

*Scientific WorkPlace / Scientific Word*

*version 5.5*

## 徹底活用ガイド

(株) ライトストーン

## はじめに

本ガイドは SWP/SW の機能を最大限活用していただくことを企図して作成された手引書です。SWP/SW を使用して文書作成を行う場合、T<sub>E</sub>X の機能を意識することは余りありません。T<sub>E</sub>X を知らなくても高品位な文書作成が行える点に SWP/SW の特長があることも事実ですが、T<sub>E</sub>X の機能を併用するとより高度の文書編集が可能になります。

本ガイドはそれぞれ 5 ページ前後の章から構成されています。通勤通学の途上、あるいは休憩時間に眺めていただくだけでも 1 つのテーマを読みきることができると思います。なお、基本操作については

- SWP/SW 使用の手引き
- 等で習得が済んでおられることを前提としています。

第 1 版 (2008/3)

© 2008 Lightstone Corporation All rights reserved.

このマニュアルのいかなる部分も、(株)ライトストーンの文書による許可なく、理由の如何によらず、どのような形式であっても複製することを禁じます。

- Scientific WorkPlace, Scientific Word は MacKichan Software 社の登録商標です。
- 本書では®や™マークを付していない箇所があります。

本書は *Scientific WorkPlace* を用いて作成しました。

# 目次

<b>第 1 章</b>	<b>クラスファイルとは</b>	<b>1</b>
1.1	クラスファイルとは . . . . .	1
1.2	クラスファイルの実体 . . . . .	2
1.3	パッケージファイル . . . . .	2
1.4	クラスオプション . . . . .	3
1.5	学会固有クラスファイル . . . . .	5
<b>第 2 章</b>	<b>フロントマターの作成</b>	<b>7</b>
2.1	フロントmatterダイアログ . . . . .	7
2.2	フロントmatter作成例 . . . . .	8
<b>第 3 章</b>	<b>見出しの制御</b>	<b>13</b>
3.1	見出し番号 . . . . .	13
3.2	見出し番号の参照 . . . . .	14
3.3	書籍型文書での留意点 . . . . .	14
3.4	見出し番号の設定 . . . . .	15
3.5	標準見出しの変更 . . . . .	15
3.6	見出しの様式調整 . . . . .	15
<b>第 4 章</b>	<b>スペースの制御</b>	<b>17</b>
4.1	SWP/SW におけるスペースの制御 . . . . .	17
4.2	横スペース . . . . .	18
4.3	縦スペース . . . . .	19
4.4	ブレーク . . . . .	20
4.5	ルール . . . . .	21
<b>第 5 章</b>	<b>箇条書きアイテム</b>	<b>23</b>
5.1	リストアイテム . . . . .	23

5.2	基本操作 . . . . .	23
5.3	リードインオブジェクト . . . . .	25
<b>第 6 章</b>	<b>テキストタグ</b>	<b>27</b>
6.1	テキストタグ . . . . .	27
6.2	タグの重畳 . . . . .	28
6.3	文字サイズの変更 . . . . .	29
6.4	数式文字の調整 . . . . .	30
<b>第 7 章</b>	<b>数式 TeXnique [1]</b>	<b>33</b>
7.1	数式の種別 . . . . .	33
7.2	数式のプロパティ . . . . .	35
7.3	数式番号 . . . . .	36
7.4	位置揃え . . . . .	38
<b>第 8 章</b>	<b>数式 TeXnique [2]</b>	<b>41</b>
8.1	数式用ツールバー . . . . .	41
8.2	数式入力 . . . . .	41
8.3	数式の拡大/縮小 . . . . .	43
8.4	ディスプレイ数式 . . . . .	43
<b>第 9 章</b>	<b>数式 TeXnique [3]</b>	<b>45</b>
9.1	数式番号の様式 . . . . .	45
9.2	数式番号の操作 . . . . .	46
<b>第 10 章</b>	<b>数式 TeXnique [4]</b>	<b>49</b>
10.1	定理型環境の定義 . . . . .	49
10.2	ラベル . . . . .	50
10.3	番号付け . . . . .	50
10.4	定理型環境のカスタマイズ . . . . .	51
<b>第 11 章</b>	<b>表 TeXnique [1]</b>	<b>53</b>
11.1	表の種別 . . . . .	53
11.2	配置の制御 . . . . .	55
11.3	キャプションと表番号 . . . . .	55
11.4	参照用キー . . . . .	57
<b>第 12 章</b>	<b>表 TeXnique [2]</b>	<b>59</b>

---

12.1	表のプロパティ . . . . .	59
12.2	アラインメント . . . . .	60
12.3	罫線 . . . . .	60
12.4	列幅 . . . . .	60
12.5	行や列の追加 . . . . .	62
12.6	セルの結合 . . . . .	63
<b>第 13 章</b>	<b>画像 TeXnique [1]</b>	<b>65</b>
13.1	画像の配置 . . . . .	65
13.2	pLATEX 特記事項 . . . . .	68
<b>第 14 章</b>	<b>画像 TeXnique [2]</b>	<b>69</b>
14.1	<i>wrapfig</i> パッケージ . . . . .	69
14.2	図表番号の調整 . . . . .	71
<b>第 15 章</b>	<b>段組み</b>	<b>73</b>
15.1	クラスオプション . . . . .	73
15.2	<i>multicol</i> パッケージ . . . . .	75
15.3	<code>figure*</code> 環境/ <code>table*</code> 環境 . . . . .	76
<b>第 16 章</b>	<b>カラー</b>	<b>79</b>
16.1	<i>color</i> パッケージ . . . . .	79
16.2	<i>colortbl</i> パッケージ . . . . .	82
<b>第 17 章</b>	<b>文献参照</b>	<b>83</b>
17.1	文献一覧 . . . . .	83
17.2	文献引用 . . . . .	85
<b>第 18 章</b>	<b>用紙のレイアウト</b>	<b>87</b>
18.1	用紙サイズの設定 . . . . .	87
18.2	余白の調整 . . . . .	88
18.3	用紙の向き . . . . .	88
<b>第 19 章</b>	<b>書籍型/報告書型文書</b>	<b>91</b>
19.1	書籍型/報告書型シエルの特性 . . . . .	91
19.2	項目別詳細 . . . . .	92
<b>第 20 章</b>	<b>マスター/サブ文書</b>	<b>97</b>
20.1	マスター/サブ文書構成 . . . . .	97

---

20.2	マスター文書の作成 . . . . .	98
20.3	サブ文書の作成 . . . . .	99
20.4	統合タイプセット . . . . .	100
<b>第 21 章</b>	<b>スライド</b>	<b>101</b>
21.1	jsarticle slide オプション . . . . .	101
21.2	Beamer スライド . . . . .	103
<b>第 22 章</b>	<b>その他</b>	<b>105</b>
22.1	ハイパーリンク . . . . .	105
22.2	HTML, RTF エクスポート . . . . .	105
22.3	学会への論文投稿 . . . . .	106
22.4	その他の注意事項 . . . . .	107
<b>索引</b>		<b>109</b>

## 1

# クラスファイルとは


SWP/SW は L<sup>A</sup>T<sub>E</sub>X2e を基盤とする製品であるため、クラスファイルやパッケージについて基本的な理解を持っていないと応用がきかないことになります。

## 1.1 クラスファイルとは

T<sub>E</sub>X の世界で文書を作成する場合、**クラスファイル**として何を使うかは

```
\documentclass{jsarticle}
```

のように文書の先頭で指定する項目です。このクラスファイルの選択によって整形（タイプセット）される文書の基本様式が決まってきます。article 型のクラスファイルを使用すれば論文型の文書が、book 型のクラスファイルを使用すれば書籍型の文書が生成されます。

一方、SWP/SW の場合には**シェル**という概念が前面に据えられているため少々わかりにくいですが、「新規作成」ボタン  を操作することによってシェルを選択した時点でクラスファイルの選択が行われたこととなります。例えば“Standard LaTeX”シェルフォルダ内の“Japanese Article[jsarticle]”というシェルを選択すればクラスファイルとして日本語論文作成用の jsarticle を選択したこととなりますし、“Articles”シェルフォルダ内の“REVTeX 4”というシェルを選択した場合には revtex4 というクラスファイルが選択されたこととなります。SWP/SW の場合、クラスファイルが何であるかは裏に隠れて良く見えませんが、「タイプセット」メニュー：「オプションとパッケージ」：「クラスオプション」タブと操作することによって確認することができます。

## 1.2 クラスファイルの実体

クラスファイルは SWP/SW によって生成された  $\text{T}_{\text{E}}\text{X}$  のソース（ユーザによって作成された文書）を処理し、所定の様式に仕上げる機能を持った  $\text{T}_{\text{E}}\text{X}$  のコードです。その実体は拡張子 `.cls` が付いたファイルとして `\swp55 (\sw55)` プログラムフォルダ、あるいは `\ptex` プログラムフォルダ内に収容されています。それらは通常のテキストファイルとして構成されているので、WordPad のようなテキストエディタを用いればその内容を確認することができます。例えば欧文用の `article.cls` であれば

```
\swp55\TCITeX\TeX\LaTeX\base
```

に、和文用の `jsarticle.cls` であれば

```
\ptex\texmf\ptex\platex\jsclass
```

に格納されています。ただしくれぐれもそれらに変更を加えないように注意してください。

## 1.3 パッケージファイル

クラスファイルと似たものに**パッケージファイル**というものがあります。クラスファイルは文書の**基本構成**にかかわりますが、パッケージはどちらかという特定の**周辺機能**にかかわるものがほとんどです。例えばカラーを扱う機能はクラスファイルには含まれていませんが、`color` パッケージを付加することによってその扱いが可能になります。要はクラスファイルの機能を補うものと考えれば良いでしょう。実体はクラスファイル同様  $\text{T}_{\text{E}}\text{X}$  のコードですが、ファイルの拡張子としては `.sty` が用いられます。

SWP/SW の場合、文書にパッケージを追加するには「**タイプセット**」メニュー：「**オプションとパッケージ**」：「**パッケージオプション**」タブと操作し、一覧の中から特定のパッケージを

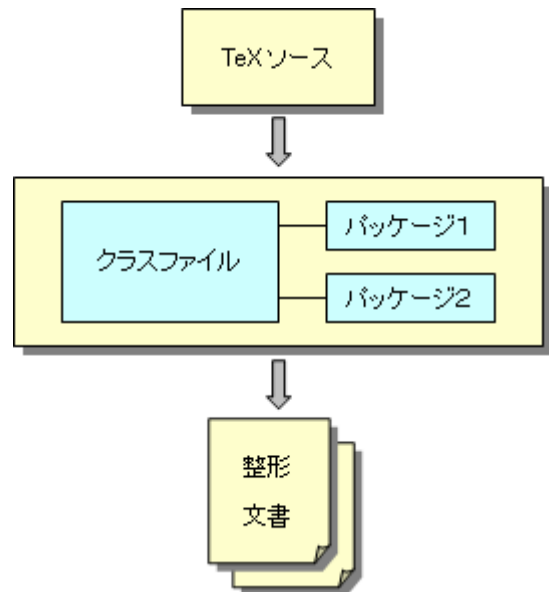


図 1-1 クラスファイルとパッケージ

選択する形を取りますが、結果的には T<sub>E</sub>X のソース上で文書の先頭部に

```
\usepackage{color}
```

といった宣言文が追加されることとなります。これによってパッケージ固有の各種コマンド（例えば `\color` コマンド等）が文書中で利用できるようになります\*<sup>1</sup>。

## 1.4 クラスオプション

ほとんどのクラスファイルには整形様式をさまざまに調整するための**クラスオプション**が数多く用意されています。これらをうまく利用して目的に合った文書作成を行ってください。ここでは SWP/SW v5.5 で提供されている `jsarticle` を例にとって有用なオプションのいくつかを紹介します。なお、クラスオプションの設定/変更は「**タイプセット**」メニュー：「**オプションとパッケージ**」：「**クラスオプション**」タブと操作し、「**編集**」ボタンをクリックすることによって行えます\*<sup>2</sup>。

### (1) 文字サイズ

“Body text point size” という項目によって基盤となる文字サイズ\*<sup>3</sup>を選択できます。Body text (テキスト体) という表記にはなっていますが、数式用の文字サイズもそれに連動した形で設定されます。従来のクラスファイルでは 10/11/12pt の 3 種類しか選択肢はありませんでしたが、`jsarticle` ではスライド作成機能をサポートした関係で、9pt から 43pt までの多彩な文字サイズがサポートされています。

### (2) 用紙サイズと向き

“Paper size”、“Orientation” という項目によって、それぞれ用紙サイズとその向きが設定できます。用紙サイズのデフォルトは日本語文書の場合 A4 ですが、欧文書の場合にはレターサイズがデフォルトとなっているのでご注意ください。

---

\*<sup>1</sup> SWP/SW 文書中でこれらの T<sub>E</sub>X コマンドを利用するためには T<sub>E</sub>X フィールドの機能を用いる必要があります。操作方法については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0009.html> をご参照ください。

\*<sup>2</sup> パッケージにも種々のオプションが用意されていることがあります。こちらは「**パッケージオプション**」タブを用いることによって設定/変更が行えます。

\*<sup>3</sup> 「テキストタグ」フィールドで拡大/縮小操作を行う際の基盤となる文字サイズのことを言います。

### (3) Papersize special

従来の日本語文書用クラスファイル (jarticle、等) の場合、クラスオプションで用紙サイズや向きを設定しても dviout, dvipdfmx といったドライバはそれに反応せず、ドライバ側での設定変更が別に必要でした。jsclass (jsarticle, jsbook, jsreport の総称) では“Papersize special”コマンドがサポートされているため、このオプションを有効にしておくとう紙の設定はドライバ側と連動されるようになります。

### (4) 表紙

論文型文書の場合、通常表題部は本文の先頭に置かれ専用のページを占めることはありませんが、“Title page”オプションを指定した場合には表題部と概要部とがそれぞれ独立したページに出力され、本文とは切り離された形で整形されます。なお、書籍型文書の場合には表紙として独立させる形がデフォルトとなります。

### (5) 2 段組

“Columns”オプションを用いることによって 2 段組文書の作成が容易に行えます。2 段組を指定した場合、デフォルトでは概要部も段組部分に配置される仕様となっていますが、jsarticle の場合には概要部を表題部と同様、段組外に配置することも可能になりました。“Standard LaTeX”シェルフォルダ中に

- [jsarticle] twocolumn-type1
- [jsarticle] twocolumn-type2

という 2 種類のシェルを用意していますのでご活用ください。

### (6) ディスプレイ数式

数式単独で独立した行に配置されるものをディスプレイ数式と呼びます。通常それらは中央揃えの様式で整形されますが、“Displayed equations”オプションで“Flush left”を選択した場合には左揃えの配置となります。

## (7) 数式番号

ディスプレイ数式には数式番号を付加することができます\*4。通常、数式番号は行の右端に設定されますが、“Equation numbering”オプションで“Numbers on left”を選択した場合には行の左端に設定されるようになります。

## (8) スライド

これも jsclass に固有のオプションです。“Slide”オプションを指定した場合にはプレゼンテーションスライド用の様式で文書が整形されます。数式に関し  $\text{T}_\text{E}\text{X}$  のクオリティを維持したまま、簡単にスライドが作成が行えます。スライド作成の便宜を図るため、SWP/SW v5.5 では“Standard LaTeX”シェルフォルダ内に

- [jsarticle] slide1
- [jsarticle] slide2

というスライド作成用のシェルを用意してありますのでご活用ください。さらに v5.5 では Beamer スライドの機能もダウンロードして使用することができます。詳しくは SWP/SW BetterUse ページ：“スライド”の項

[http://www.lightstone.co.jp/latex/bu\\_index9.html](http://www.lightstone.co.jp/latex/bu_index9.html)

をご参照ください。

## 1.5 学会固有クラスファイル

SWP/SW 環境下で文書作成を行う場合、通常は“Standard LaTeX”シェルフォルダ中の標準シェルを使用します。しかし学会とか出版社の中には独自の様式を設定するため、固有のクラスファイル (.cls) やスタイルファイル (.sty) を用意しているところも少なくありません。このような場合にはセクション 1.2 に記したフォルダ内に該当クラスファイルやスタイルファイルを組み込む形で対応することになります。ただし欧米の主要な学会のものについては“Articles”シェルフォルダ内にあらかじめ組み込まれて提供されているため、特別な作業を経ずとも利用できる場合があります。しかし製品に組み込まれているものでは古くなってしまったり、あるいは投稿先が日本の学会だったりした場合には、別途組み込み作業が必要になります。

ライトストーンではこれら学会固有クラスファイル/スタイルファイルに対するサポートサービスを提供しており、SWP/SW 環境下での投稿論文作成を支援しています。詳しくは SWP/SW BetterUse ページ：“論文投稿”の項

---

\*4 ディスプレイ数式の右端を右クリック、「プロパティ」と操作します。

[http://www.lightstone.co.jp/latex/bu\\_index1.html](http://www.lightstone.co.jp/latex/bu_index1.html)

をご参照ください。



SWP/SW 環境下でタイプセットを行う場合、2種類の  $\text{T}_{\text{E}}\text{X}$  の体系が関係してきます。一つは欧文書用の  $\text{TrueT}_{\text{E}}\text{X}$  で `\swp55(\sw55)` プログラムフォルダ内に格納されています。もう一つは和文書用の  $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  で `\ptex` プログラムフォルダ内に存在します。これは SWP/SW 本体とは別の WinForme をインストールすることによってシステム内に組み込まれます。「タイプセット」メニューで「英語タイプセット」を使用した場合には  $\text{TrueT}_{\text{E}}\text{X}$  の体系が、「日本語タイプセット」を使用した場合には  $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  の体系がそれぞれ起動されます。



## 2

## フロントマターの作成

SWP/SW の場合、表題等はフロントmatter専用のダイアログ中に設定するのが一般的ですが、使い方には慣れを要しますので少々説明を加えておきます。

## 2.1 フロントmatterダイアログ

SWP/SW を起動し適当なシェルを選択するとシェル文書の内容が操作画面に表示されます。最初は誰でもそこが文書の先頭と思うわけですが実はそうではありません。「タイプセット」メニューの中に「プリアンブル」と「フロントmatter」という項目がありますが、SWP/SW によって生成される  $\text{T}_{\text{E}}\text{X}$  文書上はこれら2つが本文に先行する形となります。

フロントmatter部には表題情報と概要 (abstract) が配置されます。書籍型文書だと独自の表紙を作成する関係から「フロントmatter」ダイアログを使用せずに文書を作成することもあります。論文型文書の場合にはこのダイアログ上で項目を設定するケースがほとんどです\*1。

この表題部、概要部の整形様式はクラスファイルによって大きく異なります。ここでは `jsarticle` の場合を例にとってフロントmatterの作成要領を説明します。

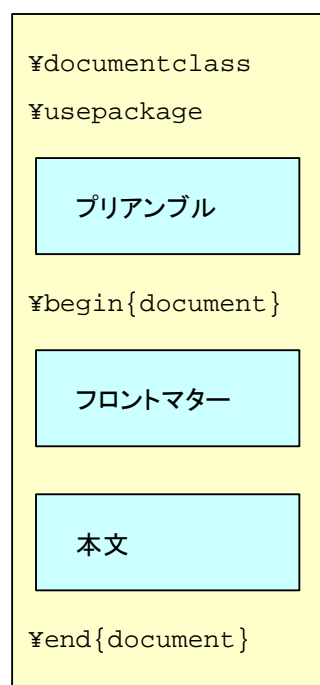


図 2-1 文書の基本構成

\*1 学会固有のクラスファイル/スタイルファイルを使用した場合にはこの限りではありません。

## 2.2 フロントマター作成例

標準的なシェルの中にはあらかじめフロントマター関連項目を設定済みのもありますが、ここでは白紙の状態からスタートすることにします。図 2-2 はフロントマターダイアログへの入力例、図 2-3, 2-4 はそのタイプセット結果を示しています。

