



Labs for prototyping future mobility data sharing solutions in the cloud

## **D4.4 Reference Data Catalogue (V2)**

30/11/2023

Author(s): Renée OBREGON-GONZALEZ (AKKODIS), Thierry CHEVALLIER (AKKODIS), Maroua-Dorsaf DJELOUAT (AKKODIS), Mohamed KARAMI (AKKODIS), Johannes LAUER (HERE), Huy Minh NGUYEN (HERE) and Sorel SIGHOKO (AKKODIS)



MobiDataLab is funded by the EU under the H2020 Research and Innovation Programme (grant agreement No 101006879).

## Summary sheet

Deliverable Number	D.4.4
Deliverable Name	D.4.4 Reference Data Catalogue (V2)
Full Project Title	MobiDataLab, Labs for prototyping future Mobility Data sharing cloud solutions
Responsible Author(s)	Renee OBREGON-GONZALEZ (AKKODIS)
Contributing Partner(s)	AKKODIS, HERE
Peer Review	POLIS, URV
Contractual Delivery Date	30-11-2023 (instead of 30-09-2023)
Actual Delivery Date	30-11-2023
Status	Final
Dissemination level	Public
Version	V1.0
No. of Pages	88
WP/Task related to the deliverable	WP4/T4.2
WP/Task responsible	AKKODIS
Document ID	MobiDataLab-D4.4-ReferenceDataCatalogueV2_v1.0.docx
Abstract	This deliverable is a report to provide an overview of the Task 4.2 demonstrator version 2 of the MobiDataLab Reference data catalogue. This document will provide an update on the reference data catalogue solutions selected in the context of the Transport Cloud and it will cover their implementation evolution in relation to the virtual and living labs proposed by MobiDataLab.

## Legal Disclaimer

MOBIDATALAB (Grant Agreement No 101006879) is a Research and Innovation Actions project funded by the EU Framework Programme for Research and Innovation Horizon 2020. This document contains information on MOBIDATALAB core activities, findings, and outcomes. The content of this publication is the sole responsibility of the MOBIDATALAB consortium and cannot be considered to reflect the views of the European Commission.

## Project partners

Organization	Country	Abbreviation
AKKODIS	France	AKKODIS
HERE GLOBAL B.V.	Netherlands	HERE

## Document history

Version	Date	Organization	Main area of changes	Comments
0.1	17/07/2023	AKKODIS	Table of content	Initial version
0.2	20/09/2023	AKKODIS, HERE	All	Draft version
0.3	27/10/2023	AKKODIS	CKAN	Draft version
0.4	06/11/2023	HERE	GeoNetwork	Draft version
0.5	15/11/2023	AKKODIS	Executive summary, introduction and conclusion	Draft consolidation
0.6	17/11/2023	URV	All	Peer Review
0.7	22/11/2023	POLIS	All	Peer Review and feedback
0.8	24/11/2023	AKKODIS	All	Rework
0.9	24-30/11/2023	AKKODIS	All	TL + Coordinator Quality Check
1.0	30/11/2023	AKKODIS	All	Submission

## Executive Summary

The reference data catalogue (task 4.2) is part of the Transport Cloud (i.e., a federation of cloud services) demonstrators. This task aims to improve the findability of open data catalogue referencing transport datasets (using common catalogue software systems) and corresponding metadata (in the territorial context and specific domains) of the “Reference Group” of the MobiDataLab stakeholders and to provide a state-of-the-art. In this delivery, it is going to be demonstrated how the goals of the task were achieved by:

- cataloguing transport data in the local context of the project stakeholders (that can be reused by mobility digital services like journey planners)
- cataloguing the use case data that can be used to enrich stakeholder transport datasets
- providing human-readable and machine-readable metadata available in multiple formats:
  - metadata can be discovered in the MobiDataLab CKAN and GeoNetwork catalogues via a platform and the catalogue’s API,
- reusing existing standards and popular vocabularies (i.e., DCAT, DCAT-AP and CSW),
- providing explicit dataset metadata (title, description, keywords, publication date, source, publisher, data format, spatial coverage, etc.) allowing both human understanding and automatic discovery by software agents,
- providing structural information about the internal structure of the datasets (e.g., JSON-LD, XML schema, etc.), making possible to interpret manually/automatically data schemas.

This deliverable recalls the reasoning behind the choices regarding the cataloguing solutions, their portability and the relevance of combining them. This document will cover their implementation evolution in relation to the virtual and living labs proposed by MobiDataLab.

In this version, it was shown how to discover, manage, visualize, harvest and translate metadata in the chosen data catalogues.

Furthermore, to improve the findability of datasets we provided improvements to the existent CKAN functionalities by translating all the metadata of datasets, adding a tag filter into the harvester and correcting the bugs of Solr. In this delivery, a scrapping method was also covered as a temporary solution for obtaining open data which is published on websites which are not yet fully interoperable.

This deliverable is a report whose main components are two video demonstrations of the MobiDataLab catalogues. This video is accessible on the following MobiDataLab GitHub repository: [Reference Data Catalogue demo v2](#)

The contributions made are also available on the following links:

- [Addition of tags filters and fix multivalue query syntax](#)
- [Code implementation of the translation added on the harvester](#) ([new optional argument](#) and support for dataset auto translation for [DCAT](#) and [CSW inspire](#))
- [Reported issues](#)

## Table of contents

1. INTRODUCTION.....	11
1.1. PURPOSE OF THE DELIVERABLE.....	11
1.2. STRUCTURE OF THE DELIVERABLE.....	12
1.3. REFERENCE GROUP OF MOBILITY STAKEHOLDERS .....	12
1.3.1. Reference group of local organizations .....	12
1.3.2. Reference group of local organizations proposing challenges .....	12
1.3.3. Reference group of international organizations.....	13
1.4. DATA SOURCES FROM THE REFERENCE GROUP .....	13
1.4.1. Initial inventory .....	13
1.4.2. Challenges from the reference group and use case data .....	16
1.4.3. Data providers and data consumers.....	16
1.4.4. Catalogue software systems.....	17
2. MOBIDATALAB SOFTWARE CATALOGUE SYSTEMS FOR MOBILITY DATA DISCOVERY ..	20
2.1. CATALOGUES SERVICES DESIGN .....	21
2.1.1. Metadata architecture adoption .....	21
2.1.1.1. What is a metadata catalogue service and its role?.....	21
2.1.2. Why CKAN?.....	22
2.1.1. Why GeoNetwork?.....	22
2.1.2. Why both CKAN and GeoNetwork? .....	23
2.1.3. Why not other catalogues? .....	23
2.1.4. Initial implementation .....	24
3. CKAN DEMONSTRATION.....	25
3.1. INITIAL IMPLEMENTATION.....	25
3.2. AUDIENCE.....	25
3.3. DISCOVERING DATA .....	25
3.4. CKAN MANAGEMENT .....	34
3.4.1. Registration and log in .....	34
3.4.2. Managing organizations.....	35
3.4.3. Managing datasets .....	37
3.4.4. Managing groups .....	39
3.4.5. Data and metadata storage .....	41
3.4.5.1. FileStore API.....	42
3.4.5.2. DataStore.....	42
3.4.5.3. DataPusher installation .....	42
3.4.6. Visualising data.....	43
3.4.7. Geospatial search extension .....	44
3.4.8. Multi-lingual management.....	45
3.4.9. Harvesting.....	46

3.4.9.1.	Configuration .....	48
3.4.9.2.	Filters .....	48
3.4.10.	<i>Translation .....</i>	<i>49</i>
3.4.11.	<i>CKAN API .....</i>	<i>50</i>
3.4.11.1.	CKAN API clients .....	50
3.4.11.2.	CKAN API authentication .....	51
3.4.11.3.	CKAN Harvester API .....	51
3.5.	<i>DIFFICULTIES .....</i>	<i>52</i>
3.5.1.	<i>Incompatibility CKAN-Solr .....</i>	<i>52</i>
3.5.2.	<i>Needs and difficulties encountered while harvesting CKAN datasets .....</i>	<i>52</i>
3.5.2.1.	Harvesting limitations .....	52
3.5.2.2.	Scraping an alternative solution when harvesting is not possible .....	53
3.5.2.3.	Scraping definition and terms of use .....	53
3.5.2.4.	Scraping with Python - usage example .....	54
3.5.2.5.	Limits .....	55
3.5.3.	<i>Cybersecurity .....</i>	<i>55</i>
3.5.4.	<i>Documentation/support .....</i>	<i>55</i>
3.5.4.1.	Debug .....	56
3.5.5.	<i>Interoperability .....</i>	<i>57</i>
3.5.6.	<i>Usability .....</i>	<i>58</i>
4.	GEONETWORK DEMONSTRATION .....	59
4.1.	INITIAL IMPLEMENTATION .....	59
4.2.	AUDIENCE .....	59
4.3.	GEONETWORK MANAGEMENT .....	60
4.3.1.	Web interface .....	60
4.3.2.	Metadata management .....	61
4.3.3.	Harvesting .....	62
4.3.4.	Interoperability .....	63
4.3.5.	User management .....	63
4.4.	CONSTRAINTS .....	64
4.4.1.	Deficiencies in harvested metadata .....	64
4.5.	DOCUMENTATION/SUPPORT .....	65
4.6.	PRODUCT MATURITY .....	65
4.7.	LEARNED FROM THE VIRTUAL AND LIVING LABS .....	66
5.	GROUND TESTING .....	67
5.1.1.1.	Virtual and living labs .....	67
5.1.1.2.	Datathon and hackathon feedback .....	67
5.1.1.3.	Expected evolutions from the virtual and living labs .....	68
5.1.1.4.	To be done .....	68
5.1.1.5.	Upcoming changes in the cloud services .....	68

5.1.1.6. Updates to come .....	69
6. CONCLUSIONS .....	70
7. ANNEXES .....	71
7.1. CONTAINERIZATION.....	71
7.2. DATA MUTUALISATION .....	73
7.3. LOAD BALANCING .....	74
7.4. ISSUES MANAGING GROUPS.....	75
7.5. FILESTORE API FUNCTIONS .....	76
7.6. HARVESTING CONFIGURATION .....	77
7.7. TRANSLATION TECHNICAL IMPLEMENTATION ON CKAN.....	79
7.8. TRANSLATION LIMITATIONS .....	80
7.9. LIST OF AVAILABLE CKAN API CLIENTS.....	81
7.10. CKAN HARVESTER API.....	83
7.10.1. Ckan API toolbox example for handling datasets .....	84

## List of figures

Figure 1: Datasets inventory file of the Reference group metadata .....	14
Figure 2: Data portals inventory file of Reference Group .....	14
Figure 3: Organizations and harvesting follow-up.....	15
Figure 4: Simple search - "mandatory term(s)" .....	26
Figure 5: Simple search "prohibited term" .....	26
Figure 6: Simple search - "quotations" .....	27
Figure 7: Advanced fielded search.....	28
Figure 8: Advanced fuzzy search .....	28
Figure 9: Search on the home CKAN page .....	29
Figure 10: Search results .....	30
Figure 11: Dataset display .....	31
Figure 12: Resources display .....	32
Figure 13: Preview of CSV file .....	33
Figure 14: Registration.....	34
Figure 15: Organizations in the MobiDataLab CKAN instance .....	35
Figure 16: Organization information .....	36
Figure 17: Parent-son tree structure display on CKAN (organizations page level).....	37
Figure 18: Parent-son tree structure display on CKAN (datasets page level) .....	37
Figure 19: Create dataset .....	38
Figure 20: Add Resource(s) .....	39
Figure 21: Edit or delete dataset .....	39
Figure 22: Groups created in CKAN.....	40
Figure 23: Add a dataset into a Group .....	41
Figure 24: Create a Group .....	41
Figure 25: Data pusher test.....	43
Figure 26: Managing views in a CKAN resource page .....	43
Figure 27: Dataset added to the DataStore preview .....	44
Figure 28: Multi-lingual management .....	45
Figure 29: Add a harvest source .....	46

Figure 30: Harvest source creation page .....	47
Figure 31: Harvesting configuration .....	48
Figure 32: Configuration box .....	48
Figure 33: Translation diagram .....	49
Figure 34: API token generation.....	51
Figure 35: CKAN Harvester API endpoint .....	51
Figure 36: Debug .....	56
Figure 37: GeoNetwork home page for User Administrator .....	60
Figure 38: Search in GeoNetwork .....	61
Figure 39: Map viewer with WMS layer and annotations .....	62
Figure 40: CSW Harvester in GeoNetwork.....	62
Figure 41: User and profile management in GeoNetwork.....	63
Figure 42: Group management in GeoNetwork.....	64
Figure 43: Harvester translation command in CKAN .....	79
Figure 44: Datasets update through the API .....	85
Figure 45: Metadata file elements explanation .....	86
Figure 46: Metadata file elements example.....	86
Figure 47: Example of use of the metadata file on the CKAN API .....	87

## **List of tables**

Table 1: Commonly used metadata catalogues .....	17
Table 2: Open data catalogue systems used by Reference Group of public authorities .....	18



## Abbreviations and acronyms

Abbreviation	Meaning
API	Application Programming Interface
ArcSDE	Arc for Spatial Database Engine
CKAN	Comprehensive Knowledge Archive Network
CPU	Central Processing Unit
CSV	Comma-Separated Values
CURL	Client URL
CSW	Catalogue Service for the Web
DCAT	Data Catalog Vocabulary
DCAT-AP	DCAT Application Profile for data portals in Europe
EC	European Commission
ETA	Estimated Time of Arrival
EFTA	European Free Trade Association
FAIR	Findable, Accessible, Interoperable and Reusable
GIS	Geographic Information System
GUI	Graphic User Interphase
KML	Keyhole Markup Language
KMZ	Keyhole Markup Language, Zipped
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
JAR	Java ARchive

JVM	Java virtual machine
NAP	National Access Point
NAPCORE	National Access Point Coordination Organization for Europe
ODS	OpenDataSoft
OSGeo	Open Source Geospatial Foundation
CloudOps	Cloud Operations
RDF	Resource Description Framework
SysAdmin	System Administrator
SaaS	Software as a Service
URV	Universitat Rovira I Virgili
WFS	Web Feature Service
WM	Virtual Machine
WMS	Web Map Service
WMTS	Web Map Tile Service
WP	Work Package

# 1. Introduction

The MobiDataLab Transport Cloud is a cloud-based prototype platform for sharing transport data, accessible to interested mobility actors. This platform, technically designed according to federated cloud principles, shows how to facilitate access to mobility data in an open, interoperable and privacy-preserving way, using open tools. In particular, the aim of the MobiDataLab Transport Cloud is to make mobility data FAIR, i.e.:

- Findable – allowing the discovery of data, either static or dynamic
- Accessible – providing access to mobility data
- Interoperable – prototyping data processors for adding value to the data
- Reusable – demonstrating anonymisation and privacy-preserving tools

To showcase that the project meets these four objectives, four demonstrators have been prepared as part of the prototype implementation of the WP4, each one of them presented in two versions. In the context of the H2020 program, a demonstrator (pilot or prototype) is a specific type of deliverable, which differs from other deliverables in the sense that it is not primarily in a written form; although, it is accompanied by a report such as the present document.

In the MobiDataLab context, where the open-source approach is preferred, demonstrators are available as a web server, a database, an open data portal, a source code on a repository, etc.

## 1.1. Purpose of the deliverable

The current deliverable describes the Transport Cloud demonstrators, namely the reference data catalogue (version 2), which aims to improve the Findability of transport datasets in the territorial context and specific domains of the “Reference Group” of MobiDataLab stakeholders use-cases and challenges, using common catalogue software systems. This catalogue meets a double objective:

- cataloguing transport data in the local context of the project stakeholders (that can be reused by mobility digital services like journey planners)
- cataloguing the use case data that can be used to enrich stakeholder transport datasets

Since several catalogue solutions are available in the open data ecosystem, it is often difficult for data publishers to know which solution to turn to. MobiDataLab reviewed the most used solutions about their features, interoperability capabilities, delivery methods, Software as a Service (SaaS) or on-premises, etc.

As a result, this demonstrator also recalls the reasoning behind the choices regarding these cataloguing solutions, their portability and the relevance of combining them.

## 1.2. Structure of the deliverable

The delivery structure is split into four major sections, the first one provides an overview of the reasoning behind the choices regarding the cataloguing solutions chosen {which will be presented in sections 2 and 3}, the second covers the CKAN catalogue and the third one the GeoNetwork catalogue. Both sections 2 and 3 will provide basic information about their implementation, design, and audience, but especially there will be a demonstration on how to discover metadata and how to manage the catalogues. These demonstrations are followed by information regarding the challenges faced (constraints, limitations or difficulties), the documentation and support, as well as the maturity of the product for each of the metadata catalogues. The last section, before the conclusion and the annexes, will cover the lessons learned after ground tests (during living and virtual labs).

## 1.3. Reference Group of mobility stakeholders

The MobiDataLab catalogue references all the possible open transport datasets and corresponding metadata in the territorial context and specific domains of the Reference Group of MobiDataLab stakeholders. Therefore, it is important to recall which organizations are involved, whether they are public authorities, operators, or international transport actors. The coordination of this reference group is the subject of the ongoing task T6.4 “multi-stakeholder group creation and coordination”.

### 1.3.1. *Reference group of local organizations*

- Comune di Roma, RSM, ATAC (Italy)
- Brussels MIVB (Belgium)
- Municipality of Malaga (Spain)
- Municipality of Trikala (Greece)
- Primaria Timisoara (Romania)
- Baden-Wurttemberg (Germany)
- Nouvelle-Aquitaine Mobilités (France)
- New York State (United States)

### 1.3.2. *Reference group of local organizations proposing challenges*

- City of Leuven (Belgium)
- City of Eindhoven (Netherlands)
- City of Paris (France)
- Comune di Milano, AMAT (Italy)

### 1.3.3. *Reference group of international organizations*

- Mobility Data
- Cubic Transportation Systems
- Tier Mobility

## 1.4. Data sources from the reference group

There is a vast diversity of data catalogues and data services available online.

### 1.4.1. *Initial inventory*

The MobiDataLab consortium partners identified relevant datasets and organizations in the respective areas of the Reference Group. This resulted in an inventory of “metadata” in which it is possible to filter according to city/municipality/region and other criteria such as country, organization, description, themes, keywords, data format, licenses, etc.

This inventory served as a first step to identify if the data portals contained the type of metadata required to cover the use cases – challenges datasets, the FAIR data standards, verify the access policies and licenses used, the type of data portals existent and geographical coverage, the API and services associated with them, their level of maintenance (how up to date the portal and datasets were). It also helped to add datasets to the GeoNetwork catalogue (see Figure 1).

This inventory allowed us to refine the selection of data portals to harvest through CKAN and GeoNetwork. This inventory also helped to complete a more general inventory of data sources (see Figure 2), which then evolved into an inventory of organizations and harvesting information.

This last inventory contains the information on endpoints and key information about the data portals and organizations providing data services to reference them and answer the main fill-in requisites on the CKAN (see Figure 3).

Data set description	Language description	Themes	Key words / tags	Country	Producer/Publisher/Distributor	Organization (data portal)	Link to the source website	Type of data being represented (data set)	City/Municipality/Region	Access policy	Anonymous policy	API associated
Geolocated schematic representation of the current equipment of the Infrabel network with ETCs or equivalent. The ETCs (European Train Control System) is an automatic European safety system that greatly reduces the risk of a train passing a red signal. Excessive speed, which is the cause of many train accidents, also becomes virtually impossible. Both the train and the infrastructure must be equipped with ETCs. / Représentation schématisée géolocalisée de l'équipement actuel du réseau infrabel avec le système ETCs ou équivalent. L'ETCS (European Train Control System) est un système de sécurité automatique européen qui réduit fortement le risque qu'un train dépasse un signal rouge. Une vitesse excessive, cause de nombreux accidents de train, devient aussi pratiquement impossible. Tant le train que l'infrastructure doivent être équipés de ETCs.	EN, FR	Security, Transport, Mobility	ETC, railway, train, network, speed, security system, Géolocalisation, Voies, Lignes	Belgium	Build	Infrabel	<a href="https://infrabel.opendatasoft.com/es/en/infos/etcs/etcs-informations/etcs-informations-section-noms-et-symboles-etcs-level&amp;id=etcs-level-noms-et-symboles">https://infrabel.opendatasoft.com/es/en/infos/etcs/etcs-informations/etcs-informations-section-noms-et-symboles-etcs-level&amp;id=etcs-level-noms-et-symboles</a>	Deployment of the ETCs system on the Infrabel rail network / Déploiement du système ETCs sur le réseau ferré infrabel	Leuven (Flemish Brabant)	open access		yes
Identify traffic flows and congestion through urban environments and highways to adjust streetlights and issue safety alerts. Create long-term infrastructure and transportation plans based on real, comprehensive and accurate vehicle data.	EN	Transport, Mobility	traffic management for smart cities, municipal planning, hospital planning, driver safety, mapping services, vehicle health, parking, research & analysis	Netherlands	Municipality of Eindhoven	Eindhoven Open Data	<a href="#">Wijknet - Eindhoven Open Data</a>	Neighbourhood boundary / Wijkgrens	Eindhoven	open access		yes
Information sobre las ubicaciones de los paneles de mensajes variables.	EN	Transport, Communication	messages, feed	Spain	Ajuntament de Màlaga	DatosGoVes	<a href="#">Ayuntamiento de Málaga</a>	Panels de messages variables - Ayuntamiento de Málaga	Málaga	open access		yes
Information about the Málaga bike (bike sharing) system for people having a transports card	EN	Transport, Mobility	parking, bicycle, bicycle sharing, bicleta, aparcamientos	Spain	Ajuntament de Màlaga	Ajuntament de Màlaga	<a href="https://datosabiertos.malaga.eu/datos-et/malaga-bici">https://datosabiertos.malaga.eu/datos-et/malaga-bici</a>	bike sharing service	Málaga	open access	not necessary	yes
Insurers can leverage construction equipment data to provide more accurate underwriting and better premiums to the construction industry. Location data, as well as, equipment movement and utilization can be used to more precisely characterize risk. Leverage connected construction equipment data to detect, alert, and recover stolen equipment. Improve many aspects of fleet management, from equipment deployment to maintenance schedules.	EN	Transport, Mobility	real estate research, economic trend analysis, fueling services, fleet optimization, theft prevention and recovery, insurance underwriting, connected car data, automotive data, traffic data	International	Otonomo	Otonomo	<a href="#">Construction Equipment Data Dataset - Otonomo.net</a>	Construction Data	International	premium: registration is required for a data sample		yes

