Karlsruhe Institute of Technology

FAIR DIGITAL OBJECTS FORUM

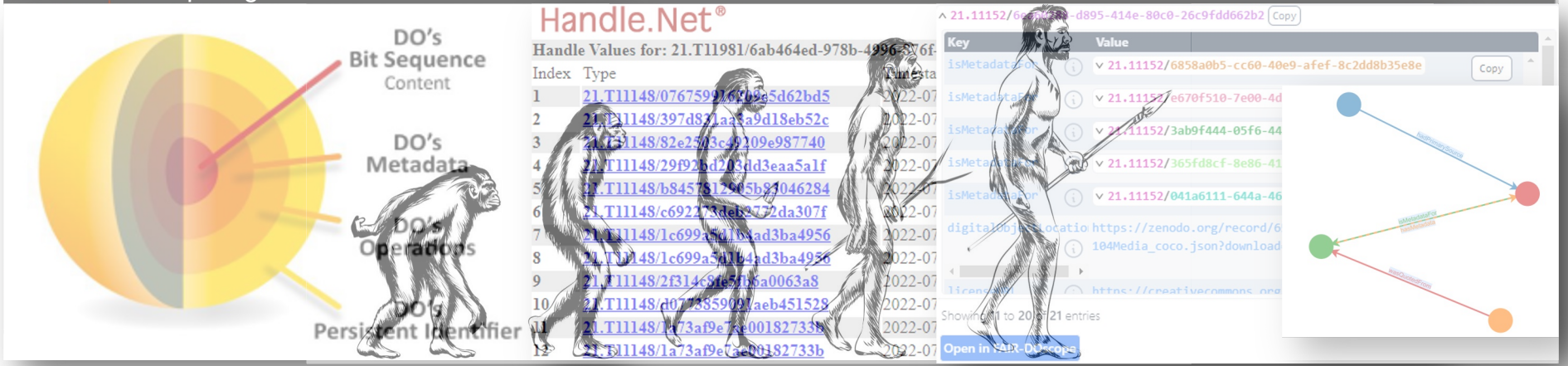
HELMHOLTZ Metadata Collaboration

Bring FAIR DOs into Light

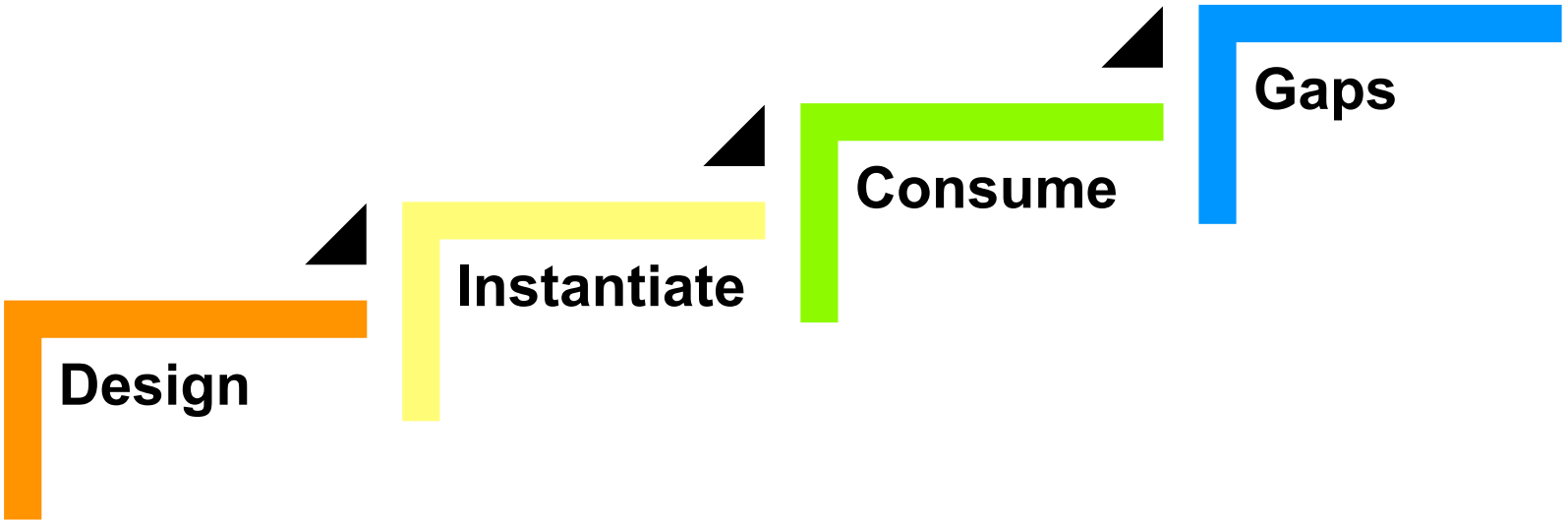
Towards a generic tool set to make FAIR DOs visible and tangible for users

Thomas Jejkal, et al.

Scientific Computing Center



Section Overview



thomas.jejkal@kit.edu

Section Goals

Gaps

Consume

Instantiate

Design

Learn about good and „not so good“ DataTypes and Profiles.

Learn how to use the FAIR DO Cookbook for DataType and Profile design.

Gain awareness of which gaps currently exist and planned activities.