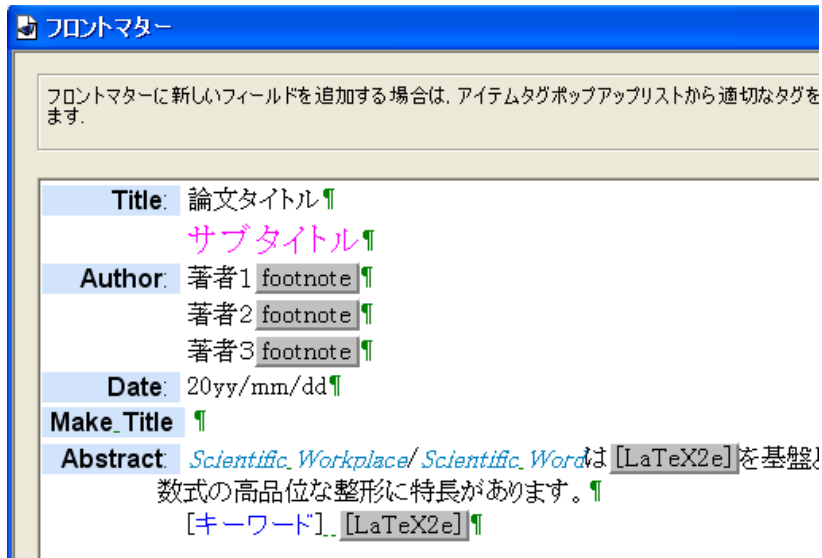


図 2-2 フロントマター設定例

### (1) アイテムの設定

白紙のフロントマター上に“Title”等のアイテムを設定するにはメイン操作画面上の「アイテムタグ」フィールドを使用します。



「フロントマター」ダイアログに限りませんが、これらのダイアログが表示されている状態でも、その背景にあるメイン操作画面上のメニューやボタンは有効です。それによって数式モードへの切替えやテキストタグの設定、脚注の設定等の操作がダイアログ上でも可能です。

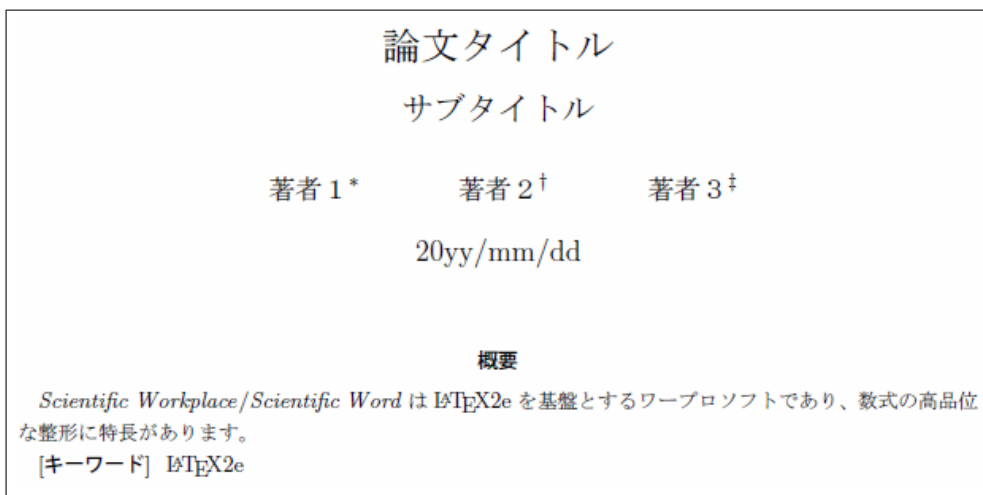


図 2-3 タイプセット結果

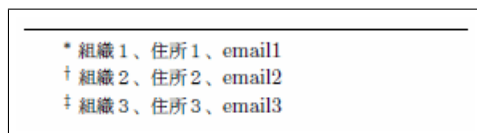


図 2-4 脚注部


## (2) 継続行の入力

“Title”等のアイテムフィールド末尾で改行すれば、該当アイテム内の継続行を入力できます。サブタイトルや英文タイトルはこのようにして入力します。


## (3) テキスタグ

jsarticle の場合、論文タイトル用の文字は `\LARGE` サイズ（標準の約 1.7 倍）でタイプセットされます。何も指定しないと 2 行目にあるサブタイトルでも同じ文字サイズになってしまうので、上の例ではサブタイトルに対し `\Large`（標準の約 1.4 倍）というテキストタグを付加してあります。

#### (4) Author の設定

サブタイトル行で改行すると“Title”タグが付いた状態で3行目が設定されます。これを“Author”タグに変更するにはまず  ボタンを使って“Title”タグを除去、次に「アイテムタグ」中から“Author”アイテムを選択してください。2人目、3人目の著者については継続行に入力できます。



アイテムタグの除去には  ボタンを使用してください。なお、行や文字列にどのようなタグが付いているかはタグバーフィールド中の表示で確認できます。

#### (5) Address の設定

著者の所属する組織や email 等の情報は“Address”アイテムを設定し表題部に入力する方法もありますが、ここでは脚注部に設定するアプローチを示します。著者名右端で「挿入」メニュー:「ノート」と操作し、ダイアログ中に所属等の情報を入力してください。



脚注情報を確認したり変更する場合にはグレイボックスを右クリック、「プロパティ」と操作します。



脚注に用いられる記号の様式はフロントマター部と本文中とは異なります。調整方法については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0075.html> をご参照ください。

#### (6) Date の設定

日付けを設定するには“Date”アイテムを使用します。このアイテムが設定されていない場合にはタイプセットした当日の日付けが出力されます。日付けを全く出力したくない場合には、“Date”アイテムを設定後、そのフィールドを空欄にしてください。

#### (7) 表題部の出力

表題部の出力は `\maketitle` コマンドで制御します。「アイテムタグ」フィールド中の“Make Title”アイテムを設定してください。論文型文書では余り関係ありませんが、目次作成や図表一覧作成用のアイテムも用意されています。

### (8) 概要部の作成

「アイテムタグ」フィールド中の“Abstract”アイテム\*2を設定し、論文概要を記述します。キーワードについては特にアイテムが用意されていないので、概要テキストの一部という形で設定してください。上の例では“キーワード”の部分に太字のタグ（テキストタグ）を設定してあります。



“概要”という見出しはクラスファイルによって自動生成されます。表現を変更したい場合には SWP/SW BetterUse ページ  
<http://www.lightstone.co.jp/latex/kb0073.html>  
をご参照ください。



---

\*2 書籍型文書の場合には選択できません。



## 3

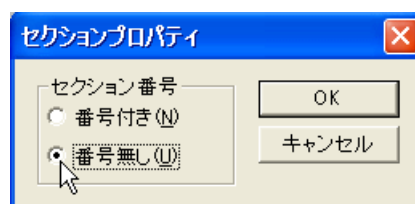
## 見出しの制御

章や節（セクション）等の見出しに関連した事項について整理しておきます。

### 3.1 見出し番号

章とか節（セクション）の設定は「セクション/ボディタグ」フィールド内のタグを付加することで行います\*1。これらの division に対する番号付けは TeX によって自動的に行われるので通常は意識する必要はありませんが、時には見出しに番号を付けたくないとか、独自様式の番号を付けたいといったニーズが発生します。

そのような場合には見出し左端にカーソルを位置付け右クリック、「プロパティ」と操作すると右に示したようなダイアログが表示されるので、“番号なし”を選択してください。TeX による番号付けは行われなくなるので自由な番号を設定できるようになります。ただし書籍型文書でこの指定を行う場合にはいろいろな注意事項が伴います。セクション 3.3 を参照してください。




\*1 論文型のクラスファイルを使用している場合には章というタグは利用できません。

## 3.2 見出し番号の参照

上述のように章とか節に対しては自動的に番号が振られます。一方、本文中で“セクション □ 参照”と書くことは良くあります。数式番号や表番号を参照する場合と同様、クロスリファレンスの仕掛けを使うことになるのですが、そのためには章とか節に対して参照用のキーを定義しておかなくてはなりません。見出しの次の行で「挿入」メニュー：「マーカー」と操作し参照用のキーを設定してください。例えば sec1 というキーを設定した場合には次のような  $\text{T}_{\text{E}}\text{X}$  コードが生成されることとなります。

```
\label{sec1}
```

一方、“セクション □ 参照”という形で参照する際には □ の部分でクロスリファレンスボタン  を押し、該当する参照用キーを選択します。 $\text{T}_{\text{E}}\text{X}$  のコンパイルを通すとこのキーの部分が実際の division 番号で置き換えられます。

## 3.3 書籍型文書での留意点

章とか節の見出し (division headers) にはそれなりの **タグ** を付けるのが  $\text{T}_{\text{E}}\text{X}$  の流儀です。しかし標準設定される見出し様式では飽き足らないといった場合に、独自様式の見出しを設定することがよくあります\*2。その場合には所定のタグが付かないため、 $\text{T}_{\text{E}}\text{X}$  は章とか節といった認識をしてくれません。論文型の文書の場合、それほど弊害は目立ちませんが、書籍型文書の場合には次に示すような悪影響が生じます。また、所定のタグを使っても見出し番号を省いた場合には同列の扱いとなるのでご注意ください。

- 章とか節の見出しは目次の作成に際して利用されます。所定のタグが付けられていないと目次に反映されません。見出し番号を省いた場合も同様の扱いとなりますが、`\addcontentsline` という  $\text{T}_{\text{E}}\text{X}$  コマンドを使用すれば目次への反映が可能になります。用法については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0083.html> をご参照ください。
- 書籍型文書の場合には通常、各ページの上部に **ヘッダ** (柱) が出力されますが、それにも見出しのテキストが用いられます。章や節にタグを付けなかった場合、あるいは番号を省略した場合には、いつまでも一つ前の章や節の見出しテキストがヘッダを飾ることになるので注意が必要です。なお、`\markboth` という  $\text{T}_{\text{E}}\text{X}$  コマンドを使用すればヘッダへの出力情報を制御することができます。

\*2 本書でも章の見出しには画像を用いています。

- 書籍型文書の場合、数式番号の様式は (ch.n) という形になります (デフォルト)。ここに *ch* は章番号を意味します。図や表に対しても  $\TeX$  が自動生成する番号は同様の様式となります。章のタグを設定しなかったり番号を省略した場合にはこれらのオブジェクトに対する番号にも影響が及ぶので注意してください\*3。

### 3.4 見出し番号の設定

章や節の番号はデフォルトでは 1 から開始されます。しかし時には別の番号から開始したいといったこともあります。 $\TeX$  の `\setcounter` コマンドを使用することによって番号の設定が行えます。詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0025.html>

をご参照ください。

### 3.5 標準見出しの変更

章や節の見出しは一般的にはユーザが設定しますが、中には“参考文献”とか“概要”のようにクラスファイルが自動生成する見出しもあります。これらの見出しテキストも  $\TeX$  の `\renewcommand` コマンドを使用することによって変更が可能です。詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0073.html>

をご参照ください。

### 3.6 見出しの様式調整

article や book, report といった欧文用文書の場合には次のようなパッケージを利用することで見出し様式をカスタマイズできます。ただしこれらは和文用ドキュメントクラスでは使用できません。

<i>sectsty</i> パッケージ	章や節の見出しの配置 (左寄せ/中央/右寄せ) や文字フォント (タイプ、サイズ) を調整できます。
<i>fncychap</i> パッケージ	章見出しの様式として少々凝った 6 種類のものが用意されています。

\*3 論文型文書であっても *numinsec* パッケージを用いセクションごとの番号様式を採用した場合には同様の依存性が生じます。



本書では jsbook 標準の章見出しを用いつつ、その上に MS Word で作成した画像をかぶせるという少々強引な手法で独自の章見出しを設定しています。興味のある方は SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0087.html>

をご参照ください。



## 4

## スペースの制御


スペースの調整は見やすい文書を作成する上で重要な項目です。本章では SWP/SW の持つ木目の細かいスペース調整機能について説明します。

## 4.1 SWP/SW におけるスペースの制御

通常の WYSIWYG(What You See Is What You Get) スタイルのワープロソフトと異なり、SWP/SW の場合は  $\text{T}_{\text{E}}\text{X}$  が基盤にあるため、コンパイルというプロセスを経ないと印刷イメージを確認できません。実際 SWP/SW の操作画面上にはページの区切りもなければ、その横幅を示すガイド也没有。SWP/SW の画面上で 1 行空けたつもりが実際にタイプセットすると空いていないため、戸惑われた方も多いのではないのでしょうか。これは 1 つの改行は無視し、2 つ続いた場合に初めて空行を置くという  $\text{T}_{\text{E}}\text{X}$  の特性によるものです。

いずれにせよ SWP/SW での文書作成においてスペースの制御には少々コツを要しますので、ここではまず基本的な事項について整理しておきます。

- 制御記号ボタン

制御記号ボタン  をオンにすると、設定されているスペースの量や改行、改ページの位置が確認できるようになります。

- $\text{T}_{\text{E}}\text{X}$  に一任

セクション見出しの前後には空白を確保する等、 $\text{T}_{\text{E}}\text{X}$  がタイプセットの過程で配慮してくれる部分が少なからずあります。基本は  $\text{T}_{\text{E}}\text{X}$  に一任し変更は微調整に留めるとした方が、一般的に良い結果が得られます。

- 「スペース」コマンド

$\text{T}_{\text{E}}\text{X}$ には多様なスペース制御機能が用意されており、高品位な文書作成の重要な一面を担っています。SWP/SWにおいては「挿入」メニュー：「スペース」と操作することにより、これらの機能が簡便に利用できるようになっています。

## 4.2 横スペース

「挿入」メニュー：「スペース」：「横スペース」と操作することによって設定できる横スペースには次のような種類があります。

横スペース	$\text{T}_{\text{E}}\text{X}$ コマンド	備考
通常のスペース		半角空白、4/18 em
強制スペース	<code>\</code>	半角空白、4/18 em
改行できないスペース	<code>~</code>	半角空白、4/18 em
エムスペース	<code>\quad</code>	全角空白、em (“M”の横幅)
2エムスペース	<code>\qquad</code>	2 em
小スペース	<code>\,</code>	3/18 em
大スペース	<code>\;</code>	5/18 em
斜体補正	<code>\/</code>	左側の文字に依存
負の小スペース	<code>\!</code>	-3/18 em
ゼロスペース	<code>{}</code>	0 em
インデントなし	<code>\noindent</code>	インデント抑止
カスタム - 固定長	<code>\hspace</code>	任意長
カスタム - stretchy	<code>\hfill</code>	伸長

このうち知っておくと役に立つ情報について補足しておきます。

- **M**で示される数式モード中ではスペースバーは使えません（半角スペースの場合）。数式中のスペースの配分は基本的に  $\text{T}_{\text{E}}\text{X}$  が制御しているからです。しかし「挿入」メニュー：「スペース」：「横スペース」と操作し強制的にスペースを入れることはできます。例えば次の例は上付き添え字に半角スペース（“強制スペース”）を付加した例です。

標準形	スペースを挿入
$x_i^2$	$x_i^2$

- 逆に数式等で間隔が開き過ぎという場合には負のスペース（“負の小スペース”、または“カスタム”で負値を指定）を挿入し調整することができます。
- “インデントなし”を使うことによってパラグラフ先頭部における字下げ(indentation)を抑制することができます\*<sup>1</sup>。
- `\hfill` の機能を使うと、次のような例で番号の部分を右端に追いやることができます。

## 論文タイトル

No. xxx
---------

### 4.3 縦スペース

「挿入」メニュー：「スペース」：「縦スペース」と操作することによって設定できる縦スペースには次のような種類があります。

縦スペース	TeX コマンド	備考
小スキップ	<code>\smallskip</code>	1/3 空行
中スキップ	<code>\medskip</code>	1/2 空行
大スキップ	<code>\bigskip</code>	空行
支柱	<code>\strut</code>	支柱
数式支柱	<code>\mathstrut</code>	数式用支柱
カスタム - <code>\vspace*</code>	<code>\vspace*</code>	任意長
カスタム - <code>\vspace</code>	<code>\vspace</code>	任意長

このうち知っておくと役に立つ情報について補足しておきます。

- 空行を挿入するには無闇に改行キーを叩くのではなく、“大/中/小スキップ”の機能を使用した方が確実です。
- “カスタム”を選択することにより任意長の縦スペースを設定できます。スペースが空きすぎたような場合には負の値を指定することにより、無駄な空白を狭めることができます。

\*<sup>1</sup> 文書全体で字下げを抑制する方法については SWP/SW BetterUse ページ  
<http://www.lightstone.co.jp/latex/kb0032.html>  
 をご参照ください。

- 任意長の縦スペースを設定する場合、 $\text{T}_{\text{E}}\text{X}$  フィールド\*<sup>2</sup>中に `\vspace` コマンドを埋め込む形でも同じことが行えます。その際、長さを 3cm のように直接指定するのではなく、

```
\vspace{1.5\baselineskip}
```

のように指定することもできます。このようにすれば行送りの幅を単位として空行の量を制御できます。



`\vspace` の場合、ページの上端、下端では縦スペースが入りません。  
`\vspace*` の場合には上下端であっても指定された縦スペースが設定されます。

- フラグメントフィールド中の `\vfill` の機能を使用するとページ縦方向に伸長性のスペースを挿入することができます。ページ下端でも有効なものとして 2 種類が用意されています。

## 4.4 ブレーク

「挿入」メニュー：「スペース」：「ブレーク」と操作することによって設定できるブレークの機能も改ページ、強制改行等の操作で良く使用されます。




ブレーク	$\text{T}_{\text{E}}\text{X}$ コマンド	備考
Allowbreak	<code>\allowbreak</code>	改行の許可
Discretionary hyphen	<code>\-</code>	hyphenation の許可
No break	<code>\nolinebreak</code>	改行抑止
ページブレーク	<code>\pagebreak</code>	改ページ
ニューページ	<code>\newpage</code>	改ページ後、新たなページに位置付け
ラインブレーク	<code>\linebreak</code>	強制改行 + justification
ニューライン	<code>\newline</code>	強制改行
カスタムニューライン	<code>\\[]</code>	改行 + 縦スペース

- 改ページを行うには“ページブレーク”、または“ニューページ”を使用します。

\*<sup>2</sup>  $\text{T}_{\text{E}}\text{X}$  フィールドについては SWP/SW BetterUse ページ  
<http://www.lightstone.co.jp/latex/kb0009.html>  
 をご参照ください。

- 脚注とかキャプションを入力する領域内で通常の改行ボタンを押すとダイアログが閉じてしまうので、操作に戸惑われた方も多いのではないのでしょうか。このようなダイアログ中で改行を行い、新たな行を入力する場合にはこのように“ニューライン”**⌵**を設定してください。“ラインブレイク”**⌵**でも同様の強制改行は行えますが、justification の機能が働いてしまうのでご注意ください。
- 行末で  $\arctan x$  といった数式を入力したり、タイプライタ書式で <http://www.lightstone.co.jp> といった長い文字列を入力すると hyphenation が効かないため、文字列が余白部にはみ出してしまふことがあります。このような場合には“Allowbreak”の機能を用いブレイク可能な位置を  $\TeX$  に示してやる方法が有効です。

## 4.5 ルール

「挿入」メニュー：「スペース」と操作すると「ルール」という選択肢も目に止まると思います。元々ルールとは  のような縦の線を引く機能を提供するものですが、スペース制御という観点からすると幅 0 のルールが重要な意味を持てきます。もちろん幅が 0 なのでルール自体は見えませんが、見えない支柱として便利な機能を提供します。例えば  と  を用いて次のような数式を作成してみましょう。

$$\left\{ \begin{array}{l} P_1 = \frac{1}{\sqrt{(x-1)^2 + y^2}} \\ P_2 = \frac{1}{\sqrt{(x+1)^2 + y^2}} \end{array} \right.$$

この場合、数式は  $2 \times 1$  の行列中に配置されることになるため、行間がほとんどない整形結果となります。

$$\left\{ \begin{array}{l} P_1 = \frac{1}{\sqrt{(x-1)^2 + y^2}} \\ P_2 = \frac{1}{\sqrt{(x+1)^2 + y^2}} \end{array} \right.$$

このようにときに幅 0 のルールが役に立ちます。例えば今の場合、2 行目の数式右端に高さ 20pt のルール（幅、持上げは共に 0）を挿入してみてください。次のようなタイプセット結果が得られるはずですよ。

$$\left\{ \begin{array}{l} P_1 = \frac{1}{\sqrt{(x-1)^2 + y^2}} \\ P_2 = \frac{1}{\sqrt{(x+1)^2 + y^2}} \end{array} \right.$$

この例で言えば、1 行目の数式内に持上げが負の値のルールを設定する形でも同様の効果を得ることができます。それぞれの局面に応じて適切なルールをお試しください。



## 5

# 箇条書きアイテム

文書中に箇条書きアイテムを設定することは良くありますが、SWP/SW の場合、用法には慣れを要しますので少々説明を加えておきます。

## 5.1 リストアイテム

SWP/SW 環境で箇条書きアイテムを設定するには「アイテムタグ」フィールド中の次の 3 種類のアイテムを使用します。

- 記号付きリストアイテム (Bullet List Item)
- 番号付きリストアイテム (Numbered List Item)
- 見出し付きリストアイテム (Description List Item)

