ISO/IEC JTC 1/SC 2
Coded Character Sets
Secretariat: Japan (JISC)

DOC. TYPE		Liaison Organization Contribution
TITLE		Liaison statement from SC 34 to SC 2 regarding ISO/IEC FCD 19757-7, Information technology -- Document Schema Definition Languages (DSDL) -- Part 7: Character Repertoire Description Language (CRDL)
SOURCE		SC 34
PROJECT		
STATUS		This document is forwarded to WG 2 for consideration.
ACTION ID		FYI
DUE DATE		
DISTRIBUTION		P, O and L Members of ISO/IEC JTC 1/SC 2 ; ISO/IEC JTC 1 Secretariat; ISO/IEC ITTF
ACCESS LEVEL		Def
ISSUE NO.		299
FILE	**NAME**	02n3997.pdf
	SIZE (KB)	
	PAGES	19

Secretariat ISO/IEC JTC 1/SC 2 - IPSJ/ITSCJ (Information Processing Society of Japan/Information Technology Standards Commission of Japan)* Room 308-3, Kikai-Shinko-Kaikan Bldg., 3-5-8, Shiba-Koen, Minato-ku, Tokyo 105-0011 Japan *Standard Organization Accredited by JISC
Telephone: +81-3-3431-2808; Facsimile: +81-3-3431-6493; E-mail: kimura @ itscj.ipsj.or.jp

Liaison Statement from JTC 1/SC 34 to JTC 1/SC 2

SC 34 has developed ISO/IEC FCD 19757-7 (Character Repertoire Description Language). The goal of this FCD is to provide a machine-readable format for specifying which characters in ISO/IEC 10646 are permitted or not permitted for a given XML document.

Since this FCD strongly relates to ISO/IEC 10646, SC 34 cordially requests that SC 2 kindly provide comments on the FCD and helpful advice on further development of this project.

ISO/IEC JTC 1/SC 34

Date: 2008-01-11

ISO/IEC FCD 19757-7

ISO/IEC JTC 1/SC 34/WG 1

Secretariat: Japanese Industrial Standards Committee

Document Schema Definition Languages (DSDL) — Part 7: Character Repertoire Description Language

Document type: International Standard
Document subtype:
Document stage: (40) Enquiry
Document language: E

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

ISO/IEC 19757-7 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 34, Document Description and Processing Languages.

ISO/IEC 19757 consists of the following parts, under the general title *Document Schema Definition Languages (DSDL)*:

— *Part 1: Overview*

— *Part 2: Regular-grammar-based validation — RELAX NG*

— *Part 3: Rule-based validation — Schematron*

— *Part 4: Namespace-based validation dispatching language — NVDL*

— *Part 5: Datatypes*

— *Part 6: Path-based integrity constraints*

— *Part 7: Character repertoire description language — CREPDL*

— *Part 8: Declarative document manipulation*

— *Part 9: Datatype- and namespace-aware DTDs*

— *Part 10: Validation management*

Introduction

This International Standard defines a set of Document Schema Definition Languages (DSDL) that can be used to specify one or more validation processes performed against Extensible Markup Language (XML) documents. A number of validation technologies are standardized in DSDL to complement those already available as standards or from industry.

The main objective of this International Standard is to bring together different validation-related technologies to form a single extensible framework that allows technologies to work in series or in parallel to produce a single or a set of validation results. The extensibility of DSDL accommodates validation technologies not yet designed or specified.

This part of ISO/IEC 19757 provides a language for describing character repertoires. Descriptions in this language may be referenced from schemas. Furthermore, they may also be referenced from forms and stylesheets.

NOTE As the time of this writing, no schema languages provide mechanisms for referencing CREPDL schemas.

Descriptions of repertoires need not to be exact. To provide non-exact descriptions, this part of ISO/IEC 19757 provides kernels and hulls, which provide the lower limit and upper limits, respectively.

The structure of this part of ISO/IEC 19757 is as follows. Clause 5 introduces kernels and hulls of repertoires. Clause 6 describes the syntax of CREPDL schemas. Clause 7 describes the semantics of a correct CREPDL schema; the semantics specify when a character is contained by a repertoire described by a CREPDL schema. Clause 8 defines CREPDL validators and their behaviours. Clause 9 defines conformance of CREPDL processors. Finally, Annex A provides examples of the application of CREPDL.

