	18
2.7.3 強制改頁	19
2.8 複数行・連結列・連結行	22
2.8.1 表セル内の複数行	22
2.8.2 連結列	23
2.8.2.1 連結列の基礎	23


2.8.2.2	連結列の計算	23
2.8.3	連結行	25
2.9	フォーマル形式の表	25
2.10	表の垂直揃え	28
2.11	色付きの表	28
2.11.1	色付きのセル	28
2.11.2	色付き罫線	30
2.12	表の調整	31
2.12.1	行の余白取り	31
2.12.2	特殊なセル揃え	33
2.12.3	自製のセル書式および列書式	34
2.12.4	罫線幅	35
2.12.5	点罫	37
第3章	フロート	39
3.1	はじめに	39
3.2	フロート型	39
3.2.1	アルゴリズムフロート	40
3.2.2	折返しフロート	40
3.3	フロートの連番	41
3.4	フロートの参照	42
3.4.1	相互参照の書式	43
3.4.2	参照名の自動命名	44
3.4.3	参照位置	44
3.5	フロートの配置	45
3.6	回転フロート	47
3.7	従属フロート	47
3.8	横並びのフロート	47
3.9	キャプションの書式設定	47
3.10	キャプションの配置	51
3.11	フロート一覧	52

第4章	注釈	55
4.1	LyX 注釈	55
4.2	脚注	56
4.2.1	脚注の付番	57
4.2.2	脚注の配置	58
4.3	傍注	60
第5章	ボックス	63
5.1	はじめに	63
5.2	ボックスダイアログ	63
5.2.1	寸法	63
5.2.2	文字揃え	64
5.2.3	装飾	66
5.3	ボックスの詳細な設定	67
5.4	ミニページ	68
5.5	Parbox	69
5.6	単語や文字のボックス	70
5.6.1	ハイフネーションの回避	70
5.6.2	垂直揃え	70
5.7	色付きボックス	71
5.7.1	文字の着色	71
5.7.2	段落の着色	72
5.8	ボックスの伸縮回転	73
5.8.1	ボックスの回転	73
5.8.2	ボックスの伸縮	74
第6章	外部文書構成要素	77
6.1	外部素材	77
6.2	子文書	78
6.2.1	外部文書の第1小節	80
6.2.2	外部文書の第2小節	81
第7章	プログラムコードリスト	83

付録 A	$\text{L}_\text{Y}\text{X}$ で使用できる単位	87
付録 B	画像を含む場合のファイル出力形式	89
B.1	DVI	89
B.2	PostScript	89
B.3	PDF	90
付録 C	第 (2.1) 式の説明	91

第 1 章 図

1.1 画像ダイアログ

画像を文書に挿入するには、文中の挿入箇所にカーソルを置いて、ツールバーアイコン  をクリックするか、挿入▷画像メニューを使用して下さい。すると、ダイアログが現れるので、読み込むファイルを選択して下さい。出力中での画像は、本文中の挿入位置に正確に出力されます。

画像ダイアログは、画像をクリックすることによって、いつでも呼び出すことができます。このダイアログには、以下の三つのタブがあります。

画像 ここでは、画像ファイルを選択し、その出力での表示のしかたを調整することができます。画像寸法で使用することのできる単位については付録 A に説明があります。回転角と回転原点を指定することによって、反時計廻り方向へ画像を回転することができます。画像は、 LyX 内でも回転して表示されます。

画像は、パーセント値を指定するか、幅と高さを明示的に設定することによって、伸縮させることができます。幅と高さのどちらか一方のみを指定した場合には、残りの寸法は自動的に決定されます。両方とも指定した場合には、指定した寸法に画像が変換されますので、変形してしまう可能性があります。画像が変形するのを防ぐには、縦横比を維持するオプションを指定して下さい。すると画像は、幅と高さが指定した寸法を越えないように伸縮されます。

画像を右クリックすると現れるコンテキストメニューで外部で編集という項目を選ぶことによって、画像を指定したプログラムで開くことができます。このプログラムは、 LyX 設定のファイル書式設定で、各画像書式ごとに設定することができます。

切り抜き 伸縮単位を使用する代わりに、出力における画像の縦横の長さを調整するのに、画像座標の設定をすることができます。座標は、ファイルから取得ボタンを押すことによって、自動的に計算することもできます。境界枠で切り抜くオプションは、指定された座標内の画像領域のみ印字します。通常は、画像座標を気にする必要はないので、このタブは無視して構いません。

\LaTeX と LyX のオプション このタブでは、 LyX 内での画像表示法を調整できる他、 \LaTeX のエキスパート向けに \LaTeX オプションを追加することができるようになっています。

下書きモードオプションを指定すると、画像は出力には現れず、画像と同じ大きさの枠だけが表示されます。

エクスポートする際に解凍しないオプションは、*x.eps.gz* のような、zip 圧縮された EPS 画像のみに有効です。このオプションが有効になっていると、 \LaTeX は zip 圧縮 EPS 画像をそのまま取り扱えるので、画像は解凍されません。

zip 圧縮 EPS 画像は、出力書式として PostScript を選択する場合には、ディスク領域を節約できるので便利です。付録 B.2 をご参照下さい。EPS 画像を zip 圧縮するには、UNIX シェルか Windows コンソールから以下のコマンドを実行して下さい。

```
gzip x.eps
```

```
zgrep %%Bounding x.eps.gz > x.eps.bb
```

二つめのコマンドは、 \LaTeX が圧縮画像を取り扱う際に必要な領域ボックスファイル「x.eps.bb」を生成します。

画像グループフィールドでは、画像設定グループを定義したり既存のグループに所属させたりすることができます。これらのグループ内の画像は、共通した設定を持つので、グループ内の一つの画像の設定をいじると、自動的にグループ内の全ての画像の設定が同じように変更されます。したがって、たとえば、たくさんの画像の寸法を変更したいときにも、一つ一つの寸法を手動で変更せずに済むのです。既存のグループに帰属させるには、画像のコンテキストメニューを使用して、望むグループ名をチェックすることで行うこともできます。

下記は EPS 書式¹の画像の例で、単独の中央揃えの段落に置いてあります。




下記は上と同じ画像ですが、下書きモードのもので、



47C__Program_Files__x86__LyX20_bin____Resources_doc_clipart_mobius.ep

¹画像書式は、第 1.3 節に説明されています。

1.2 図フロート

フロートについての一般的な説明は、第 3.1 節をご覧ください。

ツールバーボタンのが挿入▷フロート▷図メニューを用いると、「図#」(#は実際には数字です)というラベルのキャプションを持ったフロートが挿入されます。画像は、図 1.0 のようにキャプションの上に置くこともできますし、図 1.1 のようにキャプションの下に置くこともできます。キャプションの配置について詳しくは、第 3.10 節に述べられています。

第 1.0 図と第 1.1 図は、参照された図の例です。図は、本文中でラベルを参照することによって参照できます。これをするには、まず、挿入▷ラベルメニューかツールバーボタンを使用して、キャプションにラベルを挿入してください。その後、挿入▷相互参照メニューかツールバーボタンを使用して、ラベルを参照することができます。L^AT_EX は仕上文書でのフロートの位置を調整するので、「上図」のような曖昧な参照のしかたをするのではなく、フロートを直接参照することが重要です。仕上文書では、「上図」ではないことが、大いにありうるからです。

参照は、第 3.4 節で詳細に説明されています。

通常、図フロートには一つの画像だけを挿入しますが、場合によっては、二つの画像に別々の副キャプションをつけて挿入したいこともあるかもしれません。その場合は、既存の図フロートの中に図フロートを入れ込むことによって、実現することができます。ただし、図一覧にはフロートの主キャプションのみが表示されることに留意してください。図 1.2 は、二つの画像が横に並んでいる図フロートの例です。二つめの画像を一つめの下に持ってくることも可能です。図 1.2a と図 1.2b が内部図になります。

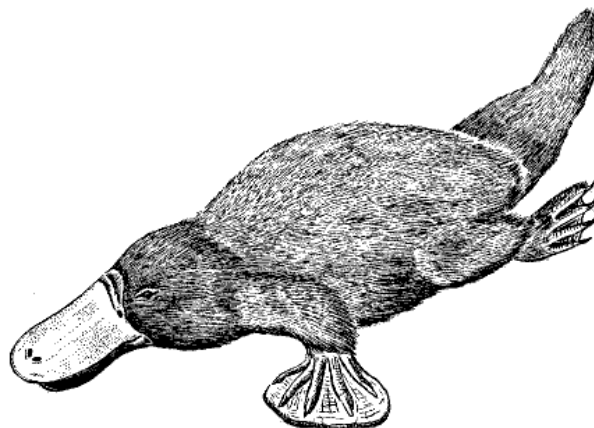
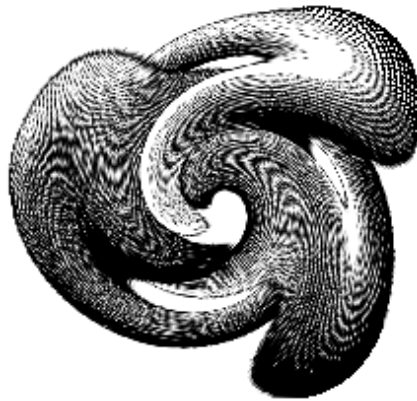
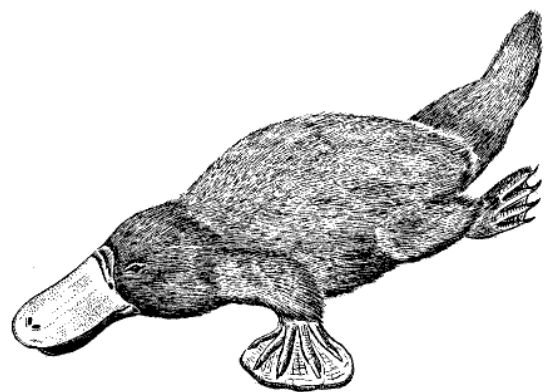


図 1.1: フロート中の激しく歪んだカモノハシ

図 1.2: 行ってしまった M.C. エッシャー



(a) 何かわからない構造



(b) カモノハシ

図 1.3: 歪められた二つの像。両画像とも、「歪曲」画像設定グループに属しています。

1.3 画像書式

画像は、既知のファイル形式であれば、どのような形式でも挿入することができます。しかし、付録 B で説明されているように、出力に関しては、どの出力文書形式も、いくつかの画像形式しか受け付けません。したがって L_X は、画像を正しい形式に変換するのに、バックグラウンドで ImageMagick というプログラムを使用しています。バックグラウンドで変換が発生することを避けて作業能率を上げようとお考えならば、出力ファイル書式に直接埋め込める画像書式のみを使用するとよいでしょう。出力ファイル書式は、付録 B に説明されています。

フォントと同じく、画像形式にも以下の二つの型があります。

ビットマップ画像 は、ピクセル値を（時により圧縮された形で）保持しています。したがって、この画像形式は完全に伸縮可能ではなく、拡大するとピクセルが目立つようになります。よく知られたビットマップ画像形式には、「Graphics Interchange Format（画像交換形式）」（GIF・ファイル拡張子は「.gif」）、「Portable Network Graphics（可搬性ネットワーク画像ファイル）」（PNG・ファイル拡張子は「.png」）、「Joint Photographic Experts Group（写真専門家連合組織）」（JPG・ファイル拡張子は「.jpg」または「.jpeg」）などがあります。

ベクトル画像 はベクトル値を保持しているので、データ損失なしに任意の寸法の伸縮させることができます。プレゼンテーションは、ビデオプロジェクタがかならず伸縮操作を行うことになるので、プレゼンテーションを作成したいときには画像が伸縮できることが必須です。また、伸縮性は、オンライン文書でユーザーが図の拡大ができるようにするのに便利です。


よく知られた伸縮可能画像形式には、「Scalable Vector Graphics（伸縮可能ベクトル画像ファイル）」（SVG・ファイル拡張子は「.svg」）、「Encapsulated PostScript（カプセル化ポストスクリプト）」（EPS・ファイル拡張子は「.eps」）、「Portable Document Format（可搬文書形式）」（PDF・ファイル拡張子は「.pdf」）、「Windows メタファイル」（WMF・ファイル拡張子は「.wmf」）がありえます。「ありえます」と言ったのは、ビットマップ画像はいつでも PDF や EPS に変換することができますが、変換後も依然としてビットマップ画像のままだからです。このような場合には、画像特性を示すヘッダが元々の画像に付け加えられるだけなのです²。たとえば、Adobe Photoshop が生成する PDF ファイルは、ビットマップ画像です。

通常、ビットマップ画像を伸縮可能画像には変換できず、その逆のみが可能です。PDF や PostScript 出力ファイルに直接埋め込むことができるのは、それぞれ PDF 画像書式と EPS 画像書式のみです。いまのところ適切な WMF/SVG→PDF/EPS 変換子がないので、SVG および WMF 画像は、出力ファイルが生成される際にビットマップに計算し直されます。

²PDF の場合には、さらに元画像が圧縮されます。

第2章 表

2.1 はじめに

表を挿入するには、ツールバーボタンか、挿入▷表メニューを選択します。ツールバーボタンを押すと、グラフィカルな選択ができます。マウスを動かして、生成する表の行数・列数を設定し、マウスボタンを押して下さい。表を作るのにメニューを使う場合は、ダイアログが現れ、行数と列数を訊いてきます。

既定の表では、各セルの周囲に罫線が引かれ、第一行が表の他の部分から分離して表示されます。この分離が起こるのは二重線のためです。第一行のセルが下に罫線を伴っているのと同時に、第二行のセルも上に罫線を伴っているのです。以下は表の一例です。

	1	2	3
い			
ろ			
は			

2.2 表ダイアログ

表を右クリックすると、表ダイアログが表示されて、表を変更することができます。ここでは、現在カーソルが置かれているセルないし行・列の設定を調節することができます。また、ダイアログオプションのほとんどは、選択範囲に対しても動作します。つまり、複数のセルや行・列を選択していれば、操作は選択範囲の全てに対して行われます。セルの内容を選択するのと、セル自体を選択するのは違うことに注意して下さい。

ダイアログオプションの直ちに適用を使用すると、ダイアログ中に行った変更が直ちに適用されるようになります。カーソルを別のセルに移動すると、ダイアログも新しいセルのパラメーターに更新されます。ダイアログ中の長さの値を変更した場合には、リターンを押さないと変更が適用されません。

表ダイアログの以下の各タブを使用して、表に変更を加えることができます。

表の設定 ここでは、現在列の幅と水平揃えを設定することができます。小数点でとある揃えは、第 2.12.2 で説明されています。幅を設定した場合には、現在行の垂直揃えも調整することができます。列幅を与えると、改行や多段落の文章をセルに入れることができます。これに関しては、第 2.8.1 節をご参照下さい。列幅を設定しない場

合には、列幅は、もっとも幅の広いセル内容の幅になります。

さらに、同一行の一つまたは複数のセルを連結列セルとして指定することができます。これに関しては、第2.8.2節をご参照下さい。あるいは同一列のセルを連結行セルとして指定することもできます。これについては、第2.8.3節をご参照ください。回転関連のチェックボックスは、現在セルや選択範囲あるいは表全体を反時計回りに90度回転させます。回転は、 L_YX 内部では反映されず、出力にのみ現れます。

【注】DVIビューアのすべてが回転を表示できるわけではありません。

下記は、本文行中の表に、表全体の垂直揃えを適用したものです。

各種の揃えを適用した同一行中の表：

	1		1		1
A		A		A	

また、特殊な表設定に必要な $\text{L}_\text{A}_\text{T}_\text{E}_\text{X}$ 引数を入力することもできます。これに関しては、第2.8.2.2節および第2.11節をご参照下さい。

罫線 このタブでは、現在列ないし現在行の罫線を引いたり消したりすることができます。様式オプションとしてフォーマル様式を使用すると、第2.9節に述べられているフォーマル様式の表に表を変換します。

またここで、第2.12.1節に述べられているように、表の行に空白を追加することもできます。

長尺表 このタブは、複数のページにわたる、いわゆる「長尺表」を作るためのものです。第2.6節および第2.7節に、長尺表の機能に関する詳細な説明があります。

2.3 表ツールバー

表ツールバーは、表ダイアログの代わりに、表をすばやく変更できるようにするためのものです。通常、カーソルが表の内部にあるときに、 L_YX のメインウィンドウの底部に表示されます。ご希望であれば、 L_YX のメインメニューバーを右クリックすることによって、つねに表示されるように切り替えることもできます。

ツールバーには、以下のアイコンがあります。



は、現在セルあるいは選択範囲の下に行を追加します。



は、現在セルあるいは選択範囲の右に列を追加します。




は、現在行あるいは選択範囲行を削除します。





は、現在列あるいは選択範囲列を削除します。





は、現在セルや現在行、あるいは選択範囲の上部に罫線を描画します。


は、現在セルや現在行、あるいは選択範囲の下部に罫線を描画します。

は、現在セルや現在行、あるいは選択範囲の左側に罫線を描画します。


は、現在セルや現在行、あるいは選択範囲の右側に罫線を描画します。


は、現在セルあるいは選択範囲の周囲に罫線を描画します。現在セルが連結列でなければ、現在行


は、現在セルあるいは選択範囲のすべての罫線を消去します。現在セルが連結列でなければ、現在

は、現在セルあるいは現在列の内容を左揃えにします。


は、現在セルあるいは現在列の内容を水平方向中央揃えにします。

は、現在セルあるいは現在列の内容を右揃えにします。

は、現在セルの内容を特定文字位置で揃えます。


は、現在セルの内容を垂直方向上揃えにします。


は、現在セルの内容を垂直方向中央揃えにします。

は、現在セルの内容を垂直方向下揃えにします。

は、現在セルあるいは選択範囲を反時計回りに 90 度回転します。

は、表全体を反時計回りに 90 度回転します。

は、現在セルあるいは選択範囲を連結列にします。

は、現在セルあるいは選択範囲を連結行にします。

【註】出力では、行の最初のセルの垂直揃えが、同一行の他のセルすべてに適用されます。

2.4 表編集メニュー


表ダイアログと表ツールバーの他に、編集▷表メニューでも、現在行や現在列の罫線を引いたり消したりすることができ、あるいは現在の選択範囲を連結列に指定したりすることができます。このメニューは、カーソルが表内部にあるときのみ、表示されます。

2.5 表フロート

フロートの一般的な説明については、第3.1節をご覧ください。

表 2.1: 表フロート

1	2	3
森嶋	宇沢	二階堂
$\int x^2 dx$	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	$1 + 1 = 2$

