Open Data Management in Agriculture and Nutrition

This e-learning course is the result of a collaboration between ***GODAN Action*** *partners, including* ***Wageningen Environmental Research*** *(WUR),* ***AgroKnow****,* ***AidData****,* ***the Food and Agriculture Organization of the United Nations*** *(FAO),* ***the Global Forum on Agricultural Research*** *(GFAR), and* ***the Institute of Development Studies*** *(IDS),* ***the Land Portal****,* ***the Open Data Institute*** *(ODI) and* ***the Technical Centre for Agriculture and Rural Cooperation*** *(CTA).*

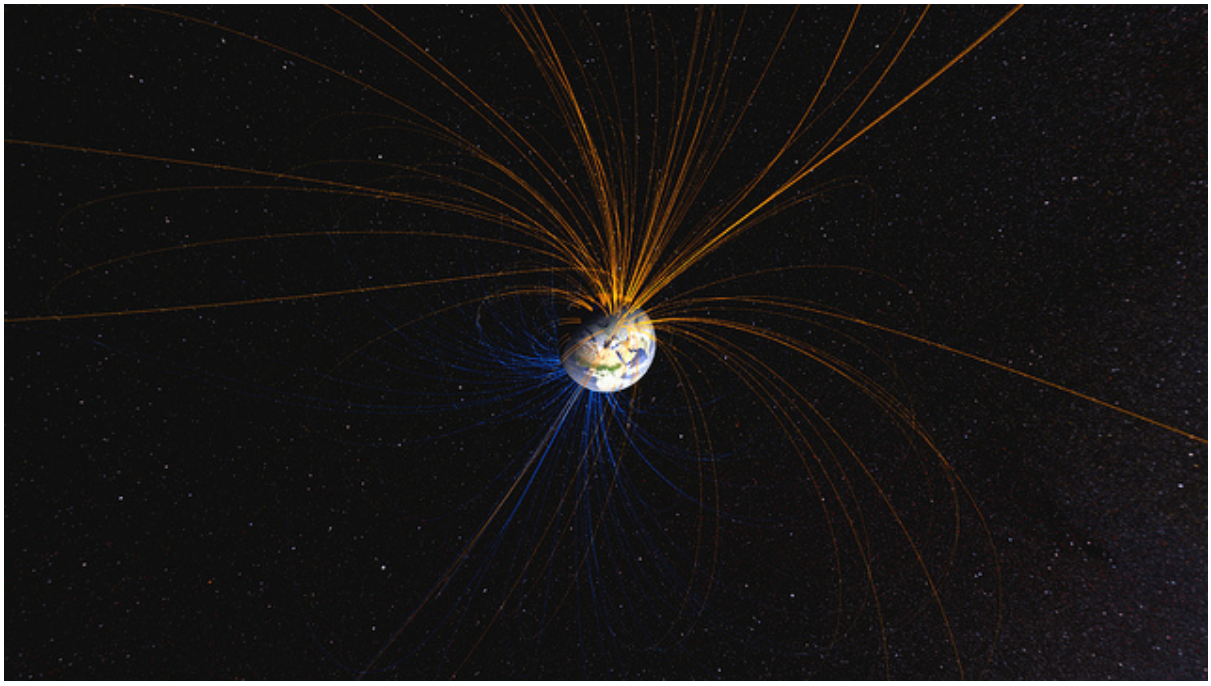
GODAN ACTION
Global Open Data for Agriculture & Nutrition

UKaid
from the British people

GODAN Action is a three-year project UK's Department for International Development to enable data users, producers and intermediaries to engage effectively with open data and maximise its potential for impact in the agriculture and nutrition sectors. In particular we work to strengthen capacity, to promote common standards and best practice and to improve how we measure impact. [www.godan.info]

UNIT 3: MAKING DATA OPEN

LESSON 3.2: MANAGING DYNAMIC DATASETS



Aims and learning outcomes

This lesson aims to;

- provide an outline of automated processes for publishing data and manual input/aggregation and provide examples
- introduce simple ways to ensure data consistency.