Document Schema Definition Languages (DSDL) — Part 7: Character Repertoire Description Language

1 Scope

This part of the International Standard specifies a Character Repertoire Description Language (CREPDL). A CREPDL schema describes a character repertoire.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Each of the following documents has a unique identifier that is used to cite the document in the text. The unique identifier consists of the part of the reference up to the first comma.

RELAX NG, *ISO/IEC 19757-2, Document Schema Definition Languages (DSDL) — Part 2: Grammar-based validation — RELAX NG*

W3C XML, *Extensible Markup Language (XML) 1.0*, W3C Recommendation, available at http://www.w3.org/TR/xml

W3C XML-Names, *Namespaces in XML*, W3C Recommendation, available at http://www.w3.org/TR/xml-names

W3C XML Schema Part 2, *XML Schema Part 2: Datatypes*, W3C Recommendation, available at http://www.w3.org/TR/xmlschema-2/

IETF RFC 3986, *Uniform Resource Identifiers (URI): Generic Syntax*, Internet Standards Track Specification, January 2005, available at http://www.ietf.org/rfc/rfc3987.txt

IETF RFC 3987, *Internationalized Resource Identifiers (IRIs)*, Internet Standards Track Specification, January 2005, available at http://www.ietf.org/rfc/rfc3987.txt

ISO/IEC 10646, *Universal multiple-octet coded Character Set*

IANA Charsets, *IANA CHARACTER SETS*, available at http://www.iana.org/assignments/character-sets

Unicode, *The Unicode Standard*, The Unicode Consortium, available at http://www.unicode.org/

CLDR, *Unicode Common Locale Data Repository*, The Unicode Consortium, available at http://www.unicode.org/cldr/

3 Terms and definitions

For the purposes of this part of ISO/IEC 19757, the terms "character" and "repertoire" as defined in ISO/IEC 10646 and the following apply.

3.2
kernel
a set of characters that are guaranteed to be in the repertoire

3.3

hull
a set of characters that may be in the repertoire

4 Notation

in(*x*, *A*): character *x* is in repertoire *A*

not-in(*x*, *A*): character *x* is not in repertoire *A*

unknown(*x*, *A*): it is unknown whether character *x* is in repertoire *A*

5 Repertoire, kernel, and hull

This part of ISO/IEC 19757 can represent repertoires not containing C0 control functions with the exception of U+0009 (CHARACTER TABULATION), U+000A (LINE FEED), and U+000D (CARIIAGE RETURN).

NOTE 1 XML documents cannot contain C0 control functions other than U+0009 (CHARACTER TABULATION), U+000A (LINE FEED), and U+000D (CARIIAGE RETURN).

This part of ISO/IEC 19757 represents a repertoire by specifying a kernel and hull. A kernel contains characters that are guaranteed to be in the repertoire; the repertoire may contain other characters. A hull gives an outer boundary so that characters which are not in the hull are guaranteed not to be in the repertoire; some characters in the hull may actually be outside of the repertoire.

NOTE 2 Since characters may continue to be added to a repertoire, it may be impossible to specify a repertoire exactly. However, it is often possible to specify which character is absolutely included, and which character is absolutely excluded.

6 Syntax

6.1 General

An CREPDL schema in the full syntax shall be an XML document valid against the combination of the RELAX NG schema in 6.2 and the NVDL script in 6.3. Further constraints on the character content of the char, kernel or hull elements are shown in 6.4

6.2 RELAX NG schema

```
default namespace = "http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0"

start = coll

coll = union | intersection | difference |
      ref | repertoire | char

union = element union { commonAtts, coll+ }
intersection = element intersection { commonAtts, coll+ }
difference = element difference { commonAtts, coll+ }

ref = element ref {commonAtts, attribute href { xsd:anyURI }}
repertoire = element repertoire {commonAtts,  attribute registry { text }, (attribute name { text } | attribute number {xsd:int}  }

char =
   element char {
      commonAtts,
      (text |
         element kernel {commonAtts, text } |
         element hull {commonAtts, text } |
```

```
        (element kernel {commonAtts, text }, element hull {commonAtts, text }))
    }

commonAtts =
    attribute minUcsVersion {text}?,
    attribute maxUcsVersion {text}?
    #Note that xml:id is allowed, since any foreign attribute is implicitly allowed.
```