表フロートは、挿入▷フロート▷表メニューかツールバーボタンを使うと挿入することができます。



フロートは折りたたむことのできるボックスとして表示され、中には「表 #:」(#は実際には表番号です)というラベルのついたキャプションがあります。表は、このキャプションの上にも下にも入れることができます。

表 2.1は、表フロートに入れた表の例です。

キャプションを表の上に入れるのは、標準的な組版方法ですが、残念ながら \LaTeX の標準クラスではサポートされていません。つまり、article・book・letter・reportのような文書クラスを使用した場合には、キャプションと表のあいだに余白が入りません。必要な余白を入れるには、使用中の文書のプリアンブルに、 \LaTeX パッケージの caption を

`tableposition=top`

というオプションとともにload コマンドを使って読み込んでください¹。キャプション書式を調整するには、第3.9節にものべられているようにcaption パッケージを使用します。

表は、本文中において、ラベルを参照することによって相互参照することができます。そのためには、挿入▷ラベルメニューかツールバーボタンのを用いて、キャプションにラベルを挿入して下さい。然る後に、挿入▷相互参照メニューかツールバーボタンのを用いて、このラベルを参照することができます。

参照は、第3.4節で詳細に説明されています。

2.6 長尺表

表が長すぎて1ページに収まりきらないときには、表ダイアログの長尺表タブにある長尺表を使うオプションを使用して、複数ページに自動的に分割させることができます。このオプションを有効にすると、いくつかのチェックボックスが使えるようになり、以下のものを定義することができるようになります。

¹詳しくは、第3.10節をご覧ください。

ヘッダ：カーソル位置の行が、長尺表の全ページに表示されるヘッダ行として定義されます。ただし、冒頭ヘッダが定義されているときは、冒頭ページを除きます。これと区別して、このヘッダは本体ヘッダと呼びます。

冒頭ヘッダ：カーソル位置の行が、長尺表の最初のページに表示されるヘッダ行として定義されます。

フッタ：カーソル位置の行が、長尺表の全ページに表示されるフッタ行として定義されます。ただし、末尾フッタが定義されているときは、末尾ページを除きます。

末尾フッタ：カーソル位置の行が、長尺表の最後のページに表示されるフッタ行として定義されます。

キャプション：カーソル位置の行を表のキャプションにします。この行は一列にリセットされ、キャプションが挿入されます。長尺表のキャプションについての詳細は、第2.6.3節で説明されています。

加えて、表が分割される行を指定することができます。長尺表の動作を見るために以下の例を参照してください。

例示用電話帳（名前は無視してください）		
氏名		TEL.
Annovi	Silvia	111
Bertoli	Stefano	111
Bozzi	Walter	111
Cachia	Maria	111
Cachia	Maurizio	111
Cinquemani	Giusi	111
Colin	Bernard	111
Concli	Gianfranco	111
Dal Bosco	Carolina	111
Dalpiaz	Annamaria	111
Feliciello	Domenico	111
Focarelli	Paola	111
Galletti	Oreste	111
Gasparini	Franca	111
Rizzardi	Paola	111
Lassini	Giancarlo	111
Malfatti	Luciano	111
Malfatti	Valeriano	111

次頁へつづく

例示用電話帳		
氏名		TEL.
Meneguzzo	Roberto	111
Mezzadra	Roberto	111
Pirpamer	Erich	111
Pochiesa	Paolo	111, 222
Radina	Claudio	111
Stuffer	Oskar	111
Tacchelli	Ugo	111
Tezzele	Margit	111
Unterkalmsteiner	Frieda	111
Vieider	Hilde	111
Vigna	Jürgen	111
Weber	Maurizio	111
Winkler	Franz	111
Annovi	Silvia	555
Bertoli	Stefano	555
Bozzi	Walter	555
Cachia	Maria	555
Cachia	Maurizio	555
Cinquemani	Giusi	555
Colin	Bernard	555
Concli	Gianfranco	555
Dal Bosco	Carolina	555
Dalpiaz	Annamaria	555
Feliciello	Domenico	555
Focarelli	Paola	555
Galletti	Oreste	555
Gasparini	Franca	555
Rizzardi	Paola	555
Lassini	Giancarlo	555
Malfatti	Luciano	555
Malfatti	Valeriano	555
Meneguzzo	Roberto	555
Mezzadra	Roberto	555

次頁へつづく

例示用電話帳		
氏名		TEL.
Pirpamer	Erich	555
Pochiesa	Paolo	555, 222
Radina	Claudio	555
Stuffer	Oskar	555
Tacchelli	Ugo	555
Tezzele	Margit	555
Unterkalmsteiner	Frieda	555
Vieider	Hilde	555
Vigna	Jürgen	999
Weber	Maurizio	555
Winkler	Franz	555

以上

2.6.1 長尺表での脚注

脚注は、どの長尺表セルにも入れることができます。脚注は、脚注のついた表セルのあるページの下部に表示されます。たとえば、表 2.6 には脚注が付いています。

2.6.2 長尺表の縦横揃え

表設定ダイアログでは、長尺表の水平揃えを指定することができます。左揃えや右揃えでは、表は直接ページ境界に寄せられます。表と境界の間に余白を加えるには、中央揃えに設定してから

`\setlength{\LTleft}{値}`

という行を当該長尺表の前に $\text{T}_{\text{E}}\text{X}$ コードとして挿入して、長さ `\LTleft` および `\LTRight` の値を変更しなくてはなりません。ここで「値」は、第 A.1 表に挙げてある単位であれば、どの単位を取ることもできます。`\LTleft` は、左頁余白から長尺表までの水平方向の距離を制御し、`\LTRight` は、右頁余白からの距離を制御します。これらの長さの既定値は `\fill` となっており、 $\text{L}_{\text{Y}}\text{X}$ の水平フィルと同等になっています。

以下の長尺表は、中央揃えの上、`\LTleft` を `0.1\columnwidth` に設定してあります。

1	2	3	4	5
---	---	---	---	---

いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	いろは	いろは	いろは	いろは

2.6.3 長尺表のキャプション

フロートは一頁に収まらなくてはならないので、長尺表を表フロートに入れることはできません。しかし、第2.6節に述べられているように、ある行で長尺表オプションのキャプションを使用すれば、フロートのキャプション環境を長尺表でも使用することができます。キャプションを入れることができる行はひとつだけです。

以下は、動作を確認するための短い長尺表です。

表 2.2: キャプション付き長尺表

1	2	3	4	5
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	いろは	いろは	いろは	いろは

【註1】表番号は、長尺表にキャプションを付けなかったとしても、各長尺表毎に加番されます。このため、あいだにキャプションのない二つの長尺表が挟まっていたならば、表一覧では、たとえば表2.1の後に表2.4が来ることがあります。これを回避するためには、キャプションのないすべての長尺表の後に、 \TeX コードで

```
\addtocounter{table}{-1}
```

というコマンドを挿入する方法があります。

全長尺表にキャプションがなかったり、文書プリアンブルに以下のコードを書き加えていれば、以上のことは必要ありません。

```
\let\myEnd\endlongtable
```

```
\renewcommand{\endlongtable}{\myEnd\addtocounter{table}{-1}}
```

【註2】相互参照をリンクするために、文書設定ダイアログのPDF特性でhyperrefを使用している場合、長尺表キャプションへのリンクは、つねに文頭に向けられます。

2.6.3.1 長尺表への参照

表 2.3: 参照された長尺表

1	2	3	4	5
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	いろは	いろは	いろは	いろは

長尺表を参照するには、キャプションにラベルを挿入してください。

これは表 2.3 への参照です。

キャプションのレイアウトは、文書中の他のすべてのキャプションと一緒に、 \LaTeX パッケージの `caption` を使用して設定することができます。第 3.9 節をご覧ください。

2.6.3.2 キャプションの幅

キャプション行の最大長は、長さ `\LTcapwidth` によって指定することができます。既定値は `4 in` です。これを変更するには、現文書の文書プリアンプルか、調整したい長尺表の前の文中に \TeX コードとして

```
\setlength{\LTcapwidth}{幅}
```

というコマンドを書き加えて下さい。ここで「幅」には、付録 A に挙げられている単位を取ることができます。

以下の各表で違いを示します。

表 2.4: 既定幅のなが〜いフルタイトル・既定幅のなが〜いフルタイトル・既定幅のなが〜いフルタイトル

1	2	3	4	5
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	いろは	いろは	いろは	いろは

表 2.5: 5 cm 幅のなが〜いフルタイトル・5 cm 幅のなが〜いフルタイトル・5 cm 幅のなが〜いフルタイトル

1	2	3	4	5
---	---	---	---	---

いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	ろ	ろ	ろ	いろは
いろは	いろは	いろは	いろは	いろは

【註】この文書のように \LaTeX パッケージの`caption`を使用している時には、`\LTcapwidth`に既定値の`4in`を使用している場合、ページの全幅が用いられます。この場合、正確に`4in`幅のキャプションを作るには、`3.99in`のように`4.0in`と少し違う値を指定するか、`caption`パッケージが提供している`\captionsetup{width=値}`という \LaTeX コマンドを用いる方法があります。

2.6.3.3 表の各ページごとに異なるキャプション

次頁以降の表のキャプションを、1頁めとは異なるものにしたい場合には、キャプションを`\caption*{キャプション文}\%`

という \TeX コードコマンドを使って、ヘッダとして指定したダミーのキャプション行に入れて下さい。表2.6は、異なるヘッダを持つ長尺表の一例です。2番目のキャプションには表番が現れないようにしてあります。

表 2.6: 例示用電話帳

例示用電話帳（名前は無視の事）		
氏名		TEL.
Annovi	Silvia	111
Bertoli	Stefano	111
Bozzi	Walter	111
Cachia	Maria	111
Cachia	Maurizio	111
Cinquemani	Giusi	111
Colin	Bernard	111
Concli	Gianfranco	111
Dal Bosco	Carolina	111
Dalpiaz	Annamaria	111
Feliciello	Domenico	111
Focarelli	Paola	111
Galletti	Oreste	111
Gasparini	Franca	111

次頁に続く

例示用電話帳の続き

例示用電話帳		
氏名		TEL.
Rizzardi ²	Paola	111
Lassini	Giancarlo	111
Malfatti	Luciano	111
Malfatti	Valeriano	111
Meneguzzo	Roberto	111
Mezzadra	Roberto	111
Pirpamer	Erich	111
Pochiesa	Paolo	111, 222
Radina	Claudio	111
Stuffer	Oskar	111
Tacchelli	Ugo	111
Tezzele	Margit	111
Unterkalmsteiner	Frieda	111
Vieider	Hilde	111
Vigna	Jürgen	111
Weber	Maurizio	111
Winkler	Franz	111
Annovi	Silvia	555
Bertoli	Stefano	555
Bozzi	Walter	555
Cachia	Maria	555
Cachia	Maurizio	555
Cinquemani	Giusi	555
Colin	Bernard	555
Concli	Gianfranco	555
Dal Bosco	Carolina	555
Dalpiaz	Annamaria	555
Feliciello	Domenico	555
Focarelli	Paola	555

次頁に続く

²例示用脚注

例示用電話帳の続き

例示用電話帳		
氏名		TEL.
Galletti	Oreste	555
Gasparini	Franca	555
Rizzardi	Paola	555
Lassini	Giancarlo	555
Malfatti	Luciano	555
Malfatti	Valeriano	555
Meneguzzo	Roberto	555
Mezzadra	Roberto	555

2.7 長尺表にまつわる諸点

2.7.1 長尺表の計算

\LaTeX は、表ページの頁高と改頁をいわゆる単位塊（チャンク chunk）を使って計算します。単位塊は、一度に \LaTeX のメモリに読み込まれる表の断片です。歴史上の経緯から、既定値はたった表 20 行分に設定されています。多頁にわたる長尺表がある場合、この設定では文書の生成に時間がかかるかもしれません。この場合、

```
\setcounter{LTchunksize}{100}
```

というコマンド行を文書のプリアンブルに書き加えることで、単位塊の大きさを 100 から 1000 の値に支障なく増やすことができます。

2.7.2 フロートと長尺表

長尺表が開始するのと同じページにフロートが置かれると問題が発生する可能性があります。このような状況を避けるためには、`\clearpage` コマンドを長尺表の前に \TeX コードとして書き加えてください。

2.7.3 強制改頁

既定の設定では、表は行の間でのみ改頁されます。もし複数行を含むセルがあり、そのセル中で改頁を行いたい時には、セル中の改頁してもよい場所に $\text{T}_{\text{E}}\text{X}$ コードで改行コマンド「`\\`」を挿入します。`\\`コマンドの前には、現セルよりも右にある列数と同じ数の「`&`」文字を、 $\text{T}_{\text{E}}\text{X}$ コードで挿入してやる必要があります。この`&`は、表セルの区切り文字です。各`&`の後に、右側の本来のセルに入るべき内容を $\text{T}_{\text{E}}\text{X}$ コードで書き込み、右側のセルの内容は削除してください。

`\\`コマンドの後ろには、現在列の左側にある列数と同じ数の`&`文字を挿入してください。表 2.7 では、改頁さるべきセルは、全 3 列中 2 列めにあります。したがって、

```
& 111\\ \newpage
&
```

というコマンドが、セル中「*Castelchiodato,*」の後に $\text{T}_{\text{E}}\text{X}$ コードとして挿入してあります。同一行 3 列目の「111」は削除してあります。`\newpage` は、改行がこの場所でかならず起こるように指定したいときのみ必要です。これを書かなければ、改頁の候補として指定するに過ぎません。何らかの理由で長尺表のフッタ行に上罫線を引いていないけれども、セルを改頁する場所で水平線を引きたい場合には、代わりに

```
& 111\\
\hline &
```

というコマンドを使用して下さい。

改頁をしたいセルが最右列にある場合には、

```
\setlength{\parfillskip}{0pt}
```

というコマンドを当該セルの初めに $\text{T}_{\text{E}}\text{X}$ コードで挿入しなくてはなりません。こうすることで、新規頁部分に表示されるセル内の部分が、正しく全幅を使用するようにできます。

表 2.7: 表セル内で強制改頁をしている表

例示用電話帳（氏名は無視の事）		
氏名		TEL.
Annovi	Silvia	111
Bertoli	Stefano	111
Bozzi	Walter	111
Cachia	Maria	111
Cachia	Maurizio	111
Cinquemani	Giusi	111
Colin	Bernard	111
Concli	Gianfranco	111

次頁に続く

例示用電話帳の続き

例示用電話帳		
氏名		TEL.
Dal Bosco	Carolina	111
Dalpiaz	Annamaria	111
Feliciello	Domenico	111
Focarelli	Paola	111
Galletti	Oreste	111
Gasparini	Franca	111
Lassini	Giancarlo	111
Malfatti	Luciano	111
Malfatti	Valeriano	111
Meneguzzo	Roberto	111
Mezzadra	Roberto	111
Pirpamer	Erich	111
Pochiesa	Paolo	111, 222
Radina	Claudio	111
Rizzardi	Paolo, 11. Fürst von Montecompati, 11. Fürst von Sulmona und Vivaro, 10. Fürst von Rossano, 5. Herzog von Canemorte, 11. Herzog von Palombara, 5. Herzog von Castelchiodato,	111

次頁に続く

例示用電話帳の続き

例示用電話帳		
氏名		TEL.
	11. Herzog von Poggionativo, 11. Markis von Mentana, Norma, Civitella, Pratica, Moricone und Percille, 11. Graf von Valinfreda, 11. Baron von Cropalati, 11. Herr von Scarpa, Edelmann von Rom, Patrizier von Venedig, Neapel und Genua	
Stuffer	Oskar	111
Tacchelli	Ugo	111
Tezzele	Margit	111
Unterkalmsteiner	Frieda	111
Vieider	Hilde	111
Vigna	Jürgen	111
Weber	Maurizio	111
Winkler	Franz	111

2.8 複数行・連結列・連結行

2.8.1 表セル内の複数行

表 2.8: セル内に複数の行がある表

複数の 行	ろ	は
に	ほ	へ
と	ち	り

列に固定幅を指定することによって、文章をハイフネーションを伴った複数行の段落として、入れることができるようになります。

表 2.8 を生成するには、まず 3×3 表を作り、最初のセルを右クリックして現れるメニューから、設定メニューを選択します。すると表ダイアログが現れるので、セル幅を 2.5 cm 幅とし、水平揃え及び垂直揃えをともに中央揃えに設定します。垂直揃えは、その行のすべてのセルに適用されます。この例の文章は 2.5 cm よりも短いため、一行しか表示されません。そこで二行とも表示するために、両端揃え改行（短絡キー $\text{Ctrl}+\text{Shift}+\text{Return}$ ）を挿入してあります。もし文章が設定されたセル幅よりも広い場合には、文章は自動的にいくつかの行に改行されます。

固定幅のセルに長い単語を入れた場合、もしそれがセルの最初にあると、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ はハイフネーションを行うことができません。したがって、その単語がセルの最初に来ないようにするためには、何かを入れなくてはなりません。そこで、その単語の前に 0 pt 幅の水平空白を挿入します。空白がゼロ幅なので、出力は変わりません。表 2.9 が違いを示しています³。

表 2.9: ハイフネーション有/無の表


verylongtablecell	ろ	は
に	ほ	へ
と	ち	り

very-longtablecell-word	ろ	は
に	ほ	へ
と	ち	り

³ 【訳註】 $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ を使用している場合、日本語の文章は正しく改行されます。

2.8.2 連結列

2.8.2.1 連結列の基礎

一つのセルが複数の列にまたがるようにするためには、行中から一つの連結セルにまとめるべき各セルを選択して、表ツールバーボタンを押すか、編集▷表▷連結列メニューを使用するか、あるいは選択したセルを右クリックして現れるメニューから連結列を有効にするかしてください。

連結列はセル設定が独立しています。つまり、セル罫線やセルの縦横揃え、幅設定等を変更した場合には、現在の連結列のみに適用されるのです。以下は、第一行に連結行があり、最下行に上罫線のない連結行がある表の例です。

いろは	にほへとちり		ぬるを
甲	乙	丙	丁
一	二	三	四

2.8.2.2 連結列の計算

L_YX は、連結列を直接にサポートしていますが、同列にある連結セル以外のセル幅については、考慮が必要です。

表 2.10: 中央揃え連結列の下にちょうど半分の幅を持つ列が 2 つある表

複数行にわたる連結列		は
に	ほ	へ
と	ち	り

たとえば、表 2.10 を生成するには、3×3 表の第一行の最初の二つのセルを選択して、右クリックして現れるメニューから設定を選択します。そして現れる表の設定ダイアログで、このセルを連結列、中央揃え、2.5 cm 幅に設定します。被連結列のセルが連結列セルのちょうど半分の幅になるようにしたいので、第一列の幅を 1.25 cm 幅に設定します。すると第二列は自動的に 1.25 cm 幅（連結列幅マイナス第一列の幅）になります。表 2.11 もこのようにして作成してあります。

