



CUTLER - Coastal Urban developmentT through the LEenses of Resiliency

Project Title: CUTLER - Coastal Urban developmentT through the LEenses of Resiliency
Contract No: 770469 - CUTLER
Instrument: Research and Innovation Action
Thematic Priority: H2020-SC6-CO-CREATION-2016-2017 / H2020-SC6-COCREATION-2017
Start of project: 1 January 2018
Duration: 36 months

Deliverable No: D3.3

Final version of the data collection, management & protection framework integrated within CUTLER architecture

Due date of deliverable: 30 April 2019
Actual submission date: 24 May 2019
Version: 1.0
Main Authors: Ekaterina Gilman (UOULU), Panos Kostakos (UOULU), Marta Cortés (UOULU), Hassan Mehmood (UOULU), Andrew Byrne (DELL), Pierre Dewitte (KUL), Aleksandra Kuczerawy (KUL), Stavros Tekes (DRAXIS)



Project funded by the European Community under the H2020 Programme for Research and Innovation.



Project ref. number	770469
Project title	CUTLER - Coastal Urban development through the Lenses of Resiliency

Deliverable title	Final version of the data collection, management & protection framework integrated within CUTLER architecture
Deliverable number	D3.3
Deliverable version	1.0
Previous version(s)	-
Contractual date of delivery	30 April 2019
Actual date of delivery	24 May 2019
Deliverable filename	CUTLER_D3.3_final
Nature of deliverable	Demonstrator
Dissemination level	Public
Number of pages	93
Workpackage	WP3
Task(s)	T3.1, T3.2, T3.3
Partner responsible	UOULU
Author(s)	Ekaterina Gilman (UOULU), Panos Kostakos (UOULU), Marta Cortés (UOULU), Hassan Mehmood (UOULU), Jukka Riekkilä (UOULU), Andrew Byrne (DELL), Katerina Valta (DRAXIS), Stavros Tekes (DRAXIS), Pierre Dewitte (KUL), Aleksandra Kuczerawy (KUL), Chandan Kumar (UNIKO), Jun Sun (UNIKO), Steffen Staab (UNIKO), Ioannis Pragidis (DUTH), Panagiotis Tsintzos (DUTH), Georgios Geronikolaou (DUTH), Ioannis Chantas (CERTH), Yiannis Kompatsiaris (CERTH), Georgios Papastergios (THESS), Paraskevi Tzoumaka (THESS), Apostolos Kelessis (THESS), Petros Papafilis (THESS), Christantonis Charistes (THESS), Vasileios Kontos (THESS), Manolis Mimitidis (THESS), Charalampos Chatzis (THESS), Konstantinos Doudouliakis (THESS), Yiannis Kompatsiaris (CERTH), Salih Sandal (ANTALYA), Ozlem Alpaslan (ANTALYA), M. Serdar Yümlü (SAMPAS),

	İbrahim Acar (SAMPAS), Caner Tosunoğlu (SAMPAS), Rebecca Beeckman (ANTWERP), Ronny Van Looveren (ANTWERP), Philip Leroux (IMEC), Darragh O'Suilleabhain (CORK), Maire Daly (CORK), Elaine Walsh (CORK), Anthony O'Reilly (CORK), Kieran Thornton (BLP), Noreen O'Brien (BLP)
Editor	Ekaterina Gilman (UOULU)
EC Project Officer	Giorgio Costantino

Abstract	This report elaborates further on the output of D3.2 and integrates solutions developed so far into the actual cloud infrastructure of CUTLER platform. Namely, it: i) provides an update on the data sources used for the pilots; moreover an assessment of the legal requirements of these data sources is conducted ; ii) discusses further data pre-processing and privacy issues; iii) informs on porting the Hadoop crawlers to the cloud infrastructure of CUTLER platform; iv) provides detailed information about new crawlers; as well as v) discusses challenges and future work.
Keywords	Big data, data sources, environmental data, social data, economic data, data collection, crawling, cleaning, harmonization, interoperability, anonymization, encryption, legal requirements

Copyright

© Copyright 2019 CUTLER Consortium

This document may not be copied, reproduced, or modified in whole or in part for any purpose without written permission from the CUTLER Consortium. In addition to such written permission to copy, reproduce, or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.

All rights reserved.

Deliverable history

Version	Date	Reason	Revised by
0.1	08/03/2019	Initial version	Ekaterina Gilman, Panos Kostakos, Marta Cortes, Hassan Mehmood
0.2	10/05/2019	Contribution from partners	Ekaterina Gilman, Panos Kostakos, Marta Cortes, Hassan Mehmood, Andrew Byrne, Pierre Dewitte, Aleksandra Kuczerawy, Stavros Tekes, Katerina Valta
0.3	15/05/2019	Pre-final version ready for internal review	Ekaterina Gilman, Panos Kostakos, Marta Cortes, Hassan Mehmood, Andrew Byrne, Pierre Dewitte, Aleksandra Kuczerawy, Stavros Tekes, Katerina Valta
1.0	24/05/2019	Final version ready for submission	Ekaterina Gilman, Panos Kostakos, Marta Cortes, Hassan Mehmood, Filareti Tsalakanidou, Rebecca Beeckman, Andrew Byrne, Aleksandra Kuczerawy

List of abbreviations and acronyms

Abbreviation	Meaning
ABAC	Attribute Based Access Control
API	Application Programming Interface
CDH	Cloudera's Distribution including Apache Hadoop
EU	European Union
GDP	Gross Domestic Product
GDPR	General Data Protection Regulation
GeoJSON	Geospatial data interchange format based on JavaScript Object Notation (JSON)
GVA	Gross Value Added
HDFS	Hadoop Distributed File System
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
JSON-stat	simple lightweight JSON format for data dissemination
KML	Keyhole Markup Language
NDJSON	Newline Delimited JSON
NLP	Natural Language Processing
PII	Personally Identifiable Information
RBAC	Role Based Access Control
REST	Representational State Transfer
TSV	Tab-separated values
URL	Uniform Resource Locator
VPN	Virtual Private Network
WMS	Web Map Service
XML	Extensible Markup Language

Table of Contents

1. Executive summary	9
2. Introduction.....	10
3. Data used for the final version of the data collection and processing framework ...	11
3.1 Summary of the data sources used	11
3.2 General description of the data	14
3.2.1 Time and location data.....	14
3.2.2 Environmental data	18
3.2.3 Economic data	25
3.2.4 Social data	27
3.2.5 Miscellaneous data	28
4. Final version of the framework for data collection and pre-processing	29
4.1 Overall description	29
4.2 Data collection	30
4.3 Data pre-processing.....	30
4.3.1 Cleaning.....	30
4.3.1.1 Handling pattern violations.....	30
4.3.1.2 Handling rule-violations, duplicates, outliers.....	31
4.3.2 Harmonization.....	31
4.3.2.1 Time information	31
4.3.2.2 Geographical information	31
4.3.2.3 Environmental information.....	31
4.3.2.4 Economic information.....	32
4.3.2.5 Social information	33
4.3.2.6 Miscellaneous information.....	33
4.3.3 Interoperability	34
4.3.3.1 Time information	34
4.3.3.2 Geographical information	34
4.3.3.3 Environmental information.....	34
4.3.3.4 Economic information.....	34
4.3.3.5 Social information	35
4.3.3.6 Miscellaneous information.....	35
4.4 Data privacy.....	35
4.4.1 Data & Information Security Governance	36
4.4.2 Encryption.....	38
4.4.3 Anonymization	38

4.5	From the sandbox to the CUTLER cloud.....	39
4.5.1	Manual data ingestion	40
4.5.2	Automated data ingestion.....	40
4.5.2.1	One-time data acquisition	40
4.5.2.2	Scheduled data acquisition	41
4.5.2.3	Stream data acquisition.....	42
4.5.3	Hadoop to/from Elasticsearch	42
5.	Software description	45
5.1	Template for software description	45
5.2	Environmental Data.....	47
5.3	Economic Data.....	53
5.4	Social Data.....	54
6.	Challenges and future work	55
7.	Conclusions.....	56
8.	Appendix 1 – Data sources description.....	57
8.1	Template for data description.....	57
8.2	Environmental data	58
8.3	Economic data	80
8.4	Social data	84
9.	Appendix 2 - Assessment of confirmed data sources by DS_ID	85
9.1	Data sources for Antalya.....	85
9.2	Data sources for Antwerp.....	87
9.3	Data sources for Cork	88
9.4	Data sources for Thessaloniki	90
9.5	Data sources for all city pilots.....	91
10.	References	93

1. Executive summary

Deliverable D3.3 elaborates further on the output of D3.2 “*First version of the framework for the collection, cleaning, integration & anonymization of big data*” [1] and integrates the solutions presented there into the actual CUTLER cloud infrastructure [2].

More specifically, this deliverable ports the solutions from the first version of the CUTLER data processing framework, covering the steps required to collect, store, and pre-process the data, from a local sandbox cluster to the actual CUTLER cloud infrastructure, so that the data is available for further analysis and visualization, supported by WP4, WP5, and WP6. This deliverable is a software and test report; hence, we do not present new solutions here, but emphasis is placed on porting the solutions presented in D3.2 into the cloud. For a review of state-of-the-art methods, please refer to D3.1 “*Requirements for data crawling, integration and anonymization*” [3]. An updated version of this deliverable, including new data sources and data collection, storage, and pre-processing methods will be delivered in M30 of the project.

The deliverable provides a general summary of the data sources used in the four city pilots (Thessaloniki, Antalya, Antwerp, and Cork) that were ported into the CUTLER cloud infrastructure. Appendices provide detailed information about new data sources not previously included in D3.2 [1]. Moreover, the data sources have been qualified according to the legal taxonomy developed in D1.1 and its annexes [4]. Besides, the corresponding legal requirements were elicited following the assessment made in D1.2 [5].

Furthermore, we provide an update on data privacy issues. In addition, we present the software developed to collect and pre-process the new data sources. The software presented in this deliverable is available in the CUTLER GitHub repository. We also report on the porting of these solutions into the CUTLER cloud infrastructure.

Finally, we discuss key challenges and future work.

2. Introduction

CUTLER is focused on supporting evidence-based decision-making enabled by big data and it builds its research based on four city pilots (Thessaloniki, Antwerp, Antalya, and Cork) [6]. To achieve its goal, CUTLER utilizes Environmental, Economic, and Social data available from diverse data sources, ranging from deployed sensors, authority reports, open data repositories, specialized databases, and user-contributed content [3][1]. To be able to integrate all these data sources, CUTLER needs to have a data management platform, enabling data collection, pre-preprocessing, analysis, and presentation of results to policy makers.

Deliverable D3.3 “*Final version of the data collection, management & protection framework integrated within CUTLER architecture*” focuses on prototyping the solutions described in D3.2 “*First version of the framework for the collection, cleaning, integration & anonymization of big data*” and the crawlers developed for new data sources into the CUTLER cloud infrastructure so as to support data access and analysis by WP4, WP5, and WP6. The corresponding software is available at the CUTLER GitHub repository [7].

The deliverable is organized as follows:

Section 3 provides an update on data sources confirmed so far by pilot cities and corresponding leaders of WP4, WP5, and WP6. Moreover, an initial legal assessment of these data sources is provided. This is an initial step; further inspection of the legal requirements for the data sources will be conducted in the context of D1.4.

Guided by D3.2, section 3 also provides an update on data harmonization, cleaning and interoperability

Section 4 focuses on the porting of the solutions presented in D3.2 into the CUTLER cloud infrastructure.

Section 5 presents the crawlers developed for new data sources.

Section 6 summarizes the challenges faced and possible future work for improving the data collection procedures in the next development phase of CUTLER.

Section 7 concludes the deliverable.

Appendix - 1 (Section 8) describes the new data sources used for the different city pilots.

Appendix - 2 (Section 9) provides an assessment of the data sources from the legal requirements point of view.

3. Data used for the final version of the data collection and processing framework

CUTLER is a large project with four city pilots: Thessaloniki, Antalya, Antwerp, and Cork, providing Environmental, Economic, and Social data to help in achieving project goals.

The pilot cases are elaborated in D9.1 “*Report on pilot preparations and pilot execution plan*” [6] while the mechanisms to collect and describe the data are reported in D3.1 “*Requirements for data crawling, integration and anonymization*” [3]. In D3.2 “*First version of the framework for the collection, cleaning, integration & anonymization of big data*”, we identified the data sources to be used in the first version of the data collection framework. However, the identification and clarification of data sources is a continuous process. Thus, in this deliverable, we present an update of D3.2, including new data sources. Additional changes are still inevitable and will be reported in D3.5 in M30.

This section provides an updated version of the data to be used by the CUTLER platform during the first pilot phase. In addition, we present an updated list of the various attribute/variable names, units and/or value representations used among the various data sources, grouped in five categories, i.e. time and location, environmental, economic, social and miscellaneous data. The initial list can be found in D3.2.

3.1 Summary of the data sources used

The data sources described in D3.2 were further examined to confirm which of them will be used by the first version of the CUTLER platform. New data sources have been also identified by pilot partners and WP4,5,6 leaders. Based on this investigation, we summarise the confirmed data sources in Table 3.1-1 [8]. Corresponding details are also updated in the project wiki [9]. Each data source has a unique number (**DS_ID**). Moreover, each data source can supply different data; therefore, there is also unique data number (**D_ID**). The **Reference** column points to the data source description either in D3.2 (meaning that the data source was already described) or in this deliverable, D3.3 (meaning that this is a new data source). We describe additional data sources in full details (not described previously in D3.2) in Appendix 1 (Section 8).

Table 3.1-1: Confirmed data sources for final version of the data collection framework

DS_ID	D_ID	Reference
THESS_ENV_CITYOFTHESS_DAILY YEARLY	THESS_ENV_CITYOFTHESS_DAILY YEARLY_1 ... THESS_ENV_CITYOFTHESS_DAILY YEARLY_6	D3.2 8.2-14
THESS_ENV_IMET_SPEED_15MIN	THESS_ENV_IMET_SPEED_15MIN_1 ... THESS_ENV_IMET_SPEED_15MIN_4	8.2-21
THESS_SOC_IMC_MONTHLY	THESS_SOC_IMC_MONTHLY_1	
THESS_ECO_THESSALONIKI MUNICIPALITY_BUDGET	THESS_ECO_THESSALONIKI MUNICIPALITY_BUDGET_1_1, THESS_ECO_THESSALONIKI MUNICIPALITY_BUDGET_1_2	D3.2 8.3-1, D3.2 8.3-2
THESS_ECO_THESSALONIKI PARKING_SCANS	THESS_ECO_THESSALONIKI PARKING_SCANS_1	D3.2 8.3-11
THESS_ECO_THESSALONIKI PARKING_ECONOMIC_DATA	THESS_ECO_THESSALONIKI PARKING_ECONOMIC_DATA_1	D3.2 8.3-12
antalya_env_cityofantalya_perminute	antalya_env_cityofantalya_perminute_1	D3.2 8.2-20

antalya_env_cityofantalya2_monthly	antalya_env_cityofantalya2_monthly_1... antalya_env_cityofantalya2_monthly_8	8.2-22
anta_env_waterqualityflow_cityofantalya_monthly	anta_env_waterqualityflow_cityofantalya_monthly_1	8.2-23
anta_eco_cityofantalya_otopark_monthly	anta_eco_cityofantalya_otopark_monthly	8.3-16
anta_eco_cityofantalya_visitorticket_monthly	anta_eco_cityofantalya_visitorticket_monthly	8.3-17
anta_eco_cityofantalya_ShopsRentEarn_year	anta_eco_cityofantalya_ShopsRentEarn_year	8.3-22
anta_eco_cityofantalya_operationemployeessalary_monthly	antalya_econ_cityofantalya_operationemployeessalary_monthly	8.3-18
anta_eco_cityofantalya_generalelectricitybill_monthly	anta_eco_cityofantalya_generalelectricitybill_monthly	8.3-19
anta_eco_cityofantalya_waterpumps_monthly	anta_eco_cityofantalya_waterpumps_monthly	8.3-20
anta_eco_cityofantalya_cityzonepublictransportationpasengernumber_monthly	anta_eco_cityofantalya_cityzonepublictransportationpasengernumber_monthly	8.3-21
ant_env_cityofant_histprec	ant_env_cityofant_prehist_P1... ant_env_cityofant_prehist_P12	D3.2 8.2-15
ant_env_cityofant_gwl	ant_env_cityofant_gwl_1-460	D3.2 8.2-16
ant_env_imec_prec2018	ant_env_imec_prec2018_P1... ant_env_imec_prec2018_P4	D3.2 8.2-17
ant_env_imec_openwater	ant_env_imec_openwater_H1... ant_env_imec_openwater_H6	D3.2 8.2-18
ant_env_imec_sewer2018	ant_env_imec_sewer2018_D1... ant_env_imec_sewer2018_D3	D3.2 8.2-19
ant_env_cityofant_maps	ant_env_cityofant_maps_X	8.2-37
CORK_ENV_OPW_WL_15min	CORK_ENV_OPW_WL_15min	8.2-36
CORK_ENV_MET_W_HOURLY	CORK_ENV_MET_W_HOURLY	8.2-35
CORK_ENV_EPA_SAC_2015	CORK_ENV_EPA_SAC_2015	8.2-24
CORK_ENV_EPA_NHA_2012	CORK_ENV_EPA_NHA_2012	8.2-25
CORK_ENV_EPA_SPA_2015	CORK_ENV_EPA_SPA_2015	8.2-26
CORK_ENV_EPA_GWWFD_20102015	CORK_ENV_EPA_GWWFD_20102015	8.2-27
CORK_ENV_EPA_LWFD_20102015	CORK_ENV_EPA_LWFD_20102015	8.2-28
CORK_ENV_EPA_RWFD_20102015	CORK_ENV_EPA_RWFD_20102015	8.2-29
CORK_ENV_EPA_CWFD_20102015	CORK_ENV_EPA_CWFD_20102015	8.2-30
CORK_ENV_EPA_TWFD_20102015	CORK_ENV_EPA_TWFD_20102015	8.2-31
CORK_ENV_OPW_FLOODS_2016	CORK_ENV_OPW_FLOODS_2016	8.2-32
CORK_ENV_CCC3_LAND_2014	CORK_ENV_CCC3_LAND_2014	8.2-33
CORK_ECO_VISITORS_DAILY	CORK_ECO_VISITORS_DAILY	8.3-23

