Rexx the Data Converter

RexxLA, Hursley — May 2009

Mike Cowlishaw
IBM Fellow
Overview

• Two projects processing text data
  – weather analysis (for flying)
  – documents marked up with GML

… nice examples of Rexx usage?
Why analyse the weather?

• Microlight aircraft can be very light and susceptible to wind (especially gusty winds)

• Also cannot fly in the dark, or in poor visibility, rain, storms, etc.

• So how many days a year can they fly?
3-axis Microlight

• Maximum 30–35 mph winds (20 mph crosswind)
Weight-shift (flexwing) Microlight

- Maximum 20 mph winds (10-15 mph crosswind)
Powered parachute Microlight

- Maximum 10 mph winds (takeoff and land only into wind)
So what data are needed?

- In the UK, the Met Office has been recording observations at 300 stations since 1854; data free for academic research

- Hourly data needed, for:
  - Storms, snow, winds, gust data
  - Rainfall

- Only 107 stations matching (47 with gusts)
Data formats

• Weather file: all 300 stations in one file, one file (~1GB) per year (1998–2007 = 9 GB)
  – all comma-separated-values (.csv)
  – rainfall files similar but smaller (2 GB total)

• Geographic information for each station

• Sunrise/sunset times generated on the fly
  – 19 hours maximum in Shetland, 16 in Cornwall
Processing steps

1. Extract data for a given station (1 hour)

2. Clean the data (remove duplicates, etc.) – at this point exactly one line per hour

3. Merge and filter the data, along with sunrise and sunset times, to create a ‘simplified hourly’ file (purely met. data)
4. Process simplified hourly file against flying criteria (next slide) to create a fixed-format ‘flying bits’ file (4MB/station), e.g:

23 2004-01-01 00:00 001110111 000111 0100000000

5. Analyse the flying bits file in various ways to generate web pages (merging in station geographic data, etc., as needed)
Flying criteria

• No ‘bad weather’ (rain, snow, storms, etc.)

• Sufficient visibility for visual flight rules:
  – daylight
  – horizontal visibility at least 5 km (3 miles)
  – cloudbase at least 1000 feet (or < 6/8 cover)

• Winds below a certain maximum
Generating web pages

• Could generate HTML pages directly

• Decided to generate .wiki files that could be part of a MemoWiki project (and hence automatically converted to web pages and publishable)

• All of the above is text processing (in Rexx)
Demo

1. Running programs (except Extract)

2. Wiki pages in MemoWiki

3. Final web pages  
   ... available at:

   http://speleotrovoe.com/weather/
Problem 2 – GML

- 25+ years of documents marked up in GML (generalized markup language)
  - processed on mainframe (VM DCF SCRIPT)
  - no one else can process

- How to format on PC/laptop?

- How to make NetRexx documents available to RexxLA?
Blanks and White Space

(i.Blanks:ei. (spaces) may be freely used in a program to improve appearance and layout, and most are ignored. Blanks, however, are usually significant

:ul.

:li.within literal strings (see below)
:li.between two tokens that are not special characters (for example, between two symbols or keywords)
:li.between the two characters forming a comment delimiter
:li.immediately outside parentheses (:q.:hp4.(:ehp4.:eq. and :q.:hp4.):ehp4.:eq.) or brackets (:q.:hp4.&lbrk.:ehp4.:eq. and :q.:hp4.&rbrk.:ehp4.:eq.).
:eul.
Which might format as…

Blanks and White Space

*Blanks* (spaces) may be freely used in a program to improve appearance and layout, and most are ignored. Blanks, however, are usually significant

- within literal strings (see below)

- between two tokens that are not special characters (for example, between two symbols or keywords)

- between the two characters forming a comment delimiter

- immediately outside parentheses (“(” and “)”) or brackets (“[” and “]”).
What format to convert to?

- Something open source, with a reasonable formatter (with index, tables, etc.)

- (X)HTML alone not powerful enough, and not always easily convertible to other formats

- To save time: something I had already used
OpenOffice format

- A simple zip file, containing plain text files:

  META-INF/manifest.xml
  mimetype
  meta.xml
  settings.xml
  styles.xml
  content.xml

- All fixed, except `content.xml`
OpenOffice Writer

• Application for opening and editing .odt files, with formatting, indexing, etc. (very similar to Microsoft Word)

• Sufficient formatting power (though could be better, especially indexing)

• Can export to PDF, XHTML, LaTeX, and MediaWiki (Wikipedia)
1. Header stuff (fonts used, etc.)

2. Styles used only in this document (definition lists and tables layouts, etc.)

3. Body content
   - Front matter (Title page, copyrights, ToC)
   - Multiple sections (one per GML file)
   - Back matter (Index)
Rexx programs used

• Overall builder – calls .odt-specific programs for each part

• `gml2odt` – checks GML and generates XML
  – 783 loc for front and back matter generation
  – 1986 loc for general GML → XML converter

• `odtwrap` – checks and makes .zip file
  – 163 loc
Demo

1. nrl2.nrl files
2. Build
3. Resulting content.xml
4. The .odt and PDF files
Questions?