CSV Validation

Adam Retter

Evolved Binary Ltd

@adamretter / adam.retter@googlemail.com

Adam Retter

- Consultant
- Software Engineer
- Data(base) Geek
- Last 2.5 Years with The National Archives (UK)
  - Building a new Digital Archive for the UK -> DRI

Talking about

- CSV Schema Language
- CSV Validation Tool

It all started with...

The National Archives

- Archive Records of UK from OGDs, NGOs and Special Interest
- Excellent at traditional Paper records
  - One of the largest collections in the world
  - Over 11 million historical Government and Public Records
- However, most records today are not created on paper!
  - Predicted 2013 - 2020:
    - >6PB of Digital Records to Archive
    - 50% of which will be ***Born Digital***
  - 2009: Existing Digital Records System will not cope...
    - 2011: Build new Digital Records Infrastructure

Digital Repository Infrastructure

1. Records arrive via:
   - Hard Disks (USB etc)
   - DVD / CD / Digital Video Cassette / Tape (mostly LTO 1 to 6)
   - SFTP
2. Load Records
3. Test, Secure and Examine Records (Pre-Ingest)
4. Extract Metadata and Archive (Ingest)
5. Enable Digital Archivists (Search, Retrieval and Edit)
6. Export Transcoded Records and Metadata (Publish / Sell)

Q: Digital Preservation...

1. What constitutes a Record?
2. Given a disk of files - What do ***you*** Accession?
3. How does DRI know what it should process and how?

A: Metadata!

1. One or more Files ***and*** Metadata (Technical, Provenance, Transcription, Closure)
2. Records Selection Process by OGD, provided as metadata
3. Search source for metadata and process described records

Collecting Metadata

- TNA creates Metadata Standards for their records
- Expects suppliers to provide Metadata alongside files (records)
- CSV was decided upon as file format for metadata
  - XML and RDF were both considered
  - Must be achievable by non-technical staff
    - Often Gov IT Departments are outsourced
    - Installing even free applications is prohibitive (cost)
    - Likely familiarity with MS Excel (and available)
- Past experience has shown that if the barriers to entry are too high, then suppliers will not comply

CSV Metadata Problems

- TNA has complex metadata requirements
  - Conditional Values and Co-variance Constraints
  - Relationships: row -> row, csv -> csv, csv -> files
- Errors are introduced
  - Human
    - Transcription mistakes
    - Rename .xls file to .csv
  - Computer
    - Poorly implemented Metadata generation
    - MS Excel can hide/mangle data e.g. #NAME?
  - Commercial - Suppliers try and cut corners


FAIL
FAST
AND
REPEAT

CSV Validation

- Version 0.1 (Internal Only)
  - Command Line tool developed in Java
  - Validated metadata across 3 types of CSV files
  - Validation rules were expressed in Java DSL
  - Home Guard Collection (Proof of Concept)
    - 82,800 Records Checked
    - >250,000 rows of CSV data
    - ~4.5TB of JP2000 Images validated
- Still... many failures detected!
  - However, faster feedback (Pre-Ingest).
  - Eventually... shared with digitisation supplier

CSV Validation

- Version 0.1 was nice... but in Version 0.2 can we have:
  - Validation rules DSL should
    - Be External (no need to recompile)
    - Writable by Domain Experts not Developers (no Java!)
    - Easily sharable with suppliers
  - Application(s) should be
    - Freely available to suppliers
    - Useable in DRI Pre-Ingest and Ingest processing

The CSV Schema Language

- Started at TNA as text based DSL for CSV Validation Rules
- As interest grew... Requirements exploded!
- Now:
  - A generic CSV Schema Language
  - 60+ Expression for forming Validation Rules
  - 10+ High-level data types (Dates, Times, Numbers etc.)
  - Flexible Support for any tabular text data (CSV, TSV, etc.)
  - Open Standard (Currently... guided by TNA)
  - Freely available under MPL v2.0

https://github.com/digital-preservation/csv-schema

Design Principles of CSV Schema

- Simple Plain-Text Expression
  - Composable by non-techies with text editor
- Implicit Context
  - Natural to write, rules are per-column, applied row-by-row
- Sane Defaults
  - CSV files come in all shapes, e.g. default to RFC 4180.
- Streamable
  - CSV files may be large. Do not prohibit efficient processing.
- NOT a Programming Language!
  - Powerful? Yes! For programmers? No!

CSV Schema 101

- A CSV Schema consists of:
  - Directives - modify behaviour of CSV parsing and rules
  - Rules - 1 per column, composed of expressions

CSV Data

```
first_name,last_name,gender,dob
Adam,Retter,33,M,1981-02-04
Elisabeth,Roberts,33,F,1980-11-13
```

CSV Schema

```
version 1.0
@totalColumns 4
first_name: length(2, *)
last_name: length(2, *)
gender: is("M") or is("F") @optional
dob: xDate
```

CSV Schema - Example 2

- Global Directives control parsing of CSV

CSV Data

```
"Huxley"$"feline"$"Short Haired Domestic"$"10"
"Precious"$"feline"$"Short Haired Domestic"$"6"
"Mac"$"canine"$"Dalmatian"$"12"
```

CSV Schema

```
version 1.0
@separator '$' @quoted @totalColumns 4 @noHeader
name: notEmpty
class: is("feline") or is("canine")
breed: length(3, 255)
age: positiveInteger
```

CSV Schema - Example 3

- Conditional Expressions and Co-Variance

CSV Data

```
name,animal,age,short description,notes
James,Mouse,4,,
Louise,Elephant,45,In good health,
```

CSV Schema

```
version 1.0
name: notEmpty
animal: notEmpty
age: if($animal/is("mouse"), range(0, 3), positiveInteger)
"short description": length(*, 255) @optional
notes:
```