PIALL-SOC_TWITTER	PIALL-SOC_TWITTER	D3.2 8.4-3
PIALL-SOC_NEWS	PIALL-SOC_NEWS	D3.2 8.4-4
PIALL-SOC_GDELT	PIALL-SOC_GDELT	D3.2 8.4-2
PITHESS_ECO_OECD_REGIONLABOUR PIANTW_ECO_OECD_REGIONLABOUR PICORK_ECO_OECD_REGIONLABOUR PIANTAL_ECO_OECD_REGIONLABOUR	PITHESS_ECO_OECD_REGIONLABOUR_EMPPLARES PIANTW_ECO_OECD_REGIONLABOUR_EMPPLARES PICORK_ECO_OECD_REGIONLABOUR_EMPPLARES PIANTAL_ECO_OECD_REGIONLABOUR_EMPPLARES	D3.2 8.3-3
PITHESS_ECO_OECD_REGIONLABOUR PIANTW_ECO_OECD_REGIONLABOUR PICORK_ECO_OECD_REGIONLABOUR PIANTAL_ECO_OECD_REGIONLABOUR	PITHESS_ECO_OECD_REGIONLABOUR_LFPARTR PIANTW_ECO_OECD_REGIONLABOUR_LFPARTR PICORK_ECO_OECD_REGIONLABOUR_LFPARTR PIANTAL_ECO_OECD_REGIONLABOUR_LFPARTR	D3.2 8.3-4
PITHESS_ECO_OECD_REGIONLABOUR PIANTW_ECO_OECD_REGIONLABOUR PICORK_ECO_OECD_REGIONLABOUR PIANTAL_ECO_OECD_REGIONLABOUR_	PITHESS_ECO_OECD_REGIONLABOUR_UNEMREG PIANTW_ECO_OECD_REGIONLABOUR_UNEMREG PICORK_ECO_OECD_REGIONLABOUR_UNEMREG PIANTAL_ECO_OECD_REGIONLABOUR_UNEMREG	D3.2-8.3-5
PITHESS_ECO_OECD_REGIONECO PIANTW_ECO_OECD_REGIONECO PICORK_ECO_OECD_REGIONECO PIANTAL_ECO_OECD_REGIONECO	PITHESS_ECO_OECD_REGIONECO_R EGG DPTL2 PIANTW_ECO_OECD_REGIONECO_R EGG DPTL2 PICORK_ECO_OECD_REGIONECO_R EGG DPTL2 PIANTAL_ECO_OECD_REGIONECO_R EGGDPTL2	D3.2 8.3-6
PITHESS_ECO_OECD_REGIONECO PIANTW_ECO_OECD_REGIONECO PICORK_ECO_OECD_REGIONECO PIANTAL_ECO_OECD_REGIONECO	PITHESS_ECO_OECD_REGIONECO_G DP LT3 PIANTW_ECO_OECD_REGIONECO_G DP LT3 PICORK_ECO_OECD_REGIONECO_G DP LT3 PIANTAL_ECO_OECD_REGIONECO_G DP LT3	D3.2 8.3-7
PITHESS_ECO_OECD_REGIONECO PIANTW_ECO_OECD_REGIONECO PICORK_ECO_OECD_REGIONECO PIANTAL_ECO_OECD_REGIONECO	PITHESS_ECO_OECD_REGIONECO_ REGEMINDU PIANTW_ECO_OECD_REGIONECO_R EGEMINDU PICORK_ECO_OECD_REGIONECO_R EGEMINDU PIANTAL_ECO_OECD_REGIONECO_R EGEMINDU	D3.2 8.3-8
PITHESS_ECO_OECD_REGIONECO PIANTW_ECO_OECD_REGIONECO PICORK_ECO_OECD_REGIONECO	PITHESS_ECO_OECD_REGIONECO_ REGGVAWORKER PIANTW_ECO_OECD_REGIONECO_ REGGVAWORKER	D3.2 8.3-9

PIANTAL_ECO_OECD_REGIONECO	PICORK_ECO_OECD_REGIONECO_REGGVAWORKER PIANTAL_ECO_OECD_REGIONECO_REGGVAWORKER	
PITHESS_ECO_OECD_REGIONECO PIANTW_ECO_OECD_REGIONECO PICORK_ECO_OECD_REGIONECO PIANTAL_ECO_OECD_REGIONECO	PITHESS_ECO_OECD_REGIONECO_REGINCPC PIANTW_ECO_OECD_REGIONECO_REGINCPC PICORK_ECO_OECD_REGIONECO_REGINCPC PIANTAL_ECO_OECD_REGIONECO_REGINCPC	D3.2 8.3-10
EUROSTAT_REG	EUROSTAT_REG_ID-1... EUROSTAT_REG_ID-7	D3.2 8.3-13
EUROSTAT_REG_TYPE	EUROSTAT_REG_TYPE_ID-1... EUROSTAT_REG_TYPE_ID-3	D3.2 8.3-14
EUROSTAT_URB	EUROSTAT_URB_ID-1	D3.2 8.3-15

Moreover, KUL checked the list of confirmed data sources and linked them with elements of the legal taxonomy presented in D1.1 “*Legal Taxonomy of Datasets*” [4], providing also the legal requirements for them as explained in D1.2 “*Legal requirements*” [5]. The legal assessment for the confirmed data sources is summarized in Appendix 2 (Section 9). The actual taxonomy is provided in D1.1 while legal requirements are summarised in D1.2.

3.2 General description of the data

This section updates the information on the various attribute/variable names, units and/or value representations used among the various data sources (please refer to D3.2 for details).

3.2.1 Time and location data

Table 3.2.1-1 describes the time and location information. Numbers in the brackets link to the actual data source descriptions in Appendix 1 or D3.2 where the corresponding variable is met.

Table 3.2.1-1: Time and location data

Parameter of interest	Naming among data sources	Units or value representation among data sources
Time and Date	datetime (D3.2 8.2-1) DateTime (D3.2 8.2-2) time (D3.2 8.2-3, D3.2 8.2-4, D3.2 8.2-5, D3.2 8.2-6, D3.2 8.2-7, D3.2 8.2-8, D3.2 8.2-9, D3.2 8.2-17,18,19, D3.2 8.3-13) Timestamp (D3.2 8.2-10) (8.2-21) TIME (D3.2 8.3-	YYYY-MM-DD HH:mm:ss+HH:mm (D3.2 8.2-1) YYYY-MM-DDHH:mm:ss (D3.2 8.2-2) YYYY-MM-DDTHH:mm:ssZ (D3.2 8.2-3,4, 5, 6, 7, 8, D3.2 8.3-13,14,15, D3.2 8.4-4) “LATEST WEATHER REPORTS ON DD-Month-YYYY FOR HH:mm” (D3.2 8.2-9)

Parameter of interest	Naming among data sources	Units or value representation among data sources
	<p>3, 4, 5, 6, 7, 8, 9, 10)</p> <p>Ημερο - μην (D3.2 8.2.12)</p> <p>created (D3.2 8.4-1)</p> <p>updated (D3.2 8.4-1)</p> <p>Date (D3.2 8.2-14) (8.2-36) (8.3-21) (8.3-23)</p> <p>Tarih (D3.2 8.2-15)</p> <p>SQLDATE (D3.2 8.4-2)</p> <p>MonthYear (D3.2 8.4-2)</p> <p>Year (D3.2 8.4-2)</p> <p>FractionDate (D3.2 8.4-2)</p> <p>DATEADDED (D3.2 8.4-2)</p> <p>created_at (D3.2 8.4-3)</p> <p>publishedAt (D3.2 8.4-4)</p> <p>Detection Time (D3.2 8.3-11)</p> <p>updated (D3.2 8.3-13,14,15)</p> <p>TARİH (8.2-22)</p> <p>Sonuç Tarihi (8.2-22)</p> <p>DATE (8.2-23)</p> <p>DateChange (8.2-27,28,29,30,31)</p> <p>INS_WHEN (8.2-27,30,31)</p> <p>date (8.2-35)</p> <p>No_field_name (Year) (8.3-16,17,18,19,20,22)</p> <p>No_field_name (Month) (8.3-16,17,18,19,20)</p>	<p>DD-MM-YYYY HH:mm (D3.2 8.2-10)</p> <p>YYYY (D3.2 8.3-3,4,5,6,7,8,9,10,13)</p> <p>DD.MM.YY (D3.2 8.2-12)</p> <p>YYYY-MM-DD HH:mm:ss (D3.2 8.4-1, D3.2 8.2-13, D3.2 8.3-11)</p> <p>YYYY-MM-DD (D3.2 8.2-14) (8.2-27,28,29,30,31)</p> <p>DD.MM.YY HH:mm (D3.2 8.2-15)</p> <p>float (UCT) (D3.2 8.2-17,18,19)</p> <p>float (8.2-22)</p> <p>YYYYMMDD (D3.2 8.4-2)</p> <p>YYYYMM (D3.2 8.4-2)</p> <p>YYYY (D3.2 8.4-2) (8.3-16,17,18,19,20,22)</p> <p>YYYY.FFFF, where FFFF is the percentage of the year completed by that day (D3.2 8.4-2)</p> <p>YYYYMMDDHHMMSS format in the UTC timezone (D3.2 8.4-2)</p> <p>Day(3 letters) Month (3 letters) DD HH:mm:ss ±HHmm YYYY, e.g.</p> <p>"Wed Aug 27 13:08:45 +0000 2008" (D3.2 8.4-3)</p> <p>YYYY-MM-dd HH:mm:ss.SSS (8.2-21)</p> <p>mm/dd/YYYY (8.2-23) (8.3-21)</p> <p>DD-MMM-YYYY HH:mm (Month: 3 letters) (8.2-35)</p> <p>YYYY/MM/DD HH:mm:ss (8.2-36)</p> <p>DD-MMM-YYYY (Month: 3 letters) (8.3-23)</p> <p>Month name in Turkish) (8.3-16,17,18,19,20)</p>
Location	<p>latitude, longitude (D3.2 8.2-1, D3.2 8.2-3, 4, 5, 6, 7, 8, D3.2 8.4-1) altitude (D3.2 8.2-8)</p> <p>lat, lon (8.2-23)</p> <p>LAT, LON (8.2-27,28,30,31)</p>	<p>degrees north and east (D3.2 8.2-1, D3.2 8.2-3, 4, 5, 6, 7, D3.2 8.4-1, D3.2 8.2-14, D3.2 8.3-11) (8.2-23,27,28,30,31)</p> <p>Name of city (D3.2 8.2-9)</p> <p>"degrees north, degrees east"</p>

Parameter of interest	Naming among data sources	Units or value representation among data sources
	<p>location (D3.2 8.2-9, D3.2 8.4-1)</p> <p>address (D3.2 8.4-1)</p> <p>LOCATION (D3.2 8.2-13)</p> <p>X, Y (D3.2 8.2-13) (8.2-37)</p> <p>TL (D3.2 8.3-3,4,5,6,7,8,9,10)</p> <p>REG_ID (D3.2 8.3-3,4,5,6,7,8,9,10)</p> <p>X-coordinate (Lambert72), Y-coordinate (Lambert72) (D3.2 8.2-14)</p> <p>X (WGS84), Y (WGS84) (D3.2 8.2-14)</p> <p>Location (D3.2 8.2-14)</p> <p>geohash (D3.2 8.2-17,18,19)</p> <p>Διεύθυνση (D3.2 8.3-12)</p> <p>Actor1Code (D3.2 8.4-2)</p> <p>Actor1Name (D3.2 8.4-2)</p> <p>Actor1CountryCode (D3.2 8.4-2)</p> <p>Actor1Geo_Type (D3.2 8.4-2)</p> <p>Actor1Geo_Fullname (D3.2 8.4-2)</p> <p>Actor1Geo_CountryCode (D3.2 8.4-2)</p> <p>Actor1Geo_Lat (D3.2 8.4-2)</p> <p>Actor1Geo_Long (D3.2 8.4-2)</p> <p>coordinates (D3.2 8.4-3)</p> <p>places (D3.2 8.4-3)</p> <p>Vehicle Center Latitude, Vehicle Center Longitude (D3.2 8.3-11)</p> <p>geo (D3.2 8.3-13)</p> <p>metroreg (D3.2 8.3-14)</p> <p>cities (D3.2 8.3-15)</p> <p>PathID (8.2-21)</p> <p>Name (8.2-21)</p> <p>ZONE (8.2-23)</p> <p>Coordinates (geometry.coordinates) (8.2-24,25,26,27,28,29,30,31,32,33,34,37)</p> <p>COUNTY (8.2-24,25,26)</p>	<p>(8.4-1)</p> <p>Address (D3.2 8.4-1, D3.2 8.3-12)</p> <p>Name (D3.2 8.2-13, D3.2 8.2-14, D3.2 8.4-2) (8.2-24,25,26)</p> <p>Belgian Lambert 72, EPSG:31370 (D3.2 8.2-13, D3.2 8.2-14)</p> <p>string (Geohash value) (D3.2 8.2-17,18,19)</p> <p>String representation (Digit or Digit_2Letters, e.g. 3_IN) (D3.2 8.3-3,4,5,6,7,8,9,10)</p> <p>Alphanumeric code, representing geographical unit (D3.2 8.3-3,4,5,6,7,8,9,10)</p> <p>country or capital/major city name (D3.2 8.4-2)</p> <p>3-character CAMEO code for the country affiliation of Actor1 (D3.2 8.4-2)</p> <p>integer (D3.2 8.4-2, geographic resolution)</p> <p>String (D3.2 8.4-2, 2-character FIPS10-4 country code for the location)</p> <p>coordinates JSON object, where inner coordinates array is formatted as geoJSON (longitude,latitude), e.g. "coordinates":</p> <pre>{ "coordinates": [-75.14310264, 40.05701649], "type":"Point" }</pre> <p>(D3.2 8.4-3)</p> <p>place JSON object {id, url, place_type, name, full_name, country_code, country, bounding_box, attributes} (D3.2 8.4-3)</p> <p>Code:String name in JSON object:</p> <pre>"geo":{"label":"geo", "category":{" "index":{"BE":0,(...)}, "label":{"BE":"Belgium",(...)}}</pre>

Parameter of interest	Naming among data sources	Units or value representation among data sources
	<p>Centroid_X (TM65/Irish Grid) , Centroid_Y (TM65/Irish Grid) (8.2-24)</p> <p>Easting, Northing (8.2-27,28,30,31)</p> <p>Hydrometri (Hydrometric Area) (8.2-28)</p> <p>SITECODE (8.2-24,25,26)</p> <p>REGION_ID (8.2-27,28,30,31)</p> <p>BASIN_CD (8.2-28,29)</p> <p>EU_CD (8.2-27,28,29,30,31)</p> <p>OBJECTID (8.2-24,25,26)</p> <p>N2k_Code (EU Site Code) (8.2-24)</p> <p>SITE_NAME (8.2-24,25,26)</p> <p>DIST_CD (District code) (8.2-27,28,29,30,31)</p> <p>EdenCode (8.2-27)</p> <p>EDENEntity (8.2-30,31)</p> <p>ENDENLACode(8.2-30,31)</p> <p>MS_CD (Member State Code) (8.2-27,28,29,30,31)</p> <p>NAME (8.2-29) (8.2-27,28,30,31)</p> <p>BasinSubCo</p> <p>SEG_CD (Lake segment code) (8.2-28)</p> <p>RBD (8.2-29)</p> <p>RBDcode (8.2-32)</p> <p>ID(8.2-33) (8.2-34)</p> <p>Sub_ID(8.2-33)</p>	<p>} (D3.2 8.3-13)</p> <p>Code:String name in JSON object</p> <p>"metroreg":{"label":"metroreg", "category":{"index":{"BE":0, (...)}, "label":{"BE":"Belgium", (...)}} (D3.2 8.3-14)</p> <p>Id and name (of path) (8.2-21)</p> <p>Code:String name in JSON object:</p> <p>"cities":{"label":"cities", "category":{"index":{"BE":0, (...)}, "label":{"BE":"Belgium", (...)}} } (D3.2 8.3-15)</p> <p>Float (8.2-21)</p> <p>Sampling point number and Name (8.2-23)</p> <p>Array of nx2 X,Y values in TM65/Irish Grid (8.2-24,25,26,27,28,29,30,31,32,33,34,37)</p> <p>Name of the County (8.2-24,25,26)</p> <p>M in TM65 / Irish Grid (8.2-24,27,28,30,31,37)</p> <p>Name of the hydrometric Area (8.2-28)</p> <p>Code: 6 digits (8.2-24,25,26)</p> <p>Code for ecoregion to which waterbody belongs: Integer (8.2-27,28,30,31)</p> <p>Code of the parent river basin: alphanumeric string ("IE_EA_159") (8.2-28,29)</p> <p>Unique code for waterbody at EU level: alphanumeric code ("IE_EA_07_178") (8.2-27,28,29,30,31)</p> <p>ID: int (8.2-24,25,26) (8.2-34)</p> <p>Code European for site: (IE0002327) alphanumeric string 9 elements (8.2-24)</p> <p>Name of the District (8.2-</p>

Parameter of interest	Naming among data sources	Units or value representation among data sources
		27,28,29,30,31) Eden monitoring system codes: 4 digits (8.2-27) (8.2-30,31) Code: Alphanumeric String ("EA_G_031") (8.2-27,28,29,30,31) name in alphanumeric string ("ATHBOY_010") (8.2-29) locally used name (8.2-27,28,30,31) River Basin Districts name: string (8.2-29) Code: String (8.2-32) ID: alphanumeric (8.2-33) (8.2-34)
Area	Catchment area (D3.2 8.2-10) type (geometry) (8.2-24,25,26,27,28,29,30,31,32,33,34,37) AreaHectar (8.2-27,28,30,31) AreaKm2 (8.2-27,28,29,30,31) HA (8.2-24,25,26) Shape_Area (8.2-24,25,26,29) LAEA_Area (8.2-24) Shape_STAr (8.2-27)	Kilometre square(km ²) (D3.2 8.2-10) (8.2-27,28,29,30,31) String ("polygon", MultiPolygon") (8.2-24,25,26,27,28,29,30,31,32,33,34,37) Hectarea (ha) (8.2-27,28,30,31) (8.2-24,25,26) square meters (8.2-24,25,26,29) (8.2-24) (8.2-27)
Length	Shape_Leng Shape_STLe	meters (8.2-24,25,26,29) (8.2-27)

3.2.2 Environmental data

Table 3.2.2-1 describes the environmental parameters of interest. Numbers in brackets link to the actual data source descriptions in Appendix 1 or D3.2 where the corresponding parameter is met.

