An EBNF Grammar for Wiki Creole 1.0

MARTIN JUNGHANS, DIRK RIEHLE, RAMA GURRAM,
MATTHIAS KAISER, MARIO LOPES, AND UMIT YALCINALP

SAP Research, SAP Labs LLC
3475 Deer Creek Rd
Palo Alto, CA 94304 U.S.A.

Today's wiki engines are not interoperable. This is an unfortunate consequence of the lack of rigorously specified standards. This technical report presents a complete and validated EBNF-based grammar for Wiki Creole, a community standard for wiki markup. Wiki Creole is also the only standard currently available. Wiki Creole is being specified using prose, leading to inconsistencies and ambiguities. Our grammar uncovered those ambiguities which we fed back into the specification process. The Wiki Creole grammar presented in this report makes the creation of Wiki Creole parsers simple using parser generators, ANTLR in our case. Using a precise specification of wiki markup lets us decouple wiki editors from wiki storage from further wiki processing tools. Based on this decoupling layer we expect innovation on these different parts to proceed independently and at a faster pace than before.

1. INTRODUCTION

Wikis were invented in 1995 [2]. They have become a widely used tool on the web and in the enterprise since then [1]. In the form of Wikipedia, wikis are having a significant impact on society [10]. Many different wiki engines have been implemented since the first wiki was created. All of these wiki engines integrate the core page rendering engine, its storage backend, the processing tools, and the page editor in one software package.

Wiki pages are written in wiki markup. Almost all wiki engines define their own markup language. Different software components of the wiki engine like the page rendering part are tied to that particular markup language. In addition, users have to learn different wiki markup languages if they want to work with different wiki engines. Also, corporate IT departments have to implement custom migration tools if they want to switch from one wiki engine to another.

Basically, each wiki engine is its own vertically integrated technology stack. A consequence of this vertical integration is that wiki engines are generally not interoperable. **Such vertical integration significantly hinders wiki innovation.**

At the ACM SIGWEB sponsored 2006 International Symposium on Wikis [11], a community effort was started to develop a common wiki markup standard called Wiki Creole [7]. The specification represents the effort of multiple involved parties, mostly wiki engine implementers and their corresponding sponsors. Having an agreed-upon core

wiki markup standard will allow users to use different wiki engines more easily, will allow for wiki contents exchange, and will generally improve interoperability.

Unfortunately, Wiki Creole is being specified using prose. Prose is highly susceptible to ambiguities. The original Wiki Creole specification is also inconsistent and incomplete. However, Wiki Creole is the only available attempt at a wiki markup standard that has broad support in the community, and we expect it to enter a more formal standardization process soon.

To improve interoperability between wiki engines, it is instrumental that we rely on a more precise syntax and semantics specification for wiki markup. This report presents the first complete grammar for Wiki Creole [8]. A semantics definition is being worked on in parallel. To the best of our knowledge, no other complete grammar has been published before, not for Wiki Creole or for any other wiki markup.

2. WIKI MARKUP AND WIKI CREOLE

Wiki markup is the text users write in a wiki page that constitutes the page's contents. Like in HTML, markup allows for formatting instructions like bold or italic. For example, the text

```
//EBNF grammar//
```

would be interpreted as the words "EBNF grammar" in italics, i.e.

```
<i>EBNF grammar</i>
```

and typically displayed as

EBNF grammar

A community effort was started in 2006 to develop a common wiki markup syntax called Wiki Creole. It is the only known attempt for standard markup that has been undersigned by a larger number of wiki engine implementers [9]. Wiki Creole 1.0 [8] was released in mid 2007 and at the time of writing is the most recent release. The prose specification explains in general terms how to interpret and render specific markup, for example, how // serves as opening and closing markup for italic.

