

# Introduction to Smart Data Models

## I4trust training

Alberto Abella

Data Modelling expert - FIWARE Foundation

15-12-22



[i4Trust Website](#)

[i4Trust Community](#)



# Contents and goals

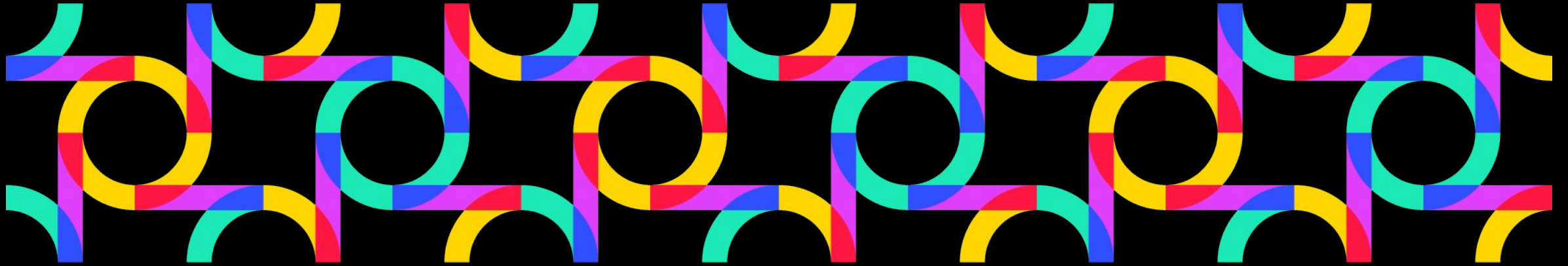
- About:
  - During this part of the session you will get introduced to the Smart Data Models Program
- This session will:
  - Explain basics about the Smart Data Models program and how it is governed
  - Introduce you to data models for specific verticals that are already available
  - Explain how you may contribute extensions to existing data models under Smart Data Models
  - Explain how you may contribute new data models under Smart Data Models
- Goals:
  - After this session you will be able to implement a service using a Smart Data Model
  - You will be able to explain how to become an active contributor to the Smart Data Model program

# Contents and goals

- Question:
  - What industrial sector are you interested in so we can visit this while the online demonstration (use the chat)
  - This presentation is available at <https://bit.ly/i4trust3> (Recommended to get the links)
  - Write it down in this link [http://bit.ly/github\\_users](http://bit.ly/github_users)

# Agenda

- What is the Smart Data Models Program.
  - Structure and governance
  - Current Status
- Becoming a contributor of Smart Data Models Program
- Practical exercises
  - Services for creating a data model
  - Creating an official data model



## Smart Data Models: Introduction and Governance

# Introduction and governance

## Steering Board :

- Four members (December-2022)



- IUDX: Indian Urban Data Exchange
  - Public entity supporting data interchange for Smart cities in India



- TMForum:
  - Worldwide Telecommunications association



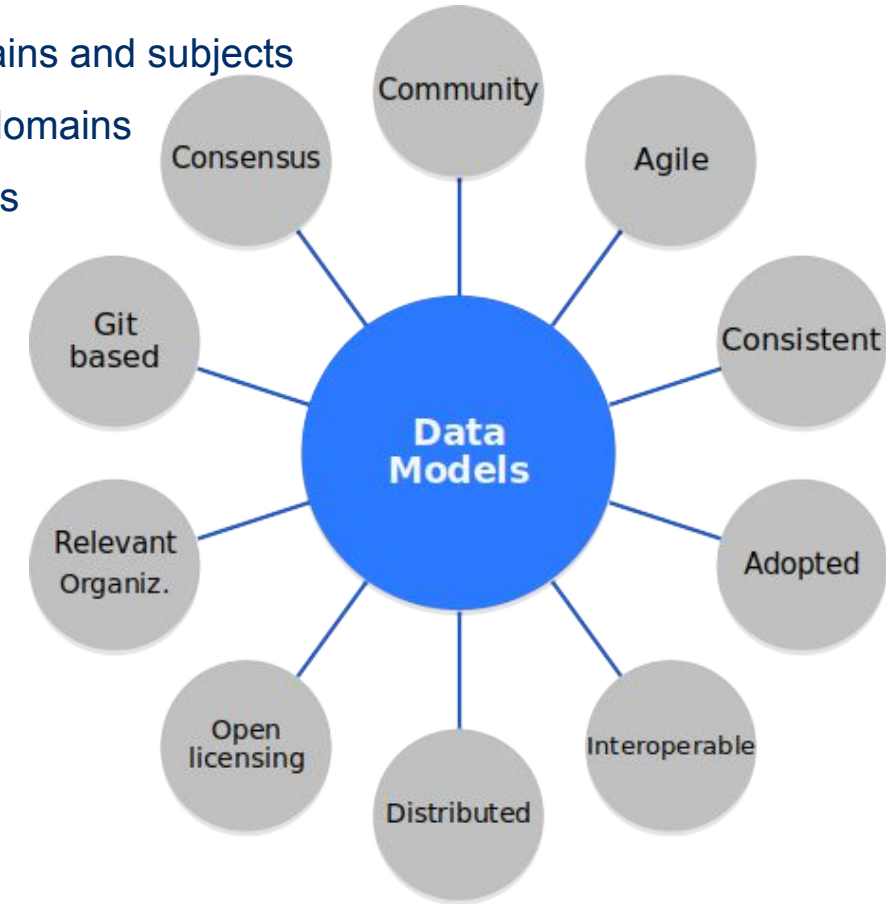
- FIWARE Foundation
  - Curator of the open source FIWARE platform and its ecosystem

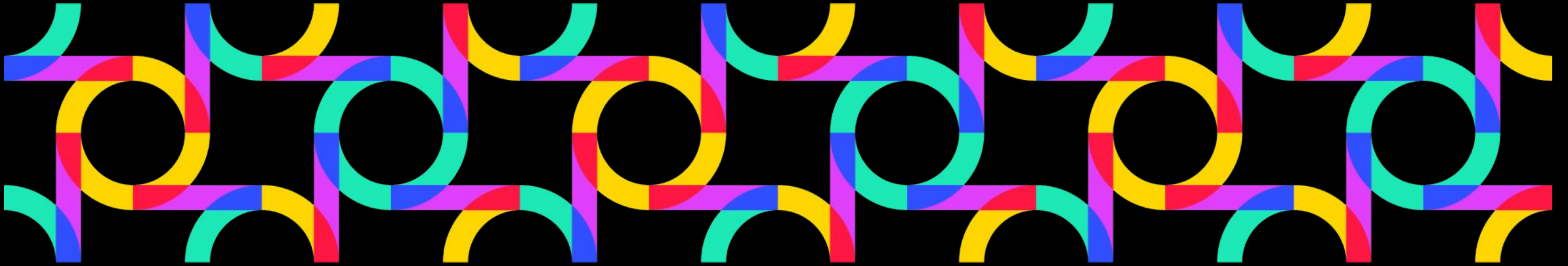


- Open and Agile Smart Cities
  - Smart cities association for the cooperation and mutual learning

# Introduction and governance

- A **community** site with **detailed data models** available for open use for multiple sectors
- Together with other **relevant organizations** in the curation of the different domains and subjects
- Providing **coherence and consistency** between data models across different domains
- To create a method for **AGILE standardization** and evolution these data models
- To provide extended usefulness to I4Trust / FIWARE platform users in terms of:
  - Extended **interoperability**
  - Reduced time dedicated to data model coding
  - Accumulated experience tested in real case scenarios
  - Mapped to be integrated with other platforms
- Using **open licensing** to allow extensive use and adoption
- Allowing customization and evolution of data models
- Based on **real** case scenarios
- Based on git platform and github as development frontend
- Connecting with linked data
- Based on **widely adopted** standards including ontologies and international schemas.