After studying this lesson, you should be able to:

- understand the processes that play a role to maintain and expose dynamic data sets
- be aware of existing examples of dynamic data sets
- be aware of methods to check for data consistency in dynamic data sets.

Contents

List of figures

1. Introduction

In the previous lesson, we looked at the data management principles and the broad range of activities from administrative to technical aspects. Data can be published in a variety of ways and this is influenced by whether the data are manually collected and aggregated or collected and aggregated through automated processes.

It is important to understand the differences between in the two types of datasets that will be under discussion in this unit – static and dynamic – along with the processes for managing them. The improvements we see in technology today have led to the development of tools we can use to provide access to the data, and we will provide examples for tools that can be used to expose dynamic data. This unit will also describe simple data consistency routines to use for checking and verifying dynamic datasets.

2. Dynamic and static datasets

Dynamic data denotes data that is asynchronously changed as further updates become available[1]. The opposite of this is **static data**, also referred to as persistent data, which is infrequently accessed and not likely to be modified. Dynamic data is different from streaming data in that there is not a constant flow of information; rather, updates may come at any time, with periods of inactivity in between.

In computer science, static data structures are given a fixed area of memory, which they can operate in. It is not possible to expand this fixed size in the run time. Therefore, locations of each element are fixed and known by the programme. Dynamic data structures on the other hand have a flexible area where they can operate. It is possible to expand or contract the area as required, by adding or removing elements from the data structure. It therefore would ineffective to use dynamic structures to store sets of data that will not change. Using static data structures in such case will save system resources and provide faster access to elements. Users or developers are responsible for using appropriate data structures, according to the situation.

The rise of agriculture precision technologies is changing the way farmers manage their land and livestock, such as with satellite-driven geo-positioning systems and sensors that detect nutrients and water in soil. These technologies ultimately result in the collection of more dynamic data, which is processed automatically.

Next, we will look at the various methods and processes used for publishing data – manually or automatically.

[1] https://en.wikipedia.org/wiki/Dynamic_data

2.1. Publishing data – manual input

2.1.1. Publishing as files on a website

Static datasets can be shared through static channels such as websites. Some organisations only have a few datasets to share, which they do by publishing the files on their website. The software solution used could be any Content Management System (CMS), such as Drupal, Wordpress, in combination with an internal database. For example, the Dutch city of Hengelo included one webpage to share data on their city website as shown in Figure 1.

Figure 1 The Hengelo Open Data website http://www.hengelo.nl/opendata

2.1.2. Web portal uploads

Uploading to a portal is one of the most used channels for publishing open data. Figure 2 below shows a screenshot of the CKAN portal on which the data is uploaded manually[2]. This is the manual CKAN interface, which has some restrictions, such as no reusability of similar templates and no multilingual support.