We have developed a context-free grammar for Wiki Creole 1.0. The benefits of using our grammar over using prose are:

1. (Almost) **trivial parser construction** using parser generators. We used the parser generator ANTLR to create a first Wiki Creole parser.
2. A formal language specification is needed as the base for a subsequent **semantics specification** and will **open up wikis for machine processing and automation.** This supports the integration of wikis into the Semantic Web [6].
3. A clear wiki markup specification **improves communication** between wiki engine developers. There can be no different interpretation of the specification, because grammar-based parsers show uniform behavior. Such parsers have a precisely defined set of valid markup.
4. Usability will also increase because **users can rely on the same rendering behavior** in different wiki engines.
5. **Simplified extension of standard markup with new features.** Extensions can be added to the grammar in a straightforward way. In contrast to this, today's regular-expression-based parsers make it hard to extend the markup language because side effects are hard to control.
6. The ability to **make performance predictions** based on proven and well-understood language theory. Today's regular-expression-based parsers are all multi-pass parsers, leading to hard-to-predict performance behavior.
7. **Discovery of ambiguities** in the prose specification through a more rigorous specification mechanism.
8. The base for a **well-defined interchange** of wiki page content between wiki engines. Without a precise markup definition, different markup interpretations may result, and a well-working interchange becomes impossible.

Appendix A shows our EBNF-based grammar for the Wiki Creole 1.0 specification. We use the EBNF syntax of the parser generator ANTLR. ANTLR's EBNF syntax is illustrated in Table 1 and discussed in detail in [5]. In [3] we provide a more detailed discussion of the design and implementation of the grammar, the parser, our test suites, and the performance gains and issues. In [4], the interested reader can download the grammar as a text file.

3. SUMMARY AND CONCLUSIONS

This technical report presents the first complete grammar for the first community standard for wiki markup called Wiki Creole. The grammar is an EBNF grammar specified

using ANTLR's EBNF syntax, and the markup standard being specified is Wiki Creole 1.0, released in July 2007. A precise syntax specification is necessary for decoupling different wiki technology components, and it forms the necessary base for further work in wiki markup syntax and semantics. By providing this specification to the community and in aiding the Wiki Creole specification process, we hope to foster wiki technology innovation and make it progress at a faster pace than presently possible.

REFERENCES

[1] GOOGLE TRENDS. *Trends on Wiki Keyword Search.* See http://www.google.com/trends?q=wiki

[2] LEUF, B., AND CUNNINGHAM, W., 1999. *The Wiki Way: Quick Collaboration on the Web.* Addison-Wesley.

[3] JUNGHANS, M., RIEHLE, D., GURRAM, R., KAISER, M., LOPES, M., AND YALCINALP, U. 2008. A Grammar for Standardized Wiki Markup. In submission. Please contact the authors for a copy.

[4] JUNGHANS, M., AND RIEHLE, D. 2007. *Wiki Creole 1.0 EBNF Grammar.* See http://www.wikimarkup.org. Web-published.

[5] PARR, T. 2007 *ANTLR 3 Grammar Syntax.* See http://www.antlr.org/wiki/display/ANTLR3/Grammars. Web-published.

[6] SCHAFFERT, S., VOELKEL, M., AND DECKER, S. 2006 *Proceedings of the First Workshop on Semantic Wikis*. Web-published.

[7] WIKI CREOLE 2007. See http://www.wikicreole.org. Web-published.

[8] WIKI CREOLE 2007. *Wiki Creole 1.0 Specification*. See http://www.wikicreole.org/wiki/ Creole1.0. Web-published.

[9] WIKI ENGINES 2007. See http://www.wikicreole.org/wiki/Engines. Web-published.

[10] WIKIPEDIA, ENGLISH. See http://en.wikipedia.org. Web-published.

[11] WIKI SYMPOSIUM. See http://www.wikisym.org. Web-published.

Martin Junghans is an M.S. student at Technical University of Cottbus, Germany. The work presented in this report was performed during an internship at SAP Labs LLC in Palo Alto, California, under the supervision of Dirk Riehle.

Dirk Riehle is the principal investigator of the open source and Web 2.0 applications research group at SAP Research, SAP Labs LLC, in Palo Alto, California. He is the founder of the *International Symposium on Wikis* conference series.

Rama Gurram, Matthias Kaiser, Mario Lopes, and Umit Yalcinalp are researchers at SAP Research who supported and contributed to this work.

Table 1: ANTLR's syntax for EBNF-based grammars.