Figure 1: Datasets inventory file of the Reference group metadata

Country	Country code	Region/Province	City/Municipality	Themes	Organization (data portal)	Description portal/website	Link to the catalogue website	Data format	Exchange format for scheduled information	Standard / Specification	Geographical information data format	API associated	Catalogue/Platform/APP/Projects linked to this data	Recently updated	Access policy	User Case
Belgium	BE	National	National	Transport, mobility	De Lijn	Flemish Transport Company Provides connections between Brussels and neighbouring towns and cities in Flanders, with numerous stops throughout Brussels.	<a href="https://data.delijn.be/">https://data.delijn.be/</a> <a href="https://www.delijn.be/info/interactieve-ovkaart">https://www.delijn.be/info/interactieve-ovkaart</a>	csv, json, xml		GTFS, NetEx		yes		yes	registration is required to get open access and there is a limit	24825833
Belgium	BE	Brussels-Capital Region	Brussels		DataStoreBrussels	datstorebrussels is the regional platform for opening up and sharing data and services in the Brussels-Capital Region. This platform aims to meet the needs of both users and data producers, allowing them to open up and freely reuse the data and services available for the Brussels-Capital Region.	<a href="https://datstorebrussels.be/">https://datstorebrussels.be/</a>						GeoNetwork		open access	
Belgium	BE	National	National	Transport, mobility	NAP ITS BELGIUM	Belgium national access point for intelligent transport systems (NAP ITS)	<a href="https://www.transportdata.be/dataset">https://www.transportdata.be/dataset</a>	csv, json, xml	NetEx, GTFS, TN-ITS, DATEXII		WMS	yes	CKAN	yes	registration is required to get open access	21622824 8258316 32633
Germany	DE	Berlin	Berlin		Geodata Infrastructure Berlin	Spatial data infrastructure describes the efforts to create uniform access to spatial data and to provide spatial data via standardized interfaces. The geospatial offers central access to maps and other spatial data, services and applications. <a href="https://gd.berlin.de/geonetwork/intercatalogsearch/#home">https://gd.berlin.de/geonetwork/intercatalogsearch/#home</a>	<a href="https://gd.berlin.de/geonetwork/intercatalogsearch/#home">https://gd.berlin.de/geonetwork/intercatalogsearch/#home</a>	HTML, PDF, JSON, ZIP, CSV, TEXT			WMS, WFS, CSW		GeoNetwork			
Germany	DE	Berlin	Berlin		OpenDataBerlin	Open data readable by humans and machines. You will find all STIR records. The data sets are arranged in 22 categories, but can also be found using other.	<a href="https://daten.berlin.de/dataset">https://daten.berlin.de/dataset</a>	CSV, HTML, JSON, XSL, XML, XLS, DOCX, ODS, PDF		REST, OGC	TIFF, KML, WFS		CKAN	yes		
Spain	ES	National	National	metadatas catalogue	IGNES	National Geographic Institute / Instituto Geográfico Nacional (IGN)	<a href="https://www.ign.es/ign/portal/ign/portal.htm">https://www.ign.es/ign/portal/ign/portal.htm</a>	csv		REST	csv, vms				public data, open access	24
France	FR	Ile-de-France	Paris	Air quality	AirParif	Airparif measures and maps pollution to around ten meters across the whole of Ile-de-France	<a href="https://data.airparif.asso.opendata.europa.com/search?query=airparif">https://data.airparif.asso.opendata.europa.com/search?query=airparif</a>	csv, dashboard, pdf, raster		OGC-APP, OGC-APP, RSS, OGC	layer, shapefile	yes	ArcGIS	yes	open access	
France	FR	National	National	Address	DataGouvFr	French National Access Point (NAP)	<a href="https://www.data.gouv.fr/fr/datasets/4-adresse-nationale-couleur-themat">https://www.data.gouv.fr/fr/datasets/4-adresse-nationale-couleur-themat</a>	csv, pdf, json, xml		OGC					open access	22623824 832833
Greece	GR	Thessaly	Trikala	geospatial data, parking data, entertainment and culture	Trikala Open Data	Trikala Open Data / Trikala Open Data	<a href="https://data.trikala.gr/">https://data.trikala.gr/</a>	csv, pdf				yes	PyTrikala/CKAN API/ArcGIS	no	open access	24833
Greece	GR	National	National	Transport	GeoDataGovGr	GeoData.gov.gr providing	<a href="https://geodata.gov.gr/">https://geodata.gov.gr/</a>	csv, xlsx		OGC web	vms, vms	yes	CKAN	not		24

Figure 2: Data portals inventory file of Reference Group

Figure 3: Organizations and harvesting follow-up

This list is constantly being updated with the help of the reference group members during dedicated workshops. For example, on April 26th, 2022, a workshop was organized in this respect.

### 1.4.2. *Challenges from the reference group and use case data*

The scope of use case data is slightly different from the reference group data in the sense that they combine transport data and other kind of data, for example socio-demographic data. Ideally, facilitating the discovery of data should not depend on its domain and application. However, there can be more specific methods for improving the discovery of more specialised data. In the geospatial domain, specific standards can be applied, for example the Catalogue Service for the Web (CSW) improving the discovery of spatial data. Therefore, we decided to use GeoNetwork in combination to more generic open data portals such as CKAN and OpenDataSoft (Navitia).

Datasets from the Use Cases / Groups:

- a. Data for Estimated Time of Arrival (ETA) computation (traffic real-time and historic data, static map data, weather, rest time regulations, planned events like road closures etc.)
- b. Operational data (telematics data of vehicles, location of vehicles, completed stops, tour plans, driver shift time).
- c. Public transport data (static data, transportation lines, schedules, stop points, stop areas, real-time/dynamic data, disruptions, traffic alerts, next arrivals and departures, vehicle occupancy, etc.)
- d. Geographical data (cartography, addresses, points of interest)
- e. Other transport data (free-floating, ridesharing, road traffic)
- f. Environmental data portal (INSPIRE road network, Buildings (INSPIRE harmonised))
- g. Tourism data (e.g., DataTourisme)
- h. Demography and socioeconomics (population distribution, education, household income, census data)
- i. Land use (zoning plan, construction areas, neighbourhood)

### 1.4.3. *Data providers and data consumers*

These demonstrators aim to show how to make mobility data easier to find, discover and reuse. This is particularly necessary in the context of the datathon, hackathon and of the upcoming codagon organized as part of the project (see WP5 – Living and Virtual Labs).

However, this single objective must be approached very differently, depending on whether one is the owner of the data to be reused or the re-user of the data. Consequently, it is necessary to differentiate between two categories of data catalogue users:

- data providers (or producers, or publishers) – i.e., persons or groups responsible for generating and maintaining data.
- data consumers – i.e., persons or groups accessing, using and potentially post-processing data.

Data providers aim to share data either openly or with controlled access. Data consumers (who may also be producers themselves) want to be able to find, use and link to the data.



Datasets could be used by different groups of data consumers, with different interests – which data publishers cannot all know in advance. The catalogue must provide certain information that can help the reuse, such as structural metadata, descriptive metadata, access information, data quality information, provenance information, licensing information and usage information.

#### 1.4.4. Catalogue software systems

The following table lists a set of commonly used metadata catalogues.

*Table 1: Commonly used metadata catalogues*

Name	Purpose	Website
CKAN	Metadata management system for data hubs led by the Open Knowledge Foundation. CKAN is an open-source solution with an active community. Many transport authorities use it as their open data portal.	<a href="https://ckan.org/">https://ckan.org/</a>
GeoNetwork	Reference implementation for geospatial data, harvesting options, network-based system, only stores metadata	<a href="https://geonetwork-opensource.org/">https://geonetwork-opensource.org/</a>
OpenDataSoft	Used for many open data formats, option to store data and metadata.	<a href="https://www.opendatasoft.com/">https://www.opendatasoft.com/</a>
Socrata	Popular open data solution in Northern America	<a href="https://dev.socrata.com/">https://dev.socrata.com/</a>
Esri Geoportal Server	An open-source metadata catalogue management app for data discovery	<a href="https://www.esri.com/">https://www.esri.com/</a>

These different metadata catalogues each have specific capabilities for searching and managing data.

These differences correspond to different typical use cases and the support of different metadata standards.

Public authorities from the reference group use different open data catalogues:

Table 2: Open data catalogue systems used by Reference Group of public authorities

	Municipality data portal	Regional data portal	National Access Point	Other
	Primaria Timisoara		DataGovRo	
	Comune di Milano		Il portale Italiano dell'Open Data	
			NAP GR	
	Ayuntamiento de Malaga			
			NAP ITS BELGIUM	
		MobiData BW	GovDataDe	Deutsche Bahn
	Roma Capital Open Data	Lazio Region Open data	CCISS Italy	
			DataGov	
				OverheidNI
		GDI-BW GeoPortal BW		
	Geoportal of the Ministry of Environment and Energy			
	NationaalGeoregisterNL			
		PIGMA		Atmo Nouvelle-Aquitaine
	DataStoreBrussels Geo-Brussels		Infrabel	Royal Meteorological Institute
	European Environment Agency			
	Eindhoven Open Data		National Road Traffic Data Portal of the Netherlands	Navitia
	Open Data Brussels Brussels Mobility			
	Paris open Data			SNCF
	Ville d'Agen			
	Data Toulouse Métropole	DataLaRegionFr		
		Open Data of Lombardia		
		GIS NY GOV		
				AirParif
				Atmo-Occitanie
	Trikala Open Data			
	Roma Servizi per la Mobilità			
	OpenDataNederland			
		GeoportalGovRo NAP CESTRIN Romania GeoportalAncpiRo		
		Bureau of Transportation Statistics of USA		
		ERS.USDA.GOV		
		U.S. Departement of Transportation FRA		

Support is an objective of the MobiDataLab project. Therefore, a range of data reuse and metadata standards are available.

For instance, Data Catalog Vocabulary (DCAT), DCAT Application Profile for data portals in Europe (DCAT-AP), Catalogue Service for the Web (CSW), Open Geospatial Consortium (OGC), Dublin Core enable interoperability between catalogues and data users.

The MobiDataLab catalogue builds on several of these solutions, namely:

- CKAN
- OpenDataSoft
- GeoNetwork

These solutions follow different approaches which will be highlighted in this document.

## 2. MobiDataLab software catalogue systems for mobility data discovery

In version 1 of this deliverable, it was discussed the selection of two software catalogue systems, CKAN and GeoNetwork, for mobility data discovery and an overview of their main functions that could be implemented was provided. In this version, we provide a demonstration of how to discover, manage, visualize, harvest and translate metadata in the chosen data catalogues. Furthermore, some of the difficulties encountered, the actions executed to overcome them, and the improvements made to achieve the expected functions in the MobiDataLab catalogue will be covered.

Before recalling the reasons behind this selection, it is important to mention that several catalogue software systems exist, each proposing a different approach (e.g., free open-source solution or proprietary SaaS solution). Data publishers like the reference group stakeholders have had to choose one or two of these solutions for their open data portal solution, after a thorough study of their needs, resources, and expectations. A general criterion for choosing, between SaaS and on-premises, is related to the human capacity for maintenance:

- Organizations with skilled staff capable of maintaining an open data platform may find attractive to host and maintain the solution. In this case, an open-source solution such as CKAN is often more suitable.
- Organizations not having such a staff or that need a quick solution may prefer to opt for an all-in-one hosted solution such as OpenDataSoft that is more suitable in this situation.

Choosing between these different approaches may also depend on the size of the organization, the number of datasets and the frequency of updates (CKAN is interesting when there are many datasets with frequent updates).

Apart from technical and financial aspects, it was also necessary to consider the use cases. For instance, when dealing exclusively with geo-referenced data, a GIS-specific solution like GeoNetwork appears preferable. Even though there is a degree of overlap between geographic and generic catalogues, their functions might differ. In some use cases, CKAN or OpenDataSoft could be sufficient, but in a context where users may want to enrich mobility data with data from other sectors such as environmental data, it might be necessary to provide support for GIS standards (OGC, INSPIRE profiles, etc.). In this matter, GeoNetwork offers serious advantages as it has a large and well-established user base GIS community. For these specific users, well acquainted with geospatial Free and Open-Source Software, switching to more generic solutions like CKAN or OpenDataSoft may prove unnecessary.

Since a federated approach for the MobiDataLab Transport Cloud is required to demonstrate the portability of mobility digital services and to avoid any lock-in to a particular vendor or solution, an integrated catalogue is necessary.

## 2.1. Catalogues services design

MobiDataLab decided to provide CKAN and GeoNetwork catalogue services that work in tandem for complementary duties as well as for drawing the attention of different audiences. In terms of computing, their service provides a complete set of functionalities that allow to achieve a standardized action in a single place. The data catalogue format was favoured for its simplicity in comparison to a website, where data would have been hosted on a web server as a file service or a database service.

### 2.1.1. *Metadata architecture adoption*

Metadata is a descriptor structure with fields that will identify the labels used for describing the underlying data that will be reached by the user if the description matches the subject of interrogation. Metadata is the first glimpse of information that a data consumer will read before reaching the actual data: the metadata is the cover sheet that says what to expect once the data file is opened.

The metadata concept covers two goals: the data comes without being wrapped in any kind of cover and cannot be interpreted if no description comes along or before browsing that data. The metadata infrastructure layout allows the distribution of metadata while the actual data remains hosted, managed and updated at the source by the data supplier.

#### 2.1.1.1. What is a metadata catalogue service and its role?

Mobility data is scattered all around open data portals managed by the territorial entities in charge of exposing public territorial information, managing the accuracy of the data and maintaining the portal with the data history, updates and evolutions within the territory.

Among the assets proposed to the users, the platform allows them to find references to the data of interest to solve the use cases of MobiDataLab. The natural dissemination of territorial data would impose an arduous endeavour if the data consumer had to crawl the web to find data from different websites, here comes the metadata catalogue within the MobiDataLab platform to bring in a centralised location all the harvested content summarized as metadata for the data consumer to query and get the actual location of the relevant content.

The metadata catalogue role in the project is the add value to the data consumer by drastically reducing the browsing source of data and offering the user a unique source for getting references to the hosting actual data since MobiDataLab will not copy nor import the data but, rather will reference the data source along with the description of what is to be expected from the data provider.

### 2.1.2. *Why CKAN?*

CKAN is a major historical metadata server with development dating from the 2010s with deployment and customization in many cities in Europe and Oceania. Most of the European National Access Points (NAPs) use this solution, see Table 2, and often follow the recommendations of the European Union and the National Access Point Coordination Organization for Europe (NAPCORE).

The CKAN evolution makes it more and more attractive due to its modularity, the number of available extensions, and the variety of these plugins. Supplementary functionality and standards are to be noted when comparing with GeoNetwork. For instance, CKAN extensions can handle DCAT and particularly DCAT-AP which is a standard encouraged to be used by NAPCORE<sup>12</sup> for metadata exchange to achieve harmonization across NAPs.

The CKAN project being open source is advantageous, nevertheless, its main drawback is that it is not managed by any commercial entity for its development. This results in a core software that evolves but the plugins associated cannot always follow the trend: they lack proper follow-up and maintenance. For instance, the Python language on top of which CKAN is developed has not only stepped from generation 2.x to version 3.x but additionally, the Python 2.x is no longer maintained, supported and supplied by Linux editors. Therefore, any extension unconverted to Python 3 will become inoperable.

### 2.1.1. *Why GeoNetwork?*

The MobiDataLab project has passed through a comprehensive investigation to select mobility-related data catalogue services mature and standardized. Whilst this investigation, processes were conducted, some solutions were evaluated and tested. The first admissible candidate was GeoNetwork which checked all the boxes regarding both maturity and standards. This solution allows the discovery of geodata in a harmonised format, which follows the requirements of INSPIRE Directive<sup>3</sup> (Directive 2007/2/EC), that can be viewed and downloaded.

Despite all its qualities, GeoNetwork had a limitation in terms of several standards commonly expected in the mobility industry as well as in public interfaces with a lack of translation capability when it comes to importing metadata from a country whose language is not English. In addition to this inconvenience, it must be noted the absence of the plugin offer to develop adequate extensions for the project or a repository of plugins secured by GeoNetwork to choose from.

<sup>1</sup> <https://napcore.eu/metadata/>

<sup>2</sup> <https://napcore.eu/release-of-the-mobilitydcat-ap/>

<sup>3</sup> <https://inspire.ec.europa.eu/data-specifications/2892>

### 2.1.2. *Why both CKAN and GeoNetwork?*

Using both CKAN and GeoNetwork act as complementary metadata services to embrace the landscape of functionalities and standards that the platform has announced to the users. Together they also provide a visual experience (geographic maps representation over OpenStreetMap, etc.) and technical functionalities which are not public (Datastore) as they are not necessary for the discovery of datasets.

The risks of harvesting data in both platforms are known and expected (having data duplicates and splitting the users according to the metadata service with which they are the most familiar and better equipped). Still, since the audience for each technology differs it could accommodate some data consumers, for instance, data scientists, to use a solution which is compatible or that can be easily found with their tool of choice, rather than browsing metadata with a new/different metadata service.

### 2.1.3. *Why not other catalogues?*

In the US, other tools like Socrata are common practice when it comes to open-data portals and they are deployed by territorial entities to expose their local data to be made available at a data consumer range. Despite that availability, it appears that the audience will be segregated by the open-data portal due to the protocols embedded within the technology and protocol offered by the portal metadata or dataset server. That segregation distinguishes US data consumers from European and world data consumers. As a result, the standards carried by only US-deployed open-data services will not be of any use to a worldwide audience.

**DSpace** is an open-source repository software package typically used for creating open access repositories for scholarly and/or published digital content. DSpace acts as a digital archives system focused on the long-term storage, access and preservation of digital content.

**Dataverse** is an open-source web application to share, preserve, cite, explore and analyse research data NOT to be confused with Microsoft Dataverse.

These last two solutions are data servers and are rather focused on archiving, so are not the most adapted for metadata.

**DKAN** is presented as a Drupal-based build of CKAN: Drupal is a website content manager that is widely used over the internet and whose core language is PHP rather than Python in the case of CKAN. The use of DKAN will challenge the availability of plugins as well as the maintenance of these plugins compared to the plugin's counterparts in CKAN.

### 2.1.4. *Initial implementation*

The initial implementation of metadata services was motivated by the natural dissemination of workable data hosted on data servers scattered all around the internet and under the authority of territorial entities. The inner consequence of this natural distribution are the advantages of the data being managed, edited, updated (by the source), made available and hosted by the owner. The inconvenience of that natural distribution is the accuracy, data quality and file format exposed and often unusable as (pictures, pdf, text files).

Hence the necessity to get across all the data that appeared relevant to the use cases described in the deliverable D2.9 and to consult, validate and import this content to the metadata platform hosted by the MobiDataLab for the data consumers to browse in one single place data that is naturally scattered all around internet.

This initial deployment let the MobiDataLab platform with two metadata servers to complement functionality and offer more interfacing protocols to data consumers according to the way each of them is used to work (from desktop tooling to data presentation via software interconnectivity) when it comes to data browsing and data import. Therefore, the initial deployment was conducted bearing in mind the functionality aspect of the service supplying metadata and satisfying different data consumer-driven technologies.

Without the approach to take into consideration the infrastructural aspect, the platform would be exposed to an influx of the users of the internet, to withstand the full force of the queries and to reply to the data volume.

The initial deployment of the instances covered Containerization (high availability functionality), Data mutualisation and Load balancing which are detailed in the annexes.



## 3. CKAN demonstration

CKAN is an open-source open data platform used by most governments around the world to publish and manage their open data. The basis of CKAN is to make data more discoverable for users and re-users, for citizens, and for data consumers at large. CKAN helps governments manage their data better and gives the ability to browse and discover data easily. CKAN is a highly extensible product, as will be shown in the following sections. Several useful external extensions that do not come packaged with CKAN (e.g., data storage, harvesting, CSW support, etc.) can be installed separately.

Several Reference Group members and public authorities already use CKAN to publish their transport datasets: Rome, Milan, Malaga, Timisoara, etc.

### 3.1. Initial implementation

The CKAN implementation is covered more in detail in the annexes of the present document.

### 3.2. Audience

As mentioned before, the catalogue was created to make transport data as Findable, Accessible, Interoperable and Reusable (FAIR) as possible, especially for the mobility and transport community. Nevertheless, our catalogues are not limited to transport datasets so they could be used by other sectors since most of the National Access Points and municipal catalogues harvested cover many categories of datasets. In the context of the project, the main expected users are the participants of the X-thons, eventually the reference group members and data providers since we harvested datasets of their specific municipality, region, country and/or portal.

### 3.3. Discovering data

CKAN is a metadata-driven catalogue. It is precisely these metadata which provide a wealth of data exploration and discovery functionalities. To search datasets, there are usually two search boxes available on each page of the catalogue. A simple or an advanced search is supported in these boxes.

A simple search is composed of a single keyword or more without characters. A plus (+) and a minus (-) symbol can be used as modifiers in the search for a term. The + symbol designates a mandatory term to be included and the - symbol a prohibited term (to be excluded) in the search. For example, the search **+Leuven** will provide all the results of datasets containing the term Leuven. However, if we search for **Leuven -city** the results will show the same results, but without the datasets containing the word city.

Quotation marks (") are used to signal that a sentence is searched. This means that all the terms within the quotes are searched and required in that order. For example, "municipality of Eindhoven" will show all the datasets comprising that sentence (see Figure 6).