[2] https://www.europeandataportal.eu/en/providing-data/goldbook/publishing-data

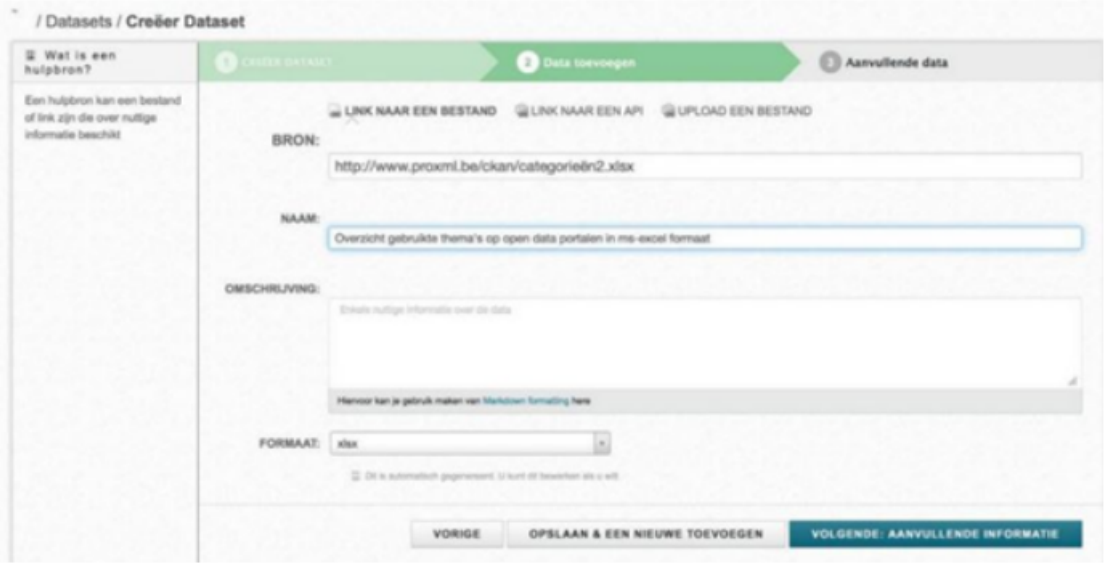


Figure 2 Example of a data set manually added to the Flemish data portal using the CKAN interface[3]

A portal provides, via a metadata catalogue, a single point of access to data. Below are examples of open data portals with agriculture data:

- EU: https://data.europa.eu/euodp/en/data
- USA: https://www.data.gov/
- UK: https://data.gov.uk/
- FAO: http://www.fao.org/data/en/
- World Bank: http://data.worldbank.org/
- Ending Rural Hunger (ERH): https://endingruralhunger.org/
- CGIAR: http://www.cgiar.org/resources/open/data-management-system/
- Food Aid – WFP: the Food Aid Information System (FAIS) makes extensive data available on world food aid. FAIS is at http://www.wfp.org/fais/ .
- Animal Nutrition – FAO: data on the nutritional composition of animal feeds is available at http://www.feedipedia.org/
- Fisheries – FAO: links to publications with fisheries data are at http://www.fao.org/fishery/statistics/en
- Food and Nutrition: for food and nutrition links in this web site click here.

All open data portals (2500+) are listed here: https://www.opendatasoft.com/a-comprehensive-list-of-all-open-data-portals-around-the-world/

California Open Data – example of a manual workflow for publishing data on an open data portal

The publishing of open data is controlled through a sequential workflow. A data coordinator or steward, is assigned one of these two roles:

[3] https://www.europeandataportal.eu/en/providing-data/goldbook/publishing-data

- Content Creator, Workflow Contributor – this role is assigned to department level data specialists who create and upload datasets.
- Editor, Workflow Moderator – this role is assigned to a department level data steward who reviews and approves data prior to release.

The publishing process is described fully at https://office-of-digital-innovation.github.io/data-training/publishing/.