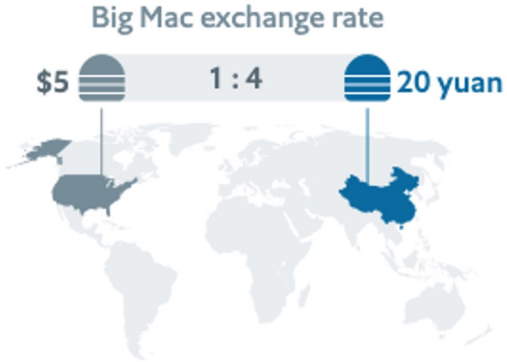
FAIR DO-Design
Example: Big Mac Index

How it works

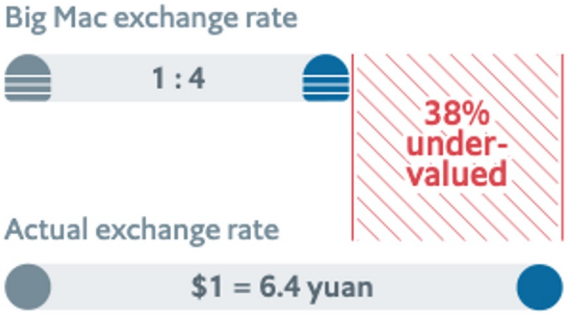
Raw index /GDP-adjusted

Purchasing-power parity implies that exchange rates are determined by the value of goods that currencies can buy

Differences in local prices – in our case, for Big Macs – can suggest what the exchange rate should be



Using burgernomics, we can estimate how much one currency is under- or over-valued relative to another



FAIR DO-Design
Example: Big Mac Index

Karlsruhe Institute of Technology

Data

	date	iso_a3	currency_code	name	local_price	dollar_ex	dollar_price	usd_raw	eur_raw	gbp_raw	jpy_raw	cny_raw
2	2000-04-01	ARG	ARS	Argentina	2.5	1	2.5	-0.00398	0.05007	-0.16722	-0.09864	1.09091
3	2000-04-01	AUS	AUD	Australia	2.59	1.68	1.54166666666667	-0.38579	-0.35246	-0.48645	-0.44416	0.28939
4	2000-04-01	BRA	BRL	Brazil	2.95	1.79	1.64804469273743	-0.34341	-0.30778	-0.45102	-0.40581	0.37836
5	2000-04-01	CAN	CAD	Canada	2.85	1.47	1.93877551020408	-0.22758	-0.18566	-0.35417	-0.30099	0.62152
6	2000-04-01	CHE	CHF	Switzerland	5.9	1.7	3.47058823529412	0.3827	0.45774	0.15609	0.2513	1.90267
7	2000-04-01	CHL	CLP	Chile	1260	514	2.45136186770428	-0.02336	0.02964	-0.18342	-0.11618	1.05023

SCAN ME

https://t1p.de/fdo-data

FAIR DO-Design
Example: Big Mac Index

Karlsruhe Institute of Technology

Metadata

big-mac.csv

variable	definition	source
date	Date of observation	
iso_a3	Three-character [ISO 3166-1 country code][iso 3166-1]	
currency_code	Three-character [ISO 4217 currency code][iso 4217]	
name	Country name	
local_price	Price of a Big Mac in the local currency	McDonalds; *The Economist*
dollar_ex	Local currency units per dollar	*Reuters*
dollar_price	Price of a Big Mac in dollars	
USD_raw	Raw index, relative to the US dollar	
EUR_raw	Raw index, relative to the Euro	
GBP_raw	Raw index, relative to the British pound	
JPY_raw	Raw index, relative to the Japanese yen	
CNY_raw	Raw index, relative to the Chinese yuan	

ollar_price	usd_raw	eur_raw	gbp_raw	jpy_raw	cny_raw
5	-0.00398	0.05007	-0.16722	-0.09864	1.09091
54166666666667	-0.38579	-0.35246	-0.48645	-0.44416	0.28939
54804469273743	-0.34341	-0.30778	-0.45102	-0.40581	0.37836
93877551020408	-0.22758	-0.18566	-0.35417	-0.30099	0.62152
47058823529412	0.3827	0.45774	0.15609	0.2513	1.90267
45136186770428	-0.02336	0.02964	-0.18342	-0.11618	1.05023

SCAN ME

https://t1p.de/fdo-data

FAIR DO-Design
Solution 1 - The easy one

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
MetadataLocation	https://github.com/rfordata[...]

FAIR DO-Design
Solution 1 - The easy one

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
MetadataLocation	https://github.com/rfordata[...]

- Easily applicable
- Potential increase of findability
- Generic profile and data types

- Very limited overall machine-actionability
- Hard to reuse due to lack of machine-readable metadata
- Human evaluation required

FAIR DO-Design
Solution 2 - Frictionless Data-based

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	FrictionlessDataPackage
DigitalObjectLocation	https://repo/fric_data_pkg.zip

Data v1 Package

https://specs.frictionlessdata.io/

FAIR DO-Design
Solution 2 - Frictionless Data-based

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	FrictionlessDataPackage
DigitalObjectLocation	https://repo/fric_data_pkg.zip

Data v1 Package

https://specs.frictionlessdata.io/

- Easily applicable
- Potential increase of findability and reuse
- Generic profile and data types
- Machine-readable data and metadata

- Republishing required
- Limited machine-actionability on FAIR DO-level
- Full download required
- Format knowledge required

FAIR DO-Design
Solution 3 - FAIR DO times two

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
MetadataLocation	Metadata-FAIR-DO

FAIR DO-Design
Solution 3 - FAIR DO times two

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
MetadataLocation	Metadata-FAIR-DO

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVTableDefinition
DigitalObjectLocation	http://repo/CSVTableDefinition.json

FAIR DO-Design
Solution 3 - FAIR DO times two

PID →

Key	Value
Profile	SimpleProfile
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
MetadataLocation	Metadata-FAIR-DO

- Enrichment of existing (legacy) data
- Potential increase of findability and reuse
- Generic profile and data types
- Machine-actionable via linked metadata

- Republishing of metadata required
- Requires metadata download and format knowledge
- Increased complexity

FAIR DO-Design
Solution 4 - All-in-one

PID →

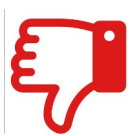
Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