Table 3.2.2-1: Environmental parameters of interest

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
1	Water level	sensor.ref=0001 (D3.2 8.2-1) Water_Level (D3.2 8.2-6, D3.2 8.2-8) Peil_cor2 (D3.2 8.2-16) Value (8.2-36)	metres(m) (D3.2 8.2-1, D3.2 8.2-6, D3.2 8.2-8) (8.2-36) mTAW (D3.2 8.2-16)
2	Predicted Tide	MeasuredTideHeight	mm (D3.2 8.2-2)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
	Height	(D3.2 8.2-2)	
3	Measured Tide Height	MeasuredTideHeight (D3.2 8.2-2)	mm (D3.2 8.2-2)
4	Significant wave height	significant_wave_height (D3.2 8.2-3)	m (D3.2 8.2-3)
5	Swell wave height	swell_wave_height (D3.2 8.2-3)	m (D3.2 8.2-3)
6	Mean wave direction	mean_wave_direction (D3.2 8.2-3) MeanWaveDirection (D3.2 8.2-7)	degrees (D3.2 8.2-3, D3.2 8.2-7)
7	Mean wave period	mean_wave_period (D3.2 8.2-3)	s (D3.2 8.2-3)
8	Wind speed	WindSpeed (D3.2 8.2-4, D3.2 8.2-7) Speed(Kts) (D3.2 8.2-9) Beaufort F3 (D3.2 8.2-13) Ruzgar Hizi(m/s) (D3.2 8.2-20)	kn (D3.2 8.2-4, D3.2 8.2-7) kts (D3.2 8.2-9, D3.2 8.2-13) String, e.g. light breeze (D3.2 8.2-13) m/s (D3.2 8.2-20)
9	Wind direction	WindDirection (D3.2 8.2-4, D3.2 8.2-7) Dir (D3.2 8.2-9) Wind Bearing (D3.2 8.2-13) Ruzgar Yönü(Derece) (D3.2 8.2-20) wddir(8.2-35)	degrees (D3.2 8.2-4, D3.2 8.2-7, D3.2 8.2-13, D3.2 8.2-20) cardinal direction (D3.2 8.2-9) Degrees (deg) (8.2-35)
10	Sea surface height	sea_surface_height (D3.2 8.2-5)	metres (m) (D3.2 8.2-5)
11	Sea surface temperature	sea_surface_temperature(D3.2 8.2-5)	Celsius (D3.2 8.2-5)
12	Sea bottom temperature	sea_bottom_temperature(D3.2 8.2-5)	Celsius (D3.2 8.2-5)
13	Sea surface salinity	sea_surface_salinity (D3.2 8.2-5)	Practical salinity unit (PSU) (D3.2 8.2-5)
14	Sea bottom salinity	sea_bottom_salinity (D3.2 8.2-5)	Practical salinity unit (PSU) (D3.2 8.2-5)
15	Sea surface x velocity	sea_surface_x_velocity (8.2-5)	Metre per second (m/s) (8.2-5)
16	Sea surface y velocity	sea_surface_y_velocity (D3.2 8.2-5)	Metre per second (m/s) (D3.2 8.2-5)
17	Sea bottom x velocity	sea_bottom_x_velocity (D3.2 8.2-5)	Metre per second (m/s) (D3.2 8.2-5)
18	Mixed layer depth	mixed_layer_depth (D3.2 8.2-5)	Metres (m) (D3.2 8.2-5)
19	Atmospheric pressure	AtmosphericPressure (D3.2 8.2-7) Barometer (D3.2 8.2-13) Pressure (D3.2 8.2-	Millibars (mb) (D3.2 8.2-7, D3.2 8.2-13) Hectopascals (hPa) (D3.2 8.2-9, D3.2 8.2-19) (mbar) (D3.2 8.2-20)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
		9) Hava Basinci(mbar) (D3.2 8.2-20) Value (D3.2 8.2-19)	
20	Gust	Gust (D3.2 8.2-7) Wind Speed(gust) (D3.2 8.2-13)	Knots (kn) (D3.2 8.2-7) Knots (kts) (D3.2 8.2-13)
21	Wave height	WaveHeight (D3.2 8.2-7)	Metres (m) (D3.2 8.2-7)
22	Wave period	WavePeriod (D3.2 8.2-7)	Seconds (s) (D3.2 8.2-7)
23	Hmax	Hmax (D3.2 8.2-7)	Metres (m) (D3.2 8.2-7)
24	Air temperature	AirTemperature(D3.2 8.2-7) Temp (D3.2 8.2-9) Temperature (D3.2 8.2-13) Hava Sicakligi(°C) (D3.2 8.2-20) sensor.ref=0002 (D3.2 8.2-1) temp (8.2-35)	Celsius (degree_C) (D3.2 8.2-1, D3.2 8.2-7, D3.2 8.2-9, D3.2 8.2-13, D3.2 8.2-20) (C) (8.2-35)
25	Dew point temperature	DewPoint (D3.2 8.2-7) Dew Point (D3.2 8.2-13) dewpt (8.2-35)	Celsius (degree_C) (D3.2 8.2-7, D3.2 8.2-13) (C) (8.2-35)
26	Sea temperature	SeaTemperature(D3.2 8.2-7)	Celsius (degree_C) (D3.2 8.2-7)
27	Salinity	salinity (D3.2 8.2-7) SALINITY(8.2-30,31)	Practical salinity unit (PSU) (D3.2 8.2-7) String ("E", "P") (8.2-30,31)
28	Relative humidity	RelativeHumidity (D3.2 8.2-7) Humidity (D3.2 8.2-9, D3.2 8.2-13) rhum (8.2-35)	percentage(D3.2 8.2-7, D3.2 8.2-9, D3.2 8.2-13) (8.2-35)
29	Water_Level_LAT	Water_Level_LAT (D3.2 8.2-8)	Metres (m) (D3.2 8.2-8)
30	Water_Level_OD_Malin	Water_Level_OD_Malin (D3.2 8.2-8)	Metres (m) (D3.2 8.2-8)
31	Rain / Precipitation	Rain (D3.2 8.2-9) P1,P2,P3... P12(D3.2 8.2-15) Value (D3.2 8.2-17) rain (8.2-35)	Millimetre (mm) (D3.2 8.2-9, D3.2 8.2-15, D3.2 8.2-17) (8.2-35)
32	Dawn	Dawn (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
33	Sunrise	Sunrise (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
		13)	
34	Moonrise	Moonrise (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
35	Dusk	Dusk (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
36	Sunset	Sunset (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
37	Moonset	Moonset (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
38	Length	Length (D3.2 8.2-13)	HH:mm (D3.2 8.2-13)
39	Phase	Phase (D3.2 8.2-13)	moon phases (D3.2 8.2-13)
40	Wind chill	Windchill (D3.2 8.2-13)	Celsius (D3.2 8.2-13)
41	Heat index	Heat Index (D3.2 8.2-13)	Celsius (D3.2 8.2-13)
42	Apparent temperature	Apparent Temperature (D3.2 8.2-13)	Celsius (D3.2 8.2-13)
43	Solar radiation	Solar Radiation (D3.2 8.2-13)	Watt per square metre (W/m ²) (D3.2 8.2-13)
44	Evapotranspiration Today	Evapotranspiration Today (D3.2 8.2-13)	Millimetre (mm) (D3.2 8.2-13)
45	Rainfall Today	Rainfall Today (D3.2 8.2-13)	Millimetre (mm) (D3.2 8.2-13)
46	Rainfall Rate	Rainfall Rate (D3.2 8.2-13)	Millimetre per hour (mm/hr) (D3.2 8.2-13)
47	Rainfall This Month	Rainfall This Month (D3.2 8.2-13)	Millimetre (mm) (D3.2 8.2-13)
48	Rainfall This Year	Rainfall This Year (D3.2 8.2-13)	Millimetre (mm) (D3.2 8.2-13)
49	Rainfall Last Hour	Rainfall Last Hour (D3.2 8.2-13)	Millimetre (mm) (D3.2 8.2-13)
50	Last Rainfall	Last Rainfall (D3.2 8.2-13)	YYYY-MM-DD HH:mm (D3.2 8.2-13)
51	Average wind speed	Wind Speed (avg) (D3.2 8.2-13)	Knots (kts) (D3.2 8.2-13)
52	Rising slowly	Rising slowly (D3.2 8.2-13)	Millibars per hour (mb/hr) (D3.2 8.2-13)
53	NO	NO (D3.2 8.2-14)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14)
54	NO ₂	NO ₂ (D3.2 8.2-14) (8.2-22)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14) (8.2-22)
55	O ₃	O ₃ (D3.2 8.2-14)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14)
56	PM ₁₀	PM ₁₀ (D3.2 8.2-14) PM ₁₀ (µg/m ³) (D3.2 8.2-20)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14, D3.2 8.2-20)
57	PM _{2.5}	PM _{2.5} (D3.2 8.2-14)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
58	CO	CO (D3.2 8.2-14)	milligrams per cubic meter (mg/m ³) (D3.2 8.2-14)
59	SO ₂	SO ₂ (D3.2 8.2-14) SO ₂ (µg/m ³) (D3.2 8.2-20)	micrograms per cubic meter (µg/m ³) (D3.2 8.2-14, D3.2 8.2-20)
60	Height	Height well (D3.2 8.2-16) Height ground level (D3.2 8.2-16) Value (D3.2 8.2-18)	mTAW (D3.2 8.2-16) mm (D3.2 8.2-18)
61	Remarks	Remarks (D3.2 8.2-16)	int (0) or string (D3.2 8.2-16)
62	Car Speed	Value (8.2-21)	Kilometres per hour (km/h) (8.2-21)
63	Number of Samples	Samples (8.2-21)	int (8.2-21)
64	Total coliform	Toplam Koliform (8.2-22)	Colony Forming Units per 100 millimeters (CFU/100ml) (8.2-22)
65	Fecal coliform	Fekal Koliform (8.2-22) Fecal coliform (8.2-22)	Colony Forming Units per 100 millimeters (CFU/100ml) (8.2-22,23)
66	Fecal Streptococcus	Fekal Streptokok (8.2-22) Fecal Streptococcus (8.2-23)	Colony Forming Units per 100 millimeters (CFU/100ml) (8.2-22,23)
67	pH	pH (8.2-22,23)	Float (8.2-22,23)
68	Physical characteristics (color, turbidity, smell)	Renk, Koku Bulanıklık (8.2-22)	String (8.2-22)
69	NH ₄ -N	NH ₄ -N (8.2-22)	Milligrams per liter (mg/L) (8.2-22)
70	SO ₄	SO ₄ -2 (8.2-22)	Milligrams per liter (mg/l) (8.2-22)
71	NO ₃	NO ₃ -2 (8.2-22)	Milligrams per liter (mg/L) (8.2-22)
72	Cl	Cl-1 (8.2-22)	Milligrams per liter (mg/l) (8.2-22)
73	Biological Oxygen Demand (BOD)	BOİ (8.2-22) BOD (8.2-23)	Milligrams per liter (mg/l) (8.2-22) Milligrams per liter (mg/L) (8.2-23)
74	Total Organic Carbon (TOC)	T. Org. Krb (TOC) (8.2-22)	Milligrams per liter (mg/L) (8.2-22)
75	Chemical Oxygen Demand (COD)	KOİ (8.2-22) COD (8.2-23)	Milligrams per liter (mg/l) (8.2-22) Milligrams per liter (mg/L) (8.2-23)
76	Total Dissolved Solids (TDS)	TDS (8.2-22)	microsiemen per centimetre (µS/cm) (8.2-22)
77	Total Nitrogen (TN)	Top. N (8.2-22) Total Nitrogen (8.2-23)	Milligrams per liter (mg/l) (8.2-22) Milligrams per liter

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
			(mg/L) (8.2-23)
78	Total Kjeldahl Nitrogen (TKN)	Kjeldahl Azotu (TKN) (8.2-22)	Milligrams per liter (mg/L) (8.2-22)
79	Total Phosphorus (TP)	Top. P (8.2-22) Total Phosphorus (8.2-23)	Milligrams per liter (mg/l) (8.2-22) Milligrams per liter (mg/L) (8.2-23)
80	Cd	Cd (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
81	Cr	Cr (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
82	Pb	Pb (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
83	Cu	Cu (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
84	Zn	Zn (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
85	Fe	Fe (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
86	Al	Al (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
87	Mn	Mn (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
88	Ni	Ni (8.2-22)	Micrograms per liter (ug/l) (8.2-22)
89	Dissolved Oxygen	Dissolved Oxygen (8.2-23)	Milligrams per liter (mg/L) (8.2-23)
90	Volumetric Flow	Volumetric Flow (8.2-23)	Cubic meter per second (m3/s) (8.2-23)
91	Water velocity	Water velocity (8.2-23)	Meter per second (m/s) (8.2-23)
92	Shapefile Type	type(8.2-24,25,26,27,28,29,20,31,32,33,34,37)	String (8.2-24,25,26,27,28,29,20,31,32,33,34,37)
93	Full type	FULL_TYPE(8.2-27)	String (8.2-27)
94	Horizon	HORIZON(8.2-27)	String (8.2-27)
95	Horizon Type	HorzType(8.2-27)	String (8.2-27)
96	Inserted By	INS_BY(8.2-27)	String, Acronym of operator (8.2-27)
97	Local Authority	LocalAutho(8.2-27,28,30) LOCALAUTHO (8.2-31)	String (8.2-27,28,30,31)
98	Out of River Basin districts	OutOfRBD (8.2-27)	String (8.2-27)
99	Parent waterbody	ParentWate(8.2-27)	String (8.2-27)
100	Transbound	Transbound(8.2-27)	String (8.2-27)
101	WaterBodyG	WaterBodyG(8.2-27)	String (8.2-27)
102	Artificial	ARTIFICIAL (8.2-28,31) Artificial (8.2-30)	String ("Yes","No") (8.2-28,30,31)
103	Modified	MODIFIED(8.2-28,30,31)	String ("Yes","No") (8.2-28,30,31)
104	System	SYSTEM(8.2-28,30,31)	String ("A","B") (8.2-28,30,31)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
105	Water Manag	WaterManag(8.2-28)	String (8.2-28)
106	Order	ORDER(8.2-29)	String (8.2-29)
107	Adjacent Hydrographic	AdjacentHy(8.2-30, 31)	String (8.2-30, 31)
108	Depth typology	DEPTH_CAT(8.2-30)	String (8.2-30) or int (0)
109	Donor Water	DonorWater(8.2-30, 31)	String (8.2-30, 31)
110	Intercalib	Intercalib(8.2-30, 31)	String (8.2-30, 31)
111	Processing Status	Processing(8.2-30, 31)	String (8.2-30, 31)
112	Protected Area	ProtectedA(8.2-30, 31)	String ("Yes", "no") (8.2-30, 31)
113	SubsitePre	SubsitePre(8.2-30, 31)	String (8.2-30, 31)
114	WISE refere	WISERefere(8.2-30, 31)	String (8.2-30, 31)
115	Type (Water typology)	Type (8.2-30, 31) TYPE (8.2-32)	String (8.2-30, 31, 32)
116	Depth	DEPTH(8.2-31) DEPTH2D (8.2-37)	Meters (m) (8.2-31, 37)
117	Tidal typology	TIDAL(8.2-31)	String (8.2-31)
118	Transitional Waterbody Category	TWCategory(8.2-31)	String (8.2-31)
119	Annual Exceedance Probability	AEP(8.2-32)	float (8.2-32)
121	Ext Id	EXT_ID(8.2-32)	String (alphanumeric) (8.2-32)
122	Model Code	ModelCode(8.2-32)	int (8.2-32)
123	Run Type	RunType(8.2-32)	String (8.2-32)
124	Scenario	Scenario(8.2-32)	String (8.2-32)
125	Source Code	SourceCode(8.2-32)	String (8.2-32)
126	Status	Status(8.2-32)	String (8.2-32)
127	Type Code	TypeCode(8.2-32)	String (8.2-32)
128	UoM Code	UoMCode(8.2-32)	int (8.2-32)
129	go id	goid(8.2-32)	String (alphanumeric) (8.2-32)
130	Potential	Potential (8.2-34)	int (8.2-34)
131	Wet Bulb Temperature	wetb (8.2-35)	Milimeters (mm) (8.2-35)
132	Vapour Pressure	vappr (8.2-35)	hectopascals (hPa) (8.2-35)
133	Mean Sea Level Pressure	msl (8.2-35)	hectopascals (hPa) (8.2-35)
134	Mean Wind Speed	wdsp(8.2-35)	knots (kt) (8.2-35)
135	Indicator	ind(8.2-35)	int (8.2-35)
136	Quality	Quality(8.2-36)	int (8.2-36)

3.2.3 Economic data

Table 3.2.3-1 describes the economic parameters of interest. Numbers in brackets link to the actual data source descriptions in Appendix 1 or D3.2 where the corresponding parameter is met.

Table 3.2.3-1: Economic parameters of interest

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
1	Indicator	VAR (D3.2 8.3-3,4,5,6,7,8,9,10) indic_de (D3.2 8.3-13) na_item (D3.2 8.3-13) wsstatus (D3.2 8.3-13,14) nace_r2 (D3.2 8.3-13,14) indic_ur (D3.2 8.3-15)	String (D3.2 8.3-3,4,5,6,7,8,9,10) code is the string lable in JSON format (D3.2 8.3-13,14,15): "indic_ur":{"label":"indic_ur", "category":{ "index":{"CR1003I":0, (...)}, "label":{"CR1003I":"Number of cinema seats per 1000residents", (...)} } "wstatus":{"label":"wstatus", "category":{ "index":{"EMP":0, "SAL":1}, "label":{"EMP":"Employed persons", "SAL":"Employees"} } }
2	Gender	SEX (D3.2 8.3-3,4,5) sex (D3.2 8.3-14)	1 letter string ("M", "F", or "T") (D3.2 8.3-3,4,5) code: string label in JSON object (D3.2 8.3-14): "sex":{"label":"sex", "category":{ "index":{"F":0, "M":1, "T":2}, "label":{"F":"Females", "M":"Males", "T":"Total"}} }
3	Position	POS (D3.2 8.3-3,4,5,6,7,8,9,10)	String "ALL" (D3.2 8.3-3,4,5,6,7,8,9,10)
4	Unit	UNIT (D3.2 8.3-3,4,5,6,7,9,10) unit (D3.2 8.3-13,14)	String (D3.2 8.3-3,4,5,6,7,9,10) Code: String label in JSON object (D3.2 8.3-13,14): "unit":{ "label":"unit", "category":{ "index": {"THS_PER":0}, "label":{"THS_PER":"Thousand persons"}} }
5	Unit multiplier	POWERCODE (D3.2 8.3-3,4,5,6,7,8,9,10)	String representation of number (D3.2 8.3-3,4,5,6,7,8,9,10)
6	Reference period	REFERENCEPERIOD (D3.2 8.3-3,4,5,6,7,9,10)	YYYY_3Digits (D3.2 8.3-3), YYYY (D3.2 8.3-6,7,9,10)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
7	Observed value	obsValue (D3.2 8.3-3,4,5,6,7,8,9,10) value (D3.2 8.3-13,14,15)	Floating point number (D3.2 8.3-3,4,5,6,7,8,9,10) code: string label in JSON object (D3.2 8.3-13,14,15): "value":{ "0":328.7,"1":329.9,"2":331.2, (...)},
8	Observation status	OBS_STATUS (D3.2 8.3-3,4,5)	String, e.g. "B" (D3.2 8.3-3,4,5)
9	SNA classification	SERIES (D3.2 8.3-6,7,8,9,10)	Last SNA classification (D3.2 8.3-6,7,8,9,10)
10	Measure	MEAS (D3.2 8.3-6,7,8,9,10)	String (D3.2 8.3-6,7,8,9,10)
11	Directorate	Υπηρεσία (D3.2 8.3-1)	String (D3.2 8.3-1)
12	Code Number	K.A. (D3.2 8.3-1,2)	String representation (Digits with format dddd.dd.dd) (D3.2 8.3-1,2)
13	Projected revenues	Προϋπολογισθέντα (D3.2 8.3-1,2)	Euros (D3.2 8.3-1,2)
14	Actual revenues	Διαμορφωθέντα (D3.2 8.3-1,2)	Euros (D3.2 8.3-1,2)
15	Commitments	Δεσμευθέντα (D3.2 8.3-1)	Euros (D3.2 8.3-1)
16	Payments orders	Ενταλθέντα (D3.2 8.3-1)	Euros (D3.2 8.3-1)
17	Paid liabilities	Πληρωθέντα (D3.2 8.3-1)	Euros (D3.2 8.3-1)
18	Established revenues	Βεβαιωθέντα (D3.2 8.3-2)	Euros (D3.2 8.3-2)
19	Collected revenues	Εισπραχθέντα (D3.2 8.3-2)	Euros (D3.2 8.3-2)
20	Device Id	Device (D3.2 8.8-11)	String (D3.2 8.8-11)
21	Type of detection	Moving (D3.2 8.8-11)	String (D3.2 8.8-11)
22	Number of spots	Αριθμός θέσεων (D3.2 8.3-12)	Integer (D3.2 8.8-12)
23	Income per spot (monthly)	Έσοδα / θέση (D3.2 8.8-12)	Euros (D3.2 8.8-12)
24	Income per spot (daily)	Έσοδα / θέση / ημέρα (D3.2 8.8-12)	Euros (D3.2 8.8-12)
25	Percentage of	Απόδοση τομέα	Euros (D3.2 8.3-12)

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
	sector Yield	(D3.2 8.3-12)	
26	Age	age (D3.2 8.3-14)	Code: string label in JSON object (D3.2 8.3-14): <pre>"age":{"label":"age", "category":{" "index":{"Y15-24":0,"Y15-74":1, (...)}, "label":{"Y15-24":"From 15 to 24 years","Y15-74":"From 15 to 74 years", (...)} } }</pre>
27	Nr of tickets	Otopark Bileti (for parking) (8.3-16) Ziyaretçi Sayısı (entrance) (8.3-17)	Integer (8.3-16,17)
28	Fee (per number of tickets) revenue	Otopark Ücreti (8.3-16) Giriş Ücreti (8.3-17)	(Turkish Lira) (8.3-16,17)
29	Employee Salary	Çalışan Maaş (8.3-18)	(Turkish Lira) (8.3-18)
30	General Electric Bill	Elektrik Faturası (8.3-19)	(Turkish Lira) (8.3-19)
31	Water pump Electric Bill	Pompa Elektrik Faturası (8.3-20)	(Turkish Lira) (8.3-20)
32	Number of passengers for different categories	NUMBER OF TOUR, FREE1, FREE2, FREE3, TICKET, STUDENT, PERSON, KREDI KART PERSON, TEACHER, RETRIED, S.KART INDIRIMLI, AIRPORT EMPLOYER, TAX AUDIT CARD, TOTAL SUM (8.3-21)	Integer (8.3-21)
33	Number of visitors	Numbervisitors	Integer (8.3-23)
34	Number of paying visitors	Number of paying visitors	Integer (8.3-23)

3.2.4 Social data

No update is given for social data, please refer to D3.2.