各々  $\text{T}_{\text{E}}\text{X}$  の `itemize` 環境、`enumerate` 環境、`description` 環境に展開されることになります。

## 5.2 基本操作


リストアイテムの構成は次のようになります。

lead-in object (● 等)	パラグラフ 1 パラグラフ 2 ...
----------------------	---------------------------


### (1) パラグラフの追加

パラグラフ長に制限はありませんが、ENTER キーを押すと次のアイテムが設定されてしまうので注意が必要です。例えばパラグラフ 1 の末尾で ENTER キーを押した後、同一リストアイテムに属するパラグラフ 2 を開始するには BACKSPACE キーを押してください。Lead-in object が消え、継続パラグラフの入力が可能になります。なお、パラグラフ間の行間を調整するには「挿入」メニュー：「縦スペース」と操作してください。

### (2) 環境からの離脱

リストアイテムの環境から脱するためには  ボタンをクリックしてください。例えば記号付きリストアイテムを設定した場合、 $\text{T}_{\text{E}}\text{X}$  コード上は次のような `itemize` 環境が生成されます。


```
\begin{itemize}
\item パラグラフ 1
\item パラグラフ 2
\end{itemize}
```

このリスト環境中で ENTER キーを押すと次の `\item` コマンドが生成されてしまいます。  ボタンをクリックすることは、`\begin{itemize}` と `\end{itemize}` の間に挟まれたこの環境から離脱することを意味します。

### (3) ネスティング

リストアイテムは 4 階層の深さまでネストさせることができます。

記号付きリストアイテム	番号付きリストアイテム
<ul style="list-style-type: none"> <li>• XXX</li> <li>– XXX</li> <li>* XXX</li> <li>· XXX</li> </ul>	<ol style="list-style-type: none"> <li>1. XXX</li> <li>(a) XXX</li> <li>i. XXX</li> <li>A. XXX</li> </ol>

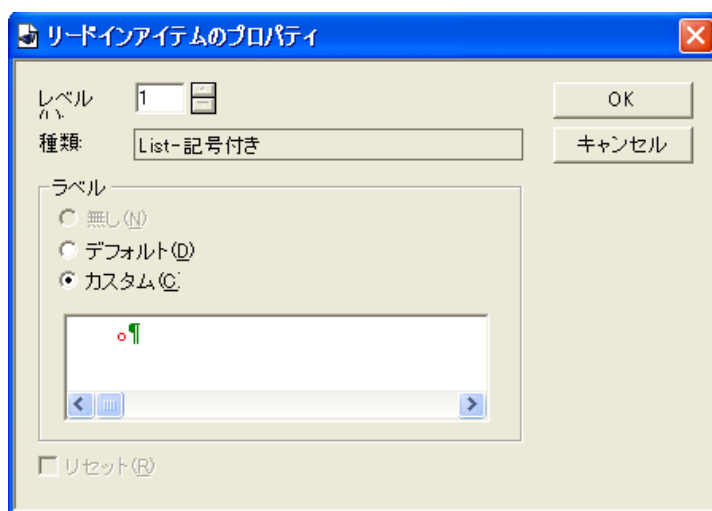
具体的にはリスト環境中で再度リストアイテムを設定することによって環境はネスト化されます。逆に  ボタンをクリックすればネスト階層が一つ浅くなります。カーソルが置かれている位置がリスト環境中であればその旨「アイテムタグ」フィールド内にタグの表示が現れますが、階層の深さまでは表示されないのでご注意ください。



リスト環境から脱したつもりがそうでなかった場合には種々のコンパイルエラーを招く可能性があります。余計なタグが残留していないか、タグフィールド中の表示情報には十分注意してください。

### 5.3 リードインオブジェクト

リードイン (lead-in) オブジェクトとして使用する記号や番号の種類、様式はあらかじめ設定されていますが、それを変更することも可能です。オブジェクトの右側にカーソルを位置付け右クリック、「プロパティ」と操作すると次のようなダイアログが表示されるので、“カスタム”を選択の上、適切な記号や数字を入力してください。



個々にその都度変更するのではなく文書を通して様式を変更することもできますが、その方法については SWP/SW BetterUse ページ

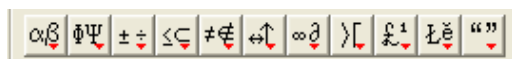
<http://www.lightstone.co.jp/latex/kb0089.html>

<http://www.lightstone.co.jp/latex/kb0088.html>

をご参照ください。



リードインオブジェクトとしては記号パネル



中にある数式記号も使用できます。



①、②、③のような丸囲いの文字は日本語文字コード中にはありますが、OS 依存となるので TeX としてのサポートは限定的なものとなります。DVI タイプセットでは出力できても PDF タイプセットでは文字が出力されません。なお、Adobe **Acrobat** の機能が利用できる環境であれば、DVI 文書を Adobe PDF プリンタに出力する形で、丸囲いの文字を含む PDF 文書の作成が行えます。



## 6

## テキストタグ

文字に関する属性（フォント、サイズ）の設定には通常テキストタグを使用しますが、いくつかわかりにくい点があるので説明を補足します。

## 6.1 テキストタグ

SWP/SW の「テキストタグ」フィールド内には文字列の属性を制御するための種々の属性が用意されています。

- テキスト属性といったときの第 1 の柱は文字フォントです。英文テキスト用の書体としては Roman（デフォルト）、**Bold**、*Italics*、*Slanted*、**Sans Serif**、**Typewriter**、**SMALL CAPS** が選択できます。一方、和文用書体については通常体が明朝、太字がゴシックという設定になっています。なお、強調タグを選択した場合には、全角文字は太字ゴシック体で、半角文字はイタリック体でタイプセットされます。

数式用フォントとしては Computer Modern が使用されます。標準的な書体の他に *CALLOGRAPHIC*、**BLACKBOARD BOLD**、*Fraktur* といった数式モード専用の書体も用意されています。なおパッケージを組み込むことで Computer Modern 以外のフォントも使用できるようになります。

- テキスト属性のもう一つの柱はサイズです。基盤となるフォントサイズに対し 50% から 250% の間の固定倍率で拡大/縮小が行えます。



基盤となる文字サイズはクラスオプション（セクション 1.4 参照）によって規定されます。また、数式の文字サイズはテキストタグによっては変更できません。



カラーの使用は *color* パッケージを組み込むことによって可能となりますが、それについては章を改めて説明を行います。

## 6.2 タグの重畳

SWP/SW のテキストタグ操作では原則、同じ文字列上に複数のタグを重ねることはできません。例えば太字のイタリック体などというのが良い例です。このような場合に関連する TeX のコマンドを知っていると対応が可能です。

文字フォント制御コマンド		文字サイズ制御コマンド	
<code>\textrm</code>	Roman	<code>\tiny</code>	縮小 50%
<code>\textbf</code>	<b>Boldface</b>	<code>\scriptsize</code>	縮小 70%
<code>\textit</code>	<i>Italics</i>	<code>\footnotesize</code>	縮小 80%
<code>\textsl</code>	<i>Slanted</i>	<code>\small</code>	縮小 90%
<code>\textsf</code>	Sans Serif	<code>\normalsize</code>	標準 100%
<code>\texttt</code>	Typewriter	<code>\large</code>	拡大 120%
<code>\textsc</code>	SMALL CAPS	<code>\Large</code>	拡大 140%
<code>\textgt</code>	ゴシック	<code>\LARGE</code>	拡大 170%
		<code>\huge</code>	拡大 200%
		<code>\Huge</code>	拡大 250%

例えば TeX フィールド<sup>\*1</sup>中に

```
\textbf{\textit{Bold Italics}}
```

と書けば ***Bold Italics*** とタイプセットされますし、

```
{\Large \textgt{ 太字で拡大 }}
```

と書くと**太字で拡大**のようにタイプセットされます。

<sup>\*1</sup> TeX フィールドについては SWP/SW BetterUse ページ  
<http://www.lightstone.co.jp/latex/kb0009.html>  
 をご参照ください。



たまにイタリック書体の代りに数式モードで文字列を入力するケースを見かけます。確かに斜体にはなるのですが、数式モードの場合には文字間のスペースを微妙に  $\text{\TeX}$  が調整することになるのでご注意ください。

数式書体	<i>difference</i>
イタリック体	<i>difference</i>

### 6.3 文字サイズの変更

パラグラフ全体の文字サイズを変更したいようなときなど、 $\text{\Large}$  とか  $\text{\small}$  といった  $\text{\TeX}$  コマンドを  $\text{\TeX}$  フィールド中に設定する方法もあります。例えば特定のパラグラフのみ 80% の縮小文字でタイプセットしたい場合には、パラグラフの先頭に  $\text{\TeX}$  フィールドを配置し、

$\text{\footnotesize}$

というコマンドを設定、一方末尾にはサイズを元に戻すべく

$\text{\normalsize}$

というコマンドを設定します。この場合、数式や表も一律拡大/縮小されます。

一例としてディスプレイ数式

$$y = \int_{-\infty}^{\infty} \frac{1}{x^2 + 1} dx \quad (6.1)$$

や表

<b>A</b>	<b>B</b>
aaa	bbb

を含んだようなパラグラフ全体を 80% 縮小の文字サイズでタイプセットしてみましょう。

これはディスプレイ数式

$$y = \int_{-\infty}^{\infty} \frac{1}{x^2 + 1} dx \quad (6.2)$$

や表

<b>A</b>	<b>B</b>
aaa	bbb

を含んだパラグラフ全体を 80% 縮小の文字サイズでタイプセットした例です。

## 6.4 数式文字の調整

### (1) ベクトル表記

数式モードで入力した  $u$  や  $v$  という文字に対し“太字 (Bold)”というテキストタグを付加すると、`\mathbf` という  $\text{\TeX}$  コマンドに展開されるため  $\mathbf{u}$  や  $\mathbf{v}$  のように直立体になってしまいます。これに対し“Bold Symbol”というタグを付けると  $\mathbf{u}$  や  $\mathbf{v}$  のように斜体のまま太字にできます。この場合、`\boldsymbol` という  $\text{\TeX}$  コマンドが使用されるわけですが、これは `amsmath` パッケージが提供する機能です。SWP/SW で作成された文書には無条件に `amsmath` パッケージが組み込まれるので、パッケージの追加という操作はこの場合必要ありません。この他に

- `\boldmath` コマンドを使用する
- `bm` パッケージを使用する

等の方法もありますが、詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0035.html>

をご参照ください。



SWP/SW によって自動的に組み込まれるパッケージには次のものがあります。

- `amsmath`
- `amsfonts`
- `amssymb`
- `graphicx`

### (2) 数式斜体

$\lambda$  や  $\mu$  といったギリシャ文字の小文字は斜体でタイプセットされますが、ギリシャ文字の大文字は  $\Psi$  のように直立体で出力する慣行となっています。例えば  $\Delta$  (`\Delta`) を斜体としたい場合には、数式モードの中で次のような  $\text{\TeX}$  コマンドを含む  $\text{\TeX}$  フィールド

```
\mathit{\Delta}
```

を設定すれば  $\Delta$  のような出力を得ることができます。

また数式モード中で

$$diff(x)$$

と記した場合、 $\text{\TeX}$  は  $d \times i \times f \times f(x)$  と解釈するため美しい整形は得られません。このような場合も `\mathit` コマンドを用いて

```
\mathit{diff}
```

と書いてやれば

$$\mathit{diff}(x)$$

のようなより整った表現を得ることができます。





## 7

## 数式 TeXnique [1]

SWP/SW 文書の主役は何と言っても数式でしょう。TeX の存在価値は数式主体の文書を高品位に整形できる点にあるからです。しかし数式に絡む“TeXnique”は山ほどあるので、まずは基本的な事項から整理して行くことにします。

## 7.1 数式の種別

TeX で数式を扱う場合には、**インライン数式**と**ディスプレイ数式**という 2 つの区分がある点に注意する必要があります。


### (1) インライン数式

インライン数式はテキスト行の中に数式を配置するものです。その場合、**テキストモード**と**数式モード**の切替えには **T** / **M** ボタンを使用します\*1。数式モードで例えば  $y = f(x)$  と入力した場合には

```
$y=f(x)$
```

という\$ではさまれた TeX コードが生成されます。数式によっては  $\sum_{k=1}^n$  のようにディスプレイ数式の場合とは異なる様式で整形されるものがあるので注意してください。

### (2) ディスプレイ数式

ディスプレイ数式は数式専用の行に配置されるもので、 ボタンのクリックによって設定されます。ディスプレイ数式は通常中央揃えで配置されますが、クラスオプションの指定によって左

\*1 数式関連のボタンをクリックした場合には自動的にモードが切り替わります。

揃えに設定を変更することもできます（セクション 1.4 参照）。ディスプレイ数式はそれが 1 行のみの場合と複数行からなる場合とで異なる TeX コードに展開されます。

- 単一行の場合

1 行のみのディスプレイ数式は TeX の `equation` 環境に展開されます。例えば

$$y = f(x)$$

のように入力した場合には

```
\begin{equation}
y=f(x)
\end{equation}
```

のようなコードに展開されます。なお、ディスプレイ数式として  $\sum$  を入力すると

$$\sum_{k=1}^n$$

のようにインライン数式の場合とは異なる様式で整形されます。

- 複数行の場合

複数行からなるディスプレイ数式の場合には行と行の間での位置合せの機能が必要になるため、単一行の場合と異なり、*amsmath* パッケージ\*2の `align` 環境に展開されます\*3。例えば

$$y = f(x)$$

$$y = g(x)$$

のように入力した場合には

\*2 SWP/SW によって自動的に文書に組み込まれるので、パッケージ追加（セクション 1.3 参照）の操作は必要ありません。

\*3 *amsmath* パッケージに依存しない `eqnarray` 環境での展開も不可能ではありません。操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0146.html>

等をご参照ください。

```

\begin{align}
y & =f(x) \\
y & =g(x)
\end{align}

```

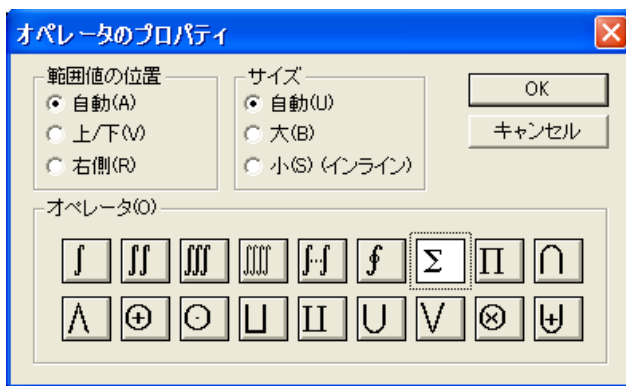
のようなコードに展開されます。ここに&は位置合せに用いられる記号です。この行間での位置合せの機能は大変重要な意味を持つのでこれについてはセクション 7.4 で説明します。

## 7.2 数式のプロパティ

前セクションではインライン数式とディスプレイ数式とで整形様式が異なる場合があると述べました。実際、インラインで  $\frac{b}{a}, \sum_{k=1}^n, \int_{-\infty}^{\infty}$  といったやや窮屈な数式や数式記号はディスプレイでは

$$\frac{b}{a}, \sum_{k=1}^n, \int_{-\infty}^{\infty}$$


といったゆったりした様式となります。しかしこれはあくまでデフォルト設定の場合の話です。例えばインライン数式  $\sum_{k=1}^n$  の右端にカーソルを位置付け右クリック、「プロパティ」と操作してみてください\*4。次のようなダイアログが表示されるはずですが。



“サイズ”や“範囲値の位置”がデフォルトでは“自動”となっているため、 $\text{T}_{\text{E}}\text{X}$  はテキスト行に適した形での整形を行うわけですが、複雑な数式の場合、判読がむずかしくなることもあります。そのような場合には適宜設定を変更してください。 $\sum_{k=1}^n$  がその一例ですが、インライン配置である以上、上下のテキスト行との間隔には留意する必要があります。

\*4 カーソルの位置が微妙に異なると添え字のプロパティが表示されてしまうこともあります。 $\sum_{k=1}^n$  という演算子全体をマウスカーソルで選択してプロパティを表示させた方が確実かも知れません。

### 7.3 数式番号

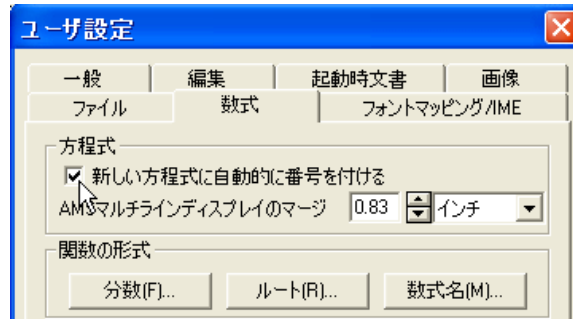
ディスプレイ数式を  ボタンによって設定すると行の右端に“#”のマークの入ったグレイボックスが表示されます。

$$y = f(x)$$

#

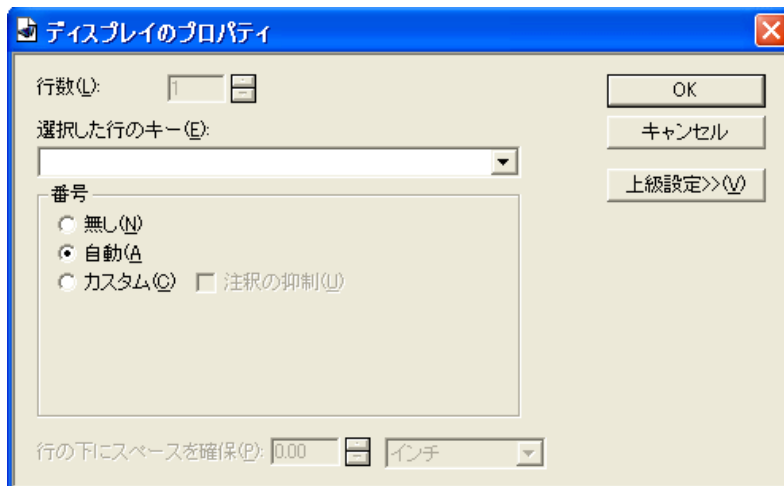



表示されない場合は「ツール」メニュー：「ユーザ設定」：「数式」タブと操作し、“新しい方程式に自動的に番号を付ける”にチェックマークを入れてください。




この数式番号に関する基本事項は次の通りです。

- 数式番号を付けないようにするにはグレイボックス右端にカーソルを位置付け右クリック、「プロパティ」と操作し、“番号 - なし”を選択します。



- “#”マークの数式番号については  $\text{T}_{\text{E}}\text{X}$  が自動的に番号を生成します。
- article 等の論文型文書の場合、番号様式の標準形は (n) のようになります。ここに  $n$  は 1 から始まる通し番号です。これに対し、book, report 等の書籍型文書の場合には、番号様式の標準形は (ch.n) のようになります。ここに  $ch$  は章番号を表します\*5。
- 「プロパティ」ダイアログ中で“番号 - カスタム”を選択した場合には任意の数式番号を設定できます。
- “キー”のフィールドに“eqn1”等の参照用キーを設定することができます。本文中から該当数式を参照する際、クロスリファレンスボタン  を使い、参照用キーを用いて参照するようにすれば、数式番号に変更が生じても参照側には影響が及ばなくなります。

複数行からなるディスプレイ数式の場合には各行に数式番号が設定されます。

$y = f(x)$	
$y = g(x)$	

この場合も行の右端にカーソルを位置付け右クリックすれば次ページのようなダイアログが表示されますが、単行の場合と異なり、“行数”指定のフィールドが有効になっている点、さらには複数行数式固有の制御項目が加わっている点に注意する必要があります\*6。