FAIR DO-Design
Solution 4 - All-in-one

PID →

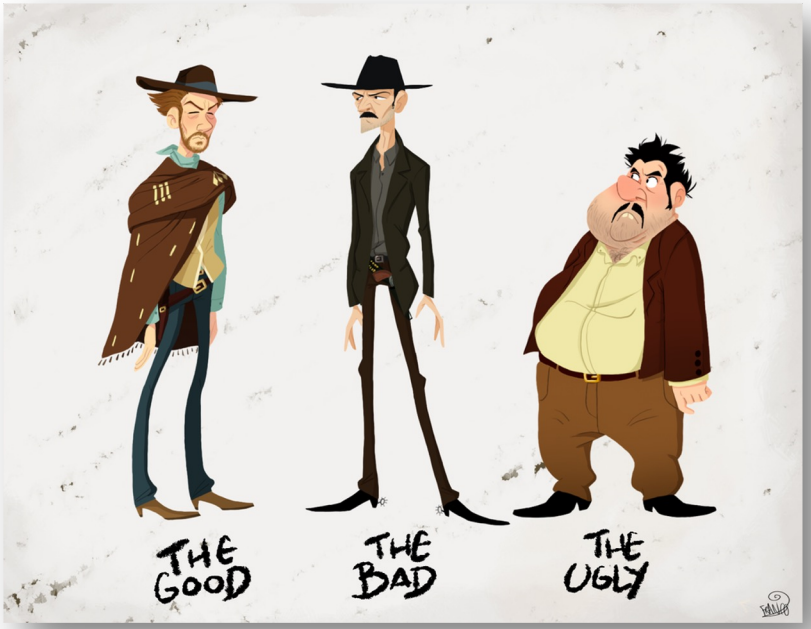
Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

- Easily applicable
- Potential increase of findability and reuse
- Specific but reusable profile and data types
- Machine-actionability on PID-level



- Huge entry barrier as profile and complex data type are required
- Unfolds full potential only if broadly applied

FAIR DO-Design
- The Good, the Bad, and the Ugly -



Mentimeter

Enter the code to join

It's on the screen in front of you

1442 2670

Join

https://www.menti.com/al9tppotna6g

FAIR DO Cookbook

Features

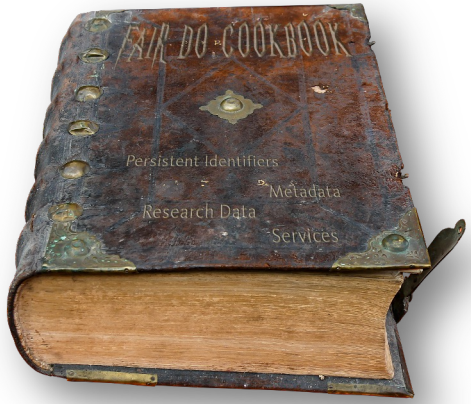
- Collection of recipes on common FAIR DO tasks
- Focus on concrete FAIR DO implementation
- Shows good practices and alternate paths

Target Audience

- Developers
- FAIR DO creators and designers

Status

- Released



SCAN ME

https://t1p.de/fdo-cb-about

The FAIR DO Cookbook and DataTypes

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

SCAN ME

https://t1p.de/fdo-cb-dt

The FAIR DO Cookbook and DataTypes

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

Ingredients

- Data Type Registry (DTR), e.g. http://dtr-test.pidconsortium.eu/
- Registered user account at Data Type Registry
- Web Browser

The FAIR DO Cookbook and DataTypes

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

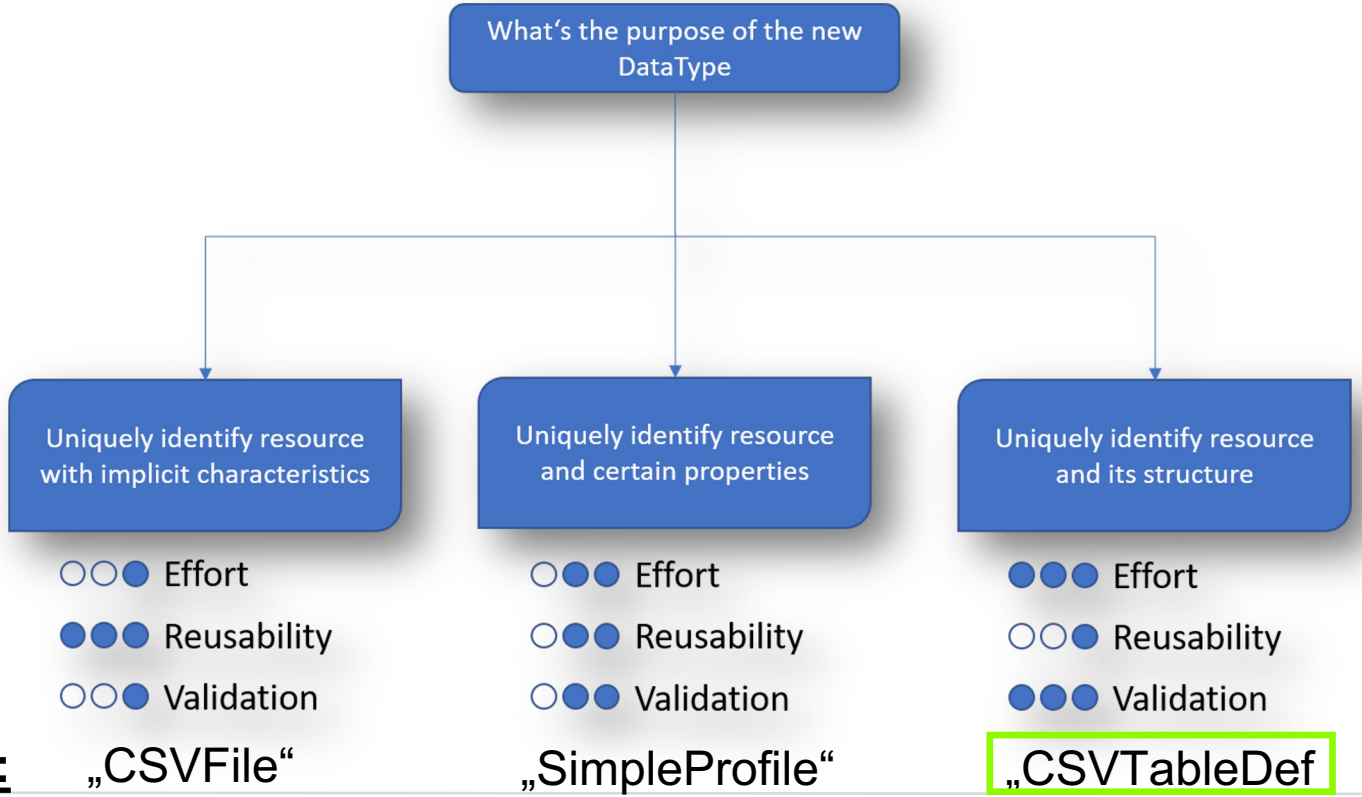
Ingredients

- Data Type Registry (DTR), e.g. http://dtr-test.pidconsortium.eu/
- Registered user account at Data Type Registry
- Web Browser