## Standardization in digital markets



# Standardization in a digital market

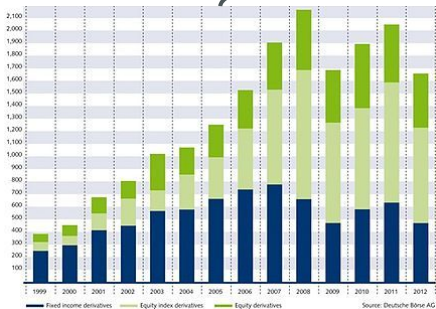
8 years

years

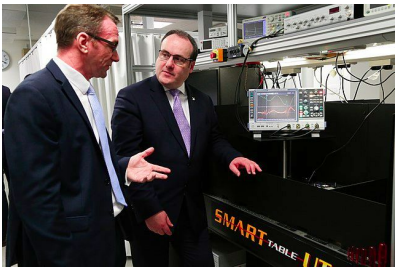
Standardization  
bodies

years

end



Emerging markets



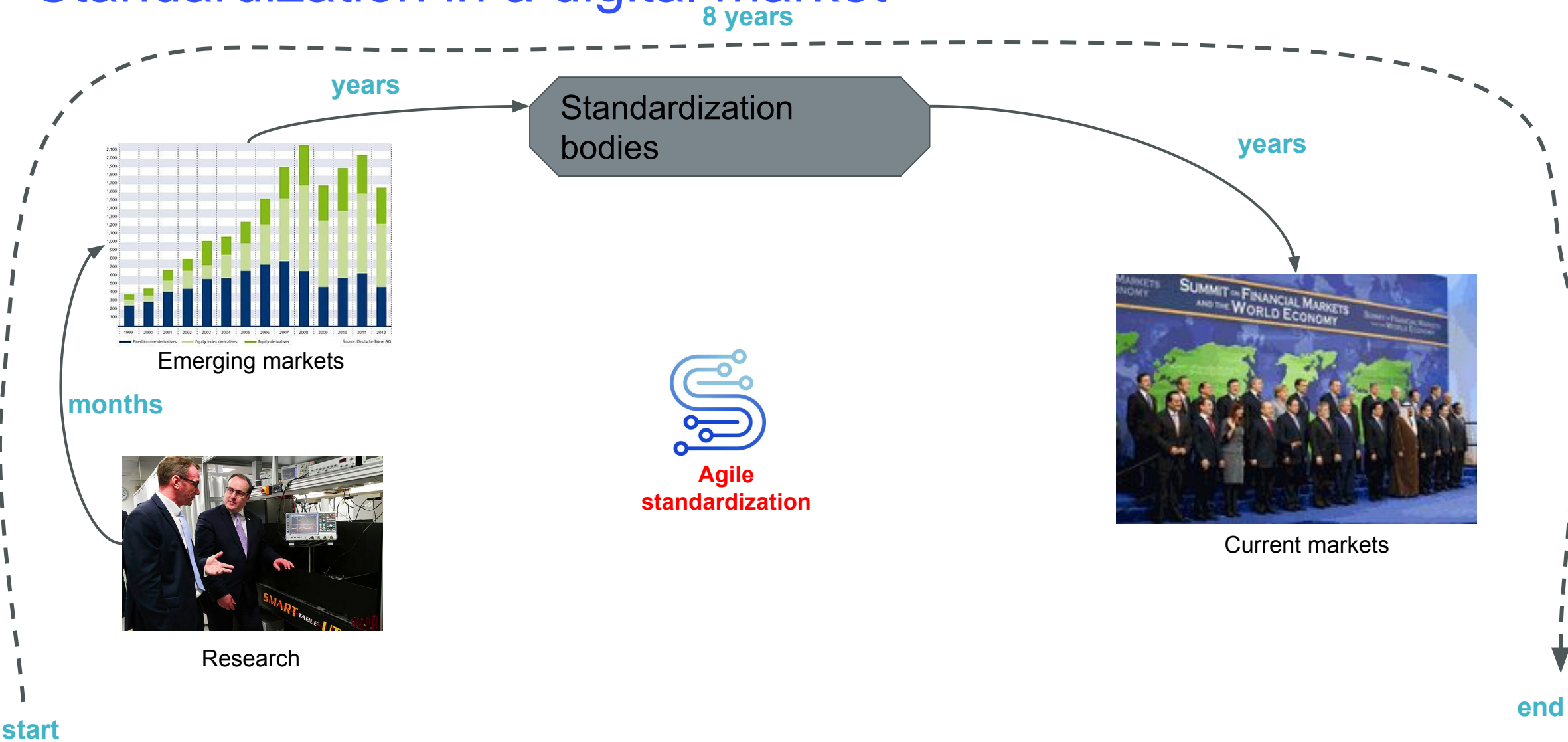
Research



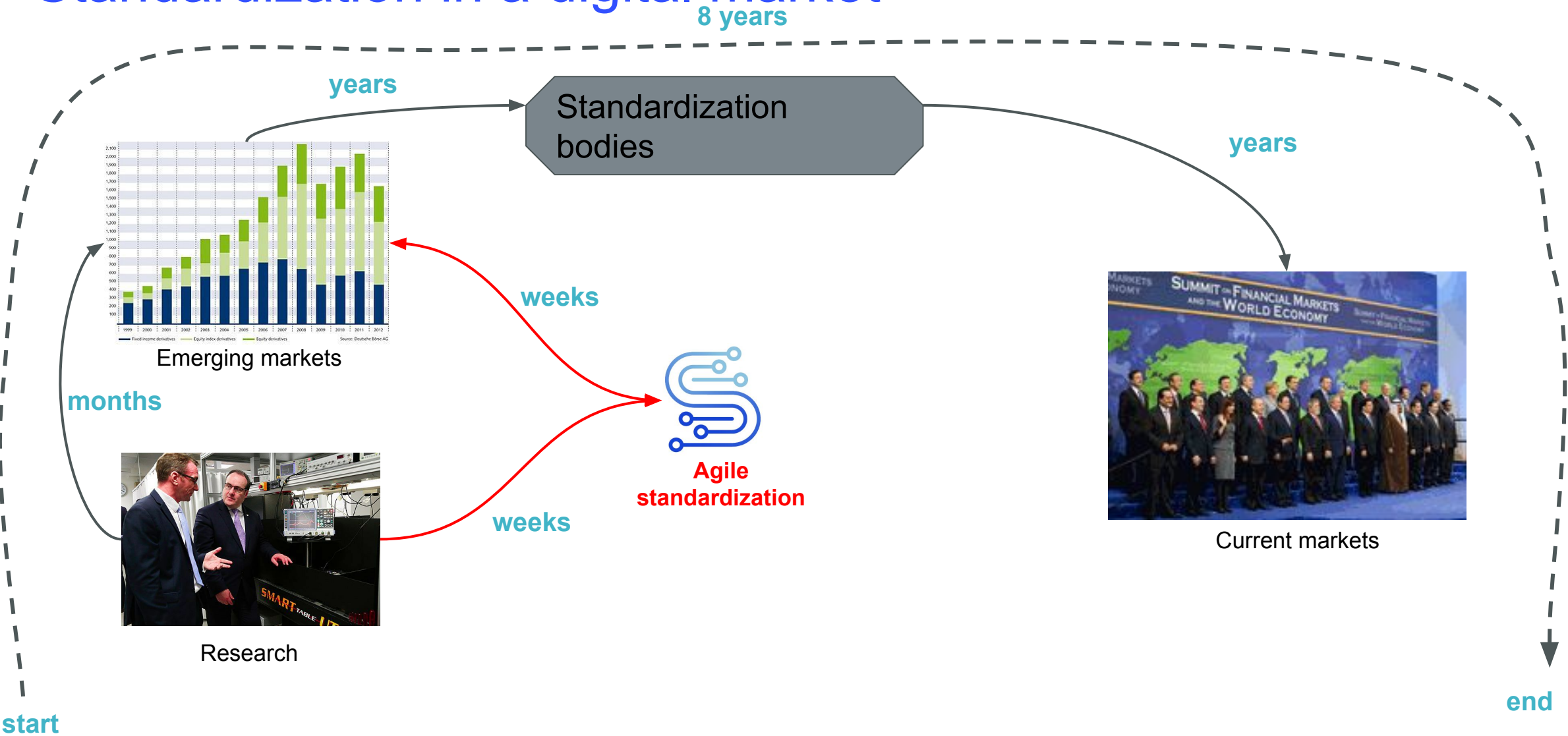
Current markets

start

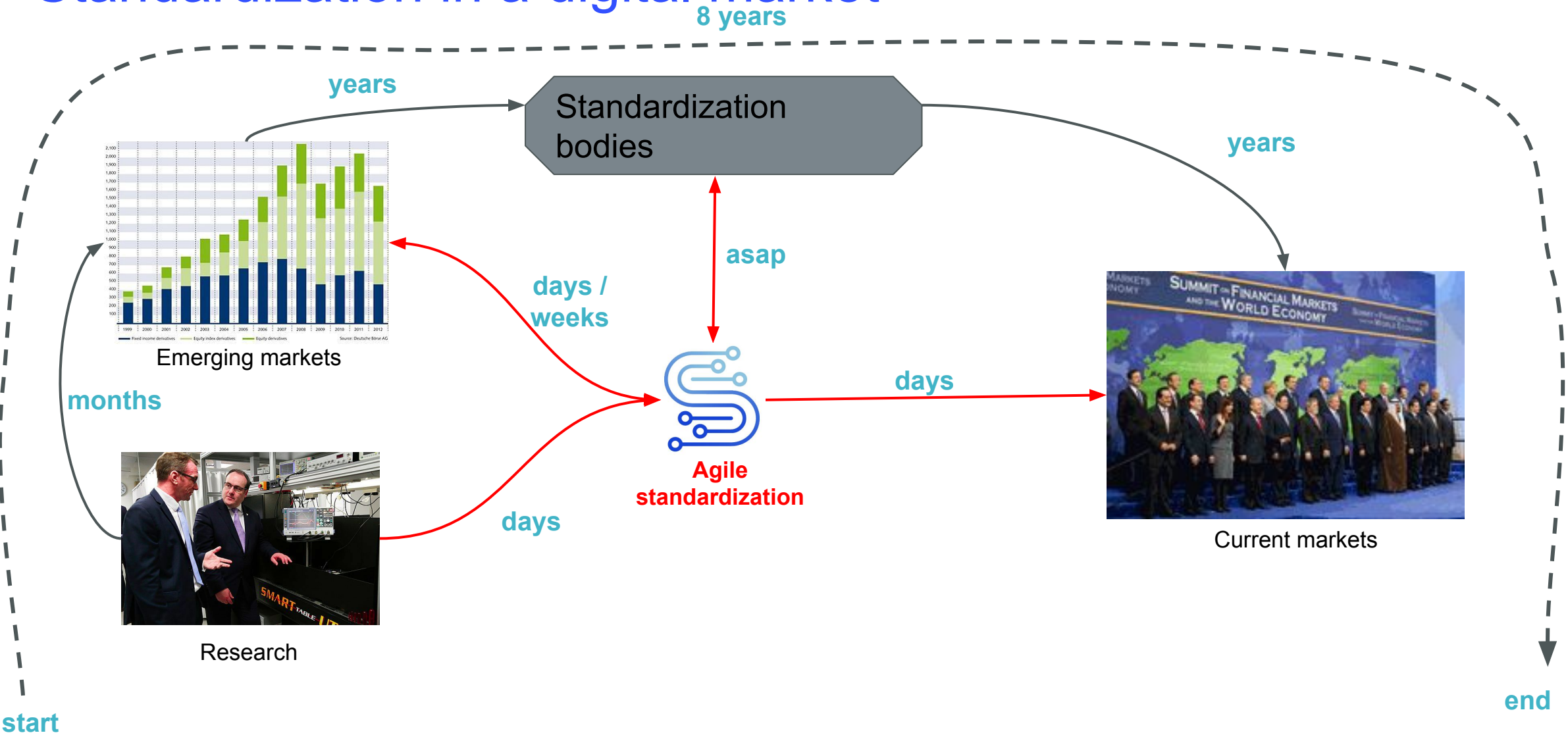
# Standardization in a digital market

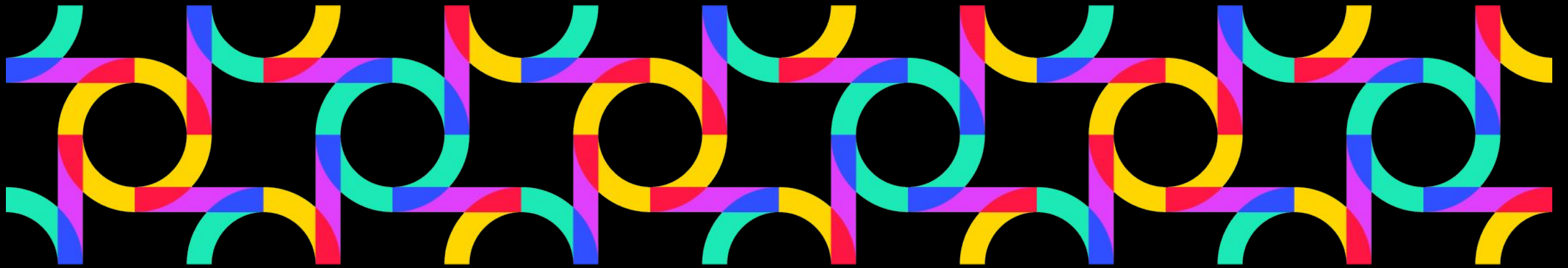


# Standardization in a digital market



# Standardization in a digital market





## Smart Data Models: Introduction and Governance



# Introduction and governance

## Differential advantages

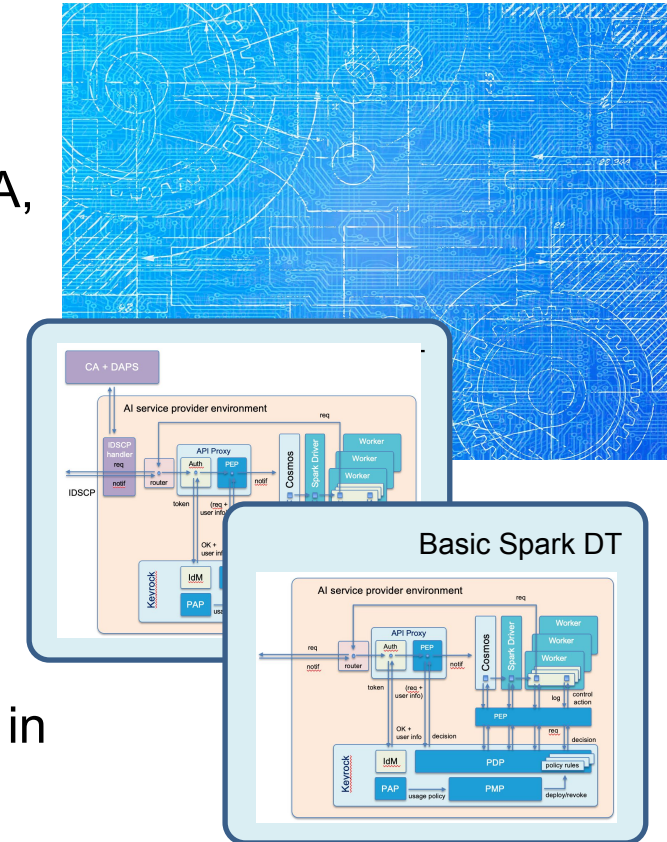
1. **Agile standardization.** Standardization time takes days vs months/years
2. **Easy contribution.** One single file source of truth.
3. **Based on actual experience.** All data models are based on real case scenarios.
4. **Multi Language (7).** Automation translation of specs.
5. **1 change for all domains.** Changing a data model impact on all the domains related to it.



By Pieter Bruegel the Elder - bAGKOdJfvfAhYQ at Google Cultural Institute, zoom level maximum, Public Domain, <https://commons.wikimedia.org/w/index.php?curid=22178101>

# Why you should use Smart Data Models

- Don't reinvent the wheel. Take advantage of the experience of other users
- Lots of attributes (>19.000 already available). Don't create definitions yourself unless you really want to do it.
- Specification automatic generated in 7 languages (EN, FR, GE, SP, IT, JA, CH). No work for the creators
- Examples of the data models available
- Following open and adopted standards
- Everything is open-licensed (Free reuse, free modification and free sharing). Customize to your needs.
- It does not impose restriction on using additional attributes
- It does not impose restrictions on how many attributes has to implement in your use case
- It is required to describe the classes that you are going to create and eventually contribute to SDM



# Mapped standards

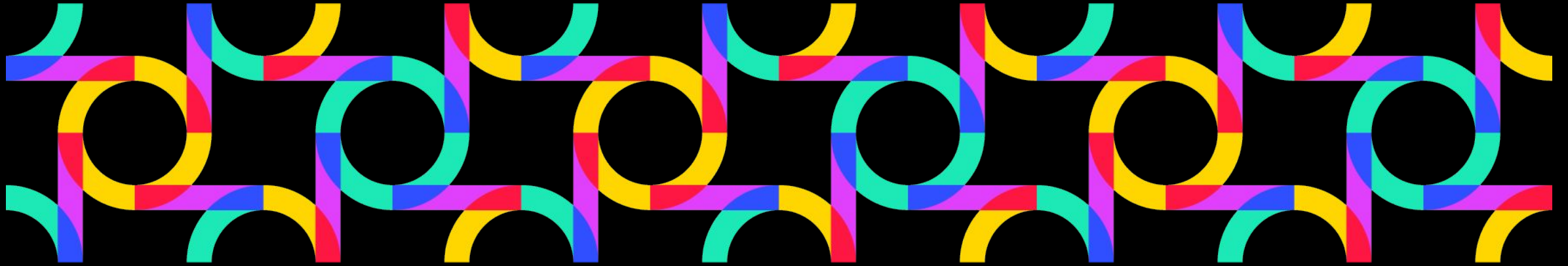
- **Smart data models** does not create data models but adopt other proved at market. Some of them come from open and adopted standards