3.2.5 Miscellaneous data

Table 3.2.5-1 describes the parameters of interest that do not fit to any of the four aforementioned categories, but may nevertheless appear as part of the previous types of data: environmental, economical and/or social data. Numbers in brackets link to the actual data source descriptions in Appendix 1 or D3.2 where the corresponding parameter is met.

Table 3.2.5-1: Miscellaneous parameters of interest

#	Parameter of interest	Naming among data sources	Units or value representation among data sources
1	Version	VERSION	String (8.2-24,25,26)
2	Site URL	URL	URL string (8.2-24,25,26)
4	Category	CATEGORY (8.2-27,31) Category (8.2-30)	String (8.2-27,30,31)
5	Changes in versions	Change (8.2-27,28,29,30,31)	String (8.2-27,28,29,30,31)
6	Description	DESCRIP (8.2-27), Descriptio (8.2-34), Περιγραφή (D3.2 8.3-1,2)	String (D3.2 8.3-1,2,8.2-27,34)
7	Sensor ID	NR (D3.2 8.2-15) ID (D3.2 8.2-16) sourceID (D3.2 8.2-17,18,19)	String (Pn, n=1..12) (D3.2 8.2-15, 16) String (Alphanumeric id eg. "lora.0004A30B001FD07B") (D3.2 8.2-17,18,19)
8	Station	Station (8.2-20)	Id and name of station (8.2-20)
9	Sample Point	NKT (8.2-22)	String (Dx,x=1..8) (8.2-22)

4. Final version of the framework for data collection and pre-processing

4.1 Overall description

Big data technologies are employed by CUTLER to support data collection and pre-processing. The CUTLER platform was introduced with the deliverables D8.1 “*Integration Protocol and Technical Verification*” [10] and D2.2 “*Threat analysis for policy-supporting hybrid cloud infrastructure*” [11]. A recent update is provided in D2.3 “*First version architecture for scalable hybrid cloud infrastructure*” [2], discussing technical details of the cloud infrastructure as well. Therefore, in this section we do not repeat the content of previous deliverables, but discuss the porting of D3.2 solutions into the CUTLER cloud infrastructure.

The CUTLER platform is a shared cloud platform, delivering an architecture and software stack that is uniform across all pilots [1]. Figure 4.1-3 presents a simplified CUTLER architecture, which was described in more detail in D8.1 “*Integration Protocol and Technical Verification*” [10] and D2.2 “*Threat analysis for policy-supporting hybrid cloud infrastructure*” [11]. D3.2 focused on data collection from the original data sources, initial data preprocessing, and possible solutions to ingest this data to the storage for further processing, analysis, and visualization. This deliverable is focused on porting the solutions presented in D3.2 into the CUTLER cloud.

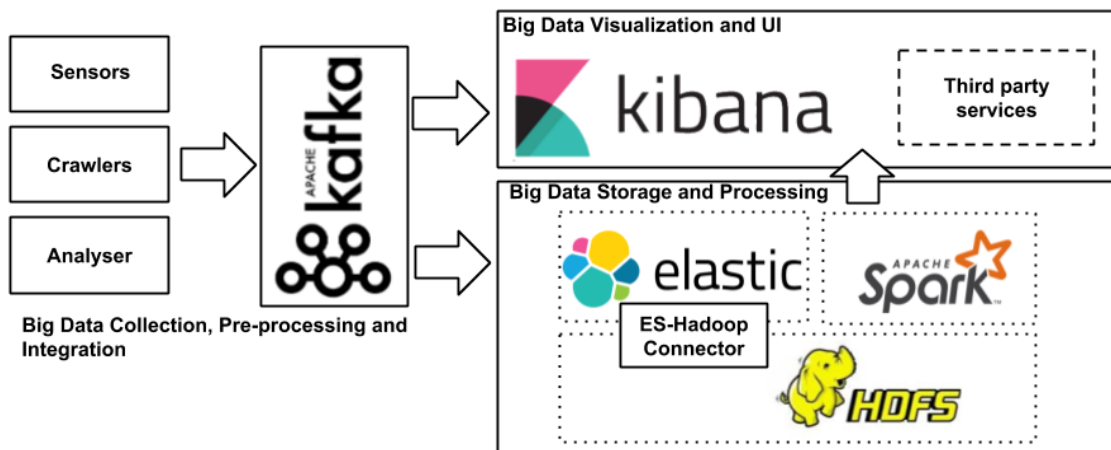


Figure 4.1-3: Simplified CUTLER architecture, taken from D3.2 (more details can be found in D8.1 [10] and D2.2 [11]).

Eventhough, both the CUTLER solution and the sandbox cluster reported in D3.2 rely on Hadoop [12], porting of the developed solutions appeared to be challenging. For data visualization, the CUTLER platform relies on Elastic search and Kibana [Elasticsearch] and solutions provided in D3.2 just tried these technologies. To provide interconnection between Hadoop and Elastic clusters, the CUTLER infrastructure provides ES-Hadoop [13], as well as configures ElasticSearchSink for Flume [14].

4.2 Data collection

The data collection and storage approach remains almost the same as described in D3.2: data is stored in .csv and .json files directly in the Hadoop Distributed File System (HDFS). However, a simplified version of the dedicated folder structure presented in D3.2 is followed in this version. Namely, the Data_code_number subfolder has been removed. Therefore, the HDFS structure implemented in this version is as follows:

- <Pilot name>
 - <Type of data> (*environment, economic, social*)
 - <Data source code> (*as provided in data description*)
 - File1
 - File2
 - File3

Both manual and automated data ingestion are supported as described in detail in D3.2.

4.3 Data pre-processing

Data pre-processing includes data cleaning, harmonization, and interoperability. In this deliverable, we follow the approach presented in D3.2.

4.3.1 Cleaning

We utilise the techniques introduced in D3.2 for data cleaning [3],[15],[1]. In the subsection below, the names of the data sources requiring data cleaning are provided in brackets along with the corresponding table number that can be found in Appendix 1 (Section 8).

4.3.1.1 Handling pattern violations

Pattern violations refer to values violating syntactic or semantic constraints [17]. That is errors related to formatting, misspelling, or semantic data types. Similarly to D3.2, we use data models, providing common vocabulary among all the data sources (see Sections 4.3.2, 4.3.3) and regulating the naming and unit conventions for the data sources. To comply with the data models, the following transformations are mainly applied:

- **Transformations to deal with formatting issues**
 - **Renaming attributes** is performed to fit the provided data models (e.g. ANT_ENV_IMEC_PREC2018, Table 8.2-17).
 - **Formatting values.** For example, transform a geohash value to latitude and longitude (e.g. ANT_ENV_IMEC_SEWER2018, Table 8.2-19).
 - **Removing units and special characters from the attribute value.** This guarantees that we do not have any extra characters in the value. No new cases required this transformation.
 - **Handling missing data.** Missing data involves cases that the instrument did not measure any value or it was out of order. If we know from the source that the data is missing, e.g. some check flag exists with the data, then we mark it as NA. For example, CORK_ENV_OPW_WL_15M, Table 8.2-36 has unavailable or not measured values for quality marked as *. This is replaced with NA.
- **Transformations to deal with attribute value issues**

- **Value recalculation.** This is required for the cases when the same attribute is provided in different units. Therefore, recalculations were made to comply with the data models (e.g. transforming location provided as Degrees Minutes and Seconds to decimal degrees representation in ANTA_ENV_CITYOFANTALYA2_MONTHLY, Table 8.2-22).

4.3.1.2 Handling rule-violations, duplicates, outliers

Please refer to D3.2 for details.

4.3.2 Harmonization

Data harmonization refers to the process of aligning data from heterogeneous sources into a coherent and unambiguous set [3]. Therefore, following D3.2, we extend the data representation models of D3.2 to support the new data sources as well. Different models are adopted for different types of data: time, geographical, environmental, economic, social and miscellaneous.

4.3.2.1 Time information

We use the model presented in D3.2, section 4.3.2.1. Since some data sources provide timestamp information in milliseconds, we suggest adding an additional field “**timestamp_ms**” to the model.

4.3.2.2 Geographical information

We use the recommendations presented in D3.2, section 4.3.2.2. However, since many new data sources are included containing geospatial information, we suggest the following approach. In cases, when we have a variable presenting the spatial information as geometries, e.g. line or polygon geometries composed of geographical coordinates, it is recommended to append “**_geom**” to such a field.

4.3.2.3 Environmental information

Here, we update the unified schema introduced in D3.2 to model environmental parameters in terms of naming, units, timezone and usage of spatial information (see Table 4.3.2.3-1). This model could be further extended in the second development phase of the project, if necessary.

Table 4.3.2.3-1: Environmental parameters name harmonization

#	Parameter	Harmonized codename	Reference Units
1	Water level (D3.2)	water_level	m (meters)
2	Wind speed (D3.2)	wind_speed	kn (knots)
3	Wind direction (D3.2)	wind_from_direction	O (degrees)
4	Air temperature (D3.2)	air_temperature	oC
5	Relative humidity (D3.2)	relative_humidity	%
6	Evapotranspiration (D3.2)	water_evaporation_amount	mm
7	Rainfall today (D3.2)	Precipitation_accumulated	mm
8	NO concentration (D3.2)	air_pollution_NO	µg/m ³
9	NO2 concentration (D3.2)	air_pollution_NO2	µg/m ³

10	O3 concentration (D3.2)	air_pollution_O3	µg/m ³
11	PM10 concentration (D3.2)	air_pollution_PM10	µg/m ³
12	PM25 concentration (D3.2)	air_pollution_PM25	
13	CO concentration (D3.2)	air_pollution_CO	mg/m ³
14	SO2 concentration (D3.2)	air_pollution_SO2	
15	Height of rainfall (D3.2)	precipitation	mm
16	Groundwater level (D3.2)	groundwater_level	m TAW
17	Volumetric water flow rate (D3.2)	water_volumetric_flow_rate	m ³ /s
18	Rainfall intensity (D3.2)	rainfall_intensity	L/m ²
19	Routed water runoff (D3.2)	water_routed_runoff	L/s
20	N concentration (D3.2)	conc_N	mg/L
21	P concentration (D3.2)	conc_P	mg/L
22	DO concentration (D3.2)	conc_DO	mg/L
23	TOC concentration (D3.2)	conc_TOC	mg/L
24	BOD concentration (D3.2)	conc_BOD	mg/L
25	COD concentration (D3.2)	conc_COD	mg/L
26	Fecal Coliform concentration (D3.2)	conc_fec_colif	number/100 mL
27	Total Coliform concentration (D3.2)	conc_tot_colif	number/100 mL
28	Fecal Streptococci concentration (D3.2)	conc_fec_strept	number/100 mL
29	Car speed	car_speed	km/h
30	Water velocity	water_velocity	m/s

Regarding time and geospatial information for the environmental data, the approaches described in chapters 4.3.3.1 and 4.3.3.2 were followed, respectively (see D3.2).

4.3.2.4 Economic information

Here, we update the unified schema introduced in D3.2 to model economic parameters in terms of naming, and units (see Table 4.3.2.4-1). This model could be further extended in the second development phase of the project, if necessary:

Table 4.3.2.4-1: Economic parameters name harmonization, in addition to Table 3.2.3-1

#	Parameter	Harmonized codename	Units or value representation
1	Reference period (D3.2)	REFERENCEPERIOD	YYYY
2	Directorate (D3.2)	DIRECTORATE	String
3	Code Number (D3.2)	CODE_NUMBER	String representation (Digits with format dddd.dd.dd)

4	Projected revenues (D3.2)	PROJECTED_REVENUES	Euros
5	Actual revenues (D3.2)	ACTUAL_REVENUES	Euros
6	Commitments (D3.2)	COMMITMENTS	Euros
7	Payments orders (D3.2)	PAYMENTS_ORDERS	Euros
8	Paid liabilities (D3.2)	PAID_LIABILITIES	Euros
9	Established revenues (D3.2)	ESTABLISHED_REVENUES	Euros
10	Collected revenues (D3.2)	COLLECTED_REVENUES	Euros
11	Revenue (D3.2)	REVENUE	EUROS
12	Number of spots (D3.2)	NR_OF_SPOTS	integer
13	Number of occupied spots (D3.2)	NR_OF_OCCUPIED_SPOTS	integer
14	Occupancy(D3.2)	OCCUPANCY	decimal
15	Monthly Income per parking spot (D3.2)	MONTHLY_INCOME_PER_SPOT	Euros (float)
16	Daily Income per parking spot(D3.2)	DAILY_INCOME_PER_SPOT	Euros (float)
17	Parking sector performance percentage (D3.2)	SECTOR_YIELD	Float
18	Number of tickets	tickets_number	Integer
19	Fee (per number of tickets) revenue	fee_revenue	Turkish Lira (float)
20	Employee salary	employee_salary	Turkish Lira (float)
21	General electric bill	electricity_bill	Turkish Lira (float)
22	Water pump electric bill	waterpump_electricity_bill	Turkish Lira (float)
23	Number of passengers for different categories	number_of_tour , free1, free2, free3, ticket, student, person, credit_card_person, teacher, retired, s_discount_card, airport_employee, tax_audit card, total_sum	Integer

4.3.2.5 **Social information**

No update is provided to social data. Please refer to D3.2 for details on the social data model.

4.3.2.6 **Miscellaneous information**

We placed general parameters in the “miscellaneous information” category and we provide here a unified schema to model them. This model could be further extended in the second development phase of the project, if necessary.

Table 4.3.2.6-1: Miscellaneous parameters name harmonization, in addition to Table 3.2.5-1

#	Parameter	Harmonized codename	Units or value representation
1	Version	version	String
2	URL	URL	URL string
4	Category	category	String
5	Changes in versions	change	String
6	Description	description	String
7	Sensor ID	sensor_id	String
8	Station ID	station_id	String
9	Sample Point	point_id	String

4.3.3 Interoperability

To facilitate exchange of information, CUTLER follows interoperability solutions and relies on accepted and widely used formats and standards [1]. The majority of solutions are inherited from D3.2, therefore, they are not repeated here.

4.3.3.1 *Time information*

Please refer to D3.2 for details.

4.3.3.2 *Geographical information*

In addition to the guidelines provided in D3.2 (which covered geographical information about regions or countries as well as about specific spatial points) , we also support cases where geographical information covers a wider area and not only a point. For such cases, depending on the requirements and the number of points that need to be represented, multiple sets of coordinates could be used in GeoJSON, KML format or alternatively a Web Map Service (WMS) standard¹. The Web Map Service interface standard is a simple HTTP interface for requesting geo-registered map images from one or more distributed geospatial databases. A WMS request defines the geographic layer(s) and area of interest to be processed and the response is one or more geo-registered map images that can be displayed in a browser application or over a map. The image can contain numerical values essential to the represented information such as temperature, precipitation, concentration etc.

4.3.3.3 *Environmental information*

Guidelines from D3.2 are followed, no update provided.

4.3.3.4 *Economic information*

Guidelines from D3.2 are followed, no update provided.

¹ <https://www.opengeospatial.org/standards/wms>

4.3.3.5 **Social information**

Guidelines from D3.2 are followed, no update provided.

4.3.3.6 **Miscellaneous information**

Miscellaneous information relies on accepted and widely used formats. Since no ontologies or taxonomies are used at the current stage, we provide as general as possible way to formulate miscellaneous attributes, see Table 4.3.2.6-1.

4.4 **Data privacy**

An initial assessment of the challenges and potential solutions for data privacy was provided in D3.2. This section will narrow the focus of data privacy within the context of the CUTLER platform to the specific processes and mechanisms required to protect sensitive data hosted by the platform. The implementation of these processes and mechanisms will support the existing CUTLER architecture, already extensively described across WP2, WP3, and WP8 deliverables.

The approaches to be adopted by CUTLER for data privacy are summarized below:

- Data governance model – defining access and usage policies for data collected from pilot sources.
- Data encryption – first line of defence to protect data confidentiality from when it is first collected, through its entire life-cycle.
- Data anonymisation – removing sensitive or identifiable fields from data sets.

Table 4.4-1 summarizes at a high level where each of these approaches must be applied across the three data domains of CUTLER. Based on the results of WP1, that identified the critical data sources where privacy is a concern, it is clear that social data sources require the most comprehensive set of solutions. While all three data domains will require data governance models to be defined, and will implement data encryption by default, only social data introduces the fine grained complexities of personally identifiable information (PII) that requires anonymisation.

Table 4.4-1: Data protection posture for CUTLER

	Economic	Environmental	Social
Data governance	✓	✓	✓
Data encryption	✓	✓	✓
Data anonymisation			✓

While social data is clearly the most complex challenge for the protection of privacy (i.e. PII), anonymisation strategies applied in CUTLER for social data will be easily re-used in future deployments of CUTLER in other municipalities. This is because social data follows standard, consistent and predictable formats that are agnostic to the specific pilot. In contrast, economic and environmental data will vary dramatically from one municipality to another. The effort to devise an anonymisation plan for each of these data sources would not be justified considering the lack of personal data sensitive concerns in those domains. Instead, an end-to-end encryption process is sufficient to protect the data from unauthorised access. Authorised access to the data is for the complete data sets. This in turn simplifies the data governance model that will be discussed further in the next section.

4.4.1 Data & Information Security Governance

The data governance model, and resulting policies applied to the data, form part of the overall information security governance strategy for CUTLER. The objective of these governance strategies are to formulate a set of policies related to the CUTLER platform data management, data optimization, data security, and privacy protection. This gives organisations running the platform the tools to plan, monitor, and act on all activities related to data governance. This is an important consideration for big data platforms, as without a centralised strategy and governance structure, it will be more difficult to make the transition from the *promise of analytics* to the actual realisation of those benefits.

Data governance and information security governance go hand in hand, and share common features. Good data governance policies and enforcement provides more visibility of the data and therefore enables better information security governance. As this section is primarily concerned with the protection of data and privacy, Table 4.4.1-1: presents an overview of the CUTLER information security maturity model. The model defines six domains that touch information security:

- *Information risk* – Processes for identification and remediation of risks
- *Policy management* – Definition and implementation of policies
- *Information access & security* – Implementation of security (e.g. access control) controls
- *Information capture & classification* – Processes for capture and classification of data
- *Information content governance* – Policy definition and implementation for content (i.e. CUTLER data)
- *Lifecycle management* – Processes for management of data storage and retention

Table 4.4.1-1: CUTLER Information Security Governance Maturity Levels

	Information Risk	Policy Management	Information Access & Security	Information Capture & Classification	Information Content Governance	Information Lifecycle Management
Optimised						
Managed	✓			✓		
Proactive		✓	✓		✓	
Reactive						
Aware						✓

The ultimate aim for the CUTLER platform (through the efforts of WP1, WP2, WP3, and WP8) is to achieve at least *Managed* status across all domains. The status of the current version of the platform is shown in the table.