ご覧になって分かるように、第一列が、連結セルの幅の半分よりも少し大きくなっています。その原因は、セル幅が、指定された幅よりもつねに少しだけ大きく描かれるために、指定したセル幅 $W_{\text{指}}$ がセルの全幅 $W_{\text{全}}$ と等しくならないためです。付録 (C) がこの事情を詳細に説明しています。

n 列を連結する際、各列の全幅が $W_{\text{全:連結}}/n$ となるために必要な指定幅 $W_{\text{指:n}}$ は、

$$W_{\text{指:n}} = (W_{\text{指:連結}} + (1 - n) \cdot (12.4 \text{ pt})) / n \quad (2.1)$$

表 2.11: 連結セルの下の列が正確に半分になっていない表

複数行にわたる連結列		は
に	ほ	へ
と	ち	り

で計算することができます。上記の例では $n = 2$ かつ $W_{\text{指:連結列}} = 2.5 \text{ cm}$ かつ長さは既定値ですから、第 2.1 式は

$$W_{\text{指:2}} = 1.25 \text{ cm} - 6.2 \text{ pt} \quad (2.2)$$

となります。L^AT_EX 中で計算を可能にするには、文書プリアンブル行に

`\usepackage{calc}`

と書いて、L^AT_EX パッケージ calc を読み込まなくてはなりません。L_YX の「表の設定」ダイアログの幅フィールドでは、長さを計算することができません。したがって、ダイアログの L^AT_EX の引数欄に書き込むことによって列を整形しなくてはなりません。以下は、その引数の概要です。

- `p{幅}` は固定幅セルを生成し、内容文は垂直方向上揃えになります。
- `m{幅}` は固定幅セルを生成し、内容文は垂直方向中央揃えになります。
- `b{幅}` は固定幅セルを生成し、内容文は垂直方向下揃えになります。

L^AT_EX 変数を入力すると、表ダイアログで設定されたすべてのセル設定は無効になります。

【註】バグのせいで、L_YX は無効にされた設定も表示してしまいます。

文章を水平方向中央揃えにしたいので、`\centering` コマンドを入れます。したがって、以下のような L^AT_EX 変数を、最初の被連結列に入れます。

`>\centering m{1.25cm-6.2pt}`

`>\{ }` コマンドは、セルが生成される前に中括弧内のコマンドを適用するという意味です。

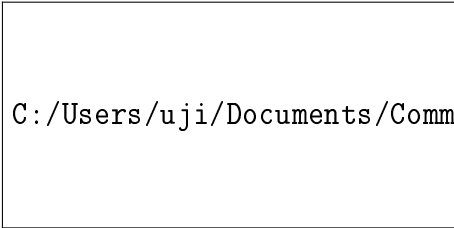
こうして被連結列セルの文章に、中央揃えを選択しましたが、連結セルの文章は依然として左揃えになっています。これは、L_YX は揃えを一行分のみに適用するためです。したがって、連結セルには

`>\centering m{2.5cm}`

という L^AT_EX 変数を指定しなくてはなりません。

2.8.3 連結行

複数行にかかるセルを連結するには、同一列中の統合したいセルをすべて選択し、ツ

ルバーボタン  を押すか、メニュー編集▷表▷連結行を選ぶか、選択したセルを右クリックして、コンテキストメニューから▷連結行を選んでください。

下記は一列目に連結列を適用した表の例です。

長いセル項目	ろ	は
連結列	に	ほ
	と	ち

連結列の幅は、列全体の幅と揃えを設定することによって設定することができます。垂直オフセットは、正または負の値を与えると、連結列の内容を上下にシフトします。

連結行セルの列に明示的な幅が与えられていない場合、連結行セルは当該列の揃えを継承します。それもなければ左揃えです。既定値を上書きするには、

`\renewcommand{\multirowsetup}{\centering}`

というコマンドを使用して、コマンド `\multirowsetup` を書き換えて下さい。すると、文書中の連結行の中身が、すべて中央揃えになります。もしいくつかの表だけ中央揃えにしたいのであれば、プリアンプルではなく表の直前に \TeX コードボックスを入れて、このコマンドを書き換えれば済みます。文章を右揃えにしたい場合には、`\centering` の代わりに `\raggedleft` を使用して下さい。左揃えに戻すには、`\raggedright` とします。

下記は、-4 ミリの垂直オフセットを設定し、3 センチ幅の列中にある右揃えの連結列セルがある表の例です。

長いセル項目	ろ	は
り	ぬ	る
連結列	に	ほ
	と	ち

2.9 フォーマル形式の表

書籍中の表は、よく表 2.12 のように組版されます。この種の表は、「フォーマル形式」と呼ばれます。表をフォーマル形式にするには、表の設定ダイアログの罫線タブでフォーマル形式オプションを有効にして下さい。

表 2.12: ブックタブ表の例

システム	Medipix 1		Medipix 2	
検知器の厚み [μm]	300	300	700	
エッジ角 [$^{\circ}$]	3.55	2.71	7.99	
空間分解能 [μm]	4.26	10.17	10.56	
f_{max} における MTF	0.53	0.37	0.39	
LSF 空間分解能				
μm 表示	129.7	52.75	50.78	
ピクセルサイズに対する%表示	76.3	95.9	92.3	

表中の行に余白を加えるには、第 2.12.1 節に述べられているように、表の設定ダイアログの罫線タブを使用します。

通常の表と違って、フォーマル形式の表には縦罫線がありません。表の横罫線は、通常の表と同様に指定することができますが、出力でこれらの線幅は異なります。

最初と最後の表罫線の既定幅は 0.08 em ですが、他の罫線の既定幅は 0.05 em です。

これらの既定幅は、以下のプリアンブル行を指定することによって変更することができます。

```
\let\mytoprule\toprule
\renewcommand{\toprule}{\mytoprule[幅]}
```

上の例は、toprule という最初の罫線を設定しています。最後の罫線の線幅を変更するには、toprule を bottomrule に置き換えて下さい。他の罫線の線幅を変更するには、toprule を midrule に置き換えます。線幅を設定する上では、付録 A に挙げてあるすべての単位を使用することができます。

全列にまたがらない罫線は、連結列セルに罫線を指定することで作成することができます。L_AT_EX は、内部的に \cmidrule コマンドを使用して、このような罫線を引きます。このコマンドの完全な書式は

```
\cmidrule[幅](刈込){開始列-終了列}
```

です。 \cmidrule のオプションは、現在のところ L_AT_EX がサポートしていないので、オプションを使用するためには、T_EX コードを使わなくてはなりません。 \cmidrule は、行の第一列の中に、最初に T_EX コードでこのコマンドを入れることによって、手動で指定することができます。この罫線は、出力では現在行の上に引かれることになります。

この線幅の既定値は 0.03 em です。「開始列」は罫線を開始する列番号で、「終了列」は罫線を終了する列番号です。終了列は、罫線を一列だけに引きたいときも含めて、常に指定しなくてはなりません。非必須パラメーターの「刈込」は、 $l\{\text{刈込幅}\}$ あるいは $r\{\text{刈込幅}\}$ の値を取り、刈込幅の指定は非必須です。たとえば、 $l\{2pt\}$ というパラメーターを使う

と、罫線が左端から 2 pt だけ刈り込まれます。刈込幅を指定しないと、罫線は既定値の 0.5 em だけ刈り込まれます。

表 2.12は、

`\cmidrule(r){2-2}\cmidrule(l){3-4}`

というコマンドを第二行の初めに使用し、

`\cmidrule(l{10pt}){1-1}`

を第六行に使用して作成されています。

ときには、表 2.13のような重なり合った`\cmidrule`が欲しいことがあるかもしれません。これは、 \TeX コードで

`\morecmidrules`

コマンドを使用することによって実現することができます。表 2.13の第二列には

`\cmidrule(r){2-2}\cmidrule(l){3-4}\morecmidrules\cmidrule{2-4}`

というコマンドが使用されており、第六列には

`\midrule\morecmidrules\cmidrule{3-4}`

というコマンドが使用されています。

罫線の余白取りにどうしても満足が行かない場合には、

`\specialrule{幅}{上余白}{下余白}`

というコマンドを使用すれば、全列にわたる罫線を生成することができます。これらの特殊機能については、 \LaTeX パッケージ `booktabs` [4] の取扱説明書をご参照下さい。

表 2.13: 特殊なブックタブ表

システム	Medipix 1		Medipix 2
検知器の厚み [μm]	300	300	700
エッジ角 [$^{\circ}$]	3.55	2.71	7.99
空間分解能 [μm]	4.26	10.17	10.56
f_{max} における MTF	0.53	0.37	0.39
LSF 空間分解能			
μm 表示	129.7	52.75	50.78
ピクセルサイズに対する%表示	76.3	95.9	92.3

2.10 表の垂直揃え

本文行中で表の垂直揃えをするには、その表をボックスの中に入れなくてはなりません。そうした上で、第5.2節に述べられているように、ボックスの垂直揃えを行います。

以下の例では、各表は15列幅%を持つ minipage⁴ボックスの中に入れてあります。

- 動作試験

い	に	と
ろ	ほ	ち
は	へ	り

い	に	と
ろ	ほ	ち
は	へ	り

- 動作試験

い	に	と
ろ	ほ	ち
は	へ	り

い	に	と
ろ	ほ	ち
は	へ	り

- 動作試験

い	に	と
ろ	ほ	ち
は	へ	り

い	に	と
ろ	ほ	ち
は	へ	り

上記のとおり、表の第一行あるいは最終行の中身が、表外の本文行と揃っていません。これらを揃えるには、minipage ボックスを raisebox⁵の中に入れなくてはなりません。上記の例では、最初の項目の第二表は、ボックスの前に

```
\raisebox{0.85\baselineskip}{
```

という TeX コードコマンドを使用して揃えています。ボックスの後には、閉じ括弧}を TeX コードとして挿入してあります。最後の項目の第二表には、

```
\raisebox{-0.32\baselineskip}{
```

というコマンドを使用しています。

【註】表の行の中身の周囲の本文行への揃えは、つねに同じ値をとるわけではありません。この揃え方に必要な\raisebox コマンドの値は、文書フォントやフォント寸法、表罫線の線幅によって変わります。

2.11 色付きの表

2.11.1 色付きのセル

もし色の付いた文章が欲しいだけならば、そのセルを選択して、編集▷文字様式メニューで色を選んでください。表2.14を作るのに、この方法が使われています。それ以外の場合

⁴minipage は第5.4節で説明されています。

⁵raisebox は、第5.6.2節で説明されています。

表 2.14: colortbl パッケージを使用しない色付き表

い	ろ	は
に	ほ	へ
と	ち	り

には、 \LaTeX パッケージの `colortbl` を使用しなくてはなりません。

色付きの表を作成するには、プリアンブルに

```
\usepackage{colortbl}
```

という行を書いて `colortbl` を読み込まなくてはなりません。列の色は、`>{ }` コマンドの中に

```
\columncolor{色彩名}
```

というコマンドを書いて指定することができます。`>{ }` コマンドに関する詳細は、第 2.8.2.2 節に述べられています。

以下の色彩名が事前に定義されています。

`red`・`green`・`yellow`・`blue`・`cyan`・`magenta`・`black`・`white`

また

```
\definecolor{色彩名}{色彩モデル}{色彩値}
```

というコマンドを使用すれば、自分自身の色彩を定義することができます。ここで色彩モデルは

`cm \underline{y} k` : シアン (`c \underline{y} an`)・マゼンタ (`m \underline{a} genta`)・黄 (`y \underline{e} llow`)・黒 (`blac \underline{k}`)

`rgb` : 赤 (`r \underline{e} d`)・緑 (`g \underline{r} een`)・青 (`bl \underline{u} e`)

`gray` : グレー

から選択することができ、色彩値は、0 から 1 の間の数字をコンマで区切ったもので、それぞれ当該色彩モデルにおける各色彩の大きさを示します。

たとえば、プリアンブルに

```
\definecolor{darkgreen}{cm $\underline{y}$ k}{0.5, 0, 1, 0.5}
```

と書くことによって、「*darkgreen*」色を定義することができますし、

```
\definecolor{lightgray}{gray}{0.8}
```

と書けば、「*lightgray*」色を定義することができます。

罫線は

```
\rowcolor{色彩名}
```

というコマンドを使用して色付けすることができ、セルは

`\cellcolor{色彩名}`

というコマンドで色付けすることができます。どちらのコマンドも、セルの冒頭に $\text{T}_\text{E}\text{X}$ コードで書き込みます。

表中の文字に色付けをするには、セルを選択してから、 $\text{L}_\text{Y}\text{X}$ のメニュー編集▷文字様式を使用します。セルに $\text{T}_\text{E}\text{X}$ コードが入っている場合には、文章部分だけ選択してください。さもないと色付け指定された $\text{T}_\text{E}\text{X}$ コードが $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ エラーを引き起こしてしまいます。

【注】DVI ビューアーのすべてが自己定義の色を表示できるわけではありません。

表 2.15 を作成するには、以下のようにしてください。第一列の色は *darkgreen* にしたいので、この列の $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ 変数として

`>\columncolor{darkgreen}\centering`c

と書き込みます。第一列は青にしたいので、この列の最初のセルに

`\rowcolor{cyan}`

という $\text{T}_\text{E}\text{X}$ コードコマンドを書き込みます。これによって、第一セルの列色指定が上書きされてしまうことに注意してください。最下行の最後のセルは、

`\cellcolor{magenta}`

という $\text{T}_\text{E}\text{X}$ コードコマンドを書き込むことで、マゼンタに色づけされています。この後、編集▷文字様式メニューを使って、各文字に色付けをすることができます。

表 2.15: colortbl パッケージを使用した色付き表

い	ろ	は
に	ほ	へ
と	ち	り

2.11.2 色付き罫線

第 2.12.4 節に述べられているように、表中の全罫線の線幅は、長さ `\arrayrulewidth` を使って調節することができます。本節の表では、すべて 1.5 pt に設定されています。

たとえば縦罫線を緑に着色するには、第 2.12.3 節の説明にしたがって、以下の列書式を文書プリアンブル中で作成して下さい。

`\newcolumntype{W}{!{\color{green}\vline}}`

表 2.16 では、最右列に $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ 変数 `WcW`、他の列に $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ 変数 `Wc` が使用されています。複数の色を用いたいときには、列書式をもっと定義してください。

表 2.16: 色付き縦罫線の表

sd		
	sd	
		sd

表 2.17のように、たとえば横罫線を赤にしたいときには、表ないし表フロートの前に $\text{T}_{\text{E}}\text{X}$ コードで

```
\let\myHlineC\hline
\renewcommand{\hline}
{\arrayrulecolor{red}\myHlineC\arrayrulecolor{black}}
```

というコマンドを入力して下さい。

表 2.17: 色付き横罫線の表

sd		
	sd	
		sd

既定値の黒罫線に戻るには、

```
\renewcommand{\hline}{\myHlineC}
```

というコマンドを、表ないし表フロートの後に $\text{T}_{\text{E}}\text{X}$ コードで入力して下さい。表 2.18は、縦横罫線を色付けした例です。

表 2.18: 色付き罫線の表

sd		
	sd	
		sd

2.12 表の調整

2.12.1 行の余白取り

表の設定ダイアログの罫線タブで、表の行に縦方向の余白を加えることができます。ここでは三つのことができます。

行上 これは表の行の文字上に余白を追加します。もし表がフォーマル形式の表⁶ならば、 L_YX は既定値として 0.5em の余白を挿入します。通常の表の場合には、余白を追加すると、残念ながら以下の表のように縦罫線が破壊されてしまいます。

い
行上に 3 mm の余白
は

したがって、通常表の行上に余白を加えるのは、縦罫線がないときのみに使用することができます。

行下 これは表の行の文字下に余白を追加します。もし表がフォーマル形式の表ならば、 L_YX は既定値として 0.5em の余白を挿入し、通常の表ならば既定値として 2pt を挿入します。

行間 これは表の現在行と次の行の間に余白を追加します。もし表がフォーマル形式の表ならば、 L_YX は既定値として 0.5em の余白を挿入します。通常の表の場合には、余白を追加すると、残念ながら以下の表のように縦罫線が破壊されてしまいます。

い
↓ 行間に 3 mm の余白 ↓
↑ 行間に 3 mm の余白 ↑

したがって、通常表の行間に余白を加えるのは、縦罫線がないときのみに使用することができます。

すべての表の全セルの高さを上げたい場合には、以下の行をプリアンプルに加えることによって実現できます。

```
\@ifundefined{extrarowheight}
{\usepackage{array}}{}
\setlength{\extrarowheight}{height}
```

しかしこの方法は、セル内の文章が縦方向に中央揃えにならないという欠点があります（自己定義の表書式を用いると、 L_YX は自動的に `array` パッケージを読み込みます。これが二度読み込まれることを防ぐために、上記のコマンドでは `\@ifundefined` が使用されています）。

標準大よりも大きいフォント寸法を使用している場合には、文字が表罫線に近寄りすぎるのがよくあります。これは、表の行の最初に $\text{T}_\text{E}\text{X}$ コードで `\strut` コマンドを挿入することで補正することができます。表 2.19 は、この効果を示したものです。

⁶フォーマル形式の表は第 2.9 節で説明されています。

表 2.19: 大きなフォント寸法の文章の垂直揃え

Normal, g	Normal, g
Large	Large
Larger	Larger
Largest	Largest
Huge	Huge
Huger	Huger

(a) 通常の表

(b) `\strut` コマンドを使用した表

2.12.2 特殊なセル揃え


ときには表 2.20 の小数点のように、同一列の各セルの内容が、特定の文字で揃えられた方が見栄えの良いことがあります。この揃え方は、当該列にカーソルを置いて、ツールバーボタン  を押すか、表設定中で揃えを小数点でに設定します。

表 2.20: 一列が小数点で揃えられた表セル

見出し
12.6
0.68
-123.0

表 2.21: 一列が演算子で揃えられた表セル

見出し
12 + 6
0 - 68
-123 / 0

揃え文字を指定するフィールドには、数式を用いることができないので、この方法を使って、数式中の文字で揃えることはできません。このような場合には、表 2.21 のような方法を用いる必要があります。この表は、 4×2 表として作成されており、見出しは中央揃え連結列です。第 1 列には、右揃えで演算子を含む前半部を収め、第 2 列には、左揃えで残りを収めます。演算子の周りに通常加えられる空白を得るために、第 2 列の各セルの冒頭には、小空白が加えられています。通常、表の列間に加えられる余白を削るために、