- [CPSV-AP](#): Mapping of the core vocabulary of public services 2.2.1
- [DCAT-AP](#): Definition of open data datasets. Mapping DCAT-AP 2.1.0
- [STAT-DCAT-AP](#): Definition of open data datasets. Mapping STAT-DCAT-AP 1.0.1 (in progress)
- [Urban mobility](#): Maps GTFS for Urban Mobility
- [GBFS](#): Mobility for bicycles and scooters GBFS
- [EnergyCIM](#): Mapping Common Information Model (CIM) specified by the IEC61970
- [EPANET](#): Mapping the interaction with the open source tool EPANET for water distribution
- [Frictionlessdata](#): Mapping frictionless data standard for interaction with open source tool OpenSDG for sustainable development goals
- [Issuetracking](#): Mapping elements of open311 standard
- [OPCUA](#): Mapping the standard OPCUA (in progress)
- [OCF](#): Mapping standardization by Open Connectivity Foundation
- [OSLO](#): Mapping OSLO ontology for mobility
- [HL7](#): Mapping version 4.3 of HL7 standard for smart health (just started)



# Mapping standards and ontologies

- **Conditions:**
  - a. **Original Standard/Ontology has to be open:**
    - i. Free use
    - ii. Free modification
    - iii. Free sharing of the modifications
    - iv. Only requesting credits to the authors
- Standard/ontology has to be adopted by the market. So capable to provide examples or organizations using it and examples of payloads
- Standard/ontology has to be documented and their data types defined\*

\*Sometimes SDM complete the standard for the undefined sections



Smart Data Models: Structure and contents. Verticals

# Smart Data Models: Structure and contents. Verticals

## ■ What is a Smart Data Model:

- It is the definition of a set of attributes, their definitions and their data types that can be combined into a single entity.
- It is the combination of 3 elements
  - A **technical description** of the data types and relationships of the attributes of an element that has to be modeled
  - A **documented specification** of these attributes aligned with the technical description
  - Some **examples** of the use of the data model
- Based on real case scenario (not theory or academic research)
- With an open license allowing its use, share and modification