- 数式番号の表示/非表示等の設定は“行数”を指定した上で行います。
- 標準設定の場合、各行には独立した番号が設定されます。

$$y = f(x) \tag{1}$$

$$y = g(x) \tag{2}$$

しかし“サブの式番号設定”にチェックマークを入れるとサブ数式としての番号設定となります。またこの場合には、ディスプレイ数式全体に対して一つのキーを設定することが可能になります。

$$y = f(x) \tag{3a}$$

$$y = g(x) \tag{3b}$$

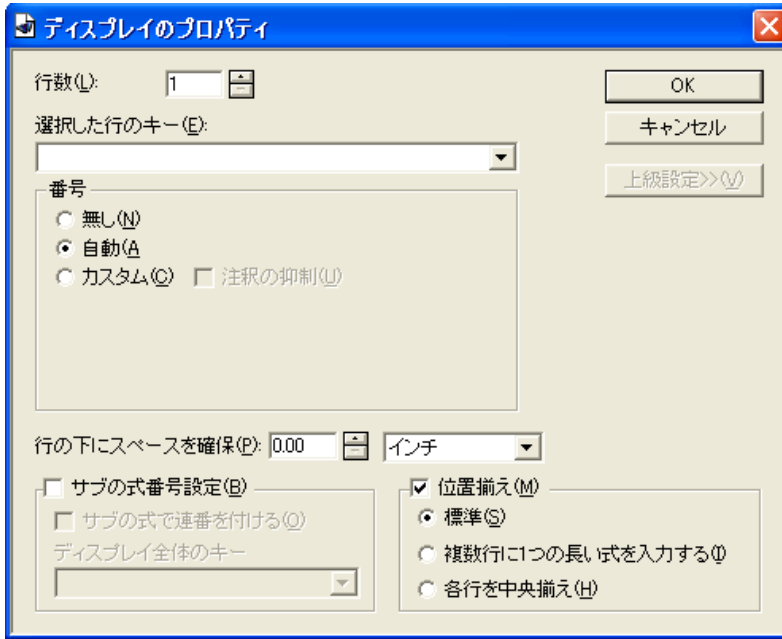
- 行と行の間のスペースは“行の下にスペースを確保”というフィールドを用いることによって制御できます。

\*5 論文型文書の場合、章というタグは設定できません。書籍型の数式番号様式（この場合はセクション番号との組合せ）を希望される場合、その操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0039.html>

をご参照ください。

\*6 “サブの式番号設定”と“位置揃え”の項が表示されていない場合には“上級設定”ボタンをクリックしてください。



## 7.4 位置揃え

複数行からなるディスプレイ数式の場合、次に示す例のようにデフォルトでは等号や不等号の位置で上下の位置揃え (alignment) が行われます。

$$\begin{aligned} y &= x^2 - 3x + 2 \\ &= (x - 1)(x - 2) \end{aligned}$$

しかしこれが常に最良の結果を生むとは限りません。

$$\begin{aligned} z &= x^2 + y^2 \\ \text{where } x &\geq 0, y \geq 0 \end{aligned}$$

複数行からなるディスプレイ数式の場合には「編集」メニュー内に「位置揃え」という項目が加わります。この機能を使用すれば任意の位置での位置合せが可能になります。

$$\begin{aligned} z &= x^2 + y^2 \\ \text{where } x &\geq 0, y \geq 0 \end{aligned}$$

この例の場合、1行目については  $x^2$  の直前に、2行目については where の直前に位置揃えのマークを設定しています。



伸縮自在な括弧ボタン `[O]` を用いて左に `[`、右に `]` を配置した上で複数行の数式を設定することは良く行われます。

$$\left\{ \begin{array}{l} y = f(x) \\ y = g(x) \end{array} \right.$$

また、数式番号を束ねるために逆向きの括弧を用いて次のような様式を設定することもあります。

$$\left. \begin{array}{l} y = f(x) \\ y = g(x) \end{array} \right\} \quad (4)$$

しかしこれらはいずれも `TeX` コード上は `align` 環境に展開されるわけではないので、複数行のディスプレイ数式とは呼べず、上で説明した位置揃え等の機能は使用できません。別の対応を図る必要があります。





## 8

## 数式 TeXnique [2]

本章では数式や数式記号の入力方法の他、ディスプレイ数式に関連した技法について解説します。



## 8.1 数式用ツールバー

数式入力に際しては多くのボタン操作が必要になります。「表示」メニュー：「ツールバー」と操作することによって各種ツールバーの表示/非表示を制御することができますが、次のツールバーは常時表示しておかれることを推奨します\*1。

- 数式テンプレート
- 数式オブジェクト
- 記号キャッシュ
- 記号パネル

## 8.2 数式入力

### (1) 添字

上付き添字 (superscripts)、下付き添字 (subscripts) はそれぞれ 、 ボタン使って入力します。なお  $\int_{-\infty}^{\infty}$  のように上付き添字、下付き添字双方を入力する場合には TAB キーを使うことによって入力位置をシフトさせることができます。あるいは一方の添字を入力後、→ キーを押して入力位置を中立的な位置に戻した後、もう一方の添字入力操作を行う形でも構いません。


\*1 それぞれのツールバーは外周部をドラッグすることによって移動できます。

一方、 ${}_nC_k$  のような prescripts を入力する場合には最初にゼロスペース\*2を入力した後に prescript を入力してください。添字の大きさ、位置は左側の文字によって規定されるため、あらぬ影響を排除すべくゼロスペースを入れるものです。

## (2) スペース

数式モード中では文字間の間隔を TeX が制御しているため、キーボード上の SPACE バーは利用できません（半角スペースの場合）。スペースを調整したい場合には「挿入」メニュー：「スペース」：「横スペース」と操作して適当なスペースを設定してください。詳しくはセクション 4.2 を参照ください。



## (3) ベクトル表記

ベクトルの表記方法として文字飾りボタン  を使用するのも一法ですが、 $\vec{a}$  や  $\vec{b}$  のように高さが不揃いになるのがやや難点です。一方、数式文字に“太字”というテキストタグを付加すると **a** や **b** のように直立体になってしまいます。*a* や *b* のように斜体のまま太字にするには“Bold Symbol”というテキストタグを使用してください。

## (4) ペア括弧

キーボードから括弧を入力した場合、中にどのような数式が入っても括弧の高さは伸び縮みするわけではありません。


$$\left( \frac{1}{\sqrt{x^2 + 1}} \right)$$

のように中の数式のサイズに合わせて伸縮する括弧を使いたい場合にはペア括弧ボタン  を使用してください。なお、片側の括弧として  を選択すれば

$$\left\{ \begin{array}{l} y = f(x) \\ y = g(x) \end{array} \right.$$

のような整形が行えます。

## (5) 関数名

数式モードで sin と入力するとそれが関数名であると認識され直立体で整形されます。しかし例えばベクトル解析で良く使用する *div* (divergence, 発散) や *rot* (rotation, 回転) といった演算子名を数式モードで入力すると、このように斜体で整形されてしまいます。これは SWP/SW の持っている関数名/演算子名の一覧の中にこれらの名前があらかじめ登録されていないことによるものです。これらを sin, cos のように直立体で出力したい場合には数式名ボタン  をクリックし、そこに関数名/演算子名を入力します。結果は *div* や *rot* のような形式となります。

\*2 「挿入」メニュー：「スペース」：「横スペース」と操作し“ゼロスペース”を選択します。



良く使用する関数名/演算子名については「ツール」メニュー：「自動置換」と操作し、キー入力とその出力形との対応関係を登録しておくとう便利です。すると例えば数式モードで  $div$  とキーインしただけで自動的に  $div$  に変換されるようになります。

### (6) 微分記号

$\frac{dx}{dt}$  や  $\frac{d^2x}{dt^2}$  といったライプニッツ記法の代りに  $\dot{x}$  や  $\ddot{x}$  のようなニュートン記法を用いることもできます。数式モードの状態では  $\text{T}_{\text{E}}\text{X}$  フィールドを設定し、その中にそれぞれ次のようなコマンドを入力してください。

```
\dot{x}
\ddot{x}
```

## 8.3 数式の拡大/縮小

「テキストタグ」フィールド中には“拡大 xxx%”とか“縮小 yy%”といったタグが用意されていますが、これらは数式モードでは使えません。特定の数式を拡大/縮小する方法についてはセクション 6.3 をご参照ください。

## 8.4 ディスプレイ数式


### (1) 改行による違い

ディスプレイ数式の前後に改行を置くか否かによって上下のスペースの量が多少異なりますので注意してください。

前後に改行あり	前後に改行なし
$y = f(x) \quad (1)$	$y = f(x) \quad (2)$

ディスプレイ数式の直後に改行を入れた場合には新たなパラグラフが続くものとみなされ、その先頭にインデントーション（字下げ）が入ります。これを修正するためには「挿入」メニュー：「スペース」：「横スペース」と操作し“インデントなし”という  $\text{T}_{\text{E}}\text{X}$  コードを設定するという操作が必要になります。改行を入れなければインデントーションは入りません。

## (2) 配置様式

ディスプレイ数式ボタン  を用いて設定されるディスプレイ数式はデフォルトでは中央揃えで整形されます。しかしクラスオプション“Displayed equations”で“Flush left”を選択した場合には左揃えの配置となります（クラスオプションについてはセクション 1.4 参照）。なお、プリアンプル\*3中に次のようなコマンドをセットしておくことで左端からの距離を調整することができます。この例で `2zw` というのは全角 2 文字分の意味です。

```
\setlength{\mathindent}{2zw}
```

## (3) 改ページの許可

複数行からなるディスプレイ数式の場合、それらは一つの塊りとして扱われるため、通常ページ境界をまたがって配置されることはありません。しかし `\allowdisplaybreaks` というコマンドを使用すると途中での改ページを許可することができるようになります。詳細は SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0081.html>

をご参照ください。

## (4) 位置揃え

複数行からなるディスプレイ数式の場合にはセクション 7.4 で説明したように、上下の数式間で位置合せをする機能が利用できます。これは数式が `amsmath` パッケージの `align` 環境に展開されることによるものです。しかし

$$\left. \begin{aligned} y &= x^3 + 3x^2 + 3x + 1 \\ &= (x + 1)^3 \end{aligned} \right\}$$

のようにペア括弧でくくったようなケースでは `array` 環境に展開されるため、等号位置での位置揃えといったサービスは受けられなくなります。このような場合の対処方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0092.html>

をご参照ください。

■



\*3 「タイプセット」メニュー：「プリアンプル」と操作します。

## 9

## 数式 TeXnique [3]

本章では数式番号に関連したトピックスについて解説します。

## 9.1 数式番号の様式

ディスプレイ数式ボタン  を使って数式を設定すると行の右端に  のマークが表示されます\*1。このマークの付いた数式に対しては T<sub>E</sub>X によって自動的に番号が振られますが、その様式の標準形は次のようになります。

- 論文型文書（ドキュメントクラスが `article` 型の文書）の場合には文書の先頭からの通し番号が  $(n)$  の形で生成されます ( $n = 1, 2, \dots$ )。
- 書籍型文書（ドキュメントクラスが `book`、または `report` 型の文書）の場合には、各章単位の番号が  $(ch.n)$  の形で生成されます。ここに  $ch$  は章番号を意味し、 $n$  は 1 から始まる整数です。

論文型文書の場合にも `numinsec` パッケージを使えば書籍型の様式に切替えられます。詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0039.html>

をご参照ください。

\*1 数式番号の自動生成機能を抑止したい場合には、「ツール」メニュー：「ユーザ設定」：「数式」タブで“新しい方程式に自動的に番号を付ける”のチェックマークを外してください。その場合、数式番号を付けないという状態がデフォルトとなります。数式番号が必要な箇所では数式右端を右クリック：「プロパティ」と操作することによって番号の設定が行えます。

## 9.2 数式番号の操作

### (1) 配置様式

デフォルトの場合、数式番号は数式右端に設定されます。しかしクラスオプション“Equation numbering”で“Numbers on left”を選択すれば数式番号の位置を左端に変更できます（クラスオプションについてはセクション 1.4 参照）。

### (2) 複数行からなる数式

複数行からなるディスプレイ数式に対する数式番号の付け方としては、

$$y = f(x) \tag{1}$$

$$y = g(x) \tag{2}$$

のようにそれぞれの数式に番号を付けることもできれば、

$$\begin{aligned} y &= x^3 + 3x^2 + 3x + 1 \\ &= (x + 1)^3 \end{aligned} \tag{3}$$

のように特定の行に対してだけ番号を振ることもできます。さらに

$$x(\theta) = \sin 3\theta \tag{4a}$$

$$y(\theta) = \sin 4\theta \tag{4b}$$

のような番号設定も可能です。具体的な操作方法についてはセクション 7.3 をご参照ください。これに対しペア括弧ボタン **[()]]** を使えば

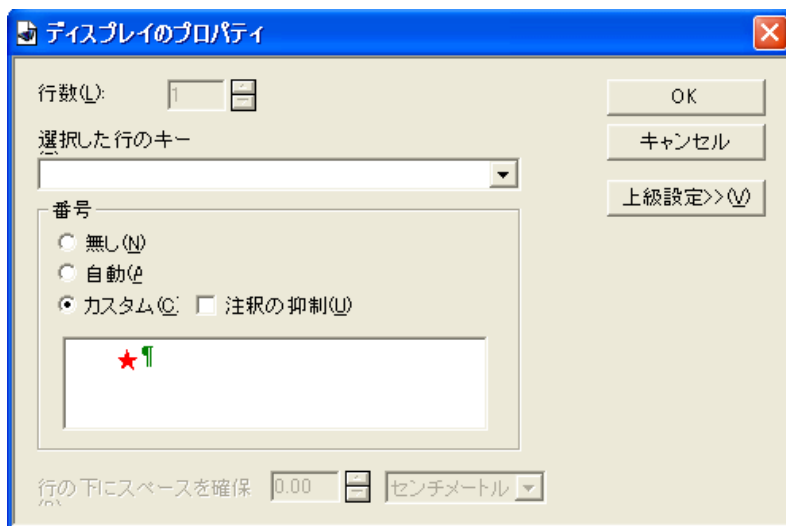
$$\left. \begin{aligned} y &= f(x) \\ y &= g(x) \end{aligned} \right\} \tag{5}$$

のように複数行を束ねた形での数式番号の設定も可能になります。

### (3) 独自番号の設定

「ディスプレイ数式のプロパティ」ダイアログ中で“カスタム”を選択すれば独自の番号を設定することができます。例えば次の例のように記号パネルから ★ のマークを選んだとすると次のような数式番号が生成されます。

$$y = f(x) \tag{★}$$



独自の番号/文字として全角文字を使用する場合には注意が必要です。例えば  
 ①、② のような文字を使用した場合、これらは機種依存となるため、dvipdfmx（日本語 PDF タイプセット用ドライバ）からの出力は空白となってしまいます。どうしても PDF 出力が欲しい場合には dviout（日本語 DVI タイプセット用ドライバ）の出力を Adobe Acrobat を使って PDF に変換してください。

#### (4) 長い数式の入力

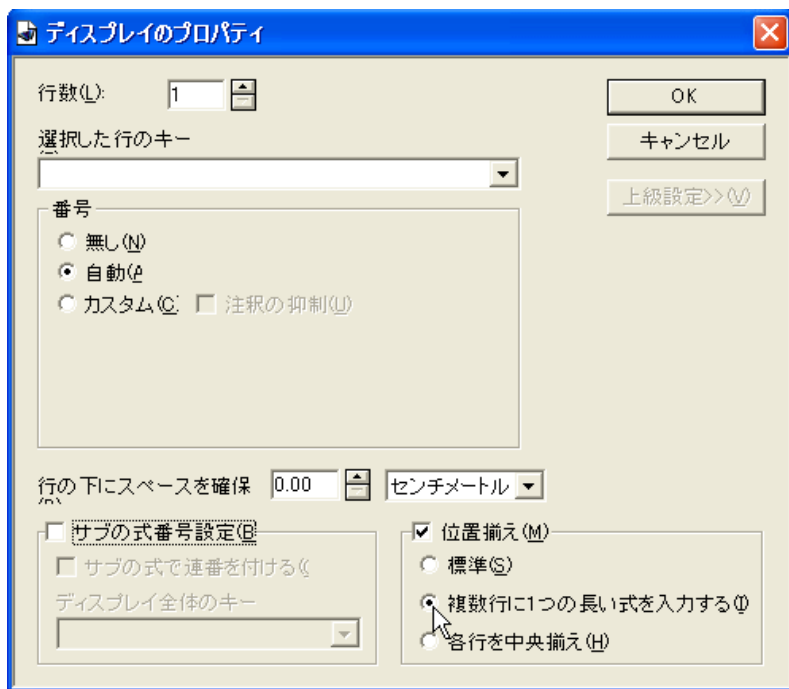
数式が長くて 1 行に収まらない場合、それを複数行からなる数式として構成することがあります。しかし何も指定しないと通常の複数行からなる数式として処理されるため、それぞれの行に数式番号が付加されてしまいます。

$$y = a1 + a2 + a3 + a4 + a5 + a6 + a7 + a8 + a9 + a10 + a11 + a12 \quad (6)$$

$$+ b1 + b2 + b3 + b4 + b5 + b6 + b7 + b8 + b9 + b10 + b11 + b12 \quad (7)$$

このような場合には「ディスプレイ数式のプロパティ」ダイアログ中で“複数行に一つの長い式を入力する”にチェックマークを入れてください\*2。

\*2 この選択項目が表示されていない場合には「上級設定」ボタンをクリックしてください。



これによって次のような整形結果を得ることができます。数式番号は最後の行にのみ付加される形となります。

$$y = a1 + a2 + a3 + a4 + a5 + a6 + a7 + a8 + a9 + a10 + a11 + a12$$

$$+ b1 + b2 + b3 + b4 + b5 + b6 + b7 + b8 + b9 + b10 + b11 + b12 \quad (8)$$

■

## 10

## 数式 TeXnique [4]

本章では定理型環境に関連する各種トピックについて解説します。

## 10.1 定理型環境の定義

本文中に“定義”とか“定理”といった項目を設定するには「アイテム」タグフィールド中から該当する項目を選択すれば良いわけです。その上で“定義”とか“定理”の記述を行えば所定のフォント、様式でタイプセットされます（欧文書の場合には基本的に斜体フォントが使用されます）。標準設定のラベリングとか番号付けがニーズに合致する場合はこれ以上特に深入りする必要はありませんが、多少なりカスタマイズしたいという場合には  $\text{T}_{\text{E}}\text{X}$  の仕掛けに対する習熟が必要となります。基本となるのは「プリアンプル」\*<sup>1</sup>中に設定されている次のコードです（`jsarticle` の場合の例）。

```
\newtheorem{theorem}{ 定理 }
\newtheorem{acknowledgement}[theorem]{ 承認 }
\newtheorem{algorithm}[theorem]{ アルゴリズム }
\newtheorem{axiom}[theorem]{ 公理 }
\newtheorem{case}[theorem]{ 場合 }
\newtheorem{claim}[theorem]{ 主張 }
\newtheorem{conclusion}[theorem]{ 結論 }
\newtheorem{condition}[theorem]{ 条件 }
\newtheorem{conjecture}[theorem]{ 推論 }
\newtheorem{corollary}[theorem]{ 系 }
```

\*<sup>1</sup> 「タイプセット」メニュー：「プリアンプル」と操作します。

```

\newtheorem{criterion}[theorem]{基準}
\newtheorem{definition}[theorem]{定義}
\newtheorem{example}[theorem]{例}
\newtheorem{exercise}[theorem]{練習}
\newtheorem{lemma}[theorem]{補題}
\newtheorem{notation}[theorem]{記号}
\newtheorem{problem}[theorem]{問題}
\newtheorem{proposition}[theorem]{命題}
\newtheorem{remark}[theorem]{記事}
\newtheorem{solution}[theorem]{解法}
\newtheorem{summary}[theorem]{要約}
\newenvironment{proof}[1][証明]{\noindent\textbf{#1.}}{\rule{0.5em}{0.5em}}

```

## 10.2 ラベル

アイテムタグにより定理型環境を設定すると、`\newtheorem` 文の第 3 引数<sup>\*2</sup>で指定されたラベル（青字）が文書上に表示されます。文言を変更したい場合にはプリアンブル中のこれらラベル引数を適宜変更してください。

## 10.3 番号付け

上記 `\newtheorem` 文の第 2 引数<sup>\*3</sup>（赤字で記した部分）はみな、先頭の `\newtheorem` 文で定義された環境名“theorem”を参照しています。これは定理型環境の番号付けに際し、定理型環境全体で統一的な番号が設定されることを意味しています。例えば定義、定理、補題の順に環境を設定したとすると

定義 1  
定理 2  
補題 3

のように番号が振られることとなります。この第 2 引数 `[theorem]` をすべて除去してしまうと

定義 1  
定理 1  
補題 1

のように各環境ごとに番号が設定される形となります。

<sup>\*2</sup> 先頭の `\newtheorem` 文の場合には第 2 引数。

<sup>\*3</sup> 先頭の `\newtheorem` 文の場合には存在しません。

なお、この `\newtheorem` 文には

```
\newtheorem{env_name}{label}[within]
```

という構文と

```
\newtheorem{env_name}[numbered_like]{label}
```

という構文の 2 種類があります。jsarticle 用シエルの例で言えば先頭の文のみがこの最初の構文に該当するわけですが、その第 3 引数 `within` で `chapter` と指定した場合には番号が `ch.n` のように章番号と通し番号の組合せの形となります。この場合、通し番号 `n` は章が変わる都度初期値にリセットされます。論文型文書の場合には `section` と指定することにより同様の効果を期待することができます。