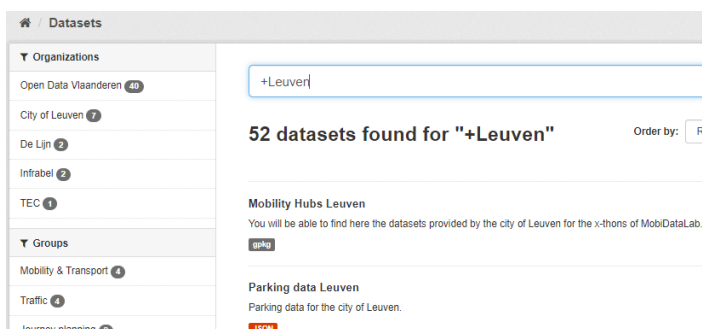


Figure 4: Simple search - "mandatory term(s)"

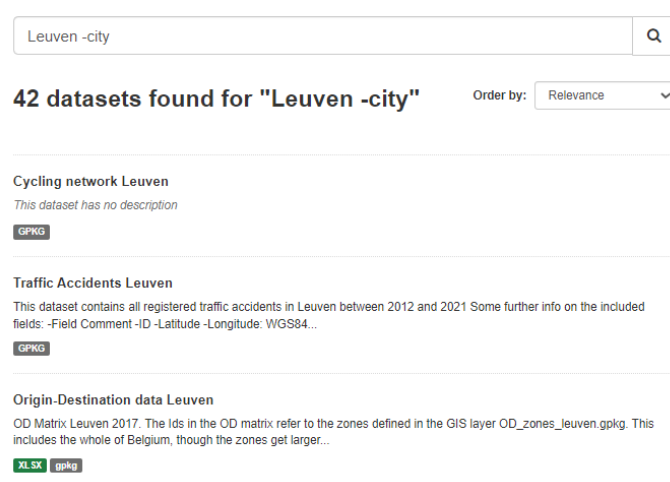


Figure 5: Simple search "prohibited term"

**95 datasets found for ""municipality of Eindhoven""**
Order by: Relevance ▼

---

**Tenders from the municipality of Eindhoven**

Administration of the final documents associated with the various phases of a tender. Documents associated with a request include: assignment announcement, request,...

<https://www.iana.org/assignments/media-types/text/csv>
<https://www.iana.org/assignments/media-types/application/json>
<https://www.iana.org/assignments/media-types/application/octet-stream>
<https://www.iana.org/assignments/media-types/application/rdf+xml>
<https://www.iana.org/assignments/media-types/application/ld+json>
<https://www.iana.org/assignments/media-types/text/turtle>
<https://www.iana.org/assignments/media-types/text/n3>
<https://www.iana.org/assignments/media-types/application/xls>
<https://www.iana.org/assignments/media-types/application/vnd.openxmlformats-officedocument.spreadsheetml.sheet>

---

**Tenders of the municipality of Eindhoven**

Administration of the final documents associated with the various phases of a tender. Documents belonging to a request include: order announcement, request for tender,...

<https://www.iana.org/assignments/media-types/application/json>
<https://www.iana.org/assignments/media-types/text/csv>
<https://www.iana.org/assignments/media-types/application/vnd.openxmlformats-officedocument.spreadsheetml.sheet>

---

**Tenders of the municipality of Eindhoven**

Administration of the final documents belonging to the different phases of a tender. Documents belonging to a request include: order announcement, request, specifications,...

<http://publications.europa.eu/resource/authority/file-type/JJSON>
<http://publications.europa.eu/resource/authority/file-type/CSV>

---

**Trees Eindhoven**

Trees in the municipality of Eindhoven

<http://publications.europa.eu/resource/authority/file-type/CSV>

Figure 6: Simple search - "quotations"

An advanced search is composed of a keyword and at least one colon (":"). This function allows to search terms per field and use wildcards ("\*", proximity matching "~" terms). The basic syntax is `field:term`.

Advanced Search Examples:

- `title:micro-mobility` will look for all the datasets containing in its title the word "micro-mobility"
- `title:share*` will look for all the datasets containing in its title a word that starts with "share" like "shared" and "sharing".
- `title:leuven || title:flemish` look for datasets containing "leuven" or "flemish" in its title.
- `title:"pedestrian zone" ~ 4` is a proximity search that looks for terms that are within a specific distance from one another. This example will look for datasets whose title has the words "pedestrian" and "zone" within a distance of four words.
- `text:ycle~` CKAN supports fuzzy searches based on the Levenshtein Distance or Edit Distance algorithm. To do a fuzzy search use the "~" symbol at the end of a single-word term. In this example, words like "cycle" or "CLE" will also be found.

title: micro-mobility

**1 dataset found for "title: micro-mobility"**

Order by: Relevance

#### Localization zones of the parking areas for micro-mobility offers

The dataset contains the localization zones of the parking areas for micro-mobility offers.

[WFS](#) [PDF](#)

Figure 7: Advanced fielded search

title: micro-mob~

**10 datasets found for "title: micro-mob~"**

Order by: Relevance

---

**Storage areas for micromobility offers**

The data set contains the locations of the parking areas for micromobility offers.

[WFS](#) [PDF](#)

---

**Storage areas for micromobility offers**

The database includes the previously set up parking areas in public and publicly accessible space for micromobility rental vehicles. Those using the rental vehicles should...

[download](#) [view](#)

---

**Storage areas for micromobility offers**

The database includes the previously set up parking areas in public and publicly accessible space for micromobility rental vehicles. Those using the rental vehicles should...

[download](#) [view](#)

---

**Storage areas for micromobility offers**

Figure 8: Advanced fuzzy search

In the figure below a simple search was done on the home page of the catalogue by typing the keywords *GTFS Milan*.

Welcome - CKAN

ckan.mobidatalab.eu

**MOBIDATALAB**  
Labs for prototyping future mobility data sharing solutions in the cloud

Log In Register

Datasets Organizations Groups About Search

**Search data**

GTFS Milan

Popular tags: opendata EU open data

**CKAN statistics**

115.1k datasets 275 organizations 25 groups

**CODAGON**  
Dive into live co-creation and data-driven solution development, embrace community feedback and explore synergies in a groundbreaking 3-week event uniting innovators, domain experts and data providers in the mobility realm!  
6 - 24 November 2023  
ONLINE  
Awards ceremony in Leuven on 27 November  
REGISTER SOON  
Codagon - Leuven

**Mobility & Transport**  
mobility- Information about public transport, transport...

**OpenDataBerlin**  
Open data readable by humans and machines. You...

**VBB timetable data via GTFS**  
The Verkehrsverbund Berlin-Brandenburg (VBB) regularly provides the current bus and train timetable data from Berlin...

**Statistics rental car 2021**  
The data set contains the number of motor vehicle rental companies registered in the specified period and the number...

**Conexxion**  
Datasets available on the Ndov Loket website. For more information please refer directly to their website. Conexxion...

**Origin-Destination travel time by public transportation**  
This information was provided by the Municipality of Milan for the x-thons of MobidataLab.

About CKAN  
CKAN API  
CKAN Association

Powered by ckan  
Language: English

Figure 9: Search on the home CKAN page

The displayed results are shown in the next figure.

The results can be filtered by organization(s), group(s), tag(s), format(s) and license(s).

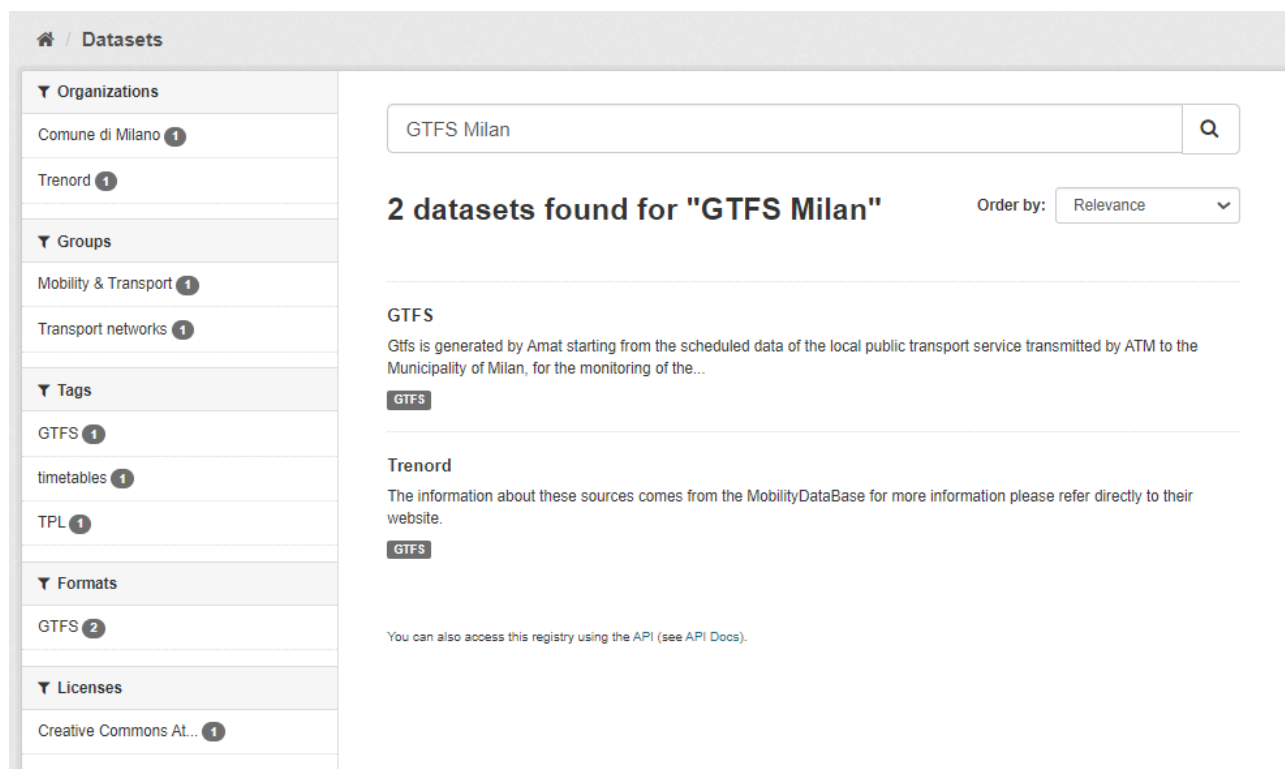


Figure 10: Search results

By selecting a dataset result, the metadata and resources attached to this dataset will be displayed (see the two examples below).

This includes the name, description, resources, tags, and other information about the dataset.

**GTFS**

Followers  
**1**

Organization

**Comune di Milano**

The Municipality of Milan identifies the Open Government paradigm as a way to create an open Public Administration that empowers citizens and businesses to innovate: open data...

[read more](#)

**Social**

Twitter

Facebook

**License**

Creative Commons Attribution

[dataset](#)

**GTFS**

Dataset Groups Activity Stream

**GTFS**

Gtfs is generated by Amati starting from the scheduled data of the local public transport service transmitted by ATM to the Municipality of Milan, for the monitoring of the Service Contract which regulates the management of the service within the territory of Milan and the urban area. The dataset is updated periodically, varies from 2 weeks to a month and is you can download it at the following link: <https://dati.comune.milano.it/gtfs.zip> For information on the structure and format of gtfs refer to the following link: <https://developers.google.com/transit/gtfs/reference?hl=it> ## Additional information No information provided This dataset was released by the municipality of Milan.

**Data and Resources**

**ds929\_gtfs.zip**  
preview NOT available

[Explore -](#)

GTFS TPL timetables

**Additional Info**

Field	Value
Maintainer	AMAT - Territory Environment Mobility Agency
Version	332
Last Updated	September 12, 2023, 7:42 PM (UTC+02:00)
Created	July 3, 2023, 11:52 AM (UTC+02:00)
Alternate identifier	[]
Conforms to	[]
Frequency	BIWEEKLY
Identifier	DS929
Issued	
Language	ITA
Modified	2023-09-12
Publisher name	Direzione Mobilità, Ambiente e Energia
Theme	[{"subthemes": [{"http://eurovoc.europa.eu/100238"}, {"theme": "TRAN"}]}
creator	[{"creator_identifier": "01199250158", "creator_name": {"en": "Unitu00e0 Open Data", "it": "Unitu00e0 Open Data"}}]
geographical_geonames_url	<a href="https://www.geonames.org/3173435">https://www.geonames.org/3173435</a>

Figure 11: Dataset display

The resource description can be displayed by a simple click on “Explore” + “More information” or it is also possible to be directly redirected to the source of the resource by a simple click on “Explore” + “Go to resource”.

According to the type of resource, sometimes the option to download directly a file is available when clicking on “Explore”.

In the example below, it is possible to see a dataset with more than one resource and how these resources can be explored or downloaded.

The screenshot shows the MobidataLab interface for the 'Bike path section' dataset. The left sidebar includes the Eindhoven Municipality logo and a list of social media links (Twitter, Facebook, License). The main content area shows the dataset name, a description, and a list of resources. Each resource has a 'DATA' icon, a name, a description, and an 'Explore' button. Below the resources is an 'Additional Info' table.

Field	Value
Source	<a href="https://www.eindhoven.nl/">https://www.eindhoven.nl/</a>
Last Updated	August 18, 2023, 3:16 PM (UTC+02:00)
Created	May 15, 2023, 1:24 PM (UTC+02:00)
Country code	NL
Group	Micro-mobility
Municipality	Eindhoven
Region	North Brabant

Figure 12: Resources display

Many types of resources can also be previewed directly on the resource page. Spreadsheets (.CSV and .XLS) can be previewed in a grid view. Map and graph views can also be previewed if the data is suitable.



Here is an example of how the resources' description metadata and the preview for a CSV looks like (Figure 13):

Organizations / Ministry of the... / Unfinished works / Unfinished Works 2016 CSV format

Unfinished Works 2016 CSV format

[Manage](#)
[Download](#)

URL: <https://dati.mit.gov.it/catalog/dataset/35772d25-5b83-42c0-b7fd-0cf874ba4ac3/resource/7749fod7-1df8-4a4f-a24c-eed5bd96726e/download/opereincompiute2...>  
Data for 2016 published on June 30, 2017.

[Data Explorer](#)

[Fullscreen](#)
[Embed](#)

Grid
Graph
Map
752 records
1 - 100

Search data ...
Go
Filters

public...	anno_rif	cup	cup_ma...	ambito...	denomi...	codice...	codice...	natura...	tipolog...	codice...	tipolog...	localizz...	localizz...
Regione...	2016	B59D10...			COMU...	489010...	3	REALIZ...	b) i lavo...	1	NUOVA...	3018080	
Regione...	2016	G44E89...			COMU...	820005...	3	REALIZ...	a) i lavo...	1	NUOVA...	17078019	ITF5
Regione...	2016	D31B91...			Comun...	800019...	3	REALIZ...	b) i lavo...	1	NUOVA...	17078038	ITF5
Regione...	2016	D69D11...			comune...	8000411...	3	REALIZ...	a) i lavo...	1	NUOVA...	17078084	ITF5
Regione...	2016	I15E080...			Comun...	850004...	3	REALIZ...	b) i lavo...	51	COMPL...	17078043	ITF5
Regione...	2016	E83B06...			COMU...	800044...	3	REALIZ...	b) i lavo...	4	RISTR...	17078014	ITF5
Regione...	2016	D38D14...			COMU...	800183...	3	REALIZ...	b) i lavo...	51	COMPL...	3018089	
Regione...	2016	E45I870...			AMM.N...	800005...	3	REALIZ...	a) i lavo...	3	RECUP...	17077008	
Regione...	2016	H28B80...			COMU...	800049...	3	REALIZ...	a) i lavo...	51	COMPL...	17078017	ITF5
Regione...	2016	B34B82...			comune...	8100117...	3	REALIZ...	a) i lavo...	1	NUOVA...	17077007	ITF5
Regione...	2016	I73G01...			Comun...	820009...	3	REALIZ...	a) i lavo...	1	NUOVA...	17077029	ITF5
Regione...	2016	H11B03...			PROVI...	800027...	3	REALIZ...	b) i lavo...	51	COMPL...	17078007	ITF5
Regione...	2016	C76J15...			COMU...	800025...	3	REALIZ...	a) i lavo...	54	COMPL...	17077011	ITF5
Regione...	2016	E53D06...			COMU...	800042...	3	REALIZ...	a) i lavo...	51	COMPL...	17078002	ITF5
Regione...	2016	E33F09...			Consoz...	91800787	3	REALIZ...	b) i lavo...	51	COMPL...	17078083	
Regione...	2016	J72J10...		Regionale	COMU...	238090...	3	REALIZ...	c) i lavo...	7	MANUT...	18079033	ITF6
REGIO...	2016			Regionale	Fondazi...				b) i lavo...			10054039	
Regione...	2016	F71B07...		Regionale	Provinci...	800037...	3	REALIZ...	a) i lavo...	1	NUOVA...	18078003	ITF6
REGIO...	2016	I33F050...		Regionale	Region...				b) i lavo...			10054024	
Regione...	2016	I69B110...		Regionale	Comun...	800202...	3	REALIZ...	a) i lavo...	1	NUOVA...	15085135	ITF3
Regione...	2016	G47B90...		Regionale	Comun...	581980...	3	REALIZ...	b) i lavo...	1	NUOVA...	15083083	ITF3
Regione...	2016	H49G09...		Regionale	COMU...	800215...	3	REALIZ...	b) i lavo...	55	COMPL...	15085014	ITF3

Resources

Unfinished Works 2016...

metadata

metadata (PDF)

Unfinished Works 2017...

Unfinished works 2016...

Unfinished works 2017...

Social

Additional Information

Field	Value
Data last updated	March 4, 2022
Metadata last updated	August 22, 2023
Created	unknown
Format	CSV
License	CC-BY-4.0

Show more

Figure 13: Preview of CSV file

Datasets can be also discovered on the datasets, organizations, and groups pages with the help of filters.

**MOBIDATALAB**

MOBIDATALAB – H2020 G.A. No. 101006879

Funded by the  
European Union

## 3.4. CKAN management

### 3.4.1. Registration and log in

Registration is useful for “following” datasets, but it is not necessary for discovering and accessing data, so registration has more interest for administrators and organizations publishing and personalizing features. To create a user ID, click on “Register” at the top of any page. Then, it is required to enter the following information: a username (mandatory, using only letters, numbers, - and \_ characters), full name, e-mail address (mandatory) and password (mandatory). To complete this step, click on “Create Account”.

**MOBIDATALAB**  
Leads for prototyping future mobility data sharing solutions in the cloud

Log in **Register**

Datasets Organizations Groups About Search

**Registration**

**Why Sign Up?**  
Create datasets, groups and other exciting things

**Register for an Account**

\* Username:  
username

Full Name:  
Joe Bloggs

\* Email:  
joe@example.com

\* Password:  
\*\*\*\*\*

\* Confirm:  
\*\*\*\*\*

Profile picture URL:  
http://example.com/my-image.jpg

Profile picture:

\* Required field

Create Account

Figure 14: Registration

When the fields are filled in correctly, CKAN will create your user account and automatically log you in. Otherwise, CKAN will tell you the problem and enable you to correct it.

### 3.4.2. Managing organizations

CKAN uses organizations to represent different data publishers on the same data portal. In general, organizations correspond to public authorities, ministries, government departments, etc. In the MobiDataLab catalogue, organizations mainly correspond to reference group stakeholders or organizations providing data services or datasets related to the challenges reference group.

Organizations are part of CKAN's authorization system and different user rights (create, edit and delete) can be assigned to different organizations. Therefore, organizations are one of the main elements to provide before adding any metadata since a dataset must be linked directly to an organization that can control its modification and make it public or private.

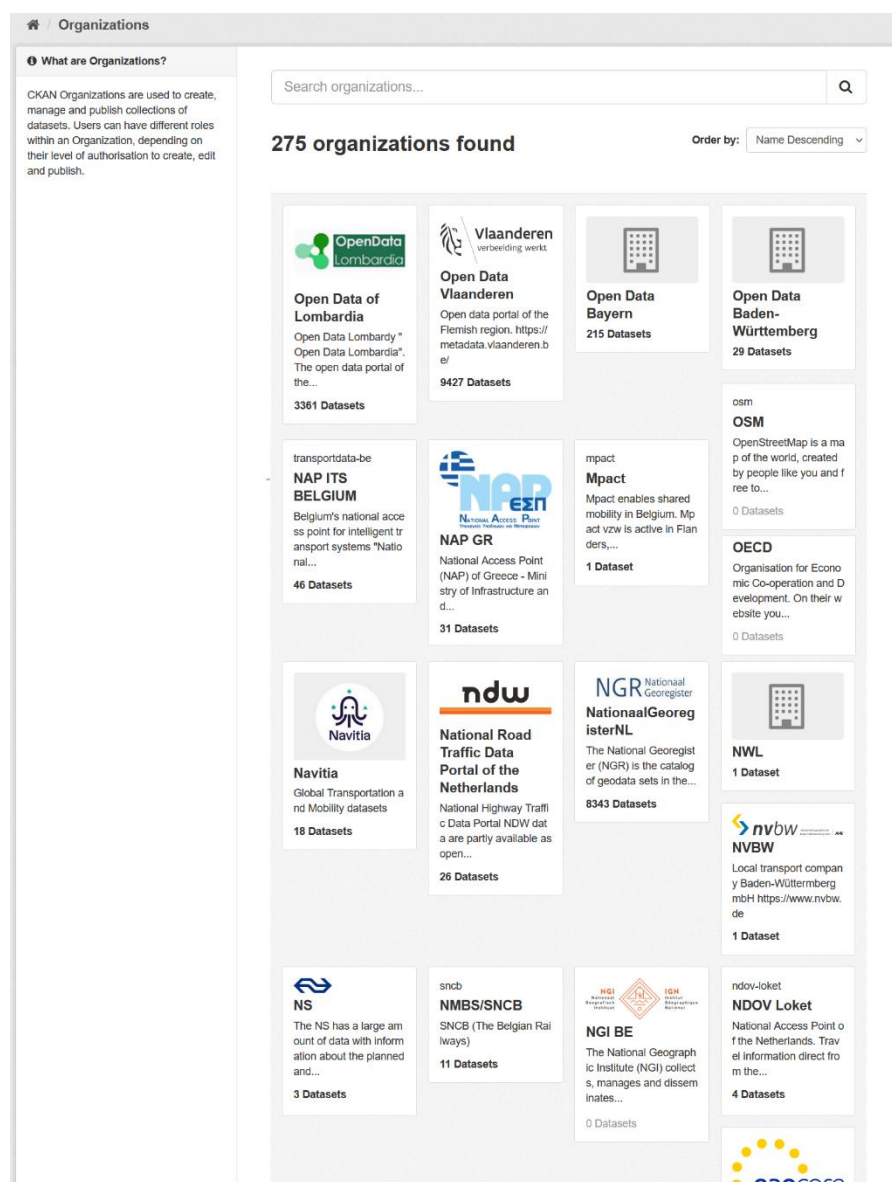
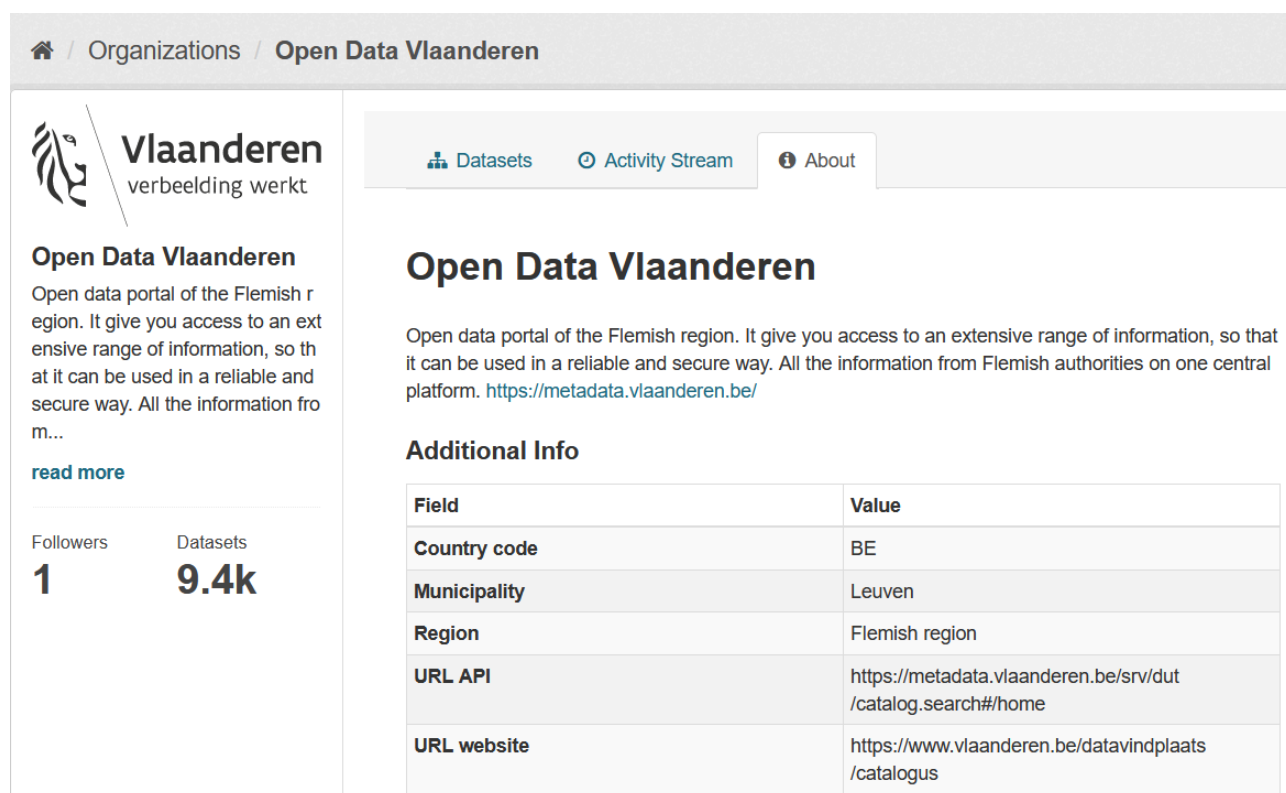


Figure 15: Organizations in the MobiDataLab CKAN instance

To create a new organization, the data publisher should log in to the CKAN portal, then go to **Organizations** and select **Add organization**. The data publisher enters its name (e.g., myCity) and a Uniform Resource Locator (URL) is automatically created.