## ■ What is NOT a Smart Data Model

- It is not an ontology describing the elements of an area of knowledge
- A new standard
- An academic exercise

# Smart Data Models: Structure and contents. Verticals

## DATA-MODELS

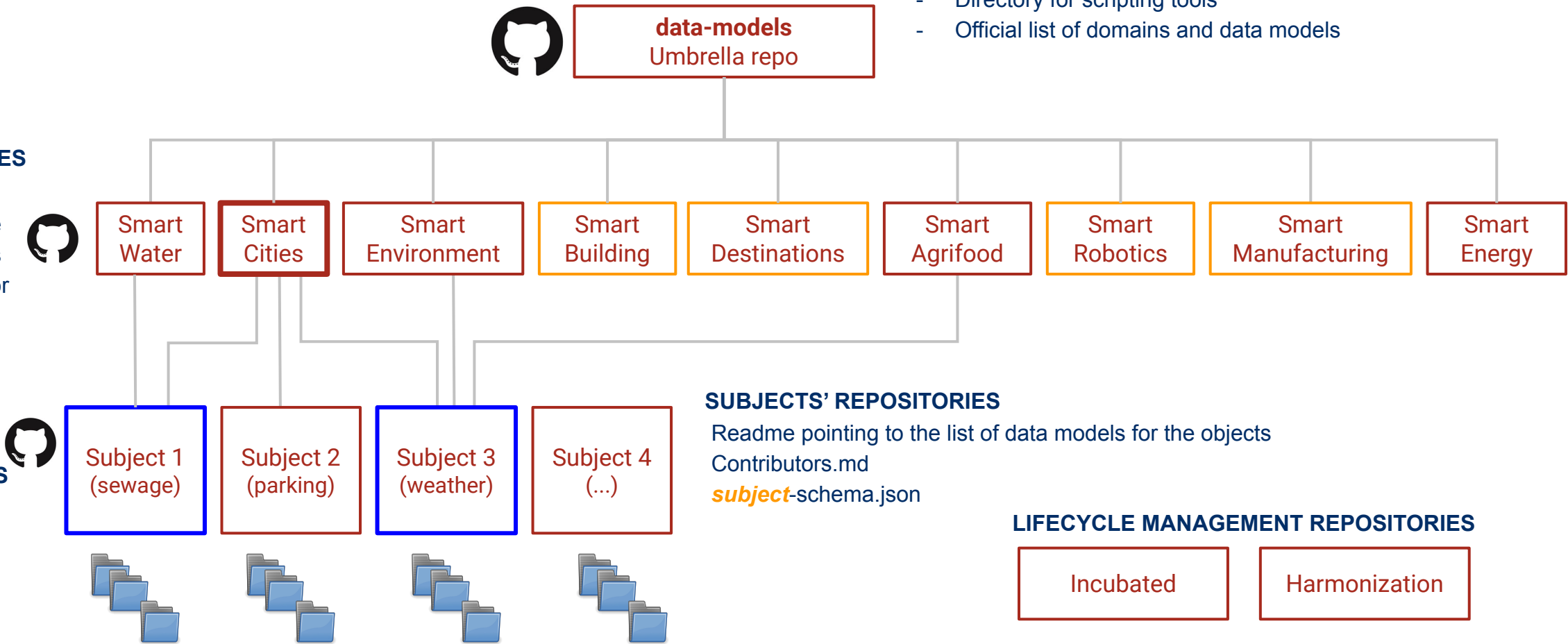
- Guides for coding new data models
- Template for new data models and examples
- Directory for scripting tools
- Official list of domains and data models

## DOMAINS REPOSITORIES

Readme pointing to the list of subjects  
General info or shared resources

## DATA MODELS

README.md  
/doc/spec.md  
/examples  
schema.json  
Adopters



# Smart Data Models: Structure and contents. Verticals

## GITHUB

<http://github.com/smart-data-models>

The screenshot shows the GitHub repository for Smart Data Models. The repository is public and has 75 repositories, 3 teams, and 10 people. It is pinned to the user's profile. The repository description states: "A program led by FIWARE, IUDX, TM Forum, OASC and others to support the adoption of common compatible data models in smart cities." The repository is categorized under "Smart Cities" and "Smart Energy". It includes data models for Buildings, Parking, Urban Mobility & etc. (SmartCities), Energy (SmartEnergy), and Water Management (SmartWater). It also includes a joint collaboration program to support the adoption of a reference architecture and compatible common data models underpinning a digital market of interoperable and replicable smart solutions (data-models). The repository is categorized under "Python" and "Incubated".

- Oriented to **developers**
- All resources available
- Contribution by PR
- Issues on data models
- Demo visit

## SITE (WordPress)

<http://smartdatamodels.org>

The screenshot shows the header of the Smart Data Models website. It features the Smart Data Models logo, a navigation menu with links to Tools, Support, Documentation, About, Search, Community, and New data model request, and a banner for a global program led by FIWARE, IUDX, and TM Forum.

### Domains

Smart Cities	Smart Agrifood	Smart Water	Smart Energy
Smart Environment	Smart Robotics	Smart Sensing	Cross sector
Smart Aeronautics	Smart Destination	Smart Health	Smart Manufacturing
Lifecycle repositories	Incubated	Harmonization	

- Oriented to **end users and contributors**
- News on updates (subscription)
- Check attributes
- Services for contributors
- Check descriptions
- Demo visit



## Smart Data Models: Current Status

# Current status (domains, subjects & data models)

1	Smart Energy	424
2	Smart Sensoring*	138
3	Smart Cities	92
4	Cross Sector	79
5	Smart Water	36
6	Smart Environment	30

7	Smart Agrifood	28
8	Smart Aeronautics	13
9	Smart Robotics	12
10	Smart Destination	11
11	Smart Health	6
12	Smart Manufacturing	3
13	Smart Logistics	1

Updated 1-12-22

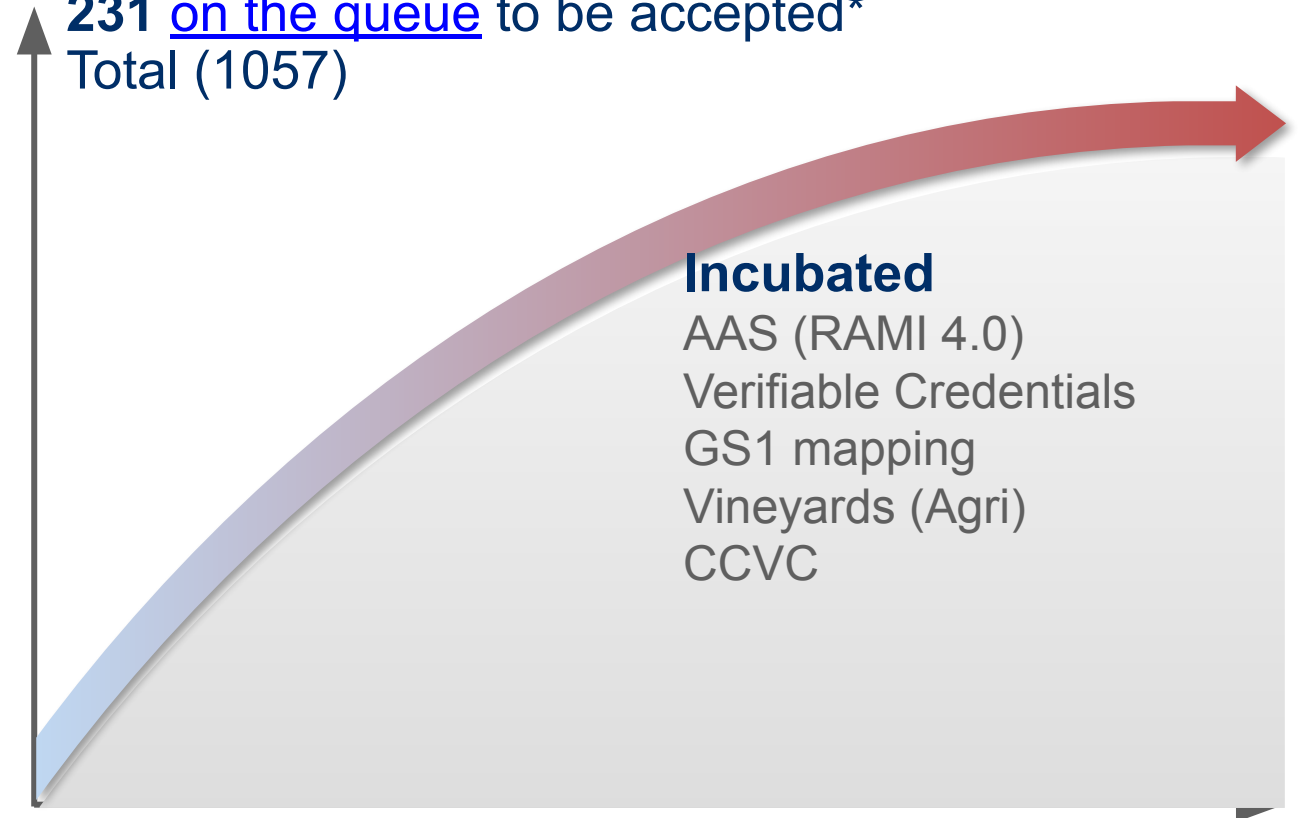
\* Many sensors are specific from other domains but not counted there  
Smart cities 27, Health 19, Environment 12, Energy 5, Water 4, Agrifood 1, Robotics 1

