マイクロコンピュータと
ワークステーション 54－5
（1989. 2. 21）

Prologによる標準ページ記述言語SPDLの早期試作

－1バイト文字も2バイト文字も，共通文字とする試み－

若鳥陸夫

日本ユニシス株式会社

プログラム言語の日本語機能の水準として，ローマ文字符号に加えJIS漢字符号の英数字等を同等に構文解析する水準を想定し，プログラム言語の字句解析系・構文解析系並びに部分的な実行部を作成してその実用性を確認した。漢字符号による英数字の変換表は，外部ファイルで定義し，その支持の如何をファイルの有無とファイルの内容とで自動選択できるようにして，言語処理系の国際流通性を損ねない方式とした。言語の一例として，SPDLを対象としてみた。

標準ページ記述言語SPDLは国際標準化機構のJTC 1/SC18で，機械対機械界面の言語として，開発中であり，1989年1月現在，第4次作業案(WD)から規格草案(DP)への移行期にある。

Rapid Proto-Typing of SPDL processor in Prolog

- A generalized 1 byte & 2 byte character hundling -

Rikuo WAKATORI

Nihon Unisys Limited
2-17-51, Akasaka, Minato-ku, Tokyo, 107
Japan

The author has tried to improve the multi-byte code(e.g. Chinese, Japanese, Korean) capability in parsing of SPDL, as example.
This paper includes a proposal for language to include multi byte code parser using external code conversion table and an including mechanism which can be applied to both its international standards and international processors.
SPDL(Standard Page Description Language) is a standardized machine to machine language which is developing by ISO/IEC JTC 1/SC18.
Now the stage is from 4th working draft to 1st draft proposal.
This article was written in Japanese.

1. 序(introduction)

1.1 SPDLの概要(introduction of SPDL)

標準ページ記述言語SPDL[1]は，割付け処理後の文書の機械表現の一つであり，書体表現，絵の表現，字間調整，文字進行方向などを記述できる言語である。その言語の記述法は，逆ポーランド記述法(倒置法)を採用し，「引き数」を先に記述し，その後で「演算子」等を記述する。その言語の母型は，米国Adobe社のPostScript言語[3]で，現在の作業案を見る限りでは国際規格版PostScriptと言える位に良く近似している。しかし，企業規格水準から国際規格とする工程で基本概念を変更しているが，少々の記述追加により各ページ記述言語間の変換性を保持できることを身上としている。その主な違いとしては，次のことがある。

(1) 座標系の省略時想定値(default)をミリメートルとする。PostScriptでは，1ポイントが省略時想定値である。
(2) 書体や字体の指示方法の国際整合
従来のページ記述言語の書体の指示方法や名前付けは，企業規格のまま，整合性のない状態であったが，国際規格案9541を基本とした。
(3) クライエント及びサーバーモデルを印刷要求。

1.2 ページ記述言語の日本語機能への要件

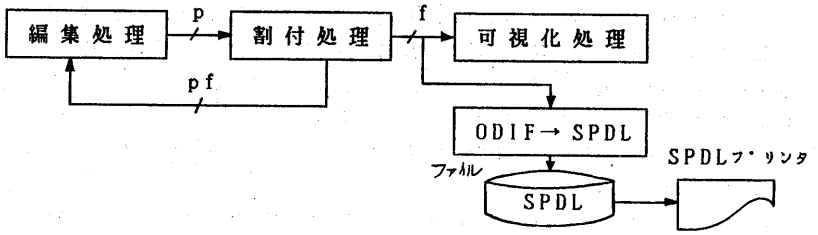
標準ページ記述言語では，日本語(ここでは漢字符号系を使うことを意味し，対象言語は英語・露語・仏語・独語・計算機言語など何でもできるものを指す)に印字修飾をする場合には，次の機能が備わっている必要がある。

(1) 2バイト文字列を操作できること。
例：(中国語) show ⇒ 中国語
(Nihon = 日本) show ⇒ Nihon = 日本
(2) 日本語書体を指定できること。
a) ISO 9541に基づいた指定ができること。
b) 登録された書体を対象とする。
(3) 漢字符号系の英数字も同一文字として扱えること。
例：(韓国語) SHOW ⇒ 韓国語
(日本語) show ⇒ 日本語
備考：第一行のshowは，漢字符号系による点に注意。
(4) 異言語文字との均衡を良く指定できること。
例：設計基準線の相違の調整