6.3 NVDL script

```
<?xml version="1.0" encoding="UTF-8"?>
<rules xmlns="http://purl.oclc.org/dsdl/nvdl/ns/structure/1.0">
   <namespace ns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
      <validate schema="crepdl.rnc" schemaType="application/relax-ng-compact-syntax">
         <mode>
            <anyNamespace match="elements">
               <allow/>
            </anyNamespace>
            <namespace ns="" match="attributes">
               <attach/>
            </namespace>
            <anyNamespace match="attributes">
               <allow/>
            </anyNamespace>
         </mode>
         <context path="char | kernel | hull">
            <mode>
               <anyNamespace match="elements">
                  <reject/>
               </anyNamespace>
               <namespace ns="" match="attributes">
                  <attach/>
               </namespace>
               <anyNamespace match="attributes">
                  <allow/>
               </anyNamespace>
            </mode>
         </context>
      </validate>
   </namespace>
</rules>
```

NOTE This NVDL script allows foreign attributes everywhere. It also allows foreign elements everywhere with the exception of char, kernel, and hull elements.

6.4 Character classes

The character content of a char, kernel or hull element is a regular expression that defines a set of characters. This regular expression matches charClass as specified in W3C XML Schema Part 2.

NOTE 1 While XQuery[4] slightly modifies the definition of regular expressions, charClass remains unchanged.

NOTE 2 The following rules are copied from W3C XML Schema Part 2. The semantics of [29] through [37] depend on the version of the Unicode standard.

```
[11] charClass  ::= charClassEsc | charClassExpr | WildcardEsc
[12] charClassExpr ::= '[' charGroup ']'
[13] charGroup ::= posCharGroup | negCharGroup | charClassSub
[14] posCharGroup ::= ( charRange | charClassEsc )+
[15] negCharGroup ::= '^' posCharGroup
[16] charClassSub ::= ( posCharGroup | negCharGroup )
```

```
                '-' charClassExpr
[17] charRange ::= seRange | XmlCharIncDash
[18] seRange ::= charOrEsc '-' charOrEsc
[20] charOrEsc ::= XmlChar | SingleCharEsc
[21] XmlChar ::= [^\#x2D#x5B#x5D]
[22] XmlCharIncDash ::= [^\#x5B#x5D]
[23] charClassEsc ::= ( SingleCharEsc | MultiCharEsc
                | catEsc | complEsc )
[24] SingleCharEsc ::= '\' [nrt\|.?*+(){}#x2D#x5B#x5D#x5E]
[25] catEsc ::= '\p{' charProp '}'
[26] complEsc ::= '\P{' charProp '}'
[27] charProp ::= IsCategory | IsBlock
[28] IsCategory ::= Letters | Marks | Numbers
            | Punctuation | Separators | Symbols | Others
[29] Letters ::= 'L' [ultmo]?
[30] Marks ::= 'M' [nce]?
[31] Numbers ::= 'N' [dlo]?
[32] Punctuation ::= 'P' [cdseifo]?
[33] Separators ::= 'Z' [slp]?
[34] Symbols ::= 'S' [mcko]?
[35] Others ::= 'C' [cfon]?
[36] IsBlock ::= 'Is' [a-zA-Z0-9#x2D]+
[37] MultiCharEsc ::= '\' [sSiIcCdDwW]
[37a] WildcardEsc  ::= '.'
```

7 Semantics

7.1 General

Let x be a character that is either outside the C0 area or one of U+0009 (CHARACTER TABULATION), U+000A (LINE FEED), and U+000D (CARIIAGE RETURN). Given a repertoire description *A*, either in(*x*, *A*), not-in(*x*, *A*), or unknown(*x*, *A*) holds.

7.2 char

The semantics of <char> ... </char> is defined below.

— Case 1: the content of the char element is text

  It is assumed that this element has a kernel element and a hull element whose contents are identical to that of this element. The rest is the same as in Case 4.