# Current status: New subjects new data models

13 domains (Groups of subjects)  
62 subjects (Groups of data models)

1	SDMX
2	HL7
3	Consumption
4	MarineTransport
5	OSLO
6	Organization

826 Official Smart Data Models  
231 on the queue to be accepted\*  
Total (1057)



\* Not all of them will become official

Updated 1-12-22



# Contributors and dissemination



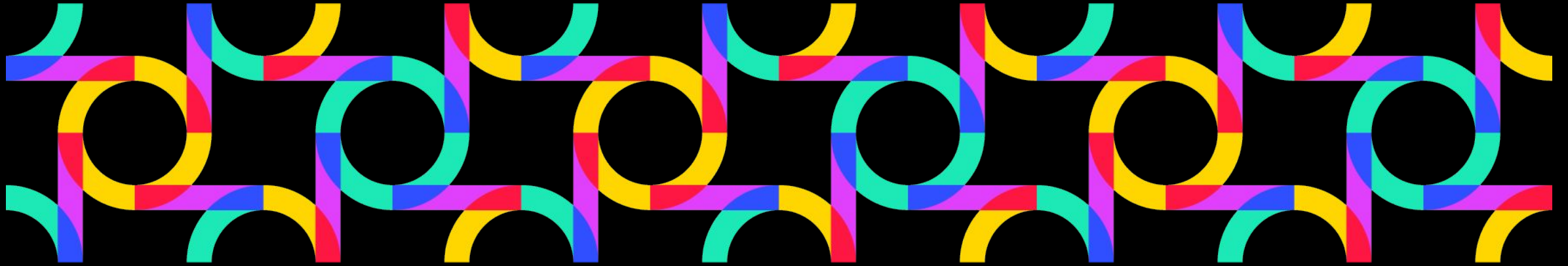
- 131 active contributors
- 247 contribution in data models
- 23 services to contributors in data models



- Contributors belong to 86 different organizations
- Terms available for search 19.165
- Documented adopters 153



- Every term in data models has an associated page  
<https://smartdatamodels.org/Subject/term>
- Google finds 2610 pages in smartdatamodels.org



## Smart Data Models: Conceptual elements

# Conceptual elements 1 / 4

## 1.-Schema

- Format: json schema
- Include descriptions
- validates examples in key values format
- File name *schema.json* at the root of the data model folder
- Single-source-of-truth for the attributes' definitions
- Never stored in a context broker
- Defines attributes' names and their data types
- Manual contribution

## 2.-Examples

- Format: json, jsonld, csv, etc
- Always id and type
- File names *exampleXXXXX.NNN* at *examples* directory in data model folder
- Origin: from actual use cases or derived from them
- Always in actual systems
  - keyvalues only export format
  - normalized usual format
- From actual use case

# Conceptual elements 2 / 4

## 3.-Specification

- Format: markdown
- 7 languages
- File name *specXX.md* at the *docs* data model folder
- Generated automatically. Never edited
- Never contributed
- Explanatory of the data model

## 4.-Subject

- Format: repository in github
- Contains several folders for data models
- Root of the repository
- includes file CONTRIBUTORS (list of people)
- README: List of data models included
- Created by the organization
- Belongs to one or several domains

# Conceptual elements 3 / 4

## 5.-Context

- Format: jsonld
- Long URI for all subject's attributes
- Filename context.jsonld at subject root
- Generated automatically. Never edited
- Never contributed
- For linked data solutions

## 6.-Adopters

- Format: yaml
- List of entities using the data model
- Voluntary. Incentives
- At the root of every data model
- Contributed manual

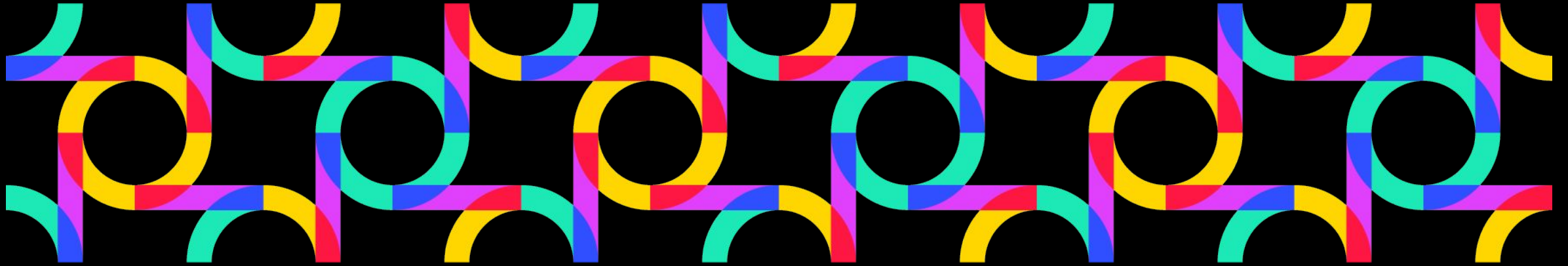
# Conceptual elements 4 / 4

## 7.-Contributors

- Format: yaml
- At the root of the subject
- File name *CONTRIBUTORS.yaml*
- Manually contributed. Incremental
- Voluntary to appear

## 8.-notes.yaml

- Format: yaml
- Text for customized specifications and README
- Voluntary content
- File name *notes.yaml* at the root of subject or root of data models
- Contributed manual



Smart Data Models: Structure and contents. Verticals

# Smart Data Models: Structure and contents. Verticals

## ■ Contents:

- *doc*: directory for specification
- *examples*: directory for examples
- *ADOPTERS.yaml* use cases of the data model
- *LICENSE.md* license of the data model. I.e. CC-BY
- *README.md* pointer to the main elements of the data model, including examples, specifications and other services
- *model.yaml* technical description of the attributes for embedding into the specifications
- *notes.yaml* additional file for customization contents for the specifications
- *schema.json* single source of truth of the model. Validates only the key-values payloads.
- *swagger.yaml* yaml file required for the interactive specification and future test of services

■	doc
■	examples
■	ADOPTERS.yaml
■	LICENSE.md
■	README.md
■	model.yaml
■	notes.yaml
■	schema.json
■	swagger.yaml

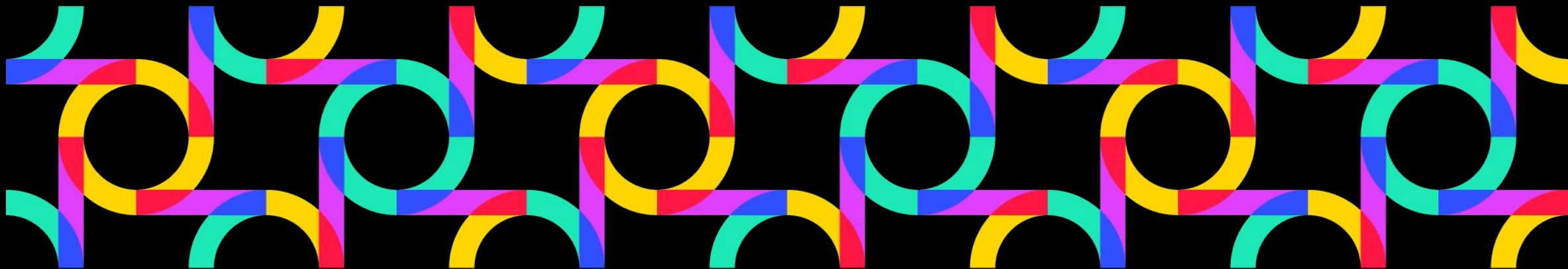
- Mandatory
- Optional
- Automatic



# Smart Data Models: Structure and contents. Verticals

## Tools

- [List data models](#): Find a data model from data model name or subject name
- [Generator of data model from json example](#): Drafts a **1.- Schema** from a **3.-Example** in json
- [Create data model from csv payload](#): Drafts a **1.- Schema** from a **3.-Example** in json
- [Data Model Editor](#): Drafts a **1.- Schema** manually (text editor)
- [Data model documentation checker](#): Checks that a **1.- Schema** includes all the descriptions
- [Generator of NGSI-LD normalized](#): Creates a normalized **3.-Example** from an existing **1.- Schema**
- [Generator of NGSI-LD keyvalues](#): Creates a keyvalues **3.-Example** from an existing **1.- Schema**
- [Mapper @context to external ontologies](#): Generates a **5.-Context** mapped to external ontologies
- [Subjects' @context merger](#): Generates a **5.-Context** from several subjects mapped with SDM URI