表 2.22: 表セルの揃えの例

単位	冪数	関係子
12×24 本の瓶	$10 \cdot 10^{-17}$	$\Gamma(t) \propto \Upsilon(t)$
1024×768 ピクセル	$5.78 \cdot 10^7$	$A \neq B_{\text{red}}$
32×6 cm	$-33.5 \cdot 10^4$	$\sin(\alpha) \geq \sin(\beta)$

@{}l

という \LaTeX 引数を第2列に指定します。

表 2.22は、揃え方の例をいくつか示しています。関係子で揃えるために、表 2.21の冒頭には、小空白が加えられています。

2.12.3 自製のセル書式および列書式

連結列セルを含む表がいくつもある場合には、第 2.8.2.2節のように連結部分を含む列に必要とされる幅を計算することはたいへん面倒です。そこで楽をするために、プリアンブルでセル・列書式を定義してしまえば、それを文書中のすべての表で使うことができます。書式は、以下のように定義します。

`\newcolumntype{書式名}[引数の数]{コマンド}`

書式名は、一文字だけでも構いません。 $b \cdot c \cdot l \cdot m \cdot p \cdot r$ の文字は定義済みですので使用することができませんが、これらの文字の大文字は使用可能です。

固定幅の縦横中央揃えの連結列セルの場合は、以下のセル書式を定義するとよいでしょう。

`\newcolumntype{M}[1]{>\centering\hspace{0pt}}m{\#1}`

ここで`\hspace{0pt}`は、第 2.8.1節で述べたように、最初の単語がハイフネートされる問題を回避するためのものです。すると連結列を作成する際には、表ダイアログの \LaTeX 引数で

`M{width}`

と入力するだけですみます。

連結列セルにまたがられる各セルについては、以下の書式を定義するとよいでしょう。

`\newcolumntype{S}[2]{>\centering\hspace{0pt}}m{(\#1+(2\tabcolsep+\arrayrulewidth)*(1-\#2))/\#2}`

この書式は、またがられる各セルが同じ幅になる幅を計算するのに、第 (2.1) 式を使用しています。

こうしておけば、当該セルの \LaTeX 引数に

`S{連結列セルの幅}{連結列セルがまたがる列の数}`

と入力するだけですみます。

色付き列については、以下のように定義するとよいでしょう。

```
\newcolumntype{K}[1]{>{\columncolor{#1}\hspace{0pt}}c}
```

最後の「c」は、文字列が水平方向中央揃えになった可変幅の列を生成します。こうしておいて、 \LaTeX 引数に

```
K{色名}
```

と入力します。

表 2.23 を生成するには、第 1 列と連結列に \LaTeX 引数として

```
M{2.5cm}
```

を指定し、最終列に

```
K{red}
```

そして第 2 列の各セルに

```
S{2.5cm}{2}
```

と指定します。

表 2.23: ユーザー定義の表書式を使用した表

とても長い表 セル中の単語	複数行の連結 列		c
d	e	f	g
h	i	j	k

2.12.4 罫線幅

表中のすべての罫線の罫線幅を調整するには、長さ `\arrayrulewidth` を使用します。たとえば、表 2.24 のように罫線幅を 1.5 pt にするには、表や表フロートの前に \TeX コードで `\setlength{\arrayrulewidth}{1.5pt}`

というコマンドを挿入します。こうして変更した罫線幅は、これ以降のすべての表に適用されます。ふたたび既定値を使用するには、表か表フロートの前に \TeX コードで `\arrayrulewidth` を 0.4 pt に設定してください。

表 2.25 のように横罫だけを 1.5 pt 幅にするには、表または表フロートの前に \TeX コードとして以下のコマンドを挿入してください。

表 2.24: 1.5 pt 幅罫線の表

sd		
	sd	
		sd

表 2.25: 1.5 pt 幅横罫の表

sd		
	sd	
		sd

```
\let\myHline\hline
\renewcommand{\hline}
{\noalign{\global\arrayrulewidth 1.5pt}
\myHline\noalign{\global\arrayrulewidth 0.4pt}}
```

既定幅に戻すには、表または表フロートの後に $\text{T}_{\text{E}}\text{X}$ コードとして以下のコマンドを挿入します。

```
\renewcommand{\hline}{\myHline}
```

縦罫だけを 1.5 pt 幅にするには、第 2.12.3 節の説明にしたがって、文書プリアンブルに以下のような列書式を作成してください。

```
\newcolumntype{V}{!{\vrule width 1.5pt}}
```

表 2.26 を作るには、最右列に $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 引数

VcV

を指定し、他の列には

Vc

を指定します。

表 2.26: 1.5 pt 幅縦罫の表

sd		
	sd	
		sd

2.12.5 点罫

本節の内容を出力に表示するためには、`arydshln` パッケージを導入しなくてはなりません。

第3章 フロート

3.1 はじめに

フロートは、何かしらのラベルに関連付けられた文章の塊で、固定された居場所を持ちません。フロートはもっとも適した場所を求めて、1 頁ないし 2 頁、前後にただよう（フロート）ことができます。脚注や傍注も、当該ページに註が多すぎる場合には次頁にただよって行きますので、フロートの一種です。

フロートを使うと、質の良いレイアウトを組むことができます。図表が均等に各ページに割り当てられるので、文章のない空白やページが出ることを避けることができます。このようなフローティングは、文章と図表のあいだの連関を破壊してしまうことがあるので、各フロートは文中から参照することができるようになっています。このため、フロートには連番が割り当てられます。参照については、第 3.4 節で述べられています。

フロートを挿入するには、挿入▷フロートメニューを使えば、ラベルのついたボックスすなわちキャプション差込枠が挿入されます。このラベルは、出力の際に自動的に文書言語に翻訳されます。ラベルの後ろには、キャプションの文章を入れ、画像ないし表は、フロート中、キャプションの上ないし下に別の段落として挿入します。キャプションの配置に関して詳しくは、第 3.10 節に述べられています。また、 $\text{L}_\text{Y}\text{X}$ 文書を読みやすくするために、ボックスラベルを左クリックすれば、フロートボックスを広げたり畳んだりすることができますようになっています。畳んだフロートボックスは、`float: Figure` のように、灰色ボタンに赤字ラベルで表示されます。

周囲の文章に特殊な整形がなされているときには $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ エラーが発生し得るので、これを避けるために、フロートは独立した段落として挿入することをお勧めします。

既存の図表は、選択した後に新規フロートのツールバーボタンを押せば、フロート中の図表にすることができます。

3.2 フロート型

$\text{L}_\text{Y}\text{X}$ には、第 1.2 節および第 2.5 節で各々説明されている図フロートと表フロート以外に、アルゴリズム型と折返し型のフロート型があります。

アルゴリズム 3.1 アルゴリズムフロートの例

```
for I in 1..N loop
  Sum:= Sum + A(I); /*comment*/
end loop
```

3.2.1 アルゴリズムフロート

このフロート型は、挿入▷フロート▷アルゴリズムメニューを使うと挿入することができます。これは、プログラムコードやアルゴリズムの記述に用いられ、第7章で説明されているプログラムコードリストの代わりとみなすことができます。L_YX ユーザーの手引きに述べられているように、アルゴリズム向けの環境としては、L_YX コードがあります。アルゴリズム 3.1は、アルゴリズムフロートの一例であり、下の罫線が最後の文字行のちょうど下に来るように、フロートの終わりに-4 mm の垂直空白が入れてあります。

フロートラベルは、自動的に文書言語に翻訳されるわけではありません。使用中の文書が英語でない場合には、文書プリアンブルに

```
\floatname{algorithm}{付けたい名称}
```

という行を手動で加える必要があります。ここで付けたい名称には、お使いの言語で「*algorithm*」に対応する単語を指定します。

アルゴリズム一覧を挿入するには、文書言語が L_YX のメニューに表示されている言語と同じであるならば、挿入▷一覧/目次▷アルゴリズム一覧メニューを使用します。それ以外の場合には、代わりに

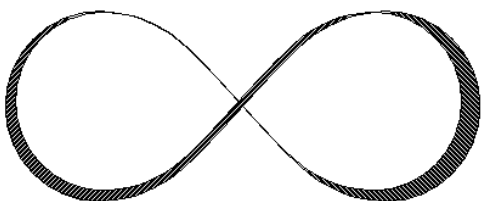
```
\listof{algorithm}{付けたい名称}
```

というコマンドを T_EX コードとして用います。ここで付けたい名称とは、「*List of Algorithms*」という単語に相当する、ご使用になりたい言語での単語です。

アルゴリズムフロートは、多くの文書クラスにおける図表フロートとは違って、「chapter.algorithm」スキームでは既定で付番されていません。このスキームでアルゴリズムフロートに連番を振るには、使用している文書のプリアンブルに

```
\numberwithin{algorithm}{chapter}
```

というコマンドを追加してください。\\numberwithin というコマンドを使用するには、文書設定の数式オプションタブで AMS math パッケージを使用するオプションを有効にしてください。

3.2.2 折返しフロート

このフロート型は、本文が図の周りに折返して入ることによって、図が段幅の一部しか占拠しないようにしたいときに用います。折返しフロートは、L^AT_EX パッケージの wrapfig を導入済みであれば、挿入▷フロート▷折返し図

図 3.1: これは折返し図フロートです。

フロート・折返し表フロートメニューで挿入することができます¹。フロートの設定は、フロートボックスを右クリックすることで変更することができます。必須の設定項目は、フロートの配置と幅です。非必須項目として、ぶら下げと行幅があります。前者は、フロートがどれくらい段落ないしページ余白に入り込むかを指定するものであり、後者は、フロートが必要とするおおよその本文行数を指定するものです。行幅は、推測するのが難しいことがよくあるので、フロートの配置に問題が生じたときのみ使用するようにした方が良いでしょう。さらに、フロートを段落内でフロートするようにするか、周囲の段落までフロートすることを \LaTeX に許可するかを指定することができます。第 3.1 図は、40 列幅%の幅と 1 cm のぶら下げを持ち、左揃えの折返しフロートを例示してあります²。

【註】折返しフロートは壊れやすいことがあります！例えば、図をページの下すぎる位置に置いてしまうと、ごちゃごちゃになって、フロートが出力に現れなかったり、他の本文の上に上書きされて出力されたりします。

原則として、

- 折返しフロートは、改頁に掛かってしまう段落に置くべきではありません。つまり、折返しフロートは、文書がほぼ完成する段階になって、どこに改頁が現れるかを予測できるようになってから、正確な位置に挿入しなくてはなりません。
- 折返しフロートは、入れ込む段落の前に単独の段落を設けて挿入するか、その段落自体の中に挿入しなくてはなりません。
- 折返しフロートを二つの段落に連続して入れると問題を引き起こす可能性があるため、二つの折返しフロートのあいだには本文のみの段落が入るようにしてください。
- 折返しフロートは、節見出しや表中に入れることは認められていません。

3.3 フロートの連番

フロートは通常、フロートのある節とは関係なく番号が振られるか、あるいは「章. 番号」や「節. 番号」といった形で番号が付けられます。これは、使用している文書クラスによって変わります。

節から独立して付けられている連番を変更するには、文書プリアンブルで

```
\renewcommand{\thetable}{\roman{table}}
```

¹ \LaTeX パッケージの導入方法は、取扱説明書の『 \LaTeX の設定』に説明されています。

²使うことのできる単位は付録 A に収録してあります。

というコマンドを使うと良いでしょう。`\thetable` は表番号を印字するコマンドで、図フロートの場合であれば、代わりに `\thefigure` コマンドを使用します。上のコマンドで使用されている `\roman` は、表番号を小文字のローマ数字で印字するコマンドです。

連番の付け方をたとえば「小節. 番号」に変更するには、プリアンブルで


```
\numberwithin{table}{subsection}
```

というコマンドを使用してください。`\numberwithin` コマンドを使用可能にするためには、文書設定の数式オプションタブで、AMS math 数式パッケージを使用するを使用します。

連番コマンドについての詳細と大切な注意点については、第 4.2.1 節もご覧ください。

3.4 フロートの参照

フロートを参照するには、挿入▷ラベルメニューかツールバーボタンを使用して、フロートのキャプションにラベルを挿入してください。 `fig:This-is-a` のような灰色のラベルボックスが挿入され、ラベルウィンドウがポップアップして、ラベルの字句を入力するよう促します。 \LaTeX は、キャプションの最初の数単語に前置句をつけたものを候補として示します。前置句はフロート型に依存し、たとえば図フロートには「fig:」という前置句がつけられます。

ラベルは、参照のための名称兼位置標として用いられます。挿入▷相互参照メニューかツールバーボタン  を使えば、ラベルを参照することができます。 `参照:sec:貢献` のような灰色の相互参照ボックスが挿入され、文書中のすべてのラベルを表示した相互参照ウィンドウが現れます。複数の \LaTeX 文書が開かれている場合には、ダイアログの上方にあるドロップリストから作業中の文書を選択してください。ここでラベルを整序してから選ぶこともできます。出力中では、相互参照ボックスの位置にはフロート番号が表示されます。

相互参照名と数字の間に改行が入るのを防ぐために、両者の間の空白には保護された空白を用いた方が良いでしょう。もし相互参照が存在しないラベルを参照している場合には、出力では、参照の代わりに二つの疑問符が表示されることになります。

ラベルは、ラベルボックスをクリックすることによって、いつでも変更することができます。変更されたラベルへの参照は、自動的に新しいラベル文へのリンクに変更されますので、リンクの更新に手間を割く必要はありません。

相互参照ウィンドウのラベルへ移動ボタンは、参照されているラベルの前にカーソルを移動します。すると、ボタンの文字が戻るに変わるので、相互参照にカーソルを戻すのに使用することができます。相互参照ボックスを右クリックしても、カーソルを参照されているラベルの前に移すことができますが、この場合には元に戻ることができません。

3.4.1 相互参照の書式

相互参照には、以下の 6 種類があります。

<参照>: これはフロート番号を表示します。これが既定値です。例: 1.2

(<参照>): これはフロート番号を 2 つのかっこの中に表示します。これはふつう数式を、とくに参照名「Equation」を省略して参照する際に用いられる様式です。例: (2.1)

<参照ページ>: これはページ番号を表示します。例: Page 4

on page <参照ページ>: これは「on page」という文とページ番号を表示します。例: on page 4

<参照> on page <参照ページ>: これはフロート番号と「on page」という文とページ番号を表示します。例: 1.2 on page 4

整形された参照: これは自製の相互参照書式で出力します。

【註】この機能は、 \LaTeX パッケージ `prettyref` または `refstyle` が導入されている場合のみ使用することができます。

メニュー文書▷設定▷文書クラスでオプション相互参照に (`prettyref` ではなく) `refstyle` を使用を設定すると、この機能で使用する \LaTeX パッケージを選択することができます。書式は、文書プリアンプルに `\newrefformat` コマンド (`prettyref` の場合) あるいは `\newref` コマンド (`refstyle` の場合) を書き加えることによって指定することができます。たとえば、図への参照 (これは短絡ラベル「fig」を持ちます) をすべて定義し直すには、

```
\newref{fig}{refcmd={Image on page \pageref{#1}}}
```

というコマンドを用います。書式についての詳細は、パッケージの取扱説明書 [16, 17] をご覧ください。

名称参照: これは、右のように参照のキャプションを出力します。図フロート

<参照ページ> は、ラベルが同一あるいは前後ページにある場合には、ページ番号を出力しません。その代わりに「on this page」のような文章が表示されます。様式 <参照> on page <参照ページ> では、ラベルが同一ページにある場合には、ページに関することは出力されません。

出力中で参照されている頁番号と文書中の位置は、 \LaTeX が自動的に計算します。参照の種類は、相互参照ボックスをクリックすると現れる相互参照ウィンドウ中の書式フィールドで選ぶことができます。

Note: It is recommended to use the \LaTeX -package `refstyle` because `prettyref` does not know all of \LaTeX 's possible label shortcuts³ and is not internationalized.

³“fig:” is for example \LaTeX 's shortcut for labels in figure captions.