定理型環境の中でもマーカーを設定することができるので、番号参照をクロスリファレンスの形で行わせることが可能です。操作法についてはセクション 3.2 を参照ください。

## 10.4 定理型環境のカスタマイズ

以上述べた範囲を越えて定理型環境をカスタマイズしたいというケースがいくつか考えられます。

### (1) 番号の除去

フェルマーの定理等、普遍的な定理型環境を記述する際には番号を付けたくないといったニーズが考えられます。`\newtheorem` コマンドで独自の環境を定義することによって対応することができますが、具体的な操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0078.html>

をご参照ください。

### (2) 番号様式の変更

定理型環境の番号に A, B, C 等のアルファベットや I, II, III 等のローマ数字を使用したいといったニーズも考えられます。これは番号様式を `\renewcommand` で規定することによって対処可能ですが、具体的な操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0097.html>

をご参照ください。

### (3) フォントの変更等

欧文書の場合、定理型環境の記述には一般的にイタリック体が用いられますが、学会や出版社の要請等により直立体に変更する必要があることもあります。また定理とか定義といった見出しの直後に改行を置き、本文を別個の行から開始するといったスタイルも考えられます。このような多様なニーズにきめ細かく対応できるのが *theorem* パッケージです。その用法については *Typesetting Documents* マニュアルをご参照ください。




## 11

## 表 TeXnique [1]

表を  $\text{T}_\text{E}\text{X}$  のコードで直接記述するのは容易なことではありません。しかし SWP/SW を使えば簡単に作成できます。また画像や数式の配置を整える上でも便利な機能を提供します。

## 11.1 表の種別

SWP/SW で表を作成するには 2 通りの方法があります。一つは表ボタン  を利用する方法です。インライン配置として横方向の位置をカーソルで制御することもできますが、「セクション/ボディタグ」フィールド内の「中央揃え」タグを設定すれば行の中央に配置することもできます。いずれにせよこのボタンを用いて生成される表の配置はユーザ自らが行うこととなります（後述する“浮動型の表”との対応から“**固定型の表**”と呼ぶことにします）。いま表ボタンを使って右下のような表を作成したとすると、 $\text{T}_\text{E}\text{X}$  コード上は次のような `tabular` 環境に展開されることになります。

```
\begin{tabular}
[c]{|l|l|l|}\hline
\textbf{Head} & \textbf{Head} & \textbf{Head}\\\hline
entry & entry & entry\\
entry & entry & entry\\
entry & entry & entry\\\hline
\end{tabular}
```

Head	Head	Head
entry	entry	entry
entry	entry	entry
entry	entry	entry

これに対し「フラグメント」フィールド内の「4 行 3 列の表 (LaTeX 用)」と書かれたフラグメントを用いて表を作成することもできます。

Head	Head	Head
entry	entry	entry
entry	entry	entry
entry	entry	entry

[B] caption marker: TableKey [E]

画面に表示される表の部分は自由に変更できるので“4 行 3 列”という表記には意味はありませんが、ここではそのまま使用したとすると次のような TeX コードが生成されます。

```
\begin{table}[tbp] \centering
\begin{tabular}
[c]{|l|l|l|}\hline
\textbf{Head} & \textbf{Head} & \textbf{Head}\\\hline
entry & entry & entry\\
entry & entry & entry\\
entry & entry & entry\\\hline
\end{tabular}
\caption{Table Caption}\label{TableKey}%
\end{table}%
```

`\begin{tabular}` から `\end{tabular}` までの部分は表ボタンを使ったときと何ら変わりませんが、その前後に

- `\begin{table}`
- `\caption`
- `\label`
- `\end{table}`

等の TeX コマンドが追加されているわけです。これらが表の前後に付加されたグレイボックスに対応しています。実際、各グレイボックスの右端にカーソルを置き、右クリック、「プロパティ」と操作すればこれらの TeX コマンドが確認できるはずですが、単なる `tabular` 環境ではなく `table` 環境として表を設定した場合、オブジェクトの配置や番号付けは TeX に一任され、表は `floatable object` (“[浮動型の表](#)”)としての扱いを受けることになります。

## 11.2 配置の制御

固定型の表の場合、配置に関し考慮すべきことはほとんどありません。カーソルで指定された位置に表が生成され、タイプセットの過程でその位置が変わることもありません。しかし浮動型の場合には、その名が示すように配置は  $\text{T}_\text{E}_\text{X}$  に委ねられます。文書のレイアウトを考慮する形で  $\text{T}_\text{E}_\text{X}$  が配置場所を決定します。しかしその配置に関し何も注文が付けられないというわけではありません。Preference を次のコードで指定することができます。

h	その位置 (here) に表を配置。
t	ページの上端 (top) に表を配置。
b	ページの下端 (bottom) に表を配置。
p	独立したページ (page) に表を配置。

[B] と表示された  $\text{T}_\text{E}_\text{X}$  フィールド\*<sup>1</sup>中には次のような  $\text{T}_\text{E}_\text{X}$  コードがデフォルトで設定されています。

```
\begin{table}[tbp] \centering
```

この `tbp` の部分が preference を示しています (第 1 志望が `t` で第 3 志望が `p` です)。浮動型の表を使用する場合にはこの設定を適宜編集してください。



これはあくまで preference であって 100% 希望通りになるという保証はありません。しかし *float* パッケージを使用すれば希望通りの配置が可能になります。詳しくは Typesetting Documents マニュアルをご参照ください。

## 11.3 キャプションと表番号

固定型の表の場合、キャプションや表番号の設定はユーザーが行う形となります。右の図はその一例です。表の実体は上の 3 行のみですが、空白行とキャプション用の行をその下に付加してあります。なお、キャプション

A	B	C
aaaaaa	bbbbbb	cccccc
aaaaaa	bbbbbb	cccccc
□	□	□
表1 表のキャプション		

\*<sup>1</sup> 操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0009.html>  
をご参照ください。

行についてはセルの結合操作を行っている点に注意してください\*2。タイプセットすると次のようになります。

A	B	C
aaaaaa	bbbbbb	cccccc
aaaaaa	bbbbbb	cccccc

表 1 表のキャプション

これに対し、浮動型の表の場合には表番号の付加は T<sub>E</sub>X によって自動的に行われます。このためキャプション中に表番号を明示する必要はありません。キャプション自体の設定は `caption` と表示された T<sub>E</sub>X フィールドを編集する形で行います。以下は浮動型の表の設定例です。

A	B	C
aaaaaa	bbbbbb	cccccc
aaaaaa	bbbbbb	cccccc
□	□	□

[E] caption marker: tbl1 [E]

ちなみに `caption` という名の T<sub>E</sub>X フィールド中には次のコードを設定してあります。

```
\caption{ 表のキャプション }
```

この状態でタイプセットすると次のような結果が得られます。

A	B	C
aaaaaa	bbbbbb	cccccc
aaaaaa	bbbbbb	cccccc

表 11.1 表のキャプション



`caption` と表示された T<sub>E</sub>X フィールドを表より前に移動するとキャプションの設定位置は表の上部となります。

\*2 キャプション行の 3 セルをマウスカーソルで選択した状態で右クリック、「セルの結合」と操作します。




本ガイドが書籍型クラスファイルを使用しているため、自動生成されるオブジェクト番号は章番号との組合せとなっています。論文型の場合は表 1 となります。

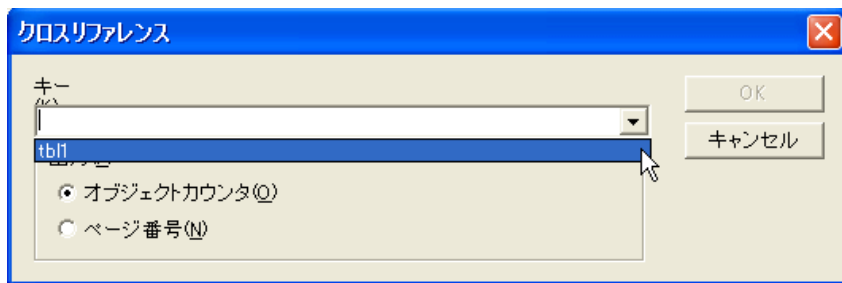
ここで固定型の表に付けた番号との重複が生じている点に注意してください。これは固定型の表がオブジェクト管理の対象外であることによるものです。このような場合には  $\text{T}_{\text{E}}\text{X}$  が管理する番号を調整してやる必要があります。詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0140.html>

をご参照ください。

## 11.4 参照用キー

固定型の表の場合、表番号の設定はユーザ自身によって行われるため、表を本文中から参照する際もその番号を直接指定することになります。これに対し浮動型の場合には、番号付けはタイプセットの過程で  $\text{T}_{\text{E}}\text{X}$  によって行われます。従って本文中から該当オブジェクトを参照する際にはキーを使った間接的な方法を用いるのが一般的です。例えばセクション 11.3 の例では `marker:` と表示されたグレイボックス内に `tbl1` というキーを設定してあります。一方、この表を参照する側ではこのキーを指定するわけです。例えば“表 □ 参照”という文字列中に表番号を入れたい場合には、□の部分でクロスリファレンスボタン  をクリックします。次のようなダイアログが表示されるので一覧の中から該当するキーを選択してください。



□の部分には次のような  $\text{T}_{\text{E}}\text{X}$  コードが設定され、タイプセットを通すとそれが実際の表番号で置き換えられることとなります。

```
\ref{tbl1}
```




## 12

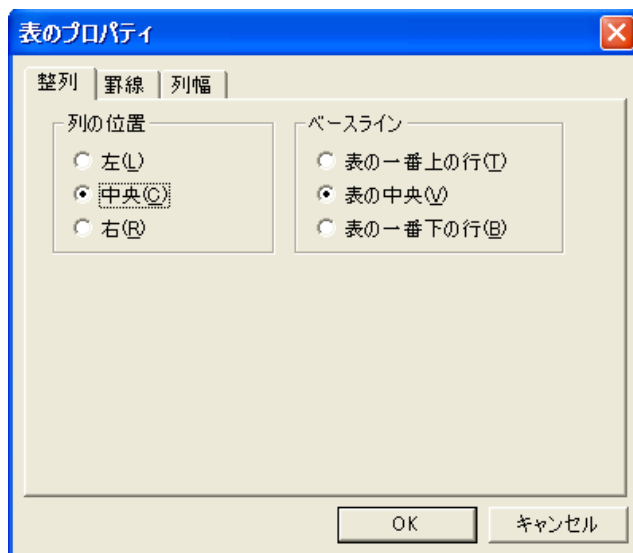
## 表 TeXnique [2]

SWP/SW の場合、表の属性設定がダイアログを介して簡便に行えますが、中には少々慣れを要する項目もありますので説明を補足しておきます。

## 12.1 表のプロパティ

第 11 章で説明したように、表はテーブルボタン 、あるいはテーブルフラグメントを用いて作成できます。生成された表の全体、あるいは特定のセルのみを選択した上で右クリック、「プロパティ」と操作すると右のようなプロパティダイアログが表示されます。このダイアログを用いることによって次のような属性が設定できます。

- 整列 (アラインメント)
- 罫線
- 列幅



## 12.2 アラインメント

アラインメントに関する項目としては次の 2 種類があります。

- 列の位置  
選択されたセルの中に文字列や画像等のオブジェクトをどう配置するかを規定します。左寄せ、中央揃え、右寄せの中から選択できます。
- ベースライン  
これは表全体の配置に関わる項目です。テキスト行のベースラインに対する相対位置を規定します。

## 12.3 罫線

罫線を引くには次のように操作してください。

- 罫線を引く対象のセル（一つ、または複数）をマウスカーソルで選択する。
- その状態で右クリック：「プロパティ」と操作し「罫線」タブを開く（右図参照）。
- 線種を選択後、それを引く位置を選択する。



## 12.4 列幅

デフォルトの場合、それぞれの列の幅はセル中に収容すべきオブジェクト（文字列や画像）のうち最長の横幅を持つものによって規定されます。いくら長い文字列であっても改行されることはありません。

A	B	C
aa	bbbbbbbbbbbbbb	cccc

これに対し「列幅」タブを用いセル幅を指定すると文字列はその幅で折り返されるようになります。右はその一例ですが、SWP/SW 操作画面上、列幅を指定したセルはグレイボックスとしての表示となります。中にテキストが埋め込まれているわけですが、その内容を確認するにはグレイボックス右端にカーソルを位置付け右クリック：「プロパティ」と操作する必要があります。タイプセットを行うと次のような表が整形されて出力されます。

2cm	8cm
2cm	8cm
2cm	8cm

EPS	Encapsulated PostScript の略。印刷業界で広く使用されている画像形式。PostScript のサブセット。ベクトル画像。
GIF	Web ブラウザ環境で広く普及している画像形式。扱えるカラーの数が 256 色までという制約がある。ラスタ画像。Unisys が特許ライセンス料を主張していた時期がある。
PNG	GIF の問題点を解消した画像形式。ラスタ画像としては最も品質に優れる。ただし Web ブラウザの中には未対応のものもある。



列幅タブを開くには **2cm** のようにマウスカーソルでセルを選択した状態で右クリック：「プロパティ」と操作してください。**2cm** のように灰色の状態で操作するとセルの内容が表示される結果となります。

なお、列幅を指定した場合にはアラインメントの指定が無効になってしまう点に注意してください。列幅を指定したセルは左寄せで整形されることとなります。中央揃え（あるいは右寄せ）と列幅指定を両立させるには右のようなアプローチが考えられます。すなわち追加の行を設け、そこに列幅指定のセルを置く方法です。整形結果は次のようになります。

<b>A</b>	<b>B</b>	<b>C</b>
aaaa	bbbb	cccc
2cm	2cm	2cm

<b>A</b>	<b>B</b>	<b>C</b>
aaaa	bbbb	cccc



表として 1 行余分なスペースを必要とすることから、後続パラグラフとの間隙が気になる方もおられるかも知れません。しかし後続パラグラフの先頭に TeX フィールドを設定し、

```
\vspace{-1.0\baselineskip}
```

というコマンドを入れてやれば 1 行分戻すことができます。

上に紹介した手法は追加の行を設けて中央揃えと列幅指定を両立させたわけですが、より直接的な解決法もあります。

右の図は列幅指定のセル 3 個からなる表であり、中には **A**、**B**、**C** の文字が入っています。このままでは左寄せとなって

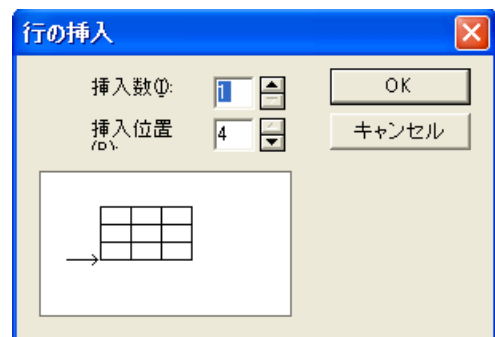
2cm	2cm	2cm
-----	-----	-----

しまうわけですが、それぞれの文字の両端に伸縮性のある横スペース（目に見えないバネのようなもの）を挿入することによって中央揃えが行えます。具体的には「挿入」メニュー：「スペース」：「横スペース」と操作し「カスタム」を選択します。その上で、「サイズ」フィールド中で「端まで伸ばす」を選択、さらに「タイプセット」フィールドで「固定」を選択してください\*1。これによって次のような表が生成されます。

A	B	C
---	---	---

## 12.5 行や列の追加

既に作成済みの表に対して行や列を追加することもできます。表の右端にカーソルを位置付け右クリック、表示されるコンテキストメニューの中から「行の挿入」、あるいは「列の挿入」を選択してください。この操作の結果右のようなダイアログが表示されるので、挿入位置と挿入行数、あるいは列数を指定してください。



\*1 「固定」を選択しないと文字列の右側に置かれたスペースが無視されてしまいます。

## 12.6 セルの結合

併合したいセルをマウスマウスカーソルで選択し右クリックすると、表示されるコンテキストメニュー中に「**セルの結合**」という項目があることに気が付かれると思います。これによってセルの併合が可能になります。ただしサポートされているのは横方向の併合（列の併合）のみである点にご注意ください。縦方向の併合（行の併合）はサポートされていません。





## 13

## 画像 TeXnique [1]

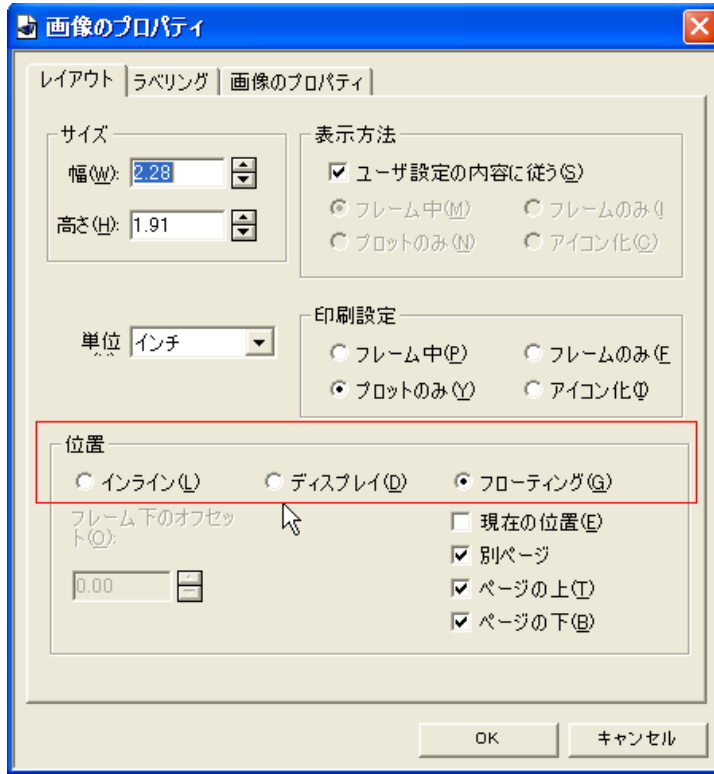
SWP/SW 環境で画像を扱う場合、他のアプリケーションで作成された画像をインポートするケースがほとんどです（50 種類を越すの画像形式に対応）。本章では画像データの扱いに関する基本事項について説明します。

### 13.1 画像の配置

SWP/SW 文書上に画像をインポートする場合、その配置様式によって扱いが大きく異なって来るので注意が必要です。画像を右クリック：「プロパティ」と操作することによって開かれる“画像のプロパティ”ダイアログ（次ページの図参照）中で配置様式を規定することになりますが、選択肢としては

- インライン
- ディスプレイ
- フローティング

の 3 種類があります。



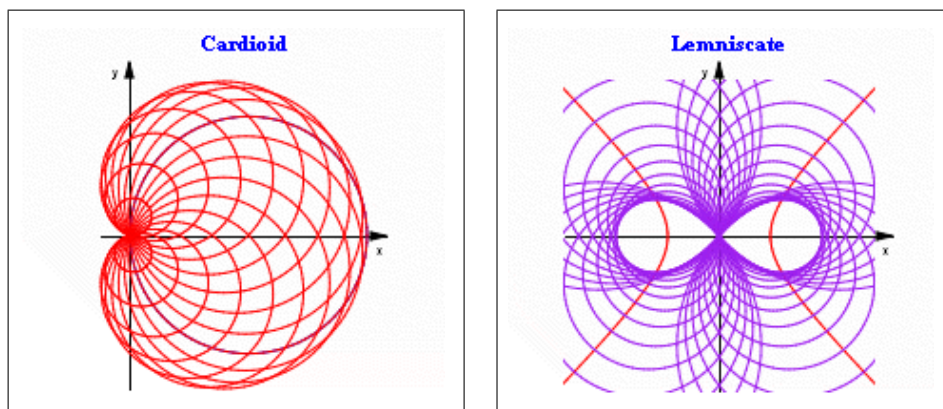
## (1) ディスプレイ画像

最も一般的な画像の配置形態であり、次のような特質を持ちます。

- 一つの画像で単独の行を占有する（数式の場合のディスプレイ数式と同じ扱い）。
- 画像は行の中央部に配置される。
- 図番号は自動的に振られないのでキャプション中で明示する必要がある。

