

XPDFJ 標準マクロ概説

2005 年 2 月 19 日
(株) ネットストック
中島 靖
<nakajima@netstock.co.jp>

1. はじめに	1
2. 原稿の基本パターン	2
2.1 原稿の基本パターン	2
2.2 XML の書き方	2
3. 原稿の書き方	4
3.1 stddefs.inc の読み込み	4
3.2 本文	4
3.2.1 本文を囲む <BODY>	4
3.2.2 見出し	5
3.2.3 通常の段落	6
3.2.4 文字の修飾	6
3.2.5 段落の体裁	7
3.2.6 フォント	8
3.2.7 箇条書き	9
3.2.8 表	10
3.2.9 その他の本文用マクロ	12
3.3 ヘッダとフッタ	13
3.4 索引	13
3.5 目次	14
4. 簡単なマクロ定義	15
索引	16

1. はじめに

XPDFJ は、PDFJ を生成する Perl モジュール PDFJ の、XML フロントエンドです。XML 形式で書いた原稿から PDF を生成することができます。

XPDFJ の基本機能は、PDFJ の各種の命令を XML 形式で書けるようにすることですが、それだけであれば PDFJ を直接使った Perl スクリプトを書くのをより面倒にしかけです。XPDFJ には一連の命令をまとめて新たな命令を定義できるマクロ機能が用意されており、これによってより簡潔な原稿から PDF を生成することができるようになります。

この文書は、XPDFJ の標準マクロである `stddefs.inc`、`toc.inc`、`index.inc` の三つのマクロファイルを使った文書作成の基本を、デモを兼ねて説明するものです。この文書自体がこれらの標準マクロを使って作成されていますので、原稿テキストである `stdmacro.xp` を実例としてご覧いただくと理解が深まるかと思います。

XPDFJ の標準マクロは HTML に慣れた方が使いやすいように HTML ライクに作られています。HTML のタグ名と同じ名前のマクロでも属性に違いがある場合がほとんどで、互換性はありません。XPDFJ の標準マクロの属性は、PDFJ の機能に由来しているものが多くあります。標準マクロを十分使いこなすためには PDFJ の機能を理解していただくことが不可欠となりますので、PDFJ の説明書を併せてご覧いただくようお願いします。

2. 原稿の基本パターン

2.1 原稿の基本パターン

XPDFJ の標準マクロを使った文書原稿の基本パターンは次のようになります。まず目次も索引もない簡単な文書の場合です。フッタにページ番号だけは出すようにしています。

先頭行の xml 宣言で、文字コードを正しく指定する必要があります。Windows のシフト JIS の場合は "x-sjis-cp932"、日本語EUC の場合は "x-euc-jp-jisx0221" を指定します。

xml 宣言の後で、必ず全体を <XPDFJ>...</XPDFJ> で囲まねばなりません。

```
<?xml version="1.0" encoding="x-sjis-cp932"?> <!-- 文字コードを指定する -->
<XPDFJ version="0.2">
<do file="stddefs.inc"/> <!-- 標準マクロファイルの読み込み -->
<FOOTER page="all" pstyle="align:m">{page}</FOOTER> <!-- フッタにページ番号 -->
<BODY>
...ここに本文を書く...
</BODY>
<print file="$Args{outfile}"/> <!-- 原稿ファイル名に .pdf を付加した名前で出力 -->
</XPDFJ>
```

表紙、目次、本文、索引を持った文書の場合は、次のようになります。表紙と目次は最後に挿入していることに注意してください。本文を作った後でないと目次が作れないからです。

```
<?xml version="1.0" encoding="x-sjis-cp932"?> <!-- 文字コードを指定する -->
<XPDFJ version="0.2">
<do file="stddefs.inc"/> <!-- 標準マクロファイルの読み込み -->
<do file="toc.inc"/> <!-- 目次用マクロファイルの読み込み -->
<do file="index.inc"/> <!-- 索引用マクロファイルの読み込み -->
<HXOUTLINE/> <!-- 見出しのアウトライン化の指定 -->
<FOOTER page="all" pstyle="align:m">{page}</FOOTER> <!-- フッタにページ番号 -->
<BODY>
...ここに本文を書く...
</BODY>
<BODY cols="3"> <!-- 索引は3段組で -->
<H2> 索引 </H2>
<MAKE INDEX/> <!-- 索引 -->
</BODY>
<FOOTER page="all" clear="1"/> <!-- 表紙と目次はフッタのページ番号はなし -->
<BODY page="1" evenpages="1"> <!-- 表紙と目次は先頭へ挿入し複数ページとする -->
...ここに表紙の内容を書く...
<TOC/> <!-- 目次 -->
</BODY>
<print file="$Args{outfile}"/> <!-- 原稿ファイル名に .pdf を付加した名前で出力 -->
</XPDFJ>
```