In the context of the x-thons the organizations section is very important as it not only helps to find metadata, but also allows the users to learn more about the multiple organizations involved in the mobility and transport sector (see Figure 15Figure 16). Since participants might not know the transport operators in each municipality, it is very important to provide a description and information about which organizations provide services and tools in certain municipalities, regions or even countries (some cover more than one and not always in the same geographical area).

In case the participants might want to know more about the organization or want to consult their Application Programming Interface (API) documentation (that often changes from one organization to another), their website and the link to their API documentation and endpoint<sup>4</sup> is also often given (see Figure 16). Any additional fields can be easily added with a “key” and “value”.



**Open Data Vlaanderen**  
verbeelding werkt

**Open Data Vlaanderen**  
Open data portal of the Flemish region. It give you access to an extensive range of information, so that it can be used in a reliable and secure way. All the information from m...

[read more](#)

Followers **1** Datasets **9.4k**

**Open Data Vlaanderen**

Open data portal of the Flemish region. It give you access to an extensive range of information, so that it can be used in a reliable and secure way. All the information from Flemish authorities on one central platform. <https://metadata.vlaanderen.be/>

**Additional Info**

Field	Value
Country code	BE
Municipality	Leuven
Region	Flemish region
URL API	<a href="https://metadata.vlaanderen.be/srv/dut/catalog.search#/home">https://metadata.vlaanderen.be/srv/dut/catalog.search#/home</a>
URL website	<a href="https://www.vlaanderen.be/datavindplaats/catalogus">https://www.vlaanderen.be/datavindplaats/catalogus</a>

Figure 16: Organization information

With the idea of organizing better the datasets sources and making clearer the association and administration levels of data catalogues, a parent-son relationship function was explored and tested in our test VM of CKAN.

The idea was to create “sub-organizations” and apply this also for “sub-groups”. However, this functionality was not retained in our official CKAN data catalogue as this one was not flexible enough to display the organizations attractively (in the main Organizations page as in Figure 15 versus Figure 17) and it was not possible to assign more than one parent to an organization.

<sup>4</sup> “An endpoint is the point of entry in a communication channel when two systems are interacting.” (<https://rapidapi.com/blog/api-glossary/endpoint/>)

- Agenzia per l'Italia Digitale
  - CCISS Italy
    - Ministry of the Infrastructure and Sustainable Mobility of Italy
  - Lazio Region Open data
    - Citta Metropolitana di Roma Capitale
      - Roma Capital Open Data
        - ATAC
          - Roma Servizi per la Mobilità
  - Open Data of Lombardia
    - Open-Data-Milano
- AirParif

Figure 17: Parent-son tree structure display on CKAN (organizations page level)

The screenshot shows the MOBIDATALAB CKAN interface. The header includes the MOBIDATALAB logo and navigation links for Datasets and Organizations. The breadcrumb trail indicates the current location: Organizations / Lazio Region Open data. The main content area displays the organization's name, a brief description, and a 'read more' link. A parent-son tree structure is shown, listing the organization and its sub-organizations: Citta Metropolitana di Roma Capitale, Roma Capital Open Data, and ATAC. The ATAC sub-organization is further detailed with 'Roma Servizi per la Mobilità'. On the right side, there are tabs for Datasets, Activity Stream, and About. A search bar is present, and a checkbox for 'Include Sub-Organizations' is checked. The results show '394 datasets found' and a specific dataset titled 'Agrometeo historical series'.

Figure 18: Parent-son tree structure display on CKAN (datasets page level)

### 3.4.3. Managing datasets

In CKAN data is published in units called “datasets”. A dataset is a parcel of data, for example, the representation of public transport in each municipality.

To upload a new dataset in CKAN the data publisher needs to log into the CKAN portal, go to the **Datasets** page and click on **Add Dataset**. First, the data publisher should enter a descriptive title (e.g., “bus stations”) and a description. Some helpful tags such as “bus stops” or “bus stations”, separated with commas, could also be provided. A URL is then created automatically. The data publisher should choose the right license for the dataset and provide a source URL showing where the data comes from. As with organizations, custom fields can be added (see Figure 19). Then it is necessary to click on **Next: Add data**. The second step is to add the **Resource(s)** of the dataset (see Figure 20). On this second page, it is possible to choose either a file from the local computer or provide a link to an external resource. The name, description and format of the resource can be added next. To end uploading the dataset, click on **Finish**.

Datasets
Create Dataset

### What are datasets?

A CKAN Dataset is a collection of data resources (such as files), together with a description and other information, at a fixed URL. Datasets are what users see when searching for data.

1 Create dataset
2 Add data

**Title:**

**\* URL:**

**Description:**

demo description

You can use Markdown formatting here

**Tags:**

**License:**

Creative Commons Non-Commercial (Any)

License definitions and additional information can be found at <https://creativecommons.org/licenses/by-nc/4.0/>

**Organization:**

**Visibility:**

**Source:**

**Version:**

**Author:**

**Author Email:**

**Maintainer:**

**Maintainer Email:**

**Custom Field:**

Key: Country code

Value: BE, FR, ES, AU, JP, LT, LU, NO, SE, SI

**Custom Field:**

Key: Municipality

Value: Leuven

This data license you select above only applies to the contents of any resource files that you add to this dataset. By submitting this form, you agree to release the metadata values that you enter into the form under the [Open Database License](#).

\* Required field

[Next: Add Data](#)

Figure 19: Create dataset

**What's a resource?**

A resource can be any file or link to a file containing useful data.

**1 Create dataset** **2 Add data**

**Data:**

**Name:**

**Description:**

You can use [Markdown formatting](#) here

**Format:**

This will be guessed automatically. Leave blank if you wish

Figure 20: Add Resource(s)

To add, edit or delete a dataset or a resource that you have created, owned by an organization that you are a member of (if a dataset is not owned by any organization, then any registered user can edit it), you can click on Manage to do the necessary changes (see Figure 21).

Figure 21: Edit or delete dataset

### 3.4.4. Managing groups

Another way of organizing data in CKAN is according to groups, which are mostly used to categorize thematically related datasets. Groups are useful for organizing, managing, searching and finding more easily datasets. A dataset might belong to several groups. For way of organizing data in CKAN is according to groups, which are mostly used to categorize thematically related datasets.

For example, some typical group names might be transport, tourism, environment, or infrastructure.



In the MobiDataLab reference data catalogue, we choose to define groups in relation to use cases challenges of the MobiDataLab x-thons, but also in relation to the identified transportation themes and categories of the work done by DATA4PT<sup>5</sup> and the INSPIRE Directive (Directive 2007/2/EC).

Although there were many more categories covered by other organizations, it was decided to create only 25 different groups since this is the amount that can be easily discovered through the QGIS<sup>6</sup> extension of CKAN (deliverable 4.6 covers Data Access with this tool). Moreover, since many datasets could fall under several groups more, having more groups created could have created a difficulty for the user when having to decide on which group to search for a particular dataset (see Figure 22)Figure 22: Groups created in CKAN.

## 25 groups found

Order by: Name Ascending ▾

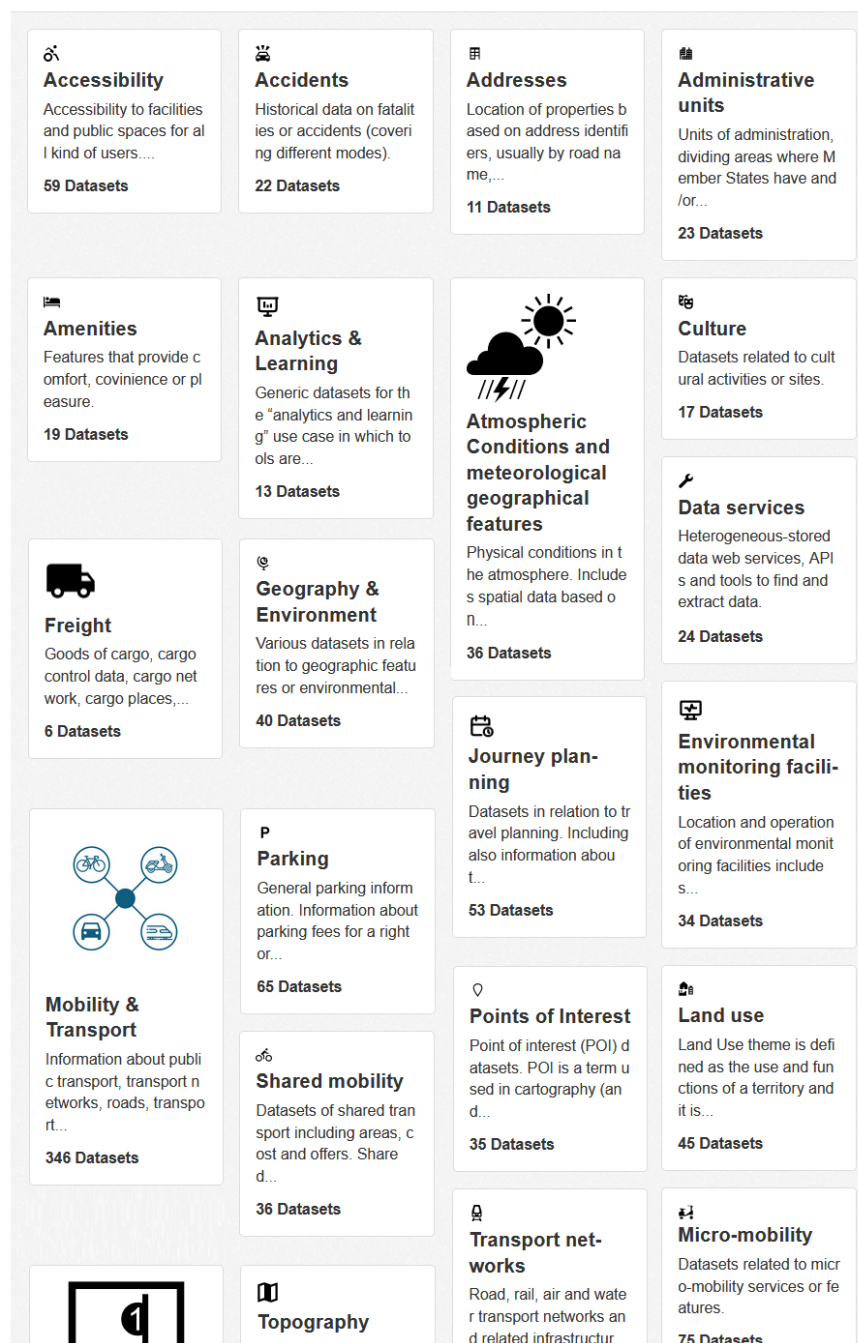


Figure 22: Groups created in CKAN

<sup>5</sup> <https://data4pt-project.eu/siri-webinar-material/>

<sup>6</sup> <https://www.qgis.org>



To add a group is simple, the data publisher needs to log into the CKAN portal and go to **Groups** page. Then click on **Add Group**, add the name and the description of the group and save it (Figure 23).

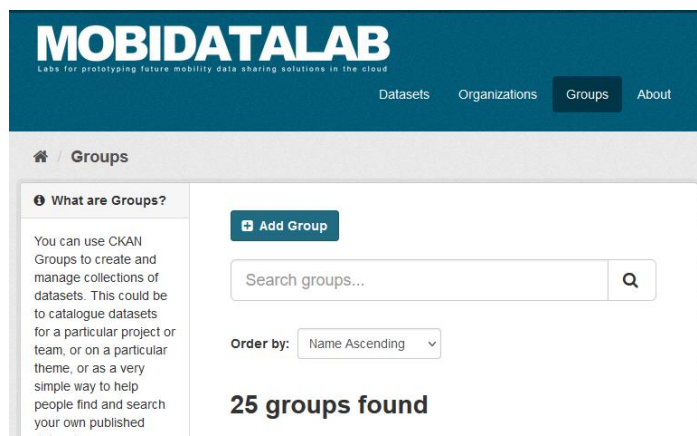


Figure 23: Add a dataset into a Group

To add a dataset manually into a group, select a dataset, click on the **Groups** tab, select a group from the menu and click on **Add to Group** (Figure 24).

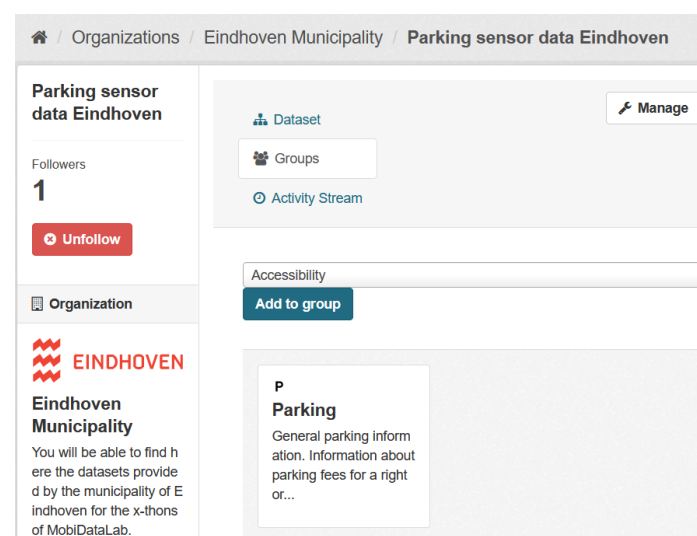


Figure 24: Create a Group

### 3.4.5. Data and metadata storage

Information can be added manually on CKAN, but when the quantity of information is important and the action is very repetitive, the best is to prepare a file or create a DataFrame<sup>7</sup> with all the required information to add or modify the content of the resources.

To achieve this with CKAN with the help of a text file, such as a Comma-Separated Values (CSV), it is essential to enable the FileStore.

<sup>7</sup> "A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns." [https://www.w3schools.com/python/pandas/pandas\\_dataframes.asp](https://www.w3schools.com/python/pandas/pandas_dataframes.asp)

### 3.4.5.1. FileStore API

FileStore is already included on CKAN, but it had to be activated. Files can be uploaded to the FileStore by using the `resource_create()` and `resource_update()` action API functions. To learn more about these functions, see the Annex FileStore API functions.

### 3.4.5.2. DataStore

Besides metadata, CKAN can also store data that is hosted on other platforms around the web. The DataStore extension provides a database for the storage of structured data from CKAN resources. Data can be extracted from resource files and stored in the DataStore, this one then enables data previews and a data API for the resources. The DataStore is distinct but complementary to the FileStore (see FileStore and file uploads<sup>8</sup>). In contrast to the FileStore which provides ‘blob’ storage of whole files with no way to access or query parts of that file, the DataStore is like a database in which individual data elements are accessible and queryable. To illustrate this distinction, consider storing a spreadsheet file like a CSV or Excel document. In the FileStore this file would be stored directly. To access it you would download the file as a whole. By contrast, if the spreadsheet data is stored in the DataStore, one would be able to access individual spreadsheet rows via a simple web API, as well as be able to make queries over the spreadsheet contents. DataStore is bundled within CKAN and it can be enabled as an extension on the CKAN configuration file.

### 3.4.5.3. DataPusher installation

Often, one wants data that is added to CKAN (whether it is linked to or uploaded to the FileStore) to be automatically added to the DataStore. This requires some processing, to extract the data from your files and to add it to the DataStore in the format the DataStore can handle.

This task of automatically parsing and then adding data to the DataStore is performed by the DataPusher (see Figure 25), a service that runs asynchronously and can be installed alongside CKAN. You can find the step-by-step guide of the DataPusher extension on GitHub: <https://github.com/ckan/datapusher>.

<sup>8</sup> <https://docs.ckan.org/en/2.9/maintaining/filestore.html>

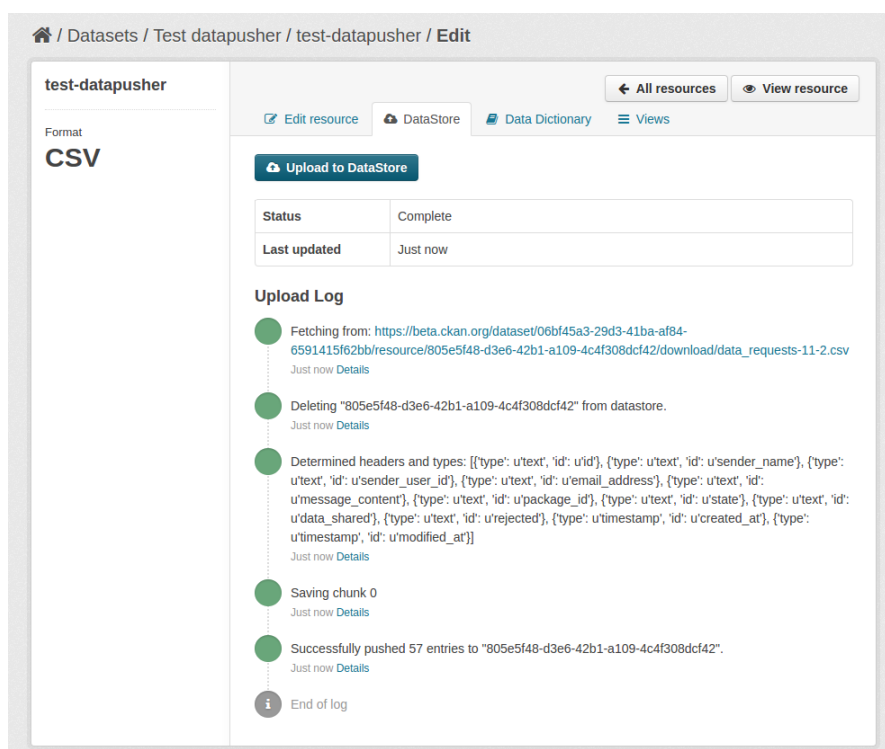


Figure 25: Data pusher test

### 3.4.6. Visualising data

The CKAN resource page can contain one or more visualizations of the resource data or file contents (a table, a bar chart, a map, etc.). These are commonly referred to as resource views.

Users who are allowed to edit a particular dataset can also manage the views for its resources. To access the management interface, the data publisher should click on the **Manage** button on the resource page and then on the **Views** tab. From here it is possible to create new views, update, or delete existing ones, and reorder them (see Figure 26).

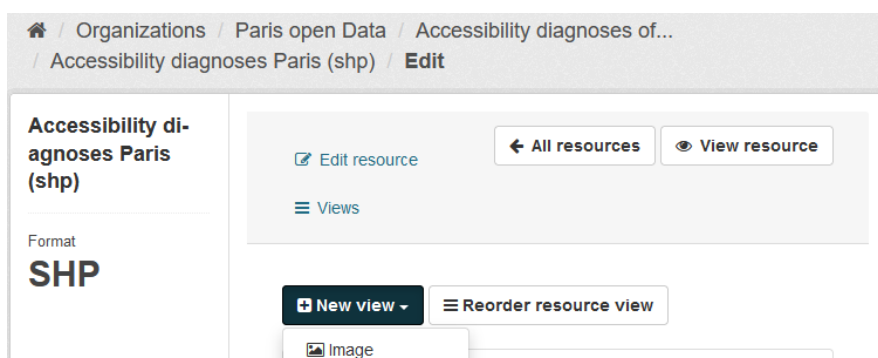


Figure 26: Managing views in a CKAN resource page

The “New view” dropdown will show the available view types for the specific resource format. If the list is empty, it might be necessary to add the relevant view plugins to the `ckan.plugins` settings on your configuration file (e.g. grid view, map view, etc.).

