Cladistics
The Willi Hennig Society

FASTC: A file format for multi-character sequence data

Journal:	*Cladistics*
Manuscript ID	Draft
Manuscript Type:	Article
Date Submitted by the Author:	n/a
Complete List of Authors:	Wheeler, Ward; American Museum of Natural History, Department of Invertebrates Washburn, Alex; American Museum of Natural History, Department of Invertebrates
Keywords:	Methodology, Molecular systematics < Genomics

FASTC: A file format for multi-character sequence data

Ward C. Wheeler and Alex J. Washburn
Division of Invertebrate Zoology
American Museum of Natural History
200 Central Park West
New York, NY 10024-5192
USA
wheeler@amnh.org
212-769-5754

August 9, 2018

Running Title: FASTC file format

1 Abstract

Here, we define a sequence file format that allows for multi-character elements (FASTC). The format is derived from the FASTA format of Lipman and Pearson (1985) and the custom alphabet format of POY4/5 (Varón et al., 2008; Wheeler et al., 2015). The format is more general than either of these formats and can represent a broad variety of sequence-type data.

Contents

2 Introduction and Motivation

The FASTA (or Pearson) sequence format was first articulated as a component of sequence database searching tools (Lipman and Pearson, 1985). This format is nearly universally recognized and used for sequence data input by a large variety of sequence analysis software packages.

The format is admirably simple (defined in FASTA program documentation, see `www.cse.unsw.edu.au/~binftools/birch/birchhomedir/doc/fasta/fasta20.doc`) with only two control characters ('>' and ';'). The semicolon (';') denotes a comment (to end of line) and '>' labeling a sequence. The sequence label line begins with '>' and continues until white space or end of line is encountered. Sequences consist of single-character IUPAC protein and nucleic acids codes. All other symbols are ignored (such as numbers and white space). Line length is limited to 120 characters.

An example, valid file could look like this:

```
;This is an example file
>First_DNA sequence
ACGTTT @GGA;This is a comment
>Second_DNA sequence
1 GT-A 4 TTCA
```

This would result in the input of two sequences: First_DNA ACGTTTGGA and Second_DNA GTATTCA.

POY4 (Varón et al., 2008) and POY5 (Wheeler et al., 2013, 2015) extended the legal character symbol set to include '-' to represent alignment gaps (an IUPAC symbol for nucleotides but not protein sequences), 'X' to represent any nucleotide (in addition to 'N' for nucleotides), and '?' to represent 'X' or '-' (unknown element or gap).

2.1 Multi-character alphabets

Situations arise where multi-character sequence elements are required or at least convenient. These can include gene synteny, developmental, and comparative linguistic data (explained further in Schulmeister and Wheeler, 2004; Wheeler, 2007, 2012; Wheeler and Whiteley, 2015). POY4 and POY5 contain the 'custom alphabet' sequence type from which the FASTC format described here is derived.

The custom alphabet file format allows for multi-character element representations (e.g. alpha, beta, gamma), but these must be prefix-free. This allows sequence parsing to proceed more easily, but has the limitation that it requires a prefix-free alphabet and not all are. Furthermore, it can lead to less that easily human-legible data files such as:

```
>First_sequence
alphabetagammadelta
>Second_sequence
betagammaalphadelta
```

The custom alphabet elements can also be preceded by a tilda ('∼') denoting reverse orientation (useful for gene synteny data).

3 FASTC

The FASTC format grows out of the FASTA and custom alphabet formats by adding a mandatory white space between elements and allowing for non-prefix free multi-character sequence element specification.

3.1 Grammar specification

The file grammar is specified as follows:

$\langle FILE \rangle$ ::= $\langle SPACEMAYBE \rangle$ ('>' $\langle IDENTIFIER \rangle$ $\langle ID\_END \rangle$ $\langle SEQUENCE \rangle$)*

$\langle IDENTIFIER \rangle$ ::= $\langle SPACE\_PAD \rangle$ ($\langle VALID\_CHAR \rangle$)+ $\langle SPACE\_PAD \rangle$

$\langle ID\_END \rangle$ ::= '\n' $\langle SPACEMAYBE \rangle$
| $\langle COMMENT \rangle$ $\langle SPACEMAYBE \rangle$

$\langle COMMENT \rangle$ ::= ';' ($\langle INLINE\_CHAR \rangle$)* '\n'

$\langle SEQUENCE \rangle$ ::= $\langle ELEMENT \rangle$ ($\langle WHITESPACE \rangle$ $\langle ELEMENT \rangle$)* $\langle MAYBESPACE \rangle$

$\langle ELEMENT \rangle$ ::= $\langle SYMBOL \rangle$
| '[' $\langle MAYBESPACE \rangle$ $\langle SYMBOL \rangle$ $\langle SYMBOL\_LIST \rangle$ $\langle MAYBESPACE \rangle$ ']'

$\langle SYMBOL \rangle$ ::= ($\langle VALID\_CHAR \rangle$)+

$\langle SYMBOL\_LIST \rangle$ ::= ($\langle WHITESPACE \rangle$ $\langle SYMBOL \rangle$)*

$\langle MAYBESPACE \rangle$ ::= ($\langle SPACING \rangle$)*

$\langle WHITESPACE \rangle$ ::= ($\langle SPACING \rangle$)+

$\langle SPACING \rangle$ ::= [\s]+ ◁ **one or more spaces**
| $\langle COMMENT \rangle$

$\langle SPACE\_PAD \rangle$ ::= ($\langle SPACE\_CHAR \rangle$)*

$\langle SPACE\_CHAR \rangle$ ::= [^\n\S] ◁ **not a new-line or a non-space**

$\langle INLINE\_CHAR \rangle$ ::= [^\n] ◁ **not a new-line**

$\langle VALID\_CHAR \rangle$ ::= [^;>\[\]\s] ◁ **not ';', '>', '[', ']', or a space**

It is worth noting that parentheses, '(' and '), *are allowed* in an identifier but may cause conflict with with E/Newick tree file format (Cardona et al., 2008) containing the same identifier that has not been properly quoted.