2.2 XML の書き方

XML の書式の詳しいことは XML の解説本をご覧ください。ここではとりあえずこれだけ注意すれば書けるというポイントに絞って説明しておきます。

- ・先頭には必ず <?xml version="1.0" encoding="..."?> を書かねばなりません。原稿の文字コードと指定した encoding が合わないとエラーになります。
- ・大文字小文字は区別されます。<BODY> を <Body> と書いたりしてはダメです。特に HTML に慣れた方は注意してください。
- ・<XXX...> が開始タグ、</XXX> が終了タグで、入れ子になっても構いませんが、開始タグと終了タ

グがきちんと対応するように注意してください。<XXX...></XXX> と内容がない場合は <XXX... /> と書くことができます。HTML のように <P> だけ書いて </P> を書かないのはダメですし、
 は
 と書かねばなりません。

- ・ 開始タグの中の「名前=" 値 "」あるいは「名前=' 値 '」を属性と言います。
- ・ タグを書くため以外に < や > を書くときは、< > とします。その他、& は &、' は '、" は " と書けます。
- ・ < や > をたくさん含むようなテキストを書くときに一つ一つ < などと書くのが面倒なら、<![CDATA[と]]> で囲むとそのまま書けます。
- ・ <!-- から --> まではコメントになります。コメント中には -- という文字列は入れられないので、コメントは入れ子にはできません。

3. 原稿の書き方

3.1 stddefs.inc の読み込み

`<do file="stddefs.inc"/>` で、標準マクロファイル `stddefs.inc` を読み込むと、それ以降でこのマクロファイルで定義された各種のマクロが使えるようになります。このとき、文書の各種のパラメータが次のように設定されます。単位はポイント(1ポイント = 1/72 インチ)、パーセントはフォントサイズに対する比率です。つまりこれはA4用紙に周囲の余白を1インチ取り、本文フォント10ポイントで行送り15ポイントということになります。

linefeed は行送りであって行間ではないことに注意してください。linefeed が 150% なら行間は 50% ということになります。

【表：stddefs.inc の属性と省略時設定】

属性名	省略時値	意味
pagewidth	595	ページ幅
pageheight	842	ページ高さ
leftmargin	72	左余白
rightmargin	72	右余白
topmargin	72	上余白
bottommargin	72	下余白
fontsize	10	本文フォントサイズ
linefeed	150%	本文行送り

これを例えば、B5 の用紙で余白 1.5cm、フォントは 9 ポイントとしたければ、

```
<do file="stddefs.inc" pagewidth="516" pageheight="728"
margin="42" fontsize="9"/>
```

とすれば良いわけです。margin 属性を指定すると、上下左右の余白がすべてその値に設定されます。もちろん各 margin 属性で個別に指定することもできます。なお、通常は横書き用のフォントのみが設定されます。縦書き用のフォントを設定する方法も用意されていますが、ここでは扱いません。

3.2 本文

3.2.1 本文を囲む <BODY>

前述の基本パターンのように、本文全体は<BODY> で囲みます。HTML と違い、<BODY> の中に直接文字を書くことはできず、後述する <H*><P><TABLE> などが <BODY> の内容になることに注意してください。<BODY> は何度書いても構いませんが、各 <BODY> は新しいページから始まることに注意してください。

<BODY cols="2"> のように cols 属性を指定すると 2 段組や 3 段組にすることもできます。段間(段と段の間隔)はデフォルトでは 20 ポイントですが、<BODY cols="3" colskip="15"> のように colskip 属性で変更できます。

通常は <BODY> による新しいページは末尾に追加されていきますが、<BODY page="1"> のようにすると指定のページ番号の位置へ挿入できます。ただしページの挿入をおこなうとそれ以降のページ番号がずれることに注意してください。この機能は後述する表紙や目次で使います。

次に <BODY> の属性をあげておきます。

【表：<BODY>の属性】

属性名	値と機能
page	開始ページ番号（この位置にページを挿入。省略すると末尾に追加）
hfpag	ヘッダやフッタに表示するページ番号の開始値（省略すると page と同じ）
cols	段数（省略時値 1）
colskip	段間（省略時値 20）
evenpages	真ならページ数が偶数になるように空ページが付加される