3.4.2 参照名の自動命名

文書設定ダイアログの PDF 特性で有効にされた `hyperref` L^AT_EX パッケージは、相互参照が参照されたフロート（あるいは節のような文章中の箇所）の名称を自動的に取り入れてくれるという、たいへん便利な機能を提供します。つまり、図への参照毎に前に「図」と書き込む必要はなくなるのです。この機能を使用するには、`hyperref` を有効にして、

```
\AtBeginDocument{\renewcommand{\ref}[1]{\mbox{\autoref{#1}}}}
```

という行を L^AT_EX プリアンブルに挿入してください。たとえば「section」の代わりに「sec.」としたいなど、既定の参照名以外のものが欲しい場合には、

```
\addto\extrasenglish{\renewcommand{\sectionautorefname}
{sec.\negthinspace}}
```

のようにプリアンブルに挿入することで参照名を再定義することができます。英語以外の文書言語を使用している際には、

`\extrasenglish` を `\extras***` に置き換えてください（***は使用中の言語名）。

自動命名を行いたいけれども、ある参照型以外、たとえば数式以外に対して行いたいときには、以下のようなプリアンブルコードを書いてください。

```
\newlength{\abc}
\settowidth{\abc}{\space}
\addto\extrasenglish{\renewcommand{\equationautorefname}
{\hspace{-\abc}}}
```

この話題に関する詳細は、`hyperref` の説明書 [10] をお読みください。

【注】参照の自動命名は、相互参照を整形参照様式（第 3.4.1 節）で使用しているときには、使用することができません。

取扱説明書数式篇は、参照の自動命名が使用されている一例です。

3.4.3 参照位置

文書設定ダイアログの PDF 特性で、`hyperref` を使用して出力中で相互参照をリンクするようにした場合には、画像フロート参照をクリックすると、画像ラベルに移動できるようになります。キャプションは、スクリーン上のテキストの最初の部分なので、スクロールしないと画像を見ることができません。これは、参照リンク標がラベルの位置に置かれるからです。L^AT_EX パッケージの `hyperref` の一部である `hyperref` パッケージを使うと、リンク標はフロートの最初に置かれます。この機能を図フロートに対して使用するには、文書プリアンブルに

```
\usepackage[figure]{hypcap}
```

という行を書いて、`hypcap` を読み込んでください。`hypcap` をすべてのフロート型に対して使用することもできますが、安定性の問題から推奨できません。詳しくは、`hypcap` の説明書 [9] をご覧ください。

【註】`hypcap` は内部図への参照には効力がありません。

3.5 フロートの配置

フロートボックスを右クリックすると、ダイアログが開いて、ここで \LaTeX がフロートを配置するのに使う配置オプションを変更できるようになります。

段をまたぐオプションは、二段組文書でのみ有効です。これを選択すると、フロートは段の中のみには収められないで、ページの両段にまたがって広がります。

横倒しに回転オプションは、フロートを回転するのに用います。第3.6節をご参照ください。

フロートダイアログでは、既定の配置を使うオプションを無効にしたときには、特定のフロートの配置を設定するのに以下のオプションの一つ以上を設定することができます。

可能ならば現在位置に： フロートを挿入位置に配置することを試みます

ページ上部： フロートを現在ページの上部に配置することを試みます

ページ下部： フロートを現在ページの下部に配置することを試みます

フロートを独立したページに： フロートを独立したページに配置することを試みます

\LaTeX は、つねに上記のオプション順を用います。すなわち、既定の配置を使用すると、 \LaTeX はまず可能ならば現在位置にを試み、次にページの上部、その後他のオプションを試みます。既定値を使わないときには、 \LaTeX は有効化されたオプションのみを使用しますが、やはり同じ順序でこれらを試みます。これら四つの配置がいずれも無理な場合は、フロートを内部的に次のページにおいてみて同様の手続きがうまくいくかどうかを試みます。

既定では、各オプションはそれぞれ以下の規則を持っています。

ページ上部 は、フロートのページに占める割合が7割以下の場合のみ、ページ上部に配置します (`\topfraction`)。

ページ下部 は、フロートのページに占める割合が3割以下の場合のみ、ページ下部に配置します (`\bottomfraction`)。

フロートを独立したページ には、フロートのページに占める割合が5割以下の場合のみ、複数のフロートを同一ページに配置します (`\floatpagefraction`)。

これらの規則に従いたくないときは、 \LaTeX の規則を無視するオプションを追加することによって、これらを無視することができます。

上記説明の後ろに括弧でくくってある \LaTeX コマンドを使うと、これらの規則を定義しなおすこともできます。たとえば、往々にして小さすぎる、ページ下部規則の既定値を、ページの5割に変えるには、文書プリアンブルに

```
\renewcommand{\bottomfraction}{0.5}
```

という行を加えてください。

ときにはフロートを、挿入位置に正確に、しかもかならず配置させたい場合があるかもしれません。このような場合には、何としても現在位置に置くオプションを使用します。このオプションはできる限り使わないようにし、使う場合でも印刷する直前にまで文書が仕上がってから、使用するようにしてください。なぜならば、このオプションを使うことによって、フロートはもはや「漂う」ことができなくなりますから、文書に変更を加えてしまうと、ページのレイアウトが往々にして破壊されてしまうからです。

折返しフロートは、つねにどれかの段落の本文で囲まれますので、配置オプションはありません。

対応する節がページの中程から始まるのというのに、フロートがページの先頭に置かれてしまうために、読者がフロートを前節の一部であると勘違いしてしまうという問題に直面することがあります。この問題を避けるには、 \LaTeX コマンドの `\suppressfloats` を使用することができます。これは、これが挿入されたページの特定のフロートの配置を抑制するので、節が始まる前にフロートが設定されてしまうのを避けるのに使用することができます。これを用いるには、文書プリアンブルに以下のコマンドを加えてください。

```
\let\mySection\section
\renewcommand{\section}{\suppressfloats[t]\mySection}
```

これと同じものは、章や小節など、すべての節見出しに用いることができます。この定義は、小々節のような小さい文字の部分では、 \LaTeX が適当な配置を見つけるのに失敗することがあるので、これらに対する定義はお勧めしません。

場合によっては、すべての図表を文末に置く必要があることがあります。この目的のために、 \LaTeX パッケージ `endfloat` が開発されました。このパッケージは、すべての図表を文末に独立した節として置きます。元のフロート位置には、“[Figure 3.2 about here.]” というようなヒント文が挿入されます。`endfloat` パッケージは、プリアンブルに

```
\usepackage[options]{endfloat}
```

という行を加えることで読み込まれます。このようにして生成された図表の節を整形するために、さまざまなパッケージオプションがあります。詳細は、`endfloat` 取扱説明書 [6] をご参照ください。

【註】`endfloat` は、ヒント文の自動翻訳を行わないので、手動で行わなくてはなりません。[6] の第4節をご参照ください。

【註】現在のところ、`endfloat` にはキャプションにドイツ語の「&」が含まれているときにうまく動かないバグがあります。この場合には、「&」の代わりに \TeX コードで「`\ss`」コマンドを使用してください。

フロートの割付に関する詳細は、 \LaTeX 関連書 [1, 2, 3] をご覧ください。

3.6 回転フロート

特に横長の表などの場合、フロートを回転させたいことがあります。キャプションを含むフロート全体を回転させたいときは、フロートボックスを右クリックして、横倒しに回転オプションを有効にします。

回転フロートは、つねに独立したページ（二段組文書の場合は独立した段）に配置されます。フロート設定の段をまたぐオプションを有効にすれば、複数の段にまたがるようにすることができます。また、回転フロートは、外側余白方向から読むことができるような形に回転されます。全ページにわたって回転方向を強制したい場合には、文書クラスオプションに `figuresleft` オプションあるいは `figuresright` オプションを加えることができます。

回転フロートへの相互参照は、通常フロートの場合と同じです。キャプションの形式も同じです。表 3.1 が回転表フロートの例です。

【註】すべての DVI ビューアが回転フロートを表示できるわけではありません。

3.7 従属フロート

従属フロートは、図が複数の画像から成るような場合に使用します。これは、フロートの中に新たなフロートを挿入することで作ることができます。従属フロートの揃えは、表 3.2 と表 3.3 に示されているように段落と同様に指定することができます。

従属フロートの参照は、通常フロートの参照と同様にすることができます。例：表 3.3a と表 3.3b は表 3.2 の従属フロートです。

3.8 横並びのフロート

図 3.1 や図 3.3 のようにフロートを横並びに配置するには、ひとつのフロートのみを使用し、その中に二つの `minipage` ボックスを挿入します⁴。幅は 45-50 列幅%とし、各 `minipage` のボックス配置を下に設定します。`minipage` ボックスには、フロートと同じように画像とキャプションを入れます。フロートと唯一違う点は、画像単位列幅%がここでは `minipage` ボックスの幅によって計算されることです。

3.9 キャプションの書式設定

キャプション環境は、フロートの既定の段落環境です。キャプションは、 \LaTeX 画面上では、例えば「図#：」のようなラベルの後に、キャプション文を伴って表示されます。ここで

⁴Minipage は第 5.4 節に説明があります。

表 3.1: 回転した表

試験	甲	乙	丙	丁
----	---	---	---	---

表 3.2: 従属表を横に並べたもの

試験	甲	乙	丙	丁
----	---	---	---	---

(a) これは従属表 a です。

丁	丙	乙	甲	試験
---	---	---	---	----

(b) これは従属表 b です。

表 3.3: 従属表を縦に並べたもの。(a) は 4 セルから成る表、(b) は 5 セルから成る表。

試験	試験	試験	試験
----	----	----	----

(a)

甲	乙	丙	丁	戊
---	---	---	---	---

(b)

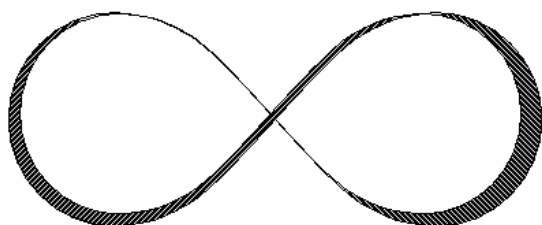


図 3.2: 左側のフロート

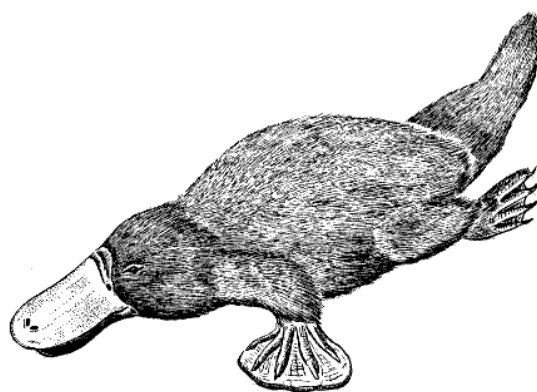


図 3.4: 右側のフロート

「#」には、実際には参照番号が入ります。既定の設定では、ラベルと番号はキャプション文と同じフォントになっており、ラベルと文を分けるために、番号の後にコロンが入ります。このキャプション書式は、すべての文書書式に適しているわけではありません。

既定のキャプション書式を変更するには、文書プリアンブルに

```
\usepackage[format definition]{caption}
```

という行を加えて、 \LaTeX パッケージ `caption` を読み込んでください。たとえば、ラベルと番号をサンセリフのボールド体にして、表キャプションを、この文書のように常に表の上に置くようにするには、

```
\usepackage[labelfont={bf,sf}, tableposition=top]{caption}
```

というコマンドを使用してください。また、各フロート型毎に別々のキャプション書式を定義することもできます。この場合には、`caption` パッケージを書式関連オプション抜きで読み込み、文書プリアンブルに

```
\captionsetup[フロート型]{書式定義}
```

というコマンドを使用して、個別の書式を指定してください。たとえば、図 3.4 と表 3.4 に使われているキャプション書式は、文書プリアンブルに以下のコマンドを指定することで作り出すことができます。

```
\usepackage[tableposition=top]{caption}
\captionsetup[figure]{labelfont={tt}, textfont=it, indentation=1cm,%
                      labelsep=period}
\captionsetup[table]{labelfont={bf,sf}}
```

【註】オプションのうち、`tableposition=top` は、文書クラスに KOMA-script が使われているときには、無効になります。この場合には、文書クラスオプション `captions=tableheading` を指定しなくてはなりません。

`caption` パッケージについての詳細は、その取扱説明書 [5] をご覧ください。

ラベル名を、たとえば「図」から「画像」に変更するには、以下のプリアンブルコマンドを使用してください。

```
\renewcommand{\fnum@figure}{画像~\thefigure}
```

なお、`\thefigure` は表番号を挿入し、「~」は保護された空白を入れます。

KOMA-script 文書クラス (`article` (KOMA-script) ・ `book` (KOMA-script) ・ `letter` (KOMA-script) ・ `report` (KOMA-script)) を使用している場合には、`caption` パッケージの代わりに KOMA-script のビルトインコマンド `etkomafont` を使用することもできます。たとえば、キャプションラベルをボールド体にするには、以下のコマンドを文書プリアンブルに加えます。

```
\setkomafont{captionlabel}{\bfseries}
```

`\setkomafont` についての詳細は、KOMA-script の取扱説明書 [?] をご覧ください。

3.10 キャプションの配置

通常、キャプションの配置原則は、

図の場合: キャプションを図の下に置く

表の場合: キャプションを表の上に置く

という風になっています。L^AT_EX の標準クラスでは、キャプションを表の上に置くことは、残念ながらサポートされていません。したがって、article・book・letter・report といった文書クラスを使用している時には、キャプションと表の間に余白が置かれなくなってしまう。本来あるべき余白を挿入するには、お使いの文書プリアンブルで L^AT_EX パッケージ caption を読み込むコマンドに

```
tableposition=top
```

というオプションを追加してください⁵。KOMA-script 文書クラス (article (KOMA-script)・book (KOMA-script)・letter (KOMA-script)・report (KOMA-script)) をお使いの場合には、caption パッケージにオプションを与える代わりに、文書クラスオプション captions=tableheading を設定することもできます。

また、キャプションは、図表の横に置くこともできます。これを実現するには、文書プリアンブルに

```
\usepackage[option]{sidecap}
```

という行を入れて、L^AT_EX パッケージ sidecap を読み込まなくてはなりません。オプションを与えなければ、ページの外側余白の側に—偶数ページでは右側、奇数頁では左側に—キャプションが置かれます。innercaption オプションを与えれば、内側余白側に配置を変更することができます。配置をつねに右側ないし左側に来るように強制するには、rightcaption ないし leftcaption オプションを用います。

L_YX でフロートのキャプションを横に配置するには、以下のコマンド群を文書プリアンブルに追加する必要があります。

```
\newcommand{\TabBesBeg}{\%
\let\MyTable\table
\let\MyEndtable\endtable
\renewenvironment{table}{\begin{SCtable}}{\end{SCtable}}}

\newcommand{\TabBesEnd}{\%
\let\table\MyTable
\let\endtable\MyEndtable
```

⁵caption パッケージについての詳細な情報は、第 3.9 節をご覧ください。

```

\newcommand{\FigBesBeg}{%
\let\MyFigure\figure
\let\MyEndfigure\endfigure
\renewenvironment{figure}{\begin{SCfigure}}{\end{SCfigure}}}

\newcommand{\FigBesEnd}{%
\let\figure\MyFigure
\let\endfigure\MyEndfigure}

```

これらのコマンドは、キャプションが脇に来るようにフロートを再定義することを可能にするものです。図フロートの場合は、

`\FigBesBeg`

というコマンドを、フロートの前に \TeX コードで入れてください。本来のフロート定義に戻すには、フロートの後に、 \TeX コードで

`\FigBesEnd`

というコマンドを挿入してください。

表フロートに対しては、対応するコマンド

`\TabBesBeg` および `\TabBesEnd`

を使用してください。図 3.6 と表 3.5 がキャプションを脇に設定した例です。

これらの例において、表フロートの場合には、キャプションの文章がフロート上部に表示され、図フロートではフロート下部に表示されることが確認いただけると思います。これを変更するには、文書プリアンプルかフロート直前に、 \TeX コードとして

`\sidecaptionvpos{フロート型}{配置}`

というコマンドを使用します。ここで「フロート型」は `figure` か `table` であり、「配置」は、上揃えの場合は `top` の

「`t`」、中央揃えの場合は `center` の「`c`」、下揃えの場合は `bottom` の「`b`」のいずれかになります。たとえば、図フロートのキャプションを縦方向中央揃えとしたい場合には、

`\sidecaptionvpos{figure}{c}`

というコマンドを使用します。これは、図 3.7 で使用されています。

`sidecap` パッケージについての詳細は、[15] の説明書をご参照ください。

【註】 第 3.4.3 節に説明がある \LaTeX パッケージ `hypcap` は、キャプションを脇に表示したフロートに対しては、効果を持ちません。

3.11 フロート一覧

文書中の各節が列挙されたものである目次と同様、文書中の図など、すべてのフロート型に一覧があります。これらは、挿入▷一覧/目次メニューで挿入することができます。

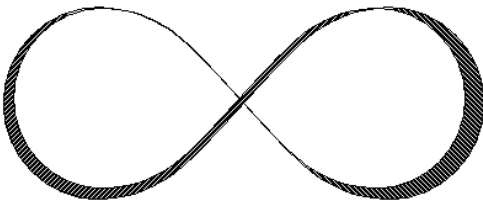


図 3.5. これは、別のキャプション書式を例示するための、一行よりも長い図キャプションです。ここでは、自製キャプション書式が使用されています。

表 3.4: これは、別のキャプション書式を例示するための、一行よりも長い表キャプションです。ここでは、この文書での表キャプションの標準書式が使用されています。

い	ろ	は	に	ほ
---	---	---	---	---



図 3.6: これは図の横にあるキャプションです。

い		ろ		は
	に		ほ	
へ		と		ち
	り		ぬ	

表 3.5: これは表の横にあるキャプションです。




図 3.7: これは図の横にある縦方向中央揃えのキャプションです。

一覧の見出しは、フロートのキャプションないし短縮タイトル、フロート番号、これらが文書中で現れるページ番号で構成されます。

本文書の終わりに、図一覧と表一覧があります。

第4章 注釈

4.1 L_YX 注釈

注釈は、ツールバーボタンか、挿入▷注釈メニューで挿入することができます。注釈には、以下の三つの型があります。

L_YX 注釈 この注釈型は、内部用途の注釈で出力には現れません。注釈ボックスは、以下のように表示されます。

Note This is text in a note box that doesn't appear in the output.

コメント この注釈も出力には現れませんが、ファイル▷書き出し▷L^AT_EX (pdf_latex) / (平文) メニューを使って文書を L^AT_EX にエクスポートする際には、L^AT_EX コメントとして出力されます。注釈ボックスは、以下のように表示されます。


Comment This is text in a note box that only appears as comment in LaTeX-files.

淡色表示 この注釈は、薄灰色の文章として出力に現れます。注釈ボックスは、以下のように表示されます。

Greyed out This is text of a comment that appears in the output as grey text.