Work Steps

1. Search data type registry for suitable data type
2. Describe/model envisioned data type (offline)
   - 2.1. Decide on Data Type's level of detail
   - 2.2. DTR Inquiry
3. Register data type(s) in data type registry
   - 3.1. Register new Child-Data Type(s)
   - 3.2. Register extended Child-Data Type(s)
   - 3.3. Register main Data Type

Decide on a DataType‘s Level of Detail



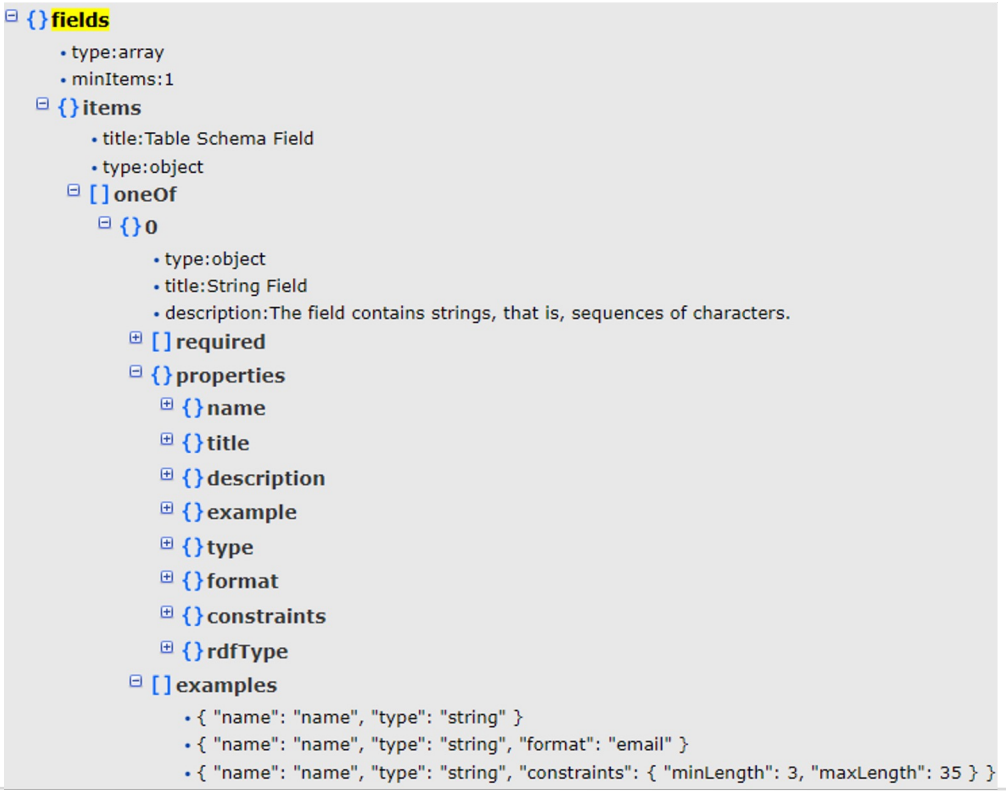
Examples:

DTR Inquiry

Karlsruhe Institute of Technology

CSV Dialect Descriptor

CSV Field Description



DTR Inquiry

CSV Dialect Descriptor

Property	DTR Inquiry Result (Reuse, Extend, Create)
csvddfVersion	
delimiter	
doubleQuote	
lineTerminator	
nullSequence	
quoteChar	
escapeChar	
skipInitialSpace	
header	
commentChar	
caseSensitiveHeader	

Register Data Types

Property	DTR Inquiry Result (Reuse, Extend, Create)
csvddfVersion	Reuse version-number (21.T11148/ac9849005793b63ac780)
delimiter	Create
doubleQuote	Create
lineTerminator	Create
nullSequence	Create
quoteChar	Create
escapeChar	Create
skipInitialSpace	Create
header	Create
commentChar	Create
caseSensitiveHeader	Create

- Create DataTypes (Extend/Create)
- Work from bottom to top
- Document DataType PIDs in table

Property	DTR Inquiry Result (Reuse, Extend, Create)
csvddfVersion	Reuse version-number (21.T11148/ac9849005793b63ac780)
delimiter	Create (21.T11148/f1627ce8538232475078)
doubleQuote	Create (21.T11148/f1627ce8538232475078)
lineTerminator	Create (21.T11148/ab56fee8538232475078)
nullSequence	Create (21.T11148/f1622135467223456249)
quoteChar	Create (21.T11148/231c28758724856f703e)
escapeChar	Create (21.T11148/8608328253c27f5e1477)
skipInitialSpace	Create (21.T11148/734172ef52685270c883)
header	Create (21.T11148/8c722276581f43875e03)
commentChar	Create (21.T11148/18f2786724c55330e827)
caseSensitiveHeader	Create (21.T11148/87253e3041f578c72862)

- Create top-level DataType

The FAIR DO Cookbook and Profiles

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

SCAN ME

https://t1p.de/fdo-cb-kip

The FAIR DO Cookbook and Profiles

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

Ingredients

- Data Type Registry (DTR), e.g. http://dtr-test.pidconsortium.eu/
- Registered user account at Data Type Registry
- Web Browser

The FAIR DO Cookbook and Profiles

PID →

Key	Value
Profile	Profile4MachineActionableCSV
DigitalObjectType	CSVFile
DigitalObjectLocation	https://github.com/rfordata[...]
CSVTableDef	JSON-Object