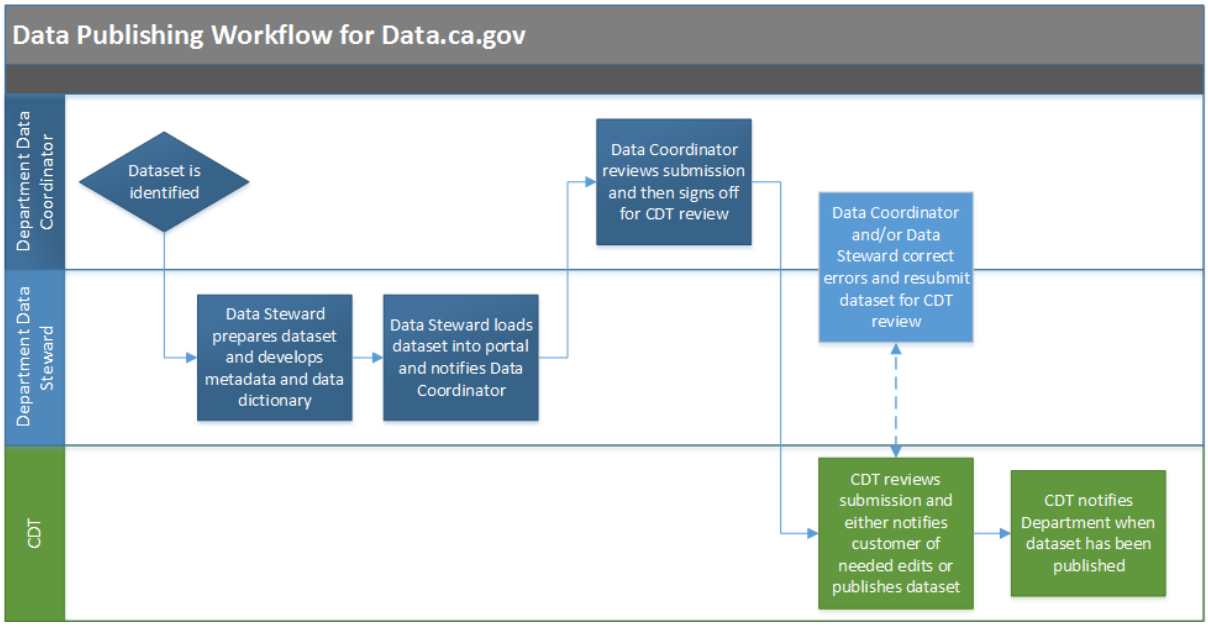


Figure 3 California Open Data provides an example of a manual workflow for publishing data on an open data portal

2.2. Publishing data – automated processes

Sources of automated data. As the internet of things (IoT) expands, particularly with the development of cheap sensors the automation of data collection possibilities expands. With precise sensors, farmers can automatically collect data on weather, soil, air quality and crop maturity, enabling them to make smarter decisions. Devices such as mobile phones are also being used to collect passive data: e.g., the WeatherSignal[4] mobile application uses native phone sensors to measure local atmospheric conditions, which are then displayed on their live-updating weather map. There is now potential for sensors in cities, homes, cars, cell towers and even mobile phones to contribute observational data that could also be fed into forecast models. As automated datasets continue to grow exponentially, new technologies for more sustainable dissemination have been introduced.

[4] https://play.google.com/store/apps/details?id=com.opensignal.weathersignal&hl=en

Data automation is the process of updating data on your open data portal programmatically, rather than manually. Automating the process of data uploading is important for the long-term sustainability of your open data program. Any data that is updated manually risks being delayed because it is one more task an individual has to do as part of the rest of their workload.[5]

Automating the publication of data will provide your users with routine and predictable data updates, and provide efficiencies to the publishing process. There are three common elements to data automation – Extract, Transform, and Load:

- *Extract*: the process of extracting your data from one or many source systems
- *Transform*: the process of transforming your data into the necessary structure, such as a flat file format like a CSV; this could also include things like changing all state abbreviations to the full state name
- *Load*: the process of loading the data into the final system, in this case the open data portal.

Automated publishing processes used to be highly customised, with publishers writing an Extract–Transform–Load (ETL) script from scratch, which is labour intensive. ETL processes do these three tasks: extracting the raw data from the original source, transforming it into a more useful format, and loading it into the open data portal. Each one of these processes is critical to fully automating your data uploads, and doing so successfully. Figure 4 is a diagram of a standard ETL process.

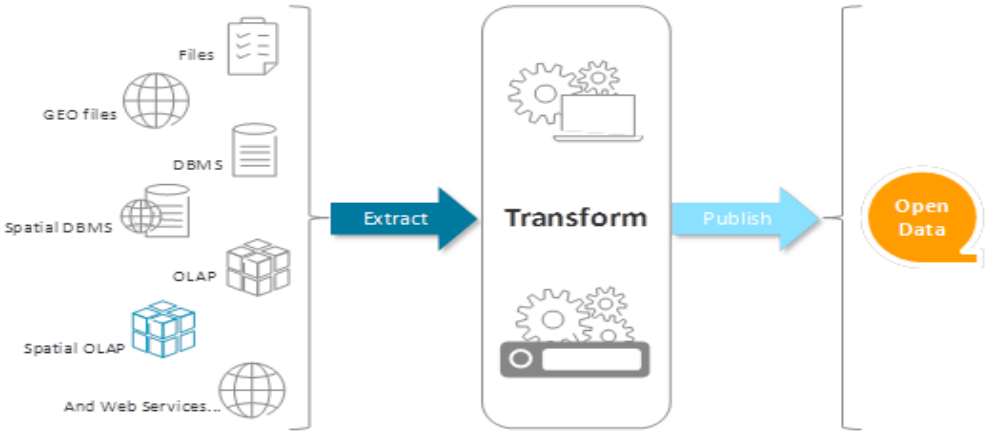


Figure 4 An ETL process: it extracts raw data from the original source, transforms it into a more useful format, and loads/publishes it into the open data portal[6]