Non-terminal symbols	small letters without space
Terminal symbols	capital letters without space
Definition	indicated by a colon
Definition separation	indicated by a pipe
Concatenation	no character, write one non-terminal or rather terminal symbol behind the other
Termination	indicated by a semicolon
Option	indicated by a question mark
Repetition (0…n)	indicated by asterisk
Repetition (1…n)	indicated by plus
Grouping	enclosed by parenthesis
Explicit characters	enclosed by single quotation marks
Exception	indicated by tilde

APPENDIX A: WIKI CREOLE 1.0 GRAMMAR

```
grammar creole10;

////////////////////////////////////////////////////////////////////////////////
//////////////////////////   P A R S E R    R U L E S  //////////////////////////
////////////////////////////////////////////////////////////////////////////////


wikipage
    :   ( whitespaces )?  paragraphs  EOF
    ;
paragraphs
    :   ( paragraph )*
    ;
paragraph
    :   nowiki_block
    |   blanks  paragraph_separator
    |   ( blanks )?
            (   heading
            |   {input.LA(1)==DASH && input.LA(2)==DASH &&
                input.LA(3)==DASH && input.LA(4)==DASH}?
                horizontalrule
            |   list_unord
            |   list_ord
            |   table
            |   text_paragraph
            )  ( paragraph_separator )?
    ;
```

```
//////////////////////   T E X T   P A R A G R A P H   //////////////////////

text_paragraph
    :   (   text_line
        |   ( NOWIKI_OPEN  ~( NEWLINE ) )  =>
            nowiki_inline  ( text_element )*  text_lineseparator
        )+
    ;
text_line
    :   text_firstelement  ( text_element )*  text_lineseparator
    ;
text_firstelement
    :   {input.LA(1)!=STAR || (input.LA(1)==STAR && input.LA(2)==STAR)}?
        text_formattedelement
    |   text_first_unformattedelement
    ;
text_formattedelement
    :   ital_markup  text_italcontent  ( ( NEWLINE )?  ital_markup )?
    |   bold_markup  text_boldcontent  ( ( NEWLINE )?  bold_markup )?
    ;
text_boldcontent
    :   ( NEWLINE )?  ( text_boldcontentpart )*
    |   EOF
    ;
text_element
    :   onestar  text_unformattedelement
    |   text_unformattedelement  onestar
    |   text_formattedelement
    ;
text_italcontent
    :   ( NEWLINE )?  ( text_italcontentpart )*
    |   EOF
    ;
text_boldcontentpart
    :   ital_markup  text_bolditalcontent  ( ital_markup )?
    |   text_formattedcontent
    ;
text_italcontentpart
    :   bold_markup  text_bolditalcontent  ( bold_markup )?
    |   text_formattedcontent
    ;
text_bolditalcontent
    :   ( NEWLINE )?  ( text_formattedcontent )?
    |   EOF
    ;
text_formattedcontent
    :   onestar  ( text_unformattedelement  onestar  ( text_linebreak )? )+
    ;
text_linebreak
    :   {input.LA(2)!=DASH && input.LA(2)!=POUND &&
        input.LA(2)!=EQUAL && input.LA(2)!=NEWLINE}?
        text_lineseparator
    ;
text_inlineelement
    :   text_first_inlineelement
    |   nowiki_inline
    ;
text_first_inlineelement
    :   link
    |   image
    |   extension
    ;
text_first_unformattedelement
    :   text_first_unformatted
    |   text_first_inlineelement
    ;
text_first_unformatted
    :   (   ~(  POUND
            |   STAR
            |   EQUAL
            |   PIPE
            |   ITAL
            |   LINK_OPEN
            |   IMAGE_OPEN
            |   NOWIKI_OPEN
            |   EXTENSION
            |   FORCED_LINEBREAK
            |   ESCAPE
            |   NEWLINE