When a resource is added to the DataStore, the user can get automatic data previews on the resource page using the Data Explorer extension (available until version 2.9, in the newest version it has been replaced by new plugins such as DataTables view and other view plugins). An example of a dataset added to the DataStore is Figure 27.

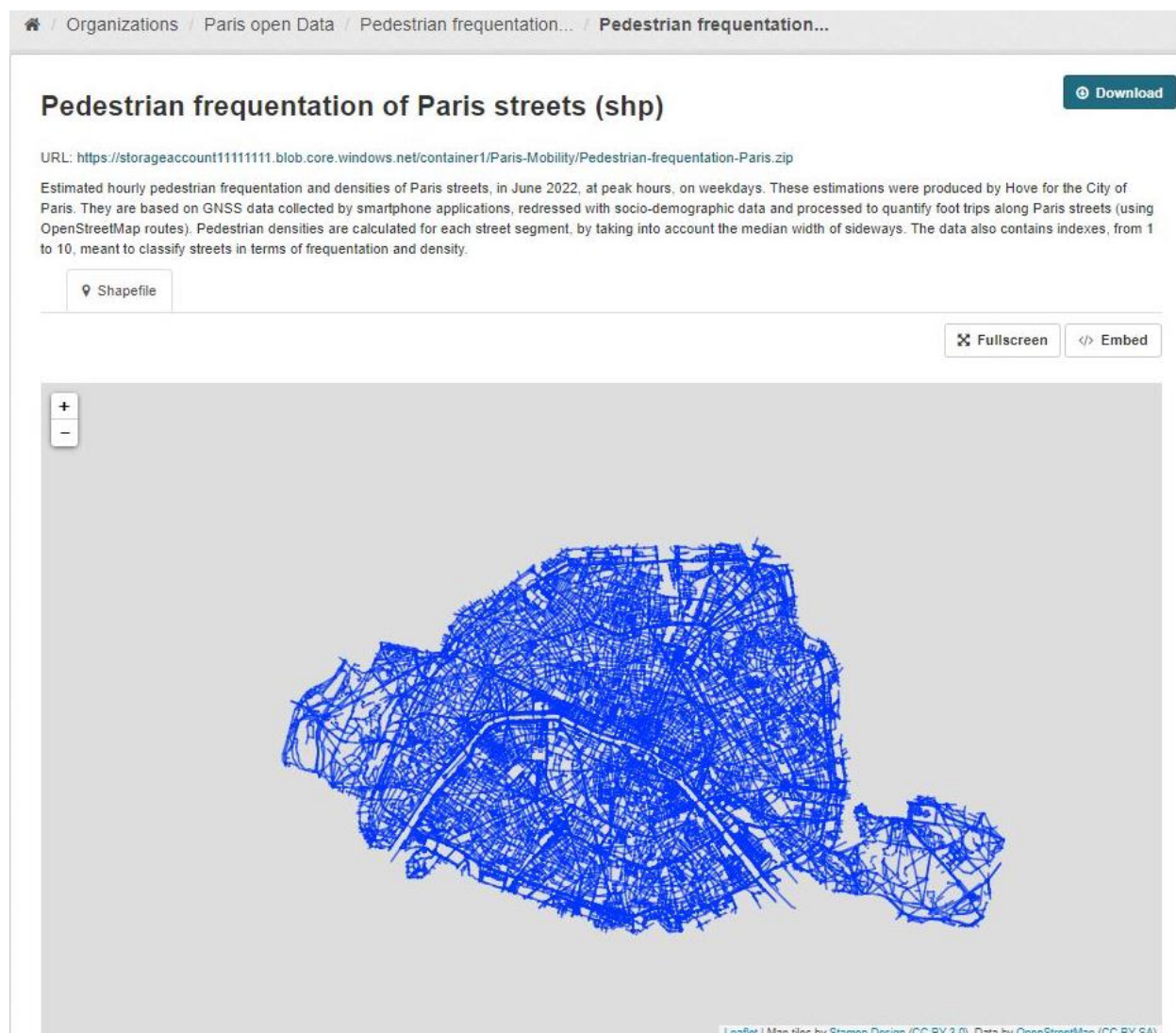


Figure 27: Dataset added to the DataStore preview

### 3.4.7. Geospatial search extension

The spatial search and the Catalogue Web Service (CWS) support plugins were installed.

This was done with the purpose to search and visualize indexed datasets based on their spatial information. However, due to problems encountered this one was deactivated.

### 3.4.8. Multi-lingual management

In the MobiDataLab project, the reference data catalogue aims to be used in the context of international living and virtual labs. Therefore, the metadata needs to be made available, besides the original language, at least in English. Data publishers are advised to provide human-readable metadata in multiple languages and, where possible, to provide the information in the language(s) that the intended users will understand. Nevertheless, most data portals are not multilingual and the native language of most of the countries in the reference group is not English. In the case of the NAPs of Belgium ([Data.gov.be](https://data.gov.be)) and Switzerland ([open.data.swiss](https://open.data.swiss)), several datasets were available in different local languages or English, but it seems that the dataset description was entered from the beginning in these languages. Some might display the general metadata keywords in English, but the description and the content will be provided in the local language (for instance [EINDHOVEN OPEN DATA](#) or [PLATEFORME OPEN DATA DE L'INSTITUT PARIS REGION](#)).

The Multilingual extension of CKAN allows the administrator to enter translations into CKAN in the same way as in the examples, but the user will be able to choose from a list of languages (see Figure 28). When a user is viewing the CKAN interface if the translation terms database holds a translation in the user's language for the name or description of a dataset or resource, such as the name, tag, group, etc. Then, the translated term will be shown to the user instead of the original term thanks to the Graphic User Interphase (GUI).

The figure displays two side-by-side screenshots of the CKAN interface, illustrating the multi-lingual management feature. Both screenshots show the metadata for a dataset titled 'shapefile-of-the-lines...'. The left screenshot shows the interface in Spanish, with the 'Idioma' dropdown set to 'español'. The right screenshot shows the interface in English, with the 'Language' dropdown set to 'English'. The metadata fields are identical in both, including 'Identificador', 'Idioma', 'Modificado', 'Nombre del publicador', 'Publicado', 'Tema', 'URI', and 'harvest\_source\_title'. The 'Tema' field contains a URL with a query parameter 'refine.theme=Description+du+%C3%A9seau'. The 'URI' field contains a URL with a query parameter 'refine.publisher=SNCF+R+%C3%A9seau'.

Figure 28: Multi-lingual management

This multilingual support does not include the full translation of terms describing the metadata since this one has to be done during the harvesting (data import) process on the database.



This is the reason why we decided to add a client-based translation with Google, so the content can be visualized by the user in English. This will be discussed further in the harvesting configuration section. Moreover, the multilingual extension had to be disabled because it was not compatible with Solr, but also because we didn't use it since what we require is a manual translation.

### 3.4.9. Harvesting

The harvesting extension functionality was implemented into the MobiDataLab CKAN, to add datasets efficiently. This extension allows to harvest metadata from other CKAN catalogues and integrate them into the CKAN portal. Most of the data in the reference data catalogue is harvested from different data sources, but mainly from governmental catalogues. The MobiDataLab data portfolio can harvest them without too much effort from either side.

This tool can be configured so that only the necessary data from groups and organizations are harvested. New developments were also made by our team to include a filter by tag (see filtering configuration). However, this configuration only works when harvesting from another CKAN catalogue.

The basic harvesting extension allows only to harvest datasets from other CKAN data catalogues. To harvest datasets from other portals, the DCAT extension, based on the Resource Description Framework (RDF), and the Catalogue Web Service harvesters were installed. The RDF vocabulary allows the consumption and aggregation of metadata in a simpler way from catalogues that use this same standard.

Two Catalogue Services for the Web harvesting extensions were installed as well. The first one is a regular CSW which allows to publish, modify and search collections of descriptive information (metadata records) of data, services, and related information objects. This service supports the discovery of registered information resources, and it is a widespread standard in geographic data catalogues. The second CSW harvesting extension installed was the Inspire version. This last allows monitoring, access and discovery of datasets and/or services from EU Member States and European Free Trade Association (EFTA) countries according to themes or priority datasets. These extensions provide support for the OGC standard and the ability to harvest records from other CWS servers.

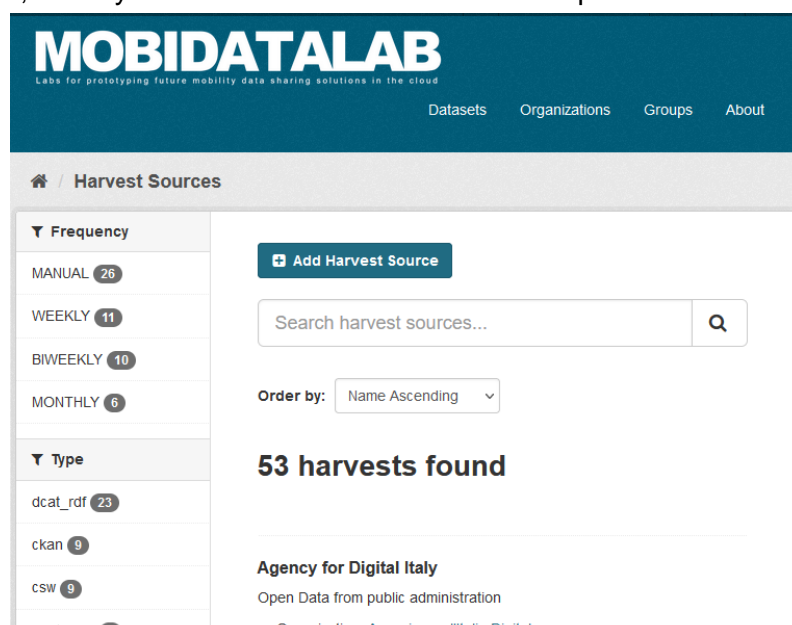


Figure 29: Add a harvest source

They are also attractive extensions for the project because within the scope of the INSPIRE Directive, there is a particular theme to categorize and discover Transport Networks datasets.

To harvest datasets from another catalogue, the information about the organization should have been registered first by an organization admin. Then, to access the menu for creating new harvesting jobs, it is necessary to add on the search bar “/harvest” after the URL of the catalogue’s main page (<https://ckan.mobidatalab.eu/>). Once there, a connected admin can add a harvesting source (see Figure 29).

To add the source, the end-point URL and the title of the source must be entered. The source type and the update frequency must be selected among the available options. The description and the configuration are optional. Here is an example of how the harvesting process looks like (we choose to harvest the data from Infrabel):

The screenshot shows the 'Harvest Sources' page in the Infrabel Open Data portal. The page has a sidebar on the left with the portal's name and a dataset count of 89. The main content area contains a form for creating a new harvesting source. The form includes fields for URL, Title, and Description, and a section for selecting the Source type. The URL field contains 'https://infrabel.opendatasoft.com/api/v2/catalog/exports/dcat?lang=en'. The Title field contains 'Infrabel Open Data portal'. The Description field contains 'Infrabel's Open Data portal. In this portal can be found datasets from the following themes : infrastructure, traffic control, human resources, security, clients and products, finance and environment'. The Source type section has several radio button options, with 'Generic DCAT RDF Harvester' selected.

Figure 30: Harvest source creation page

The update frequency of harvesting can be set to daily, biweekly, weekly, monthly or always (to constantly communicate with the data sources). The harvesting module can be configured to be manually triggered as well (see Figure 31).

Update frequency: Biweekly

Configuration:

```
{
  "default_tags": [{"name": "transport"}],
  "default_groups": ["Mobility & Transport"],
  "default_extras": {"encoding": "utf8"},
  "remote_groups": "only_local",
  "remote_orgs": "create"
}
```

Organization: infrabel

Delete Save

Figure 31: Harvesting configuration

The active harvesting sources were reduced in number in comparison to the first tests, to ensure the maximum updates and number of datasets harvested and translated for the reference group municipalities who proposed challenges for the virtual and living labs. Since the translation of datasets takes a lot of time and there are limits in terms of how many datasets can be translated per day.

This reduction of sources was also necessary since some municipalities do not necessarily have municipal-level data portals or they did not cover all the categories of the use's cases, which means that it was

necessary to harvest larger and several data portals to obtain as many possible datasets of a certain municipality. This was the case of the municipality of Leuven for which it was necessary to harvest the regional data portal, as well as the transport national access point portal. In the case of Eindhoven, they have a local open data portal, but extra datasets had to be searched in other data portals. Even, though the number of datasets harvested and translated could have been reduced in the case of some CKAN data portals through filters, this was not necessarily the case for all the data portals harvested, as the filters did not work for non-CKAN portals. Some data sets were harvested via the CSW and CKAN, while others with DCAT and DCAT-AP.

### 3.4.9.1. Configuration

The CKAN harvesters support several configuration options to control their behaviour. Those need to be defined as a JSON object in the configuration form field. Here is an example of a configuration object:

```
1 {
2   "api_version": 1,
3   "default_tags": [{"name": "mobility"}, {"name": "bus"}],
4   "default_groups": ["mobility", "transport"],
5   "override_extras": true,
6   "organizations_filter_include": [],
7   "organizations_filter_exclude": ["remote-organization"],
8   "read_only": true,
9   "remote_groups": "only_local",
10  "remote_orgs": "create"
11 }
```

Figure 32: Configuration box

The description of the objects in the configuration can be found in the Harvesting configuration annexe.

### 3.4.9.2. Filters

As mentioned earlier, on CKAN sources datasets can be filtered by adding certain parameters into the configuration of the harvester.



For instance, Groups or Organizations can be included or excluded during the harvesting process (see Figure 32). It is also possible to add default tags and groups to the datasets harvested by a particular organization. Our team encountered problems with the filtering function, so a multi-value query syntax fix was implemented into our CKAN and the issue was reported on the harvester's repository: <https://github.com/ckan/ckanext-harvest/issues/518>. Additionally, a tag filter was added into the configuration. The detailed information of the modification can be found on the [MobiDataLab GitHub repository](#).

### 3.4.10. Translation

As mentioned in the multi-langue section, to translate all the dataset's descriptions into a common language the translation must be done directly in the database. Therefore, it was necessary to use Google Translation API to translate the dataset's metadata while harvesting. Additionally, because we use many harvester extensions to support the harvesting of more data formats other than the default CKAN format, we had to implement the translation on each harvesting extension.

Unfortunately, the harvesting extensions were created by different developers using different approaches, thus the translation was implemented on the gathering stage for most of the harvesting extensions, but we had to implement it on the fetching stage for some other formats like DCAT/JSON.

To learn more about the translation's technical implementation and limitations, please consult annexe 7.7.

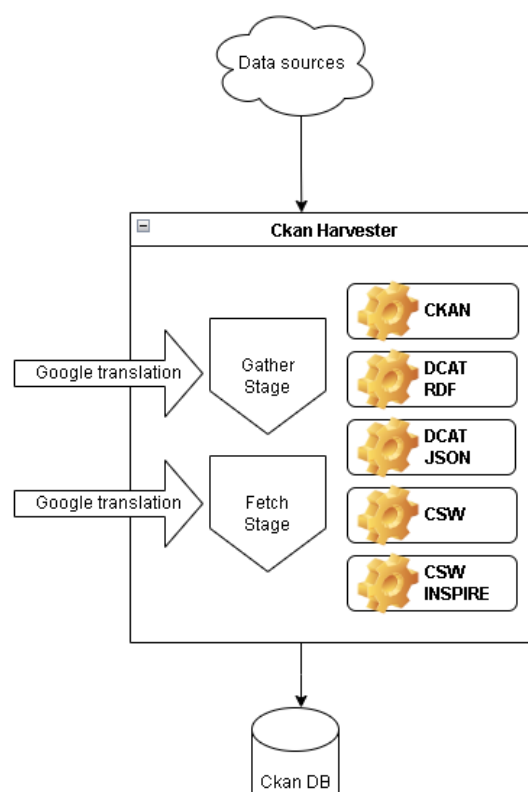


Figure 33: Translation diagram

### 3.4.11. CKAN API

The CKAN API enables developers to write code that interacts with CKAN sites and their data.

CKAN's Action API is a powerful, RPC-style API that exposes all of CKAN's core features to API clients. All the CKAN website's core functionalities can be used by external code that calls the CKAN API (meaning that everything that you can do with the web interface and more can be also done with the API). For example, by using the CKAN API your app can:

- Get JSON-formatted lists of a site's datasets, groups or other CKAN objects:
- [https://ckan.mobidatalab.eu/api/3/action/package\\_list](https://ckan.mobidatalab.eu/api/3/action/package_list)
- [https://ckan.mobidatalab.eu/api/3/action/group\\_list](https://ckan.mobidatalab.eu/api/3/action/group_list)
- [https://ckan.mobidatalab.eu/api/3/action/tag\\_list](https://ckan.mobidatalab.eu/api/3/action/tag_list)
- Get a full JSON representation of a dataset, resource or other object:
- [https://ckan.mobidatalab.eu/api/3/action/package\\_show?id=traffic-events](https://ckan.mobidatalab.eu/api/3/action/package_show?id=traffic-events)
- [https://ckan.mobidatalab.eu/api/3/action/tag\\_show?id=transport](https://ckan.mobidatalab.eu/api/3/action/tag_show?id=transport)
- [https://ckan.mobidatalab.eu/api/3/action/group\\_show?id=accessibility](https://ckan.mobidatalab.eu/api/3/action/group_show?id=accessibility)
- Search for packages or resources matching a query:
- [https://ckan.mobidatalab.eu/api/3/action/package\\_search?q=bike](https://ckan.mobidatalab.eu/api/3/action/package_search?q=bike)
- [https://ckan.mobidatalab.eu/api/3/action/resource\\_search?query=name:transport](https://ckan.mobidatalab.eu/api/3/action/resource_search?query=name:transport)
- Create, update and delete datasets, resources and other objects
- [https://ckan.mobidatalab.eu/api/3/action/package\\_create](https://ckan.mobidatalab.eu/api/3/action/package_create)
- Get an activity stream of recently changed datasets on a site:
- [https://ckan.mobidatalab.eu/api/3/action/recently\\_changed\\_packages\\_activity\\_list](https://ckan.mobidatalab.eu/api/3/action/recently_changed_packages_activity_list)

More details about the full usage of CKAN API can be found directly on the official website of CKAN: <https://docs.ckan.org/en/2.10/api/index.html>

#### 3.4.11.1. CKAN API clients

To help developers use CKAN API on different environments and coding languages this information is available in the annex List of available CKAN API clients.

### 3.4.11.2. CKAN API authentication

Some API functions require authorization. The API uses the same authorization functions and configuration as the web interface, so if a user is authorized to do something in the web interface, they'll be authorized to do it via the API as well.

When calling an API function that requires authorization, you must authenticate yourself by providing an authentication key with your Hypertext Transfer Protocol (HTTP) request.

API authentication tokens can be generated through the CKAN's web interface (see Figure 34).

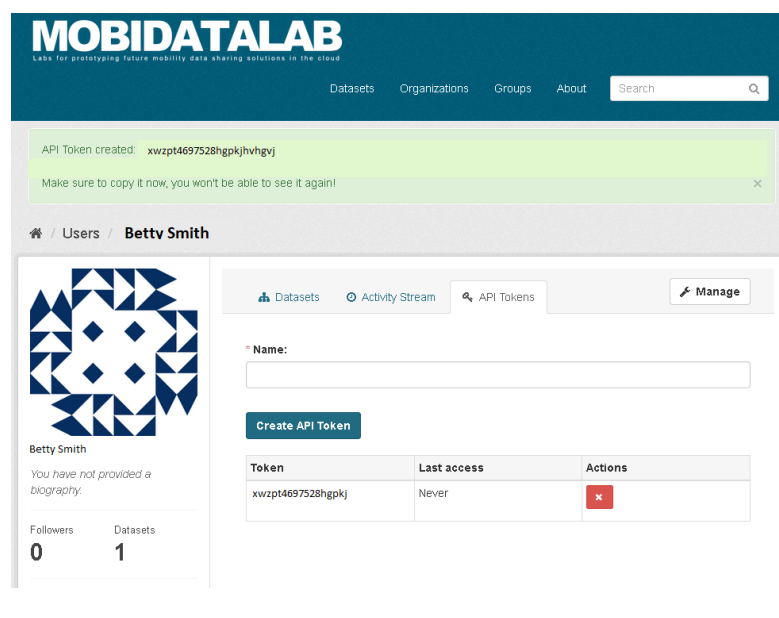


Figure 34: API token generation

To provide your API token in an HTTP request, include it in either an Authorization or X-CKAN-API-Key header:

```
curl -H "Authorization: XXX" https://ckan.mobidatalab.eu/api/3/action/am_following_user?id=bettysm
```

### 3.4.11.3. CKAN Harvester API

The API has multiple APIs exposed in the format /api/action/<endpoint>, it allows one to: access CKAN's harvest logs and harvest source list, create a harvest source and harvest jobs and get documentation of the API. To see some examples, consult the annex CKAN Harvester API.

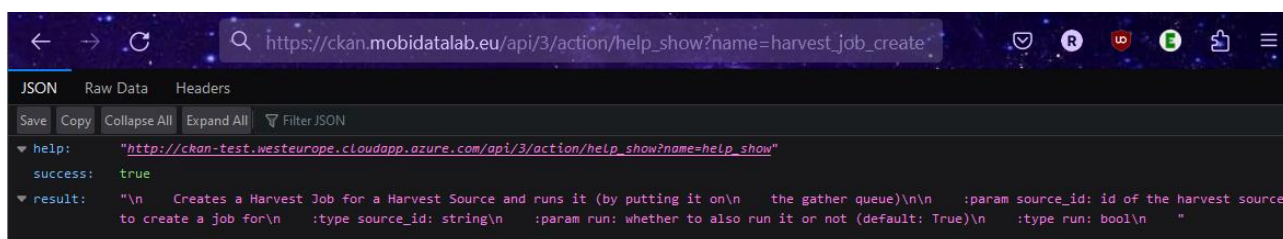


Figure 35: CKAN Harvester API endpoint

## 3.5. Difficulties

Difficulties on the platform construction have to be segregated into infrastructural constraints (cloud safety/security, resource type and size, cloud subscription allowance, topology applying interoperability with partners, etc. ), applicative constraints (application ecosystem, languages, software packaging, project status (active /inactive)), production constraints, and sociology of the user and the industry standards fulfilled by the application along with the features the application brings to the platform and the platform users.