表に「真なら」とありますが、0 と空文字列以外は真と判断されます。普通は 1 を指定します。以下でも同様です。

3.2.2 見出し

見出しは、<H1> 表題 </H1> のように<H1><H2><H3><H4> の 4 種類のタグで囲んで書きます。（以下の説明で H1 ~ H4 を総称して H* と書くことにします。）<H1> は文書の表題用で、大きなフォントで中央寄せされます。<H2><H3><H4> はそれぞれ、大見出し、中見出し、小見出しです。

見出しに「1. 大見出し」「1.1 中見出し」「1.1.1 小見出し」のように自動番号付けをしたい場合は、<HXNUMBER> を使って <BODY> の前に次のように書きます。

```
<HXNUMBER level="2" number="{H2}." />
<HXNUMBER level="3" number="{H2}.{H3}" />
<HXNUMBER level="4" number="{H2}.{H3}.{H4}" />
```

number 属性で指定した文字列中の {H*} が対応する H* が何番目かという番号に置き換えられて、見出しの前に付加されます。<H*>の属性で nonumber="1" と指定するとその見出しには自動番号付けはされず、カウントもされません。

見出しをアウトライン(しおり)にしたい場合は、<HXOUTLINE/> を <BODY> の前に書きます。デフォルトでは <H2><H3> がアウトライン化されます。<H4> もしたければ、<HXOUTLINE level="2,3,4"/> と指定します。

つぎに <H*><HXNUMBER><HXOUTLINE> の属性をあげておきます。

【表：<H*><HXNUMBER><HXOUTLINE>の属性】

属性名	値と機能
<H*>	
nooutline	真ならアウトラインにしない
nonumber	真なら自動番号付しない（カウントもしない）
pstyle	追加の段落スタイル（標準設定を元に一時変更）
tstyle	追加のテキストスタイル（標準設定を元に一時変更）
alt	マクロ名（このマクロを呼び出した結果を見出しとする）
header	ヘッダ文字列（一時変更）
headerall	ヘッダ文字列（以降の全ページ）
headereven	ヘッダ文字列（以降の偶数ページ）
headerodd	ヘッダ文字列（以降の奇数ページ）
footer	フッタ文字列（一時変更）
footerall	フッタ文字列（以降の全ページ）
footereven	フッタ文字列（以降の偶数ページ）
footerodd	フッタ文字列（以降の奇数ページ）
<HXNUMBER>	
level	対象の H* の番号

number	番号文字列 ({H*} が何番目の H* かを表す)
<HXOUTLINE>	
level	アウトラインにする番号をカンマ区切りで (省略すると 2,3)

alt を指定するとそのマクロには見出しテキストオブジェクト (アウトラインや番号付けの処理がされた) が text 属性を通じて渡される

H* の header* や footer* に '*' を指定すると内容のテキストと同じ

pstyle 属性、tstyle 属性を指定することで、stddefs.inc での標準設定を変更できます。例えば、<H3 tstyle="withline:1"> なんとら </H3> とすれば「なんとら」という中見出しに下線を付けることができます。この一回だけでなく、以後の <H3> にすべて下線を付けたい場合は、<H3 tstyle="withline:1"/> とします。このように内容のない <H*> は、それ自体は見出しを作らず、pstyle、tstyle、font、alt の設定を保存しておいて以後その設定を使うように指示する働きをします。

alt 属性を使うと、ユーザーが定義した独自マクロによってより凝ったレイアウトの見出しが実現できますが、ここでは省略します。

3.2.3 通常の段落

通常の段落は<P> で囲みます。<P> の内容のテキストでは HTML と違って、改行が改行として生きます。ただし、タグの前後の改行は無視されます。これは<P>に限らず多くのタグに共通する XPDFJ の特徴です。また、
 で改行を入れることもできます。次の三つの例はすべて同じ結果になります。

```
<P> なんとら <BR/> かんたら </P>
```

```
<P> なんとら
かんたら </P>
```

```
<P>
なんとら
かんたら
</P>
```

また、<P> などの内容のテキストでは行頭の半角スペースは無視されます。これによって、次のように書いてタグの入れ子関係をわかりやすくすることができます。その代わり、半角スペースで行を字下げすることはできません。簡易的に字下げしたい場合は全角スペースを使ってください。長い行を折り返しても行頭が揃うように字下げしたい場合は、段落スタイル beginindent を使います。

```
<P>
  なんとら
  かんたら
</P>
```

3.2.4 文字の修飾

文字の修飾のうち、太字、斜体、下線、上付き、下付き、についてはそれぞれ<I><U><SUP><SUB>でできますが、PDFJ には他にもいくつかの文字修飾の方法が用意されており、それは<S> の属性によってテキストスタイルを指定することでおこないます (<S> は HTML では取消線ですが XPDFJ ではテキストスタイルを与えるのに使います)。属性は複数指定することができます。主なテキストスタイルとそこで使われる図形スタイルは次のようになります。