## (2) インライン画像

ディスプレイ画像の場合と異なり、行内での位置を自分でコントロールできます。またテキスト行のベースラインとの位置関係を調整するための“フレーム下のオフセット”という制御項目が有効となります。複数個の画像を横に並べる場合には表中に画像を配置する方法もあります。この場合も画像の配置としてはインライン配置となります。次は1行2列の表（罫線なし）の中に2つの画像をフレーム付きで配置した例です。



図番号についてはディスプレイ画像の場合と同様、自分で設定する必要があります。

### (3) フローティング画像

次はディスプレイ画像とフローティング画像とで生成される  $\text{T}_\text{E}\text{X}$  コードにどのような違いが生ずるかを示したものです。

ディスプレイ画像	フローティング画像
<pre> \begin{center} \includegraphics キャプション文字列 \label \end{center} </pre>	<pre> \begin{figure} \begin{center} \includegraphics \caption \label \end{center} \end{figure} </pre>


基本的な違いはフローティング画像の場合、`\includegraphics` コマンドが `figure` 環境内に配置されるという点にあります。これは浮動型の表が `table` 環境内に配置されるのと同様の話ではあるわけですが、これによって対象オブジェクトが  $\text{T}_\text{E}\text{X}$  管理下におかれるようになります。その結果、フローティング画像には次のような特質が加わるようになります。

- 配置はその名のとおり浮動型となる。ただしユーザは次のような preference を設定可。

h	その位置 (here) に画像を配置。
t	ページの上端 (top) に画像を配置。
b	ページの下端 (bottom) に画像を配置。
p	独立したページ (page) に画像を配置。



これはあくまで preference であって 100% 希望通りになるという保証はありません。しかし *float* パッケージを使用すれば希望通りの配置が可能になります。詳しくは Typesetting Documents マニュアルをご参照ください。

- 図番号は  $\TeX$  によって自動生成されるためキャプション中では番号の設定は不要。
- 画像に対しては参照用のキーを設定する。本文中からの参照に際してクロスリファレンスボタン  を使用すれば、番号の解決は  $\TeX$  によって自動的に行われる。

## 13.2 p $\LaTeX$ 特記事項

日本語文書中で画像を扱う場合には、画像形式によって次のような追加モジュールの組込みが必要になります。

追加モジュール	対応画像形式
Ghostscript	EPS
Susie plug-in	JPEG, GIF, TIFF, PNG, 他

インストール方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0015.html>

をご参照ください。



## 14

## 画像 TeXnique [2]

画像や表が比較的小さなものである場合、テキストパラグラフをその周囲で折り返せたらと考えるのは自然でしょう。本章ではパラグラフの折返しに関する TeX 流のアプローチについて説明します。

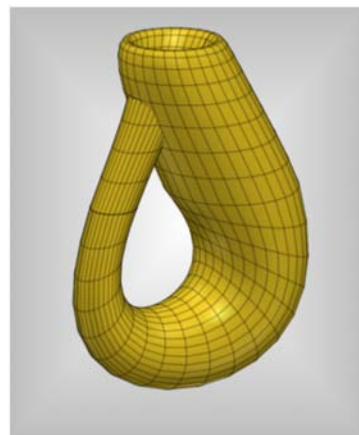
### 14.1 *wrapfig* パッケージ

SWP/SW を使って文書を作成する場合でも、このパラグラフがそうであるように、画像や表の周囲でテキスト行を折り返すことが可能です。そのためには *wrapfig* パッケージを使用します。具体的な操作手順は次のようになります。

1. *wrapfig* パッケージを文書に追加します\*1。
2. 折返しを行いたい位置にカプセル化した TeX フィールド\*2を設定し、次のいずれかのコマンドを入力します。

```
\begin{wrapfigure}[10]{r}{0pt}
```


```
\begin{wraptable}[10]{r}{0pt}
```



\*1 「タイプセット」メニュー：「オプションとパッケージ」：「パッケージオプション」タブと操作し「追加」ボタンをクリックします。

\*2 用法については SWP/SW BetterUse ページ

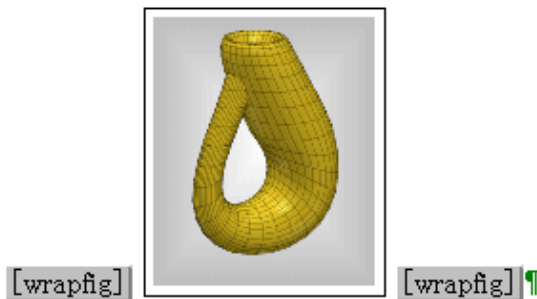
<http://www.lightstone.co.jp/latex/kb0009.html>  
をご参照ください。

- 画像を配置する場合には `wrapfigure` を、表を配置する場合には `wraptable` を使用します。
  - 最初のパラメータでは折返しの対象となる行数を指定します。画像や表のサイズに応じて調整してください。
  - 次のパラメータでは画像や表の位置を指定します。右に配置する場合には `r` を、左に配置する場合には `l` を指定してください。
  - 最後のパラメータでは画像や表の横幅を指定します。この値が 0 の場合には画像や表の実寸が用いられます。
3. 画像の場合には「ファイル」メニュー：「画像のインポート」と操作し、画像をインポートします。その際、配置としては“インライン”を選択します。  
表の場合には  ボタンを使って表を作成してください。
4. 画像、または表の右側に再度  $\text{T}_{\text{E}}\text{X}$  フィールドを設定し、画像、あるいは表に応じて次のコマンドのいずれかを入力します。

```
\end{wrapfigure}
```

```
\end{wraptable}
```

以上の操作の結果、SWP/SW の編集画面上は次のような状態となったでしょうか。



この設定に続けてテキストパラグラフを入力して行くと、指定された行数におさまる分の文字列が折り返された形で整形されるようになります。行数の指定値が適正でなかった場合には適宜調整してください。なお、`wrapfig` パッケージの用法に関する詳細については *Typesetting Documents* マニュアルをご参照ください。

## 14.2 図表番号の調整

`wrapfig` パッケージを使用した場合、画像や表はインラインオブジェクトとしての扱いとなります。このため図番号や表番号の設定はユーザ側で行わなくてはなりません。例えば `wrapfig` パッケージで設定したオブジェクトに対し  $m$  番という番号を付けたとしましょう。このとき次にくるフローティングオブジェクトに対し  $\text{T}_{\text{E}}\text{X}$  が  $m + 1$  番という番号を振ってくれば良いのですが、残念ながらそうはならず、 $m$  番という番号が重複してアサインされることとなります。インラインオブジェクトに対し  $m$  番という番号が使われたことを  $\text{T}_{\text{E}}\text{X}$  は知らないからです。

このため `wrapfig` パッケージで設定したオブジェクトに対し番号を付けた場合には、 $\text{T}_{\text{E}}\text{X}$  が管理する番号を調整してやる必要があります。次にフローティングオブジェクトを使用する前に  $\text{T}_{\text{E}}\text{X}$  フィールドを設定し、次のようなコマンドを入力してください。

```
\setcounter{figure}{m}    画像の場合
```

```
\setcounter{table}{m}    表の場合
```

この設定により次に生成されるフローティングオブジェクトに対しては  $m + 1$  番という番号がアサインされるようになります。





## 15

## 段組み

本章では論文や書籍を多段組み構成で作成する場合の TeXnique について解説します。段組み部分には取まらない図表や数式の扱いについても記述しています。

## 15.1 クラスオプション

SWP/SW で 2 段組み文書を作成する場合には Columns クラスオプションで“Two columns”を選択するのが一般的です（クラスオプションについてはセクション 1.4 をご参照ください）。どのような様式になるかは標準シェルでこのオプションを指定しタイプセットしてみるのが早道でしょう。論文型文書、書籍型文書、各々の場合につき特記事項を記しておきます。特に日本語文書の場合、`jsclass` でさまざまな改良が加わっている点は注目に値します。

### (1) 論文型文書

論文型文書の場合、表題部は段組み領域の外側に配置されますが、**概要部**(abstract) は段組み領域内に出力されてしまいます（図 15-1 参照）。`article`、`jarticle` の場合調整は効きませんが、`jsarticle` であれば Make Title アイテムと Abstract アイテムの順番を変えるだけで、概要部を段組み領域外に出力させることが可能です。実際、“Standard LaTeX”シェルフォルダ内にある

- `[jsarticle] twocolumn-type2`

というシェルではあらかじめそのような設定で文書が用意されています。

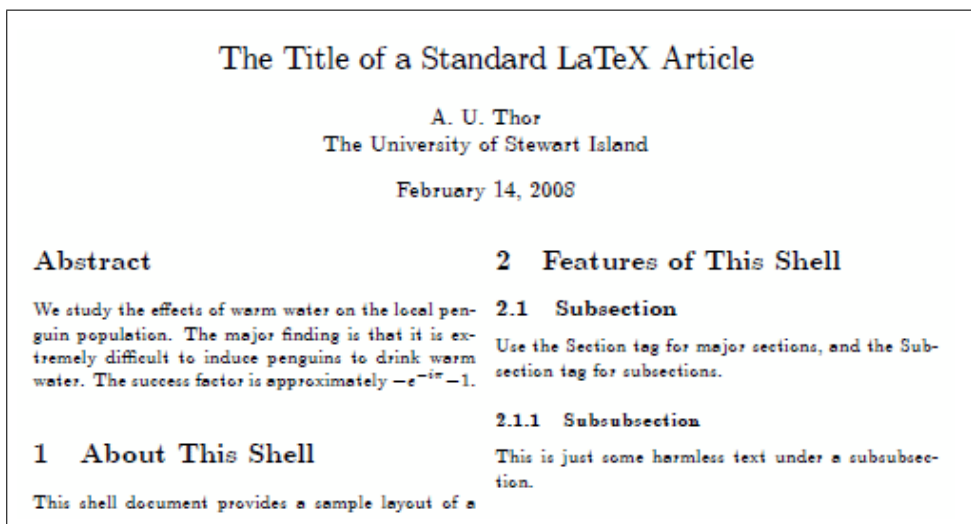


図 15-1 2 段組み構成の論文

## (2) 書籍型文書

- 表紙、目次

表紙と目次は段組みの対象にはなりません。この点は book, jbook, jsbook 間で差はありません。

- 章見出し

book, jsbook の場合、章見出しは段組み領域の外側に設定されますが、jbook の場合、章見出しも段組みの対象として整形されてしまうので注意してください (図 15-2 参照)。

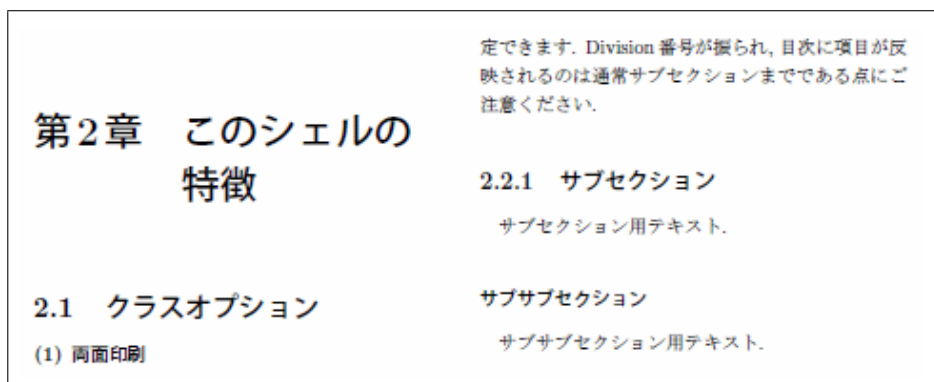


図 15-2 jbook の場合の章見出し

- 索引

索引はクラスオプションの指定に関係なく、常に 2 段組みで生成されます。なお、*jsbook* の場合にはカラム間に縦の罫線が引かれる点が他と異なります。



シェル文書をタイプセットしてみるとわかると思いますが、`verbatim` 環境を使用した部分がカラム幅を超えています。これは `verbatim` 環境内では  $\text{T}_{\text{E}}\text{X}$  が勝手に改行を入れられないことによるものです。`verbatim` 環境を使って  $\text{T}_{\text{E}}\text{X}$  コードの記述を行う場合には注意してください。

## 15.2 *multicol* パッケージ

クラスオプションで 2 段組みを指定するのではなく、*multicol* パッケージを利用する方法もあります。*multicol* パッケージを使用した場合には、一つの文書中で段組み構成を取る部分と取らない部分とを自由に混在させることが可能になります。さらに最終ページにおけるカラム長の均等化とか、3 段以上の段組み構成への対応等の機能もあり、付加価値の高いパッケージと言えます。



*multicol* パッケージを使用する場合、 $\text{T}_{\text{E}}\text{X}$  の `figure` 環境や `table` 環境に展開される浮動型の図表は段組み部分では使用できないので注意してください。後述する `figure*` 環境や `table*` 環境を使用するか、またはディスプレイ配置、インライン配置を使用する必要があります。

### 15.3 figure\*環境/table\*環境

段組み構成を取る場合に悩ましいのは、大きな図表や長い数式に対する許容量が 1 段組みの場合に比べて落ちるとい点です。テキスト領域の幅が半分以下に落ちるわけですからこれは当然なのですが、その場合の対処法としてはオブジェクトの縮小を図るとか、数式であれば複数行に分割する等、種々のアプローチが考えられます。そのうちの一つが figure\*環境/table\*環境を利用する方法です。通常の figure 環境/table 環境内に配置されたオブジェクト（いわゆるフロータブル（浮動型）オブジェクト）は段組みの**カラム**内に配置されるわけですが、figure\*環境/table\*環境の場合には**段組み領域外**に配置されます。このためカラムの幅を超えた図表や数式であっても、特に修正を施すことなくそのまま配置することが可能となります。

#### (1) 画像の配置

画像の場合には figure\*環境を使用するわけですが、それ用のフラグメントがあらかじめ用意されているわけではないので、“4 行 3 列の表”フラグメント

	Head	Head	Head	
[B]	entry	entry	entry	caption marker: TableKey [E]
	entry	entry	entry	
	entry	entry	entry	

を若干加工する形で対応します（このフラグメントの基本的な用法については第 11 章を参照してください）。

- [B] と [E] と書かれた TeX フィールド中にはそれぞれ

```
\begin{table}, \end{table}
```

というコマンドがセットされているので、それらを

```
\begin{figure*}, \end{figure*}
```

に変更します。

- 次に表の部分削除し、代わりにその部分に画像をインポートします。その際、画像は“インライン”指定で配置することが肝要です\*1。
- 画像に対するキャプションや参照用キーはそれぞれ caption, marker と書かれたグレイボックスを編集する形で情報を入力します。

\*1 「画像のプロパティ」ダイアログ中の「レイアウト」タブをクリック、「インライン」を選択してください。

- figure\*オブジェクトの配置に関する preference は [B] と書かれた T<sub>E</sub>X フィールド中の [tbp] という項を編集してください。

## (2) 表の配置

表の場合、“4行3列の表”フラグメント

[B]	Head	Head	Head	
	entry	entry	entry	caption marker: TableKey [E]
	entry	entry	entry	
	entry	entry	entry	

に対する変更は [B]、[E] フィールドに限られます。

- [B]、[E] フィールド中の

```
\begin{table}, \end{table}
```

というコマンドをそれぞれ

```
\begin{table*}, \end{table*}
```

に変更します。

- その他の操作は通常の浮動型テーブル（セクション 11.1 参照）に対するものと変わりません。

## (3) 数式の配置

長い数式を配置する場合には (2) の延長、すなわち table\*環境を流用する形で対応します\*<sup>2</sup>。表の代りに数式を配置するわけですが、その場合、ディスプレイ数式ではなくインライン数式を配置する必要がある点に注意してください。このことは数式番号もアサインされないことを意味します。table\*環境に配置された数式にどうしても数式番号を付加したい場合には自分で整形する必要がありますが、その場合、T<sub>E</sub>X が自動生成する数式番号との重複を回避するといった操作も必要となってきます。具体的な操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0164.html>

をご参照ください。



\*<sup>2</sup> figure\*環境を利用しても構いません。



## 16

## カラー

T<sub>E</sub>X の文書中であっても *color* パッケージや *colortbl* パッケージを組み込むことによってカラーの使用が可能になります。本章ではそれらのパッケージ機能の用法について解説します。

## 16.1 *color* パッケージ

T<sub>E</sub>X の文書中でカラーを使用するためには *color* パッケージ、あるいは *colortbl* パッケージを文書に追加する必要があります。表中でのカラー設定を目的とした *colortbl* パッケージについては後述することにして、本セクションでは *color* パッケージの機能と用法について説明します。



SWP/SW の場合、タグの設定によって画面上の文字の色や背景色は種々変化します。しかしそれは編集操作の便宜を図るためだけのものであって、タイプセット結果に反映されるものではありません。

### (1) ドライバの選択

*color* パッケージを文書に追加する際、パッケージオプション中におけるドライバの選択<sup>\*1</sup>には注意が必要です。指定によっては出力結果がカラーにならないことがあるからです。次の表を参考にしてください。

<sup>\*1</sup> 「タイプセット」メニュー：「オプションとパッケージ」：「パッケージオプション」タブと操作し、*color* パッケージを選択後「編集」ボタンをクリックします。

ドライバ	欧文書		和文書	
	DVI	PDF	DVI	PDF
tcidvi (デフォルト)	○	○	×	×
dvips	×	×	○	○
pdftex	N/A	○	N/A	N/A
dvipdfm	N/A	N/A	N/A	○

## (2) カラーの制御

カラーを制御するための基本的なコマンドは `color` コマンドです。このコマンドはモード切替え型のため、例えば  $\TeX$  フィールド中に次のようなコマンドを設定した場合、

```
\color{blue}
```

後続のパラグラフは数式

$$y = f(x) \tag{16.1}$$

や表

<b>A</b>	<b>B</b>
aaa	bbb

も含めすべて指定されたカラーとなるため、

```
\color{black}
```

によって元の状態に戻す必要があります\*2。これに対し特定の文字列のみをカラー化したい場合には

```
\textcolor{red}{...}
```

の方が簡便でしょう。...の部分には該当する文字列を入力します。次はその用例です。

### カラーの使用例

\*2 日本語文書用の `dviout` の場合、`color` コマンドの効果はページ境界で失われます。

### (3) カラーモデル

赤とか青といった代表的な色については上記のように `red` とか `blue` という名称が用意されていますが、任意のカラーを使用したい場合には

```
\color[rgb]{0.8,0.2,0.4}
```

のような指定方法を取ることができます。この場合のカラーモデルとしては `rgb` の他に `cmymk`, `gray` がサポートされています。

### (4) カラー名称の定義

`black`, `white`, `red`, `green`, `blue`, `cyan`, `magenta`, `yellow` の 8 つの名称についてはドライバ非依存な形で利用できます。これら以外の色については独自に名称を定義した上で利用するという方法も取れます。例えばプリアンブル中に

```
\definecolor{darkblue}{rgb}{0.0, 0.0, 0.625}
```

という設定を入れておけば `darkblue` というカラーの利用がより簡便なものとなります。

### (5) 背景色

`\colorbox` コマンド、`\fcolorbox` コマンドを使用すると特定の文字列に対し背景色を設定することができます。なお、`\fcolorbox` の場合には枠付きとなります。例えば `TeX` フィールド中に

```
\colorbox[gray]{0.8}{ 網がけ }
```

と書けば `網がけ` のような背景色が、

```
\fcolorbox{black}{yellow}{ フレーム付 }
```

と書けば `フレーム付` のような枠付きの背景色が設定されます。

なお、*color* パッケージの詳細については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0060.html> をご参照ください。

## 16.2 colortbl パッケージ

`colortbl` パッケージを使用すると表の中の行や列、あるいは特定のセルに対して色塗りを施すことが可能になります。具体的には

```
\rowcolor[rgb]{x, y, z}
```

あるいは、

```
\columncolor[rgb]{x, y, z}
```

といったコマンドを表の中に埋め込んで行く形となるため ( $x, y, z$  は RGB カラーコード)、操作はそう簡単ではありませんが、次のような整形結果を得ることができます。

行の色塗り	列の色塗り	セルの色塗り																											
<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> </tbody> </table>	A	B	C	aaaa	bbbb	cccc	aaaa	bbbb	cccc	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> </tbody> </table>	A	B	C	aaaa	bbbb	cccc	aaaa	bbbb	cccc	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> <tr> <td>aaaa</td> <td>bbbb</td> <td>cccc</td> </tr> </tbody> </table>	A	B	C	aaaa	bbbb	cccc	aaaa	bbbb	cccc
A	B	C																											
aaaa	bbbb	cccc																											
aaaa	bbbb	cccc																											
A	B	C																											
aaaa	bbbb	cccc																											
aaaa	bbbb	cccc																											
A	B	C																											
aaaa	bbbb	cccc																											
aaaa	bbbb	cccc																											