The processes for identifying *information risks*, and capturing and classifying data are the most mature (*Managed*). This is due to the thorough investigation of data sources in WP1, and the implementation of data collection mechanisms in WP3, respectively. D3.2 also proposed a simple, effective classification scheme to identify sensitive data as it is ingested to the platform. This classification scheme informs access control rules and data protection mechanisms for the data (e.g. PUBLIC data may have relaxed security

controls, whereas data marked with one of the SENSITIVE-Lx labels will require access controls, encryption, and potentially anonymisation).

The *policy management*, *information access and security*, and *information content governance* domains are not mature, and are still in development. Therefore we consider CUTLER to have a proactive stance on these information security domains. For *policy management*, WP2, and WP8 are leading the definition of security policies for the platform and their implementation. For example, following the assessment of platform components in D2.3, Kerberos was identified as a compatible service for providing an authentication service to the CUTLER big data platform. This will enforce authentication and access policies set for users of the CUTLER platform, which also ties into the *information access and security* domain. Encryption will also be used to ensure baseline security level for all data on the platform. *Information content governance* relates more to the protection of the information content, that is encryption techniques and anonymisation. The emphasis here is on the analysis of the data sources to identify specific content that needs additional protection (i.e. anonymisation) or removal.

The final domain is related to the *lifecycle management* of the data. Policies should be defined to determine where the data is stored and for how long. It has already been identified that data for the ANTALYA pilot should remain within Turkey. The remaining three EU-based pilots can all host their data in DELL facilities in Ireland. However, two aspects of this domain are under investigation now. First, WP1 and corresponding partners discuss the roles of the involved partners and negotiate the terms of the controller/ processor agreements in order to adequately attribute the data controller and data processor roles (as defined in the GDPR). Information on the roles attributed to each partner will be presented in the next legal deliverable, i.e. D1.4. Moreover, it is important to highlight that distinction has to be made between research and exploitation phases of the project, as partner involvement and the roles performed will be different. The qualification of the actors involved as controllers and processors in the exploitation phase will be addressed in the later stage of the project. This will be resolved in collaboration with WP1 to clearly define the roles. Second, a policy for lifecycle management of data is under development. It is on the roadmap for the CUTLER platform but is not yet a priority for the piloting stage. This will feed into the information content governance domain, providing policy to set the expiration date metadata on content.

The solutions under development for data and information security governance, as originally identified in D3.2, are Apache Atlas [16] and Apache Ranger [17]. A solution using RSA Archer [18] will also be proposed, that provides a framework to help organizations identify, manage and implement appropriate controls around data processing activities. This solution is specifically designed to ensure the accuracy, completeness, confidentiality and transparency of PII and assess the data protection risks associated with those activities in the context of legislation such as GDPR.

Figure 4.4.1-1 presents the proposed architecture for CUTLER's data governance implementation. Here, as data is loaded into the platform, Apache Atlas will classify data and maintain a common metadata store that is accessible by CUTLER platform components. As part of the ETL stage, all data will be encrypted and selected data sources will have fields anonymised (see Section 4.4.3). Based on the Atlas classifications, security policies can be defined and managed in Apache Ranger to set the access rules to the data. To allow for a more flexible access control mechanism, role-based access control (RBAC) rules will be assigned through Apache Ranger. User management of the platform, and access to platform components will be managed through Kerberos. In addition to the existing VM infrastructure, additional resources will be needed to host these three new security services.

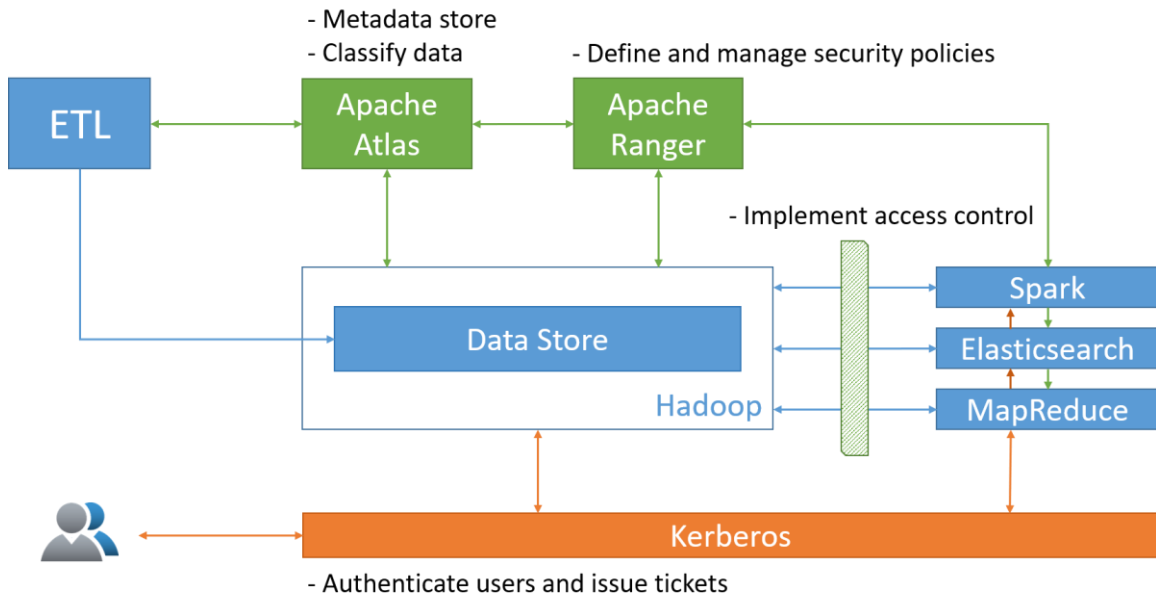


Figure 4.4.1-1 - CUTLER Data Governance Implementation Plan

Outside of the technical architecture of the data governance plan, further investigation, collaborating with WP1, is needed into defined appropriate classifications for the data that will be used in Apache Atlas. These should clearly identify which data is required to be encrypted or anonymised. Note, that encryption will be used by default to protect data hosted on the CUTLER platform. User roles also need to be defined to support the RBAC mechanism in Apache Ranger.

4.4.2 Encryption

With mechanisms and processes for data and information security governance in development, the next aspect of data protection is encryption. The requirements are simple here, all data collected by the platform should be encrypted. This will protect the data from exposure to any external or unauthorised parties. The authentication and authorisation mechanisms that are implemented for the platform (as noted in the previous section) will determine who can access the data.

As noted in D2.3, a Virtual Private Network (VPN) has been created to isolate and protect the CUTLER platform network, providing encrypted communications to/from the platform. All communications are protected using 4,096 bit RSA and 256 bit AES. Communications between the CUTLER platform components will be configured to use SSL/TLS where supported. For the purposes of facilitating testing during the piloting stage, this feature is not enabled. D2.3 provided a table listing compatible secure channels between the CUTLER platform components.

The other aspect of data encryption is at-rest encryption, when the data is stored in HDFS. For this, HDFS provides a transparent encryption solution. This implements transparent, end to end encryption of data written to HDFS across the Hadoop cluster. Using this mechanism, authorized tasks running on the cluster are unaware of the encryption/decryption processes.

4.4.3 Anonymization

As identified in Table 4.4-1, only the social data sources require anonymisation to protect sensitive data and PII. In one sense, this simplifies the challenge posed by creating a uniform, repeatable anonymisation strategy for a disparate set of data

sources. Instead, the requirement is only on social data that will come in a consistent, well-known format. Targetting these sources, will simplify the adoption of the same anonymisation techniques by future deployments of CUTLER. However, it must also be noted that despite the consistent format and structure of the data, the complexities in removing sensitive data are far greater.

Removing or substituting directly identifiable fields is, at first glance, a straightforward solution to protecting identities. It is preferred that identifiable fields such as unique ID values or usernames, are substituted with a random unique user ID (UUID) so that multiple entries (e.g. tweets) from the same person can be correlated with each other. However, this presents a problem where a profile of a person can be built up and potentially lead to identification of the person. Similarly, the content of the tweet (sticking with the example of Twitter) itself can simply be reverse-searched on Twitter to find the user account that created the tweet. This is a particularly problematic situation in the context of CUTLER, where pilots are based in specific geographic regions, and where there are various political interests involved.

Due to the complexities in understanding how collected data can be legitimately used, during the piloting phase CUTLER will ensure all data is encrypted by default. As a, proactive step to protect sensitive content and identities, directly identifiable fields could be substituted with random UUIDs that are linked to the hash of the field in a separate table as illustrated in 4.4.3-1. In this way, the primary dataset stored on the platform does not provide any correlation between multiple tweets from the same accounts. However, as further investigation is required into the trade-off between anonymisation and utility, as well as GDPR considerations, the link table will be maintained separately until a decision is made. This does not however account for the ability to search for the tweet using the tweet content, and therefore find the originator. In terms of GDPR, this does not prevent us from using the data provided, only the data that is needed for the platform is retained, [4],[5]. This will require further analysis with WP1 and WP6 in particular on social media data to determine what information is required to retain utility, [4],[5].

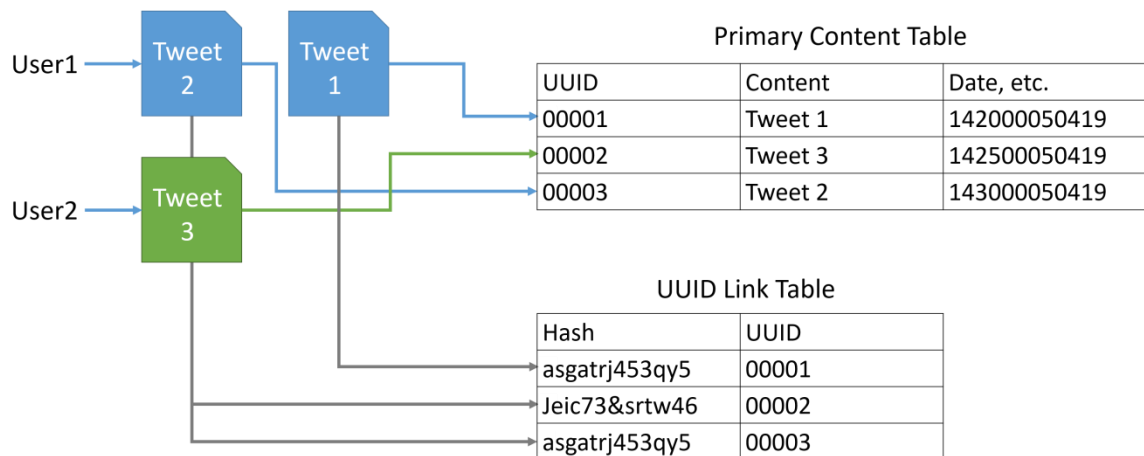


Figure 4.4.3-1 - Social Data UUID Substitution Strategy

The implementation of anonymisation of fields in a dataset will require a combination of custom tools developed for CUTLER, and will, where possible, leverage the dynamic column masking capabilities of Apache Ranger that allows for policies to be set that mask or anonymized sensitive data columns.

4.5 From the sandbox to the CUTLER cloud

This section describes the actual porting of the crawlers from the sandbox presented in D3.2 into DELL's CUTLER cloud infrastructure. Since D3.2 contained very few crawlers

for Elasticsearch cluster, and this solution was not yet fully tested by partners, porting of these crawlers is left for the next period. Here, we report on portability of the crawlers; actual data storage will start after further investigation of each data source by WP1 and corresponding partners. Technical details about the cloud infrastructure could be retrieved from D2.3, therefore we do not repeat them here.

To structure this section, we will follow with the data collection approaches presented in D3.2, namely manual and automated data collection. We will present the work conducted in a tabular structure, where each row corresponds to the crawler for a particular data source (**DS_ID**) and briefly comments on the challenges encountered and how they were solved (**CHALLENGE**). Crawlers for the listed data sources (Table 4.5.1-1, Table 4.5.2.1-1, Table 4.5.2.2-1, Table 4.5.2.3-1) have been successfully ported into the CUTLER cloud infrastructure.

4.5.1 Manual data ingestion

Manual data ingestion means that the data is directly loaded to the platform. Please refer to D3.2 for details.

Table 4.5.1-1: Porting of manually ingested data

DS_ID	CHALLENGE
THESS_SOC_IMC_MONTHLY (D3.2 5.4-1)	No issues

4.5.2 Automated data ingestion

Automated data ingestion means that a particular script is written to fetch the data automatically from the provided data source. Please refer to D3.2 for details.

4.5.2.1 One-time data acquisition

One-time load means that the data provider gives static data, that is, data that is not going to change within the course of the project. Please refer to D3.2 for details.

Table 4.5.2.1-1: Porting of one-time automated data acquisition software

DS_ID	CHALLENGE
THESS_ENV_CITYOFTHESS_DAILYEARLY (D3.2 5.2-14)	Missing library (xlrd), later installed by UOULU
ANT_ENV_CITYOFANT_HISTPREC (D3.2 5.2-16)	Missing library (pandas), later installed by DELL
ANT_ENV_CITYOFANT_GWL (D3.2 5.2-17)	Missing library (pandas), later installed by DELL
ANTALYA_ENV_CITYOFANTALYA_PERRMINUTE (Batch) (D3.2 5.2-21)	Missing library (pandas) later installed by DELL and Chromedriver issues solved by UOULU&DELL
THESS_ECO_THESSALONIKI_MUNICIPALITY_BUDGET (BATCH)(D3.2 5.3-1)	Missing library (pandas) later installed by DELL and Chromedriver issues solved by UOULU&DELL
THESS_ENV_IMET_SPEED_15MIN(BATCH) (5.2-26)	Missing libraries (pandas, dotenv) later installed by DELL& UOULU Chromedriver issues solved by UOULU&DELL
ANTA_ENV_CITYOFANTALYA2_MONTHLY (5.2-28)	Missing library (pandas), later installed by DELL
ANTA_ENV_WATERQUALITYFLOW_CITYOFANTALYA_MONTHLY(5.2-29)	Missing library (pandas), later installed by DELL

ANTA_ECO_SEVERAL_CODES (5.3-7)	Missing library (pandas), later installed by DELL
ANTA_ECO_CITYOFANTALYA_SHOP_SRENTearn_YEAR(5.3-8)	Missing library (pandas), later installed by DELL
ANTA_ECO_CITYOFANTALYA_CITY_ZONEPUBLICTRANSPORTATIONPAS_ENGernumber_MONTHLY(5.3-9)	Missing library (pandas), later installed by DELL
CORK_ECO_VISITORS_DAILY (5.3-10)	Missing library (pandas), later installed by DELL
CORK_ENV_OPW_WL_15min (5.2-23)	Missing library (Pandas), later installed by DELL
CORK_ENV_MET_W_DAILY(5.2-24)	Missing library(Pandas), later installed by DELL
CORK_ENV_EPA_SAC_2015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_NHA_2012 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_SPA_2015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_GWWFD_20102015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_LWFD_20102015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_RWFD_20102015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_CWFD_20102015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_EPA_TWFD_20102015 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_OPW_FLOODS_2016 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_CCC3_LAND_2014 (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
CORK_ENV_CAR_PARKING (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
ANT_ENV_CITYOFANT_MAPS (5.2-25)	Missing libraries(pyshp and geopandas), later installed by DELL
PIALL_ECO_OECD (D3.2 5.3-3)	R package not available on the platform yet, installation in progress

4.5.2.2 **Scheduled data acquisition**

Scheduled data acquisition means that data is queried in predefined intervals, e.g. daily, weekly, etc. Here, we use the same approach based on Apache Flume as described in D3.2.

Table 4.5.2.2-1: Porting of one-time scheduled data acquisition software

DS_ID	CHALLENGE
ANTALYA_ENV_CITYOFANTALYA_PERMINUTE (D3.2 5.2-22)	Missing library (pandas) later

	installed by DELL and Chromedriver issues solved by UOULU&DELL
THESS_ECO_THESSALONIKI_MUNICIPALITY_BUDGET (D3.2 5.3-2)	Missing library (pandas) later installed by DELL and Chromedriver issues solved by UOULU&DELL
THESS_ENV_IMET_SPEED_15MIN (5.2.27)	Missing libraries (pandas, dotenv) later installed by DELL& UOULU Chromedriver issues solved by UOULU&DELL
ANT_ENV_IMEC_PREC2018 (5.2-30)	No issues
ANT_ENV_IMEC_OPENWATER (5.2-30)	No issues
ANT_ENV_IMEC_SEWER2018 (5.2-30)	No issues

4.5.2.3 Stream data acquisition

Stream data acquisition means that data is acquired as soon (or very close) as it is generated, e.g. sensor reading data. Again, the same approach based on Apache Flume is described. Please refer to D3.2 for details.

Table 4.5.2.3-1: Porting of stream data acquisition software

DS_ID	CHALLENGE
CORK-SOC_TWITTER (D3.2 5.4-5)	not ported yet, further legal assessment is required for processing social data

4.5.3 Hadoop to/from Elasticsearch

To provide interconnection between Hadoop and Elastic clusters, ES-Hadoop is installed in the CUTLER cloud infrastructure, see Figure 4.1-3. ES-Hadoop is a library that allows data to flow bi-directionally between Hadoop and Elasticsearch. At the moment of writing of D3.3, ES-Hadoop supports Map-Reduce, Hive, Pig, Storm and Spark [13].

We have tested connectivity provided by ES-Hadoop through Spark. This is a convenient instrument, since we are able to conduct the required data processing and analysis with Spark and flush the clean data or results to Elasticsearch cluster for further visualization.



Figure 4.5.3-1: General flow of data transfer from HDFS to ES and vice versa

The code snippet below (Figure 4.5.3-2) demonstrates how environmental data from one data source is fetched from HDFS and sent to Elasticsearch cluster with Spark and ES-Hadoop connector.

```
import org.apache.spark.SparkContext
import org.apache.spark.SparkContext._
import org.apache.spark.sql._
import org.apache.spark.sql.functions._
import org.apache.spark.SparkConf
import org.elasticsearch.spark._
import org.elasticsearch.spark.sql._
val sc = SparkContext.getOrCreate()
val sparkSession =
SparkSession.builder.config(sc.getConf).getOrCreate()
val df =
spark.read.format("csv").option("header", "true").option("mode", "DROPMALFORMED").option("inferSchema", "true").load("PATH to HDFS")
df.saveToEs("Index_Name/FileName")
```

Figure 4.5.3-2: Spark code snippet to port data to Elasticsearch with ES-Hadoop

The schema of transferred data to Elasticsearch can be seen in the following Figure 4.5.3-3.

```
cutler-user@cutler-es-3:~
cutler-user@cutler-es-3:~$ cat 'user_env?pretty'
{
  "user_env" : {
    "aliases" : { },
    "mappings" : {
      "test" : {
        "properties" : {
          "AtmosphericPressure" : {
            "type" : "float"
          },
          "DateTime" : {
            "type" : "long"
          },
          "DewPoint" : {
            "type" : "float"
          },
          "Gust" : {
            "type" : "float"
          },
          "Hmax" : {
            "type" : "float"
          },
          "MeanWaveDirection" : {
            "type" : "float"
          },
          "settings" : {
            "index" : {
              "creation_date" : "1557231016911",
              "number_of_shards" : "5",
              "number_of_replicas" : "1",
              "uuid" : "3aMg_w7fTmeJe90ofEBWVw",
              "version" : {
                "created" : "6050199"
              },
              "provided_name" : "user_env"
            }
          }
        }
      }
    }
  }
}
```

Figure 4.5.3-3: Snapshot of ported data from HDFS to ElasticSearch

In addition, since the main acquisition instrument here is Apache Flume [19], we have tried also to configure ElasticSearchSink [14]. This sink facilitates writing data directly to the Elasticsearch cluster. That means that for the cases where data does not require any processing, data can be directly flushed to Elasticsearch cluster.

For instance, here the Flume agent could have two sinks: one writes data to HDFS and another one directly to Elasticsearch. Results of this experimentation will be reported in the upcoming deliverable D3.5.