【表 : テキストスタイルの主なもの】

スタイル名	値と機能
fontsize	フォントサイズ

italic	真なら斜体
bold	真なら太字
ruby	ルビ文字列
shapestyle	文字描画の図形スタイル
withline	真なら下線あり
withlinestyle	下線の図形スタイル
withbox	囲み枠。s: 枠線、f: 塗りつぶし、sf: 枠線 + 塗りつぶし s の代わりに l: 左辺、r: 右辺、t: 上辺、b: 下辺 の組み合わせも可 rX: (X は数値) 角を半径 X で丸くする
withboxstyle	囲み枠の図形スタイル
withdot	真なら圈点あり (圈点とは文字の上に点を付けて強調するもの)

図形スタイルを省略すると枠線色塗りつぶし色とも黒、線幅 1、実線

【表 : 図形スタイル】

スタイル名	値と機能
fillcolor	塗りつぶし色
strokecolor	線色
linewidth	線幅
linedash	破線パターン「オン幅, オフ幅, 開始オフセット」

色は「#RRGGBB」(16 進) または 0 (黒) から 1 (白) の間の数値

破線パターンを指定しなければ実線

破線パターンの開始オフセットは省略可

表だけではわかりにくいものもありますので、実例をいくつか挙げておきましょう。属性の値でスタイルを指定するときは、「スタイル名: 値」を「;」区切りで並べます。なお、フォントを切り替える方法は後述します。

- ・ 文字を赤くする
<S shapestyle="fillcolor:#FF0000"> 赤文字 </S>
- ・ 灰色で網掛けして白文字に
<S shapestyle="fillcolor:1" withbox="f" withboxstyle="fillcolor:0.5"> 白抜き </S>
- ・ 点線の下線
<S withline="1" withlinestyle="linedash:2,2"> 点線下線 </S>
- ・ ルビ
<S ruby=" あじさい "> 紫陽花 </S>

3.2.5 段落の体裁

<P> の属性によって段落の体裁を指定できます。次の表に属性を示します。いくつか具体例を挙げておきます。

【表 : <P> の属性】

属性名	値と機能
pstyle	追加の段落スタイル (文書の本文設定を元に一時的に変更)
tstyle	追加のテキストスタイル (文書の本文設定を元に一時的に変更)
align	b,left:左寄せ、m,center:中央寄せ、e,right:右寄せ、w:両端寄せ
font	フォントの一時変更
foreachline	真なら内容の各行をそれぞれ段落とする

align 属性の指定は pstyle 属性で align スタイルを指定するのと同じ

font 属性については後述のフォントの項を参照

【表：段落スタイルの主なもの】

スタイル名	値と機能
size	段落の行方向のサイズ
align	揃え (b: 行頭揃え m: 中央揃え e: 行末揃え w: 両端揃え W: 強制両端揃え)
linefeed	行送り
preskip	段落前の間隔
postskip	段落後の間隔
beginindent	行頭インデント
endindent	行末インデント
nobreak	真なら分割されない
postnobreak	真ならその後ろで分割されない
typename	タイプ名

- ・ 中央寄せ
`<P align="center">...</P>`
- ・ フォントサイズを 8 ポイントにして 20 ポイント下げる
`<P pstyle="beginindent:20" tstyle="fontsize:8">...</P>`
- ・ 右寄せで行間を詰め、段落前後の間隔を広く
`<P pstyle="align:e; linefeed=120%; preskip:10; postskip:10">...</P>`

3.2.6 フォント

stddefs.inc を読み込むと自動的に stdfontsH.inc という横書用フォント定義をしたマクロファイルも読み込まれます。その中では次のようなフォントオブジェクトが変数にセットされていて、`<P><DD><TABLE><TR><TD>` の属性で `font="$FontH{gothic}"` のように指定することができます。

【表：stdfontsH.inc のセットするフォント】

変数名	セットされたフォントオブジェクトの内容
\$FontH{mincho}	和文欧文とも Ryumin-Light (欧文は半角固定ピッチ) italic 指定 Ryumin-Light を傾ける bold 指定 GothicBBB-Medium italic + bold 指定 GothicBBB-Medium を傾ける
\$FontH{mincho_times}	和文 Ryumin-Light、欧文 Times-Roman italic 指定 和文は Ryumin-Light を傾ける、欧文 Times-Italic bold 指定 和文 GothicBBB-Medium、欧文 Times-Bold italic + bold 指定 和文は GothicBBB-Medium を傾ける、欧文 Times-BoldItalic
\$FontH{gothic}	和文欧文とも GothicBBB-Medium (欧文は半角固定ピッチ) italic 指定 GothicBBB-Medium を傾ける bold 指定 GothicBBB-Medium italic + bold 指定 GothicBBB-Medium を傾ける