            |   EOF )
        |   forced_linebreak
        |   escaped )+
    ;
```

```
text_unformattedelement
    :   text_unformatted
    |   text_inlineelement
    ;
text_unformatted
    :   (   ~(  ITAL
            |   STAR
            |   LINK_OPEN
            |   IMAGE_OPEN
            |   NOWIKI_OPEN
            |   EXTENSION
            |   FORCED_LINEBREAK
            |   ESCAPE
            |   NEWLINE
            |   EOF )
        |   forced_linebreak
        |   escaped )+
    ;
```

```
////////////////////////////   H E A D I N G   ////////////////////////////

heading
    :   heading_markup  heading_content  ( heading_markup )?  ( blanks )?
        paragraph_separator
    ;
heading_content
    :   heading_markup  heading_content  ( heading_markup )?
    |   ( ~( EQUAL | ESCAPE | NEWLINE | EOF ) | escaped )+
    ;
```

```
////////////////////////////////   L I S T   ////////////////////////////////

list_ord
    :   ( list_ordelem )+  ( end_of_list )?
    ;
list_ordelem
    :   list_ordelem_markup  list_elem
    ;
list_unord
    :   ( list_unordelem )+  ( end_of_list )?
    ;
list_unordelem
    :   list_unordelem_markup  list_elem
    ;
list_elem
    :   ( list_elem_markup )*  list_elemcontent  list_elemseparator
    ;
list_elem_markup
    :   list_ordelem_markup
    |   list_unordelem_markup
    ;
list_elemcontent
    :   onestar  ( list_elemcontentpart  onestar )*
    ;
list_elemcontentpart
    :   text_unformattedelement
    |   list_formatted_elem
    ;
list_formatted_elem
    :   bold_markup  onestar  ( list_boldcontentpart  onestar )*  ( bold_markup )?
    |   ital_markup  onestar  ( list_italcontentpart  onestar )*  ( ital_markup )?
    ;
list_boldcontentpart
    :   ital_markup  list_bolditalcontent  ( ital_markup )?
    |   ( text_unformattedelement )+
    ;
list_italcontentpart
    :   bold_markup  list_bolditalcontent  ( bold_markup )?
    |   ( text_unformattedelement )+
    ;
list_bolditalcontent
    :   ( text_unformattedelement )+
    ;
```

```
////////////////////////////   T A B L E   ////////////////////////////

table
    :   ( table_row )+
    ;
table_row
    :   ( table_cell )+  table_rowseparator
    ;
table_cell
    :   { input.LA(2)==EQUAL }?  table_headercell
    |   table_normalcell
    ;
table_headercell
    :   table_headercell_markup  table_cellcontent
    ;
table_normalcell
    :   table_cell_markup  table_cellcontent
    ;
table_cellcontent
    :   onestar  ( table_cellcontentpart  onestar )*
    ;
table_cellcontentpart
    :   table_formattedelement
    |   table_unformattedelement
    ;
table_formattedelement
    :   ital_markup  ( table_italcontent )?  ( ital_markup )?
    |   bold_markup  ( table_boldcontent )?  ( bold_markup )?
    ;
table_boldcontent
    :   onestar  ( table_boldcontentpart  onestar )+
    |   EOF
    ;
table_italcontent
    :   onestar  ( table_italcontentpart  onestar )+
    |   EOF
    ;
table_boldcontentpart
    :   table_formattedcontent
    |   ital_markup  table_bolditalcontent  ( ital_markup )?
    ;
table_italcontentpart
    :   bold_markup  table_bolditalcontent  ( bold_markup )?
    |   table_formattedcontent
    ;
table_bolditalcontent
    :   onestar  ( table_formattedcontent  onestar )?
    |   EOF
    ;
table_formattedcontent
    :   ( table_unformattedelement )+
    ;
table_inlineelement
    :   link
    |   image
    |   extension
    |   nowiki_inline
    ;
table_unformattedelement