## Smart Data Models: Other services

# Smart Data Models: Other services

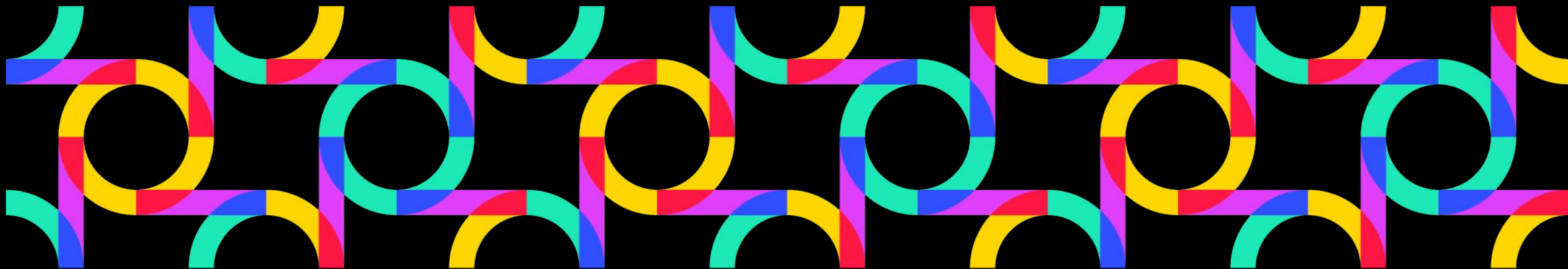
## Community

[Adopters](#): Find the examples of entities adopting a data model

[Contributors](#): Find the people who have contributed to a group of data models

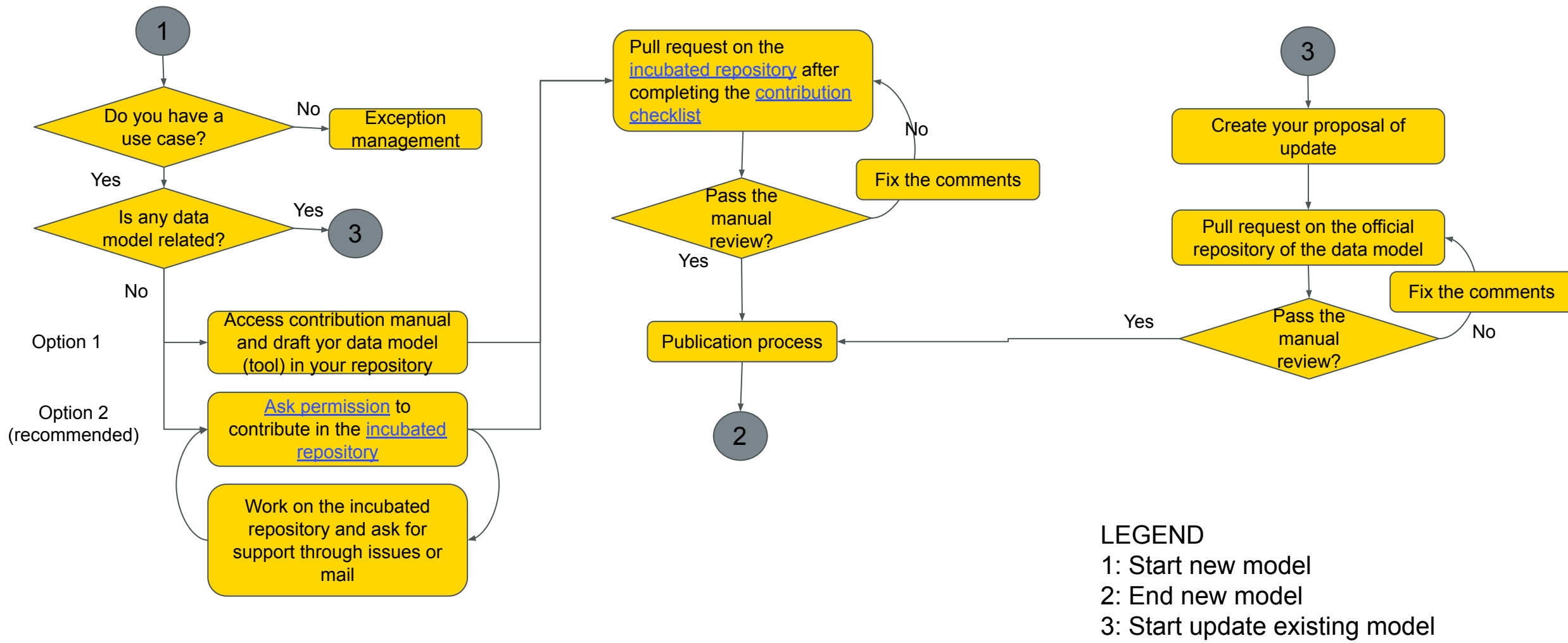
## Tools

[Versions](#) (beta): Provides links to the different versions of a data model (in progress)




Smart Data Models: Become a contributor

# Smart Data Models: Become a contributor.

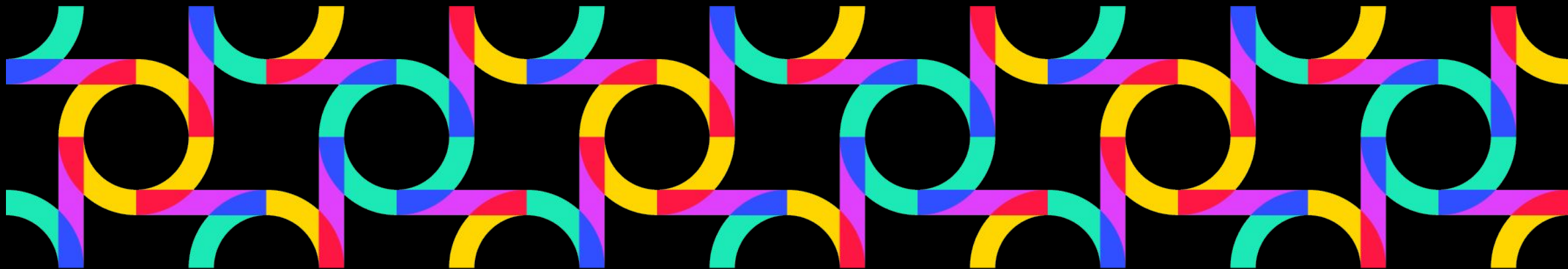


LEGEND  
1: Start new model  
2: End new model  
3: Start update existing model

 i4Trust

# Smart Data Models: Contribution manual

- Available at [https://bit.ly/contribution\\_manual](https://bit.ly/contribution_manual)



## Smart Data Models: Exercises

# Exercises

- Expected knowledge from the participants
  - Knowledge of NGSI-v2 and NGSI-LD and their differences
  - Some work with a Context Broker (any of them)
  - JSON and JSON Schema
  - Git and GitHub concepts
  - A code editor like PyCharm



# Before we start. Checking services @smartdatamodels

- Generate the key-values of payload
  - Use the script or manually with your code

editor



JSON		Raw Data	Headers
Save		Copy	Collapse All Expand All Filter JSON
id:		"urn:ngsi-ld:StreetlightModel:id:NYEG:63921792"	
▼ dateCreated:		"Property"	
type:		"Property"	
▼ value:		"DateTime"	
@type:		"DateTime"	
@value:		"2018-03-28T16:16:36Z"	
▼ dateModified:		"Property"	
type:		"Property"	
▼ value:		"DateTime"	
@type:		"DateTime"	
@value:		"2018-05-04T15:59:29Z"	
▼ source:		"Property"	
type:		"Property"	
▼ value:		"Experience dark cup none alone. Southern official east mission benefit. Follow bar focus blood."	
▼ name:		"Property"	
type:		"Property"	
▼ value:		"Third smile public three share. Close third less maybe prove."	
▼ alternateName:		"Property"	
type:		"Property"	
▼ value:		"Develop voice bit entire forget. Whose finally option baby modern. Describe fall nor by can. Indicat"	
▼ description:		"Property"	
type:		"Property"	
▼ value:		"Suffer air debate mission this leader. Identify moment treatment tonight. Notice young trouble telev"	
▼ dataProvider:		"Property"	
type:		"Property"	
▼ value:		"Candidate other candidate blood may let world. Base when people better reason task follow many. Stat"	
▼ owner:		"Property"	
type:		"Property"	
▼ value:		"urn:ngsi-ld:StreetlightModel:items:SDQY:46616655"	
0:		"urn:ngsi-ld:StreetlightModel:items:SDQY:46616655"	
1:		"urn:ngsi-ld:StreetlightModel:items:EAB0:19562265"	

# Exercises

- It is required **your github users** to be written here ([http://bit.ly/github\\_users](http://bit.ly/github_users)) to grant you access to the test repository.
  - If you do not have a github user, then go here: [http://bit.ly/register\\_github](http://bit.ly/register_github)
- Use the repository test. <https://github.com/smart-data-models/test>
- The final exercise is to **submit a complete data model**.
- Complete the creation of an official data model through all its steps.
- It will be done with official sources, so the result of the exercise, if completed, will become an official data model of the program.
- Comments to the contribution manual will be incorporated
- If not completed during session time It can be completed afterwards

# Become a contributor

Get live support by booking a 30 min slot [here](#)

[Search descriptions](#) [Contributor agreement](#) [Contact](#) [Learning zone](#) [Attributions](#) [How to use data models](#)

Smart Data Models

A GLOBAL PROGRAM LED BY

FIWAREIUDXOPEN & AGILE SMART CITIEStmfor

Tools ▾

Support ▾

Documentation ▾

About ▾

Search

Community ▾

New data model request ▾

### Domains

Smart Cities	Smart Agrifood	Smart Water	Smart Energy
Smart Environment	Smart Robotics	Smart Sensoring	Cross sector
Smart Aeronautics	Smart Destination	Smart Health	Smart Manufacturing
Lifecycle repositories	Incubated	Harmonization	

### schema.json

1.- ☐ Do you have a schema.json?