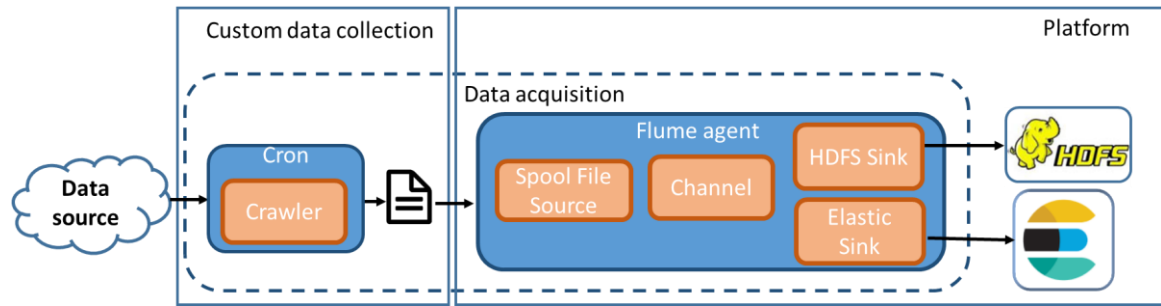


Figure 4.5.3-4: Automated data ingestion to HDFS and Elastic Search using Apache Flume

5. Software description

Since some additional data sources were collected for the pilots that were not reported in D3.2, here we report on the development of crawlers for these data sources. We do not repeat the ones reported in D3.2. Sections 5.2, 5.3, and 5.4 below contain tables numbered continuously from D3.2. All crawlers developed within CUTLER are maintained in the GitHub repository of the project [7].

5.1 Template for software description

For convenience, we use the same software description template presented in D3.2 (see Table 5.1-1). Each data source is identified with a unique number (**DS_ID**). Each data source can supply different data; therefore, there is also unique data number (**D_ID**). We associate each data source with a particular pilot city and corresponding data category (Environmental, Economic, and Social) (**CODE**).

Table 5.1-1: Abbreviations used for coding the pilot cities and data categories (from D3.2)

CODE	Description
PiThess	Municipality of Thessaloniki pilot
PiAntal	Metropolitan Municipality of Antalya pilot
PiAntw	City of Antwerp pilot
PiCork	Cork County pilot
PiAll	All pilots
Soc	Society
Env	Environment
Eco	Economy
DataAll	Society, Environment and Economy

Also, we specify the link to CUTLER's GitHub repository (**LINK**) and how the data is actually collected to the platform (**DATA ACQUISITION MODE**) (refer to Section 4.2 of D3.2 for further explanation). This information is summarized in Table 5.1-2.

Table 5.1-2: Coding the data acquisition mode (from D3.2)

DATA ACQUISITION MODE	Description
Manual	Data is directly loaded to the platform
Automated	Data is loaded with the program (crawler)
One-time	Data is loaded only once
Scheduled	Data is queried with predefined interval, like daily, hourly, etc. Pulled by the crawler.
Streaming	Data is acquired as soon (or very close) as it is generated. Pushed to the platform.

We also describe: a) the programming language and libraries of the crawler (**LANGUAGE AND LIBRARIES**), b) the parameters that the crawler accepts (**PARAMETERS**), c) the crawler workflow description (**DESCRIPTION**), and d) crawler

output information (**OUTPUT**). If the crawler belongs to the Elasticsearch and Kibana testbed, it is indicated in the table title. Table 5.1-3 provides an example of a crawler description as a codified table.

Table 5.1-3: Crawler description format (example for River Estuary Weather data in Cork), from D3.2

DS_ID	D_ID	CODE
CORK_ENV_NMCI_HOURLY	CORK_ENV_NMCI_HOURLY_3	PiCork-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/tree/master/Environmental		
DATA ACQUISITION MODE: Automated - Scheduled Scheduled with the Cron service – The script runs every hour and stores information to specified location (location information is provided in the code itself)		
LANGUAGE AND LIBRARIES: Python 3.x, requests, uuid, csv, beautifulsoup4		
PARAMETERS: URL String of the data source provider: “ http://86.43.106.118/winfiles/cumulus/ ”		
DESCRIPTION: <ul style="list-style-type: none"> - The software sends requests to provided URL (PARAMETERS) - Data received is in the form specified with Table 8.2-11 - Crawler separates headers and values, special characters - Removes units and other unnecessary characters - Time information is formatted according to ISO 8601 (“Time”, Table 3.2.1) - Data is stored in .CSV format in the file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS 		
OUTPUT: The crawler generates CSV. Dawn,Sunrise,Moonrise,Dusk,Sunset ,Moonset,Daylight,Length,Phase,Temperature,Dew Point ,Windchill,Humidity,Heat Index,Apparent Temperature,Solar Radiation,Evapotranspiration Today,Rainfall Today,Rainfall Rate,Rainfall This Month,Rainfall This Year,Rainfall Last Hour,Last rainfall,Wind Speed (gust),Wind Speed (avg),Wind Bearing,Beaufort F2,Barometer ,Rising slowly,Time 9.9 ,8.2 ,9.7 ,89,9.9 ,7.4 ,0 ,0.25 ,2.8 ,0.0 ,73.8 ,120.2 ,0.0 ,2018-11-12 13:16,12.2 ,5.8 ,244 WSW,Light breeze,999.12 ,0.42 ,2018-11-12T14:54:00+00		

5.2 Environmental Data

This subsection presents the new crawlers developed for collecting environmental data. The data sources corresponding to these crawlers are identified by the DS_ID field and can be seen in Appendix I.

Table 5.2-23: Historic water levels data by the OPW for station 19069 Ringaskiddy NMCI for Cork Pilot

DS_ID	D_ID	CODE
CORK_ENV_OPW_WL_15min	CORK_ENV_OPW_WL_15min	PiCork-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/CORK_ENV_OPW_WL_15min.py		
DATA ACQUISITION MODE: Automated – One-time		
LANGUAGE AND LIBRARIES: Python 3.x, uuid, json, pandas		
PARAMETERS: URL String of the data source provider: “ http://waterlevel.ie/hydro-data/stations/19069/Parameter/S/complete.zip ”		
DESCRIPTION: <ul style="list-style-type: none"> - The software sends requests to provided URL (PARAMETERS) - Data received is in the form specified with Table 8.2-36 - The received response is transformed from text format to csv format - Time information is formatted according to ISO 8601 (“Time”, Table 3.2.1) - Missing values which are marked with asterisk * by data provider are replaced with NA - Additional column specifying station id is added in the software - Data is stored in .CSV format in the file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS 		
OUTPUT: The crawler generates CSV files. The field names correspond to parameters presented in Table 8.2-36, including the transformations described above, in the crawler DESCRIPTION section. water_level,Quality,DateTime,station_id -0.535,NA,2011-12-31T00:15+00,19069 Ringaskiddy -0.715,NA,2011-12-31T00:30+00,19069 Ringaskiddy		

Table 5.2-24: Weather historical information for Ireland (Cork pilot)

DS_ID	D_ID	CODE
CORK_ENV_MET_W_DAILY	CORK_ENV_MET_W_DAILY	PiCork-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/CORK_ENV_MET_W_DAILY.py		
DATA ACQUISITION MODE: Automated – One-time		
LANGUAGE AND LIBRARIES: Python 3.x, uuid, json, pandas		
PARAMETERS: URL String of the data source provider: https://cli.fusio.net/cli/climate_data/webdata/hly1075.csv		
DESCRIPTION: <ul style="list-style-type: none"> - The software sends requests to provided URL (PARAMETERS) - Data received is in the form specified with Table 8.2-36 		

<ul style="list-style-type: none"> - The received response is transformed from text format to csv format - Time information is formatted according to ISO 8601 ("Time", Table 3.2.1) - Missing values which are marked as asterisk * by data provider are replaced with NA - Additional column specifying station id is added - Data is stored in .CSV format in the file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS
<p>OUTPUT:</p> <p>The crawler generates CSV files. The field names correspond to parameters presented in Table 8.2-36, including the transformations described above, in the crawler DESCRIPTION section</p> <pre>water_level,Quality,DateTime,station_id -0.535,NA,2011-12-31T00:15+00,19069 Ringaskiddy -0.715,NA,2011-12-31T00:30+00,19069 Ringaskiddy</pre>

Table 5.2-25: Software for converting shapefiles to GeoJSON (Cork pilot)

DS_ID	D_ID	CODE
ant_env_cityofant_maps, CORK_ENV_CAR_PARKING, CORK_ENV_CCC3_LAND_2014, CORK_ENV_EPA_CWFD_20102015, CORK_ENV_EPA_GWWFD_20102015, CORK_ENV_EPA_LWFD_20102015, CORK_ENV_EPA_NHA_2012, CORK_ENV_EPA_RWFD_20102015, CORK_ENV_EPA_SAC_2015, CORK_ENV_EPA_SPA_2015, CORK_ENV_EPA_TWFD_20102015, CORK_ENV_OPW_FLOODS_2016	ant_env_cityofant_maps, CORK_ENV_CAR_PARKING, CORK_ENV_CCC3_LAND_2014, CORK_ENV_EPA_CWFD_20102015, CORK_ENV_EPA_GWWFD_20102015, CORK_ENV_EPA_LWFD_20102015, CORK_ENV_EPA_NHA_2012, CORK_ENV_EPA_RWFD_20102015, CORK_ENV_EPA_SAC_2015, CORK_ENV_EPA_SPA_2015, CORK_ENV_EPA_TWFD_20102015, CORK_ENV_OPW_FLOODS_2016	PiCork-Env, PiAntw-Env
<p>LINK:https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/SHAPE_FILE_TO_GEOJSON_CORK%26ANTWERP.py</p>		
<p>DATA ACQUISITION MODE: Automated – One-time</p>		
<p>LANGUAGE AND LIBRARIES: Python 3.x, uuid, json, pandas</p>		
<p>PARAMETERS: Shapefile provided by data provider manually</p>		
<p>DESCRIPTION:</p> <ul style="list-style-type: none"> - The software reads the content of given .shp file - The read data is then expanded to create field names and values - After extraction, the data is converted to GeoJSON structure - It is then stored in .JSON format with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS 		
<p>OUTPUT:</p> <p>The crawler generates JSON files where the structure of the data corresponds to the parameters presented in Tables 8.2-24-34.</p> <pre>{"geometry.coordinates":{"0":[[[580584.5403438584,561945.8300102253]]},"geometry.type":{"0":"Polygon"},"properties.Area":{"0":0},"properties.Descriptio":{"0":""},"properties.ID":{"0":"1","1"},"properties.Potential_":{"0":0},"type":{"0":"Feature"}}</pre>		

Table 5.2-26: Car Speed in Thessaloniki (Batch)

DS_ID	D_ID	CODE
THESS_ENV_IMET_SPEED_15MIN	THESS_ENV_IMET_SPEED_15MIN	PiThess-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/thess_env_imet_speed_15min_batch.py		
DATA ACQUISITION MODE: Automated – One time		
LANGUAGE AND LIBRARIES: Python 3.x, pandas, uuid, selenium, beautifulsoup4, datetime, time, dotenv, os Path for Selenium webdriver needs to be added.		
PARAMETERS: URL String of the data source provider: "https://www.trafficthessreports.imet.gr/export.aspx"		
DESCRIPTION: <ul style="list-style-type: none"> - The software sends requests to provided URL (PARAMETERS) - Crawler needs to log-in - Crawler interacts with the webpage to have access to all the data: 1) selects one of the 4 routes of interest 2) Introduces as period of interest different periods of time up to the current day (if done for the whole historical period it will fail) - Crawler asks for the result - It waits until the data is present (a link to a .csv file is presented at the bottom of the page) - It scrolls to the end of the webpage, in order to show all the values at the same time (by selecting Show All from the corresponding dropdown) - Data received is in the form specified with Table 8.2-21 - Field names are change according to data models - Data is stored in .CSV format in the file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS 		
OUTPUT: The crawler generates CSV files. The field names correspond to the parameters presented in Table 8.2-21, including the transformations described above, in the crawler DESCRIPTION section <pre> path_id,path_name, DateTime,car_speed,samples 1,1. Εγνατία (Συνητριβάνι - Πλατεία Δημοκρατίας), 2017-1-1, 40, 92 </pre>		

Table 5.2-27: Car Speed in Thessaloniki

DS_ID	D_ID	CODE
THESS_ENV_IMET_SPEED_15MIN	THESS_ENV_IMET_SPEED_15MIN	PiThess-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/thess_env_imet_speed_15min.py		
DATA ACQUISITION MODE: Automated - Scheduled The script runs every day and stores new information from previous day, if any		
LANGUAGE AND LIBRARIES: Python 3.x, pandas, uuid, selenium, beautifulsoup4, datetime, time, dotenv		

Path for Selenium webdriver needs to be added.
PARAMETERS: URL String of the data source provider: " https://www.traffichessreports.imet.gr/export.aspx"
DESCRIPTION: <ul style="list-style-type: none"> - The software sends requests to provided URL (PARAMETERS) - Crawler needs to log-in - Crawler interacts with the webpage to have access to all the data: 1)selects one of the 4 routes of interest 2) Introduces as period of interest different periods of time up to the current day (if done for the whole historical period it will fail) - Crawler asks for the result - It waits until the data is present (a link to a .csv file is presented at the bottom of the page) - It scrolls to the end of the webpage, in order to show all the values at the same time (by selecting Show All from the corresponding dropdown) - Data received is in the form specified with Table 8.2-21 - Field names are change according to data models - Data is stored in .CSV format in the file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS
OUTPUT: The crawler generates CSV files. The field names correspond to the parameters presented inTable 8.2-21, including the transformations described above, in the crawler DESCRIPTION section DateTIme,path_id,path_name,car_speed,samples 1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας), 2017-1-1, 40, 92

Table 5.2-28: Historic Water Quality Analysis Düden Creek (2012-2017) (Antalya pilot)

DS_ID	D_ID	CODE
ANTA_ENV_CITYOFANTALYA2_MONTHLY	ANTA_ENV_CITYOFANTALYA2_MONTHLY	PiAnta-Env
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/anta_env_cityofantalya2_monthly.py		
DATA ACQUISITION MODE: Automated – One time		
LANGUAGE AND LIBRARIES: Python 3.x, os, uuid, pandas, shutil		
PARAMETERS:		
DESCRIPTION: Prerequisites: File with data ('antalya_cutler_all_data_ (version 1).xlsx) need to be in the l_temp_path folder <ul style="list-style-type: none"> - Open data information, data in the format specified in Table 8.2-22 - Separate headers and values special characters - Removes units and other unnecessary characters from headers - Location of sample points are available in a separate list in form of Degrees minutes Seconds: transform to decimal representation and add to the dataset - Rename columns - Data is stored in .CSV format in a file with unique name at the designated folder (location information is provided in the code itself) 		

- Then this data can be consumed by Flume agent and inserted to the HDFS
<p>OUTPUT:</p> <p>The crawler generates CSV files. The field names correspond to the parameters presented in Table 8.2-22, including the transformations described above, in the crawler DESCRIPTION section.</p> <p>Date, sample_point_code, result_timestamp, conc_tot_colif, conc_fec_colif, conc_fec_strept, pH, physical_characteristics, NH4-N, So4-2, NO2-1, NO3-2, Cl-1, BOD, TOC, COD, TDS, TN, TKN, TP, Cd, Cr, Pb, Cu, Zn, Fe, Al, Mn, Ni, Latitude, Longitude</p> <p>2012-11-06, D1, 41226.0, 1500.0, 50, 250, 8.2, , , <0.5, 33, <0.5, 5.5, 46.5, , , , 14, 347.0, , , , 55.6, <2, <2, <17.5, <7.15, <5.4, 59.9, , , , 37.187222222222225, 30.790555555555557</p>

Table 5.2-29: Historical monthly data of water quality, flow and velocity (Antalya pilot)

DS_ID	D_ID	CODE
ANTA_ENV_WATERQUALITYFLOW_CITYOFANTALYA_MONTHLY	ANTA_ENV_WATERQUALITYFLOW_CITYOFANTALYA_MONTHLY	PiAnta-Env
<p>LINK:https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/anta_env_waterqualityflow_cityofantalya_monthly.py</p>		
<p>DATA ACQUISITION MODE: Automated – One time</p>		
<p>LANGUAGE AND LIBRARIES: Python 3.x, os, uuid, pandas, shutil</p>		
<p>PARAMETERS:</p>		
<p>DESCRIPTION:</p> <p>Prerequisites: File with data (anta_water_quality&flow_2018_2019.xlsx) need to be in the l_temp_path folder</p> <ul style="list-style-type: none"> - Open data information, data in the format specified in Table 8.2-23 - Separate headers and values special characters - Removes units and other unnecessary characters from headers - Transform floats to correct representation - Date to correct representation - Rename columns - Data is stored in .CSV format in a file with unique name at the designated folder (location information is provided in the code itself) - Then this data can be consumed by Flume agent and inserted to the HDFS 		
<p>OUTPUT:</p> <p>The crawler generates CSV files. The field names correspond to the parameters presented in Table 8.2-23, including the transformations described above, in the crawler DESCRIPTION section.</p> <p>Date, Zone, Latitude, Longitude, conc_BOD, conc_DO, conc_fec_colif, conc_fec_strept, conc_COD, pH, Total Nitrogen, conc_tot_colif, conc_P, water_volumetric_flow_rate, water_velocity</p> <p>2018-04-09, Sampling Point 1 - Kırkgöz, 37.0861, 30.58424, <5.00, 7.58, 800.0, 150.0, <5.00, 7.03, <1.32, 1100.0, , ,</p>		

Table 5.2-30: Antwerp Environmental Pressure data to measure sewage water levels, Water level and precipitation data (real-time)

DS_ID	D_ID	CODE
ANT_ENV_IMEC_PREC2018	ANT_ENV_IMEC_PREC2018_P1, ANT_ENV_IMEC_PREC2018_P2, ANT_ENV_IMEC_PREC2018_P3, ANT_ENV_IMEC_PREC2018_P4	PiAntw-Env
ANT_ENV_IMEC_OPENWATER	ANT_ENV_IMEC_OPENWATER_H1, ANT_ENV_IMEC_OPENWATER_H2, ANT_ENV_IMEC_OPENWATER_H3, ANT_ENV_IMEC_OPENWATER_H4, ANT_ENV_IMEC_OPENWATER_H5, ANT_ENV_IMEC_OPENWATER_H6	
ANT_ENV_IMEC_SEWER2018	ANT_ENV_IMEC_SEWER2018_D1, ANT_ENV_IMEC_SEWER2018_D2, ANT_ENV_IMEC_SEWER2018_D3	
LINK:		https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Environmental/ANT_ENV_IMEC
DATA ACQUISITION MODE: Automated - Scheduled		
LANGUAGE AND LIBRARIES: Java, gson, kafka, kafka-clients, geohash, log4j, slf4j, slf4j-api		
<p>PARAMETERS: Parameters should be provided with the .properties file.</p> <p>authorization_string = String value output_format = JSON or CSV data_folders_path = path to the folders to which the data will be stored kafka_use = true or false, whether data should be sent to kafka (experimental and requires further development)</p>		
<p>DESCRIPTION:</p> <p>Prerequisites: Access to sensor API from imec (authorization_string)</p> <ul style="list-style-type: none"> - Crawler executes the API provided from https://idlab-iot.tengu.io/api/v1/client-docs/ to get the sensor data - Data comes in the format specified in Table Open data information, data in the format specified in Table 8.2-17,18,19. - Location information is transformed from geohash to latitude and longitude, and these values are appended to each data row. - Renaming "time" to "timestamp_ms" to comply with the model - "SourceId" is renamed into "sensor_id" to comply with the model - For each sensor separate file is created where all the readings come. - Then this data can be consumed by Flume agent and inserted to the HDFS 		
<p>OUTPUT:</p> <p>The crawler generates a file for each sensor, named sensor_id_timestamp_ms.json. This file contains JSON objects. Field names correspond to those described in the Tables 8.2-17-19, including changes defined above, in the crawler DESCRIPTION section.</p> <pre>{ "timestamp_ms":1555508842118, "geohash":u155t81z8ckn, "latitude":51.2415296304971, "longitude":4.462890196591616, "sensor_id":lora.0004A30B001FA140, "value":0.0 }</pre>		

5.3 Economic Data

This subsection presents newly developed crawlers for economic data. The data sources corresponding to these crawlers are identified by the DS_ID field and can be seen in Appendix I.