Ingredients

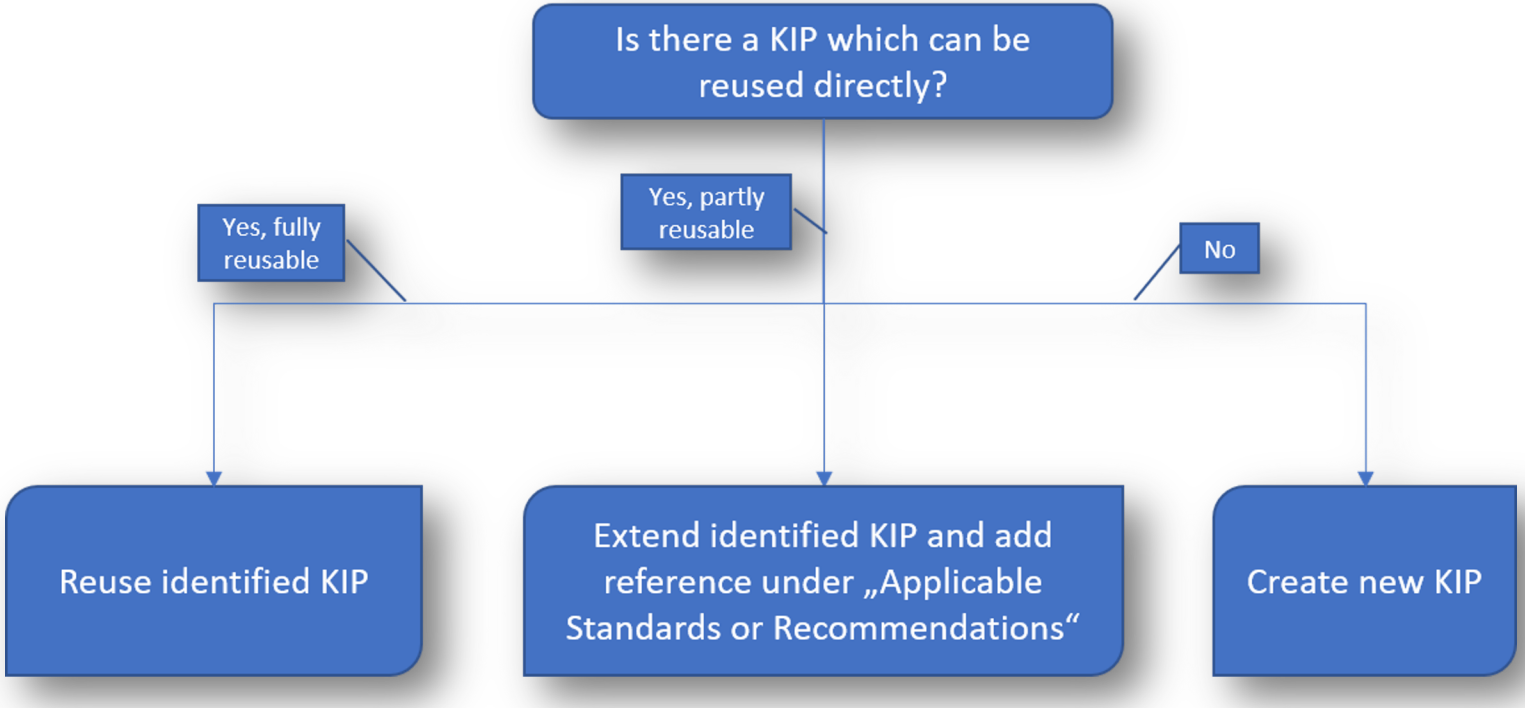
- Data Type Registry (DTR), e.g. http://dtr-test.pidconsortium.eu/
- Registered user account at Data Type Registry
- Web Browser

Work Steps

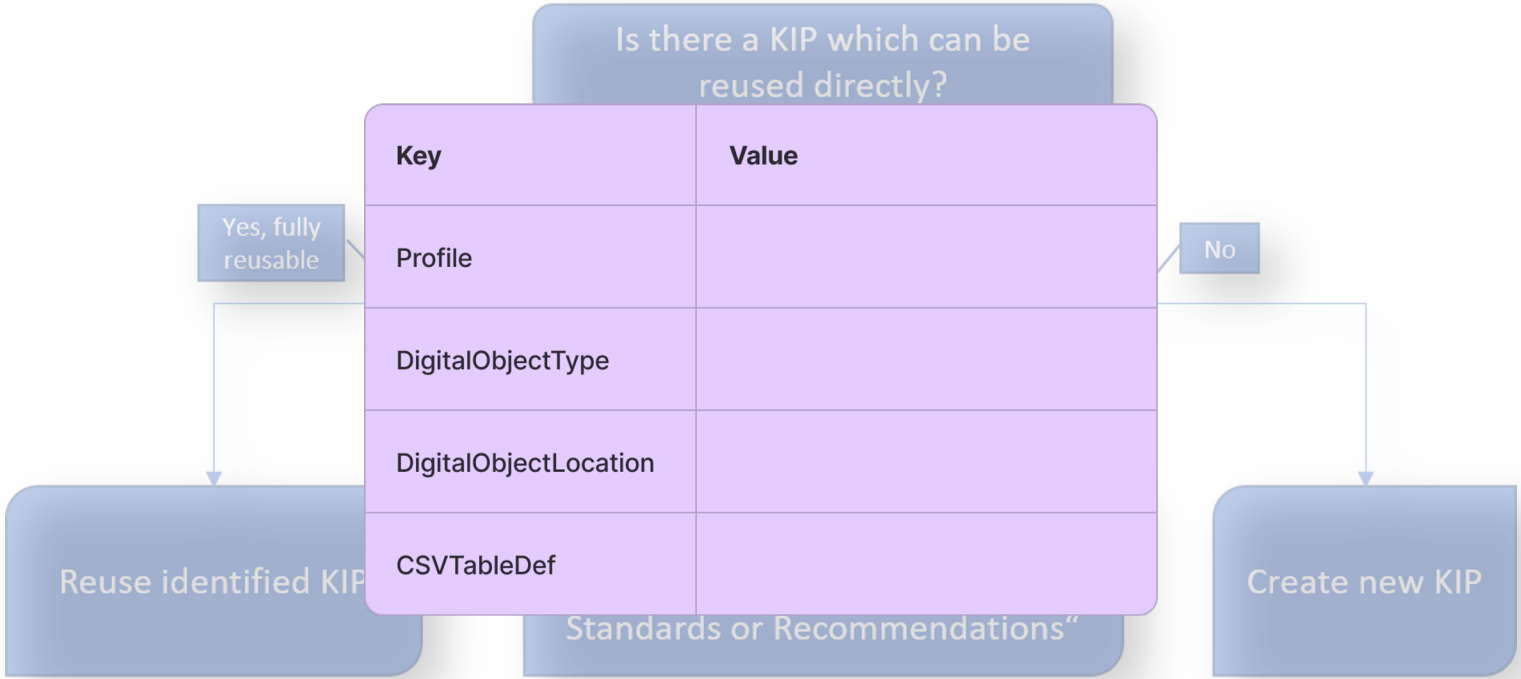
1. Search data type registry for suitable KIP
2. Describe/model envisioned KIP
    2.1. Call to mind basic KIP principles
    2.2. Decide on KIP contents
3. Register new KIP in Data Type Registry