[5] https://support.socrata.com/hc/en-us/articles/212871018-Data-Automation-Overview
[6] https://www.europeandataportal.eu/en/providing-data/goldbook/technical-preparation-and-implementation

Data automation strategy. Firstly it is important to determine a general data automation strategy for your organisation. Having a strategy beforehand will help you engage the right people, the right tools, at the right time within your organisation.

Identify who owns data automation in your organisation. Different groups will own different parts of the ETL process:

Centralised: the central IT department owns the full ETL process and all data automation (see Figure 5).

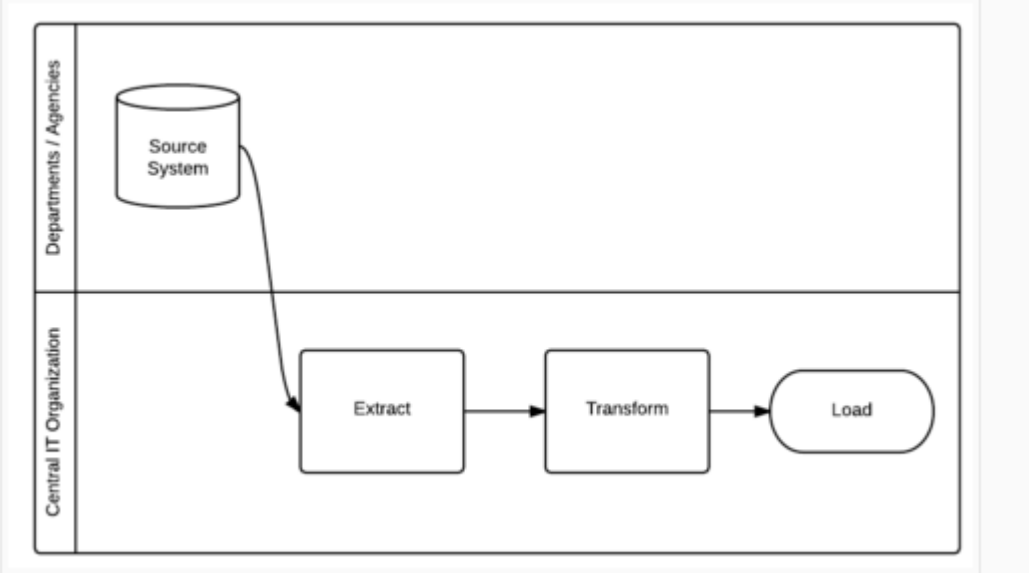


Figure 5 A centralised ETL process

Hybrid: this model may vary, but often the individual agencies/departments will own the extract and transform processes, and the central IT department will own the loading process (see Figure 6).