これは、出力に薄灰色の文章として現れるコメントに書かれた文章¹です。

この例から分かるように、淡色表示注釈の最初の行は少し字下げされるほか、淡色表示注釈には脚注を入れることができます。

注釈を挿入するのにツールバーボタンを使用すると、L_YX 注釈が挿入されます。注釈ボックスを右クリックすれば、五つの注釈型を切り替えることができます。既存の平文を注釈に変えるには、平文を選択して、注釈ツールバーボタンをクリックしてください。注釈を平文に変更するには、カーソルを注釈の先頭に置いて Backspace キーを押すか、カーソルを注釈の最後に置いて Delete キーを押してください。

¹これは、淡色表示注釈内に脚注を用いた例です。


淡色注釈の文字色は、プリアンブルに以下のコマンドを置くことで変更することができます。


```
\renewenvironment{lyxgreyedout}
{\textcolor{色彩名}\bgroup}{\egroup}
```

使用できる色彩名と、自製色の定義の仕方は、第 2.11 節で説明されています。

本文書で青で表示されている注釈は、青文字に設定した淡色表示注釈を使用しています。

4.2 脚注

脚注は、ツールバーボタンか、挿入▷脚注メニューを使って挿入することができます。

すると、脚注文を挿入したところに、のような脚注ボックスが現れます。もし既存の文章を脚注に変えたいならば、その文章を選択して、脚注ツールバーボタンをクリックしてください。脚注を平文に変えるには、カーソルが脚注の先頭にある時に Backspace キーを押すか、カーソルが脚注の最後にある時に Delete キーを押します。

これが脚注の例です²。

出力に於いては、脚注は、文中の脚注ボックスを置いた位置に上付き数字として表示され、脚注文は現ページ底部に置かれます。脚注番号は L^AT_EX が計算を行い、連番が付けられます。脚注番号が章毎にリセットされるか否かは、使用する文書クラスに依ります。

脚注もフロート同様、参照することができます。脚注にラベルを挿入し、第 3.4 節に説明されているようにして、本文中でこのラベルを相互参照してください。

これは脚注 2 の相互参照です。

L^AT_EX では、技術的要因により、表中の脚注は印字されません。しかし、それを行うには他の方法があり、まず脚注の代わりに T_EX コードで `\footnotemark{}` コマンドを入れてください。脚注の本文は、表の後で T_EX コード `\footnotetext` コマンドの引数として入力します。

以下の表ではコマンド

```
\footnotetext{出力される脚注の表}
```

を用いています。

これは例文です。 ³
これは例文です。 ⁴

長尺表中の脚注は、第 2.6.1 節で述べられています。

ミニページ中の脚注は出力されますが、ミニボックスは頁中の頁のようなものなので、ボックス内に独立した連番として出力されます（ミニページについての詳細は、第 5.4 節

²これは脚注の一例です。

⁴出力される脚注の表

を参照)。通常の脚注のように頁底部に出力されるミニページを得るには、やはり`\footnotemark{}`と`\footntoetext`による方法を用いてください。

これはミニページ中の例文です。^a

これはミニページ中の例文です。⁵

^aミニページ中の脚注です。

毎回脚注本文を出力せずに、同じ脚注を何度か参照したい場合には、 \TeX コードでコマンド`\footnotemark[番号]`を使用してください。

文章を書いている途中には、繰り返す脚注の数はわからないので、その番号は保存しておかなくてはなりません。以下の脚注例では、脚注 2 の後に、 \TeX コードで

```
\newcounter{MyRepeatFoot}
\setcounter{MyRepeatFoot}{\thefootnote}
```

というコマンドを挿入しています。そして脚注マークは、

```
\footnotemark[\theMyRepeatFoot]
```

で作成します。

これは繰り返し脚注の例です。²

直前の脚注を繰り返すには、脚注番号の保存を行わないで、直ちに

```
\footnotemark[\thefootnote]
```

を使用してください。

これは繰り返し脚注の例です。⁵

4.2.1 脚注の付番

各節毎に脚注番号を 1 にリセットするには、文書プリアンブルに

```
\@addtoreset{footnote}{section}
```

というコマンドを加えてください。

以下のプリアンブルコマンドを使うと、脚注の付番様式を小文字ローマ数字に変更することができます。

```
\renewcommand{\thefootnote}{\roman{footnote}}
```

これは、ローマ数字を使った脚注です^{vi}。

付番様式を大文字ローマ数字に変更するには、上記コマンドの`\roman`を`\Roman`に置き換えてください。脚注を大文字または小文字ラテン「文字」で付番するには、それぞれ

⁵ミニページ外の脚注です。

^{vi}これは、ローマ数字を使った脚注の例です。

`\Alph` または `\alph` を使用してください。記号を使って付「番」するには、`\fnsymbol` を使ってください。

【註】この付番方法には1文字しか使えないので、ラテン文字を使うと26個の脚注しか番号を振ることができません。

【註】記号を使うと9個の脚注しか番号を振ることができません。

付番様式を変更した後、既定様式に戻すには、上記コマンドの`\roman`を`\arabic`に置き換えて使用してください。

脚注を「章番号.脚注番号」スキームで付番したいときには、使用中文書のプリアンブルに
`\numberwithin{footnote}{chapter}`

というコマンドを加えてください。`\numberwithin` コマンドを使用できるようにするためには、文書設定の数式オプションタブでAMS math パッケージを使うオプションを有効にしてください。

これは、脚注のもう一つの例です^{4.7}。

【註】`\numberwithin` は、脚注番号を常にアラビア数字で出力します。それよりも前にアラビア数字以外に再定義されたものは無効となります。

また、例えば「章番号.`\Roman{footnote}`」スキームを得るには、`\numberwithin` の代わりに以下のコマンドを使用してください。

```
\renewcommand{\thefootnote}{\thechapter.\Roman{footnote}}
```

4.2.2 脚注の配置

同一ページに複数の脚注がある場合、それらはページ底部に縦方向の余白を空けることなく、並べられます。これを読みやすくするには、例えば以下のようなコマンドをプリアンブルに足して、1.5 mm の余白を追加することもできます。

```
\let\myFoot\footnote
\renewcommand{\footnote}[1]{\myFoot{#1\vspace{1.5mm}}}
```

二段組文書では、脚注は各段底部に表示されます（図 4.0）。図 4.1 のように、脚注を右段底部にのみ表示するようにするには、文書プリアンブルに

```
\usepackage{ftnright}
```

というコマンドを書いて、 \LaTeX パッケージ `ftnright` を読み込みます。

科学的文献に於いては、図 4.2 のように、節の末尾に別段落を設けて、脚注をまとめて出力することも普通に行われます。これは「後注」と呼ばれるものです。文書中で脚注の代

^{4.7} これは、「章番号.脚注番号」スキームで番号を振られた脚注です。

<p>Sei nun S unser normiertes Ausgangssignal und P die Phasenverteilungsfunktion, so ergibt sich die Beziehung</p> $S(t) = S_0(t) \int_{-\infty}^{\infty} P(\phi, t) e^{i\phi} d\phi \quad (2)$ <p>wobei S_0 das Signal ohne Gradient ist und die Normierungsbedingung</p>	<p>Das Spektrum wird fouriertransformiert. Die Fouriertransformation wird verwendet, um die überlagerten Signale (Netzwerk, Lösungsmittel) zu trennen. Nachdem wir die Phasenverschiebung bestimmen konnten, interessiert uns nun das Aussehen des Ausgangssignals. Im Experiment haben wir es mit sehr vielen Teilchen zu tun, so dass</p>
<p>¹Fourier transformation ²Phase distribution function</p>	<p>³Fourier transformation ⁴Phase distribution function</p>
1	

図 4.1: 二段組文書の標準的な脚注配置

<p>man über alle Phasen integrieren muss. Sei nun S unser normiertes Ausgangssignal und P die Phasenverteilungsfunktion, so ergibt sich die Beziehung</p> $S(t) = S_0(t) \int_{-\infty}^{\infty} P(\phi, t) e^{i\phi} d\phi \quad (2)$ <p>wobei S_0 das Signal ohne Gradient ist und die Normierungsbedingung $\int_{-\infty}^{\infty} P(\phi, t) d\phi = 1$ gilt. Nun dürfen</p>	<p>Das Spektrum wird fouriertransformiert. Die Fouriertransformation wird verwendet, um die überlagerten Signale (Netzwerk, Lösungsmittel) zu trennen. Nachdem wir die Phasenverschiebung bestimmen konnten, interessiert uns nun das Aussehen des</p>
	<p>1. Fourier transformation 2. Phase distribution function 3. Fourier transformation 4. Phase distribution function</p>
1	

図 4.2: L^AT_EX パッケージ ftnright を使用したときの二段組文書の脚注配置

わりに後注を使用するには、文書▷設定▷モジュールメニューで脚注から後注へを読み込んでください。まとめた注を出力するには、節ないし章末に T_EX コードで

```
\theendnotes
```

というコマンドを挿入してください。

後注の段落見出しは、文書言語に自動的に翻訳されることはありませんので、手動で翻訳しなくてはなりません。プリアンブルで

```
\renewcommand{\notesname}{Anmerkungen}
```

というコマンドを使えば、既定の英語名「Notes」を独語訳「Anmerkungen」に変えることができます。

man über alle Phasen integrieren muss. Sei nun S unser normiertes Ausgangssignal und P die Phasenverteilungsfunktion, so ergibt sich die Beziehung

$$S(t) = S_0(t) \int_{-\infty}^{\infty} P(\phi, t) e^{i\phi} d\phi \quad (2)$$

wobei S_0 das Signal ohne Gradient ist und die Normierungsbedingung $\int_{-\infty}^{\infty} P(\phi, t) d\phi = 1$ gilt. Nun dürfen

Notes

¹Fourier transformation

²Phase distribution function

³Fourier transformation

⁴Phase distribution function

図 4.3: 後注 – 注を節ないし章末に別段落として印字します。


第 4.2.1 節で述べた脚注の付番と同様に、後注の付番も変更することができます。 `\the-footnote` コマンドの代わりに `\theendnote` を用いてください。後注番号をリセットするには、第 4.2.1 節で述べたコマンド `\@addtoreset` の、パラメータ `footnote` を `endnote` に変えて、使用してください。

後注の印だけを作成するには、第 4.2 節で説明した `\footnotemark` コマンドの代わりに、`\endnotemark[番号]` コマンドを使用してください。

脚注はページ余白に置くこともでき、脚注文の揃えも変更することができます。詳細は、`LaTeX` パッケージ `footmisc`[8] を参照してください。

脚注の整形にまつわるその他多くの話題が、`LaTeX` 関連書籍 [1, 2, 3] に載っています。

4.3 傍注

`LaTeX` 内において、傍注注釈は、脚注のような外見と挙動を示します。これは挿入 ▶ 傍注メニューからツールバーボタン  で挿入することができます。すると、紫字で「傍注」と書かれた灰色のボックスが現れて、そこに傍注文を入れることができます。

注文です。横にあるのが傍注です。

片面文書においては、傍注は右側に表示され、両面文書においては外側余白—偶数ページでは左側、奇数ページでは右側—に表示されます。傍注文は、外側余白の反対側に—注が左余白に現れるときには右揃えに—寄せられます。傍注文の一行目は、傍注が挿入された文書箇所と同じ行に来るように配置されます。

傍注を内側余白に置くには、傍注の前に `TeX` コードで

`\reversemarginpar`

というコマンドを書き加えてください。この新しい配置は、以後のすべての傍注に対して効力があります。

これは内側注です。

【註】内側余白には十分な空白がないことがよくあり、出力で傍注が正しく表示されないことがあります。

既定の配置に戻すには、 \TeX コードで

`\normalmarginpar`

というコマンドを書き加えてください。【註】`\reversemarginpar` コマンドが既に挿入されている段落内では、このコマンドは無視されます。

第 2.8.1 節で述べられている事例と同様に、長い単語が傍注文の最初の単語として来るとハイフネートが行われません。これを回避するには、その単語の前に 0 pt 水平空白を挿入してください。

【註】通常、傍注は表やフロート、脚注の中では使用することができません。


本節の以下の部分を出力に表示するには、 \LaTeX パッケージ `marginnote` を導入する必要があります。

AVeryLongM
(とっても長
段落中の単
イフネート
い例。
AVeryLongM
ParWord (こ
長い傍注段
語) がハイ
される例。

第5章 ボックス

5.1 はじめに

ボックスは、一塊の文章を整形するのに使用します。ボックスは、多言語文書を書くのに使用したり（第5.4節参照）文章を囲ったり（第5.2.3節参照）単語がハイフネートされるのを予防したり（第5.6.1節参照）文章を揃えたり（第5.6.2節参照）あるいは文章の背景色を設定したり（第5.7節参照）するのに使用します。

ボックスは、挿入▷ボックスメニューからツールバーボタンを使って挿入することができます。すると、**Box (Minipage)** のようにボックス（minipage 環境）というラベルのついた灰色の箱形が挿入されます。ボックス型は、この箱形を右クリックすると出てくるメニューで指定することができます。設定メニューを選択すると現れるボックスダイアログでは、parbox コマンドと minipage 環境の二つの内部ボックス型を選択することができます。minipage 型は、新規ボックスの既定値となっており、第5.4節で説明します。parbox 型については第5.5節、makebox 型については第5.6.1節で述べます。

ボックスは付番されないので、フロートや脚注のように参照することはできません。

【註】ボックスは、箇条書き（記号）環境や箇条書き（記述）環境の単独要素とすべきではありません。

【註】理由は不明ですが、枠付きボックスを使用している場合には、内部ボックス型はなしにしか設定することができません。内部ボックス型や枠のないボックスについては、第5.6.1節で説明します。

5.2 ボックスダイアログ

5.2.1 寸法

ボックスダイアログでは、幅と高さの各フィールドでボックスのジオメトリを調整することができます。ジオメトリに使用することのできる単位は、表 A.1 に説明されています。高さフィールドには、加えて以下の寸法を使用することができます。

階層 これは平文の「高さ」です。ボックス中に複数行がある場合には、全体としての階層は無視されます。

高さを 1
階層に設
定した
ボックス

高さ これはボックス内部にある文章の高さです。この寸法でたとえば 2 の値を与えると、

ボックスの高さを文章の 2 倍の高さに設定します。

高さを 2 高さに設
定したボックス

全高 これは「高さ」+「階層」です。

高さを 1 全高に設
定したボックス

幅 これはボックスの幅を高さに設定します。

高さを 1
幅に設定
したボッ
クス

5.2.2 文字揃え

内部ボックスを選んだ場合には、ボックスの垂直揃えを以下のいずれかに設定できます。

上 これは例示のための本文行です。

このボッ
クスは上
揃えです。

これは例示のための本文行です。

中央 これは例示のための本文行です。

このボッ
クスは中
央揃え
です。

これは例示のための本文行です。

下 これは例示のための本文行です。

このボッ
クスは下
揃えです。

これは例示のための本文行です。

【註】同一行に二つのボックスがあり、一方だけに影付きなどが設定されている場合には、ボックスの垂直文字揃えが出力中で無効となる場合があります。

ボックスを独立した段落においた場合には、ボックスの水平文字揃えは、 L_YX の段落ダイアログから設定できます。

内部ボックスを設定したときには、ボックスの内容を以下のように垂直揃えを行うことができます。

上 これは例示のための本文行です。

このボッ
クスの文
章は上揃
えです。

これは例示のための本文行です。

中央 これは例示のための本文行です。

このボッ
クスの文
章は中央
揃えです。

これは例示のための本文行です。

下 これは例示のための本文行です。

このボッ
クスの文
章は下揃
えです。

これは例示のための本文行です。

広げる これは例示のための本文行です。

このボック
スの

文章は広げて

あります。

これは例示のための本文行です。

ボックスの内容を広げるには、ボックスに複数の段落がなくてはなりません。上述の例では、文章の各行を独立の段落に設定してあります。

内部ボックスを設定しているときに、ボックスの内容を水平方向に揃えるには、 L_YX の段落ダイアログを用います。

このボックスの

文章は広げて

あります。

内部ボックスを設定していないときには、ボックスの内容は、ボックスダイアログの中で水平揃えの設定をすることができます。

このボックスの文章は水平方向に広げてあります。

5.2.3 装飾

ボックス型は、ボックスダイアログ中の装飾ドロップダウン式一覧から指定することができます。以下の型を指定することができます。

縁なし 枠を描きません。

簡素な長方形の枠 これはボックスの周りに長方形の枠を描きます。枠線の太さは`\fboxrule`の値になります。

長方形の箱型

楕円形の枠（細線） これはボックスの周りに楕円形の枠を描きます。枠線の太さは0.4ptです。

楕円形の箱型
(細線)

楕円形の枠（太線） これはボックスの周りに楕円形の枠を描きます。枠線の太さは0.8ptです。

楕円形の箱型
(太線)

影付き これはボックスの周りに影付きの長方形の枠を描きます。枠線の太さは`\fboxrule`の値になり、影の幅は4ptになります。

影付きの箱型

影の背景 これは背景色が赤のボックスを描きます。色付きボックスとは違って¹、これは列幅全体を使用し、ボックスは独立した段落に設定されます。

影付き背景の箱型

二重線の長方形の枠 これはボックスの周りに二重線の長方形の枠を描きます。内側の枠線の太さは`0.75\fboxrule`であり、外側の枠線の太さは`1.5\fboxrule`です。枠線間の距離は`1.5\fboxrule + 0.5pt`です。

二重の長方形の
箱型

L_AT_EX のボックスラベルには、使用中の枠の型が表示されます。すべての型を使用可能にするには、L_AT_EX パッケージ `fancybox` が導入済みでなくてはなりません。

¹第5.7節を参照のこと。

装飾として簡素な長方形の箱型を使用し、内部ボックスをなしにした場合には、ボックス内で改行をすることができます。その際、他の枠付きボックスと異なり、枠は列幅全体に広げられて独立した段落に置かれ、このボックス型では`\fboxrule`と`\fboxsep`は無効となります。枠線の太さは`\FrameRule`の値になります。

改行可能な箱型

5.3 ボックスの詳細な設定

寸法`\fboxrule`の既定値は 0.4pt です。これは、 \TeX コードで以下のコマンドを使うことによって、たとえば 2pt に変更することができます。

```
\setlength{\fboxrule}{2pt}
```

`\fboxrule = 2 pt` の長
方形の箱型

枠とボックスの内容との空白は、すべての枠様式について 3pt が既定値になっています。これは、長さ`\fboxsep`を別の値に設定することによって変更することができます。たとえば

```
\setlength{\fboxsep}{10pt}
```

というコマンドは、この値を 10pt にし、以下のボックスのようになります。

`\fboxsep = 10 pt` の
長方形の箱型

楕円形ボックスの丸い角の直径は、コマンド`\cornersize`で設定することができます。

```
\cornersize*{1cm}
```

というコマンドは、直径を 1cm に設定します。

```
\cornersize{数値}
```

というコマンドは、直径を数値 \times (ボックスの幅と高さの小さい方)に設定します。既定値は`\cornersize{0.5}`です。

`\cornersize = 1.5 cm`
の楕円形の箱型

影の大きさは、長さ`\shadowsize`を変更することで調整できます。以下のボックスでは

```
\setlength{\shadowsize}{2pt}
```

というコマンドを使って、 2pt に設定しています。

`\shadowsize = 2pt`
 の影付きボックス

寸法`\FrameRule`の既定値は0.4ptです。ボックスの中身と枠の間の空白の既定値は9ptですが、`\FrameSep`の値を変えることで変更することができます。たとえば、下記のボックスの枠の外見は、

```
\setlength{\FrameRule}{5pt}
\setlength{\FrameSep}{0.5cm}
```

という \TeX コードコマンドで設定してあります。

これは改頁可能ボックスの文章です。

影付き背景ボックスでは、ボックスの内容と枠の余白の既定値は3ptになっていますが、`\fboxsep`の値を変更することで変えられます。既定の背景色である赤色は、`\definecolor{shadebox}`コマンドで局所的に変更することもできますし、ツール▷設定▷色▷影付きボックスメニューからグローバルに変更することもできます。`\definecolor`コマンドの用法は第2.11節²で説明されています。

たとえば、下記の影付き背景ボックスの外見は

```
\setlength{\fboxsep}{0.5cm}
\definecolor{shadecolor}{cmyk}{0.5,0,1,0.5}
```

という \TeX コードコマンドを使って設定してあります。

これは、濃緑色の背景の影付き背景ボックスに入れた黄色の文章です。

変更した長さや幅は、変更を設定したコマンド以後のすべてのボックスに有効となります。

5.4 ミニページ

ミニページは、 \LaTeX によってページの中のページとして取り扱われるので、たとえばそれ自身で脚注を持つことができます。

ミニページは、複数の言語を含む文書を執筆する際に便利です。

下掲は、二つの例示用ミニページを横に並べたものです。これらの幅は45列幅%に設定され、挿入▷整形▷水平方向の空白メニューで挿入した水平フィルで隔てられています。

² \LaTeX の`\definecolor`は、 \LaTeX パッケージの`color`をプリアンブルに読み込む必要があります。第5.7節をご参照ください。

Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text. Dies ist ein deutscher Text.^a Dies ist ein deutscher Text. Dies ist ein deutscher Text.