KIP = Kernel Information Profile

Decide on a Profile



Decide on a Profile



Key	Value
Profile	
DigitalObjectType	
DigitalObjectLocation	
CSVTableDef	

Register Profile

1. Identify and create all missing DataTypes, just like before.
2. Copy existing/create new profile.
3. In case of copy, refer to original profile.
4. Name it.
5. Add all attributes and their type information (must be unique!).
6. Save it and become a gardener.

Section Summary

Learn about good and „not so good“ DataTypes and Profiles.

- *There is no clear „good“, „bad“, or „ugly“.*
- *Effort strongly depends on what you want to achieve.*
- **Will improve with a growing number of DataTypes and Profiles.**

Learn how to use the FAIR DO Cookbook for DataType and Profile design.

- *Comprehensive collection of good practices with clear focus on single implementation can lower entry barriers.*
- **More of such guidelines required?**

Gain awareness of which gaps currently exist and planned activities.

- *It can be a long way from the idea to a new FAIR DO.*
- *DataType Registry requires a major overhaul.*
- **FAIRCORE4EOSC and alternative implementation at KIT.**

Section Goals

Design

Instantiate

Consume

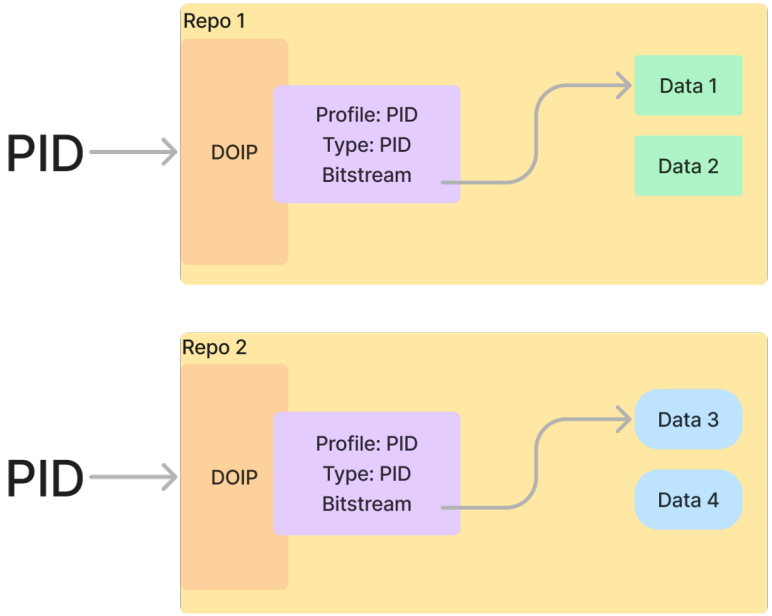
Gaps

Understand different approaches of instantiating FAIR DOs.

Learn how to use available tools for FAIR DO creation.

Get informed about planned activities.

Two Sides of a Similar Medal

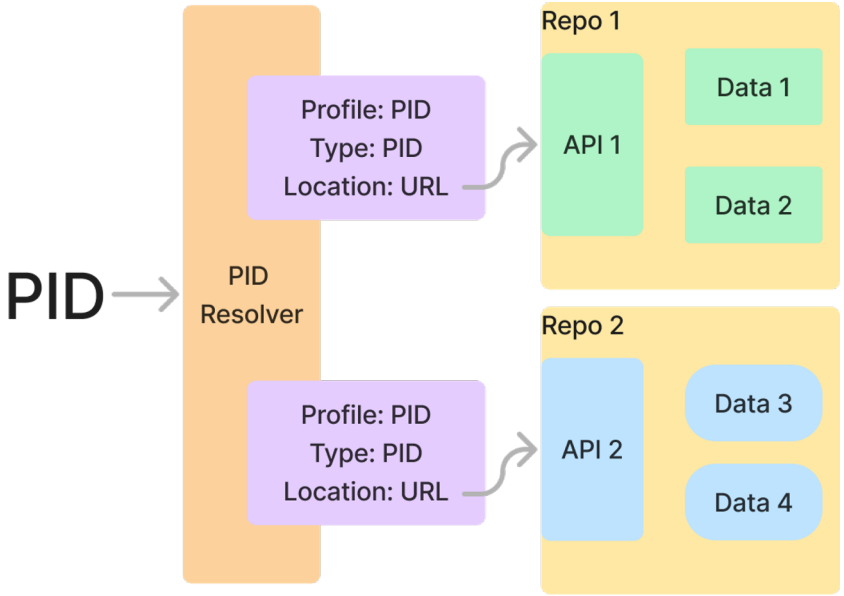


DOIP = Digital Object Interface Protocol

+ Native FAIR DOs with authorative contents
+ Supports vision of one-for-all interface
+ Built-in support for extended operations

- Requires infrastructure providers to implement DOIP or operate proxy
- Pure DOIP is TCP/IP-based, i.e., very low-level
- Seems to be poorly adopted