\$FontH{gothic_helvetica}	和文 GothicBBB-Medium、欧文 Helvetica italic 指定 和文は GothicBBB-Medium を傾ける、欧文 Helvetica-Oblique bold 指定 和文 GothicBBB-Medium、欧文 Helvetica-Bold italic + bold 指定 和文は GothicBBB-Medium を傾ける、欧文 Helvetica-BoldOblique
---------------------------	--

表からわかるように、\$FontH{gothic} に対して bold 指定しても効果がありません。PDFJ には任意の TrueType フォントを使用する機能があり、これを使えばより太いゴシックにするようなことも可能ですが、ここでは省略します。

3.2.7 箇条書き

HTML と同様、 で記号付き箇条書きが、 で番号付き箇条書きが、<DL><DT><DD> で語句説明箇条書きができます。ただし、属性の指定がかなり違います。次に属性の一覧を示します。なお、入れ子にしてもレベルによって UL のラベルの記号が自動的に変わるような機能はなく、label 属性で自分でラベルを指定する必要があります。

【表：箇条書きの属性】

属性名	値と機能
	
label	ラベル文字列（省略すると「・」）
labelsiz	二行目以降の字下げ（省略すると本文フォントサイズに同じ）
pstyle	追加の段落スタイル（文書の本文設定を元に一時的に変更）
indent	全体の字下げ
	
label	ラベル初期値（半角英数字、省略すると「1」） 「-」を指定すると前回の から引き継ぐ
prelabel	ラベルの前置文字列
postlabel	ラベルの後置文字列（省略すると「.」）
labelsiz	二行目以降の字下げ（省略すると本文フォントサイズに同じ）
indent	全体の字下げ
pstyle	追加の段落スタイル（文書の本文設定を元に一時的に変更）
<DL>	
labelsiz	二行目以降の字下げ（省略すると本文フォントサイズの倍）
labelskip	語句と本体の間隔（省略すると本文フォントサイズに同じ）
indent	全体の字下げ
pstyle	追加の段落スタイル（文書の本文設定を元に一時的に変更）
	
tstyle	追加のテキストスタイル（文書の本文設定を元に一時的に変更）
font	フォントの一時変更
foreachline	真なら内容の各行をそれぞれ段落とする
<DD>	
tstyle	追加のテキストスタイル（文書の本文設定を元に一時的に変更）
font	フォントの一時変更

注意していただきたいのは、箇条書きを入れ子にするときは、 や <DD> の中に次の箇条書きを入れるのではなく、同列に入れるということです。<P> も や <DD> と同列に入れると や

<DD> の二行目以降と同じ字下げが適用されます。例えば次のようになります。

```
<OL>
  <LI> 外側の箇条書き 1 </LI>
  <P> 外側の箇条書き 1 の続き </P>
  <OL label="a">
    <LI> 内側の箇条書き 1 </LI>
    <LI> 内側の箇条書き 2 </LI>
  </OL>
  <LI> 外側の箇条書き 2 </LI>
</OL>
```

これを処理すると、次のようになります。今のところ、HTML とは違って箇条書きを入れ子にしてもその深さに応じて自動的にラベルの記号や番号が変化したりはしません。label 属性で一々指定してやる必要があります。

1. 外側の箇条書き 1
 - 外側の箇条書き 1 の続き
 - a. 内側の箇条書き 1
 - b. 内側の箇条書き 2
2. 外側の箇条書き 2

箇条書きの時に一行ずつに と を書くのは面倒です。<LI foreachline="1"> とすると、その内容は一行ずつが で囲まれているものと見なされます。これは便利なので、代わりに <MLI> と書けるようになっていました。例えば次のように書くと「なんとら」「かんとら」がそれぞれ箇条書きになります。

```
<OL><MLI>
  なんとら
  かんとら
</MLI></OL>
```