^aDies ist eine deutsche Fußnote.

This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text. This is an English Text.^a This is an English Text.

^aThis is an English footnote.

ミニページのもう一つの応用例としては、表内での脚注があります。L^AT_EX の制約のせいで、表中の脚注は、同一ページの底部には表示されません。しかし、脚注を含む表をミニページ内に置けば、脚注は底部に表示され、ラテン文字で付番されます。脚注番号は、ミニページ毎に 1 にリセットされますが、ミニページ外ではリセットされません。

この表の脚注は表示されません。

1	2	3 ³	4
い	ろ	は	に
ほ	へ	と	ち

1	2	3 ^a	4
い	ろ	は	に
ほ	へ	と	ち

^aこれは表中の脚注です。

ミニページ内では、文書全体の段落設定は無視されます。つまり、たとえば文書設定でスキップを設定していたとしても、ミニページ内の段落間には余白が置かれません。

ミニページは、文章部に背景色を与えるのに使用することもできます。第 5.7.2 節をご参照ください。

【註】ミニページ内にフロートや傍注を使用することはできませんが、ミニページを表・フロート・他のボックス内で使用することはできます。

5.5 Parbox

Parbox はミニページに非常によく似ていますが、脚注を付けることができない点が違います。

これは parbox 内の文章です。
 これは parbox 内の文章です。
 右の脚注は表示されません。⁴

5.6 単語や文字のボックス

5.6.1 ハイフネーションの回避

単語や文章がハイフネートされるのを防ぐには、makebox 型ボックスを使うことができます。

以下が例示用の文章です。

This line is an example to show how you can prevent the hyphenation of “veryverylongword”.

単語「veryverylongword」がハイフネートされるのを防ぐには、単語をボックスの中に入れて、内部ボックスを makebox コマンドに設定し、幅を 1 幅にします。

以下がその結果です。

This line is an example to show how you can prevent the hyphenation of “veryverylongword”.

別の方法としては、以下のように、単語の前に直接 T_EX コードで「_-」コマンドを設定する方法もあります。

This line is an example to show how you can prevent the hyphenation of “veryverylongword”.

もちろん、こうすることでこの単語は右余白に突き出てしまいます。これを回避するには、以下のように挿入▷整形▷整形なし改行メニュー（短絡キー Ctrl+Return）から改行を単語の前に入れてください。

This line is an example to show how you can prevent the hyphenation of “veryverylongword”.

5.6.2 垂直揃え

\raisebox コマンドを使うと、単語や文字や他のボックスを、周囲の文章に対して垂直方向に整列させることができます。 \raisebox は以下の書式で使います。

\raisebox{持ち上げ幅}[高さ][深さ]{ボックスの内容}

ここで持ち上げ幅は、正の値ならばボックスを持ち上げ、負の値ならばボックスを引き下げます。

たとえば、「preventing」という単語を、そのもっとも「深い」文字「p」の底部がベースラインに来るように揃えるには、単語の前に T_EX コードで

\raisebox{\depth}{

というコマンドを挿入してください。単語の後には、閉じ括弧「}」を \TeX コードで挿入してください。

以下がその出力です。

This is a text line with the word “preventing” as raised word.

行中の文字を上げ下げすると、以下のように行間が広がります。

This is a text line with the word “preventing” as lowered word.
“testing”

This is a text line with the word as raised word.

何らかの理由でこれを抑制したい場合には、ボックスの高さを零に設定してください。たとえば、

```
\raisebox{-\depth}[0pt]{
```

とすると、

This is a text line with the word “preventing” as lowered word.

This is a text line with the word as raised word.

5.7 色付きボックス

5.7.1 文字の着色

文章の背景を着色するには、文章をいわゆる「色付きボックス」に入れなくてはなりません。このためには、 \LaTeX パッケージ `color` が、以下のようなコマンドで文書プリアンブルに読み込まれていなくてはなりません。

```
\@ifundefined{textcolor}
{\usepackage{color}}{}
```

`color` パッケージは、文章を着色すると \LaTeX が自動的に読み込みます⁵。

色付きボックスは、`\colorbox` コマンドを使って作ります。これは、以下のような書式で使われます。

```
\colorbox{color}{ボックスの内容}
```

ボックスの内容には、ボックスが入ることもできますし、色付きボックスを他のボックスに入れることもできます。

事前定義されている色には、

黒・青・シアン・緑・マゼンタ・赤・白・黄

があります。第 2.11 節に述べられているようにすれば、自前の色を定義することもできます。

⁵これが二度読み込まれることを回避するには、`\@ifundefined` コマンドを使用します。

たとえば、単語の背景を赤にするには、単語の前に $\text{T}_\text{E}\text{X}$ コードで

```
\colorbox{red}{
```

というコマンドを入れてください。単語の後には、 $\text{T}_\text{E}\text{X}$ コードで閉じ括弧「}」を入れてください。

以下がその結果です。

この行では、**「注意！」** という単語が赤い背景になっています。

ボックスの枠を別の色にしたい場合には、以下の書式で `\fcolorbox` コマンドを使います。

```
\fcolorbox{枠の色}{ボックスの色}{ボックスの内容}
```

`\fcolorbox` は `\colorbox` の拡張です。枠の太さと枠とボックスの中身との余白は、第 5.2.3 節で述べられているように、それぞれ `\fboxrule` と `\fboxsep` という長さで調節できます。

以下の例では、

```
\fcolorbox{cyan}{magenta}{
```

というコマンドを使用しています。

これは、枠の太さを 1 mm に設定した例です。

これは着色した色付きボックス中の文章です。

もちろん、色付きボックスの中に色付き文字を入れることもできます

これは着色した色付きボックス中の色付き文章です。

【註】色付きボックス中の文章は改行できません。複数行あるいは段落に色を付けるには、以下に述べるように、色付きボックス中でボックスを使用してください。

5.7.2 段落の着色

二行以上の背景色を設定するには、文章をミニページに入れてください。ミニページの前に、 $\text{T}_\text{E}\text{X}$ コードで `\colorbox` コマンドを

```
\colorbox{色名}{
```

のように挿入します。ミニページの後には、 $\text{T}_\text{E}\text{X}$ コードで閉じ括弧「}」を挿入してください。

これは背景色を付けた文章です。これは背景色を付けた文章です。

ここの文章には脚注^aを付けることもできますし、図表を挿入することもできます。

a	!	3
<	b2''	>
1	§	c

^a脚注の例

5.8 ボックスの伸縮回転

本節で述べられているコマンドを使用するには、文書プリアンブルで

```
\@ifundefined{rotatebox}
{\usepackage{graphicx}}{}
```

というコマンドを使用して、graphicx_{LaTeX} パッケージを読み込む必要があります。

【註】DVI ビューアによっては、回転伸縮したものの表示ができないものがあります。

【註】フロートは回転伸縮したボックスの中に入れることはできません。

5.8.1 ボックスの回転

何かを回転させたい場合には、回転させたボックスに入れましょう。そのようなボックスは、_{TeX} コードで以下のような書式で`\rotatebox` コマンドを使用すれば作成することができます。


```
\rotatebox[回転の原点]{回転角}{ボックスの内容}
```

回転の原点は、`origin=位置`の形で指定します。位置として指定可能なのは、*c* (中央揃え)・*l* (左揃え)・*r* (右揃え)・*b* (下揃え)・*t* (上揃え) と、これら五つの適切な組み合わせです。たとえば、*lt* と指定した場合には、回転の原点をボックスの左上隅にすることです。回転原点が指定されなかった場合には、位置として *l* が使用されます。回転角は、角度を指定する数字で負の値もとることができます。回転の方向は、反時計回りです。

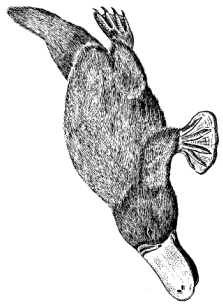
以下の例では、「回転した」文章の前に _{TeX} コードで`\rotatebox[origin=c]{60}` というコマンドを挿入し、文章の後に _{TeX} コードで閉じ括弧`}`でボックスを閉じています。

これは、回転した文章を含む行です。

ボックスの中身には、以下のように、もう一つボックスを入れたり行内数式を入れたりすることもできます。

これは、 枠付き文章と $\int A dx = B$ 数式を含む行です。

あるいは、以下のように図や表を入れることもできます。

これは、 回転した画像と

e	b
r	w

 表を含む行です。

5.8.2 ボックスの伸縮

素材を伸縮させるには、 $\text{T}_\text{E}\text{X}$ コードで `\scalebox` コマンドと `\resizebox` コマンドが使えます。

`\scalebox` は以下の書式で使います。

`\scalebox{水平方向}[垂直方向]{ボックスの内容}`

水平方向と垂直方向は、それぞれ対応する伸縮因子です。垂直方向の伸縮因子を省略した場合には、水平方向の伸縮因子が垂直方向にも適用されます。

たとえば、`\scalebox{2}{ここにちは}` というコマンドは、文書の本文の大きさに比して2倍に引き延ばされた **ここにちは** を生成します。

これに対し、`\scalebox{2}[1]{ここにちは}` は、**ここにちは** を変形させます。

伸縮因子に負の値が与えられると、ボックスの内容は鏡像になります。したがって、`\scalebox{-1}[1]{ここにちは}` というコマンドは、右のように反転文字を生成するのに使用できます：
おさこひこ

`\scalebox{1}[-1]{ここにちは}` とすると $\overline{\text{ここにちは}}$ のようにベースラインで鏡像にします。

`\scalebox{-1}[1]{ボックスの内容}` と同じ働きをするものとして



`\reflectbox{ボックスの内容}`

というコマンドがあります。

指定した幅や丈にボックスを伸縮させるには、`\resizebox` を使います。このコマンドの書式は

`\resizebox{幅}{丈}{ボックスの内容}`

です。上記二つのコマンド引数のうち一つが感嘆符「!」であるならば、ボックスの内容の縦横比が一定に保たれるように大きさが調節されます。

`\resizebox{2cm}{1cm}{こんにちは}`というコマンドの出力は、となり、
`\resizebox{2cm}{!}{こんにちは}`というコマンドの出力は、となります。

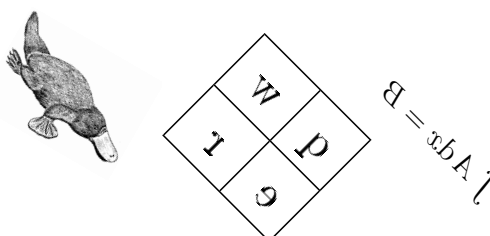
【註】`\scalebox` や `\resizebox` の引数を零に設定した場合には、文書を書き出す際に \LaTeX エラーは発生しませんが、書き出したファイルは表示できなかったり、一部しか表示されなかったりします。

ボックスは好きな順序で組み合わせることができます。たとえば、

`\rotatebox[origin=c]{-45}{\resizebox{2cm}{!}{\reflectbox{こんにちは}}}`
 というコマンドは、



という出力になります。ボックスの内容には、以下のように、画像や表や行内数式を入れてもかまいません。



文書で大域的に数式様式 `fleqn` を使用している場合⁶には、別行建て数式も伸縮させることができます。

⁶つまり、文書クラスオプションに「`fleqn`」を書き加えた場合。

第6章 外部文書構成要素

挿入▷ファイルメニューを使うと、使用中の文書に外部素材を挿入することができます。挿入できるのは、

L_YX 文書 別の L_YX 文書。その内容は直接使用中の文書に挿入されます。

平文 テキスト文書。そのテキスト行各々が、使用中の文書に独立した段落として挿入されます。

平文（行を連結して） テキスト文書。各テキスト行がその形のまま挿入されます。空の行があると、使用中の文書に新規段落を生成します。

外部素材 様々な書式のファイル。

子文書 L_YX あるいは L^AT_EX 文書。

6.1 外部素材

外部素材機能は、文書の出力書式が読むことのできる形式に前もって変換しなくても、L_YX が必要な変換を行うことで、使用中の文書にファイルを挿入することができるようにするものです。これは、画像が様々な画像形式のままで L_YX 文書に挿入することができるのと同様の働きをします。L_YX 設定の操作性▷表示で自動プレビューを有効にすると、外部素材の Dia 形式と Xfig 形式は、直接 L_YX 中に表示されるようになります。

外部素材は、挿入▷ファイル▷外部素材メニューで開くことのできる外部素材ダイアログから挿入することができます。目下、以下のファイル形式（ひな型）が使用できます。

チェス棋譜 このひな型は、[XBoard](#) プログラムで作成されたチェス棋譜をサポートします。

日付 これは、日-月-年の書式で日付を挿入します。右は、外部素材として挿入された日付です：15-05-2011

日付は L_YX 内では表示されず、出力のみで表示されます。日付を挿入するには、ほかに二つの方法があります。一つは、挿入▷日付メニューを使用する方法で、他は L^AT_EX コマンドの `\today` を T_EX コードで使用方法です。これらの各方法は、表 6.1 に比較を掲載しておきます。

Dia このひな型は、**Dia**プログラムで作成したダイアグラムをサポートします。

LilyPond このひな型は、**LilyPond**プログラムで作成した楽譜の組版に使用されます。

PDF ページ このひな型を使えば、PDF 文書を使用中の文書に挿入することができます。

PDF の一部頁ないし全頁を挿入するには、ダイアログ中のひな型の解説に沿って、 \LaTeX と \LyX オプションタブ中のオプションフィールドに *pages* オプションを指定してください。 *pages* オプションを指定しないと、PDF の最初のページのみが挿入されます。

ラスターイメージ これはビットマップ画像に使用します。よく使われる画像形式のほとんどすべてがサポートされています。外部素材ダイアログを使って入れる画像は、第 1.1 節にあるように画像ダイアログを使って挿入される画像と同じように取り扱うことができます。異なる点といえば、ラスターイメージのみを取り扱うことができる点、すなわち PDF や EPS 画像はサポートしていないという点のみです。

XFig このひな型は、**Xfig**プログラムで作成した画像をサポートします。

表 6.1: 日付の挿入方法の比較

文書書式	外部素材▷日付	挿入▷日付	<code>\today</code> コマンド
\LyX	差込枠として	日付として	\TeX コード差込枠として
\LaTeX	日付として	日付として	コマンドとして
DVI, PDF, PS	日付として	日付として	日付として

外部素材ダイアログのファイルタブ中で下書きオプションを指定すると、出力中には挿入したファイルのパスだけが表示されます。

\LyX 中での外部素材は、ダイアログ内の \LaTeX と \LyX のオプションタブにある \LyX 内に表示オプションの指定に従って、**ビットマップ: catalan** のようなボックスとして表示されるか、画像として表示されます。

取扱説明書カスタマイズ篇には、自製のひな型を定義する方法が説明されています。

6.2 子文書

子文書は、大きな部や節を複数含む長大な文書を作成するときに使われます。メンテナンス上の理由から、文書をいくつかのファイルに分割して、個別に修正できるようにした方が便利なのはよくあり、場合によってはそうすることが必要な場合さえあります。こうした個別文書がいわゆる子文書であり、親文書はこれら子文書を連結して、文書全体やその一部を印刷できるようにします。子文書は、 \LaTeX プリアンブルや書誌情報や相互参照ラベルなどの親文書の内容を引き継ぎます。

親文書を開くことなく子文書で作業できるようにするには、子文書の文書▷設定▷文書クラスメニューで親文書を指定してください。すると、子文書を編集する際には、 $\text{L}_\text{Y}\text{X}$ がバックグラウンドでこの親文書を扱うようになります。

取り込んだ文書は、 $\text{L}_\text{Y}\text{X}$ 中で `Include: Customization.lyx` のようなボックスとして表示されます。子文書を親文書に取り込むには、挿入▷ファイル▷子文書メニューを使用します。するとダイアログが現れ、以下のように 4 つの取り込み方法を選択することができます。

Include (インクルード) $\text{L}_\text{Y}\text{X}$ 文書と $\text{L}^\text{A}\text{T}_\text{E}\text{X}$ 文書は、インクルードすることができます。子文書ダイアログの編集ボタンを押すと、インクルードされた文書が、 $\text{L}_\text{Y}\text{X}$ 中に新規ファイルタブとして開き、修正ができるようになります。

以下は、Include を使用して挿入した子文書です。

6.2.1 外部文書の第 1 小節

これは、ファイルが別の文書にどのように挿入されるかを示すための、短いダミー子文書です。

取り込んだファイルの節番は、各節を親文書に挿入した順序にしたがって取り込まれます。たとえば、上で取り込んだ例示文書には、この節の小節として番号を振られた小節があります。取り込んだ文書のラベルは、第 6.2.1 小節、のように参照することができます。

子文書のプリアンブルは無視され、親文書のプリアンブルだけが使用されます。親文書と同じ名称の派生枝がない限り、親文書は子文書の派生枝を無視します。取り込まれた文書は、新規頁から始まるように挿入され、改頁して終わります。

大きな文書のうち、特定の章だけの作業を行っている際に、コンパイル時間を短縮したい場合、取り込んだファイルの各々を派生枝の中に入れる方法があります¹。派生枝を有効化したり無効化したりすることによって、取り込んだファイルのどれを出力に表示させるかを決めることができます。

【註】 $\text{L}_\text{Y}\text{X}$ のバグにより、どのファイルを取り込むかを定める他の方法として、現在のところ、 $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ コマンドの `\includeonly` を使用することはできません。

【註】 $\text{L}_\text{Y}\text{X}$ または $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ ファイルを取り込んだ際、子文書が親文書と異なる文書クラスを使用している場合には、予期せぬ出力を避けるために、文書を書き出したり閲覧する際に警告が発せられます。

Input (インプット) この方法は、Include による方法に非常によく似ています。異なっている点は、

- Input ファイルは、新規頁として始まることはなく、終わりに改頁されることもありません。
- Input ファイルは、 $\text{L}_\text{Y}\text{X}$ 設定の操作性▷表示の下にある画像を表示を有効にすると、 $\text{L}_\text{Y}\text{X}$ 中でプレビューすることができます。
- $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ コマンドの `\includeonly` は使用することができません。