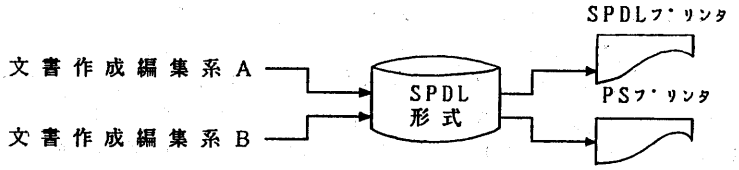
1.3 ページ記述語と文書交換系などの関係

ページ記述言語は，図１のようなさまざまな界面で利用できる。

a) 事務文書体系(ODA)の可視化界面規格として利用
- Imaging interface for ODA -



b) 文書作成編集系の共通界面
- Printer interface -



c) 複数プログラム界面としての利用
- Graphic program interface -

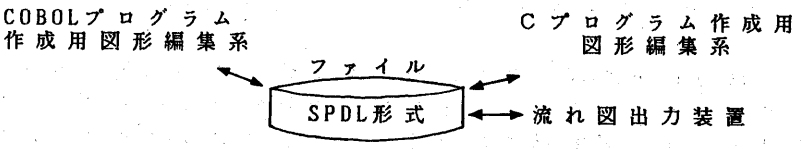


図1. 標準ページ記述言語の利用
(Figure 1. Variety of SPDL usage)

1.4 SPDL処理系試作の目的(Purpose of Study)

筆者のSPDL早期試作の目的は，自社商品としてSPDL処理系を準備とするためではない。むしろ，この実験の目的は，さまざまな国際規格への２バイト符号の取扱いを反映し，２バイト符号を取扱う言語処理系が国際的に流通する基盤の調査の方に重きを置いている。
その中でも，今回の早期試作では，本稿1.2節3)で述べたような，「漢字符号系の英数字も，１バイト系の英数字と同一文字として取扱う」ことが，国際的な言語処理系として存在しえる路を模索するのが目的である。また副目的として，英語MS-DOS上で動くTurbo-Prolog[2]を購入したので，その負荷試験も兼ね，第三の目的は，試作品が他目的の道具としても使えることであった。

2. 漢字符号系の英数字を同一文字とする概念
(Concepts on two byte character hundling)

2.1 文字寸法の違いは，言語に無影響と考える。

在来の計算機の文字表現は，概して2種類の体系を混用し，小さい方の寸法にだけ言語解析系が感じるように処理系を作成してきた。2バイト文字の英数は，1バイトの英数字とは別な字であるとして取扱ってきた応用例が多い。しかし，在来の印字機構では，暗黙的に両者の出力文字寸法に違いがあったから，それでも実用になったが，ページ記述言語等で処理する対象の文書では，文字寸法を自由に指定し，書体も指定できるので，「意味上，同一の文字(文字概念 が同一)ならば，寸法や書体が異なっていても同一の文字と考える」ことが必要になってきている。

そこで，あるポイントの半角(normal width)と全角(double width)も，同一の文字と考える処理系を試みた。

2.2 文字符号長の違いがあっても1文字は1文字と考える。

1バイト及び2バイト符号で表現された同一の意味の文字は，言語処理系では同一として取扱うことを意味する。符号系の違いで，文字寸法を暗黙的に変化させている実現も多いが，長期的に文字寸法可変にできる金物から，この暗黙的使用が破れ始めている。このページ記述言語あたりから，1バイト文字も2バイト文字も同一に処理するように正していく必要が強くなっている。

2.3 2バイト対1バイト変換表を外部ファイルとする。

国際間で，言語処理系を同一にし，かつある特定の言語向きの符号表を任意選択で組み込めるようにする方法であれば，国際間の合意を得られ易い。しかも，その共用手法は国際規格に記述されるなど開放的でなければ，2バイト符号の存在すら無知の欧米のプログラマ達が漢字符号系を含む言語処理系を書きはしない。

そこで，本稿では，国際共通性を維持する方法として，次の方式を提案する。

(1) 2バイト対1バイト符号対対応表を外部ファイルとし，その有無により2バイト処理か否かを判定し，必要なファイルを読み処理系を設定する。

(2) 符号表の呼び出し先を表に含む。
1バイト符号及び2バイト符号の識別方法を表の先頭に付け，その記述により処理系が設定変更を行う。国際規格では，符号系の指示シーケンスを入れておけば整合性が良いが，簡単化のために左を1バイト，右を2バイトとした。つまり，1バイトローマ字が，符号表の列2－列7，2バイトローマ字が列10－列15にあることを明示するために，1，2とした。つまり，ASCIIをG0に指示，JIS漢字をG1に指示し，それぞれGLとGRに呼び出しことを暗示する。もし，それらを解釈できない処理系であっても簡単に一行の文を読み捨てれば良いであろう。
なお，G1に指示したJIS漢字符号は，8ビット目が"1"であれば漢字と識別できる公開されたG1漢字方式である。

(3) 2バイト対1バイト符号対応表は，付録のようなものとし，右を原始番号，左を目的符号とする。

2.4 漢字符号表対1バイト変換表の特例

2バイト符号に対応する1バイト符号は，正確には1対1ではない。この取扱いは次のようにした(付録参照)。

(1) 二重引用符
漢字符号系では，左右を使い分けるが，1バイト符号では左右共用である。
ここでは，右二重引用符だけ使用することとした。
(2) 円記号対逆斜線
漢字符号の逆斜線は，ASCIIの逆斜線，JIS X 0202の円記号位置へ変換する。
(3) 負符号は，負記号へ変換し，ハイフンは無視する。