3.2.8 表

表の作成も HTML と同様に <TABLE><TR><TH><TD> でできますが、これも属性はかなり違います。次に属性を示します。

【表：表の属性】

属性名	値と機能
<TABLE>	
cellwidth	セル幅（カンマ区切りで列毎に指定、省略すると 200）
cellpadding	セル内余白
cellbox	セル枠線。s: 枠線、f: 塗りつぶし、sf: 枠線 + 塗りつぶし s の代わりに l: 左辺、r: 右辺、t: 上辺、b: 下辺 の組み合わせも可 rX: (X は数値) 角を半径 X で丸くする (カンマ区切りで列毎に指定)
cellboxstyle	セル枠線の図形スタイル（省略すると「linewidth:0.25」）
align	セル内配置。left,center,right (カンマ区切りで列毎に指定)
tablebox	表枠線。指定方法は cellbox と同じ
tableboxstyle	表枠線の図形スタイル（省略すると「linewidth:0.25」）
pstyle	追加の段落スタイル（TH や TD に渡される）
tstyle	追加のテキストスタイル（TH や TD に渡される）
font	フォントの一時変更（TH や TD に渡される）

preskip	表の前間隔
postskip	表の後間隔
indent	表の行頭側のインデント
postnobreak	真なら表の後にページ分割しない
nobreak	真ならページ分割しない
<TR>	
cellpadding	(値と機能は TABLE と同じ)
cellbox	"
cellboxstyle	"
pstyle	"
tstyle	"
font	"
postnobreak	真ならこの行の後にページ分割しない
<TH><TD>	
align	セル内配置。left,center,right。(TH では自動的に center)
cellpadding	(値と機能は TABLE と同じ)
cellbox	"
cellboxstyle	"
pstyle	追加の段落スタイル (文書の本文設定を元に一時的に変更)
tstyle	追加のテキストスタイル (文書の本文設定を元に一時的に変更)
font	フォントの一時変更
colspan	セルがまたがる列数
rowspan	セルがまたがる行数

カンマ区切りで列毎に指定するものが列数より少なければ最後の要素を繰り返し使う

HTML ともっとも異なる点は、内容に応じてセル幅を自動調節する機能はなく、cellwidth 属性ですべてのセル幅 (列幅) を指定する必要があることです。(ポイントでの指定の他、現在の行長に対する % での指定も可能です。)指定した個数以上の列がある場合は、指定した最後の値が繰り返し使われます。具体例を一つ示します。

```
<TABLE cellwidth="80, 50" cellpadding="3" cellbox="," align="center, right"
  tablebox="sr5" tableboxstyle="linewidth:1">
  <TR cellbox="b, lb">
    <TH> 年月 </TH><TH>A</TH><TH>B</TH>
  </TR>
  <TR>
    <TD>2004 年 1 月 </TD><TD>15,000</TD><TD>25,000</TD>
  </TR>
  <TR>
    <TD>2004 年 2 月 </TD><TD>12,000</TD><TD>21,000</TD>
  </TR>
  <TR>
    <TD>2004 年 3 月 </TD><TD>18,000</TD><TD>31,000</TD>
  </TR>
</TABLE>
```

これを処理すると次のようになります。

年月	A	B
2004 年 1 月	15,000	25,000
2004 年 2 月	12,000	21,000
2004 年 3 月	18,000	31,000

<TH> や <TD> の中にはこの例のようにテキストを直接書くこともできますが、<P><DL><TABLE> といったブロック要素の命令を書くことができます。テキストを直接書いた場合は<P>...</P>で囲まれているものと見なされるわけです。テキストを直接書けるという点を除けば、<TH> や <TD> の中は小さな <BODY> だと考えることができます。これにより、複雑なページレイアウトを <TABLE> を使って実現することが可能です。(HTML ではよく使われる手法ですね。)

3.2.9 その他の本文用マクロ

stddefs.inc のその他のマクロとしては、横罫線を引く<HR>、画像を挿入する、ブロックを作ったり囲み枠を付けたり配置を調整したりする<DIV>、間隔を調整する<SKIP>、改ページする<NEWPAGE> といったものがあります。これらの属性を次に示しておきます。(<NEWPAGE> には属性はなく、<NEWPAGE/> と書くだけです。)

【表：その他のマクロの属性】

属性名	値と機能
<HR>	
width	横幅 (省略すると「100%」) (% 指定すると現在の行長に対する割合)
size	線幅 (省略すると 0.25)
color	色 (省略すると黒)
align	left,center,right のいずれか (省略すると center)
preskip	前間隔
postskip	後間隔
	
src	JPEG ファイル名または URL
pxwidth	横ピクセルサイズ
pxheight	縦ピクセルサイズ
width	表示横サイズ
height	表示縦サイズ
padding	周囲余白
colorspace	rgb,cmyk,gray のいずれか (省略すると rgb)
<DIV>	
width	幅 (省略すると現在の行長)
align	left,center,right のいずれか
padding	周囲余白 (内容と囲み枠の間の余白)
withbox	囲み枠。s: 枠線、f: 塗りつぶし、sf: 枠線 + 塗りつぶし s の代わりに l: 左辺、r: 右辺、t: 上辺、b: 下辺 の組み合わせも可 rX: (X は数値) 角を半径 X で丸くする
withboxstyle	囲み枠の図形スタイル
preskip	前間隔 (省略すると先頭要素の前間隔)
postskip	後間隔 (省略すると末尾要素の後間隔)
<SKIP>	
skip	間隔

