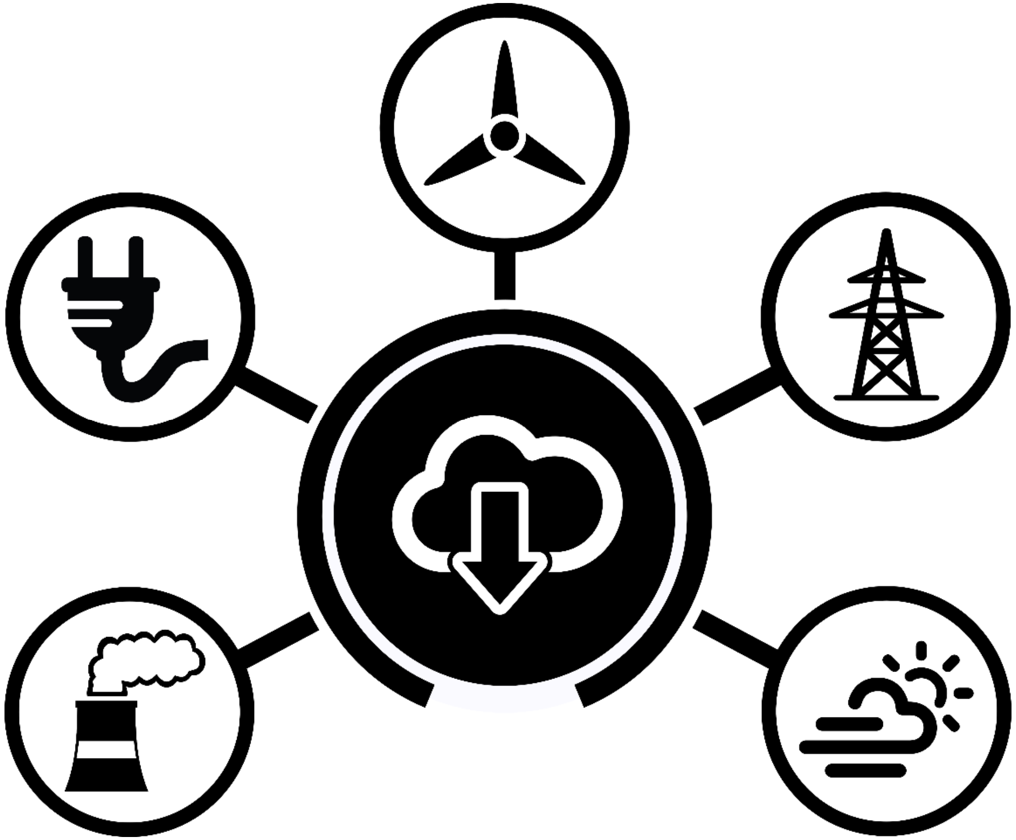


Open Power System Data

1. Project overview
2. IT concept
3. Legal issues

1. Project Overview

Lion Hirth

October / November 2015 | Open Power System Data Webinar

Background and motivation

- Energy research, policy advice, and investment decisions are based on numerical models
- These models are data-intensive
  - Information about existing power stations
  - Yearly electricity consumption
  - (Hourly) time series of load, wind and solar power generation
  - Depending on the model, much more input data might be required: hydro reservoir inflows, district heating demand, system service requirements, interconnector capacity, ...
- Much of these data are publicly available, but ...
  - ... sometimes hard to find
  - ... often poorly documented
  - ... almost always tedious to process
  - ... subject to unclear and/or restrictive licenses (terms of use)

Open Power System Data – in a nutshell

The project

- developing a platform for free and open data for electricity system modelling
- collect, check, process, document, and provide publicly available data
- not: collect new data

Types of data

- focus: power plants; time series of load and wind/solar generation
- possibly extend scope over time

Geographic scope

- focus: Germany
- extend coverage over time

Open data

- free of charge
- without restrictions on its use
- open license

The target group: you

- modelers and analysts in academia, consulting, and industry
- not: journalists, traders
- → no visualization, no real-time data

A service provider for the modelling community...

- ... providing a public good

Objectives

1. *Efficiency*: avoid double work.

2. *Quality*: improve the quality of data and documentation.

3. *Legal certainty*: using a well-established open license.

The project team

neon neue energieökonomik

DIW BERLIN

Europa-Universität Flensburg

TU berlin WIP

Funded by:

Bundesministerium für Wirtschaft und Energie

Related projects

EnergyMap.info
A user-friendly database of all renewable-based power generators in Germany.

Energy Charts
Close-to-real time charts of German power generation and prices, operated by Fraunhofer ISE.

Renewables.ninja
Generate wind and solar profiles from MERRA weather data – globally. Run by Stefan Pfenninger and Iain Staffell.

Enipedia
A wiki-based collection of global power plant data run by Chris Davis (TU Delft).

EEX Transparency
A data platform for REMIT compliance, covering Germany, Austria, and Czech Republic.

Open Energy Modelling Initiative
A wiki-based collection of open energy data sources.

Paul-Frederik Bach
A collection of time series data provided by Paul-Frederik Bach. Covers wind and solar power in-feed profiles of nine countries, some of which dating back to 2006.

SciGRID
An open transmission grid topology developed by NextEnergy (under construction). Derived from OpenStreetMap, published under ODbL.

Nationale Informationsplattform
An upcoming (2017) official German data platform announced in the power market White Paper (Measure 17), apparently modelled after RTE's data site.

Commission for Energy Regulation in Ireland
A validated Excel and PLEXOS Database of thermal generator characteristic, hourly wind profiles and hourly demand profiles in Ireland.

ENTSO-E Transparency
ENTSO-E's transparency platform.

IAEE EDS
A list of energy data links compiled by the International Association for Energy Economics.

Project schedule

2015					2016												2017						
Aug	Sep	Okt	Nov	Dez	Jan	Feb	Mrz	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez	Jan	Feb	Mrz	Apr	Mai	Jun	Jul
	x *Info Website*						**x** *Beta Version*						**x** *Public Version*									**x** *Final Version*	
		1 *Workshop*						**2** *Workshop*						**3** *Workshop*						**4** *Workshop*			

- Step by step
- Start with Germany – later more countries
- Start with "core data" (power plant data and load/wind/solar time series) – later (possibly) more types of data

On the side: info pages

In addition...

- ... to the data platform, we are providing information on our website

List of data sources *(/data-sources)*

- a link list to European power system data
- this is *not* a list of data sources we will use
- we will focus on some types of data and will use mostly official data sources

List of data sources *(/data-projects)*

- similar and related or complementary projects and initiatives

Legal situation *(/legal)*

- a layperson summary of the terms of use for energy data today
- explaining license options for open data

Our IT concept *(/it)*

- a summary of our IT approach

Stakeholder engagement

Potential users

- Workshops
- Webinars
- “Publish early, publish often”
- User group / beta version access

Data owners

- Bilateral meetings
- BMWi in the loop

Vision: an open data community

- Data upload and sharing
- An active community around power system data

Data aggregation and validation – no data creation

- The project collects existing, publicly available data
- We rely on official sources (statistical offices, Eurostat) or semi-official sources (TSOs, public agencies, sector associations)
- We correct only obvious errors
  - missing commas
  - alphabetical characters in numerical data fields
  - GW-scale rooftop solar plants
- Translation of national sources into English
- What is an “obvious error”? – details in working groups this afternoon

Work packages

Thermal and hydro power plant data

- Plant-level data of thermal and hydro power plants
- Data sources for DE: BNetzA, UBA

DIW BERLIN

TU berlin WIP

Wind, PV, biomass plants – Weather data

- Plant-level data of wind, pv and biomass power plants
- Weather data from re-analysis models