### 3.5.1. *Incompatibility CKAN-Solr*

CKAN/Solr<sup>9</sup> version compatibility has been a major issue within the CKAN ecosystem when debugging search functionalities. These were irresponsive due to the Solr versioning. This same incompatibility did not arise when fetching metadata without keywords such as getting the whole list of metadata, metadata by groups/categories, etc. To avoid problems with the dataset search, the search issue was fixed by downgrading an acceptable version of Solr for CKAN. Additionally, this update was necessary to avoid security problems (Log4J vulnerability<sup>10</sup>).

### 3.5.2. *Needs and difficulties encountered while harvesting CKAN datasets*

#### 3.5.2.1. Harvesting limitations

One of the limitations we got while trying to use multiple data source formats was the ability to harvest multiple datasets having the same name from different sources. CKAN does not allow the creation of a dataset with a name “title” that already exists on the database; thus, it refuses to harvest datasets having the same title because the URL is generated based on this title and it should be unique.

This is a design issue that should be handled natively on CKAN. Therefore, the two issues have been already reported and are still open on CKAN's GitHub repository:

- [Factory of unique \(and available\) names for entities](#)
- [Adding a suffix to the package name if exists another with the same name](#)

<sup>9</sup> “CKAN uses Apache Solr as its search engine. For further details check the Solr documentation. Please note that CKAN sometimes uses different values than what is mentioned in that documentation. Also note that not the whole functionality is offered through the simplified search interface in CKAN or it can differ due to extensions or local development in your CKAN instance.” (<https://docs.ckan.org/en/2.9/user-guide.html>)

<sup>10</sup> <https://logging.apache.org/log4j/2.x/security.html>

### 3.5.2.2. Scraping an alternative solution when harvesting is not possible

Apart from the unique naming issue, we turned to scraping because we encountered difficulties harvesting certain resources. Indeed, while CKAN hosts a vast collection of publicly available datasets, it is not always possible to harvest or retrieve all datasets from CKAN and other dataset sources for several reasons:

1. **Permissions and Licensing:** Some datasets hosted on CKAN may have specific usage restrictions or licensing agreements that prevent automated harvesting. The dataset owners or publishers might require users to agree to specific terms and conditions before accessing the data.
2. **API and Rate Limiting:** CKAN may provide an API (Application Programming Interface) to access datasets programmatically. However, APIs often have rate-limiting measures in place to prevent excessive requests from overloading their servers. Harvesting too many datasets in a short period might trigger rate limiting, restricting further access for a certain period.
3. **Data Size and Complexity:** Some datasets on CKAN can be large and complex, making it challenging to download or process them all in one go. Harvesting large datasets might consume significant bandwidth, time, and computational resources.
4. **Selective Data Publishing:** Not all datasets on CKAN may be available for public harvesting. Dataset publishers have control over whether their datasets are publicly accessible or restricted to certain users or organizations.
5. **Technical Constraints:** Automated harvesting requires specialized tools and scripts to navigate through CKAN's interface, find datasets, and download data. Certain websites with downloadable datasets are not configured as data catalogues or they do not offer a standardized endpoint for sharing metadata. The availability and compatibility of such tools might vary, affecting the ability to harvest certain datasets or all their content.

### 3.5.2.3. Scraping definition and terms of use

Web scraping is the process of extracting data or information from websites. It involves using automated software, commonly referred to as web scrapers or web crawlers, to retrieve data from web pages, and then save, process or analyse that data for various purposes.

Web scraping is commonly employed to gather large amounts of data quickly and efficiently. It allows individuals and organizations to access and utilize data from websites without having to manually visit each page and copy the information manually, which would be time-consuming and impractical for vast amounts of data.

The steps involved in scraping a web page are roughly as follows:

1. **Identify the Target Data and understand the Website's structure:** the first step is to determine the specific data to be extracted and understand the structure of the website. For that, we inspect the website's HTML source code by using the browser's developer tools to inspect the elements, class names, and IDs that will help us locate the desired information.
2. **Requesting data:** after that, a web scraper sends HTTP requests to the target website's servers, requesting specific web pages' content
3. **Retrieving HTML content:** upon receiving the request, the website's servers respond by sending back the HTML content of the requested web pages
4. **Parsing the data:** the web scraper uses tools like BeautifulSoup or lxml<sup>11</sup> to parse the HTML, extracting the relevant data elements such as text, images, URLs, or specific data points
5. **Data processing:** once the data is extracted, the scraper can perform various operations on it, such as cleaning, filtering, or transforming the data into a desired format
6. **Storing the data:** the scraped data can be saved into a structured format like CSV, JSON, or a database for further analysis or use

Note that if the data we want to extract is spread across multiple pages, we will need to implement a mechanism to navigate through the pagination and scrape data from each page iteratively. We may also need to monitor our script's performance regularly and update it as needed to ensure continued functionality.

### 3.5.2.4. Scraping with Python - usage example

To carry out this task, Python programming language was used. Python is a popular choice for web scraping due to its ease of use, a rich ecosystem of libraries, and its versatility in handling different aspects of web scraping. It provides several libraries that are commonly used for web scraping, some of which include:

- **requests:** allows Python programs to send HTTP requests to websites and receive responses. It is used to fetch the HTML content of web pages
- **BeautifulSoup:** helps parse the HTML content obtained from the websites, allowing users to extract specific data elements easily
- **Scrapy:** facilitates the crawling of websites, allowing for more complex scraping tasks and handling of data pipelines
- **Selenium:** used for automated web browsing and interaction with websites, enabling users to scrape websites that require user interaction, such as submitting forms or clicking buttons

In our case study, we mainly used the libraries **BeautifulSoup** and **requests**.

---

<sup>11</sup> <https://lxml.de/>

### 3.5.2.5. Limits

Web Scraping can be a complex and delicate process. Indeed, scraping websites without permission or in violation of their terms of service may be illegal and unethical. It's essential to review the terms of use of each website and obtain permission before attempting any web scraping activities. Some websites may also have measures in place to prevent scraping, such as CAPTCHAs or IP blocking. We also had to ensure that we were not overwhelming the website's servers with too many requests and follow any rate-limiting guidelines provided by the website.

Regarding the possibility of coding a generic script for scraping various web pages; while it's possible to create a generalized web scraping script that can be used as a starting point for scraping multiple websites, it's unlikely that a single script can successfully scrape all websites. Indeed, web scraping is highly dependent on the structure and layout of individual websites, which can vary significantly. Websites have different HTML structures, and a web scraping script needs to be tailored to extract data from specific elements on each website. For example, the CSS selectors used to locate and extract data will likely be different across websites. Furthermore, some websites load data dynamically using JavaScript, making it more challenging to scrape with a basic script that does not handle dynamic content.

So, to scrape multiple websites, it's essential to customize the script for each target site and consider the specific challenges posed by each one. We might need to create separate modules or scripts for different categories of websites, and we'll likely need to maintain and update the scripts regularly as websites change their structures or implement new features.

### 3.5.3. *Cybersecurity*

In terms of dataset import security, it is considered that data sources are public or territorial entities that are preoccupied with the territory's reputation and will certainly not publish harmful content such as a security or the technological threat. Nevertheless, malformed files can appear from data sources due to content mishandling or technical inexperience of data manipulators. The huge volume of imported datasets hardly allows to verify every dataset imported.

### 3.5.4. *Documentation/support*

Open-source technologies come with an extensive community of ideas and developers that will keep the project alive and up to date, as long as there is feedback from the users and a dynamic environment surrounding the project. This dynamic can easily be verified on main-stream software and even hardware when the user base is outstanding and is followed by new ideas and expectations that will generate comments which will in return call for Documentation/support to structure the project visibility. In the case of a niche product, the software editor will still supply documentation on the official product website by focusing essentially on the last release of the product.



The earlier version will to some extent be neglected but it will remain in production with a conservative lifecycle that will favour a working environment rather than an up-to-date release that is full of uncertainties.

As a result, product maturity is always an issue for a niche product and that maturity impacts third-party extensions that are not guaranteed to give identical results on the new version. To that odd behaviour, it is added that at the operating system level, there is no longer a supported version of Python 2.x which is the native language used to develop CKAN.

### 3.5.4.1. Debug

We implemented a Debug on CKAN to have additional information displayed on the CKAN pages and be able to debug them.

In some cases, CKAN crashes before it can load the Hypertext Markup Language (HTML) templates, thus the debug information is not loaded, and you get only a "DON'T PANIC" error stack.

The best solution in this case would be to set up a debugging environment with a Python IDE, but it will not be an easy task with all the required dependencies, [check here](#) for more details.

In this case, it is necessary to analyse the stack and try to understand the crash based on source code, and maybe add some additional information to the raised exception so you can figure out what is wrong (see Figure 36).

#### ValueError

```
ValueError: Invalid URL! args: ('dataset_resource.read',), kw: {'resource_id': '18b0480e-5e7d-4a7d-902f-30cc76ad4f9f', 'bom': True}
```

#### Traceback (most recent call last)

```
File "/usr/lib/ckan/default/src/ckan/ckan/lib/helpers.py", line 365, in url_for
    my_url = _url_for_flask(*args, **kw)
File "/usr/lib/ckan/default/src/ckan/ckan/lib/helpers.py", line 432, in _url_for_flask
    my_url = _flask_default_url_for(*args, **kw)
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/helpers.py", line 370, in url_for
    return appctx.app.handle_url_build_error(error, endpoint, values)
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/app.py", line 2275, in handle_url_build_error
    reraise(exc_type, exc_value, tb)
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/_compat.py", line 39, in reraise
    raise value
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/helpers.py", line 357, in url_for
    rv = url_adapter.build(
File "/usr/lib/ckan/default/lib/python3.8/site-packages/werkzeug/routing.py", line 2179, in build
    raise BuildError(endpoint, values, method, self)
```

#### During handling of the above exception, another exception occurred:

```
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/app.py", line 2449, in wsgi_app
    response = self.handle_exception(e)
File "/usr/lib/ckan/default/lib/python3.8/site-packages/flask/app.py", line 1866, in handle_exception
    reraise(exc_type, exc_value, tb)
```

Figure 36: Debug



### 3.5.5. Interoperability

Interoperability between data servers and data consumers using a data browser or APIs (data services that harvest content and make that content available to visitors) are examples of a situation where the data platform has to offer visitors the protocols and technical standards/format with which the data consumer is used to work since APIs are aimed to answer request regardless the type of request issuer.

Interoperability also aims to offer the intercommunication software with which data consumers are expecting to operate through HTTP/internet requests. GIS-compatible software such as QGIS can be a client of the MobiDataLab data services after the installation in the GIS client of the module opening interactivity, through dedicated plugins, with CKAN and GeoNetwork (for more information about it, please consult deliverable 4.6 on Data Services).

Among the industry standards, the MobiDataLab platform is compatible with the following protocols:

- CKAN API

The CKAN API 3 exposes all the main CKAN functionalities to API clients. All the core functionality of a CKAN website (everything that can be done with the web interface and more) can be used by external code that calls the CKAN API. This is particularly useful for developers who want to write code that interacts with CKAN sites and their data.

- DCAT

The Data Catalog Vocabulary is an RDF vocabulary designed to facilitate interoperability between data catalogues published on the Web<sup>4</sup>. The CKAN DCAT extension allows CKAN to expose and consume metadata from other catalogues using RDF documents serialised using DCAT.

- DCAT-AP

The DCAT Application Profile for data portals in Europe is an extension of the DCAT RDF ontology with a cross-border and cross-domain scope whose goal is to meet specific requirements. It is a well-known interoperable specification as through semantic equivalents for each attribute and element of data, it allows systems to interact with each other and automatically convert that data.

- CSW

Catalogue Service for the Web is a specification from the Open Geospatial Consortium for exposing geospatial catalogues over the Web. The CKAN CSW extension provides support for the CSW standard, with the ability to import records from CSW.

### 3.5.6. Usability

Usability by all the data consumers in the European Union remains a challenge since EU countries speak different languages. The trade-off is technological as harvesting metadata in the native language and translating the data in real-time into the “consumer” or most spoken language is a solution that does not facilitate the search operation and the harvesting process. Translation takes a lot of time and resources, there is also a limit on the number of translated words per day. Moreover, some issues arise with translation, due to length restrictions in CKAN for tags (certain native languages that contain very long words, often produce errors. This type of error was mitigated by limiting or shortening the tag to translate. However, this might result in another issue that is a general issue with translation which is the possibility of mistranslation. Sometimes translation alters the display of resources as sometimes a word in 3/8 resources of the same dataset might be translated in a different way, which might be appreciated by the CKAN as a different dataset. This could further confuse the data consumer about the difference between the same resource in a different format.

The MobiDataLab is seen by some users as a data hub where storage is a pool from which data is to be fetched, others see MobiDataLab as a set of services or processors, data capable, aimed to digest data and help manipulate inserted content and evaluate the resulting output so much so, the Navitia journey planner is deployed with timetables from various transit system for which, it has been expected users to leverage that deployment in a read-only mode. Yet, the relevance of attendees willing to insert custom transit data needed to propose a Navitia deployment. That will be basically empty and will give the participant the freedom to bring the data of their choice to resolve the use case in which they are working.

## 4. GeoNetwork demonstration

GeoNetwork stands as a powerful and versatile geospatial metadata catalogue, empowering users with the ability to efficiently manage, explore, and visualize metadata for an extensive range of geospatial datasets. It serves as a central repository for geospatial information with a web-based interface, making it an essential component of the project's data management strategy.

GeoNetwork offers project members, data contributors, researchers, and stakeholders a seamless experience in discovering and accessing insightful geospatial information.

Within the project's landscape, GeoNetwork plays the role of the central repository of geospatial metadata, ensuring comprehensive and reliable information about the project's datasets in complement to CKAN. It facilitates effective data discovery, promotes collaboration, and fosters the seamless flow of geospatial insights among diverse stakeholders.

Several public authorities already use GeoNetwork to publish their geospatial datasets: Flanders, Normandie, Austria, Germany, etc.

### 4.1. Initial implementation

The initial implementation of GeoNetwork involved deploying the web application on a robust web server, such as Jetty or Tomcat. The web interface was thoughtfully designed to provide five core pages: Home, Search, Map, Contribute, and Admin, catering to different user needs.

To maintain data consistency and a standardized representation, GeoNetwork incorporates metadata templates and schema validation, allowing for consistent and standardized data representation. User profiles and privilege management were established to control access and define roles within the platform.

### 4.2. Audience

GeoNetwork's audience includes project members, data contributors, researchers, and stakeholders involved in geospatial data activities.

It accommodates users with diverse roles and responsibilities, promoting collaboration and data sharing among project stakeholders.

## 4.3. GeoNetwork management

### 4.3.1. Web interface

GeoNetwork web interface is available under the path `/srv/eng/catalog.search#/home`. The web app folder name in Jetty or Tomcat web server normally needs to be included in the complete URL. For instance, the GeoNetwork instance deployed locally in the folder GeoNetwork of the web server will have its web interface available at: <https://geonetwork.mobidatalab.eu/GeoNetwork-4.0.5-0/srv/eng/catalog.search#/home>.

The web interface has 5 main pages: Home, Search, Map, Contribute and Admin.

1. Home page: shows an overview of the metadata in GeoNetwork by Topic and Category.
2. Search page: shows extensive search and filter functionalities and search results.
3. Map page: shows an interactive world map to visualize supported data sources, for instance, Web Feature Service (WFS), Web Map Service (WMS) data.
4. Contribute page: contains several subpages, notably: Editor board, Add new record, Import records.
5. Admin console: contains several subpages, notably: Metadata & templates, Users and groups, Harvesting, Settings, and Tools.

Unauthorized users can see only the Home page, Search page and Map page, which can also be adjusted in Settings. The Contribute page is accessible to users with an editor profile or above. The Admin console is reserved for users with a User Administrator profile and above. Details on User and groups and their permission management are discussed in section 4.2.3).

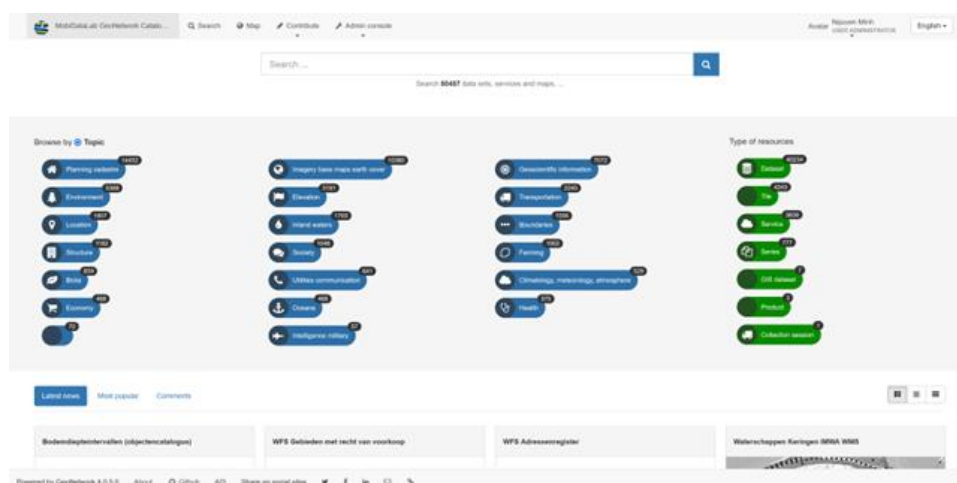


Figure 37: GeoNetwork home page for User Administrator

### 4.3.2. Metadata management

The Search page contains a search box on top for text search and temporal search, a mini map for spatial search, a filtering panel on the left and a search results panel on the right (Figure 38). In addition, an editing toolbar is placed on the right of editable metadata. Users can query for the metadata using one of the following:

1. Text search: the search box on top is used for simple text search in all accessible metadata in GeoNetwork.
2. Filtering: The filtering options on the left panel allow users to filter only metadata with specified attributes, such as keywords, organizations, data types, etc. The attributes to be used as filters can be configured in Settings.
3. Spatial search: The pen tool on the mini map is used to draw the extent of the spatial search. The metadata whose extent intersects with the query extent are returned.
4. Temporal search: The three-dot button next to the search box enables temporal search capability. Users can then specify the time range as a query.

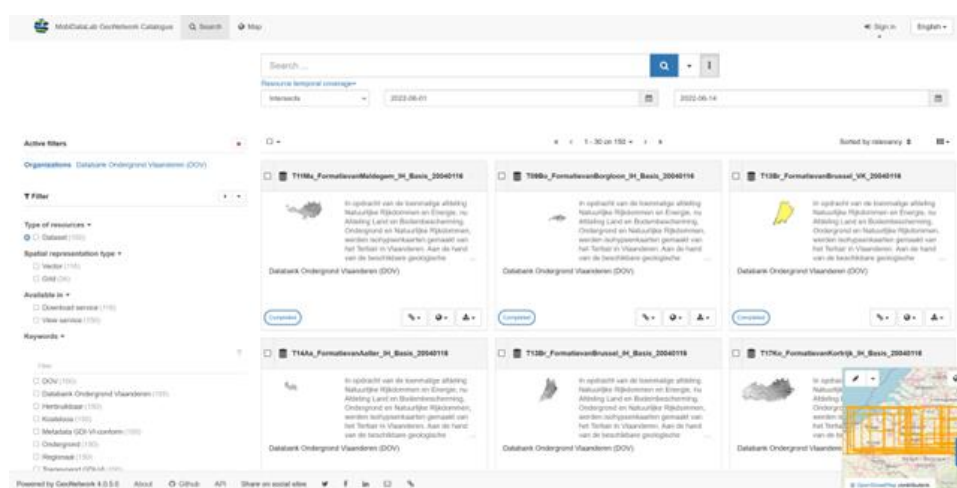


Figure 38: Search in GeoNetwork

GeoNetwork allows users to visualize standardized external data sources directly on the map viewer (Figure 39). Supported data sources are WMS, WFS, WMTS, Keyhole Markup Language (KML) services and KMZ (KML Zipped) files.

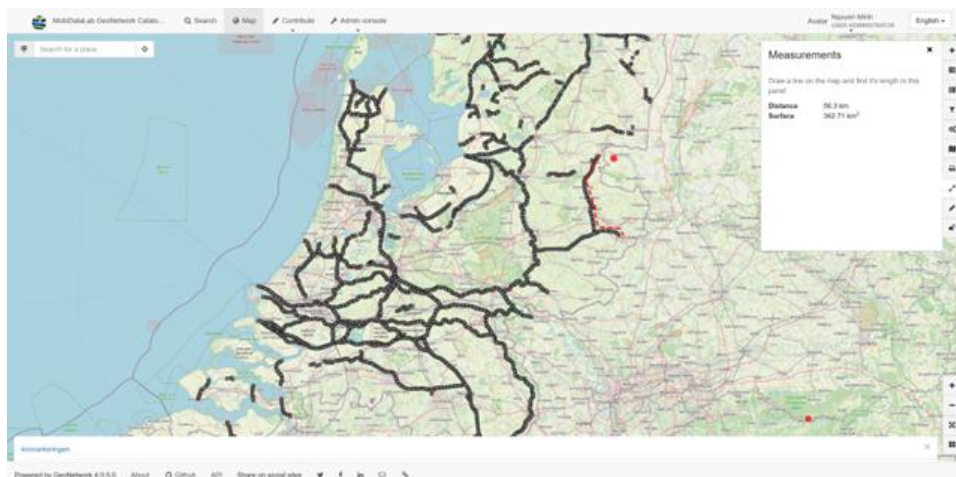


Figure 39: Map viewer with WMS layer and annotations

### 4.3.3. Harvesting

The user needs to have at least a User Administration profile to manage catalogue harvesters. From the menu Admin console > Harvesting > Catalog harvesters, users can add and manage catalogue harvesters (Figure 40). Some supported protocols for harvesting are OGC CSW 2.0.2, OGC WFS, GeoNetwork, Geoportal, and ArcSDE<sup>12</sup> (SDE for Spatial Database Engine). The harvesters shall be scheduled to run during off-hours to avoid degrading performance.