`colortbl` パッケージの具体的な用法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0165.html>

をご参照ください。なお、ドライバの選択については上記 `color` パッケージの記述に準じます。



## 17

## 文献参照

論文や書籍の中では一般に数多くの文献が参照されます。本章では文献参照に関わるトピックについて整理しておくことにします。

## 17.1 文献一覧

参照文献の一覧は通常、文書の末尾に置かれるのが一般的です。この文献一覧を作成する場合、文献情報をあらかじめデータベース化しておき、そこから  $\text{BibTeX}$ 、あるいは  $\text{jBibTeX}$  を用いて体系的に作成する方法も取れますが、ここではより素朴な手入力による文献一覧作成を前提に解説を行います。 $\text{BibTeX}$ 、あるいは  $\text{jBibTeX}$  を用いて作成する場合の手順については以下を参照ください。

$\text{BibTeX}$	Creating Documents マニュアル (10.4.3)
$\text{jBibTeX}$	<a href="http://www.lightstone.co.jp/latex/kb0028.html">http://www.lightstone.co.jp/latex/kb0028.html</a>

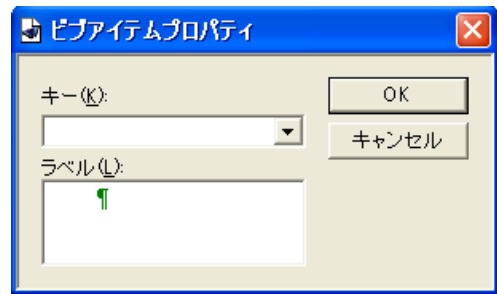
### (1) Bib item の設定

手入力で文献一覧を作成する場合には「アイテムタグ」フィールド内の“文献目録項目[Bib Item]”というアイテムを使用し、1 件ずつ文献を並べて行きます。



つい“文献一覧”等の見出しを付けたくになりますが、これはクラスファイルによって自動生成されるので特別な操作は不要です。

“文献目録項目[Bib Item]”というアイテムタグを選択すると右のようなダイアログが表示されるので、“キー”というフィールドには例えば“ref1”のような文献参照用のキーを入力します。その後、ダイアログを閉じ、実際の文献情報を記述します。一方、“ラベル”を指定した場合、文献参照は番号ではなくこのラベル名称を用いて行われます。この辺は具体例を見ていただくのが早いでしょう。次に示すのは jsarticle の場合の整形例です（文献引用部の様式については後述します）。



文献 [1] 参照  
 文献 [Farlow] 参照  
 文献 [2] 参照


## 参考文献

- [1] 奥村晴彦, *L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門*, 技術評論社, 2004  
 [Farlow] Stanley J. Farlow, *Partial Differential Equations for Scientists and Engineers*, Dover, 1993  
 [2] 俣野博, *常微分方程式入門*, 岩波書店, 2003

図 17-1 文献参照の例

この例では 2 番目の文献に対しラベルを指定しているため、それについては番号ではなくラベルによって参照が行われています。文献一覧側にしろ引用側にしろ、番号やラベルを [ ] で囲むのが標準様式です。別の様式への変更操作については後述します。



文献情報を入力後 ENTER キーを押すと次の bibitem の入力を促すダイアログが表示されます。それ以上入力しない場合にはダイアログを閉じ、さらに  を押して不要なアイテムを除去してください。

## (2) 文献一覧見出し

文献一覧に対する見出しはクラスファイルによって自動生成されますが、その名称はクラスによって異なります。

- 論文型文書の場合

article	jarticle	jsarticle
<b>References</b>	<b>参考文献</b>	<b>参考文献</b>

- 書籍型文書の場合

book/report	jbook/jreport	jsbook/jsreport
<b>Bibliography</b>	<b>関連図書</b>	<b>参考文献</b>


この設定を変更し独自の見出しを出力させる場合の操作法については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0059.html> をご参照ください。

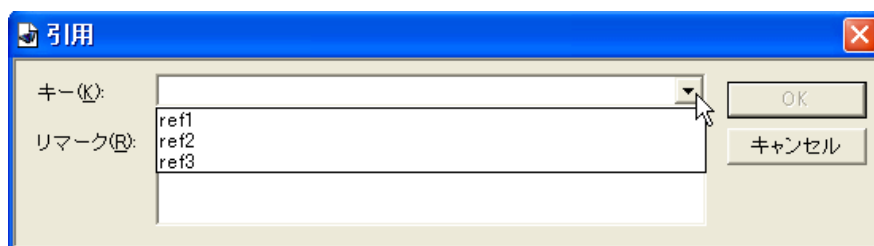
## (3) 括弧の変更

上述の通り、文献一覧中の番号は [ ] で囲まれた形式が標準様式ですが、これを ( ) に変更することも可能です。具体的な操作方法については SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0111.html> をご参照ください。

## 17.2 文献引用

## (1) 文献引用

文献一覧中の文献を本文中から引用するには引用ボタン  を使用します。このボタンをクリックすると次のようなダイアログが表示されるので、該当する文献のキーを選択してください。



SWP/SW 操作画面上には `cite:refl` のようなグレイボックスが置かれることとなりますが、これは  $\text{T}_{\text{E}}\text{X}$  の `\cite` コマンドに展開され、コンパイルの過程で文献番号、あるいは文献ラベルによって置き換えられます。

## (2) *cite* パッケージ

文献引用の標準様式は図 17-1 に示されているように

文献 [1]

文献 [2, 3, 4]

のようなインライン様式（上付きではない様式）となります。これに対し *cite* パッケージを使用すると同じインラインであっても

文献 [1]

文献 [2-4]

のように変わり、さらには上付き (superscript) の引用様式

文献<sup>1</sup>

文献<sup>2-4</sup>

も選択できるようになります。*cite* パッケージの具体的用法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0057.html>

をご参照ください\*<sup>1</sup>。



---

\*<sup>1</sup> WinForme として提供されている日本語文書用  $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  環境には *cite* パッケージが含まれていません。そのため SWP/SW 側からの移植操作が必要になります。

## 18

## 用紙のレイアウト

文書をタイプセットしてみると左右の余白がやや大きいといったことが良く起ります。本章では用紙のレイアウト調整に関するトピックについて解説します。

## 18.1 用紙サイズの設定

SWP/SW 環境下で文書を作成する場合、用紙のレイアウトに関するデフォルトの設定は次のようになっています。

- 欧文書の場合 – Letter size、縦置き
- 和文書の場合 – A4、縦置き

デフォルト以外の用紙サイズを使用したい場合には、クラスオプション（セクション 1.4 参照）によって設定を変更します。「タイプセット」メニュー：「オプションとパッケージ」：「クラスオプション」タブ：「編集」と操作し、“Paper size”の項を調整してください。



jarticle や jbook 等、従来の日本語文書の場合にはクラスオプションで用紙サイズを変更しても、dviout 等のドライバにはそれが伝わらず、ドライバ側での設定変更が別途必要でした。しかし jsc1ass の場合には“Papersize special”というクラスオプションが新設され、ドライバとの連動が可能になっています。用紙のサイズや向きを変更した場合には“Papersize special”オプションも同時に指定するようにしてください。

## 18.2 余白の調整

欧文書用の標準ドキュメントクラスである `article` や `book` を使用すると左右にかなり大きな余白が確保されます。マージンノート部を意識してのことと思われるのですが、この特性は和文書用の `jarticle` や `jbook` でも引き継がれています。



幸い `jsarticle` や `jsbook` ではこの点が是正されているので余白の調整は通常ほとんど必要ありません。

余白の大きさを調整するには `geometry` パッケージを利用するのが一般的です。パッケージを文書に追加した後、プリアンブル中に次のようなコマンドを設定します。数字の部分は適宜調整してください。

```
\geometry{left=25mm,right=25mm,top=25mm,bottom=25mm}
```

`geometry` パッケージを使用せず、`\textwidth` や `\oddsidemargin` といったページレイアウトパラメータを次のようにプリアンブル中で直接明示する方法も取れます。

```
\setlength{\textwidth}{142mm}  
\setlength{\oddsidemargin}{20mm}
```

指定できるパラメータにはどのようなものがあり、各々どの部分のサイズを意味するのかについて精通していないとこの方法は取れません。詳しくは `Typesetting Documents` マニュアルのページレイアウトの項、及び `layout` パッケージの項をご参照ください。

## 18.3 用紙の向き

### (1) 用紙の横置き

通常、用紙は縦置き (`portrait`) で使用されます。しかしスライド等で横置き (`landscape`) で使いたいという場合には“`Orientation`”というクラスオプションで“`Landscape`”を指定します。ただしこの場合、文書全体が横置きとなる点に注意してください。



`jsarticle` で“`Slide`”オプションを指定した場合には、用紙の向きは自動的に横置きとなります。

## (2) 部分的な横置き

文書全体としては縦置き (portrait) であっても、大きな表や図を収容するために文書の一部を横置きに設定したいというニーズはあります。これに対応するのが *portland* パッケージと *lscap* パッケージです。*portland* パッケージの場合、ページ単位に横置きにします。これに対し *lscap* パッケージの場合には特定パラグラフのみを転置させるというアプローチを取ります。詳細は SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0036.html>

をご参照ください。





## 19

## 書籍型/報告書型文書

これまでは主として論文型の文書を前提に説明を行ってきましたが、本章では書籍型文書、報告書型文書を対象に種々の注意事項について説明します。

## 19.1 書籍型/報告書型シェルの特性

SWP/SW 環境で書籍型、あるいは報告書型文書を作成する場合には、“Standard LaTeX”シェルフォルダ内にある `book`, `jbook`, `jsbook` シェル、あるいは `report`, `jreport`, `jsreport` シェルのいずれかを使用するのが一般的です。本セクションではまず、これらのシェルの特性やデフォルト設定値を一覧の形で整理しておきます。個々の項目に関する詳しい説明については次のセクションを参照ください。

## (1) 書籍型シェル

	<code>book</code>	<code>jbook</code>	<code>jsbook</code>
frontmatter, mainmatter の区別	あり	あり	あり
片面/両面	両面	両面	両面
章の開始ページ	右ページ	右ページ	右ページ
表紙/表題	表紙	表紙	表紙
目次	あり	あり	あり
概要(abstract)	なし	なし	(章別)
ページヘッダ	あり	あり	あり
付録	プリセット	プリセット	プリセット
索引	ユーザ	プリセット	プリセット

## (2) 報告書型シェル

	<b>report</b>	<b>jreport</b>	<b>jsreport</b>
frontmatter, mainmatter の区別	なし	なし	なし
片面/両面	片面	片面	両面
章の開始ページ	左ページ	左ページ	右ページ
表紙/表題	表紙	表紙	表紙
目次	あり	あり	あり
概要(abstract)	選択可	選択可	(章別)
ページヘッダ	なし	なし	なし
付録	プリセット	プリセット	プリセット
索引	ユーザ	ユーザ	ユーザ

## 19.2 項目別詳細

(1) `\frontmatter`, `\mainmatter` コマンド

書籍型シェル (book/jbook/jsbook) を使用する際に注意すべきは、シェル文書中に `\frontmatter`, `\mainmatter` という  $\text{T}_\text{E}\text{X}$  コマンドが埋め込まれているという点です。前者はフロントマター先頭部に、後者は本文の先頭部に配置されています。書籍型文書の場合、通常本文に先立ち序文が置かれることが多く、その中では

- ページ番号を通常のアラビア数字ではなくローマ数字とする
- 序文等の章見出しには章番号を付けない

等々、本文とは異なる制御が要求されるため、この区分が設けられています。詳細は各シェル文書中の記述を参照ください。なお、報告書型シェル (report/jreport/jsreport) においては `\frontmatter`, `\mainmatter` という  $\text{T}_\text{E}\text{X}$  コマンドは設定されていません。



SWP/SW 環境で文書を作成する場合、シェル文書のクリアというのが最初の作業となるケースが多いわけですが、書籍型シェルの場合には `\frontmatter`, `\mainmatter` コマンドまで削除してしまわないよう十分注意してください。

## (2) 片面印刷/両面印刷

これはクラスオプション“Print side”による設定項目なので自在に変更が可能ですが、デフォルト設定は書籍型が両面印刷、報告書型が片面印刷\*1となります。両面印刷の場合、左右のページで余白のサイズやページヘッダの様式に差が出てきますのでご注意ください。

## (3) 章の開始ページ

これもクラスオプション“Start chapter on left”による設定項目なので容易に変更は可能ですが、両面印刷の場合に章を左ページから始めて良いかどうかを規定します。デフォルトでは右ページという設定になっているので、各章末尾には空白ページが置かれることがあります。

## (4) 表紙/表題

これもクラスオプション“Title page”によって規定される項目です。デフォルトでは表題部を表紙として独立させる設定となっていますが、表題部と本文テキストとを同居させる設定も可能です。

## (5) 目次

書籍型、あるいは報告書型シェルの場合、フロントマター部に“Make TOC (Table Of Contents)”というアイテムがあらかじめセットされているので、表紙に続く形で目次が自動的に生成されます。不要な場合にはこのアイテムを除去してください。なお、アイテムタグフィールド中には

- Make LOF (List Of Figures)
- Make LOT (List Of Tables)

という項目も用意されているので、図表一覧を作成することもできます。



フロントマターに固有のアイテムタグはフロントマターダイアログを開いているときにのみ表示されます。

\*1 jsreport の場合は両面印刷。

通常、目次にはサブセクションのレベルまで反映されますが、jsbook, jsreport の場合にはセクションレベルまでしか反映されない設定となっています。これによって目次が煩雑化するのを防いでいるわけですが、この設定を変更したい場合には SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0154.html>

に記述されている操作を行ってください。



番号なしのセクションや章を設定した場合には、その見出しは目次中には反映されません。目次中にも反映させたい場合には SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0083.html>

に記述されている操作を行う必要があります。

## (6) 概要 (abstract)

書籍型シェルの場合には“概要/Abstract”というアイテムタグが選択できません。これに対し、報告書型シェル report, jreport の場合には“概要/Abstract”というアイテムタグが選択できるようになっています。一方、jsbook, jsreport の場合には少々独自の対応を取っており、各章の先頭部に概要の記述を設けることが可能になっています。具体的な操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0044.html>

をご参照ください。

## (7) ページヘッダ

シェルをタイプセットしてみるとわかりますが、書籍型文書の場合には各ページの上部にヘッダ情報が自動的に出力されます\*2\*3。これに対し、*fancyhdr* パッケージを使用して、ヘッダ/フッタの様式を独自に設定するといったアプローチを取ることも可能です。使用方法については Typesetting Documents マニュアルをご参照ください。

\*2 各章の先頭ページにはヘッダ情報は出力されません。

\*3 jsbook の場合には罫線も引かれます。

## (8) 付録

$\text{T}_{\text{E}}\text{X}$  の `\appendix` コマンドに続く章は付録の章とみなされ、**Appendix A**, **Appendix B** (あるいは付録 A, 付録 B) といった章番号が自動的に設定されます。この `\appendix` コマンドは  $\text{T}_{\text{E}}\text{X}$  フィールドを用いて設定しても良いのですが、`appendix` フラグメントを利用するのが確実な方法でしょう。標準シェルには皆付録の章が含まれているので一度タイプセットを行い、その様式を確認しておくといいでしょう。なお、章番号が A とか B といったアルファベットになる関係で、付録中に設定した数式や図表に対する番号様式は A.1 とか B.1 となる点にご注意ください。SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0082.html>

も併せてご参照ください。

## (9) 索引

書籍型文書にしる報告書型文書にしる、`makeidx` パッケージを組み込むことで索引を作成できるようになります。具体的な操作方法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0018.html>

を参照してください。なお、`jbook`, `jsbook` シェルの場合には `makeidx` パッケージを含め、必要な設定項目がシェル文書中に埋め込まれているので、索引を作成される際の参考になると思います。また索引の様式をカスタマイズすることも可能です。その手順については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0040.html>

をご参照ください。

## (10) 番号様式

書籍型文書、あるいは報告書型文書の場合には章が文書の基本的な構成単位となります。これに伴い、数式や図表に対し  $\text{T}_{\text{E}}\text{X}$  がアサインする番号の様式も章を持たない論文の場合とは異なったものとなります。具体的には“ch.n”という番号様式が用いられます。

- ch - 章番号
- n - 章内での通し番号

以上述べてきたように、書籍型シェル、報告書型シェルには論文型文書にはない種々の機軸が組み込まれています。目的に最も良く合致したシェルをベースに文書作成を進めることが大切です。



jsbook の場合、テキスト行の長さを全角 40 文字以下に抑えるというロジックがクラスファイル中に組み込まれています。これは読みやすさを配慮しての仕様であるわけですが、使用する用紙サイズや文字サイズによっては不自然な余白を生む結果となります。この制限を外したい場合には“Standard LaTeX”シェルフォルダ中にある

- `[jsbook] fullwidth`  
というシェルを選択してください。



書籍型文書の場合、画像等を使って少し凝った表紙を用意したいといったニーズが考えられます。ライトストーンでは jsbook を用いたときの文書構成例をテンプレートの形で紹介しています。SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0131.html>

をご参照ください。



## 20

## マスター/サブ文書

ページ数の多い書籍型/報告書型文書を作成する場合には、マスター/サブという文書構造を導入した方が運用が楽になります。本章ではマスター/サブ文書の利用法について解説します。

## 20.1 マスター/サブ文書構成

ページ数の多い文書を作成する場合には図 20-1 に示されるような形に文書を分割して管理する方法を推奨します。このようなマスター文書とサブ文書に分ける構成を取った場合、それぞれの文書ごとにコンパイルが行えるため、編集・開発作業の効率が大きく向上します。もちろんサブ文書単独でのコンパイルではクロスリファレンスが未解決になることもあるので、マスター文書コンパイルの段階での確認作業の必要性は依然残りますが、毎回大きな文書全体をコンパイルすることに比べれば文書作成ははるかに楽なものとなります。

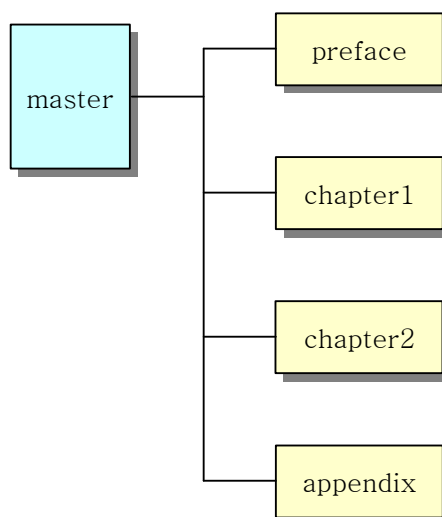

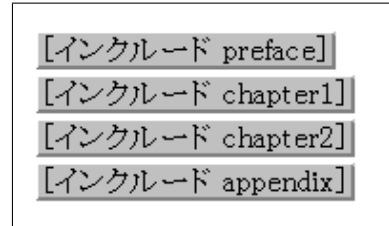


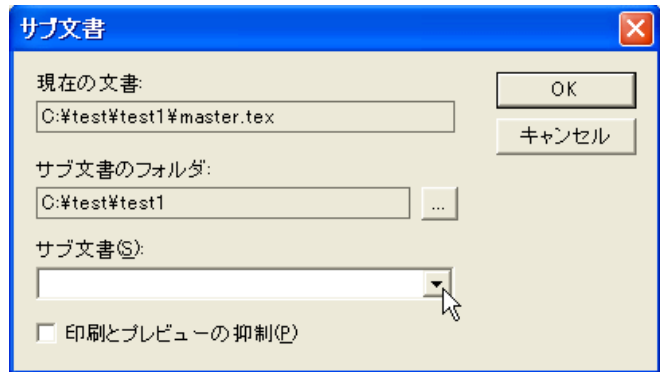
図 20-1 マスター/サブ文書構成


## 20.2 マスター文書の作成

マスター文書は通常、書籍型、あるいは報告書型シェールを用いて作成しますが、中には文書のパラグラフや画像等を含めないのが一般的です。中に配置されるのは右に示されるようなサブ文書を取り込むための制御コードのみです。この“インクルード”と書かれたグレイボックスはサブ文書ボタン  をクリックすることによって生成されます。