Europa-Universität Flensburg

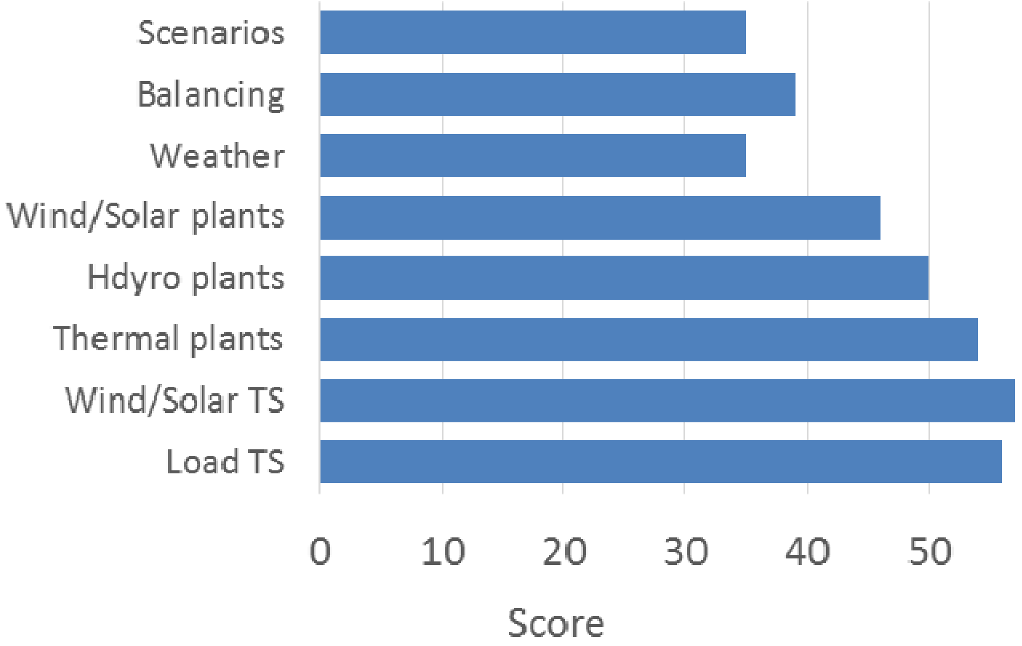
Time series

- Load (electricity consumption)
- Wind power generation
- Solar power generation

neon neue energieökonomik

Survey results: content

What kind of data should be included?



Further suggenstions (top items): CHP / heat data or time series; conventional generation time series; hydro inflow; NTCs, fuel and CO_2 prices, transmission grid topology, costs

Survey results: user wishlist

The project collects existing, publicly available data – but does not create new data.

This is a list of requirements that were mentioned (not a list of what this project will deliver):

- CHP data and heat demand time series: availability problematic
- Power plant cost data: several studies available (DIW, IRENA)
- Grid topology: SciGRID
- Hydro inflow: potentially
- NTCs: maybe – but how handle the transition to FBMC?
- Power prices: availability problematic for many countries; legal barriers
- CO_2 prices: out of scope

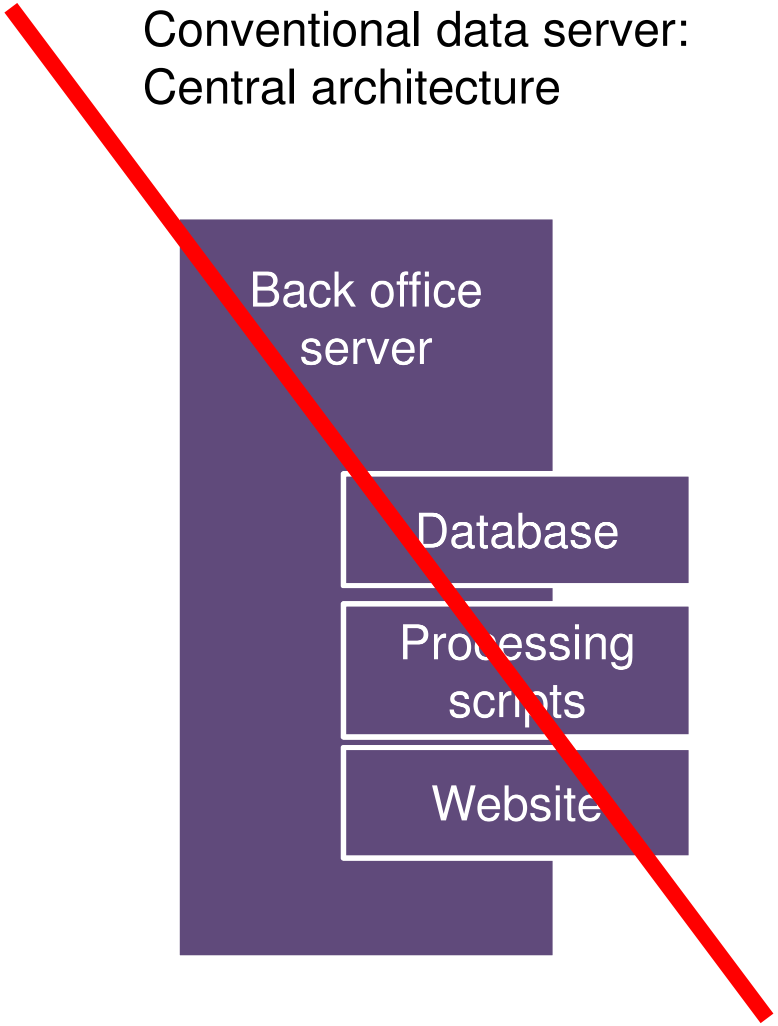
2. IT Concept

Lion Hirth

October / November 2015 | Open Power System Data Webinar

Decentral architecture

Conventional data server:
Central architecture



Conventional data server

Advantages:
- Homogenous data (all in one DB)
- Sophisticated filters possible

Disadvantages:
- Intransparent
- Difficult to maintain
- Error prone
- Versioning hard to do
- Often bad performance

Decentral architecture

Decentral architecture

- Data stored in individual packages
- Each package consists of CSV file, generation script and meta data
- Website only links to the packages

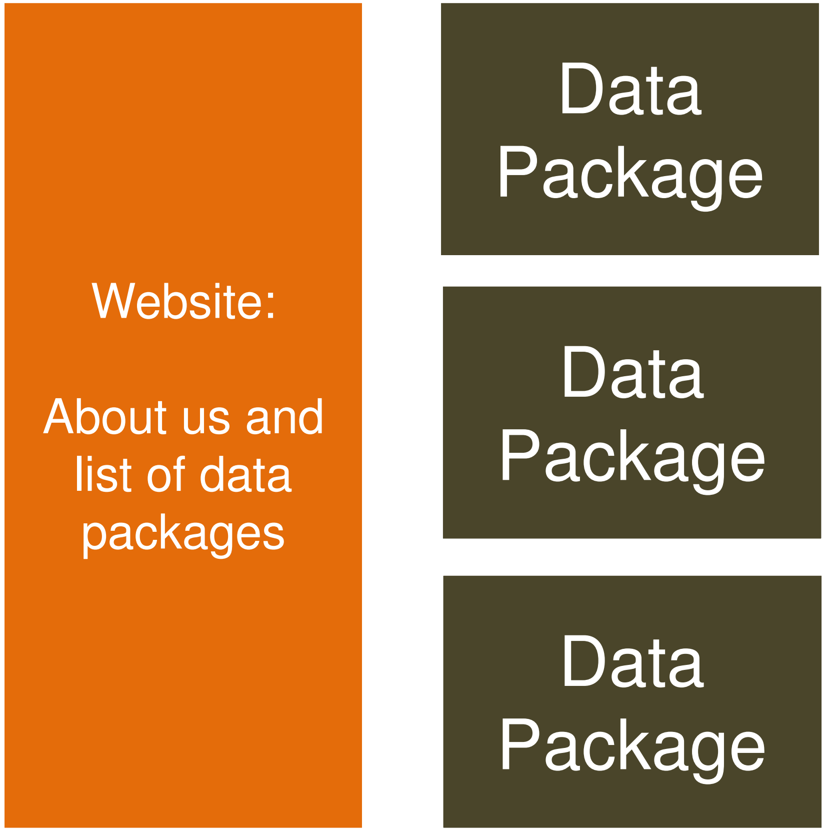
Advantages:

- Full transparency of data sources and processing
- Less error prone
- Enables opening up the platform for external package providers

Disadvantages:

- Filters are not (or hardly) possible

OPSD approach:
Decentral architecture



Website:

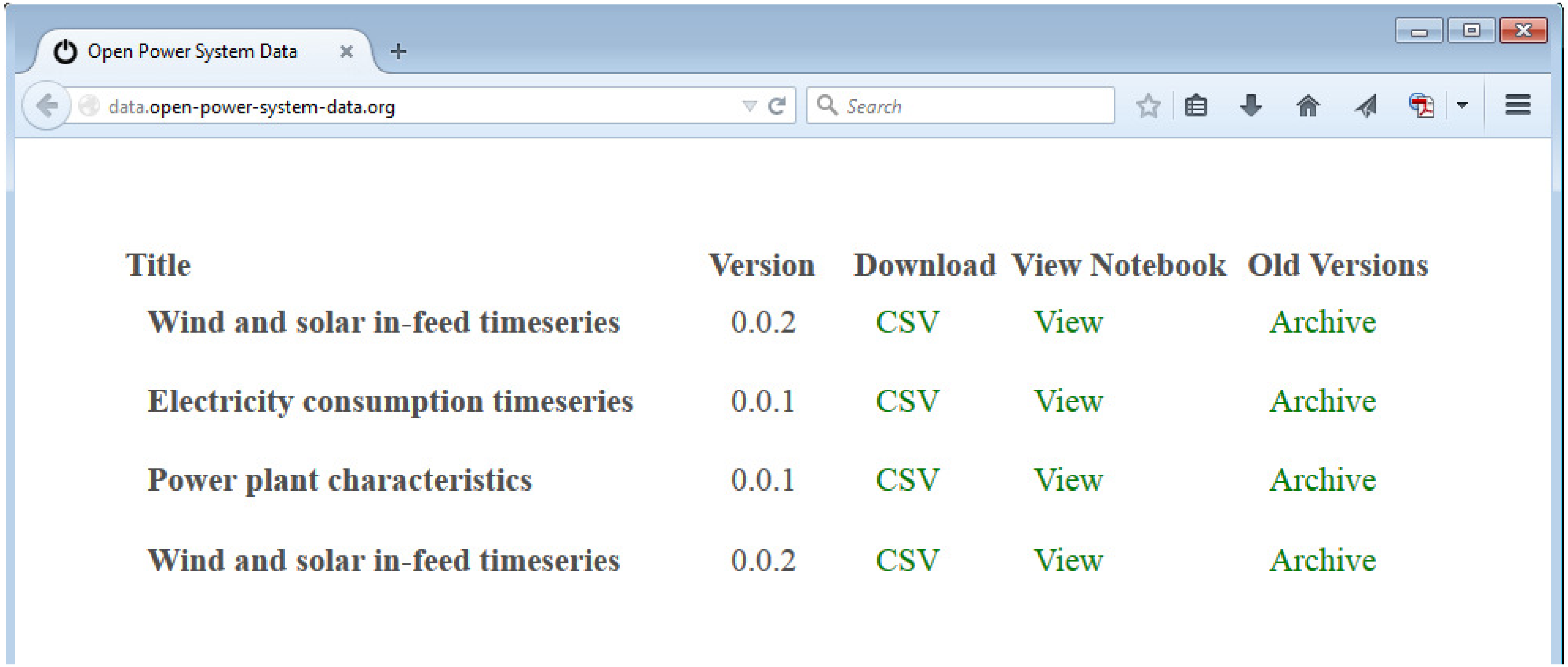
About us and list of data packages

Data Package

Data Package

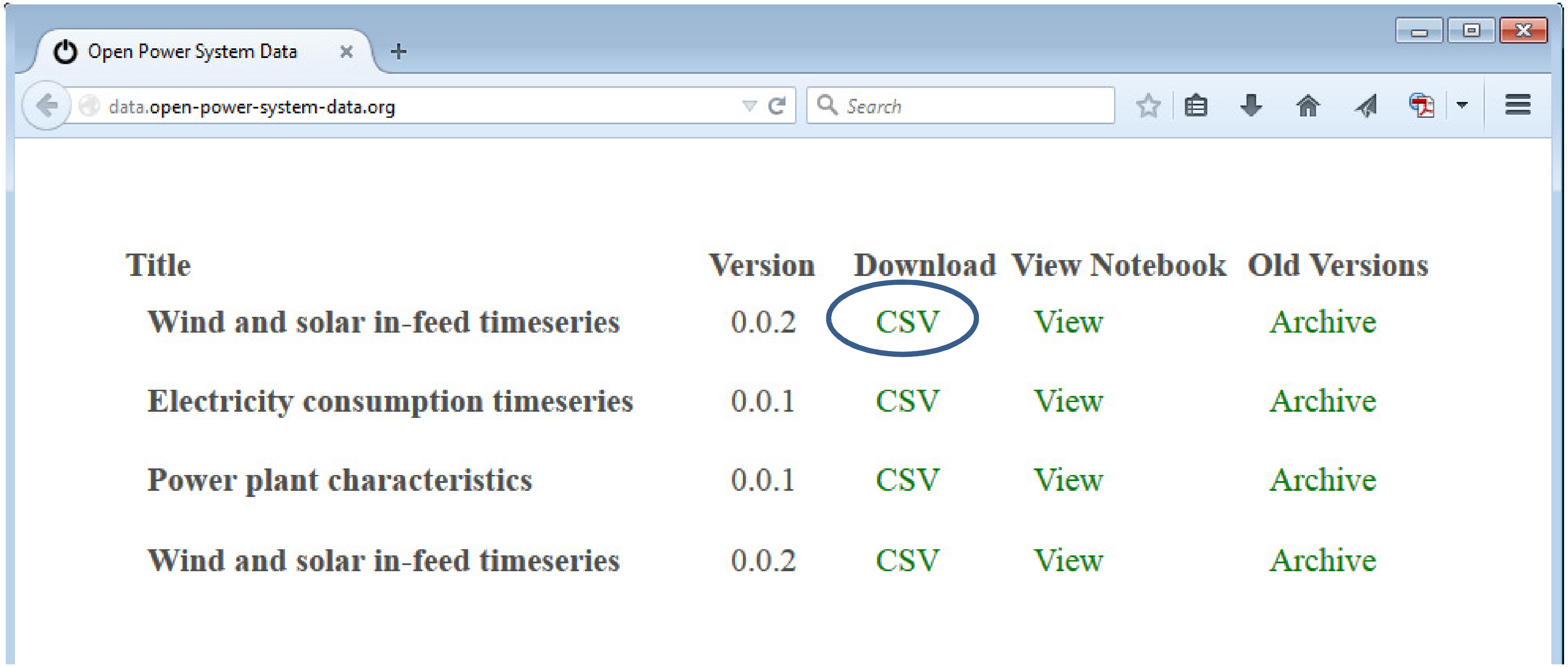
Data Package

OPSD from user‘s perspective



Title	Version	Download	View Notebook	Old Versions
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive
Electricity consumption timeseries	0.0.1	CSV	View	Archive
Power plant characteristics	0.0.1	CSV	View	Archive
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive

OPSD from user's perspective



Title	Version	Download	View Notebook	Old Versions
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive
Electricity consumption timeseries	0.0.1	CSV	View	Archive
Power plant characteristics	0.0.1	CSV	View	Archive
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive

Data formats: tabular data – CSV

- CSV files as core data format
  - Text based format
  - Small file size
  - Can be processed in all sorts of programs
  - We use comma as column separator, point as decimal point
  - Field names in first row

- Character set: UTF-8 or ISO-8859-1?

