

ISO/IEC JTC1/SC2/WG2/IRG

Authors: Richard Cook and Tom Bishop

Date: 6 June 2008

IRGN 1449

L2/08-236

*文林 Wénlin Online CDL Tool***Summary**

A new online CDL tool is now available for IRG use, and documented here:

<<http://www.wenlin.com/cgi-bin/wenlinSVGhelp.pl>>

This tool renders CDL as SVG, and also converts IDS and free-form queries into CDL. This online tool was produced in collaboration with *Wenlin Institute Inc.* *<<http://www.wenlin.com/>>*, with funding from the U.S. *National Endowment for the Humanities* (see IRGN 1390).

Login Help

- Need a valid login and password? Please email the CDL team: *<cdl@wenlin.com>*.
- Did your session time out? Please log in again through the main SVG query portal.

Query Basics*Buttons*

The query form has two main buttons:

- CDL: Clicking the CDL button converts a valid non-CDL query to CDL, and displays the result.
- SVG: Clicking the SVG button displays a valid query as SVG.

Valid Queries

The query form generates SVG for valid queries in three different CJK description languages:

- CDL: Character Description Language (CDL) descriptions [XML; see the *CDL Specification*: *<<http://www.wenlin.com/cdl/>>*]
- CRS: Character Representation Sequences (CRS) [free-form input, using simple grid framework with 4 optional operators, for rapid CDL drafting; see below]
- IDS: Ideographic Description Sequences (IDS) [see the syntax description in TUS 5.0:428]

Valid queries in any of these three languages may be named (describing an encoded character at a specific Unicode codepoint) or anonymous (describing an unencoded character, or a character at an unknown codepoint).

Example Queries: CDL, CRS, IDS**1. CDL: Character Description Language**

- *named* CDL (cdl element with char and/or uni attributes), e.g. :

```
<cdl char='字' uni='5b57'>
  <comp char='宀' uni='5b80' points='0,0 128,40' />
  <comp char='子' uni='5b50' points='0,48 128,128' />
</cdl>
```

- *anonymous* CDL (cdl element without char and uni attributes), e.g.:

```
<cdl>
  <comp char='宀' uni='5b80' points='0,0 128,40' />
  <comp char='子' uni='5b50' points='0,48 128,128' />
</cdl>
```

Valid CDL queries include `comp` and/or `stroke` elements:

- Each CDL `comp` element must have a `char` and/or `uni` attribute, drawn from the set of characters currently having CDL descriptions. This set currently includes:

- the two *Radical* blocks (Kang Xi and Supplement);
- all *BMP CJK* characters (URO and Ext. A);
- most *Extension B* characters (SIP);
- all *CJK Strokes* block characters (bxg ⇒ ㄣ; d ⇒ 丶; h ⇒ 一; hg ⇒ 冫; hp ⇒ 丿; hpwg ⇒ ㇇; hxwg ⇒ 乙; hz ⇒ 冂; hzg ⇒ 冃; hzt ⇒ 册; hzw ⇒ 冎; hzww ⇒ 冏; hzz ⇒ 冑; hzzp ⇒ ㇈; hzzz ⇒ 冒; hzzzg ⇒ 冓; n ⇒ 丶; p ⇒ 丿; pd ⇒ 丿; pg ⇒ 丿; pz ⇒ 丿; q ⇒ 〇; s ⇒ 丨; sg ⇒ 丨; sp ⇒ 丨; st ⇒ 丨; sw ⇒ 丨; swg ⇒ 丨; swz ⇒ 丨; sz ⇒ 丨; szwg ⇒ 丨; szz ⇒ 丨; t ⇒ 丨; tn ⇒ 丨; wg ⇒ 丨; xg ⇒ 丨);
- many *BMP PUA CJK* characters;
- assorted *Compatibility* ideographs (might not get what you expect using these).

- Each CDL `stroke` element must have a `type` attribute, the value of which is written with an alphabetic stroke abbreviation as defined in the *CDL Specification*. Characters from the *CJK Strokes* block cannot currently be used as `type` attributes, and some few *CJK Strokes* block character names are not currently supported as CDL stroke abbreviations (this will change in the near future).

2. CRS: Character Representation Sequences

- *named* CRS (with leading CJK Ideograph followed by colon, optional span operators, and semi-colon or carriage-return row separator), e.g.:

In-line (no span operator):

字:宀;子

or Multi-line (no span operator):

宀:
公
儿

- *anonymous* CRS (with semi-colon or carriage-return row separator, and optional span operators), e.g.:

In-line (no span operator):

宀;子

Multi-line (no span operator):

宀
子

Optional *colspan* [▪ U+25a0, ● U+262f], *rowspan* [□ U+25a1, ● U+262f], and *gap* [U+3000 “Ideographic Space”] operators, e.g. :

□ ■ ■ ■ ■ ;
□ 宀 ■ ■ ;
□ 子 ■ ■ ;
□ □ ;
□ □ ;
□ ;

or

字 ■
宀 ● ■
□ □
□

- CRS aims at rapid CDL drafting of simple and complex character descriptions.
- All CRS operators are optional: a simple grid layout of components streamlines CDL drafting.
- CDL generated from CRS can be easily refined: users do not have to write XML from scratch.
- CRS descriptions are best edited in an environment in which operators and components have the same width, so that columns and rows line up.
- The *span* operator ● (U+262f) controls both *colspan* and *rowspan*; this permits “L”-type enclosure, as in the last example.
- CRS resolution (grid size) is a variable rectangle, determined by the user for the interpreter.
- If there is a span operator anywhere in the CRS, all row lengths are automatically regularized to max row length, filling right with *colspan* operator “▪” (U+25a0).
- Override auto-right-filling with the *gap* operator (U+3000 “Ideographic Space”), as in the next-to-last example above.

3. IDS: Ideographic Description Sequences

- *named* IDS (with leading CJK Ideograph, and following IDC operator [☐☐☐☐☐☐☐☐☐☐]), e.g. :

字☐宀子

- *anonymous* IDS (with leading IDC operator), e.g. :

☐宀子