3.3 ヘッダとフッタ

ページのヘッダとフッタは、<HEADER><FOOTER> で指定します。そのページのみ、以後の全ページ、以後の奇数ページ、以後の偶数ページ、という指定方法が用意されています。<HEADER><FOOTER> の属性は次の通りです。page="this" を指定したときは <BODY> 内に置かないと効果がありません。それ以外の場合は <BODY> の外に置いて構いません。

【表：<HEADER><FOOTER> の属性】

属性名	値と機能
page	対象ページ this: そのページのみ、all: すべて、even: 偶数ページ、odd: 奇数ページ（省略すると all）
hr	横罫線の指定。位置（t: 上、b: 下） 文字との間隔（省略すると 3） 線幅（省略すると 0.25）をカンマ区切りで
pstyle	追加の段落スタイル（文書の本文設定を元に一時的に変更）
tstyle	追加のテキストスタイル（文書の本文設定を元に一時的に変更）
font	フォントの一時変更
clear	真を指定すると対象ページの指定をクリア

clear は page が this の時は無意味

また、<H*> の header* や footer* 属性でも指定できます。ただしこの場合はヘッダやフッタの文字列が指定できるだけで、段落スタイルやテキストスタイルは指定できません。

例えば、フッタには中央寄せでページ番号を表示し、偶数ページヘッダには <H2> の内容を左寄せ、奇数ページヘッダには <H3> の内容を右寄せで表示したいとすると、次のようにします。

```
<HEADER page="even" pstyle="align:b"/>
<HEADER page="odd" pstyle="align:e"/>
<FOOTER page="all" pstyle="align:m">{page}</FOOTER>
<BODY>
...
<H2 headereven="*" headerodd=" "> なんとら </H2>
...
<H3 headerodd="*"> かんたら </H3>
...
</BODY>
```

ヘッダやフッタで {page} を使ったとき表示されるページ番号は、その <BODY> がページに配置される時点でのページ番号になります。その後でそのページより前にページが挿入されても、ヘッダやフッタで表示したページ番号は変わりませんので注意してください。

3.4 索引

索引を作りたい場合は、<do file="index.inc"/> で索引用マクロを読み込み、本文中の各索引語のところに<INDEX reading="読み" term="検索語">と書いておいて、本文の後で<MAKEINDEX/>と書きます。読みには半角英数記号とひらがなカタカナが使えます。読みが検索語と同じ場合は reading は省略できます。

<MAKEINDEX/> の属性は次の通り。

【表：<MAKEINDEX> の属性】

属性名	値と機能
beginindent	ページ番号群の行頭インデント（省略時値 50）
endindent	索引語の行末インデント（省略時値 50）

3. 原稿の書き方

space	索引語とページ番号群の間の最小空白サイズ（省略時値 20）
dot	索引語とページ番号群の間を埋める文字列（省略時値 " "）
title	索引語頭文字を表示する文字列（{title} は頭文字に置換）（省略時値 " - {title} - "）
defaulttitle	英字仮名以外に対する索引語頭文字名（省略時値 " 記号・数字 "）

索引は用紙が A4 なら 3 段組、B5 なら 2 段組にするのが適当でしょう。そのためには、<BODY cols="3"> などとした <BODY> で囲みます。その場合、索引用のヘッダやフッタの指定は <BODY> の前で指定しておかないと中央寄せなどの行配置がうまくいかないので注意してください。例えば次のようになります。

```
<HEADER page="even" pstyle="align:b" hr="b"> 索引 </HEADER>
<HEADER page="odd" pstyle="align:e" hr="b"> 索引 </HEADER>
<BODY cols="3">
  <H2> 索引 </H2>
  <MAKEINDEX/>
</BODY>
```

3.5 目次

目次を作りたい場合は、<do file="toc.inc"/> で目次用マクロを読み込み、本文の後に <BODY page="1">...<TOC/>...</BODY> と書いて 1 ページから挿入します。目次の前後には文書全体の表題など必要なものを入れます。<TOC> の属性は次の通りです。