- Should we provide Excel files in addition?

Example:

```
Timestamp,PV,Wind
2015-02-01 08:15:00+1:00,134,432
2015-02-01 08:30:00+1:00,257,454
2015-02-01 08:45:00+1:00,369,467
...
```

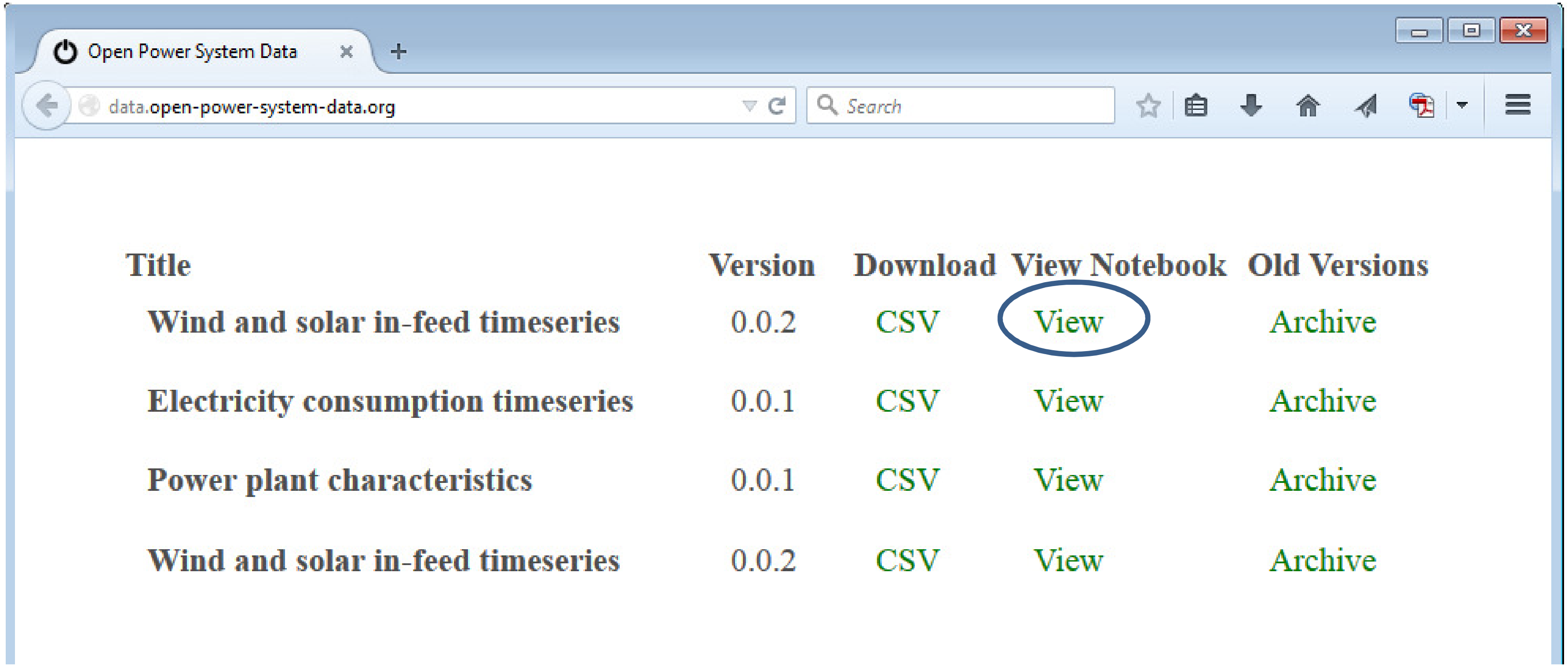
Data formats: meta data – JSON

- Data Package standard of the Open Knowledge Foundation
  - datapackage.json
  - data.csv
  - (README.md)

Example of datapackage.json:

```
{
  "title" : "A dataset title",
  "licenses" : [
      {"license": "GNU-GPL-3.0"},
      {"license": "CC-BY-SA-4.0"}
  ],
  "sources" : [...],
  "resources": [
    {"path": "data.csv"}
  ]
}
```

OPSD from user‘s perspective

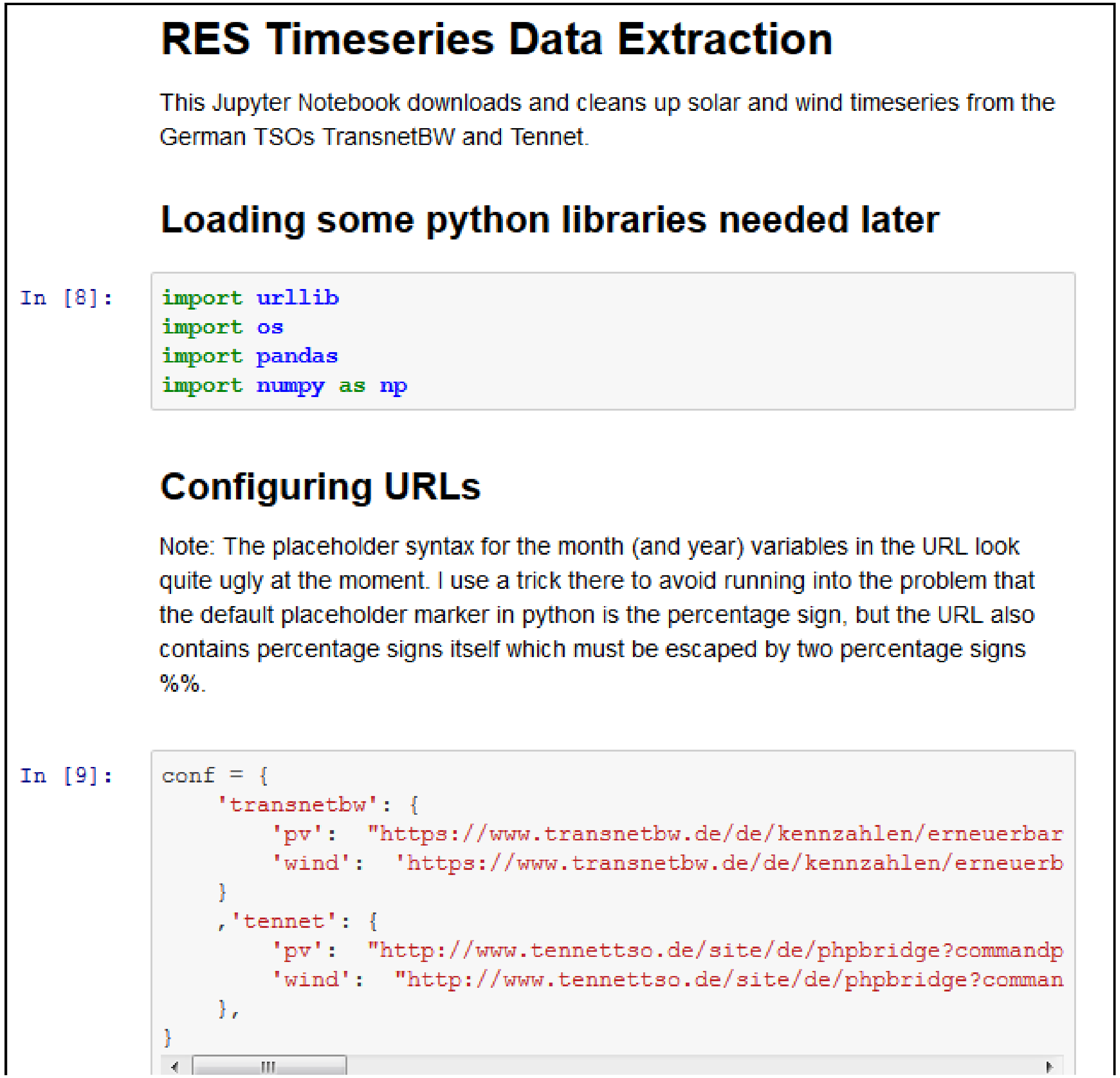


Title	Version	Download	View Notebook	Old Versions
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive
Electricity consumption timeseries	0.0.1	CSV	View	Archive
Power plant characteristics	0.0.1	CSV	View	Archive
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive

Jupyter Notebook „scripts“

- **Why Jupyter Notebook?**
  - One file for code and documentation
  - Python as programming language: Open Source and frequently used in the energy modelling community
  - Established format with good availability of tools (nbviewer, github visualisation, runs on all operating systems)
- **Usage in the OPSD project:**
  - Download, processing and generation of data packages in one notebook file
  - Scripts are provided as OpenSource: users can make their own adjustments and execute the scripts themselves
  - Transparency of scripts benefits academic accountability

Jupyter Notebook:
RES Timeseries Data Extraction



RES Timeseries Data Extraction

This Jupyter Notebook downloads and cleans up solar and wind timeseries from the German TSOs TransnetBW and Tennet.

Loading some python libraries needed later

In [8]:

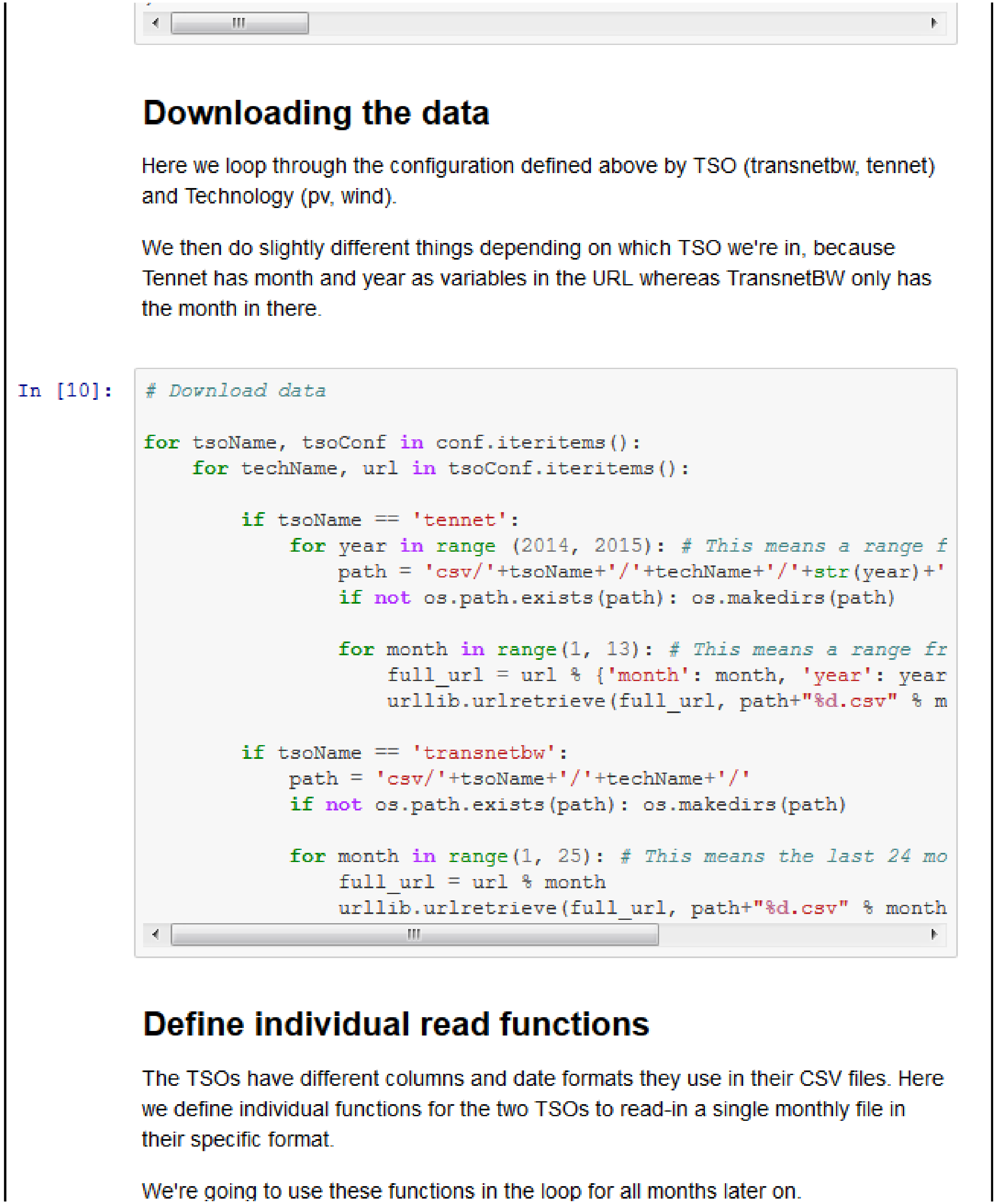
```
import urllib
import os
import pandas
import numpy as np
```

Configuring URLs

Note: The placeholder syntax for the month (and year) variables in the URL look quite ugly at the moment. I use a trick there to avoid running into the problem that the default placeholder marker in python is the percentage sign, but the URL also contains percentage signs itself which must be escaped by two percentage signs %%.

In [9]:

```
conf = {
    'transnetbw': {
        'pv':   "https://www.transnetbw.de/de/kennzahlen/erneuerbar
        'wind':   'https://www.transnetbw.de/de/kennzahlen/erneuerb
    }
    ,'tennet': {
        'pv':   "http://www.tennettso.de/site/de/phpbridge?commandp
        'wind':   "http://www.tennettso.de/site/de/phpbridge?comman
    },
}
```



Downloading the data

Here we loop through the configuration defined above by TSO (transnetbw, tennet) and Technology (pv, wind).

We then do slightly different things depending on which TSO we're in, because Tennet has month and year as variables in the URL whereas TransnetBW only has the month in there.