— Case 2: the char element has a kernel element but does not have a hull element.

  — in(*x*, <char> ... </char>) when *x* matches the regular expression specified as the content of the kernel element.

  — not-in(*x*, <char> ... </char>) never holds.

  — unknown(*x*, <char> ... </char>) when in(*x*, <char> ... </char>) does not hold.

— Case 3: the given char element has a hull element but does not have a kernel element.

  — in(*x*, <char> ... </char>) never holds

— not-in(*x*, <char> ... </char>) when *x* does not match the regular expression specified as the content of the hull element.

— unknown(*x*, <char> ... </char>) when not-in(*x*, <char> ... </char>) does not hold.

— Case 4: the given char element has a hull element and a kernel element.

— in(*x*, <char> ... </char>) when *x* matches the regular expression specified as the content of the kernel element.

— not-in(*x*, <char> ... </char>) when in(*x*, <char> ... </char>) does not hold and *x* does not match the regular expression specified as the content of the hull element.

— unknown(*x*, <char> ... </char>), otherwise.

The semantics of regular expressions depend on the version of the Unicode standard. The author of a CREPDL schema may specify the intended versions by specifying the minUcsVersion and maxUcsVersion attributes. If the CREPDL processor cannot use some version between these two, it should report an error and may stop normal processing.

EXAMPLE <char minUcsVersion="4.0" maxUcsVersion="4.0">\p{Nd}</char> represents the set of characters of the category "Nd" in Unicode Verion 4.0.

When a char element does not explicitly specify the minUcsVersion attribute, the nearest ancestor element having this attribute is searched. If it is found, its attribute value is used. If not found, there is no lower bound on Unicode versions. The same applies to maxUcsVersion.

7.3 union

A character is in <union>*A B* </union> if and only if it is in *A* or *B*. It is not in the union if and only if neither it is in *A* nor is it in *B*.

— in(*x*, <union>*A B*</union>) when in(*x*, *A*) or in(*x*, *B*).

— not-in(*x*, <union>*A B*</union>) when not-in(*x*, *A*) and not-in(*x*, *B*).

— unknown(*x*, <union>*A B*</union>), otherwise.

When a union element has one and only one child element, the semantics shall be the same as that of the child element. When a union element has more than two child elements, the semantics shall be the same as that of <union>*A B* </union> where *A* is the first child and *B* is the union of the other child elements.

7.4 intersection

A character is in <intersection>*A B* </intersection> if and only if it is in *A* and *B*. It is not in the intersection if and only if either it is not in *A* or it is not in *B*.

— in(*x*, <intersection>*A B*</intersection>) when in(*x*, *A*) and in(*x*, *B*).

— not-in(*x*, <intersection>*A B*</intersection>) when not-in(*x*, *A*) or not-in(*x*, *B*)

— unknown(*x*, <intersection>*A B*</intersection>), otherwise.

When an intersection element has one and only one child element, the semantics shall be the same as that of the child element. When an intersection element has more than two child elements, the semantics shall be the same as that of <intersection>*A B* </intersection> where *A* is the first child and *B* is the intersection of the other child elements.

7.5 difference

A character is in <difference> *A B* </difference> if and only if it is in *A* and it is not in *B*. It is not in the difference if and only if either it is not in *A* or it is in *B*.

— in(*x*, <difference>*A B*</difference>) when in(*x*, *A*) and not-in(*x*, *B*)

— not-in(*x*, <difference>*A B*</difference>) when not-in(*x*, *A*) or in(*x*, *B*)

— unknown(*x*, <difference>*A B*</difference>), otherwise.

When a difference element has one and only one child element, the semantics shall be the same as that of the child element. When a difference element has more than two child elements, the semantics shall be the same as that of <difference>*A B* </difference> where *A* is the first child and *B* is the union of the other child elements.

7.6 ref

Given <ref href="*uri*"/> , a CREPDL schema *s* shall be obtained by dereferencing *uri*. When dereferencing *uri* is not successful (e.g., network errors), the CREPDL processor should report an error and may continue normal processing by assuming that "unknown" holds. If it is successful, the semantics is defined below:

— in(*x*,<ref href="*uri*"/>) when in(*x*, *s*).