CSV Schema - Example 4

- External Expressions (mainly file checks)

CSV Data

```
"id","fn","checksum","classifications"
"1","image1.jp2","54229abfcfa5649e7003b83dd4755294",""
"2","image2.jp3",3d0ad5a7a8ef3b1d4e6ea33e92e4d3b5,""
"3","folder1/","",""
```

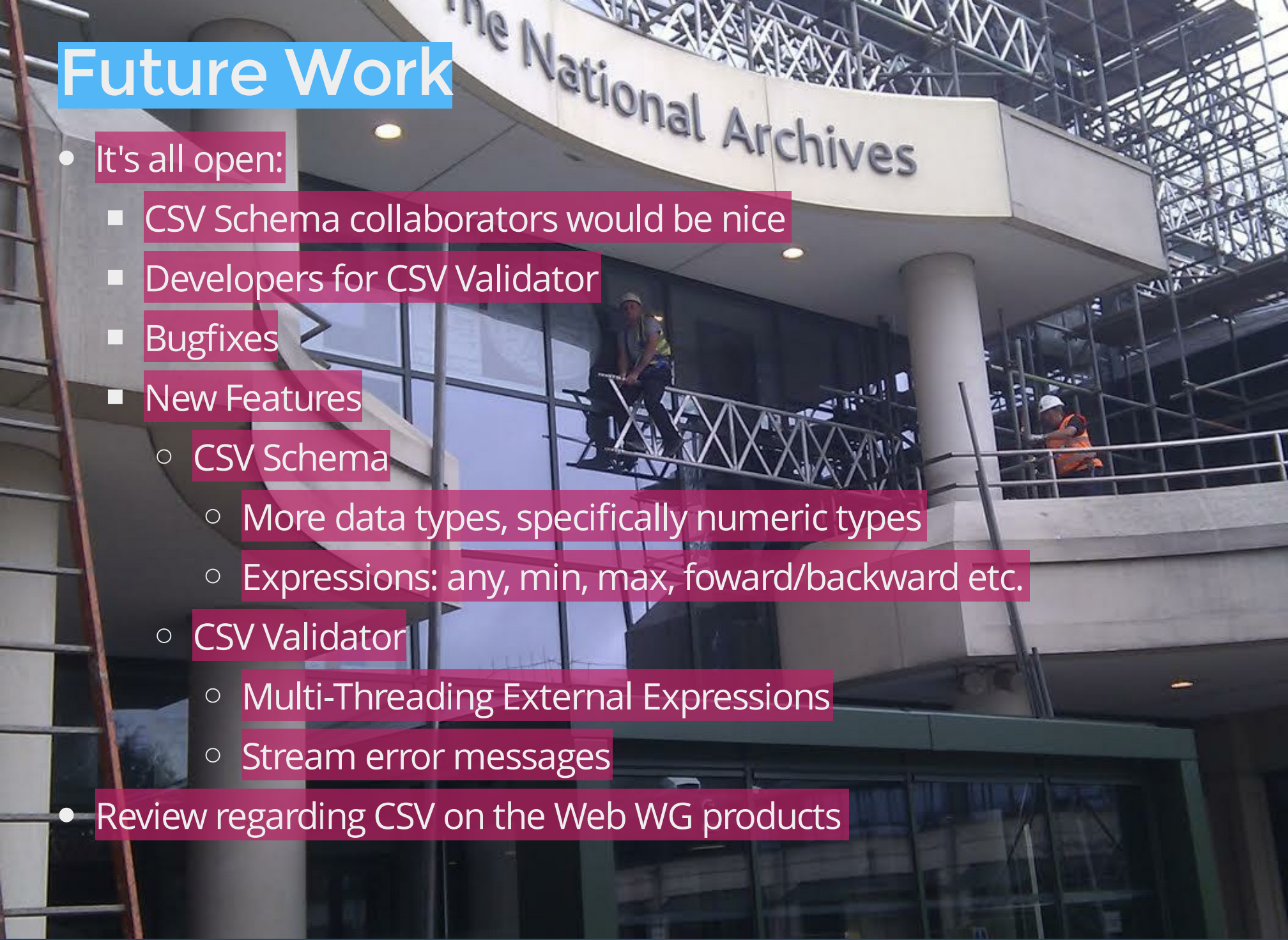
CSV Schema

```
version 1.0
id: positiveInteger unique
fn: (ends(".jp2") or ends("/")) and unique
checksum: if($fn/ends("/"), empty, checksum(file($fn, "MD5")))
classifications: regex("[0-9a-z]+(,[0-9a-z]+)*") @optional
```

The CSV Validator

- Validates CSV data against CSV Schema
- Reference Implementation
- Runs on any JVM v6+ (written in Scala 2.11)
  - Command Line Interface
  - GUI Application
  - Scala API
  - Java API
  - Open source, available under MPL v2.0
- Fast and efficient! Battle-tested against large datasets.

https://github.com/digital-preservation/csv-validator



Future Work

- It's all open:
  - CSV Schema collaborators would be nice
  - Developers for CSV Validator
  - Bugfixes
  - New Features
    - CSV Schema
      - More data types, specifically numeric types
      - Expressions: any, min, max, foward/backward etc.
    - CSV Validator
      - Multi-Threading External Expressions
      - Stream error messages
- Review regarding CSV on the Web WG products

Special Thanks to The National Archives, and staff:

Diana Newton, Peter Malewski, David Underdown, Alex Green, Nicola Welch, Richard Williams and Ian Ireland

Special Thanks to developers:

Ben Parker, David Ainslie, Andy Hicks and Jim Collins

Questions?