Table 5.3-7: Several Economic Data Sources from Antalya

DS_ID	D_ID	CODE
ANTA_ECO_CITYOFANTALYA_VISITORTICKET_MONTHLY	ANTA_ECO_CITYOFANTALYA_VISITORTICKET_MONTHLY	PiAnta-Eco
ANTA_ECO_CITIOFANTALYA_OTOPARK_MONTHLY	ANTA_ECO_CITIOFANTALYA_OTOPARK_MONTHLY	
ANTA_ECO_CITYOFANTALYA_GENERALELECTRICBILL_MONTHLY	ANTA_ECO_CITYOFANTALYA_GENERAL ELECTRICBILL_MONTHLY	
ANTA_ECO_CITYOFANTALYA_WATERPOMPS_MONTHLY	ANTA_ECO_CITYOFANTALYA_WATERPOMPS_MONTHLY	
ANTA_ECO_CITYOFANTALYA_OPERATIONEMPLOYEESALARY_MONTHLY	ANTA_ECO_CITYOFANTALYA_OPERATIONEMPLOYEESALARY_MONTHLY	
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Economic/anta_eco_several_codes.py		
DATA ACQUISITION MODE: Automated – One time		
LANGUAGE AND LIBRARIES: Python 3.x, os, uuid , pandas		
PARAMETERS:		
DESCRIPTION: Prerequisites: File with data ('gelir-gider.xlsx') Data in the format specified in Tables 8.3-16,17,18,19,20 <ul style="list-style-type: none"> - Translate values from Turkish to English - Rename columns, filter out special characters, insert new fields. - Add row values for year and month when missing - Rename fields according to models - Data in the file is stored in different folders, according to their ds_id 		
OUTPUT: The crawler generates several CSV files. Field names correspond to those presented in Tables 8.3-16,17,18,19,20, including the changes described above, in the crawler DESCRIPTION section. For example, Year,Month,tickets_number,fee_revenue 2018,February,19631.0,98155.0		

Table 5.3-8: Duden waterfall shops rent earn (Antalya pilot)

DS_ID	D_ID	CODE
ANTA_ECO_CITIOFANTALYA_SHOPSRENTEARN_YEAR	ANTA_ECO_CITIOFANTALYA_SHOPSRENTEARN_YEAR	PiAnta-Eco
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Economic/anta_eco_citiofantalya_shopsrenteam_year.py		

DATA ACQUISITION MODE: Automated – One time
LANGUAGE AND LIBRARIES: Python 3.x, os, uuid , pandas
PARAMETERS:
<p>DESCRIPTION:</p> <p>Prerequisites: File with data ('antalya_cutler_all_data_ (version 1).xlsx) Data in the format specified in Table 8.3-22, including the changes decribed above, in the crawler DESCRIPTION section</p> <ul style="list-style-type: none"> - Rename fields according to models - Data in the file is stored in folder, according to its ds_id
<p>OUTPUT:</p> <p>The crawler generates several CSV files. Field names correspond to those presented in Table 8.3-22</p> <pre>Year,shop_rent 2016,164000.0</pre>

Table 5.3-9: Number of visitors in Cork

DS_ID	D_ID	CODE
CORK_ECO_VISITORS_DAILY	CORK_ECO_VISITORS_DAILY	PiCork-Eco
LINK: https://github.com/CUTLER-H2020/DataCrawlers/blob/master/Economic/cork_eco_visitors_daily.py		
DATA ACQUISITION MODE: Automated – One time		
LANGUAGE AND LIBRARIES: Python 3.x, os, uuid , pandas		
PARAMETERS:		
<p>DESCRIPTION:</p> <p>Prerequisites: File with data (numbers_cork.xlsx) Data in the format specified in Table 8.3-23</p> <ul style="list-style-type: none"> - Combine necessary sheets in the excel file - Rename fields according to models - Data in the file is stored in folder, according to its ds_id 		
<p>OUTPUT:</p> <p>The crawler generates several CSV files. Field names correspond to those presented in Table 8.3-23, including the changes decribed above, in the crawler DESCRIPTION section</p> <pre>Date,number_visitors,number_of_paying_visitors 2018-01-01,100,50</pre>		

5.4 Social Data

There are no newly developed crawlers for social data.

6. Challenges and future work

A number of challenges were faced during the development of the updated version of the data collection and preprocessing framework. Similarly to D3.2, we group these challenges under three categories:

Specification challenges. These challenges are related to gathering, confirming, and describing the data sources [1]. As it was already mentioned, CUTLER aims for four real-world pilots. Therefore, continuous refining of the data sources' suitability for the pilots is necessary. WP3 constantly checks and confirms the suitability of the data sources with the corresponding pilot partners and the leaders of the economic, social, and environmental analytics work packages. This information is also updated in the CUTLER Data catalogue, described in D3.1, as well as in WP3 deliverables, where new data sources and crawlers are presented.

Availability challenges. The majority of the issues related to availability of the data sources have been solved. However, there are still some availability issues with new data sources, e.g. new sensor installations in Antwerp. These will be resolved and presented in D3.5. One particular challenge to address in CUTLER is the legal qualification of data sources. A legal taxonomy of data sources was provided in D1.1 and the corresponding legal requirements were detailed in D1.2. However, further collaboration is required between WP1, WP4,5,6, and WP9 when actual data processing activities start in the first pilot phase. The goal is to ensure that all the legal requirements are in place regarding data processing.

Technical challenges. Since the CUTLER cloud infrastructure utilizes similar technology as the sandbox cluster in D3.2, porting seemed to be straightforward. In practice, we were faced with a number of challenges, including missing libraries, resolving version issues, configuring service execution, etc. These challenges have been successfully resolved in cooperation with the cloud provider (WP2). Moreover, the bridges between Hadoop and Elastic clusters were implemented.

Future work will focus on overcoming the challenges described above. Similarly to D3.2, we will also continue adding other available data sources, as well as extending data models and tackling privacy issues when required.

7. Conclusions

This deliverable presented an updated version of the CUTLER data collection and processing framework, focusing on porting the developed software into the CUTLER cloud. Further updates and modifications will be addressed in D3.5.

WP3 continued the maintenance of the data sources catalogue, in close cooperation with the city pilots and the leaders of the analytics WPs. New sources have been identified and new crawlers were developed for them, now available at the CUTLER GitHub repository.

Similar to D3.2, data cleaning, harmonization, and interoperability found to be the most challenging tasks. Approaches from D3.2 were applied here as well.

In addition, legal assessment of confirmed data sources was required and WP1 addressed this task based on early deliverables (D1.1, D1.2). However, further inspection of the data sources is needed to ensure that legal requirements are properly addressed within CUTLER. The WP1 leader will conduct this work in cooperation with W9,4,5,6 partners and will report it in D1.4.

D3.3 mainly focused on porting the Hadoop-based solutions of D3.2 and new developed ones to the actual CUTLER cloud infrastructure. The porting process was rather smooth, except for some technical challenges, including service configurations, libraries installation, or libraries version mismatches, etc., which were successfully resolved with support from WP2. D3.3 also demonstrated the connectivity between Hadoop and Elasticsearch clusters. Further experimentation will be discussed in D3.5.

8. Appendix 1 – Data sources description

This section describes the format of the newly available data sources.

8.1 Template for data description

For convenience, we use the same structure introduced in D3.2. Similarly to Section 5, each data source has a unique number (**DS_ID**). As a data source can supply different data, there is also a unique data number (**D_ID**). We associate each data source with a particular pilot city and corresponding data category (Environmental, Economic, Social) (**CODE**). For reference, see Table 5.1-1. To simplify finding the data source descriptions, sections 8.2, 8.3, and 8.4 of this Appendix contain tables with numbers continued from sections 8.2,8.3, and 8.4 of D3.2, respectively.

The codified table structure also includes a description of how the data can be actually retrieved from the data provider (**ORIGIN**) (see Table 8.1-2 for reference). The (**FORMAT**) field describes the original format of the data, e.g. HTML, PDF, CSV, etc. Finally, we provide a concrete description of the data (**DESCRIPTION**), additional information clarifying it (**METAINFORMATION**) and an example (**EXAMPLE**). Table 8.1-3 provides an example of data source description.

Table 8.1-2: Abbreviations used for coding the retrieval means of data sources, from D3.2

ORIGIN	Description
URL	Means that the data is embedded into Web page available within given URL
API	Means that the data is retrieved with provided API
FILE	Means that the data is supplied manually by a data provider as a file

Table 8.1-3: Data source description table – An example, from D3.2

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_MET_DAILY	CORK_ENV_MET_DAILY_1	URL	HTML	PiCork-Env
DESCRIPTION:				
Rainfall:float Max Temp:float Min Temp:float Grass Min Temp:float Mean Wind Speed: float Max Gust:float Sunshine:float				
METAINFORMATION:				
Rainfall-(mm) Max Temp -(°C) Min Temp - (°C) Grass Min Temp- (°C) Mean Wind Speed - (knots) Max Gust- (>= 34 knots) Sunshine – (hours)				
EXAMPLE:				
<pre><table class="table weather-table"> <thead><tr><th scope="col">Date</th></pre>				

```

<th scope="col">Rainfall<br> <span>(mm)</span></th>
<th scope="col">Max Temp<br> <span>(°C)</span></th>
<th scope="col">Min Temp<br> <span>(°C)</span></th>
<th scope="col">Grass Min Temp<br> <span>(°C)</span></th>
<th scope="col">Mean Wind Speed<br> <span>(knots)</span></th>
<th scope="col">Max Gust<br> <span>(&gt;= 34 knots)</span></th>
<th scope="col">Sunshine<br> <span>(hours)</span></th></tr>
</thead>
<tbody><tr><td>18/10/2018</td>
<td> 0.1 </td>
<td> 12.5</td>
<td> 4.6 </td>
<td> 0.3</td>
<td> 5.7 </td>
<td></td>
<td>7.8 </td></tr>
</tbody></table>

```

8.2 Environmental data

This subsection presents new available environmental data sources.

Table 8.2-21: Historic data of vehicles' speed for four (4) roads of Thessaloniki

DS_ID	D_ID	ORIGIN	FORMAT	CODE
THESS_ENV_IMET_SPEED_1 5MIN	THESS_ENV_IMET_SPEE D_15MIN_1, THESS_ENV_IMET_SPEE D_15MIN_2, THESS_ENV_IMET_SPEE D_15MIN_3, THESS_ENV_IMET_SPEE D_15MIN_4	URL	HTML	PiThess- Env
DESCRIPTION:				
Timestamp:date time PathID:float Name:string Value:float Samples:float				
METAINFORMATION:				
The data is retrieved by filling a form in a webpage from the. Ministry of Environment and Urbanism. Then, the data is downloadable in csv format through a link Timestamp- format YYYY-mm-dd HH:MM:ss,mms Name- string in Greek Value- (Km/h) value of speed Samples – number of samples				
EXAMPLE:				

	A	B	C	D	E	F	G	H
1	PathID,Name,Timestamp,Value,Samples							
2	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01,40,92							
3	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 00:15:00,43,99							
4	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 00:30:00,34,99							
5	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 00:45:00,30,99							
6	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 01:00:00,26,99							
7	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 01:15:00,23,99							
8	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 01:30:00,20,99							
9	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 01:45:00,18,99							
10	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 02:00:00,18,99							
11	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 02:15:00,18,99							
12	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 02:30:00,23,99							
13	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 02:45:00,24,99							
14	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 03:00:00,28,99							
15	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 03:15:00,30,99							
16	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 03:30:00,28,99							
17	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 03:45:00,30,99							
18	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 04:00:00,29,99							
19	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 04:15:00,35,99							
20	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 04:30:00,34,99							
21	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 04:45:00,36,99							
22	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 05:00:00,33,99							
23	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 05:15:00,34,99							
24	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 05:30:00,33,99							
25	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 05:45:00,34,99							
26	1,1. Εγνατία (Συντριβάνι - Πλατεία Δημοκρατίας),2017-01-01 06:00:00,36,99							

Table 8.2-22: Historic Water Quality Analysis Düden Creek (2012-2017)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ENV_CITYOFANTALYA2_MONTHLY	ANTA_ENV_CITYOFANTALYA2_MONTHLY_1, ANTA_ENV_CITYOFANTALYA2_MONTHLY_2, ANTA_ENV_CITYOFANTALYA2_MONTHLY_3, ANTA_ENV_CITYOFANTALYA2_MONTHLY_4 ANTA_ENV_CITYOFANTALYA2_MONTHLY_5 ANTA_ENV_CITYOFANTALYA2_MONTHLY_6 ANTA_ENV_CITYOFANTALYA2_MONTHLY_7 ANTA_ENV_CITYOFANTALYA2_MONTHLY_8	FILE	EXCEL	PiAnta-Env
DESCRIPTION:				
TARİH: date NKT: string Sonuç Tarihi: timestamp "Toplam Koliform":int "Fekal Koliform":int "Fekal Streptokok":int "pH":float "Renk, Koku Bulanıklık":string				

```
"NH4-N":string(float)
"SO4-2":string(float)
"NO2-1":string(float)
"NO3-2":string(float)
"Cl-1":string(float)
"BOİ ":string(float)
"T. Org. Krb (TOC)":string(float)
"KOİ":string(float)
"TDS":string(float)
"Top. N":string(float)
"Kjeldahl Azotu (TKN)":string(float)
"Top. P":string(float)
"Cd":string(float)
"Cr":string(float)
"Pb":string(float)
"Cu":string(float)
"Zn":string(float)
"Fe":string(float)
"Al":string(float)
"Mn":string(float)
"Ni":string(float)
```

METAINFORMATION:

The data is retrieved from an excel file. Note that values defined above as `string(float)` imply that the value can be "<VAL" or VAL, being VAL a float

Locations are given in a separate list, as Degrees Minutes and Seconds representation

TARİH – date m/dd/YYYY

NKT – Dx, x=1..8

Sonuç Tarihi – timestamp?? (Not used)

Toplam Koliform adet/100ml – Total coliform

Fekal Koliform adet/100ml – Fecal coliform

Fekal Streptokok adet/100ml – fecal Streptokok

pH

Renk, Koku Bulanıklık – Physical characteristics

NH4-N (mg/L)

SO4-2 (mg/l) – SO4

NO2-1 (mg/l) – NO2

NO3-2 (mg/l) – NO3

Cl-1 (mg/l) – Cl

BOİ (mg/l) – Biological Oxygen Demand (BOD)

T. Org. Krb (TOC) (mg/L) – Total Organic Carbon (TOC)

KOİ (mg/l) – Chemical Oxygen Demand (COD)

TDS (µs/cm) – Total Dissolved Solids (TDS)

Top. N (mg/l) – Total Nitrogen (TN)

Kjeldahl Azotu (TKN) (mg/L) – Total Kjeldahl Nitrogen (TKN)

Top. P (mg/l) – Total Phosphorus (TP)

Cd (µg/l) – Cd

Cr (µg/l) – Cr

Pb (µg/l) – Pb

Cu (µg/l) – Cu

n (µg/l) – Zn

Fe (µg/l) – Fe

Al (µg/l) – Al

Mn (µg/l) – Mn

Ni (µg/l) – Ni

EXAMPLE:

	A	B	C	D	E	F	G	H	I	J	K
1	TARİH	NKT	Sonuç Tari	Toplam Kc	Fekal Koli	Fekal Stre	pH(b1)	Renk, Kok	NH4-Nmg	SO4-2mg/	NO2-1mg, NO3
2	5/2/2012	D1	41039	2000	150	200	7.9				
3	5/2/2012	D2	41039	3000	90	80	7.7				
4	5/2/2012	D3	41039	4500	350	500	7.5				
5	5/2/2012	D4	41039	6000	300	1000	7.4				
6	5/2/2012	D5	41039	7000	250	1200	7.3				
7	5/2/2012	D6	41039	4000	350	400	7.4				
8	5/2/2012	D7	41039	5000	600	1100	7.3				
9	5/2/2012	D8	41039	2500	20	500	7.2				
10	6/5/2012	D1	41072	2500	90	80	7.8	<0,5		27.9	<0,5
11	6/5/2012	D2	41072	2700	0	50	7.8	<0,5		27.3	<0,5
12	6/5/2012	D3	41072	1500	30	50	7.7	<0,5		32.1	<0,5
13	6/5/2012	D4	41072	2000	250	90	7.6	<0,5		33.4	<0,5
14	6/5/2012	D5	41072	1800	0	20	7.5	<0,5		29.8	<0,5
15	6/5/2012	D6	41072	2200	0	30	7.5	<0,5		29.5	<0,5
16	6/5/2012	D7	41072	2300	0	0	7.4	<0,5		29.6	<0,5
17	6/5/2012	D8	41072	1400	50	90	7.9	<0,5		29.5	<0,5
18	7/3/2012	D1	41099	2000	0	300	7.5	<0,5		22.6	<0,5
19	7/3/2012	D2	41099	2500	0	150	7.8	<0,5		24	<0,5
20	7/3/2012	D3	41099	1800	0	250	7.6	<0,5		23.9	<0,5
21	7/3/2012	D4	41099	1500	0	170	7.5	<0,5		21.8	<0,5
22	7/3/2012	D5	41099	2700	0	200	7.4	<0,5		24.7	<0,5
23	7/3/2012	D6	41099	1700	0	220	7.8	<0,5		25.5	<0,5
24	7/3/2012	D7	41099	1750	0	120	7.6	<0,5		22.3	<0,5
25	7/3/2012	D8	41099	1900	0	25	7.5	<0,5		21.9	<0,5
26	8/7/2012	D1	41135	2000	50	-	8.1	n/a	n/a	n/a	n/a
27	8/7/2012	D2	41135	4000	0	-	7.9	n/a	n/a	n/a	n/a

Table 8.2-23: Historical monthly data of water quality, flow and velocity in Duden Brook

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ENV_WATERQUALITY FLOW_CITYOFANTALYA_MONTHLY	ANTA_ENV_WATERQUALITY FLOW_CITYOFANTALYA_MONTHLY_1,	FILE	EXCEL	PiAnta-Env
DESCRIPTION:				
DATE: date ZONE: string Lat: float Long: float BOD: string(float) Dissolved Oxygen: float Fecal coliform: int Fecal Streptococcus: int COD: string(float) pH:float Total Nitrogen: string(float) Total Phosphorus: string(float) Volumetric Flow: float Water velocity: float				
METAINFORMATION:				
The data is retrieved from an excel file. Note that values defined above as string(float) imply that the value can be “<VAL” or VAL, being VAL a float DATE – date dd/mm/yyyy ZONE – values Sampling Point 1 – Kırkgöz, Sampling Point 2 – Kepezüstü, Sampling Point 3 – Şelaleüstü, Sampling Point 4 - Şelale Havuzu, Sampling Point 5 - Cırnık Köprüsü,Sampling				

Point 6 - ddenin Denize Döküldüğü Yer
 Lat, Lon
 BOD – mg/L (DL:5mg/L)
 Dissolved Oxygen – mg/L
 Fecal coliform – CFU/100mL
 Fecal Streptococcus – CFU/100mL
 COD – mg/L (DL:5mg/L)
 pH
 Total Nitrogen – mg/L (DL:1,32mg/L)
 Total Phosphorus – mg/L (DL:0.025mg/L)
 Volumetric Flow – m3/s
 Water velocity – m/s