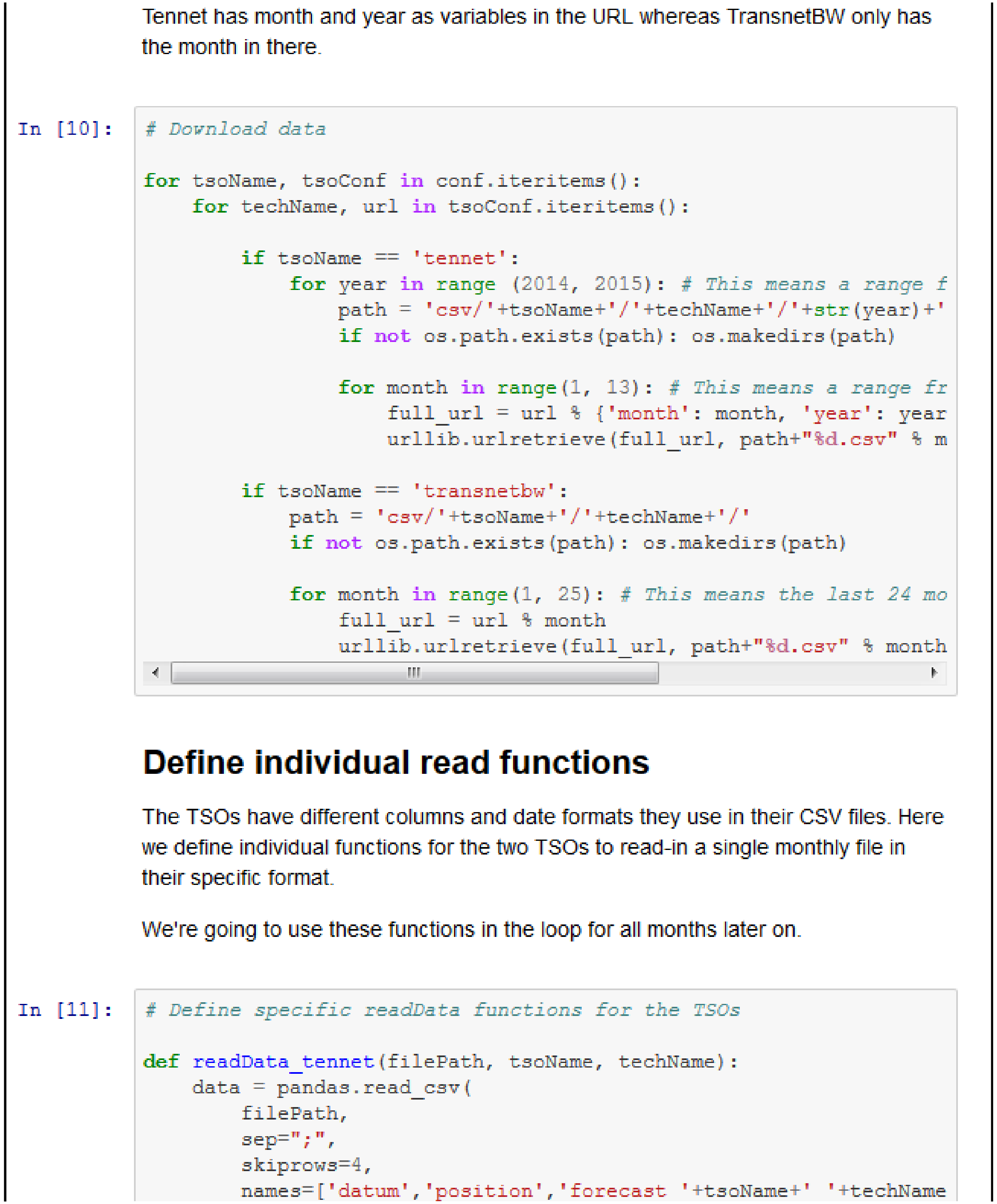
In [10]:

```
# Download data

for tsoName, tsoConf in conf.iteritems():
    for techName, url in tsoConf.iteritems():

        if tsoName == 'tennet':
            for year in range (2014, 2015): # This means a range f
                path = 'csv/'+tsoName+'/'+techName+'/'+str(year)+'
                if not os.path.exists(path): os.makedirs(path)

                for month in range(1, 13): # This means a range fr
                    full_url = url % {'month': month, 'year': year
                    urllib.urlretrieve(full_url, path+"%d.csv" % m

        if tsoName == 'transnetbw':
            path = 'csv/'+tsoName+'/'+techName+'/'
            if not os.path.exists(path): os.makedirs(path)

            for month in range(1, 25): # This means the last 24 mo
                full_url = url % month
                urllib.urlretrieve(full_url, path+"%d.csv" % month
```

Define individual read functions

The TSOs have different columns and date formats they use in their CSV files. Here we define individual functions for the two TSOs to read-in a single monthly file in their specific format.

We're going to use these functions in the loop for all months later on.



Tennet has month and year as variables in the URL whereas TransnetBW only has the month in there.

In [10]:

```
# Download data

for tsoName, tsoConf in conf.iteritems():
    for techName, url in tsoConf.iteritems():

        if tsoName == 'tennet':
            for year in range (2014, 2015): # This means a range f
                path = 'csv/'+tsoName+'/'+techName+'/'+str(year)+'
                if not os.path.exists(path): os.makedirs(path)

                for month in range(1, 13): # This means a range fr
                    full_url = url % {'month': month, 'year': year
                    urllib.urlretrieve(full_url, path+"%d.csv" % m

        if tsoName == 'transnetbw':
            path = 'csv/'+tsoName+'/'+techName+'/'
            if not os.path.exists(path): os.makedirs(path)

            for month in range(1, 25): # This means the last 24 mo
                full_url = url % month
                urllib.urlretrieve(full_url, path+"%d.csv" % month
```

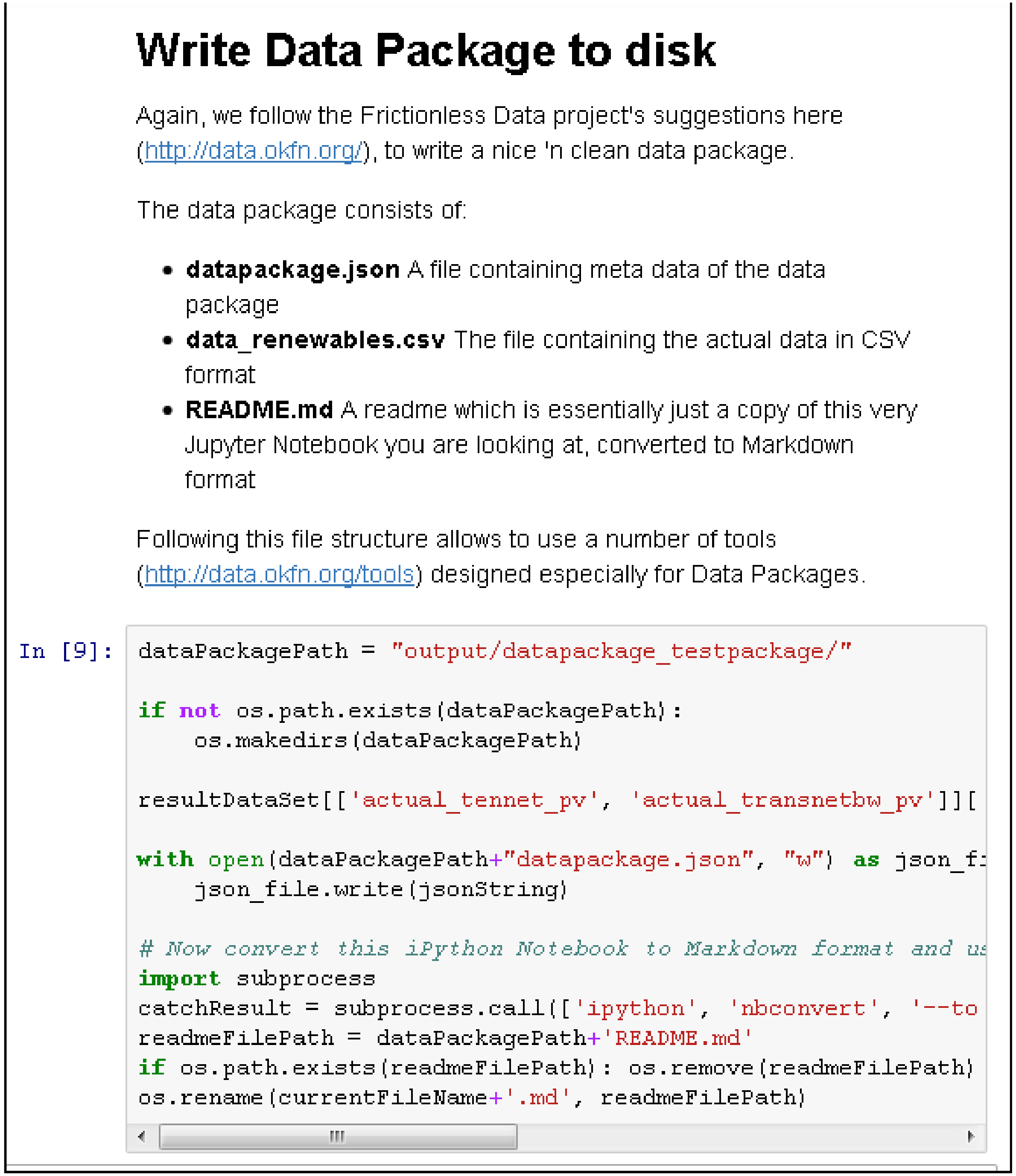
Define individual read functions

The TSOs have different columns and date formats they use in their CSV files. Here we define individual functions for the two TSOs to read-in a single monthly file in their specific format.

