

Creating Open Data with Excel

The problem

A large majority of Open Government Data is published as Comma Separated Value (CSV) files. Much of this data sourced from Microsoft Excel. There are number challenges in preparing Excel data for publication on Open Data Portals, including:

1. Staff are not confident in structuring Excel spreadsheets to make it easy to publish.
2. Discovery metadata has to be manually prepared.
3. Metadata about the structure of the data and its validation rules are not published.
4. Data is not comprehensively checked for completeness and consistency before being published.
5. Excel's "Save As CSV" function writes files using ANSI rather the UTF-8 encoding (the dominant character encoding for the World Wide Web).

The opportunity

What if someone could push a button in Excel and prepare their data for publishing on an Open Data Portal and address the challenges above?

Imagine this process:

1. After reading a guideline on how to structure spreadsheets for Open Data, a spreadsheet is created.
2. Each column is given appropriate Data Type (e.g. Date, Currency, Text, etc.).
3. Columns are optionally given Data Validation rules (e.g. a list of valid values).
4. Data is entered into the spreadsheet.
5. A "Save as CSV Data Package" option is selected to create a web friendly CSV file and an associated metadata file.
6. These files then go through the normal organisational processes to approve publishing the resources on an Open Data Portal.

What are the benefits?

The solution is generic and can be applied to all open data initiatives around the world. The benefits include:

1. Reduced effort in creating metadata saves time for open data publishers.
2. Increased accuracy in metadata saves time for data re-users, helping them quickly determine if it is fit for their purpose and understand its structure and meaning.
3. Improved metadata supports the automatic awarding of [Open Data Institute Open Data Certificates](#) – a way to quickly show your data is easy to find, use and share (see Figure 1).
4. For organisations and community members without the ability to publish open data, their CSV Data Package could be easily published to community portals such as [Open Knowledge's data portal](#) where not only can the data be accessed but the metadata can be viewed in a user-friendly format (see Figure 2).

DATA.GOV.UK Beta
Opening up Government

Home Data Apps Interact Search for data...

Datasets Map Search Data Requests Publishers Public Roles & Salaries Spend Reports Site Analytics Reports

/ Datasets / Schools in England

Schools in England

Published by Department for Education. Licensed under **OGL** Open Government Licence.
Openness rating: ☆☆☆☆☆ Open Data Certificate: [Raw Level](#)

Information on all schools in England, including primary schools, secondary schools, university technical colleges, and sixth forms. This information contains the school name, address, phone number. This information comes from the Department for Education's <http://www.education.gov.uk/education> website. The data is updated every month.

This dataset is awarded an Open Data Certificate
Level: Raw (Final)
Date: 29/10/2014
Source: Automatically awarded by ODI
Full details >

Education

Figure 1 Open Data Certificate displayed in the United Kingdom's data portal, <http://data.gov.uk/>

data.okfn.org Tools Data Standards Docs Contribute Roadmap Vision

Comprehensive country codes: ISO 3166, ITU, ISO 4217 currency codes and many more

Comprehensive country code information, including ISO 3166 codes, ITU dialing codes, ISO 4217 currency codes, and many others. Provided as a Simple Data Format Data Package. Data - Read more

Download Data

Downloadable Metadata

Human readable Schema Metadata

Data Files

country-codes Download [Local CSV - JSON]