【表：<TOC> の属性】

属性名	値と機能
levelindent	レベル毎のインデント量（省略時値 10）
endindent	表題の行末インデント（省略時値 60）
space	表題とページ番号の間の最小空白サイズ（省略時値 30）
dot	表題とページ番号の間を埋める文字列（省略時値 " "）

4. 簡単なマクロ定義

あらかじめ用意されたマクロを使うだけでなく、自分でマクロを定義することもできます。ここでは簡単なマクロ定義の方法を二つ紹介します。もっと複雑なことをやりたい場合は XPDFJ の説明書をご覧ください。

まず、ある命令の特定の属性を頻繁に使う場合、それをマクロ化する方法です。<alias> を使います。例えば文字を赤くする

```
<S shapestyle="fillcolor:#FF0000">...</S>
```

を文書の中で何度も使うようなら、

```
<alias tag="RED" aliasof="S" shapestyle="fillcolor:#FF0000"/>
```

と書いてマクロ RED を定義しておけば、それ以降は <RED>...</RED> と書くことができます。マクロの名前は、英字かアンダースコアで始め、英字、アンダースコア、ピリオド、ハイフンが使えます。何文字でも構いません。PDFJ のサブルーチンやメソッドの名前とぶつかるのを防ぐために、英小文字は使わないようにしてください。

次の方法は、命令の固まりをまとめてマクロ化するものです。<def> を使います。例えば、自分の所属と名前を右寄せで書く、

```
<P align="right">
  (株) ネットストック
  中島 靖
  mailto:nakajima@netstock.co.jp
</P>
```

という段落をどの文書でも使うのであれば、次のような内容の mymacro.inc ファイルを作っておきます。

```
<?xml version="1.0" encoding="x-sjis-cp932"?>
<do>
<def tag="MYSIGN">
  <return>
    <P align="right">
      (株) ネットストック
      中島 靖
      mailto:nakajima@netstock.co.jp
    </P>
  </return>
</def>
</do>
```

原稿と同様、XML 文書なので先頭の <?xml...?> は必須です。全体を <do>...</do> で囲んでいるのはマクロファイルには必要なおまじないだと思ってください。

マクロファイルの中には <def> や <alias> をいくつでも並べることができますので、先ほどの <RED> の定義を入れてしまっても構いません。

これで、<do file="stddefs.inc"/> の後に <do file="mymacro.inc"/> と書けば、<MYSIGN/> と書くことで所属と名前を入れられるわけです。

索引

		- N -		- け -	
		<NEWPAGE>	12	圈点	7
		- O -		- こ -	
			9	小見出し	5
				コメント	3
		- P -		語句説明箇条書き	9
		<P>	6		
		- S -		- さ -	
		<S>	6	索引	13
		<SKIP>	12	- し -	
		<SUB>	6	下付き	6
		<SUP>	6	シフト JIS	2
				斜体	6, 7
		- T -		終了タグ	2
		<TABLE>	10	字下げ	6
		<TD>	10		
		<TH>	10	- す -	
		<TOC>	14	図形スタイル	7
		<TR>	10	- せ -	
		- U -		セル内余白	10
		<U>	6	セル幅	10
			9	セル枠線	10
				線色	7
		- う -		線幅	7
		上付き	6	- そ -	
				属性	3
		- お -		- た -	
		大見出し	5	タグ	2
		- か -		段間	4
		開始タグ	2	段組	4
		改ページ	12	段落	6
		囲み枠	7	段落スタイル	8
		箇条書き	9		
		- I -			
		<I>	6		
			12		
		<INDEX>	13	- ち -	
				中央寄せ	7
				中見出し	5
		- L -			
			9	- き -	
				記号付き箇条書き	9
				行送り	4
		- M -		行間	4
		<MAKEINDEX>	13	- て -	
		<MLI>	10	テキストスタイル	6

- に -		- ら -	
日本語 EUC	2	ラベル	9
- ぬ -		- り -	
塗りつぶし色	7	両端寄せ	7
- は -		- る -	
破線	7	ルビ	7
半角固定ピッチ	8		
番号付き箇条書き	9		
- ひ -			
左寄せ	7		
表	10		
- ふ -			
フォント	8		
フォントサイズ	4, 6		
フッタ	13		
太字	6, 7		
ブロック	12		
文書の表題	5		
- へ -			
ヘッダ	13		
ページ高さ	4		
ページの挿入	4		
ページ幅	4		
- ほ -			
ポイント	4		
- ま -			
マクロ定義	15		
- み -			
右寄せ	7		
見出し	5		
- も -			
目次	14		
文字の修飾	6		
- よ -			
横罫線	12		