FASTA to Proto I/O Converter

Ian M. Dew

1. Overview

The fasta2proto program converts curated sequences (currently only contaminants, but will be expanded to handle repeats) from a Celera internal FASTA format to assembler proto I/O internal screen item messages (MESG_ISN and MESG_SCN – refer to ..\BigPicture\ProtoSpec.rtf). The sequences are specified in an instruction file, which also contains curated parameters needed to populate fields of each screen item message.

2. Memory and Processor Requirements

The fasta2proto program uses one processor (not threaded) and requires slightly more memory than the size of the largest sequence.

3. Interface

The fasta2proto program is invoked by the following command and arguments.

```
fasta2proto  [-r|-c] <instruction-filename>
```

The instruction filename must have as its prefix a UID that identifies the collection of sequences used for a sequencing project. The filename suffix must be ".inst". An example is 100000200391.inst. The fasta2proto program parses the filename to set the repeat ID of each screen item to the UID of the collection. The output filename has the same prefix as the input filename with the suffix ".pio".

The -r option directs fasta2proto to convert repeat sequences. This is not yet supported.

The -c option directs fasta2proto to convert contaminant sequences. This is the only currently supported option.

Contaminant and repeat sequences are distinguished in running fasta2proto because it is anticipated that the two types of data will require different information for populating and using their respective screen item messages. Also, contaminants are used in preassembly and are created using the MESG_ISN format whereas repeats are used in assembly and will be created using the MESG_SCN format.

For contaminants, the instruction file lists one contaminant per line, with the following bar-delimited (|) fields:

- Filename of the internal FASTA file,
- UID of the contaminant (uint64),
- Internal accession number of the contaminant (uint32),

- Required similarity (`float`) on (0,1], where 1 is a perfect match, and
- Minimum length of a match (`int32`).

An example of an instruction line is;

```
/work/data/contaminant/200000030051.int|200000030051|1002910|0.95|32
```

Each internal FASTA file must contain only one sequence (i.e., they must not be multi-FASTA files). Its FASTA header line will be used as the message's `source` field.

4. Design

The `fasta2proto` program is a simple utility that creates a proto IO message for each instruction line and associated internal FASTA file.

5. Limitations

The program does not yet handle repeat sequences. All non acgntuACGNTU characters other than \r and \n in sequences are converted to n.

6. Status

Contaminant sequence conversion functions properly. Format of instruction lines for repeat sequences has yet to be determined.

7. Architecture and Dependencies

The code for the `fasta2proto` program is contained in `AS_URT_fasta2proto.c` in the `AS_URT` module. It also depends on the `AS_MSG` module for proto I/O.

AUTHORS

Ian Dew:
Created: 19 Mar 99