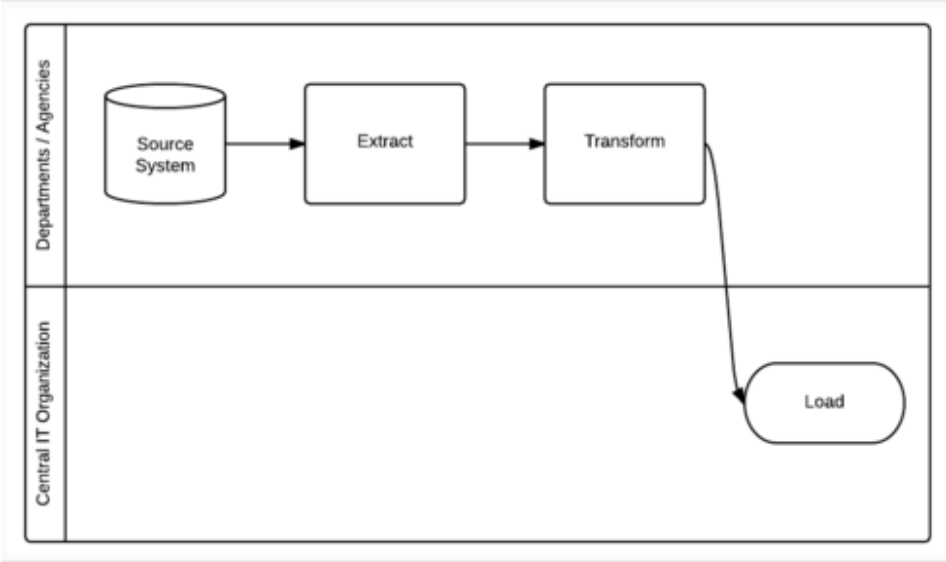


Figure 6 A hybrid ETL process

Decentralised: the individual agencies/departments will each own their own ETL process (see Figure 7).

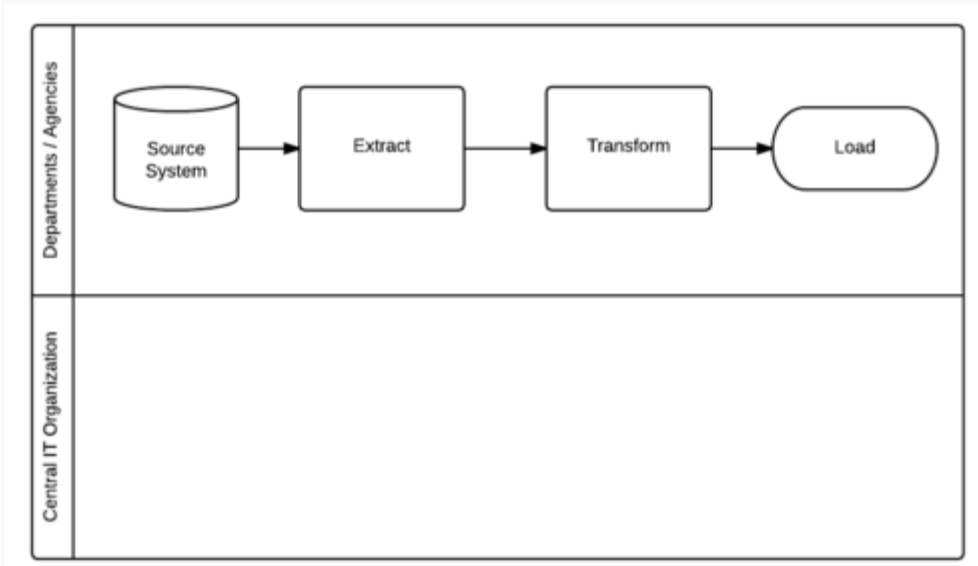


Figure 7 A decentralised ETL process

When you have defined where the ETL stages occur within the organisation, you will need to determine where your automation steps will fit into your publishing workflow. As you create your publishing flowchart, make sure to clarify the steps for Extract, Transform, and Load, as well as the individuals who will own those steps.

Selecting the data to be automated. The more that you adopt an 'automate by default' approach to uploading data, the less resources you will need over the long term for maintaining high data quality. Here are some tips for finding candidate datasets for automatic uploads:

- Is the dataset updated quarterly or more frequently?
- Are there transformations or any form of manipulation that needs to be done to the dataset prior to uploading?
- Is the dataset large (greater than 250MB)?
- Can you only get the changed rows for each subsequent update (rather than the full file)?
- Is it possible to get data from the source system, rather than from an individual?

Datasets that receive a strong 'yes' to any of the questions above are great candidates for automating updates because automation can remove the risk of a lack of time and resources for manually preparing the datasets.

Steps for automation. Once you understand the landscape of data automation within your organisation, you can start putting your automation strategy to use. Here are some steps to get started:

1. *Identify data*: select one or two high-value datasets where getting access to the source systems will be easy (i.e. start with the 'low-hanging fruit').