サブ文書ボタンをクリックすると右のようなダイアログが開かれるのでサブ文書の名称を入力してください。マスター文書と同一のフォルダ内に空白のサブ文書が生成されます。異なるフォルダ中にサブ文書を置きたい場合には“サブ文書のフォルダ”というフィールドでフォルダ名を指定してください。



マスター文書がファイル上に生成されていない状態ではサブ文書ボタン  は有効になりません。まず保存操作を実行してください。

“インクルード”というグレイボックスは  $\text{T}_{\text{E}}\text{X}$  の `\input` コマンドに展開されます。例えば

```
\input{chapter1.tex}
```

というコマンドであればマスター文書のこの位置にサブ文書 `chapter1.tex` が読み込まれる形となります。マスター文書はこのようにすべてのサブ文書を束ねる役割を果たすわけですが、これとは別にもう一つ重要な役割をマスター文書は担っています。マスター/サブという構成を取った場合、サブ文書側には文書全般に関わる情報は存在しなくなります。具体的には


- ドキュメントクラス
- 使用するパッケージ
- フロントマター
- プリアンブル

といった情報や文書要素はマスター文書側でのみ管理されます。このため文書全般に関わる設定変更は常にマスター側で行うという注意が必要となります。



jsbook の場合には、マスター/サブという文書構成を前提としたテンプレートがあらかじめ用意してあります。SWP/SW BetterUse ページ <http://www.lightstone.co.jp/latex/kb0131.html> をご参照ください。

## 20.3 サブ文書の作成


サブ文書は上記のようにサブ文書ボタン  によって作成されたものを使用してください。生成されたサブ文書をテキストエディタで開いてみるとわかりますが、ドキュメントクラスの宣言もなければ使用パッケージの宣言もありません。代りに先頭部にあるのは

```
%TCIDATA{LaTeXparent=0,0,master.tex}
```

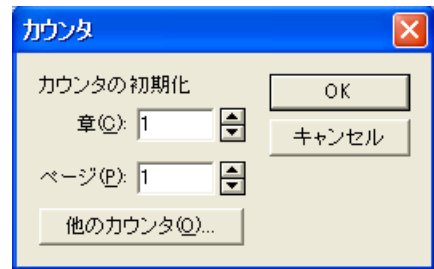
のような SWP/SW に対する制御文 (TeX に対してはコメント文) です。この情報に基き SWP/SW はそれをサブ文書と認識、マスター文書を特定し、そこから文書全体に関わる情報を抽出するといった操作を行います。このため、サブ文書として作成されていない通常の既存文書をそのままサブ文書として流用することはできません。



作成済みの文書をサブ文書として利用したい場合には

- まずサブ文書ボタン  によって白紙のサブ文書を作成
- 次にそのサブ文書を開き、「ファイル」メニュー：「内容のインポート」と操作という手順によって、既存の文書の内容をサブ文書内にコピーしてください。なお、画像等の付帯ファイルがある場合には外部フォルダへの依存関係が生ずることがあるので注意が必要です。

サブ文書は単体でタイプセットが行えます。その際、右のようなダイアログが表示されるので、必要に応じてカウンタ類を適当な値にセットしてください。ただしこれらの設定はサブ文書単体でのタイプセットに対して有効になるだけで、マスター文書からのタイプセット時には意味を持ちません。



## 20.4 統合タイプセット

サブ文書単体でのタイプセットでは、サブ文書間にまたがったクロスリファレンスや文献参照が未解決といった問題が残るので、それらについてはマスター文書からのタイプセットによって最終確認を行うことになります。マスター文書からのタイプセット時には、

- 相互参照の完全性
- 目次
- 索引

等を中心にチェックを行ってください。



PDF タイプセット時には

<http://www.lightstone.co.jp/latex/kb0031.html>

に記したような問題がありますのでご注意ください。



## 21

## スライド

数式の高品位整形が  $\text{T}_\text{E}\text{X}$  の最大の特長であるわけですが、最近ではその特長を保持したままプレゼンテーション用スライドの作成が行えるようになりました。

従来スライドを作成する場合には、 $\text{T}_\text{E}\text{X}$  で作成した数式を画像の形で PowerPoint に貼り付けるという煩わしい操作が必要でした。しかし最近では  $\text{T}_\text{E}\text{X}$  の世界でもスライド作成が可能です。実際 SWP/SW v5.5 では

- `jsarticle slide` オプション
- Beamer スライド

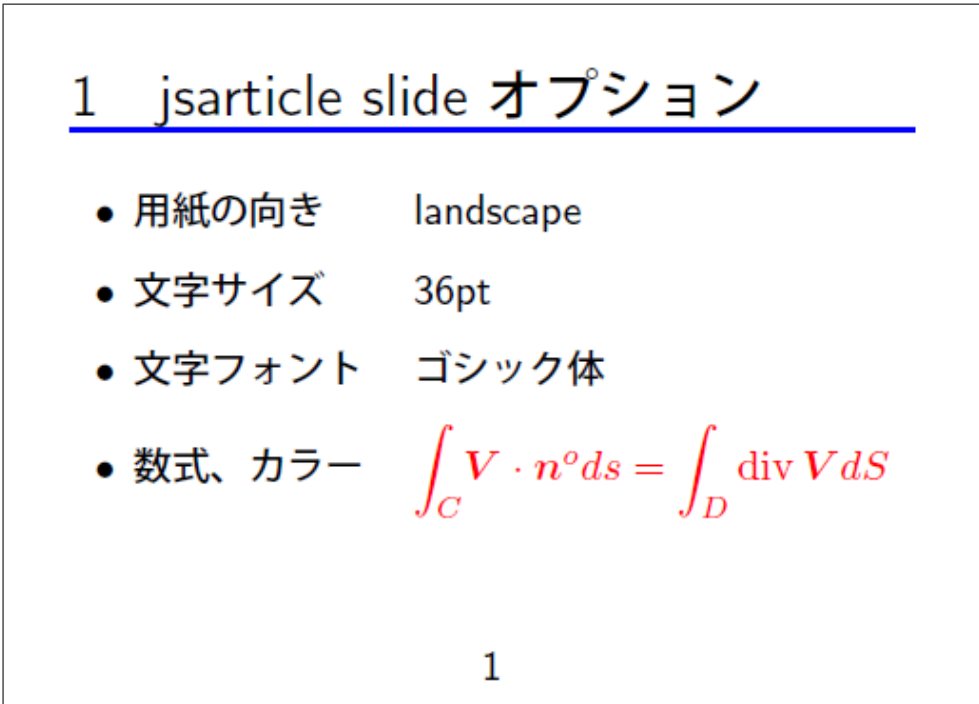
という 2 種類のスライド作成機能を提供しています。

## 21.1 `jsarticle slide` オプション

SWP/SW v5.5 で新たにサポートされたドキュメントクラス `jsarticle` にはスライド作成機能があらかじめ組み込まれています。図 21-1 はその一例ですが、クラスオプションで“Slide”を選択するだけで容易にスライド作成が可能になります。この `jsarticle` ベースのスライド作成を支援すべく、Standard LaTeX シェルフォルダ中には

- `[jsarticle] slide1`
- `[jsarticle] slide2`

というシェルが用意されています。



1 jsarticle slide オプション

- 用紙の向き landscape
- 文字サイズ 36pt
- 文字フォント ゴシック体
- 数式、カラー  $\int_C \mathbf{V} \cdot \mathbf{n}^\circ ds = \int_D \operatorname{div} \mathbf{V} dS$

1

図 21-1 jsarticle スライドサンプル

また SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0053.html>

からは

- [jsarticle] slide3

というシェルもダウンロードできますのでご利用ください。これらのシェルを使用した場合には基盤の文字サイズが大きくなるため、数式用フォントもそれに合せて拡大されます。また画像や表、カラーの機能等についても、通常の SWP/SW 文書の場合と同様利用できます。

## 21.2 Beamer スライド

このスライド作成機能は `beamer.cls` という専用のクラスファイルを組み込むことによって利用できるようになります。元々英語版のみが SWP/SW 開発元の MacKichan 社より提供されていましたが、その後ライトストーン側で日本語スライド作成に使えるものも用意しました。共に弊社 Web サイトからのダウンロード、インストールという形態になります。詳しくは SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0105.html>

<http://www.lightstone.co.jp/latex/kb0141.html>

<http://www.lightstone.co.jp/latex/kb0142.html>

をご参照ください。

What is Beamer?

- Beamer is a  $\text{\LaTeX}$  document class that produces beautiful  $\text{PDF}\text{\LaTeX}$  presentations and transparency slides.
- Beamer presentations feature
  - $\text{PDF}\text{\LaTeX}$  output.
  - Global and local control of layout, color, and fonts.
  - List items that can appear one at a time.
  - Overlays and dynamic transitions between slides.
  - Standard  $\text{\LaTeX}$  constructs.
  - Typeset text, mathematics  $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ , and graphics.
- To produce a sample presentation in *SWP* or *SW*, typeset this shell document with  $\text{PDF}\text{\LaTeX}$ .

MacKichan Software Technical Support (De)Creating Beamer presentations in Scientific V [07/06] March 2006 2 / 15

図 21-2 Beamer スライドサンプル



## 22

## その他

SWP/SW で文書作成を行う場合に知っておくと便利な機能が他にも数多くあります。本章ではそれらの中から主だったものについて簡単な紹介を行います。

## 22.1 ハイパリンク

本ガイドがその一例ですが、文書中にある種々の参照をハイパリンク化することが可能です。基本的には *hyperref* パッケージを文書に追加するだけです。ただし

- このパッケージは  $\text{T}_{\text{E}}\text{X}$  の出力に種々の修正を加えることになるため、パッケージリストの末尾に置く必要がある。
- 日本語文書で利用する際には文字コードに関連した特殊な  $\text{T}_{\text{E}}\text{X}$  コマンドをプリアンブル中にセットする必要がある。
- ハイパリンクの機能が使えるのは PDF 文書に限られる。

という 3 点に注意してください。 *hyperref* パッケージの用法については SWP/SW BetterUse ページ

<http://www.lightstone.co.jp/latex/kb0011.html>

をご参照ください。

## 22.2 HTML, RTF エクスポート

SWP/SW で作成した文書を HTML 形式の文書としてエクスポートし Web ページとして使用したり、RTF (Rich Text Format) 形式の文書としてエクスポートし Microsoft Word 環境で利用するといった運用が可能です。しかしこのエクスポートの過程には  $\text{T}_{\text{E}}\text{X}$  が関与しないという点に

注意してください。例えばセクション番号とか数式番号とかは生成されなくなります。ただし数式の整形については SWP/SW が独自に行うため、それなりの品位が維持された形で HTML や RTF 文書が作成されます。具体的には「ファイル」メニュー：「文書のエクスポート」と操作しますが、詳細は SWP/SW BetterUse ページ

[http://www.lightstone.co.jp/latex/bu\\_index8.html](http://www.lightstone.co.jp/latex/bu_index8.html)

をご参照ください。

## 22.3 学会への論文投稿

学会に論文を投稿する場合には学会固有の様式で原稿を準備する必要があります。通常、学会は固有の  $\text{T}_\text{E}\text{X}$  クラスファイルやスタイルファイルを用意しており、投稿者はそれを前提にした形で原稿を作成することになります。一般の  $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$  環境であればこれらの独自クラスファイル、スタイルファイルの組込みは比較的簡単に行うことができますが、SWP/SW環境でそれらを扱えるようにするのは簡単な作業ではありません\*1。SWP/SWとしての学会固有クラスファイル、スタイルファイルへの対応は次のようになっています。

### (1) 欧米の学会への投稿

欧米の主だった学会については、それらに対応するシェルファイルがあらかじめ製品に組み込まれて提供されています。“Articles”シェルフォルダの内容をご確認ください。ただしこれらは製品出荷時点のものに固定されているため、最新のクラスファイルやスタイルファイルには対応できていない可能性があります。このため SWP/SW の製品開発元である MacKichan 社としては、最新仕様に対応したシェルファイルを Web 上で提供する形でサービスを行っています。

<http://www.mackichan.com/index.html>

をご参照ください。

### (2) 日本の学会への投稿

ライトストーンでは主として国内の学会を対象に、学会固有クラスファイル、スタイルファイルの一部サポートを行っています。詳細については SWP/SW BetterUse ページをご参照ください。

[http://www.lightstone.co.jp/latex/bu\\_index1.html](http://www.lightstone.co.jp/latex/bu_index1.html)

またこの一覧に掲載されていない学会についても対応できる場合がございます。

ご要望があれば技術サポート

[tech@lightstone.co.jp](mailto:tech@lightstone.co.jp)

までご連絡ください。

---

\*1 学会が前提としている日本での一般的な $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$ 環境とSW/SWPの環境は異なるため。

## 22.4 その他の注意事項

SWP/SW をお使いいただく上で基本事項とも言えるものでありながら良く問合せを受ける項目がいくつかあります。ここではそれらの項目について説明を補足しておきます。

### (1) 文書の保存操作

文書を画面上で編集を行うとファイルに保管されている状態と差が生じます。この状態のままタイプセット操作を行うと、折角の DVI や PDF ファイルが一過性のファイルとして SWP/SW プログラムフォルダ内に生成される形となります。一過性ですからそれらは文書を閉じたときに消滅してしまいます。新規にシェルファイルから文書作成を行った場合も、ファイル保存操作を入れずにタイプセットを行うと同様の状況が発生します。タイプセットを行う際には必ずその直前でファイル保存操作を行うようにしてください。DVI や PDF ファイルの他、種々の付帯ファイルが恒久ファイルとして、該当文書と同一のフォルダ中に生成されるようになります。

### (2) ファイルの種類

「ファイル」メニュー：「名前を付けて保存」と操作すると次のようなダイアログが表示されます。



このうち“ファイルの種類”という項で“SW/SWP/SN Document”を選択するとプリアンブル中にある

```
\input{tcilatex}
```

というコマンドが有効になってしまいます。これは SWP/SW 固有のマクロファイルを読み込むためのコマンドであり、一般  $\text{\LaTeX}$  環境で実行するとファイルが存在しないということでエラーが発生することになります。SWP/SW で作成した文書を一般  $\text{\LaTeX}$  環境に持ち出す予定がある場合にはファイルの種類として“Portable LaTeX”を選択してください。一般  $\text{\LaTeX}$  環境と整合性のある  $\text{\TeX}$  コードが生成されます。

### (3) 文字コード

SWP/SW v5.5 では文書を保存する際、文字コードを指定できるようになりました（上記ダイアログの“キャラクタセット”の項を参照）。従来、日本語文書は Unicode で生成されていたのですが、v5.5 では一般  $\text{\LaTeX}$  環境に併せてシフト JIS コードでの保存が可能になりました。日本語文書を保存する場合にはシフト JIS コードを選択するようにしてください\*2。なお、欧文書を保存する場合には ASCII コードを選択します。

### (4) ファイルパス名

$\text{\TeX}$  のコンパイルに際しては数多くのプログラムが実行されますが、そのほとんどは UNIX 環境で開発されたものです。このため文書自体や付帯画像ファイルが“マイ $\text{\LaTeX}$ ドキュメント”等、全角文字やスペースを含むフォルダに格納されていた場合には、ファイルパス名の認識に支障を来します。SWP/SW 文書や画像ファイルを保管する場合には、ファイルパス名中に全角文字やスペース、アンダスコア (.) 等の不適切な文字が含まれないように注意してください。



---

\*2 V5.0 以前のシステムで処理する必要がある場合には ASCII コードを選択してください。日本語の文字には Unicode が使用されるようになります。

## 索引

## A ~ Z

- \Huge, 28
- \LARGE, 28
- \Large, 28
- \addcontentsline, 14
- \boldsymbol, 30
- \caption, 55
- \color, 80
- \colorbox, 81
- \documentclass, 1
- \fcolorbox, 81
- \footnotesize, 28
- \frontmatter, 92
- \huge, 28
- \label, 14
- \large, 28
- \mainmatter, 92
- \markboth, 14
- \mathbf, 30
- \mathit, 30
- \normalsize, 28
- \ref, 57
- \renewcommand, 15
- \scriptsize, 28
- \setcounter, 15
- \small, 28
- \textbf, 28
- \textcolor, 80
- \textgt, 28
- \textit, 28
- \textrm, 28
- \textsc, 28
- \textsf, 28
- \textsl, 28
- \texttt, 28
- \tiny, 28
- \usepackage, 2
- \vspace, 19
- 2 段組, 4
- 2 段組み, 73
- abstract, 7, 94
- align 環境, 33
- amsmath パッケージ, 30, 34
- Beamer スライド, 5, 103
- BibTeX, 83
- cite パッケージ, 86
- colortbl パッケージ, 82
- color パッケージ, 79
  - カラー名称, 81
  - カラーモデル, 81
  - ドライバ, 79
  - 背景色, 81
- Columns オプション, 73
- description 環境, 23
- enumerate 環境, 23
- eqnarray 環境, 33
- equation 環境, 33
- fancyhdr パッケージ, 94
- figure\*環境, 75, 76
- figure 環境, 75
- float パッケージ, 55, 68
- fncychap パッケージ, 15
- geometry パッケージ, 88
- Ghostscript, 68
- HTML エクスポート, 105
- hyperref パッケージ, 105
- itemize 環境, 23
- jBibTeX, 83
- jsarticle, 3

jsclass, 3

landscape, 88

layout パッケージ, 88

lscape パッケージ, 89

makeidx パッケージ, 95

multicol パッケージ, 75

numinsec パッケージ, 45

papersize special, 4

pLaTeX, 6

portland パッケージ, 89

portrait, 88

preference, 55

RTF エクスポート, 105

sectsty パッケージ, 15

slide オプション, 101

Susie plug-in, 68

table\*環境, 75, 76

table 環境, 54, 75

tabular 環境, 53

theorem パッケージ, 52

TrueTeX, 6

WinForme, 6

wrapfig パッケージ, 69

## あ ~ お

アイテムタグ, 23

アラインメント, 60

位置揃え  
数式の位置揃え, 38

インデント, 19

インポート  
画像のインポート, 65

引用, 85

インライン画像, 66

インライン数式, 33

## か ~ こ

改ページ, 20

簡条書き, 23

括弧, 42

カラー, 79

関数名, 42

概要, 7, 94

学会固有クラスファイル, 5, 106

キー

表の参照, 57

記号付きリストアイテム, 23

キャプション, 55

強制改行, 20

空白, 17

クラスオプション, 3

クラスファイル, 1

罫線, 60

## さ ~ そ

索引, 95

サブ文書, 97, 99

シエル, 1

書籍型シエル, 91

字下げ, 19

数式番号, 5, 14, 36, 45

独自番号, 46

長い数式, 47

配置様式, 46

複数行の数式, 46

数式モード, 33

スペース, 17

スライド, 5, 101

図表一覧, 93

整列, 60

添字, 41

## た ~ と

縦スペース, 19

段組み, 73

ツールバー, 41

テーブル

固定型のテーブル, 53

浮動型のテーブル, 53

定理型環境

カスタマイズ, 51

定義, 49

番号付け, 50

ラベル, 50

テキストタグ, 27

テキストモード, 33

ディスプレイ画像, 66

ディスプレイ数式, 4, 33, 43

位置揃え, 44

改行, 43

改ページ, 44

配置様式, 44

## な ~ の

## は ~ ほ

配置

画像の配置, 65

表の配置, 55

ハイパリンク, 105

柱, 14

番号

数式, 95

図, 95

表, 95

番号付きリストアイテム, 23

番号様式, 95

パッケージ, 2

表

固定型の表, 53

浮動型の表, 53

プロパティ, 59

表紙, 4

表題, 4, 7

表番号, 55

微分記号, 43

フローティング画像, 67

付録, 95

フロントマター, 7

ブレーク, 20

文献一覧, 83

文献引用, 85

プリアンブル, 7

プロパティ

数式のプロパティ, 35

ディスプレイ数式のプロパティ, 36

ヘッダ, 14, 94

ベクトル, 30, 42

ページヘッダ, 94

報告書型シェル, 91

ポートレイト, 88

## ま ~ も

マーカー, 14

マスター文書, 97, 98

見出し, 13

見出し付きリストアイテム, 23

見出し番号, 13

参照, 14

省略, 13

設定, 15

目次, 14, 93

文字サイズ, 3, 27, 29

文字フォント, 27

## や ~ よ

用紙サイズ, 3, 87

用紙の向き, 3, 88

横スペース, 18

余白, 88

## ら ~ ろ

ランドスケープ, 88

リードインオブジェクト, 23, 25

リストアイテム, 23

ルール, 21

列幅, 60

論文投稿, 106

わ ~ ん