— not-in(*x*,<ref href="*uri*"/>) when not-in(*x*, *s*).

— unknown(*x*,<ref href="*uri*"/>) when unknown(*x*, *s*).

7.7 repertoire

<repertoire registry="*t*" name="*n*"/> or <repertoire registry="*t*" number="*n*"/> references to a repertoire in some registry. The attribute "registry" specifies a registry. The attribute "name" or "number" specifies a repertoire by name or number, respectively.

— When the value of the attribute "registry" is "10646", a collection specified in Annex A of ISO/IEC 10646 is referenced.

— When the value of the attribute "registry" is "CLDR", a repertoire in the CLDR registry is referenced.

— When the value of the attribute "registry" is "IANA", a charset in the IANA registry of charsets (IANA Charsets) is referenced. The attribute "name" specifies a name or aliase, while the attribute "number" specifies an MIBenum.

— Otherwise, the semantics is implementation dependent.

The CREPDL processor is not required to recognise repertoires specified by repertoire elements. However, when the CREPDL processor does not recognise the specified repertoire, it should report an error and may continue normal processing by assuming that "unknown" holds.

Even when the repertoire specified by a repertoire element is recognised, different CREPDL processors may report different results.

8 Validation

A CREPDL processor is a computer program that validates characters against CREPDL schemas.

When a CREPDL schema is incorrect, a CREPDL processor shall report errors and halt.

Given a character and a correct CREPDL schema, a CREPDL processor shall report "in", "not-in", or "unknown".

Given a string and a correct CREPDL schema, a CREPDL processor first decomposes the string into a sequence of characters, and examines each of them in sequence. If every character is in the repertoire, the result is "in". If some character is not in the repertoire, the result is "not-in". Otherwise, the result is "unknown".

9 Conformance

Different conformant CREPDL processors may report different results only in the cases shown below:

— Case 1: Dereferencing IRIs may fail. However, the semantics of CREPDL is defined so that such failures make conformant CREPDL processors err on the safe side. In other words, such failures do not lead to "in" when "not-in" or "unknown" would have been reported, and do not lead to "not-in" when "in" or "unknown" would have been reported.

— Case 2: The semantics of regular expressions depends on the Unicode version. Different conformant CREPDL processors may behave very differently. For example, one may report "in", while another, "not-in".

— Case 3: a repertoire specified by repertoire may be unrecognised by the CREPDL processor. Moreover, even when the repertoire is recognised, different CREPDL processors may have different interpretations of the repertoire.

Annex A
(informative)

A.1 ISO/IEC 8859-6

The repertoire of ISO/IEC 8859-6[1] (with the exception of C0 control functions other than CHARACTER TABULATION, LINE FEED,CARIIAGE RETURN) is described by the following CREPDL schema.

```
<union xmlns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
  <char>\p{IsBasicLatin}</char>
  <char> |&#xA4;|&#xAD;|&#x60C;|&#x61B;|&#x61F;|[&#x621;-&#x63A;]|[&#x640;-&#x652;]</char>
</union>
```

An alternative representation is shown below.

```
<union xmlns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
  <char>\p{IsBasicLatin}</char>
  <char> </char>
  <char>&#xA4;</char>
  <char>&#xAD;</char>
  <char>&#x60C;</char>
  <char>&#x61B;</char>
  <char>&#x61F;</char>
  <char>[&#x621;-&#x63A;]</char>
  <char>[&#x640;-&#x652;]</char>
</union>
```

A.2 ISO/IEC 8859-15

The repertoire of ISO/IEC 8859-15[2] (with the exception of C0 control functions other than CHARACTER TABULATION, LINE FEED,CARIIAGE RETURN) is described by the following CREPDL schema.

```
<union xmlns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
  <char>\p{IsBasicLatin}</char>
  <char>[ -&#xA3;]</char>
  <char>&#xA5;</char>
  <char>&#xA7;</char>
  <char>[&#xA9;-&#xB3;]</char>
  <char>[&#xB5;-&#xB7;]</char>
  <char>[&#xB9;-&#xBB;]</char>
  <char>[&#xBF;-&#xFF;]</char>
  <char>[&#x152;-&#x153;]</char>
  <char>[&#x160;-&#x161;]</char>
  <char>&#x178;</char>
  <char>[&#x17D;-&#x17E;]</char>
  <char>&#x20AC;</char>
</union>
```