2. Refer to your source system inventory to *determine which source systems* you already have access to.
3. *Determine access*: determine how the relevant department or agency will obtain the data. Will it be through a SQL query, download of a CSV, etc? The data custodian would likely be the best resource for accessing a dataset's source system.
4. *Define transformations*: outline any transformations required for that dataset. This might be as simple as changing complex acronyms to full-text names, or as complicated as transforming a relational database into a flat-CSV file.
5. Work with both the data steward and the data custodian to *understand which fields* need to be pulled and how they should be formatted for publication.
6. *Develop and test the ETL process*: based on the requirements defined in steps 2 and 3, select an ETL publishing tool, and publish the dataset to the open data portal. Confirm that the dataset has been successfully loaded or updated through your process without errors.
7. *Schedule*: Schedule your dataset for timely updates.
8. *Refer to metadata fields* about data collection, refresh frequency, and update frequency.

3. Data Consistency

Consistent data is data that is technically correct and fit for statistical analysis. This is data that has been checked for missing values, special values, (obvious) errors and outliers, which are either removed, corrected or imputed. The data is consistent with constraints based on real-world knowledge about the subject that the data describes[7].

Data quality is assessed by applying verification and validation procedures as part of the quality control process. Verification and validation are important components of data management that help to ensure data is valid and reliable. The United States Environmental Protection Agency defines data verification as the process of evaluating the completeness, correctness, and compliance of a dataset with required procedures to ensure that the data is what it purports to be. Data validation follows data verification, and it involves evaluating verified data to determine if data quality goals have been achieved and the reasons for any deviations.

Principles of data quality need to be applied at all stages of the data management process (capture, digitisation, storage, analysis, presentation, and use). There are two keys to the improvement of data quality – prevention and correction. Error prevention is closely related to both the collection of the

[7] E de Jonge and M van der Loo, 2013, *An introduction to data cleaning with R* Statistics Netherlands, The Hague, Netherlands. Available at: https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf

data and the entry of the data into a database. Although considerable effort can and should be given to the prevention of error, the fact remains that errors in large datasets will continue to exist and data validation and correction cannot be ignored.

3.1.1. Checklists when considering the consistency of your data

1. Is your data set complete? Every data set should:
   - Contain a header row with a single description of what is shown. This means that once a data set structure is in place, it should not change when sources are added. In the metadata, the header should be described.
   - Be labelled with a version number. Once an update is done the data set should get a new version number in order for the audience to keep track of changes
   - Contain information about its origin. What is the data about, where does it come from and for what purpose has it been published?
   - Be given a status: Draft, validated, final

2. Is the data clean? Check the following aspects:
   - empty fields
   - dummy data and default values – are they correct?
   - wrong values
   - double entries
   - privacy sensitive information

Different examples of errors and inconsistencies in data, and how they can be fixed with schema validation and more advanced tools like Open Refine[8] are covered fully in Unit 2, Lesson 2.2.

3. Is the data accurate? Is your data set accurate? The most important aspects regarding accuracy are:
   - Is the data accurate enough for its potential purpose?
   - Does its accuracy affect its reliability?
   - Are the choices concerning interval described?
   - Does the data need aggregation or disaggregation?

Documentation is key to good data quality. Without good documentation, it is difficult for users to determine the fitness for use of the data and difficult for custodians to know what and by whom data quality checks have been carried out. Documentation is generally of two types, and provision for them should be built into the database design. The first is tied to each record, and records what data checks have been done and what changes have been made and by whom. The second is the metadata that records information at the dataset level.

[8] http://openrefine.org

4. Exposing Dynamic Datasets

4.1. APIs and open data

An API (Application Programming Interface) is a collective name for linking systems through a programming interface. An API can be used to make open data available by providing a reuser with direct access to the provider's open data.

Data usually contains a specific set of fields or columns and codes, which reusers need to understand to work effectively with the data. If it is served through an API or service, reusers will also need to understand how the API works[9].

The Open Data Institute suggests that the technical documentation that you must provide with an API should include:

- **format documentation** about the data formats that you are providing, possibly including schemas for any vocabularies that you use;
- **code lists** that provide more details about each of the codes that are used within your data; one way to provide this information is to have a URL that provides documentation about each code and to link to that URL within the data;
- **service documentation** that describes the way any API that you provide works; this might include links to machine-readable service descriptions if applicable.