これらの扱いは，できたら日本国内で統一しておきたい。

2.5 1バイトでなく1文字読む関数を用意する。

符号変換表のバイト数指定により，1文字関数は，1バイトか2バイトの文字を判定し，その文字を構成するビット数(例えば 16ビット)を処理する。つまり，読み込みであれば，getlbyteではなくgetlcharとする。

2.6 2バイト指定のときだけ，符号変換表を参照する。

この扱いにより，1バイト系7ビット符号のときには，符号表を参照しないので，負担増が少ない。1バイト国に負担増を課しては，国際合意を得にくい。すなわち，列10－列15の符号のときだけ(つまり日本であれば漢字符号のときだけ)読み込み直後に符号変換表を参照し，処理系内では1バイト符号にしておく(図2)。

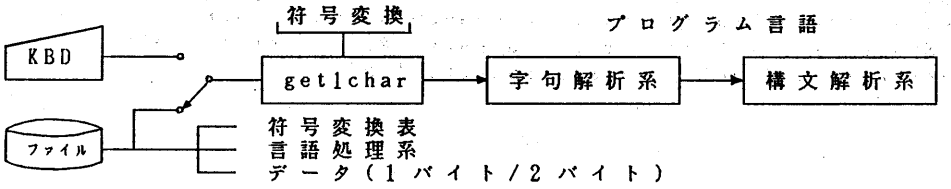


図2. 文字の読み込み部と他との関係
(Figure 1, Relationship between getlchar and others)

2.7 仮想スタック機械の記述法

スタックのPrologでの記述は，最も単純に文字列の並びとし，後入れ先出し(Last-in/First-out)機構を実現するため，スタックに積むとは，並びの頭に文字列を追加することとし，データベース(事実)に登録した。その他，状態変数も大域変数であるから，データベースとして扱った。この辺には報告するほどの手法は考案されていない。

3. 結果(Results of study)

このSPDLの試作に，漢字符号取扱い機能を付加して試験した結果，

(1) (当たり前なのだが)字句解析や構文解析系は，JIS漢字符号表の英字，間隔やかっこなどを文字として同様に取扱いできる。
(2) 符号変換表ファイルの有無により，解析系の自動設定が容易で，欧米人にも受容しえる水準にある。
(3) 1バイト符号のときの動作のときの負担は殆どない。
(4) getlcharは，思想的に統一されるので使い易い。利用者から文字を見たとき，その内部表現にまで関与しなくてもよい。他のプログラム言語も，「2バイト文字も文字として扱う」ということを貫いたらどうであろうか。

結果は，なかなか使える道具というわけにはいかないが，Prolog, PostScript及びSPDLの三言語を同時に使用する試作途上は，三言語の味の違いを楽しめた。

4. 付録(ANNEX):
符号変換表

参考文献：

(1) ISO ; "ISO/IEC JTC 1/SC18/WG8 N715Rev, Information Processing-Text Composition - Standard Page Description Language(SPDL)",1988
(2) Borland ; "Turbo-Prolog" user reference manual",1986
(3) Adobe System Inc ; "PostScript Language Reference Manual", Addison-Wesley, Reading, Massachusetts,1985

以上

- 付録 -

符号変換表　Example of Code Conversion Table

```
1,   2
 ,A1A1
!,A1AA
",A1C9
#,A1F4
$,A1F0
%,A1F3
&,A1F5
',A1AD
(,A1CA
),A1CE
*,A1F6
+,A1DC
,,A1A4
-,A1DD
.,A1A5
/,A1BF
0,A3B0
1,A3B1
2,A3B2
3,A3B3
4,A3B4
5,A3B5
6,A3B6
7,A3B7
8,A3B8
9,A3B9
:,A1A7
;,A1A8
<,A1E3
=,A1E1
>,A1E4
?,A1A9
@,A1F7
A,A3C1
B,A3C2
C,A3C3
D,A3C4
E,A3C5
F,A3C6
G,A3C7
H,A3C8
I,A3C9
J,A3CA
K,A3CB
L,A3CC
M,A3CD
N,A3CE
O,A3CF
P,A3D0
Q,A3D1
R,A3D2
S,A3D3
T,A3D4
U,A3D5
V,A3D6
W,A3D7
X,A3D8
Y,A3D9
Z,A3DA
[,A1CD
¥,A1C0
¥,A1EF
],A1CF
^,A1B0
_,A1B2
`,A1BE
a,A3E1
b,A3E2
c,A3E3
d,A3E4
e,A3E5
f,A3E6
g,A3E7
h,A3E8
i,A3E9
j,A3EA
k,A3EB
l,A3EC
m,A3ED
n,A3EE
o,A3EF
p,A3F0
q,A3F1
r,A3F2
s,A3F3
t,A3F4
u,A3F5
v,A3F6
w,A3F7
x,A3F8
y,A3F9
z,A3FA
{,A1D0
|,A1C3
},A1D
~,A1B
```