country-codes

249 records

name	name_fr	ISO3166-1	ISO3166-2	ITU	MARC	WMO	DS	Dial	FIFA	FIPS	GAUL	IOC	currency
Albania	Albanie	AL	ALB	8	ALB	xx	AL	355	ALB	AL	3	ALB	ALL
Algeria	Algérie	DZ	DZA	12	ALG	xx	AL	DZ	213	ALG	4	ALG	DZD
American Samoa	Samoa Am.	AS	ASM	16	SMA	xx	USA	1-684	ASA	AQ	5	ASA	USD
Andorra	Andorre	AD	AND	20	AND	xx	AND	376	AND	AN	7	AND	EUR
Angola	Angola	AO	AGO	24	AGL	xx	AN	244	ANG	AO	8	ANG	AOA
Anguilla	Anguilla	AI	AIA	660	AIA	xx	AI	1-264	AIA	AV	9	AIA	XCD
Antarctica	Antarctique	AQ	ATA	10		xx	AA	672	ROS	AV	10		
Antigua and Barbuda	Antigua-BB	AG	ATG	28	ATG	xx	AT	1-268	ATG	AC	11	ANT	XCD
Argentina	Argentine	AR	ARG	32	ARG	xx	AG	54	ARG	AR	12	ARG	ARS
Armenia	Arménie	AM	ARM	51	ARM	xx	AY	374	ARM	AM	13	ARM	AMD
Aruba	Aruba	AW	ABW	533	ABW	xx	NJ	297	ABU	AA	14	ABU	AWG
Australia	Australie	AU	AUS	36	AUS	xx	AU	61	AUS	AS	17	AUS	AUD
Austria	Autriche	AT	AUT	40	AUT	xx	OS	43	AUT	AU	18	AUT	EUR
Azerbaijan	Azerbaïdjan	AZ	AZE	19	AZE	xx	AZ	994	AZE	AT	19	AZE	AZN

Field Information

Field Name	Order	Type (Format)	Description
name	1	string	Country's official English short name
name_fr	2	string	Country's official French short name
ISO3166-1-Alpha-2	3	string	Alpha-2 codes from ISO 3166-1
ISO3166-1-Alpha-3	4	string	Alpha-3 codes from ISO 3166-1 (synonymous with World Bank Codes)
ISO3166-1-numeric	5	integer	Numeric codes from ISO 3166-1 (synonymous with UN Statistics M49 Codes)
ITU	6	string	Codes assigned by the International Telecommunications Union
MARC	7	string	Machine-Readable Cataloging codes from the Library of Congress
WMO	8	string	Country abbreviations by the World Meteorological Organization
DS	9	string	Distinguishing signs of vehicles in international traffic
Dial	10	string	Country code from ITU-T recommendation E.164, sometimes followed by area code
currency			Codes assigned by the Fédération Internationale de Football Association

Figure 2 CSV Data Package displayed in Open Knowledge's Data Package Portal, <http://data.okfn.org/data>

How would it work?

Create a spreadsheet, define column data types and data validation

- This is achieved using standard Excel functionality and resource formatting guidelines (e.g. the Queensland Government's [Resource Formatting Guide](#) or the [Guide from CSVLint](#)).
- The spreadsheet is saved as an Excel file.
- The user then selects the "Save as CSV Data Package" option that triggers the background process steps below.

Generate discovery metadata

- Read the spreadsheet properties to pre-fill the discovery metadata attributes.
- If any required metadata attributes are missing, prompt the user to provide them.
- Update the spreadsheet properties.
- Store the metadata attributes.

Generate schema metadata

- Read the spreadsheet column data types to derive the schema metadata that describes the structure of the data.
- Where no data types were provided, scan the data to determine if a stronger data type than "general" can be derived.
- Update the spreadsheet column with the stronger data type.
- Store the schema metadata.

Generate schema constraints

- Read the column data validation rules and store the schema constraint.
- If there are no data validation rules, read each column of data to determine if a data constraint can be derived. E.g. If the column only contains, "yes", "no", "unsure", then ask the user if that is a valid constraint and if so:
 - Add that data validation to the column.
 - Store the schema constraint.
- Propose a constraint. E.g. If the majority of values in a column contains, "yes", "no", "unsure", but there are a few exceptions:
 - Ask the user if the exceptions are valid or if they'd like to return to the spreadsheet and correct them.
 - (A nice visualisation of this is [CSV fingerprints](#)).

Things to work out

- Metadata standard(s) to follow e.g. DCAT, [Tabular Data Schema](#), [W3C work on CSV](#).
- File format of metadata RDF/XML, JSON, etc.
- Viability of making the code open source so it can be extended (e.g. add new metadata standards and formats, integration to publishing workflows).
- Options to make code easily accessible and integrated into Excel, avoiding the constraints often applied by large corporations with controlled operating environments.