Equipped with this information, reusers should be able to understand the data that you are publishing and how to create applications that use it.

4.1.1. Example of an agriculture API

Food Security Portal[10]: This IFPRI[11] site contains over 40 indicators related to food security, commodity prices, economics, and human wellbeing. Much of this data is available for every country in the world and goes back over 50 years. They draw from public, authoritative data sources like the World Bank, the FAO, UNICEF, and others, as well as their own data.

In order to make the data contained on the site as useful as possible, it is available to freely download through the Data API and to aggregate, mash-up and share. The portal is designed to pool such information in structured ways and check for data quality and relevance.

---

[9] https://theodi.org/guides/engaging-reusers
[10] http://www.foodsecurityportal.org
[11] http://www.ifpri.org

4.1.2. Example of a weather data API

OpenWeatherMap[12] : The site publishes open weather data through an API for developers which makes it easy to embed weather information into variety of applications including web and mobile weather applications, and solutions for insurance, agriculture, sport and many other areas.

The OpenWeatherMap service collects data from professional and private weather stations. Today they have more than 40,000 weather stations; most are professional stations which are installed in airports, large cities, etc.

4.1.3. The technology behind APIs

APIs are driven by a set of specific technologies, making them easily understood by developers. This type of focus means that APIs can work with any common programming language, with the most popular approach to delivering web APIs being REST (REpresentational State Transfer).
REST takes advantage of the same internet mechanisms that are used to view regular web pages, giving it many advantages that can result in faster implementations and easier for developers to understand and use[13]. REST APIs allow you to take data and functionality that may already be available on your website and make them available through a programmatic API that both web and mobile applications can use. Then, instead of returning HTML to represent the information like a website would, an API returns data in one of two possible formats:

- Extensible Markup Language (XML), or
- JavaScript Object Notation (JSON).

Developers can then take this data and use it in web and mobile applications. However XML and JSON are easily consumed by spreadsheets and other tools non-developers can use as well, making APIs accessible by potentially anyone.

Summary

Data can be published in a variety of ways and this is influenced by whether the data are collected and aggregated manually or through automated processes.

Dynamic data denotes data that is asynchronously changed as further updates become available.

Datasets can be published through the means of websites, portals or APIs.

Data automation is the process of updating data on your open data portal programmatically, rather than manually. Automating the process of data

[12] http://www.openweathermap.org
[13] https://project-open-data.cio.gov/api-basics/

uploading is important for the long-term sustainability of your open data programme.

The three common stages in data automation are **Extract, Transform, and Load**, or ETL:

- Extract: the process of extracting your data from one or many sources systems
- Transform: the process of transforming your data into the necessary structure, such as a flat file format like a CSV
- Load: the process of loading the data into the final system.

Consistent data are data that are technically correct and that are fit for statistical analysis. The data have been checked for missing values, special values, (obvious) errors and outliers are removed, either corrected or imputed. Principles of data quality need to be applied at all stages of the data management process (capture, digitisation, storage, analysis, presentation, and use). When considering the consistency of your data you must check that: your data set is complete, the data is clean, and the data is accurate.

An **API** (Application Programming Interface) is a collective name for linking systems through a programming interface. An API can be used to make open data available by providing a reuser with direct access to the provider's open data.

In the next lesson 3.3 Creating and Managing Open Data Repositories, we will explore the various options available for hosting an open data repository and how to manage it.

Further Readings

- Evans, S. (undated) Robots Set to Transform the Automotive and Agricultural Industries – Interview with Dr Robert Fitch (Australian Centre for Field Robotics). Available at: http://marketclarity.com.au/acfr-robots-set-to-transform-the-automotive-and-agricultural-industries/
- GODAN Global Data Ecosystem Publication http://www.godan.info/documents/data-ecosystem-agriculture-and-food
- Mill, E. (2016) An Introduction to Open Data and APIs (video). DigitalGov. Available at: https://www.youtube.com/watch?v=taTdJ6oOZX4