Two Sides of a Similar Medal

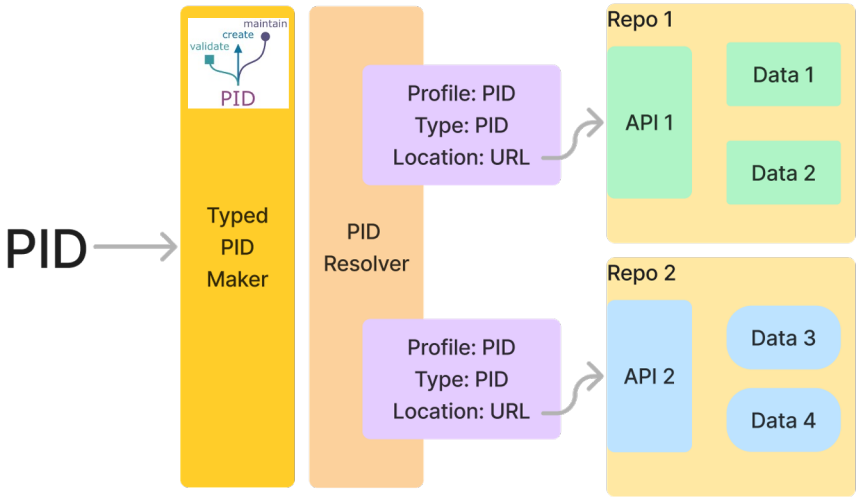


+ Non-invasive application of FAIR DOs on top of existing infrastructures
+ Accessible via PID resolver
+ May accumulate metadata from different sources

~ Location preferably (open) Web-resolvable, HTTP-based API endpoint

- PID record non-authorative source, regular validation required
- No safety net while creating FAIR DOs
- Plain HTTP calls required

The Typed PID Maker



+ Supports use of different PID resolvers
+ Includes validation of PID records
+ Supports caching of DataTypes and Profiles
+ Allows PID customization
+ Allows fine-grained authorization
+ Supports indexing and search

- Another wrapper
- Standalone service that requires to be operated
- Still, plain HTTP calls required

The Typed PID Maker – Create a FAIR DO

```
curl --location --request POST 'http://typed-pid-maker-instance/api/v1/pit/pid/' \
--header 'Content-Type: application/json' \
--data-raw '
```

record	
key	**value**
21.T11148/076759916209e5d62bd5	21.T11148/b9b76f887845e32d29f7
21.T11148/1c699a5d1b4ad3ba4956	21.T11148/1a1e620666cb1713acde
21.T11148/b8457812905b83046284	https://repo/api/v1/identifier
21.T11148/aafd5fb4c7222e2d950a	2024-01-01T12:12:12Z
21.T11148/2f314c8fe5fb6a0063a8	https://spdx.org/licenses/CC-BY-4.0.html
21.T11148/82e2503c49209e987740	{"md5sum":"449da831ecee4824fea8c48b02148c3d"}
pid	(:tba)_1709894161887

```
'
```

Webpage

https://t1p.de/tpidm-web

Source Code

https://t1p.de/tpidm-source

Further Endpoints:

```
GET     http://typed-pid-maker-instance/api/v1/pit/pid/{pid}
PUT     http://typed-pid-maker-instance/api/v1/pit/pid/{pid}
GET     http://typed-pid-maker-instance/api/v1/pit/known-pid
POST    http://typed-pid-maker-instance/api/v1/search
```

FAIR DO Builder

Features

- Easy-to-use Web UI
- Form-based FAIR DO creation including basic validation
- Visual linking of FAIR DOs

Target Audience

- Developers
- FAIR DO creators and designers

Status

- Early access (single profile)

SCAN ME

https://t1p.de/fdo-builder

FAIR DO Builder

Features
- Easy-to
- Form-
  basic
- Visual

Target A
- Devel
- FAIR

Status
- Early access (single profile)

Future Plans

- Integrate multi-Profile support
- Integrate dynamic DataType support
- Improve user experience, e.g., detailed error messages
- Support bulk-creation of FAIR DOs

du/fdo-builder-ui.html

iDORIS

Integrated Data Type and Operations Registry with Inheritance System

Features

- Registry for DataTypes, Profiles, and Operations
- Inheritence support for DataTypes and Profiles
- Support for DataType-Operation matching

Target Audience

- Developers
- Infrastructure providers

Status

- Conceptual phase (tdb. Q4 2024)

Section Summary

Understand different approaches of instantiating FAIR DOs.

- *Learned about FAIR DOs via DOIP and PID resolver.*
- *Typed PID Maker for additional functionality.*
- *(Many) other approaches exists, harmonization and mapping required.*
- **For production use automation/integration required.**

Learn how to use available tools for FAIR DO creation.

- *Low-level APIs available for automation/integration, in-depth knowledge required.*
- *Introduced FAIR DO Builder as easy-to-use UI for humans.*
- **Further tooling and flexibility required.**

Get informed about planned activities.

- *Improvements and extensions for existing tooling.*
- *iDORIS to improve DataType/Profile creation and adding Operation support.*
- **Ideas and contributions are always welcome!**

Section Goals

Learn how to possibly consume FAIR DOs depending on the target audience.

Future plans and collection of ideas.

FAIR DO Search

Features

- Easy-to-use Web UI
- Runs on top of Typed PID Maker
- Enhanced search via Elasticsearch
- Facet support for FAIR DO filtering

Target Audience

- Scientific users

Status

- Early access (single profile, static facets)

SCAN ME

https://t1p.de/fdo-search

Features
- Easy-to
- Runs
- Enhan
- Facet

Target A
- Scien

Status
- Early

/elastic-search-fdo.html

Future Plans

- Customization of facets and result view
- Integration of multi-Profile support
- Allow better reusability

FAIR-DOscope

Features

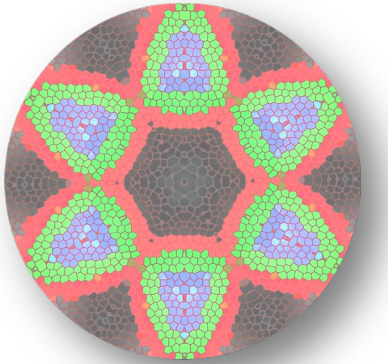
- Visualization of and navigation via PID records
- DataType-driven rendering of key/values
- Visualization of FAIR DO relationships