2.- ☐ Is the schema documented according to the contribution manual (Section 1)?  
[Contribution manual](#)

3.- ☐ Have you passed the schema checker? (see below)  
[schema checker](#)

4.- ☐ Have you checked the -don't do these- section in the contribution manual (Section 4) ?  
[Contribution manual](#)

### Examples

5.- ☐ Have you the 4 examples (NGSI and NGSI-LD, key-values and normalized each)?

6.- ☐ Have you checked the key-values validate against the schema with the validation tool?  
[Validate example](#)

### Optional documents

7.- ☐ Do you want to be credited for your contribution?

8.- ☐ If yes, Are you including in your PR modifications to the contributors.yaml?

9.- ☐ Do you want to customize the specifications?

10.- ☐ If yes, Are you including in your PR modifications to the notes.yaml?

### Contribution

11.- ☐ Have you signed the contribution agreement?  
[Contribution Agreement](#)

42



# Exercises

## Data sources

These data sources have dozens of properties. We'll only take a few for the exercise.

- <https://www.schema.org/MedicalCondition>
- <https://www.schema.org/MedicalGuideline>
- <https://www.schema.org/Drug>
- <https://www.schema.org/MedicalScholarlyArticle>
- <https://www.schema.org/LocalBusiness>
- <https://www.schema.org/Restaurant>
  
- Other available. <https://www.schema.org/docs/health-lifesci.home.html>

Take a minimum of 5 properties, one an object / array.

# Exercises

## Data sources

Other data sources could be valid as well as:

1. It has documented attributes
2. It has clear data types for the attributes
3. It is on use in some real case scenario

# Exercises

- Steps:
  1. Access to the data source
  2. Open the editor
  3. Paste the data definitions in the editor according to the instructions.
  4. Generate the json schema
  5. Validate the json schema
  6. Generate the example of payload
  7. Validate the example against the schema
  8. Submit the new data model

# Exercises

- Review the contribution manual
  1. Contribution manual is linked in the main page or in [https://bit.ly/contribution\\_manual](https://bit.ly/contribution_manual)

# Exercises

- Access to the data source. Examples available.
  - a. <https://www.schema.org/MedicalCondition>
  - b. <https://www.schema.org/MedicalGuideline>
  - c. <https://www.schema.org/Drug>
  - d. <https://www.schema.org/MedicalScholarlyArticle>
  - e. <https://www.schema.org/LocalBusiness>
  - f. <https://www.schema.org/Organization>
  - g. <https://www.schema.org/Restaurant>
  - h. <https://www.schema.org/docs/health-lifesci.home.html>



# Exercises

- Open the online editor

The screenshot shows the Smart Data Models website. The header includes the logo and the text 'Smart Data Models', followed by 'A GLOBAL PROGRAM LED BY' and logos for FIWARE, IUDX, OPEN & ABLE SMART CITIES, and tmforu. The navigation bar contains links for Tools, Support, Documentation, About, Search, Community, and New data model request. The 'Tools' dropdown menu is open, listing various tools. The 'Data model editor' option is highlighted with a red rectangle. Below the dropdown, a table lists various smart city domains.

Smart Agrifood	Smart Water	Smart Energy
Smart Robotics	Smart Sensoring	Cross sector
Smart Destination	Smart Health	Smart Manufacturing
Incubated	Harmonization	

Contributors

Smart Cities domain, Smart Energy domain, Smart Environment domain, Smart Robotics domain, Smart Water Domain, Smart-Sensing domain, Smart Manufacturing domain, SmartDestinations, SmartHealth

The Smart Data Models Program (SDM) for the first time and you want to become a contributor, you need to know about the elements compiled at the beginning of the program. If you are already a contributor maybe you want to review the contribution manual



## Smart Data Models: Exercises

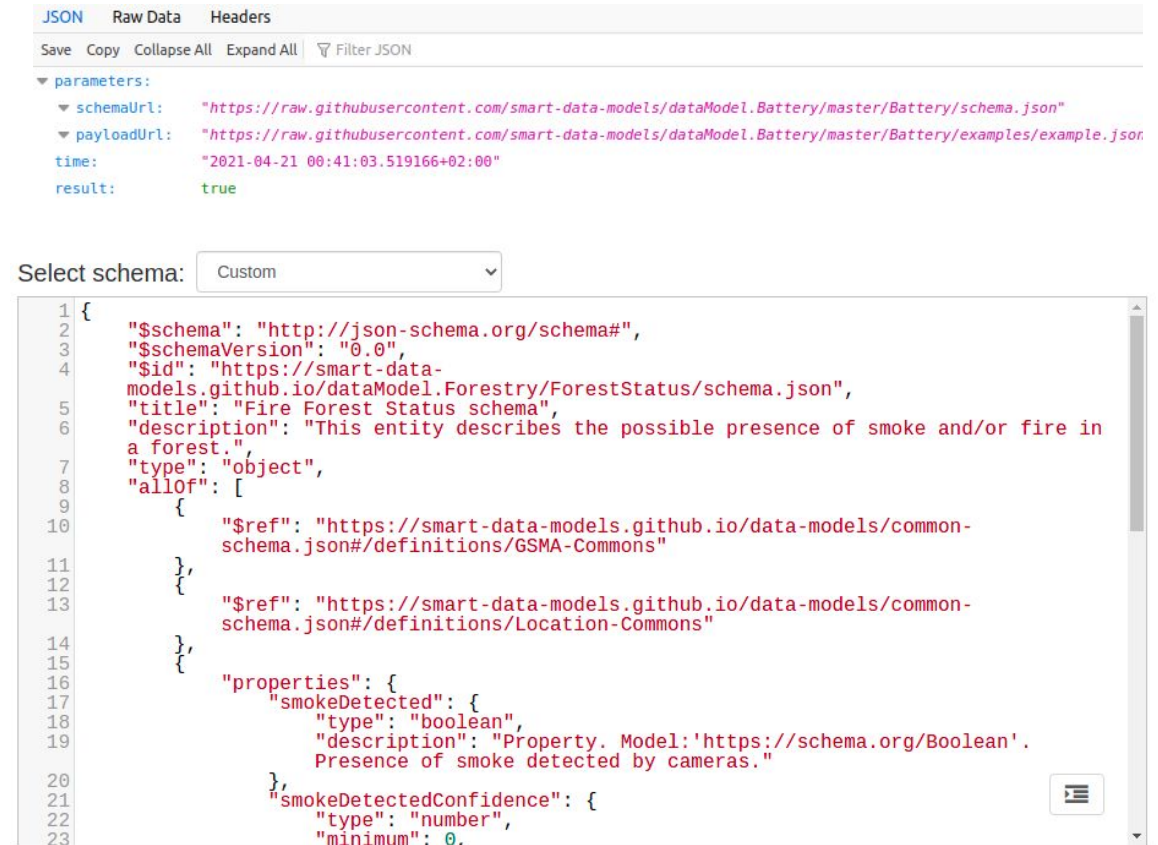
# Exercises

- Generate the json schema
  - Fix possible errors
  - Provide more detail / limits
  - Include Units
  - Include Enumerations
  - Include Privacy
  - Include limitations to values
  - Required Properties

```
$schema: "http://json-schema.org/schema#"
$schemaVersion: "0.0"
▼ $id: "https://smart-data-models.github.io/name of the subject/DataModelName/schema.json"
title: " Smart Data Models - Data Model Name"
description: "Description of the data model."
type: "object"
▼ allOf:
  ▼ 0:
    $ref: "https://smart-data-models.github.io/data-models/common-schema.json#/definitions/GSMA-Commons"
  ▼ 1:
    $ref: "https://smart-data-models.github.io/data-models/common-schema.json#/definitions/Location-Commons"
  ▼ 2:
    properties:
      ▼ property1:
        type: "string"
        description: "Property. Model:'https://schema.org/Text'. Description of the property 1"
      ▼ property2:
        type: "number"
        description: "Property. Model:'https://schema.org/Number'. Description of the property 2"
      ▼ property3:
        type: "boolean"
        description: "Property. Model:'https://schema.org/Boolean'. Description of the property 3"
      ▼ property4:
        type: "array"
        description: "Property. Description of the property 4"
        items: {}
      ▼ property5:
        type: "object"
        description: "Property. Description of the property 5"
        properties: {}
      ▼ property6:
        type: "string"
        format: "date-time"
        description: "Property. Model:'https://schema.org/DateTime'. Description of the property 6"
```

# Exercises

- Validate the json schema
  - Validation of the json schema  
<https://www.jsonschemavalidator.net/>
  - Validation of the documentation  
<https://smartdatamodels.org/index.php/data-models-contribution-api/>