A.3 The Japanese list of kanji characters for the first grade

The Japanese Ministry of Education, Culture, Sports, Science and Technology maintains six lists of kanji characters (see Gakunenbetsu kanji haitouhyou[5]). The list for the first grade is described by the following CREPDL schema, and this list contains 80 characters.

```
<union xmlns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
  <char>&#x4E00;</char>
  <char>&#x4E03;</char>
```

<char>[三-下]</char>
<char>中</char>
<char>九</char>
<char>二</char>
<char>五</char>
<char>人</char>
<char>休</char>
<char>先</char>
<char>入</char>
<char>八</char>
<char>六</char>
<char>円</char>
<char>出</char>
<char>力</char>
<char>十</char>
<char>千</char>
<char>口</char>
<char>右</char>
<char>名</char>
<char>四</char>
<char>土</char>
<char>夕</char>
<char>大</char>
<char>天</char>
<char>女</char>
<char>子</char>
<char>字</char>
<char>学</char>
<char>小</char>
<char>山</char>
<char>川</char>
<char>左</char>
<char>年</char>
<char>手</char>
<char>文</char>
<char>日</char>
<char>早</char>
<char>月</char>
<char>木</char>
<char>本</char>
<char>村</char>
<char>林</char>
<char>校</char>
<char>森</char>
<char>正</char>
<char>気</char>
<char>水</char>
<char>火</char>
<char>犬</char>
<char>玉</char>
<char>王</char>
<char>生</char>
<char>田</char>
<char>男</char>
<char>町</char>
<char>[白-百]</char>
<char>目</char>
<char>石</char>
<char>空</char>
<char>立</char>
<char>竹</char>
<char>糸</char>
<char>耳</char>
<char>花</char>
<char>草</char>
<char>虫</char>

```
  <char>&#x898B;</char>
  <char>&#x8C9D;</char>
  <char>&#x8D64;</char>
  <char>&#x8DB3;</char>
  <char>&#x8ECA;</char>
  <char>&#x91D1;</char>
  <char>&#x96E8;</char>
  <char>&#x9752;</char>
  <char>&#x97F3;</char>
</union>
```

NOTE One could use a single regular expression. However, some other lists of kanji characters have thousands of kanji characters, which prohibit the use of a single regular expression.

A.4 The Japanese list of kanji characters for the second grade

The list for the second grade is described by the following CREPDL schema. It contains 160 characters.

```
<union xmlns="http://purl.oclc.org/dsdl/crepdl/ns/structure/1.0">
<union>
  <char>&#x4E07;</char>
  <char>&#x4E38;</char>
  <char>&#x4EA4;</char>
  <char>&#x4EAC;</char>
  <char>&#x4ECA;</char>
  <char>&#x4F1A;</char>
  <char>&#x4F53;</char>
  <char>&#x4F55;</char>
  <char>&#x4F5C;</char>
  <char>&#x5143;</char>
  <char>&#x5144;</char>
  <char>&#x5149;</char>
  <char>&#x516C;</char>
  <char>&#x5185;</char>
  <char>&#x51AC;</char>
  <char>&#x5200;</char>
  <char>&#x5206;</char>
  <char>&#x5207;</char>
  <char>&#x524D;</char>
  <char>&#x5317;</char>
  <char>&#x5348;</char>
  <char>&#x534A;</char>
  <char>&#x5357;</char>
  <char>&#x539F;</char>
  <char>&#x53CB;</char>
  <char>&#x53E4;</char>
  <char>&#x53F0;</char>
  <char>&#x5408;</char>
  <char>&#x540C;</char>
  <char>&#x56DE;</char>
  <char>&#x56F3;</char>
  <char>&#x56FD;</char>
  <char>&#x5712;</char>
  <char>&#x5730;</char>
  <char>&#x5834;</char>
  <char>&#x58F0;</char>
  <char>&#x58F2;</char>
  <char>&#x590F;</char>
  <char>&#x5916;</char>
  <char>&#x591A;</char>
  <char>&#x591C;</char>
  <char>&#x592A;</char>
  <char>&#x59B9;</char>
  <char>&#x59C9;</char>
  <char>&#x5BA4;</char>
```