    :   table_unformatted
    |   table_inlineelement
    ;
table_unformatted
    :   (   ~(  PIPE
              |   ITAL
              |   STAR
              |   LINK_OPEN
              |   IMAGE_OPEN
              |   NOWIKI_OPEN
              |   EXTENSION
              |   FORCED_LINEBREAK
              |   ESCAPE
              |   NEWLINE
              |   EOF )
        |   forced_linebreak
        |   escaped )+
    ;
```

```
////////////////////////////   N O W I K I   ////////////////////////////

nowiki_block
    :   nowikiblock_open_markup  ( ~( NOWIKI_BLOCK_CLOSE | EOF ) )*
        nowikiblock_close_markup  paragraph_separator
    ;
nowikiblock_open_markup
    :   nowiki_open_markup  newline
    ;
nowikiblock_close_markup
    :   NOWIKI_BLOCK_CLOSE
    ;
nowiki_inline
    :   nowiki_open_markup  ( ~( NOWIKI_CLOSE | NEWLINE | EOF ) )*
        nowiki_close_markup
    ;
```

```
//////////////////////   H O R I Z O N T A L   R U L E   //////////////////////

horizontalrule
    :   horizontalrule_markup  ( blanks )?  paragraph_separator
    ;
```

```
//////////////////////////////   L I N K   ////////////////////////////////

link
    :   link_open_markup  link_address  ( link_description_markup
        link_description )?  link_close_markup
    ;
link_address
    :   link_interwiki_uri  ':'  link_interwiki_pagename
    |   link_uri
    ;
link_interwiki_uri
    :   'C' '2'
    |   'D' 'o' 'k' 'u' 'W' 'i' 'k' 'i'
    |   'F' 'l' 'i' 'c' 'k' 'r'
    |   'G' 'o' 'o' 'g' 'l' 'e'
    |   'J' 'S' 'P' 'W' 'i' 'k' 'i'
    |   'M' 'e' 'a' 't' 'b' 'a' 'l' 'l'
    |   'M' 'e' 'd' 'i' 'a' 'W' 'i' 'k' 'i'
    |   'M' 'o' 'i' 'n' 'M' 'o' 'i' 'n'
    |   'O' 'd' 'd' 'm' 'u' 's' 'e'
    |   'O' 'h' 'a' 'n' 'a'
    |   'P' 'm' 'W' 'i' 'k' 'i'
    |   'P' 'u' 'k' 'i' 'W' 'i' 'k' 'i'
    |   'P' 'u' 'r' 'p' 'l' 'e' 'W' 'i' 'k' 'i'
    |   'R' 'a' 'd' 'e' 'o' 'x'
    |   'S' 'n' 'i' 'p' 'S' 'n' 'a' 'p'
    |   'T' 'i' 'd' 'd' 'l' 'y' 'W' 'i' 'k' 'i'
    |   'T' 'W' 'i' 'k' 'i'
    |   'U' 's' 'e' 'm' 'o' 'd'
    |   'W' 'i' 'k' 'i' 'p' 'e' 'd' 'i' 'a'
    |   'X' 'W' 'i' 'k' 'i'
    ;
link_interwiki_pagename
    :   ~( PIPE | LINK_CLOSE | NEWLINE | EOF )+
    ;
link_description
    :   ( link_descriptionpart | image )+
    ;
link_descriptionpart
    :   bold_markup  onestar  ( link_bold_descriptionpart  onestar )+
        bold_markup
    |   ital_markup  onestar  ( link_ital_descriptionpart  onestar )+  ital_markup
    |   onestar  ( link_descriptiontext  onestar )+
    ;
link_bold_descriptionpart
    :   ital_markup  link_boldital_description  ital_markup
    |   link_descriptiontext
    ;
link_ital_descriptionpart
    :   bold_markup  link_boldital_description  bold_markup
    |   link_descriptiontext
    ;
link_boldital_description
    :   onestar  ( link_descriptiontext  onestar )+
    ;
link_descriptiontext
    :   (   ~(  LINK_CLOSE
            |   ITAL
            |   STAR
            |   LINK_OPEN
            |   IMAGE_OPEN
            |   NOWIKI_OPEN
            |   EXTENSION
            |   FORCED_LINEBREAK
            |   ESCAPE
            |   NEWLINE
            |   EOF )
        |   forced_linebreak
        |   escaped )+
    ;
link_uri
    :   ~( PIPE | LINK_CLOSE | NEWLINE | EOF )+
    ;
```

```
////////////////////////////   I M A G E   ////////////////////////////

image
    :   image_open_markup  image_uri  ( image_alternative )?
        image_close_markup
    ;
image_uri