Target Audience

- Developers
- FAIR DO creators and designers

Status

- Released (v1.1.0)



SCAN ME

https://t1p.de/fdoscope

Future Plans

- Dynamic resolution of DataTypes/Profiles
- Improved graph visualization
- Improved integration with other tools

PID-Component

Features

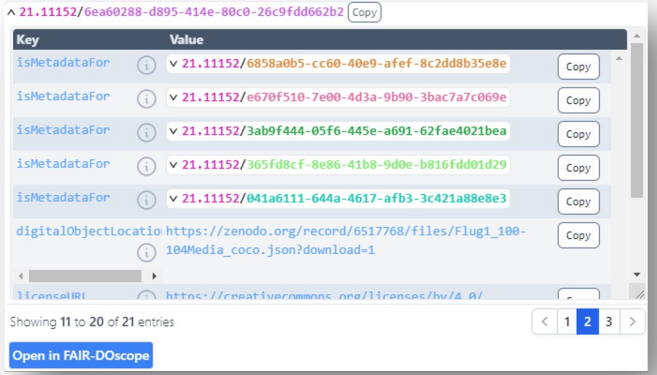
- Web component to render FAIR DOs
- Seamless integration into any Web page
- Extensible rendering, e.g. ORCiDs

Target Audience

- Platform developers (integration)
- Scientific users

Status

- Released (v0.0.11)



SCAN ME

https://t1p.de/pid-com

PID-Component

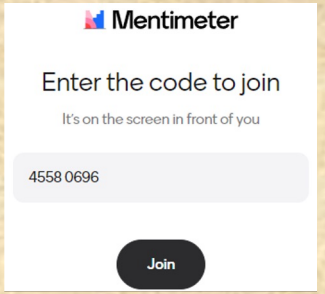
Features
- Web co
- Seaml
- Exten

Target A
- Platfo
- Scient

Status
- Releas

Future Plans

- Additional renderers (ROR, IGSN, etc.)
- Any idea?



https://www.menti.com/al9573c43ce3

Section Summary

Learn how to possibly consume FAIR DOs depending on the target audience.

- *FAIR DO Search allows search in huge FAIR DO collections.*
- *FAIR-DOscope enables fast inspection of created FAIR DOs.*
- *PID-Component for seemless integration into existing Web UIs.*
- **Move towards reusability and integrability.**

Future plans and collection of ideas.

- *Focus mainly on flexibility and customization.*
- **Ideas and contributions are always welcome!**

Section Goals

Karlsruhe Institute of Technology

Identification of other missing tools/components.

Clarification of open questions.

Summary and closing of session.

Gaps

Consume

Instantiate

Design

thomas.jejkal@kit.edu

Ideas for Missing Tools/Components

Karlsruhe Institute of Technology

Design
- DataType Designer UI

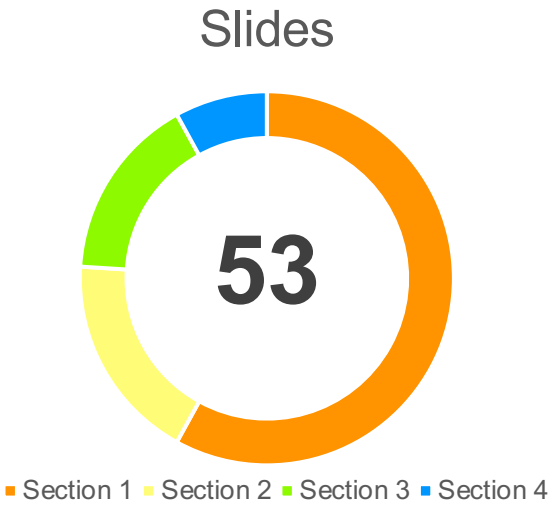
Instantiate
- Metadata2Record Mapper

Consume
- FAIR DO Finder

Others
- Maintenance tools

Open Questions

Summary



Got to know the 🍔-index.

Got 4 ideas on how to represent CSV data as FAIR DO.

Completed 2 Mentimeter.

Learned about 7 tools.

Gave 90 minutes of grateful attention.

Acknowledgements

- The Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the National Research Data Infrastructure – NFDI 38/1 – project number 460247524
- The Joint Laboratory Model and Data driven Materials Characterization (JL MDMC), a cross-centre platform of the Helmholtz Association
- NFFA-Europe-Pilot (EU H2020 – n. 101007417)
- The research program “Engineering Digital Futures” of the Helmholtz Association of German Research Centers
- The Helmholtz Metadata Collaboration Platform

Image References

Slide 1, FAIR DO, Schultes, E., & Wittenburg, P. (2019). FAIR Principles and Digital Objects: Accelerating Convergence on a Data Infrastructure., 10.1007/978-3-030-23584-0_1
Slide 1, Evolution, https://detektor.fm/wp-content/uploads/2022/02/k3starshutterstockevolution.jpg
Slide 4, How it works, https://github.com/rfordatascience/tidytuesday/raw/master/data/2020/2020-12-22/pic2.png
Slide 16, Good, Bad, Ugly, https://i.pinimg.com/originals/38/75/71/3875711647b09bfb8d6e9b2da1c7838f.png
Slide 29, Gardener, https://i.redd.it/wra9pru80t171.jpg
Slide 36, Bob Builder, https://static.wikia.nocookie.net/btb/images/c/cf/CGI_BOB.png/revision/latest/scale-to-width/360?cb=20221001033806
Slide 38, Construction, https://media.istockphoto.com/id/185244309/de/foto/arbeiten-in-bearbeitung.jpg?s=612x612&w=0&k=20&c=eaP8_lYyCDOAuXn8niT-Dp3arHnl-kgZPGBt8wXQnGQ=
Slide 41, Search, https://bibliotecnica.upc.edu/sites/default/files/actualitat/11986_sgb_googledata.png
Slide 50, Questions, https://images.squarespace-cdn.com/content/v1/5ea237e587e03021f9ef8cc2/1591321482139-5N4EC0SIATZQYDTX8I3G/Question-mark.jpg
Slide 51, BigMac, https://www.shutterstock.com/image-vector/big-mac-cheese-on-light-600nw-1787891633.jpg