<char>家</char>
<char>寺</char>
<char>少</char>
<char>岩</char>
<char>工</char>
<char>市</char>
<char>帰</char>
<char>広</char>
<char>店</char>
<char>弓</char>
<char>引</char>
<char>弟</char>
<char>弱</char>
<char>強</char>
<char>当</char>
<char>形</char>
<char>後</char>
<char>心</char>
<char>思</char>
<char>戸</char>
<char>才</char>
<char>教</char>
<char>数</char>
<char>新</char>
<char>方</char>
<char>明</char>
<char>星</char>
<char>春</char>
<char>昼</char>
<char>時</char>
<char>晴</char>
<char>曜</char>
<char>書</char>
<char>朝</char>
<char>来</char>
<char>東</char>
<char>楽</char>
<char>歌</char>
<char>止</char>
<char>歩</char>
<char>母</char>
<char>毎</char>
<char>毛</char>
<char>池</char>
<char>汽</char>
<char>活</char>
<char>海</char>
<char>点</char>
<char>父</char>
<char>牛</char>
<char>理</char>
<char>用</char>
<char>画</char>
<char>番</char>
<char>直</char>
<char>矢</char>
<char>知</char>
<char>社</char>
<char>秋</char>
<char>科</char>
<char>答</char>
<char>算</char>
<char>米</char>
<char>紙</char>
<char>細</char>
<char>組</char>

```
    <char>&#x7D75;</char>
    <char>&#x7DDA;</char>
    <char>&#x7FBD;</char>
    <char>&#x8003;</char>
    <char>&#x805E;</char>
    <char>&#x8089;</char>
    <char>&#x81EA;</char>
    <char>&#x8239;</char>
    <char>&#x8272;</char>
    <char>&#x8336;</char>
    <char>&#x884C;</char>
    <char>&#x897F;</char>
    <char>&#x89AA;</char>
    <char>&#x89D2;</char>
    <char>&#x8A00;</char>
    <char>&#x8A08;</char>
    <char>&#x8A18;</char>
    <char>&#x8A71;</char>
    <char>&#x8A9E;</char>
    <char>&#x8AAD;</char>
    <char>&#x8C37;</char>
    <char>&#x8CB7;</char>
    <char>&#x8D70;</char>
    <char>&#x8FD1;</char>
    <char>&#x901A;</char>
    <char>&#x9031;</char>
    <char>&#x9053;</char>
    <char>&#x9060;</char>
    <char>&#x91CC;</char>
    <char>&#x91CE;</char>
    <char>&#x9577;</char>
    <char>&#x9580;</char>
    <char>&#x9593;</char>
    <char>&#x96EA;</char>
    <char>&#x96F2;</char>
    <char>&#x96FB;</char>
    <char>&#x982D;</char>
    <char>&#x9854;</char>
    <char>&#x98A8;</char>
    <char>&#x98DF;</char>
    <char>&#x9996;</char>
    <char>&#x99AC;</char>
    <char>&#x9AD8;</char>
    <char>&#x9B5A;</char>
    <char>&#x9CE5;</char>
    <char>&#x9CF4;</char>
    <char>&#x9EA6;</char>
    <char>&#x9EC4;</char>
    <char>&#x9ED2;</char>
  </union>
```

Bibliography

[1] *ISO/IEC 8859-6, Information technology — 8-bit single-byte coded graphic character sets — Part 6: Latin/Arabic alphabet*

[2] *ISO/IEC 8859-15, Information technology — 8-bit single-byte coded graphic character sets, — Part 15: Latin alphabet No. 9*

[3] *A Notation for Character Collections for the WWW*, W3C Note, 14 January 2000, available at http://www.w3.org/TR/charcol

[4] *XQuery 1.0: An XML Query Language*, W3C Recommendation, available at http://www.w3.org/TR/xquery/

[5] *Gakunenbetsu kanji haitouhyou (in Japanese)*, 14 December 1998, available at http://www.mext.go.jp/b_menu/shuppan/sonota/990301b/990301d.htm