The screenshot displays the JSON Schema Validator web application. At the top, there are tabs for 'JSON', 'Raw Data', and 'Headers'. Below these, there are buttons for 'Save', 'Copy', 'Collapse All', 'Expand All', and a 'Filter JSON' dropdown. The main area is divided into two sections. The top section, labeled 'parameters:', shows the input details: 'schemaUrl' is 'https://raw.githubusercontent.com/smart-data-models/dataModel.Battery/master/Battery/schema.json', 'payloadUrl' is 'https://raw.githubusercontent.com/smart-data-models/dataModel.Battery/master/Battery/examples/example.json', 'time' is '2021-04-21 00:41:03.519166+02:00', and 'result' is 'true'. The bottom section, labeled 'Select schema:', has a dropdown menu set to 'Custom'. Below this, a JSON schema is displayed in a code editor. The schema is a JSON object with properties: '\$schema' (http://json-schema.org/schema#), '\$schemaVersion' (0.0), '\$id' (https://smart-data-models.github.io/dataModel.Forestry/ForestStatus/schema.json), 'title' (Fire Forest Status schema), 'description' (This entity describes the possible presence of smoke and/or fire in a forest.), 'type' (object), and 'allOf' (an array of two references to common schemas). The 'properties' section contains two objects: 'smokeDetected' (boolean, description: Property. Model: 'https://schema.org/Boolean'. Presence of smoke detected by cameras.) and 'smokeDetectedConfidence' (number, minimum: 0).

```
1 {
2   "$schema": "http://json-schema.org/schema#",
3   "$schemaVersion": "0.0",
4   "$id": "https://smart-data-
5 models.github.io/dataModel.Forestry/ForestStatus/schema.json",
6   "title": "Fire Forest Status schema",
7   "description": "This entity describes the possible presence of smoke and/or fire in
8 a forest.",
9   "type": "object",
10  "allOf": [
11    {
12      "$ref": "https://smart-data-models.github.io/data-models/common-
13 schema.json#/definitions/GSMA-Commons"
14    },
15    {
16      "$ref": "https://smart-data-models.github.io/data-models/common-
17 schema.json#/definitions/Location-Commons"
18    }
19  ],
20  "properties": {
21    "smokeDetected": {
22      "type": "boolean",
23      "description": "Property. Model: 'https://schema.org/Boolean'.
24 Presence of smoke detected by cameras."
25    },
26    "smokeDetectedConfidence": {
27      "type": "number",
28      "minimum": 0
29    }
30  }
31 }
```

✓ No errors found. JSON validates against the schema

# Exercises

- Generate the normalized example of payload
  - I.e. NGSI-LD is available
- Needs edit for meaningful data



JSON		Raw Data	Headers
Save		Copy	Collapse All Expand All Filter JSON
id:		"urn:ngsi-ld:StreetlightModel:id:NYEG:63921792"	
▼ dateCreated:		"Property"	
type:		"Property"	
▼ value:		"DateTime"	
@type:		"DateTime"	
@value:		"2018-03-28T16:16:36Z"	
▼ dateModified:		"Property"	
type:		"Property"	
▼ value:		"DateTime"	
@type:		"DateTime"	
@value:		"2018-05-04T15:59:29Z"	
▼ source:		"Property"	
type:		"Property"	
▼ value:		"Experience dark cup none alone. Southern official east mission benefit. Follow bar focus blood."	
▼ name:		"Property"	
type:		"Property"	
▼ value:		"Third smile public three share. Close third less maybe prove."	
▼ alternateName:		"Property"	
type:		"Property"	
▼ value:		"Develop voice bit entire forget. Whose finally option baby modern. Describe fall nor by can. Indicat"	
▼ description:		"Property"	
type:		"Property"	
▼ value:		"Suffer air debate mission this leader. Identify moment treatment tonight. Notice young trouble telev"	
▼ dataProvider:		"Property"	
type:		"Property"	
▼ value:		"Candidate other candidate blood may let world. Base when people better reason task follow many. Stat"	
▼ owner:		"Property"	
type:		"Property"	
▼ value:		"urn:ngsi-ld:StreetlightModel:items:SDQY:46616655"	
0:		"urn:ngsi-ld:StreetlightModel:items:EAB0:19562265"	
1:			



# Exercises

- Validate the payload (**keyvalues**). <https://www.jsonschemavalidator.net/>

Select schema: Custom

```
1 {
2   "$schema": "http://json-schema.org/schema#",
3   "$schemaVersion": "0.0",
4   "$id": "https://smart-data-
5 models.github.io/dataModel.Forestry/ForestStatus/schema.json",
6   "title": "Fire Forest Status schema",
7   "description": "This entity describes the possible presence of smoke and/or fire in
8 a forest.",
9   "type": "object",
10  "allOf": [
11    {
12      "$ref": "https://smart-data-models.github.io/data-models/common-
13 schema.json#/definitions/GSMA-Commons"
14    },
15    {
16      "$ref": "https://smart-data-models.github.io/data-models/common-
17 schema.json#/definitions/Location-Commons"
18    },
19    {
20      "properties": {
21        "smokeDetected": {
22          "type": "boolean",
23          "description": "Property. Model:'https://schema.org/Boolean'.
24 Presence of smoke detected by cameras."
25        },
26        "smokeDetectedConfidence": {
27          "type": "number",
28          "minimum": 0.
29        }
30      }
31    }
32  ]
33 }
```

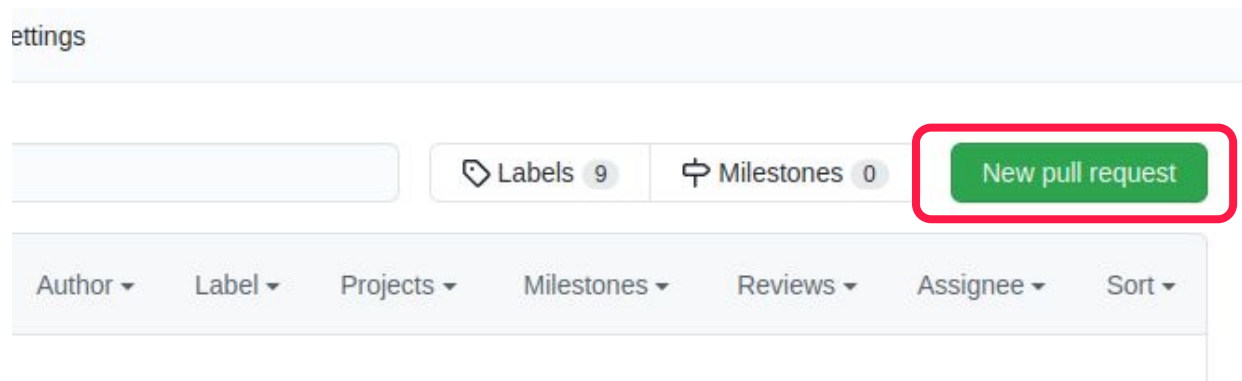
Input JSON:

```
1 {
2   "id": "FireForestStatus-South-1",
3   "dateObserved": "2021-02-24T00:00:00Z",
4   "location": {
5     "type": "Point",
6     "coordinates": [42.206302, -7.887465]
7   },
8   "name": "Ourense Forest - South",
9   "description": "Status of the Ourense Forest (south)",
10  "refDevice": {
11    "type": "Relationship",
12    "object": ["ground-humidity-sensor:1"]
13  },
14  "smokeDetected": false,
15  "smokeDetectedConfidence": 0.9,
16  "fireDetected": false,
17  "fireDetectedConfidence": 0.8,
18  "fireRiskIndex": 0.1,
19  "litterCoverage": 0.6,
20  "relativeHumidity": 0.70,
21  "soilTemperature": 25
22 }
```

✓ No errors found. JSON validates against the schema

# Exercises

- Fork the incubated repository
- PR your changes to the data model



They will appear in the front page in 5 minutes maximum (right column, bottom)

## PR and issues

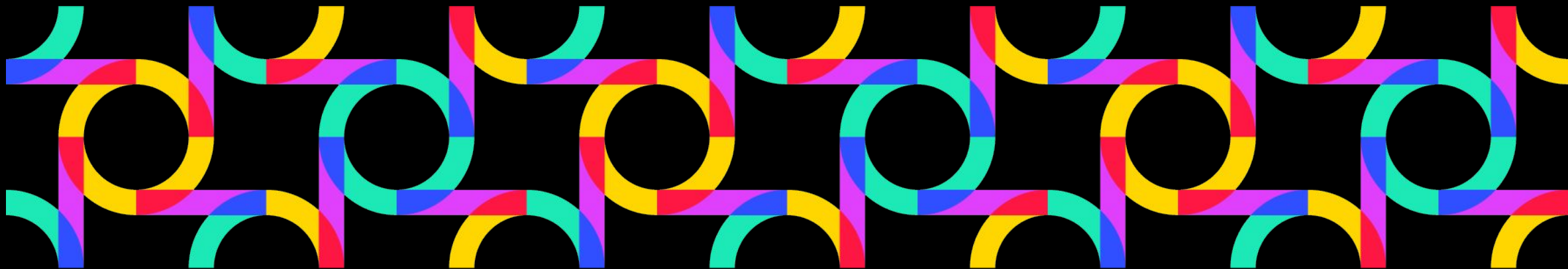
updated every 5 minutes (if empty, it means we are updating, please refresh)

dataModel.WasteManagement  
WasteContainer  
(storedWasteKind) enum, other values?

dataModel.Transportation  
remove minimum from amperage and voltage

dataModel.Battery  
Possible inconsistency in rechargeTime attr

dataModel.AutonomousMobileRobot  
Added robotics data models for autonomous mobile robots  
incubated  
commit  
added smb to OrionBroker  
dataModel.SocialMedia  
Pull request to fix inconsistencies



## Summary



# Summary

- Goal
  - Allow real interoperability between NGSII data sources
- Contribution:
  - Always possible as long as it has a use case, and comply with contribution workflow
- Use of the data models
  - Search tools for finding the right data model
  - Best not to reinvent the wheel
- Differential advantages of agile standardization
  - Quick answer
  - Do not invent
  - Easy contribution
  - Single source of truth
  - Better simple and useful than technically 'correct' and powerful

# Thank you!