We're going to use these functions in the loop for all months later on.

In [11]:

```
# Define specific readData functions for the TSOs

def readData_tennet(filePath, tsoName, techName):
    data = pandas.read_csv(
        filePath,
        sep=";",
        skiprows=4,
        names=['datum','position','forecast '+tsoName+' '+techName
```



Write Data Package to disk

Again, we follow the Frictionless Data project's suggestions here (http://data.okfn.org/), to write a nice 'n clean data package.

The data package consists of:

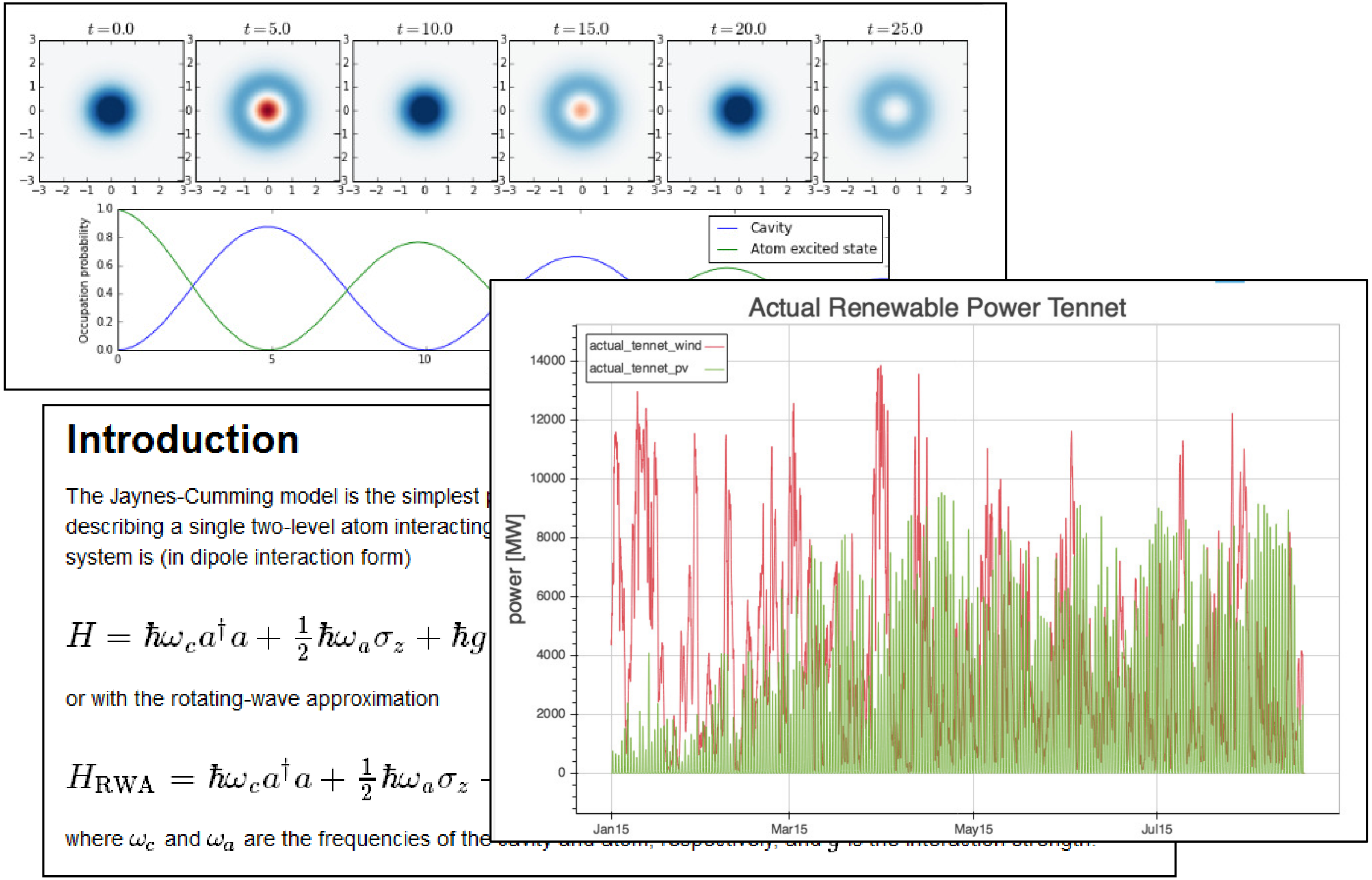
- **datapackage.json** A file containing meta data of the data package
- **data_renewables.csv** The file containing the actual data in CSV format
- **README.md** A readme which is essentially just a copy of this very Jupyter Notebook you are looking at, converted to Markdown format

Following this file structure allows to use a number of tools (http://data.okfn.org/tools) designed especially for Data Packages.

In [9]:

```
dataPackagePath = "output/datapackage_testpackage/"

if not os.path.exists(dataPackagePath):
    os.makedirs(dataPackagePath)

resultDataSet[['actual_tennet_pv', 'actual_transnetbw_pv']][

with open(dataPackagePath+"datapackage.json", "w") as json_f:
    json_file.write(jsonString)

# Now convert this iPython Notebook to Markdown format and us
import subprocess
catchResult = subprocess.call(['ipython', 'nbconvert', '--to
readmeFilePath = dataPackagePath+'README.md'
if os.path.exists(readmeFilePath): os.remove(readmeFilePath)
os.rename(currentFileName+'.md', readmeFilePath)
```

Jupyter Notebooks: Visualization possibilities



Introduction

The Jaynes-Cumming model is the simplest

describing a single two-level atom interacting

system is (in dipole interaction form)