Multiple identifier lines without sequence data are not permitted:

```
>foo
>bar
```


3.2 Example implementation

An example Haskell implementation of a FASTC file parser conforming to the grammar above can be found here:

https://github.com/amnh/fastc

3.3 Example files

Traditional sequence files can be represented in fastc with the inclusion of separating spaces (from Wheeler and Hayashi, 1998).

```
>Americhernus
T C G A G C C T C C A A T G A T A C G T T G A A A G G C G T T T A T C G T T
G G G G C C G A C A G - - C G T C G T G G G C T C G G T T G G C C T T A
A A A A G C T G A T C G G G T T C T C C G G C A A T T T T A C T T T G A A A A
  A A T T A G G G T G C T C A A G T G C C
>Chanbria;From Genbank
T C T A G A C T G G T G G T C C G C C T C T G G T G G T T A C T A C C T G
G C C T A A A C A A T T T G C C G G T T T T C C C T T G G T G C T C T T C A
C C G A G T G T C T T G G G G G A C T G G T A C G T T T A C T T T G A A G A
A A C T A G A G T G C T C A A A - C A G G C G T A A C
G C C
>Gea
```

```
 T C C G G C C G G A C G G G T C C G C C T A C C G G T G G T
T A C T G T T C G C T G C C G A G C T T C A G G G G G C C G C T G T C G
 A T G A T C T T C A T C G G T T A T C T T C C G T A A C C C T C A C
G T T T A C T T T G A A A A A A T T A G A G T G C T C A A A G C A G C - -
G C G A C G C C
>Hypochilus;An interesting spider
- T C C A G A C G G G C G G T C C G C C T A A C G G T G G T T A C T G C C T
G G C C T G A A C A A C C A G C C G G T T T C C C T A G A T G A T
C T T C A T T G A T T G T C T T G G G T G A C C G G C A C G T T T A C T T
T G A A A A A A T T A G A G T G C T C A A A G C A G C G T G A C G C C
```

Linguistic data based on the international phonetic alphabet (IPA) may contain multi-character sequence elements (from Whiteley et al., 2019).

```
>Ngombe
\ve b \'O
>Mbesa
\'{\i} f \'{\i} n j \‘{\i}
>Likile
b o s \'a m b \'a
>Mongo
l O w \'O
```

Gene synteny data can also be represented.

```
>species_1
CYTB NAD1 12SrDNA 16SrDNA
>species_2
CYTB 12SrDNA 16SrDNA
>species_3
16SrDNA 12SrDNA NAD1 CYTB
```

The following file would be invalid due to absence of data for Mbesa.

```
>Ngombe
e b \'O
>Mbesa
>Likile
b o s \'a m b \'a
>Mongo
l O w \'O
```

4 Summary

The fastc format naturally generalizes existing sequence format files and can be employed to represent a diversity of data types with linear ordering.

5 Acknowledgements

This work was supported by DARPA SIMPLEX ("Integrating Linguistic, Ethnographic, and Genetic Information of Human Populations: Databases and Tools," DARPA-BAA-14-59 SIMPLEX TA-2, 2015-2018).

References

Cardona, G., Russelló, F., and Valiente, G. 2008. Extended newick: it is time for a standard representation of phylogenetic networks. *BMC Bioinformatics*, 9(532). doi: 10.1186/1471-2105-9-532.

Lipman, D. J. and Pearson, W. R. 1985. Rapid and sensitive protein similarity searches. *Science*, 227:1435–1441.

Schulmeister, S. and Wheeler, W. C. 2004. Comparative and phylogenetic analysis of developmental sequences. *Evolution and Development*, 6:50–57.

Varón, A., Vinh, L. S., Bomash, I., and Wheeler, W. C. 2008. Poy 4.0. American Museum of Natural History. http://research.amnh.org/scicomp/projects/poy.php.

Wheeler, W. C. 2007. Chromosomal character optimization. *Mol. Phyl. Evol.*, 44:1130–1140.

Wheeler, W. C. 2012. *Systematics: A course of lectures*. Wiley-Blackwell.

Wheeler, W. C. and Hayashi, C. Y. 1998. The phylogeny of the extant chelicerate orders. *Cladistics*, 14(2):173–192.

Wheeler, W. C., Lucaroni, N., Hong, L., Crowley, L. M., and Varón, A. 2013. POY version 5.0. American Museum of Natural History. http://research.amnh.org/scicomp/projects/poy.php.

Wheeler, W. C., Lucaroni, N., Hong, L., Crowley, L. M., and Varón, A. 2015. POY version 5: Phylogenetic analysis using dynamic homologies under multiple optimality criteria. *Cladistics*, 31. 189-196.

Wheeler, W. C. and Whiteley, P. M. 2015. Historical linguistics as a sequence optimiza-

tion problem: The evolution and biogeography of Uto–Aztecan languages. *Cladistics*, 31(2):113–125.

Whiteley, P. M., Xue, M., and Wheeler, W. C. 2019. Revising the bantu tree. *Cladistics*, pages 1–20.