KItinerary Data Extraction BoF

Akademy 2019

Volker Krause

vkrause@kde.org
@VolkerKrause

Identifying Data

- HTML with structured data
- Unstructured HTML
- Barcodes (in PDF or PkPass)
- PDF
- Plain text
- Apple Wallet passes (.pkpass)
- iCal

Identifying Structured Data

- JSON-LD
  - `{"@context":"http://schema.org","@type":"FlightReservation",…}`
- Microdata
  - `<div itemscope itemtype="http://schema.org/EventReservation">`
- UIC 918.3
  - Binary barcodes starting with “#UT” or “OTI”
- IATA Barcoded Boarding Pass (BCBP)
  - ASCII barcodes starting with “M1” or “M2”

Fixing Structured Data Extraction

- HTML: structureddataextractor.cpp
- PkPass: genericpkpassextractor.cpp
- UIC 918.3: genericuic918extractor.cpp

Custom Extractors

- Define when the extractor triggers
  - e.g. mail header, pattern in barcode
- Decide on which input data it should run
  - HTML, PDF, plain text, PkPass, iCal
- Write extractor script
  - JavaScript to be run in QJSEngine

Extractor Metadata

```
[{
    "type": "html",
    "filter": [
        { "header": "From", "match": "@easyjet.com" }
    ],
    "script": "easyjet.js",
    "function": "parseHtmlBooking"
},{
    "type": "pdf",
    "filter": [
        { "property": "provider", "match": "EJU" },
        { "property": "provider", "match": "EZS" },
        { "property": "provider", "match": "EZY" }
    ],
    "script": "easyjet.js",
    "function": "parsePdfBoardingPass"
}]
```

Extractor Script Input

- HTML: KItinerary::HtmlDocument
  - DOM/XPath API
- PDF: Kitinerary::PDFDocument
  - Per page/per rect text access, image access
- Plain Text: just a string
- Ical: KCalendarCore::Event
- PkPkass:: KPkPass::Pass

Extractor Script Output

- JSON-LD following schema.org
- Single object or array
- Convenient API for creating this:
  - `JsonLd.newObject("Place")`
  - `JsonLd.newFlightReservation()`

Useful API

- KItinerary::JsApi::*
- Locale-specific date/time parsing
- Geo coordinates from Google Maps URLs
- Barcode decoding
- Bit array access
- Access to context information

Post-processing

- Saves you some work
  - Merging with structured data
  - Wikidata-based augmentation
  - Automatic arrival day rollover
- Support post-processing
  - Focus on identifiers and geo coordinates
  - Use operator-specific knowledge, e.g. “LH does not operate from HHN”

KItinerary Workbench

- Live preview of extraction results
  - Script output
  - Post-processing output
- DOM view, XPath evaluation
- Inline script editing
- Flatpak: org.kde.kitinerary-workbench
- Demo

Let’s look at your samples!