右は、Input を使って挿入した子文書です。

6.2.2 外部文書の第 2 小節

これは、ファイルが別の文書にどのように挿入されるかを示すための、もう一つの短いダミー子文書です。

Verbatim (文字通り) この方法では、テキストファイルなら何でも取り込むことができます。出力中では、ファイルはそのソースコードの通りに表示され、テキスト中に用いられているコマンドは実行されません。出力の空白に印付けをするオプションを有効にすると、ソースコード中のすべての空白文字に「`\`」文字を表示させることができます。挿入▷ファイル▷平文メニューを使う方法との違いは、こちらは文書の内容が $\text{L}_\text{Y}\text{X}$ 中に表示されないことです。

¹派生枝は、 $\text{L}_\text{Y}\text{X}$ ユーザーの手引きの派生枝の節に説明されています。

以下は、`verbatim` として挿入された子文書です。

```
This is a small dummy child text document to show how files can be inserted into a do
```

以下は、出力の空白に印付けをするオプションを指定して、`verbatim` として挿入された子文書です。

```
This is a small dummy child text document to show how files can be inserted into a do
```

【註】上記の例でわかるように、`verbatim` として取り込んだ文書の文章は、文書行の終わりで改行されません。

【訳註】`verbatim` で日本語のファイルを読み込むときは、子文書のエンコーディングを文書エンコーディングに合わせてください。さもないと出力が文字化けします。

プログラムリスト この型については、第7章に説明されています。

【註】同一文書中に、同じ文書を異なる方法で二度取り込むと、 \LaTeX で問題が発生する可能性があります。

第7章 プログラムコードリスト

プログラムコードを取り込んで組版するには、挿入▷プログラムリストメニューを使用して挿入されるプログラムリスト差込枠を使用することができます。ここで使用される L^AT_EX パッケージ listings は、使用中の文書にプログラムソースコードを挿入する強力な柔軟な方法を提供します。

プログラムリスト差込枠を右クリックすると、コンテキストメニューが開き、リスト書式を設定することができます。

既定では、出力におけるプログラムリストは、新しい段落として始まります。配置オプションで行内リストを指定すると、リストは `int a=5;` のように行内に印字されます。フロートオプションは、プログラムリストフロートを生成するものですが、ここで、第 3.5 節に説明されているフロート配置オプションに従って、「h」「t」「b」「p」の配置オプションを指定することができます。配置オプションは組み合わせて使用することができます。「htbp」のように区切りを入れることなく入力することができます。「h」オプションは効かない場合がありますが、フロートでないプログラムリストにもキャプションを入れることができ、また参照もできるので、この場合にはフロートオプションを使用しなければ大丈夫です。

挿入▷キャプションメニューを使えば、プログラムリストにキャプションを付けることができます。プログラムリストは、フロート同様、リスト 7.1 のように参照することができます。

Listing 7.1: Example Listing Float

```
# Example listing float
def func(param):
    'this is a python function'
    pass
```

プログラムリストダイアログでプログラミング言語を指定すると、その言語のキーワードが識別されて、出力中で特別に組版されます。上の例示プログラムリストでは、Python のキーワード「def」が識別され、ボールド体で出力されます。

【註】タイプライタフォントを使用している際にキーワードがボールド体にならない場合には、おそらくお使いの typewriter フォントがボールド体を提供していません。この場合には、文書▷設定▷フォントメニューで別のフォントを選択してください (LuxiMono・BeraMono・Courier の各フォントはボールド体を提供します)。

【訳註】プログラムリスト差込枠中では、日本語のエンコーディングを表示できないため、日本語は使用できません。プログラムリスト差込枠中のキャプションでも同様です。

プログラムリストダイアログの行番号のセクションでは、行番号の様式を指定することができます。行間フィールドでは、どの行に番号を振るかを指定する数字を入れることができます。たとえば、「3」と入れると、3行ごとに数字が振られます。

さらに行の範囲を指定することもでき、この場合にはこれらの行のみが出力に表示されます。プログラムリスト中でドイツ語のウムラウトのような国字を使用する場合には、拡張文字コード表を使うオプションを使用しなくてはなりません。

以下は、左側の行番号・行間「3」・言語「Python」・「拡張文字コード表を使う」オプション・「空白を記号で表示」オプション・行範囲「3-8」で設定したプログラムリストの例です。

```

1 pass
2 def_func (param):
    'This is a German word: _ Tsch'
    pass
5 def_func (param):
    'this is a python function'

```

プログラムリスト中でタブュレータを使う場合は、タブュレータが確保する文字の数をタブュレータの大きさフィールドで指定することができます。

【註】listings パッケージにはバグがあり、前にプログラムリストがある毎に、行番号が一行ずつずれます。上記プログラムリストで3行目と6行目に番号が振られる代わりに、2行目と5行目に振られているのはこのためです。

また、ファイルの各行をプログラムリストとして印字することも可能です。そのためには、挿入▷ファイル▷子文書メニューを使用してプログラムリスト型を選択してください¹。子文書ダイアログでは、テキストボックスにプログラムリストのパラメータを指定することができます。使用できる全パラメータの一覧を見たい場合は、テキストボックス中で疑問符「?」を打鍵してください。

子文書プログラムリストを参照したい場合には、子文書ダイアログの対応するフィールドにラベル文を書き込んでください。すると、このラベルは挿入▷相互参照メニューを使って参照できるようになります。

プログラムリスト 7.2は、ファイルを使ったプログラムリストの一例です。ここでは、このL_YX ファイルの10行目から15行目が表示されています。

Listing 7.2: Lines 10 - 15 of this LyX file

```

\ifnum 42146=\euc"A4A2 \AtBeginDvi{\special{pdf:tounicode EUC-
    UCS2}}\else
\AtBeginDvi{\special{pdf:tounicode 90ms-RKSJ-UCS2}}\fi

% if pdflatex is used

```

¹他の子文書型については、第 section 6.2節に説明があります。

```
\usepackage{ifpdf}
\ifpdf
```

リストに数式を入れるには、以下のようにします。

1. リストダイアログを開き、詳細タブのフィールド中に
mathescape=true
と書き加えます。
2. リスト外で数式を作成します。
3. 数式の内容（数式差込枠ではなく）をクリップボードにコピーします（短縮キー
Ctrl+C）。
4. リストに「\$\$」と書き、クリップボードの内容をドルマークの間に貼り付けます（短縮キー
Ctrl+V）。

Listing 7.3: Example Listing float containing a formula

```
# Example listing float
def func(param):
    #  $\prod_{n=1}^{\infty} \frac{1}{x^2}$ 
    pass
```

大域的なプログラムリストの設定は、文書▷設定▷リストダイアログで設定することができます。ここで利用できるオプションの一覧を得るには、プログラムリスト設定フィールドで疑問符「?」を打鍵してください。

listings パッケージについての詳しい情報については、取扱説明書 [12] をご参照ください。.

付録 A $\text{L}_\text{Y}\text{X}$ で使用できる単位

この文書で用いられる単位の理解のために、表 A.1 に、 $\text{L}_\text{Y}\text{X}$ で使うことのできる単位をまとめおきます。

表 A.1: 単位


単位	名称 / 摘要
mm	ミリメートル
cm	センチメートル
in	インチ
pt	ポイント (72.27 pt = 1 in)
pc	パイカ (1 pc = 12 pt)
sp	スケールポイント (65536 sp = 1 pt)
bp	ビッグポイント (72 bp = 1 in)
dd	ディドー (72 dd \approx 37.6 mm)
cc	シセロ (1 cc = 12 dd)
Scale%	元の画像幅の%
text%	テキスト幅の%
col%	組段幅の%
page%	用紙幅の%
line%	行幅の%
theight%	テキスト高さの%
pheight%	用紙高さの%
ex	現在のフォントの文字「x」の高さ
em	現在のフォントの文字「M」の幅
mu	数式単位 (1 mu = 1/18 em)

付録 B 画像を含む場合のファイル出力形式

B.1 DVI

このファイル型の拡張子は「.dvi」です。このファイル型は完全に可搬性があるので「device-independent (DVI: デバイス独立型)」の名称がついています。これはあるコンピュータから別のコンピュータへ、何らの変換も行うことなく移動することができます。このファイル書式が開発された当時には、これは当然のことでは全くありませんでした。DVI は手っ取り早くプレビューしたり、PostScript などの他の出力形式への中間段階として使用されます。

【注】DVI ファイルには画像は含まれていません。画像はリンクされているだけです。ですから、お使いの .dvi ファイルを別のコンピュータに移すときには、このことを忘れてはいけません。この性質は、DVI を閲覧する際にお使いのコンピュータを遅くしてしまうことがあります。DVI ビューアは、DVI をスクロールする際、画像を表示できるようにバックグラウンドで変換しなくてはならないためです。ですから、画像のたくさんあるファイルの場合は、PDF を使用することをお勧めします。

使用中の文書は、ファイル▷書き出し▷DVI メニューを使用して、DVI に書き出すことができます。また、表示メニューかツールバーボタン  を使用すれば、使用中の文書を DVI で表示させることができます。


B.2 PostScript

このファイル型の拡張子は「.ps」です。PostScript は、Adobe 社によってプリンタ言語として開発されました。したがって、このファイルにはプリンタがファイルを印字するコマンドが書かれています。PostScript は、計算を行ったり図表や画像を描いたりすることもできるので、「プログラミング言語」と見なすことができます¹。この機能のために PostScript は PDF よりも大きくなりがちです。

PostScript は、「Encapsulated PostScript (EPS: ファイル拡張子「.eps」)」形式の画像のみを含めることができます。L^AT_EX では、既知の画像形式であればどれも文書中で使用することができるようになっているため、L^AT_EX はバックグラウンドで画像を EPS に変換しなくてはなりません。文書中に、仮に 50 の画像があれば、文書を表示したり書き出し

¹これについてももう少し学ぶ気があれば、L^AT_EX パッケージ PSTricks[14] を見てみてください。

したりするたびに、 $\text{L}_\text{Y}\text{X}$ は 50 もの変換を行わなくてはなりません。すると、 $\text{L}_\text{Y}\text{X}$ で行う作業手順が、著しく遅れてしまいます。したがって、もし PostScript を使用することを考えているならば、この問題を回避するために、画像を直接 EPS として入れておく方がよいでしょう。

文書は、ファイル▷書き出し▷Postscript メニューを使用して、PostScript に書き出すことができます。使用中の文書は、表示メニューかツールバーボタンを使用することで PostScript として表示させることができます。

B.3 PDF

このファイル型の拡張子は「.pdf」です。この「Portable Document Format (PDF)」は、Adobe 社によって PostScript の派生として開発されました。PDF は、PostScript よりも圧縮されており、かなり少ないコマンドしか使いません。「portable (可搬的)」の名称が示すとおり、この形式は、どのコンピュータシステムでも処理することができ、印字出力は全く同じように出すことができます。

PDF は画像を、同じ PDF 形式か、「Joint Photographic Experts Group (JPG: ファイル拡張子「.jpg」または「.jpeg」)」形式か、「Portable Network Graphics (PNG: ファイル拡張子「.png」)」として含めることができます。それにもかかわらず、 $\text{L}_\text{Y}\text{X}$ は画像をこれらの形式のいずれかにバックグラウンドで変換を行うので、他の画像形式も自由に使用することができます。しかし、PostScript の節で説明したように、この画像変換は作業手順を遅らせます。したがって、上記で述べた 3 つの画像形式のうちいずれかを使用することをお勧めします。


使用中の文書は、ファイル▷書き出しメニューを使用して、以下の 3 つの方法のうちいずれかで PDF に書き出すことができます。

PDF (ps2pdf) これは ps2pdf プログラムを使って、現在のファイルの PostScript 版から PDF を生成します。この PostScript 版は、中間段階に DVI 版を使用する dvips プログラムによって生成されます。つまり、この書き出し方法は、3 段階の変換を伴います。

PDF (dvi2pdf) これは dvi2pdf プログラムを使って、現在のファイルをバックグラウンドで DVI に変換し、2 段階目で PDF に変換します。

PDF (pdflatex) これは pdftex プログラムを使って、現在のファイルを直接 PDF に変換します。

pdftex は現在の PDF 版のすべての機能をサポートし、高速で問題なく安定して動作するので、PDF (pdflatex) を使用することが推奨されます。dvipdfm プログラムは開発が停止しており、若干時代遅れになっています。

使用中の文書を PDF として表示するには、表示メニューかツールバーボタン (これは PDF (pdflatex) を使用します) を用います。

付録 C 第 (2.1) 式の説明

n 個の表セルの全幅 $W_{\text{tot } n}$ は

$$W_{\text{tot } n} = n \cdot (W_{gn} + 2 \cdot \backslash\text{tabcolsep}) + (n + 1) \cdot \backslash\text{arrayrulewidth} \quad (\text{C.1})$$

で計算できます。ここで W_{gn} は、すべてのセルが持つ固定幅です。 $\backslash\text{tabcolsep}$ は、セルテキストとセル罫線のあいだを指す \LaTeX 長であり、既定値は 6 pt です。 $\backslash\text{arrayrulewidth}$ は、セル罫線幅であり、既定値は 0.4 pt です。

第 (C.1) 式にしたがえば、連結列の全幅 $W_{\text{tot mult}}$ は

$$W_{\text{tot mult}} = W_{g\text{mult}} + 2 \cdot \backslash\text{tabcolsep} + 2 \cdot \backslash\text{arrayrulewidth} \quad (\text{C.2})$$

となります。 n 列を連結する時には各列の全幅は $W_{\text{tot mult}}/n$ となるので、第 (C.1) 式と第 (C.2) 式が等しいものと置けば、必要となる固定幅 W_{gn} を計算することができて、

$$W_{gn} = (W_{g\text{mult}} + (1 - n) \cdot (2 \cdot \backslash\text{tabcolsep} + \backslash\text{arrayrulewidth}))/n \quad (\text{C.3})$$

となります。

参考文献

- [1] Frank Mittelbach and Michel Goossens: *The L^AT_EX Companion Second Edition*. Addison-Wesley, 2004
- [2] Helmut Kopka and Patrick W. Daly: *A Guide to L^AT_EX Fourth Edition*. Addison-Wesley, 2003
- [3] Leslie Lamport: *L^AT_EX: A Document Preparation System*. Addison-Wesley, second edition, 1994
- [4] L^AT_EX パッケージ [booktabs](#)の取扱説明書
- [5] L^AT_EX パッケージ [caption](#)の取扱説明書
- [6] L^AT_EX パッケージ [endfloat](#)の取扱説明書
- [7] L^AT_EX パッケージ [wrapfig](#)の取扱説明書
- [8] L^AT_EX パッケージ [footmisc](#)の取扱説明書
- [9] L^AT_EX パッケージ [hypcap](#)の取扱説明書
- [10] L^AT_EX パッケージ [hyperref](#)の取扱説明書
- [11] L^AT_EX パッケージ [KOMA-script](#)の取扱説明書
- [12] L^AT_EX パッケージ [listings](#)の取扱説明書
- [13] L^AT_EX パッケージ [marginnote](#)の取扱説明書
- [14] L^AT_EX パッケージ [PSTricks](#)のウェブページ
- [15] L^AT_EX パッケージ [sidecap](#)の取扱説明書
- [16] L^AT_EX パッケージ [prettyref](#)
- [17] L^AT_EX パッケージ [refstyle](#)
- [18] L^yX 2.0 の新機能に関する [Wiki ページ](#)

索引

L^AT_EX パッケージ

booktabs, 27, 93
 calc, 24
 caption, 10, 15, 16, 50, 93
 color, 71
 colortbl, 29
 endfloat, 46, 93
 fancybox, 66
 footmisc, 60, 93
 ftnright, 58
 graphicx, 73
 hypcap, 52
 hyperref, 14, 44, 93
 KOMA-script, 50, 51, 93
 listings, 83, 93
 marginnote, 93
 prettyref, 43, 93
 PSTricks, 93
 refstyle, 43, 93
 sidecap, 51, 93
 wrapfig, 40, 93

DVI, ファイル書式

EPS, 画像書式

GIF, 画像書式

JPG, 画像書式

PDF, 5, 90

PNG, 画像書式

PostScript, ファイル書式

SVG, 画像書式

色

段落—, 72

文字の—, 71

色付き

表罫線部, 30

表セル部, 28

回転

素材の—, 73

外部文書構成要素, 77

外部素材, 77

子文書, 78

画像, 図

画像書式, 5

脚注, 56

配置, 58

付番, 57

キャプション

書式設定, 47

配置, 51

後注, 58

参照

参照位置, 44

自動命名, 44

書式, 43

図への —, 3

表への, 10

フロートに対する, 42

伸縮

素材の—, 74

図, 1

画像書式, 5

画像ダイアログ, 1

設定グループ, 2

—の回転, 1, 73

—の伸縮, 1, 74

- フロート, 3, 43
- 単位, 87
- 注釈
 - LyX ノート, 55
 - 脚注, 56
 - 傍注, 60
- 長尺表, 10
 - 脚注, 13
 - キャプション, 14
 - キャプションの幅, 15
 - 強制改頁, 19
 - 計算, 18
 - 参照, 14
 - 縦横揃え, 13
 - フロート, 18
 - ページごとに異なるキャプション, 16
- 表, 7
 - 色付き, 28
 - 改行, 22
 - 揃え, 28
 - ダイアログ, 7
 - 長尺表, 10
 - 調整, 31
 - ツールバー, 8
 - はじめに, 7
 - フォーマル形式, 25
 - フロート, 10
 - 編集メニュー, 9
 - 連結行, 25
 - 連結列, 23
- 表セル内の複数行, 22
- 表の色
 - 罫線部, 30
 - セル部, 28
- 表の調整, 31
 - 回転, 73
 - 行の余白取り, 31
 - 罫線幅, 35
 - 伸縮, 74
 - セルと列の書式, 34
 - 点罫, 37
 - 特殊なセル揃え, 33
- ファイル
 - インクルード, 78
- ファイル書式
 - DVI, 89
 - PDF, 90
 - PostScript, 89
- フロート, 39
 - アルゴリズム, 40
 - 折返しフロート, 40
 - 回転, 47
 - キャプションの書式設定, 47
 - キャプションの配置, 51
 - 参照, 42
 - 従属フロート, 47
 - 図, 3, 43
 - 配置, 45
 - はじめに, 39
 - 表, 10
 - フロート一覧, 52
 - 横並び, 47
 - リスト, 83
 - 連番, 41
- プログラムコード, 83
- ボックス
 - Parboxes, 69
 - Raiseboxes, 70
 - 色付き, 71
 - 回転した, 73
 - 詳細な設定, 67
 - 伸縮させた—, 74
 - 垂直揃えの仕方, 70
 - 寸法, 63
 - 装飾, 66
 - ハイフネーションの回避, 70
 - はじめに, 63
 - ボックスダイアログ, 63
 - ミニページ, 68
 - 文字揃え, 64
 - 文字用, 70

リスト, 83

連結行, 25

連結列, 23

計算, 23