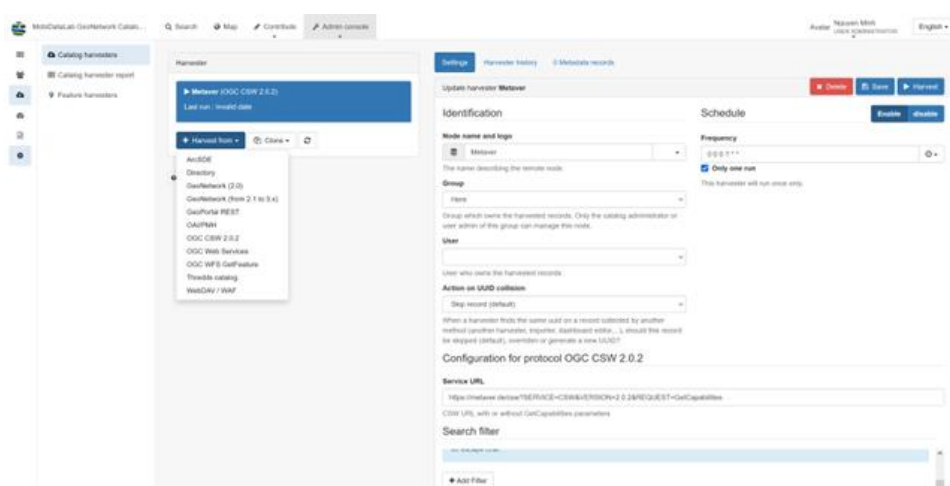


Figure 40: CSW Harvester in GeoNetwork

<sup>12</sup> <https://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/000500000001000000>

### 4.3.4. Interoperability

The Catalogue Service for the Web (CSW) endpoint exposes the metadata records in your catalogue in XML format under the path /srv/eng/csw using the OGC CSW protocol (version 2.0.2), specifically the CSW and CSW-T protocols<sup>13</sup>:

1. CSW: Provides the ability to search and publish metadata for data, services and related information.
2. CSW-T: Provides an interface for creating, modifying and deleting catalogue records via the CSW protocol.

A typical CSW request sent to GeoNetwork takes the following form:  
<https://geonetwork.mobidatalab.eu/GeoNetwork-4.0.5-0/srv/eng/csw?SERVICE=CSW&VERSION=2.0.2&REQUEST=GetCapabilities>

### 4.3.5. User management

Permissions in GeoNetwork are managed via Users and groups, which can be found in the Admin console (Figure 41, Figure 42). There are 4 user profiles in GeoNetwork, representing 4 privileges levels:

1. Registered user: has read permission
2. Editor: has read and write permission
3. Reviewer: has read, write and publish permission
4. User administrator: has read, write, publish and harvester configuring permission.

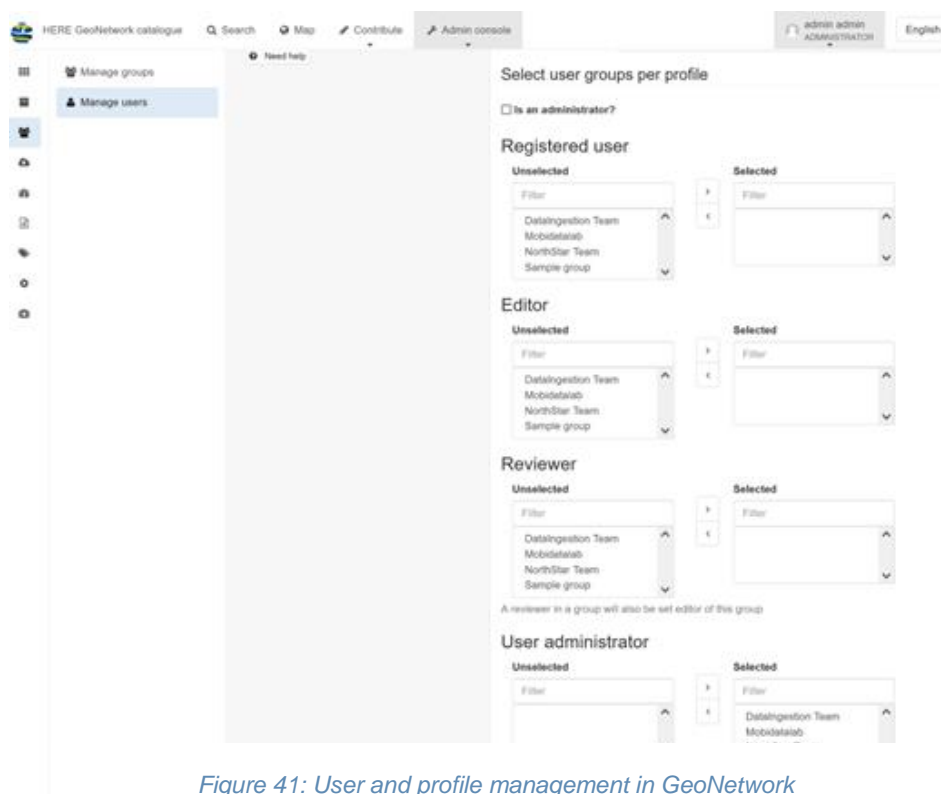


Figure 41: User and profile management in GeoNetwork

Each user is assigned a user profile that permits them to access metadata belonging to a user group. Matrix-based permission enables users to play different roles in different groups.

<sup>13</sup> <https://geonetwork-opensource.org/manuals/4.0.x/en/api/csw.html>



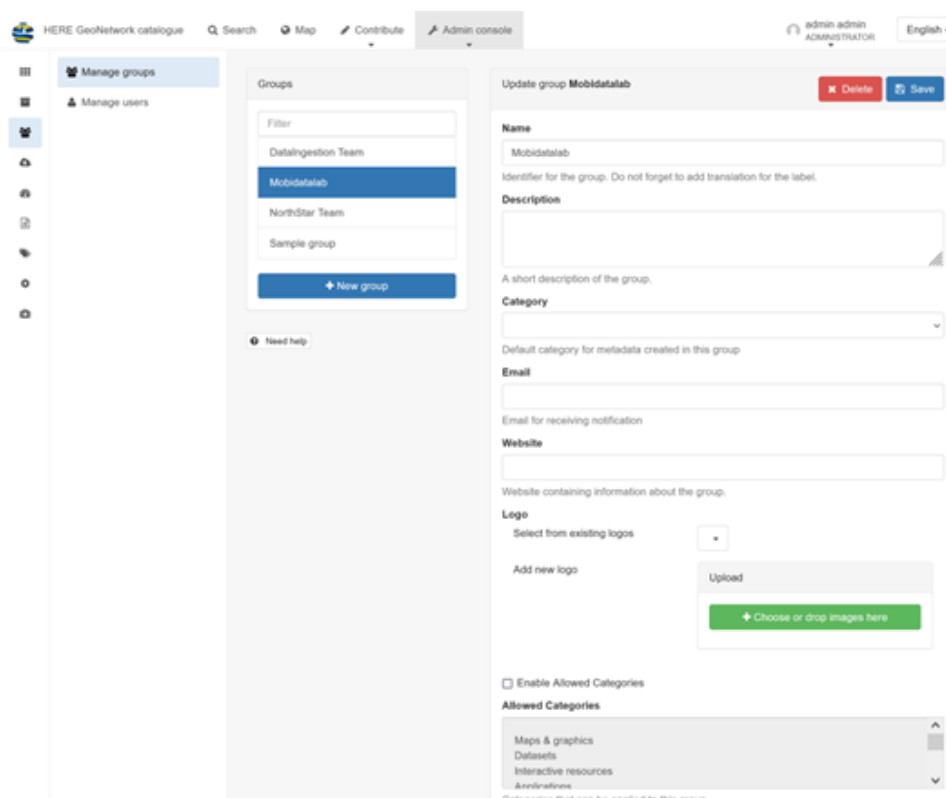


Figure 42: Group management in GeoNetwork

## 4.4. Constraints

### 4.4.1. Deficiencies in harvested metadata

GeoNetwork serves as a fundamental platform for managing and harvesting standardized geospatial catalogues and facilitates schema validation. However, despite its commendable features, achieving true interoperability remains a considerable challenge, attributed mainly to the varying standards and diverse implementations employed by individual data publishers and data catalogues.

GeoNetwork realizes interoperability by hosting standardized schemas and metadata as is, with little semantic mappings between different schemas, limited to the title, the data format, the publisher's name and the geolocation of the data. This implementation is deemed sufficient for a simple use case of querying data based on the title and/or the publisher. Complex queries based on other attributes like publishing date would not be possible out-of-the-box as the information is stored differently in each standard, with a unified semantic mapping of these attributes. These deficiencies in heterogeneous standards can be coped with by either manually updating the missing information using the built-in web interface or systematically transforming information between metadata formats. An effective solution, that requires expert knowledge from the data publisher, is out of the scope of this section.



Another deficiency in harvested metadata noted is unexpectedly the publisher information. This absence can be attributed to the fact that data publishers exclusively upload the metadata for the data they possess to their respective GeoNetwork instances. As a result, the publisher's information becomes implicit and often left out during the process. While the deduction of publisher information is straightforward within each respective instance, complications arise when the harvested metadata is aggregated into a centralized repository.

The effective solution to the deficiencies in this section requires expert knowledge and realization from the data publisher. A suboptimal resolution, detached from the publisher, involves the incorporation of an intermediary step during data ingestion. This intermediary component systematically facilitates the transformation of metadata and addresses incomplete metadata attributes where possible cohesively. Scaling this transformation layer demands significant efforts, primarily due to the need for diverse transformation logic tailored to each data source. Consequently, the complexity involved in the realization of these distinct transformation requirements falls out of the scope of the current section.

## 4.5. Documentation/support

The provision of comprehensive and up-to-date documentation and support resources emerged as an imperative requirement. As GeoNetwork's capabilities expanded, empowering users with the knowledge to harness its full potential became a top priority. GeoNetwork documentation is hand-written with comprehensive examples by community contributors in English and French<sup>14</sup>. Their API references are generated from source code, thus always staying up to date. Moreover, users and developers can also receive support from GitHub Issues<sup>15</sup> and Gitter<sup>16</sup>.

## 4.6. Product maturity

GeoNetwork development dates to 2001 when it served as a Spatial Data Catalogue System for the Food and Agriculture Organization of the United Nations (FAO), the United Nations World Food Programme (WFP) and the United Nations Environmental Programme (UNEP). Throughout the years, the project has accumulated hundreds of contributors, individuals and organizations included, has published over 70 releases since 2005 and is part of the not-for-profit Open Source Geospatial Foundation (OSGeo)<sup>17</sup>. GeoNetwork is widely used as the foundation of Spatial Data Infrastructures worldwide<sup>18</sup>. The project is licensed under GNU General Public License (GPL) version 2.0, providing royalty-free re(use) of the software, ensuring access to the source code for audit and modification and the ability to redistribute the software at no additional cost.

<sup>14</sup> <https://docs.geonetwork-opensource.org/>

<sup>15</sup> <https://github.com/geonetwork/core-geonetwork/issues>

<sup>16</sup> <https://gitter.im/geonetwork/core-geonetwork>

<sup>17</sup> <https://www.osgeo.org/projects/geonetwork/>

<sup>18</sup> <https://docs.geonetwork-opensource.org/4.2/annexes/gallery/>

Bugs and issues are reported and tracked publicly on GitHub, enabling active participation from developers and users in the development and bug-fixing process, promoting transparency and knowledge exchange with large numbers of experienced developers and contributors. New functionalities and improvements from contributors are tested and reviewed thoroughly by the maintainers, keeping decent code quality. Upon approval, new changes can be integrated into the core components automatically thanks to the comprehensive continuous integration and continuous delivery pipeline. In addition, GeoNetwork vision is aligned with OSGeo, to encourage interoperability through the integration of various well-known standards for geographic metadata <sup>19</sup>, namely iso19115-3:2018, iso19139:2007, Dublin core, etc.

## 4.7. Learned from the virtual and living labs

GeoNetwork, as CKAN, was integrated within the Virtual Lab where users were able to browse through an extensive list of dataset sources and get information about each dataset without leaving the Virtual Lab's environment. The endpoint of GeoNetwork was also presented to discover datasets.

Lessons learned from the datathon, hackathon, and codagon will guide improvements and future enhancements of GeoNetwork. These labs allow MobiDataLab to learn about the use and priority given by the participants, the difficulties encountered by them, as well as the appreciated features (this will be discussed in the next section).

Otherwise, some identified benefits are, so far, the spatial search and the integration into the virtual lab. The issues are missing metadata, the search function that brings "wrong" results and the lack of translations (the datasets are in the original language of the source).

---

<sup>19</sup> <https://docs.geonetwork-opensource.org/4.2/annexes/standards/>

## 5. Ground testing

Ground testing through the first and second virtual and living labs, the datathon & hackathon, is observed from the platform point of view through response time, services and processors, URLs, and connectivity. From the data catalogue point of view: amount, variety, classification and search. From the tooling point of view: documentation, software tools usable and standards offered.

### 5.1.1.1. Virtual and living labs

Apart from making accessible a demo on how to use the catalogues, during the virtual and living labs, the metadata of both of CKAN and GeoNetwork catalogues was exposed in the Virtual Lab and links to datasets related to the challenges were provided.

### 5.1.1.2. Datathon and hackathon feedback

Feedback from the datathon and hackathon is an element expected from the attendees. The idea is to validate the usage conditions with mobility industry players who will bring their external point of view and will be confronted with EU expectations and the product offered to the public.

During the datathon, MobiDataLab offered several challenges, many datasets and a very wide range of tools that explained in a context of a two-day event was overwhelming for some of the participants.

The strengths during the datathon were:

- the robustness of the platform that was adequately sized to serve seamlessly the data consumer audience,
- the relevance of the services put in place for the data to be findable, usable from an entry point that refers to the outside storage world,
- the ability of the platform to take advantage of cloud services and functionalities when it came to elastically growing to satisfy the demand for data storage, compute processing, services hosting, infrastructure design, constraints compliance, etc.

The weaknesses during the datathon were:

- The product was missing sharper dataset segregation. For instance, CKAN's data categories might not have been used by the data consumers, potentially, because the communication was not intended to include a tutorial of standard tools/data servers (as it was assumed that mobility industry actors were aware of what to expect from CKAN and GeoNetwork).

### 5.1.1.3. Expected evolutions from the virtual and living labs

The datathon stage expectation sends the signal that the MobiDataLab platform is fully data services oriented with all the following characteristics when browsing data: data search and filtering functionalities, GDPR enforcement, network bandwidth (to carry data transfer), data storage capability, communication encryption, as well as service continuity and data accuracy.

The hackathon stage expectation was a step beyond the datathon stage. On top of the data layer of the MobiDataLab platform, sets of APIs were offered to fetch datasets as was the case during the datathon with dedicated browsing tools. Additionally, APIs benefited from journey planners and other processors on the platform. These could be further combined with external APIs that the hackathonist could use to work with to solve their use case of interest.

At the end of the hackathon, all elements raised and learned from the event were assessed to bring potential add-ons to the infrastructure, services or data. The hackathon confirmed the public usability of the Data Catalogue, and it provided further information to prepare for the codagon (a medium size project). Since the last event started soon after the hackathon, no major changes made done to the hackathon platform, services and data (unless a breakthrough had been revealed and needed a paramount adjustment or addition, a modification would have been done).

The codagon stage expectation is the last opportunity to receive feedback about the virtual and living labs. Particularly, thanks to the developments made, expected to last between one or two weeks, before being presented as a visit card and after the application/solutions will be submitted (even eventually selected) to the MobiDataLab jury members.

### 5.1.1.4. To be done

SSL certificates are an answer to the communication privacy liability the audience is expecting from the MobiDataLab when it comes to encryption of HTTP requests between the platform and the internet visitor. The certificate involved in the process has a 3-month validity limit and it must be renewed manually by the certificate issuer. To renew the request a challenge question will have to be answered by the re-newer to validate the verification process.

### 5.1.1.5. Upcoming changes in the cloud services

Cloud service deprecation happens when a Cloud Service is planned to be discontinued or replaced by a product considered more suitable or up to date as a cloud offer. Azure users have been informed that "Application Gateway V1.0 SKU retires on April 28, 2026".

### 5.1.1.6. Updates to come

The infrastructure of the data platform, between the datathon and the hackathon, should remain as it is. However, a metadata update will be needed after each event to incorporate metadata that will match the use cases challenge described in each event.

An update of the journey planner database for timetable accuracy might be also needed to adapt the platform to the journey planner license restrictions by HOVE (editor of the tool). This stabilised state of the platform includes the hackathon and codagon layer, which consists of a set of APIs for developers to include the MobiDataLab platform in their applications.

For instance, some APIs offer metadata services, such as CKAN and GeoNetwork, which are available along with processors functionalities as well as the services catalogue.

## 6. Conclusions

This report proved the use given to MobiDataLab's reference data catalogue with its strengths, and weaknesses, but also with the improvements made. Through this deliverable, it was documented how with the help of the Reference Group of transport stakeholders and the available technologies, it is possible to build tools to facilitate the discovery of datasets for potential consumers and re-users to use in diverse use cases and solve transport and mobility challenges.

Data discoverability is improved thanks to different cataloguing solutions, generalist or thematic, that were evaluated. This demonstrator aimed to show the evolution of the cataloguing solutions that were selected to test and develop in the context of a Transport Cloud for which portability is a strong requirement. The solutions covered were CKAN and GeoNetwork. These solutions were offered to Lab participants to help them discover mobility data.

This delivery demonstrated how the goals of task 4.2 were achieved. We used popular vocabularies and data-sharing standards such as DCAT, CSW and CKAN API to harvest up to 115K datasets. The metadata of each dataset contains information such as the title, description, keywords, publication date, source, publisher, and data format, which allows both human understanding and automatic discovery by software agents. It also provides structural information about the internal structure of the datasets which makes possible to interpret manually/automatically data schemas.

This demonstrator is part of a set of demonstrators corresponding to the other tasks of WP4, regarding data access services, data processors, and data anonymisation. These four demonstrators are the software counterpart of the FAIR principles. The reference data catalogue corresponds to the first letter of this acronym "F" of findability. Findability is a prerequisite to ease data access (A), data interoperability (I) and data reusability (R). Therefore, this catalogue is an essential piece to the other tasks.

The link between catalogues, services, processors and anonymisation tools has been further developed to be more effective and better integrated as part of the work of the MobiDataLab WP4 partners, this can be explored further in the deliverables 4.6, 4.8 and 4.10.

## 7. Annexes

### 7.1. Containerization

Regarding containerization, the metadata service installation on the cloud was done at the beginning with GeoNetwork which comes as a Java ARchive (JAR). Java ARchive comes with ties on the Java/Java Virtual Machine (JVM) version that is compatible to support the whole set of instructions that reside over Java libraries on which the JAR application is built upon. Java applications are built and tightly tied on the application server for which the Java ARchive is intended to run. Since a Java application promises portability thanks to the Java language, the application server will enclose that portability to a narrow set of options due to the intrincating level that ties the Java application with the application server. The further the ties between the application server and web application, the less the web application is runnable on different application server brands or even, different versions of the same brand of application server.

Using Java technologies in a web application will induce the narrowing of the application server type and within the application server type, narrowing to an eligible version of the application server. Then finally, from the application server type and application server version will come the JVM version supported by the application server that also fits the JVM version compatible with the web application.

With such a decremental selection pattern, it is essential to select a container image hosting both the eligible server application version running the JVM that will correspond to the web application acceptable JVM version before deploying a tailor-made container into which the web application's will be loaded and configured.

GeoNetwork is not simple when it comes to setting parameters in the configuration files. The set of configuration files will see the very same information to register thus acting like a double entry that could lead to inconsistencies should a modification be done on one entry without the other entry to be accordingly modified. The configuration process is not more supported than the whole project in general and the installer must rely on pieces of information left at different times online and representing an unrelated legacy of the application relics that are no longer always relevant.

The high availability functionality, included natively in containerization, takes a different turn when hosted by the cloud due to the lack of interaction with the architect in terms of deployment style, orchestration, storage/drive to be set and the sharing mode (due to inexistence of DockerFile/DockerCompose functionalities). The choice for building a containerized environment is whether by an individual set of instances ignoring each other or by stepping into Kubernetes. The second choice is far from the application needs because the GeoNetwork ecosystem is limited to the web application stack, the Elasticsearch and its storage in a database service outside the GeoNetwork instance.

To ensure high availability, the solution is whether to instantiate the Docker image with the "restart" option at "Always" or to abandon the cloud container service to favour a virtual machine (where the system administrator will have full control). The cloud containerization option has been preferred because the service is available on any cloud platform and any Cloud Operations (CloudOps<sup>20</sup>) as it will be able to intervene while adding a System Administrator (SysAdmin) layer that would not satisfy the first expectation of the project which is to be a "cloud" oriented platform.

CKAN could not be treated identically than GeoNetwork since the CKAN ecosystem comes internally with a dedicated Nginx<sup>21</sup> and externally, a cache service: Redis<sup>22</sup>, and an indexer: Solr<sup>23</sup>. Notably, the internal CKAN ecosystem is made of plugins that must be installed along with dependencies requiring modification to the ckan.ini configuration file before the test and validation phase begin. CKAN lack of maintenance will allow a seamless installation and will often require extra re-factory of the plugins to make the extension work with the current version of Python and reliably interface with the current version of CKAN.

<sup>20</sup> CloudOps "is the practice of managing delivery, tuning, optimization, and performance of workloads and IT services that run in a cloud environment including multi, hybrid, in the data center and at the edge." (<https://www.vmware.com/topics/glossary/content/cloud-operations.html>)

<sup>21</sup> "Nginx is a web server that can also be used as a reverse proxy, load balancer, mail proxy and HTTP cache." (<https://nginx.org/>)

<sup>22</sup> "Redis is an open-source in-memory storage, used as a distributed, in-memory key-value database, cache and message broker, with optional durability." (<https://redis.io/>)

<sup>23</sup> "Solr is highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying, automated failover and recovery, centralized configuration and more." (<https://solr.apache.org/>)



## 7.2. Data mutualisation

With a cluster of virtual machines came unexpected difficulties in the initial phase. In terms of data mutualization, on N-Tier<sup>24</sup> configuration, the application ecosystem is spread over hosts based on the balance of resources expected by the application against the resources available on the host (in terms of Central Processing Unit (CPU)/cores, memory, functional isolation, applicative isolation, network zoning, etc.). There is one host as the front end, one host acting as the middleware and one host acting as the back end. A topology built upon one host at each step will not consider alternating between hosts, should the front end, middle and back end have to withstand critical activity requiring multiple actors at the same position. The multiplication of the actors may imply synchronization for the database, authentication for middleware and session sharing for the front end that will affect the deployment of the host embedding the front middle or back office.