    :   ~( PIPE | IMAGE_CLOSE | NEWLINE | EOF )+
    ;
image_alternative
    :   image_alternative_markup  ( image_alternativepart )+
    ;
image_alternativepart
    :   bold_markup  onestar  ( image_bold_alternativepart  onestar )+
        bold_markup
    |   ital_markup  onestar  ( image_ital_alternativepart  onestar )+  ital_markup
    |   onestar  ( image_alternativetext  onestar )+
    ;
image_bold_alternativepart
    :   ital_markup  link_bolditalic_description  ital_markup
    |   onestar  ( image_alternativetext  onestar )+
    ;
image_ital_alternativepart
    :   bold_markup  link_bolditalic_description  bold_markup
    |   onestar  ( image_alternativetext  onestar )+
    ;
image_bolditalic_alternative
    :   onestar  ( image_alternativetext  onestar )+
    ;
image_alternativetext
    :   (   ~(  IMAGE_CLOSE
            |   ITAL
            |   STAR
            |   LINK_OPEN
            |   IMAGE_OPEN
            |   NOWIKI_OPEN
            |   EXTENSION
            |   FORCED_LINEBREAK
            |   NEWLINE
            |   EOF )
        |   forced_linebreak )+
    ;
```

```
onestar
    :   ({input.LA(2)!=STAR}? STAR?) |
    ;
escaped
    :   ESCAPE STAR STAR
    |   ESCAPE .    // in parser rule . means arbitrary TOKEN, not character
    ;
paragraph_separator
    :   ( newline )+
    |   EOF
    ;
whitespaces
    :   ( blanks  |  newline )+
    ;
blanks
    :   BLANKS
    ;
text_lineseparator
    :   newline  ( blanks )?
    |   EOF
    ;
newline
    :   NEWLINE
    ;
bold_markup
    :   STAR STAR
    ;
ital_markup
    :   ITAL
    ;
heading_markup
    :   EQUAL
    ;
list_ordelem_markup
    :   POUND
    ;
list_unordelem_markup
    :   STAR
    ;
list_elemseparator
    :   newline  ( blanks )?
    |   EOF
    ;
end_of_list
    :   newline
    |   EOF
    ;
table_cell_markup
    :   PIPE
    ;
table_headercell_markup
    :   PIPE EQUAL
    ;
table_rowseparator
    :   newline
    |   EOF
    ;
nowiki_open_markup
    :   NOWIKI_OPEN
    ;
nowiki_close_markup
    :   NOWIKI_CLOSE
    ;
horizontalrule_markup
    :   DASH DASH DASH DASH
    ;
link_open_markup
    :   LINK_OPEN
    ;
link_close_markup
    :   LINK_CLOSE
    ;
link_description_markup
    :   PIPE
    ;
image_open_markup
    :   IMAGE_OPEN
    ;
image_close_markup
    :   IMAGE_CLOSE
    ;
image_alternative_markup
    :   PIPE
    ;
forced_linebreak
    :   FORCED_LINEBREAK
    ;
```

```
////////////////////////////////////////////////////////////////////////////////////
////////////////////////////// S C A N N E R   R U L E S ////////////////////////////
////////////////////////////////////////////////////////////////////////////////////

ESCAPE                  : '~';
NOWIKI_BLOCK_CLOSE      : NEWLINE '}}}';
NEWLINE                 : ( CR )? LF | CR;
fragment CR             : '\r';
fragment LF             : '\n';

BLANKS                  : ( SPACE | TABULATOR )+;
fragment SPACE          : ' ';
fragment TABULATOR      : '\t';

COLON_SLASH             : ':' '/';
ITAL                    : '//';
NOWIKI_OPEN             : '{{{';
NOWIKI_CLOSE            : '}}}';
LINK_OPEN               : '[[';
LINK_CLOSE              : ']]';
IMAGE_OPEN              : '{{';
IMAGE_CLOSE             : '}}';
FORCED_LINEBREAK        : '\\\\';
EQUAL                   : '=';
PIPE                    : '|';
POUND                   : '#';
DASH                    : '-';
STAR                    : '*';
SLASH                   : '/';

INSIGNIFICANT_CHAR      : .;
```