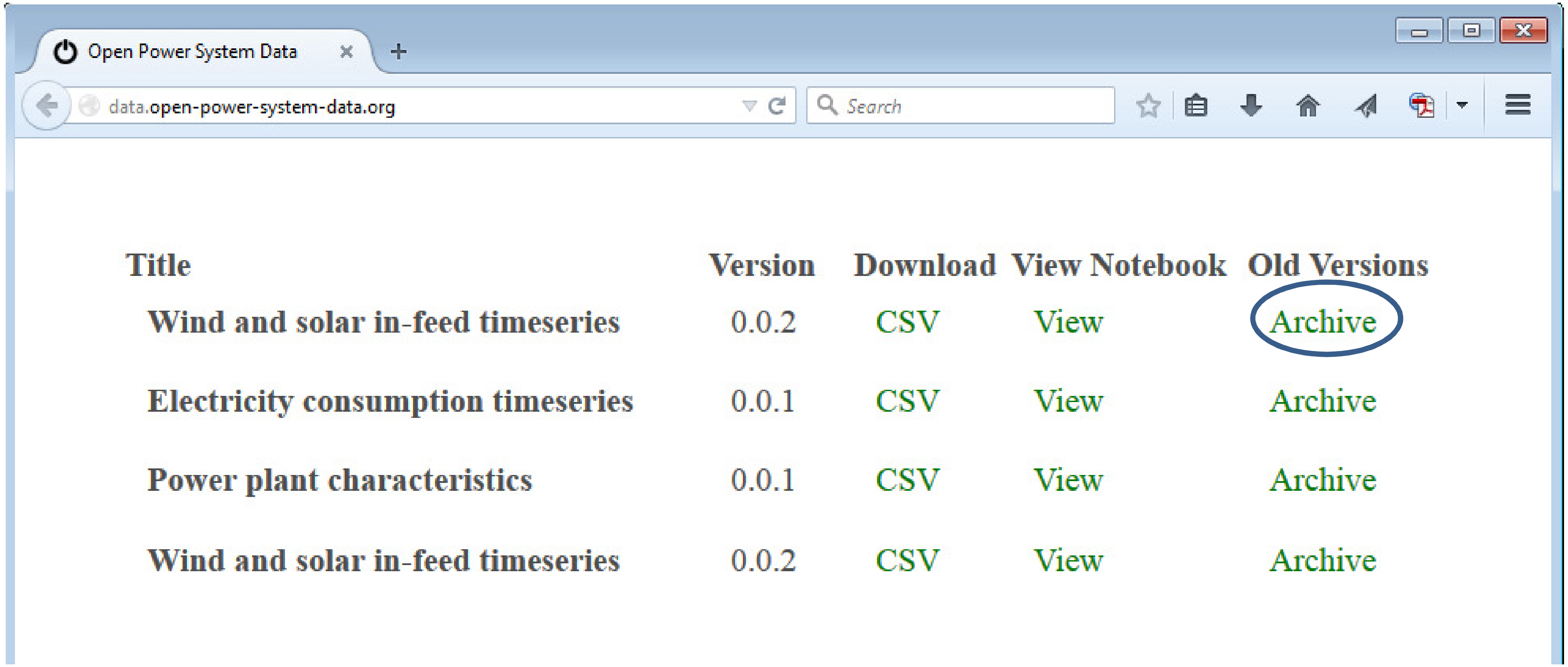
$$H = \hbar\omega_c a^\dagger a + \frac{1}{2}\hbar\omega_a\sigma_z + \hbar g$$

or with the rotating-wave approximation

$$H_{\rm RWA} = \hbar\omega_c a^\dagger a + \frac{1}{2}\hbar\omega_a\sigma_z$$

where ω_c and ω_a are the frequencies of the cavity and atom, respectively, and g is the interaction strength.

OPSD from user‘s perspective



Title	Version	Download	View Notebook	Old Versions
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive
Electricity consumption timeseries	0.0.1	CSV	View	Archive
Power plant characteristics	0.0.1	CSV	View	Archive
Wind and solar in-feed timeseries	0.0.2	CSV	View	Archive

Github

- **Jupyter Notebooks from OPSD are published on GitHub**
  - Version control of the extraction scripts (=Jupyter Notebooks)
  - Enables Open Source development processes (proposed changes to the scripts can be submitted as Github pull requests)
  - Github automatically visualizes Jupyter Notebooks

Summary

- Decentral architecture
- Jupyter Notebooks to enable reproducability of data packages
- CSV files as main data format

3. Legal Issues

Lion Hirth

October / November 2015 | Open Power System Data Webinar

Terms of use today

Can I use energy data from the internet?

What data owners write

- ENTSO-E: “you may download Content, but only for non-commercial, personal use”
- EEX: “exclusive personal, non-commercial use [is] permitted”
- Netztransparenz: “Inhalt und Gestaltung der Internet-Seiten sind urheberrechtlich geschützt.“

The details

Die auf der Internetseite verwendeten Daten in den Formaten .xls und .csv stehen, falls nicht anders gekennzeichnet, unter der Datenlizenz Deutschland – Namensnennung – Version 2.0. […] Die Inhalte des Internetauftritts stehen, soweit nicht anders gekennzeichnet, unter der Creative Commons Namensnennung-Keine Bearbeitung 3.0 Deutschland Lizenz.
– Bundesnetzagentur

[T]his website … as well as databases contained herein are protected by copyright and are owned by EEX AG, except when otherwise stated. Neither this website nor the contents made available therein … may be copied, reprinted, published, transmitted, transferred, disseminated or distributed in any manner without the prior written approval of EEX AG. However, the preparation of a single copy for exclusive personal, non-commercial use by downloading onto an individual personal computer … are expressly permitted.
– EEX

Inhalt und Gestaltung der Internet-Seiten sind urheberrechtlich geschützt. Eine Vervielfältigung der Seiten oder ihrer Inhalte sowie der automatisierte Datendownload bedürfen der vorherigen schriftlichen Zustimmung der 50Hertz Transmission GmbH, soweit die Vervielfältigung nicht ohnehin gesetzlich gestattet ist.
– 50Hertz, Netztransparenz

The ENTSO-E grants you permission to use the Site as follows:
– you may download Content, but only for non-commercial, personal use and provided that you also retain all copyright
– you may not distribute, modify, copy (except as set forth above), transmit, display, reuse, reproduce, publish, license, create derivative works from, transfer, sell or otherwise use Content without ENTSO-E’s written permission
– ENTSO-E

Intellectual property on data

Intellectual property rights

- protect “creations of the mind”
- patents (inventions)
- trademarks (brands)
- copyright (creative works), ...

Legal protection of data

- individual datum (piece of information): not protected
- structured / organized data (database): can be protected
- copyright
- database right: Database Directive (96/9/EG), in Germany *Datenbankherstellerrecht*

Licenses

- a contract between owner and user that allows to use otherwise protected material
- no license = “all rights reserved”
- can be a “standard” license (such as Creative Commons) or a “custom” license
- often mentioned in “terms of use” or “imprint”

Two crucial questions for any user

- Is the database protected, either under copyright or database right?
- If it is protected: Is a license available that allows me to use the database?
- *to us, both answers are quite unclear*

What is “commercial use”?

“No commercial use”

- Many owners of power system data exclude “commercial use”
- It is not always clear what that means

Recent ruling of a German court

- DeutschlandRadio used a photography published under CC-BY-NC
- The website is not paywalled, has no advertisement and no sponsoring
- Court: this use is commercial, only “personal use” allowed
- "Im Zweifelsfall verbleiben die Rechte beim Urheber", heißt es im Urteil.
- → “commercial“ does not mean “behind paywall“ or “operated by for-profit business“

Open Power System Data

Open data

- Goals
  - Get legal clarity on what is allowed → legal advice
  - Allow all types of users to use power system data for all purposes and with least restrictions
- “Open data and content can be freely used, modified, and shared by anyone for any purpose” (Open Definition).
- For the Open Power System Data project, we would like to publish all data under an open license.
- This, of course, requires that the data owners allow us to do so (publish their data under a compatible license).
- Two “families” of well-established open licenses
  - Creative Commons (CC)
  - Open Data Commons (ODC)

Licenses for open data: three flavors

1. Public domain licenses
   - Under these license agreements, all rights are waived.
   - Creative Commons Zero (CC0)
   - Open Data Commons Public Domain Dedication and License (PDDL)

2. Attribution licenses
   - Attribution licenses require to name the author(s) of the database.
   - Creative Commons Attribution (CC-BY) 4.0
   - Open Data Commons Attribution (ODC-BY)

3. Copyleft / share alike licenses
   - Copyleft, or “share alike” licenses require that databases, if amended or modified, are published under the same license as the original database.
   - Creative Commons Attribution Share-Alike (CC-BY-SA) 4.0. (Wikipedia)
   - Open Data Commons Open Database (OpenStreetMap)

More restrictive