Dedicated Virtual Machine (VM) images are a common reflex when it comes to industrializing resiliency and out scaling an infrastructure. VM images come with seamlessness expectations when the image is instantiated and put to service without operating system changes (the only variation is the virtual hardware sizing). Leveraging the VM images auto-deployment is possible if the OS within the image along with the pre-installed application are fitted with the proper configuration files, software/module distribution and the expected file sharing. These elements together will allow each instance to access live data generated by a similar VM and will allow the newcomer to take place in the cluster of responders as if the VM was part of the game from the beginning.

<sup>24</sup> “Multitier architecture (often referred to as n-tier architecture) is a client–server architecture in which presentation, application processing and data management functions are physically separated.”  
([https://en.wikipedia.org/wiki/Multitier\\_architecture](https://en.wikipedia.org/wiki/Multitier_architecture))

### 7.3. Load balancing

Load balancing is an asset placed between the internet and the information system to act as the mechanism that checks the availability of the targets that will receive the visitor requests and route the visitor requests to any of the hosts when the host is considered as responding. Any infrastructure is designed with capacity; however, this can be underestimated particularly if the traffic/activity follows wide variations according to daily pic hours, yearly calendars or punctual special events. Maintaining service continuity is possible by anticipating the visitor influx with an increase in capacity that can be achieved manually or thanks to services that will be monitoring the host's activity and will instantiate new hosts when an indicator reaches a threshold for a certain duration. The auto-scaling will rack new servers (performing identically as a human) to maintain all-time resilience, constant response time and user experience quality.

Once in production, keeping high availability topology in mind, the platform will not allow any operation that requires the one-to-one relationship between an internet visitor and a server. The platform exploitation must be in form for any visitor and host. The Datastore functionality is offered by both GeoNetwork and CKAN to open metadata storage capacity to users who want to store metadata on the platform and consume the data along with the references offered by the platform. The Datastore will oblige the load balance to guarantee a sticky mode<sup>25</sup> between the user and the host while that stickiness cannot be guaranteed for the day, week or month after the dataset is dropped on the host locally. The file drop has been discarded due to an incompatibility with the load balancer and it has been decided that the composition of datasets will be hosted on the S3 equivalent and referred to the data consumer through a link in the metadata of the dataset.

<sup>25</sup> Sticky mode: when the re-user remains connected to the same server within the cluster

## 7.4. Issues managing groups

The organization by group(s) can be applied to all the datasets of an organization at the harvesting step (see harvesting configuration details), this can be useful for data portals such as transport national access points of Belgium (NAP ITS BELGIUM) containing only a certain category of datasets “Mobility and Transport”. However, the issue arises if not all the datasets belong to the same group, as in this case, it is best not to categorize all the datasets within the same group. An alternative is to associate a single dataset to a particular group, but this action manually would take a lot of time since the datasets come directly from other portals (they are not created one at a time by MobiDataLab). Since there are too many datasets to do this action manually, it can be done only with a certain amount. Therefore, only selected datasets were placed into these groups (particularly the ones which had enough relevance for the use case challenges). This helped in identifying keywords and creating dictionaries related to the groups. However, another method was employed to cover more datasets more dynamically, this will be covered in the section related to the API of CKAN.

## 7.5. FileStore API functions

Multipart/form-data can be posted to the API and the key, value pairs will be treated as if they were JSON objects.

The extra key upload is used to post the binary data. For example, to create a new CKAN resource and upload a file to it with a client URL (cURL), which automatically sends a multipart-form-data heading when using the --form option:

```
curl -H'Authorization: your-api-key' 'http://yourhost/api/action/resource_create' --form
upload=@filetoupload --form package_id=my_dataset
```

To create a new resource and upload a file to it using the Python library requests. Requests automatically send a multipart-form-data heading when you use the files= parameter:

```
import requests
requests.post('http://0.0.0.0:5000/api/action/resource_create',
              data={"package_id": "my_dataset"},
              headers={"X-CKAN-API-Key": "21a47217-6d7b-49c5-88f9-72ebd5a4d4bb"},
              files=[('upload', file('/path/to/file/to/upload.csv'))])
```

To overwrite an uploaded file with a new version of the file, post to the resource\_update() action and use the upload field:

```
curl -H'Authorization: your-api-key' 'http://yourhost/api/action/resource_update' --form
upload=@newfiletoupload --form id=resourceid
```

To replace an uploaded file with a link to a file at a remote URL, use the clear\_upload field:

```
curl -H'Authorization: your-api-key' 'http://yourhost/api/action/resource_update' --form
url=http://example.com --form clear_upload=true --form id=resourceid
```

## 7.6. Harvesting configuration

Description of the objects in the harvesting configuration of CKAN:

- **api\_version:** You can force the harvester to use either version 1 or 2 of the CKAN API. Default is 2.
- **default\_tags:** A list of tags that will be added to all harvested datasets. Tags don't need to previously exist. This field takes a list of tag dictionaries (see example), which allows you to optionally specify a vocabulary.
- **default\_groups:** A list of group IDs or names to which the harvested datasets will be added to. The groups must exist.
- **default\_extras:** A dictionary of key value pairs that will be added to extras of the harvested datasets. You can use the following replacement strings, that will be replaced before creating or updating the datasets:
  - {dataset\_id}
  - {harvest\_source\_id}
  - {harvest\_source\_url} # Will be stripped of trailing forward slashes (/)
  - {harvest\_source\_title}
  - {harvest\_job\_id}
  - {harvest\_object\_id}
- **override\_extras:** Assign default extras even if they already exist in the remote dataset. Default is False (only non-existing extras are added).
- **user:** User who will run the harvesting process. Please note that this user needs to have permission for creating packages, and if default groups were defined, the user must have permission to assign packages to these groups.
- **api\_key:** If the remote CKAN instance has restricted access to the API, you can provide a CKAN API key, which will be sent in any request.
- **read\_only:** Create harvested packages in read-only mode. Only the user who performed the harvest (the one defined in the previous setting or the 'harvest' sysadmin) will be able to edit and administer the packages created from this harvesting source. Logged in users and visitors will be only able to read them.
- **force\_all:** By default, after the first harvesting, the harvester will gather only the modified packages from the remote site since the last harvesting. Setting this property to true will force the harvester to gather all remote packages regardless of the modification date. Default is False.

- **remote\_groups:** By default, remote groups are ignored. Setting this property enables the harvester to import the remote groups. There are two alternatives. Setting it to 'only\_local' will just import groups which name/id is already present in the local CKAN. Setting it to 'create' will make an attempt to create the groups by copying the details from the remote CKAN.

## 7.7. Translation technical implementation on CKAN

The code implementation of the translation was on [the base harvester](#), and a new optional argument ("-t" / "--translate") [was added](#) to the harvester command line interface. We use the [deep-translator](#) Python library with Google Translate service to translate the content of the dataset's metadata. Some changes had to be done on the other harvester extensions to support the translation: [DCAT](#), [CSW](#) [inspire](#). Various improvements were done to the translation such as adding a blacklist of attributes that should not be translated and recursive processing to translate all dataset content. The details can be consulted in the MobiDataLab GitHub repository<sup>26</sup>.

Here is how to run the harvester with English translation enabled:

1	<code>./usr/lib/ckan/default/bin/activate</code>
2	<code>sudo ckan --config=/etc/ckan/default/ckan.ini harvester gather-consumer -t en</code>

*Figure 43: Harvester translation command in CKAN*

<sup>26</sup> <https://github.com/MobiDataLab/ckanext-harvest/commit/6fdf891d588a8ba212a2fdabbc7b9ae92d356351>



## 7.8. Translation limitations

As we are using a free Google translation account, the API requests are limited to:

- 5000 characters per request, to work around this limitation for attributes such as a description that may reach this limitation, we make multiple requests by splitting the content of such attributes into chunks of 5000 characters so it can translate all the metadata content.
- 2 million characters per day and 100,000 characters per 100 seconds, to work around this limitation we introduced a small delay between requests, so it fits in the limitation window per second and day.

Moreover, some translated attributes may break the CKAN database constraints. For example, the name or the tags get translated in some datasets with more than 100 characters, thus it will not be harvested on CKAN. As a workaround, it is possible to truncate the translated tags to fit in the 100 characters constraint.

## 7.9. List of available CKAN API clients

Here you can find a list of available CKAN API clients you can use depending on your development environment:

- Python
  - ckanapi: official client for CKAN core. Simple Python wrapper around the CKAN action API. It raises exceptions on errors and works locally (from a plugin) or remotely in a very similar way. It also supports use with `paste.fixture.TestApp` for simplified tests (e.g. `ckanext-canada` tests).
  - A Python function for posting to CKAN's Action API, call a function to post a Python dict to an Action API function and get the response back also as a Python dict.
  - A `ckan-api-client`: an improved client for the CKAN API, providing a low-level client, a high-level client, a synchronization client and it attempts to get work around some common issues with that API.
  - `ckanclient`: CKAN Python Client - deprecated in favour of CKAN API. Supports CKAN's legacy RESTful Model and Search APIs (v1 & v2), the legacy Elasticsearch DataStore API, basic support for the v3 Action API, and code for uploading files to CKAN using the FileStore API.
- JavaScript (Node and Browser)
  - `ckan.js`: JavaScript client library for CKAN with support for Node and the browser. The library provides full support for accessing both the CKAN Catalogue and CKAN DataStore API.
- Ruby
  - Ruby Client
- PHP
  - `Ckan_client-PHP`
  - PHP CKAN client - from Silex. Uses Guzzle for the HTTP. Packaged using Composer; it is also on GitHub. Covers several action functions for reading/writing packages, resources, etc. Comes with tests.
- Java
  - `Ckan_client-J` (Java client)
  - Jackan Java client with API v3 support and Apache 2.0 license. Comes with integration tests.
  - `ckan-java-api` Java Client - CKAN Action API full compatibility
- Perl
  - `net-ckan` (PERL client)

- Command-line
  - ckanapi: Official client for CKAN core.
  - dpm: data package manager command-line client and Python library.
- OpenRefine (Google Refine)
  - Google Refine CKAN Extension: Google Refine client which allows you to get and push data to and from a CKAN instance using Google Refine.
- R
  - ckanr: R client for the CKAN API
- Geospatial
  - QGIS CKAN plugin to load and display spatial data in CKAN
  - Terria.js: open source framework for web-based geospatial catalogue explorers
- Content Management Systems
  - WordPress
  - WP-CKAN
  - Drupal
  - GovCMS
- Extract, Transform & Load (ETL) Tools
  - Hitachi Pentaho Kettle
  - OpenGov Implementation
  - Localdata Fork
  - Safe FME
  - OpenGov Implementation
  - Australia Open Council Data Implementation

## 7.10. CKAN Harvester API

- You can access CKAN harvest logs via the API:
- \$ curl https://ckan.mobidatalab.eu/api/3/action/harvest\_log\_list

Allowed parameters are:

- level (filter log records by level)
- limit (used for pagination)
- offset (used for pagination)
- To get the CKAN harvest sources list via API (with a default limit of 100 items):
- \$ curl https://ckan.mobidatalab.eu/api/3/action/harvest\_source\_list

The limit can be set to a bespoke value in the config for CKAN under ckan.harvest.harvest\_source\_limit.

An optional query param organization\_id can be used to narrow down the results to only return the harvest sources created by certain organization's by supplying their respective organization id -> /api/action/harvest\_source\_list?organization\_id=<some-org-id>

- To create a harvest source via the API:
- \$ curl https://ckan.mobidatalab.eu/api/3/action/harvest\_source\_create
- To create a harvest job for a source via the API:
- \$ curl https://ckan.mobidatalab.eu/api/3/action/harvest\_job\_create
- To get the documentation of an API, you can use the bellow API:
- \$ curl https://ckan.mobidatalab.eu/api/3/action/help\_show?name={action}

Where {action} is the action name, for example: harvest\_source\_create.

- \$ curl [https://ckan.mobidatalab.eu/api/3/action/help\\_show?name=harvest\\_job\\_create](https://ckan.mobidatalab.eu/api/3/action/help_show?name=harvest_job_create)

### 7.10.1. Ckan API toolbox example for handling datasets

As mentioned earlier organizations, datasets and resources can be modified directly from the API by the administrations with the help of a token.

At some point in the project, it was considered to translate the titles and tags of the datasets in CKAN through an API patch action, but this idea was abandoned. This was decided since most of the datasets are harvested and every time that there would be an update to a resource, it would have been necessary to “patch it” with the translation afterwards. Instead, it was decided to do the translation directly as the resource is harvested. However, this action was decided to be applied to categorize certain datasets into groups and sub-groups by adding a field into the metadata in relation to the available tags of the datasets. This was done with the help of the `action.package_search` API function, which searches for the datasets containing a certain tag, if it finds the tag and we want to add a dictionary to the list of metadata, we have to provide the key and the value. Here is the function:

```

1  #Enter the URL of the CKAN, your API key and the query you are searching for
2  def metadata_modif(url, api_key, query):
3
4      demo = RemoteCKAN(url, apikey=api_key)
5
6      # If you are searching for an dataset: q='+dataset:'+query
7      dataset = demo.action.package_search(q=query, rows=1000)
8
9      for i in range(len(datasets['results'])):
10
11          print('name of the dataset' +str(datasets['results'][i]['name'])) #name of the dataset
12          data = demo.action.package_show(id=datasets['results'][i]['name'])
13          extra_list=data['extras'] #list of dictionaries containing the metadata of the dataset
14          print(extra_list)
15
16          #If you want to add a dictionary to the list of metadata:
17          dict={'key': 'Group', 'value': 'Shared mobility'}
18          extra_list.append(dict)
19          demo.action.package_patch(id=datasets['results'][i]['name'], extras=extra_list)
20
21  #Example of test:
22  metadata_modif(CKAN_ADRESS['url'], CKAN_ADRESS['api_key'],'car share')
```

The screenshot shows the 'Overheid.nl' organization page. On the left, there's a sidebar with 'Followers' (0), 'Organization' (Overheid.nl logo and description), 'Social' (Twitter, Facebook), and 'License' (Creative Commons). The main content area is titled 'History economic demography' and includes a description: 'Demography of companies by business activity (SBI'93 and SBI'74) and bankruptcies (SBI'93). 1983 - 2006. Changed April 08, 2008. Frequency: Discontinued.' Below this, there's a 'Data and Resources' section with 'API' and 'Feed' links, each with an 'Explore' button. At the bottom, an 'Additional Info' table lists metadata.

Field	Value
Source	<a href="https://opendata.cbs.nl/statline/portal.html?_la=nl&amp;_catalog=CBS&amp;tableId=37912">https://opendata.cbs.nl/statline/portal.html?_la=nl&amp;_catalog=CBS&amp;tableId=37912</a>
Last Updated	October 13, 2023, 5:20 PM (UTC+02:00)
Created	July 12, 2023, 8:18 AM (UTC+02:00)
Group	Population distribution — demography & socioeconomics
harvest_object_id	4f6a0c05-52be-4812-87bb-9a92d2cb3ff6

Figure 44: Datasets update through the API

It must be noted that this method comes with its own inconveniences as a tag or rather the word stem of the tag might be associated incorrectly to a group or sub-group as Sorl<sup>27</sup> applies some pre-processing and stemming<sup>28</sup> when searching. Therefore, certain datasets might not be well categorized, an improvement should be done in relation to this issue. The other problem that arose was the fact that certain datasets should be able to fall under more than one group, but if a dataset was attributed to a certain group, this one would not be updated or added after the first group. The function should be further developed to work in the desired way. The idea of making this function more efficient using dictionaries instead of a single tag was also explored, but the parameters did not seem that allow it. Most importantly, at that point we did not have a complete dictionary to feed the function. However, the manual categorizing of some datasets into groups, helped to identify key words to be able to start the creation of dictionaries for future use, based on the metadata of the various data catalogues harvested.

This process is better suited when there are a lot of datasets to create or to update manually datasets. In our case, this was done for adding the datasets provided by the reference group providing challenges, by the consortium members, but also for the datasets scraped (National Road Traffic Data Portal of the Netherlands<sup>29</sup> or NDOV Loket<sup>30</sup>) or found in text files (such as the metadata of the Milano Geoportale<sup>31</sup> or the database of MobilityData<sup>32</sup>).

To add these datasets into CKAN via the API, a python toolbox and a csv metadata template file were created.

<sup>27</sup> stem

<sup>28</sup> "Stemmers remove morphological affixes from words, leaving only the word stem. This may cause, for example, that searching for "testing" or "tested" will show also results containing the word "test". (https://docs.ckan.org/en/latest/user-guide.html#managing-an-organization).

<sup>29</sup> https://opendata.ndw.nu/

<sup>30</sup> ndovloket.nl

<sup>31</sup> https://geoportale.comune.milano.it/sit/open-data/

<sup>32</sup> https://database.mobilitydata.org/

<b>Legend</b> M - Mandatory R - Recommended O - Optional MDL - To be added by the team of MobiDataLab  1 = required, but not repeatable 1..N = required and repeatable 0..1 = optional, but not repeatable 0..N = optional and repeatable										
Element	Element Definition	Element Occurrence	Field Name (Internal)	Visible Field Label ('Preferred Label')	Definition/help text	Data type	Characters per description field	Obligation	Default value	example 1
Name of the resource	A resource can be any file or link to a file containing useful data	1..N	name	Name	Name of the resource, e.g. "Population density 2011, CSV". Different resources in the dataset should have different names.	Text		M		Bike path section (shp)
Name of the dataset/package	Datasets are simply used to group related pieces of data. These can be then be found under a single url with a description an licensing information.	1..N	name_package	Name	Name of the dataset without spaces and capitals, words can be separated only by hyphens. Characters per description field : 100	Text(100)	100	M		bike-path-section
Title of the package	Name or title by which the dataset being described is known.	1..N	title	Dataset Title	which the dataset being described is known. A CKAN Dataset is a collection of data resources (such as files), together with a description and other information, at a fixed URL. Datasets are what users see when searching	Text		M	None	Bike path section
Organization name	Organizations act like publishing departments for datasets. This means that datasets can be pushed by and belong to a department instead of an individual user.	1..N	owner_org_name	Organization	An entity that brought into existence the dataset being described. Creators can be people, organizations and/or physical or virtual infrastructure (e.g., sensors, software).	Checkbox		M		eindhoven-municipality
Custom field	Collections of datasets	0..N	groups	Group	Refer to the groups in the "Groups" tab of this file, if any of these groups seem to feet the category of your dataset information. More than one can be choosed just add a comma between both groups.	Text		R		mobility-transport

Figure 45: Metadata file elements explanation

M	M	M	M	R	R	M
Name of the resource	Name of the dataset/package	Title of the package	Organization name	Group	Group ID	Description of the resource
name	name_package	title_package	owner_org_name	groups	group_id	description
Bicycle lanes Eindhoven (cgp)	bicycle-lanes	Bicycle lanes	eindhoven-municipality	journey-planning	b2c45cca-f6a9-423d-8780-92c45fdf93a3	Bicycle lanes Eindhoven
Bicycle lanes Eindhoven (dbf)	bicycle-lanes	Bicycle lanes	eindhoven-municipality	mobility-transport	15d0db30-aec9-46c8-b2ec-8d534769d1980	Bicycle lanes Eindhoven
Bike count data	bike-count-data	Bike count data	eindhoven-municipality	micro-mobility	c851e2eb-d1f7-49d6-ab41-953cb44ac3f	Yearly bike count data on main cycling
Bike count data	bike-count-data	Bike count data	eindhoven-municipality	micro-mobility	c851e2eb-d1f7-49d6-ab41-953cb44ac3f	Yearly bike count data on main cycling
Bike path section (shp)	bike-path-section	Bike path section	eindhoven-municipality	mobility-transport	15d0db30-aec9-46c8-b2ec-8d534769d1980	Bike path section
Main cycle routes	main-cycle-routes	Main cycle routes	eindhoven-municipality	micro-mobility	c851e2eb-d1f7-49d6-ab41-953cb44ac3f	Main cycle routes

Figure 46: Metadata file elements example



```

CKAN_CTW 133 #Create multiple new packages
134 def create_multiple_packages_from_csv(filepath: str):
135     package_data = pd.read_csv(filepath, squeeze=True).to_dict(orient='records')
136
137     for item in package_data:
138         if _DEBUG:
139             print('Line handling : ' + str(item))
140
141         result = create_package(item['name_package'], item['title_package'], item['license_package_id'],
142                                item['url_package'], # item['tags_package'],
143                                item['country'], item['region'],
144                                item['municipality'], item['metadata_modified'], item['group_id'], item['group_name'],
145                                item['owner_org_name'], item['notes_package'])
146
147         if result.json()['success'] == True:
148             PACKAGE_ID = result.json()['result']['id']
149             print('Package ' + item['name_package'] + ' created with the ID: ' + PACKAGE_ID)
150         else:
151             print(result.json())
152             print('Error, package ' + item['name_package'] + ' not created ')
153             print('Reason : ' + result.json()['error']['name'][0])
154
155 #Create new resource
156 def add_resource(_id_package: str, _name_package: str, _name: str, _description: str, _url: str, _owner_org_name: str,
157                 _format: str, _groups: str):
158     result = requests.post(CKAN_ADRESS['url'] + '/api/action/resource_create',
159                            json={"package_id": _id_package,
160                                  "package_name": _name_package,
161                                  "name": _name,
162                                  "description": _description,

```

Figure 47: Example of use of the metadata file on the CKAN API

## MobiDataLab consortium

The consortium of MobiDataLab consists of 10 partners with multidisciplinary and complementary competencies. This includes leading universities, networks and industry sector specialists.



[@MobiDataLab](https://twitter.com/MobiDataLab)

#MobiDataLab



<https://www.linkedin.com/company/mobidatalab>

For further information please visit [www.mobidatalab.eu](http://www.mobidatalab.eu)



MobiDataLab is co-funded by the EU under the H2020 Research and Innovation Programme (grant agreement No 101006879).

The content of this document reflects solely the views of its authors. The European Commission is not liable for any use that may be made of the information contained therein. The MobiDataLab consortium members shall have no liability for damages of any kind that may result from the use of these materials.