EXAMPLE:

	DATE	ZONE	Lat	Long	BOD (mg/L) (DL: 5mg/L)	Dissolved Oxygen (mg/L)	Fecal coliform (CFU/100ml)	Fecal Streptococcus (CFU/100ml)	COD (mg/L) (DL: 5mg/L)	pi
1										
2	09/04/2018	Sampling Point 1 - Kirkgöz	37.0861	30.58424	<5,00	7.58	800	150	<5,00	
3	09/04/2018	Sampling Point 2 - Kepezüstü	36.96773	30.62151	<5,00	7.28	140	120	<5,00	
4	09/04/2018	Sampling Point 3 - Şelaleüstü	36.96434	30.72705	<5,00	7.51	100	90	6.2	
5	09/04/2018	Sampling Point 4 - Şelale Havuzu	36.96434	30.72705	<5,00	7.4	80	90	6.6	
6	09/04/2018	Sampling Point 5 - Cirmik Köprüsü	36.90479	30.76644	<5,00	7.02	50	70	<5,00	
7	09/04/2018	Sampling Point 6 - ddenin Denize Döküldüğü Yer	36.85144	30.78326						
8	09/05/2018	Sampling Point 1 - Kirkgöz	37.0861	30.58424	<5,00	5.33	2000	2600	8.9	
9	09/05/2018	Sampling Point 2 - Kepezüstü	36.96773	30.62151	<5,00	4.94	1000	2400	5.8	
10	09/05/2018	Sampling Point 3 - Şelaleüstü	36.96434	30.72705	<5,00	4.78	2100	2300	<5,00	
11	09/05/2018	Sampling Point 4 - Şelale Havuzu	36.96434	30.72705	<5,00	4.32	1000	2500	<5,00	
12	09/05/2018	Sampling Point 5 - Cirmik Köprüsü	36.90479	30.76644	<5,00	4.28	900	2800	<5,00	
13	09/05/2018	Sampling Point 6 - ddenin Denize Döküldüğü Yer	36.85144	30.78326						
14	11/06/2018	Sampling Point 1 - Kirkgöz	37.0861	30.58424	<5,00	4.4	80	1500	8.9	
15	11/06/2018	Sampling Point 2 - Kepezüstü	36.96773	30.62151	<5,00	3.5	200	1800	<5,00	
16	11/06/2018	Sampling Point 3 - Şelaleüstü	36.96434	30.72705	<5,00	3.64	900	900	<5,00	
17	11/06/2018	Sampling Point 4 - Şelale Havuzu	36.96434	30.72705	<5,00	3.3	1000	3000	<5,00	
18	11/06/2018	Sampling Point 5 - Cirmik Köprüsü	36.90479	30.76644	<5,00	2.64	150	2000	<5,00	
19	11/06/2018	Sampling Point 6 - ddenin Denize Döküldüğü Yer	36.85144	30.78326						
20	18/06/2018	Sampling Point 1 - Kirkgöz	37.0861	30.58424						
21	18/06/2018	Sampling Point 2 - Kepezüstü	36.96773	30.62151						
22	18/06/2018	Sampling Point 3 - Şelaleüstü	36.96434	30.72705						
23	18/06/2018	Sampling Point 4 - Şelale Havuzu	36.96434	30.72705						
24	18/06/2018	Sampling Point 5 - Cirmik Köprüsü	36.90479	30.76644						
25	18/06/2018	Sampling Point 6 - ddenin Denize Döküldüğü Yer	36.85144	30.78326						
26	29/06/2018	Sampling Point 1 - Kirkgöz	37.0861	30.58424						
27	29/06/2018	Sampling Point 2 - Kepezüstü	36.96773	30.62151						
28	29/06/2018	Sampling Point 3 - Şelaleüstü	36.96434	30.72705						
29	29/06/2018	Sampling Point 4 - Şelale Havuzu	36.96434	30.72705						

Table 8.2-24: Historic data on Special Area of Conservation in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_SAC_2015	CORK_ENV_EPA_SAC_2015	URL	Shapefile	PiCork-Env
DESCRIPTION:				
Special Areas of Conservation (SAC) 2019 geometry.coordinates: Array nx2 of floats geometry.type:_string properties.COUNTY: string properties.HA: float properties.OBJECTID:int properties.SITECODE: int properties.SITE_NAME: string properties.Shape_Area: float properties.Shape_Leng:float properties.SourcScale:string properties.Source_CRS:string properties.URL:string properties.VERSION:string				

```
type:string
SAC_Conservation_Area_offshore_2015
geometry.coordinates: Array nx2 of floats
geometry.type:_string
properties.Centroid_X:float
properties.Centroid_Y:float
properties.LAEA_Area:float
properties.N2k_Code:String
properties.SITECODE:int
properties.SITE_NAME:string
properties.URL:string
type:string
```

METAINFORMATION:

The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: <http://www.opengis.net/def/crs/EPSSG/0/29902> (TM65 / Irish Grid)

The data comes in two set of shapefiles: one for SAC_conservation_area_2019 and other for SAC_Conservation_Area_offshore_2015

The data for SAC_conservation_area_2019 shows information regarding geometry (coordinates, type), properties (COUNTY, HA, OBJECTID, SITECODE, SITE_NAME, Shape_Area, Shape_Leng, SourcScale, Source_CRS, URL, VERSION) and type.

geometry.coordinates – format (((638597.6459999997, 823177.2037000004),...))

geometry.type – _possible values “Polygon”, “MultiPolygon”

properties.COUNTY – string

properties.HA – (ha) hectares; example 475.5865

properties.OBJECTID – integer

properties.SITECODE – format 0000XX

properties.SITE_NAME – name of the SAC

properties.Shape_Area – (m²)Example 4.755865e+07

properties.Shape_Leng – (m) Example 4.966770e+07

properties.SourcScale – scale representation format int:int as 1:10560

properties.Source_CRS –

properties.URL – url format <http://www.npws.ie/protected-sites/sac/<SITECODE>>

properties.VERSION – format X.XX as 1.02

type – value “Feature”

The data for SAC_Conservation_Area_offshore_2015 shows information for geometry (coordinates, type), properties (Centroid_X, Centroid_Y, LAEA_Area, N2k_Code, SITECODE, SITE_NAME, URL), type

geometry.coordinates – format (-11.861590487785099, 51.48900420878227)

geometry.type – _ possible values “Polygon”, “MultiPolygon”

properties.Centroid_X – example -11.861590487785099

properties.Centroid_Y – example 51.48900420878227

properties.LAEA_Area – example 41090.4

properties.N2k_Code – example “IE0002327”

properties.SITECODE – example 002327

properties.SITE_NAME – name of SAC

properties.URL – url format <http://www.npws.ie/protected-sites/sac/<SITECODE>>

type – value “Feature”

EXAMPLE:



Table 8.2-25: Historic data on Natural Heritage Areas (NHAs) in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_NHA_2012	CORK_ENV_EPA_NHA_2012	URL	Shapefile	PiCork-Env

DESCRIPTION:

National Heritage Areas (NHAs)
 geometry.coordinates: Array nx2 of floats
 geometry.type: string
 properties.COUNTY: string
 properties.HA: float
 properties.OBJECTID: int
 properties.SITECODE: int
 properties.SITE_NAME: string
 properties.Shape_Area: float
 properties.Shape_Leng: float
 properties.SourcScale: string
 properties.Source_CRS: string
 properties.URL: string
 properties.VERSION: string
 type: string

METAINFORMATION:

The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: <http://www.opengis.net/def/crs/EPSSG/0/29902> (TM65 / Irish Grid)

The data for NHA_ shows information regarding geometry (coordinates, type), properties (COUNTY, HA, OBJECTID, SITECODE, SITE_NAME, Shape_Area, Shape_Leng, SourcScale, Source_CRS, URL, VERSION) and type.

geometry.coordinates – format [[[624837.2302999999, 821170.2600999996)...]]]

geometry.type – possible values “Polygon”, “MultiPolygon”

properties.COUNTY – county code, two characters. Example: co

properties.HA – (ha) hectares 674.6777

properties.OBJECTID –

properties.SITECODE – format 6digits
 properties.SITE_NAME – name of the NHA
 properties.Shape_Area – (m2) Example 6.746777e+06
 properties.Shape_Leng – (m) Example 14944.927408
 properties.SourcScale – format int:int as 1:10560
 properties.Source_CRS – Coordinate Reference System, value“Irish Grid”
 properties.URL – url format http://www.npws.ie/protected-sites/nha/<objectid>
 properties.VERSION – format X.XX as 1.02
 type – string value “Feature”

EXAMPLE:




Table 8.2-26: Historic data on Special Protection Areas in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_SPA_2015	CORK_ENV_EPA_SPA_2015	URL	Shapefile	PiCork-Env
DESCRIPTION:				
Special Protection Areas (SPAs) geometry.coordinates: Array nx2 of floats geometry.type: _string properties.COUNTY: string properties.HA: float properties.OBJECTID:int properties.SITECODE: int properties.SITE_NAME: string properties.Shape_Area: float properties.Shape_Leng:float properties.SourcScale:string properties.Source_CRS:string properties.URL:string properties.VERSION:string				

type:string

METAINFORMATION:

The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: <http://www.opengis.net/def/crs/EPSSG/0/29902> (TM65 / Irish Grid)

The data for SPA_ shows information regarding geometry (coordinates, type), properties (COUNTY, HA, OBJECTID, SITECODE, SITE_NAME, Shape_Area, Shape_Leng, SourcScale, Source_CRS, URL, VERSION) and type.

geometry.coordinates – format [[[695763.6764000002, 599104.6137000006)...]]

geometry.type – possible values “Polygon”, “MultiPolygon”

properties.COUNTY – county code, two characters. Example: ke

properties.HA – (ha) hectares Example 870.616711

properties.OBJECTID – Example 004002

properties.SITECODE – format 6digits

properties.SITE_NAME – name of the SPA

properties.Shape_Area – (m²) Example 8.706167e+06

properties.Shape_Leng – (m) Example 13993.389683

properties.SourcScale – format int:int as 1:10560

properties.Source_CRS – Coordinate Reference System “Irish Grid”

properties.URL – url format: <http://www.npws.ie/protected-sites/spa/<objectid>>

properties.VERSION – format X.XX as 1.02

type – string value “Feature”

EXAMPLE:

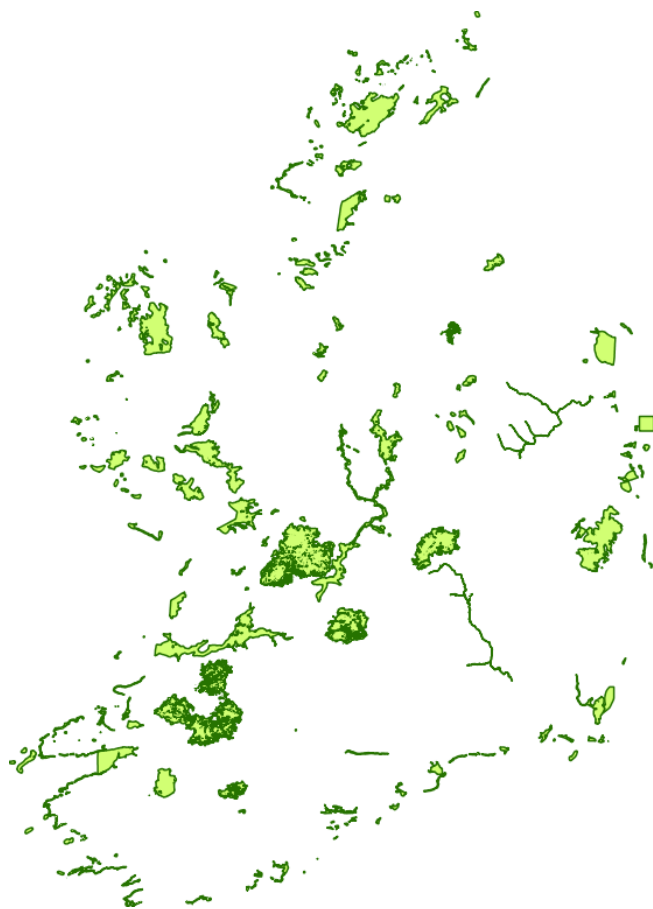


Table 8.2-27: Historic data on Groundwater Waterbody Status in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_GWWFD_2 0102015	CORK_ENV_EPA_GWWF D_20102015	URL	Shapefile	PiCork- Env
<p>DESCRIPTION:</p> <p>Groundwater Waterbody Status geometry.coordinates: Array nx2 of floats geometry.type: string properties.AreaHectar: float properties.AreaKm2: float properties.CATEGORY: string properties.Change: string properties.DESCRIP: string properties.DIST_CD: string properties.DateChange properties.EU_CD properties.Easting properties.EdenCode properties.FULL_TYPE properties.HORIZON properties.HorzType properties.INS_BY properties.INS_WHEN properties.LAT properties.LON properties.LocalAutho properties.MS_CD properties.NAME properties.Northing properties.OutOfRBD properties.ParentWate properties.ProtectedA properties.REGION_CD properties.Shape_STAr properties.Shape_STLe properties.Transbound properties.WaterBodyG type</p>				
<p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSG/0/29902 (TM65 / Irish Grid)</p> <p>The data for Ground Waterbody Status show information regarding geometry (coordinates, type), properties (AreaHectar, AreaKm2, CATEGORY, Change, DESCRIP, DIST_CD, DateChange, EU_CD, Easting, EdenCode, .FULL_TYPE, HORIZON, HorzType, INS_BY, .INS_WHEN, LAT, LON, LocalAutho, _CD, NAME, Northing, OutOfRBD, ParentWate, ProtectedA, REGION_CD, Shape_STAr, Shape_STLe, Transbound, WaterBodyG) and type.</p> <p>geometry.coordinates – format [(((299691.5284000002, 250219.99980000034)..)] geometry.type – possible values “Polygon”, “MultiPolygon” properties.AreaHectar – (ha) example 1644.026662 properties.AreaKm2 – (Km2) example 16.4402667 properties.CATEGORY – value “Ground Waterbody” properties.Change – descriptive string; example “Horizon elements removed” properties.DESCRIP – descriptive string; example “Productive fissured bedrock”</p>				

properties.DIST_CD – District Code “Eastern” “Western”
properties.DateChange YYYY-mm-dd
properties.EU_CD – European Code; format “IE_EA_G_031”
properties.Easting – (m) example 295300.15
properties.EdenCode – <Null>
properties.FULL_TYPE – examples “FI”, “PP”
properties.HORIZON – empty
properties.HorzType – value “Unknown”
properties.INS_BY – RG EPA
properties.INS_WHEN – 2015-03-20
properties.LAT – -6.565143
properties.LON – 53.484406
properties.LocalAutho – example 2300
properties.MS_CD – example EA_G_031
properties.NAME – example Dunshaughlin
properties.Northing – 249200.06
properties.OutOfRBD – No
properties.ParentWate – Always empty
properties.ProtectedA – value “Unknown”
properties.REGION_CD – 17
properties.Shape_STAr – 1.644027e+07
properties.Shape_STLe – 2.600907e+04
properties.Transbound – 2
properties.WaterBodyG – Always empty
type – value “Feature”

EXAMPLE:

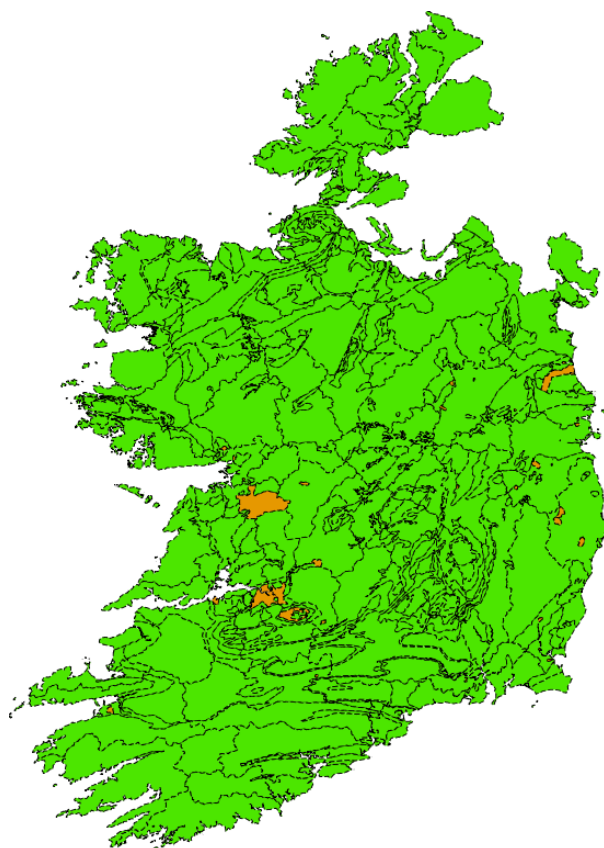


Table 8.2-28: Historic data on Lake Waterbody Status in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_LWFD_2010 2015	CORK_ENV_EPA_LWFD_ 20102015	URL	Shapefile	PiCork- Env
<p>DESCRIPTION:</p> <p>Lake Waterbody Status geometry.coordinates: Array nx2 of floats geometry.type: string properties.ARTIFICIAL: string properties.AreaHectar: float properties.AreaKm2: float properties.BASIN_CD: string properties.BasinSubCo: string properties.Change: string properties.DIST_CD: string properties.DateChange: date properties.EU_CD: string properties.Easting: float properties.Hydrometri:String properties.INS_BY: String properties.LAT:float properties.LON:float properties.LocalAutho: string properties.MODIFIED: string properties.MS_CD: string properties.NAME: string properties.Northing: float properties.REGION_CD: integer properties.SEG_CD: string properties.SYSTEM: char properties.TYPE: integer properties.WaterManag: string type:string</p>				
<p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSSG/0/29902 (TM65 / Irish Grid)</p> <p>The data for Lake Waterbody status show information regarding geometry (coordinates, type), properties (ARTIFICIAL, AreaHectar, AreaKm2, BASIN_CD, BasinSubCo, Change, DIST_CD, DateChange, EU_CD, Easting, INS_BY, LAT, LON, LocalAutho, MODIFIED, MS_CD, NAME, Northing, REGION_CD, SEG_CD, SYSTEM, TYPE, WaterManag) and type.</p> <p>geometry.coordinates – format [[(256045.06319999974, 270797.1917000003)..]] geometry.type – possible values “Polygon”, “MultiPolygon” properties.ARTIFICIAL – values “Yes” “No” properties.AreaHectar – (ha) example 4.150000 properties.AreaKm2 – (km2) example 0.041534 properties.BASIN_CD – example format “IE_EA_159” properties.BasinSubCo – example “Boyne” properties.Change – description; example “Topology Verified” properties.DIST_CD – District Code example “Eastern” properties.DateChange – format YYYY-mm-dd properties.EU_CD – example format “IE_EA_07_178” properties.Easting – (m) example “256056.40” properties.Hydrometri – example “Boyne”</p>				

properties.INS_BY – example “PMills”
 properties.LAT – example 53.685380
 properties.LON – example -7.152360
 properties.LocalAutho – example “MEATH COUNTY COUNCIL”
 properties.MODIFIED – values “Yes” “No”
 properties.MS_CD – example format “EA_07_178”
 properties.NAME – example “Glass”, “Ben”
 properties.Northing – (m) example 270954.70
 properties.REGION_CD – example 17
 properties.SEG_CD – Lake Segment Code; example “07_178”
 properties.SYSTEM – example B
 properties.TYPE – example 0
 properties.WaterManag – example format “IE_EA_Deel”
 type: “Feature”

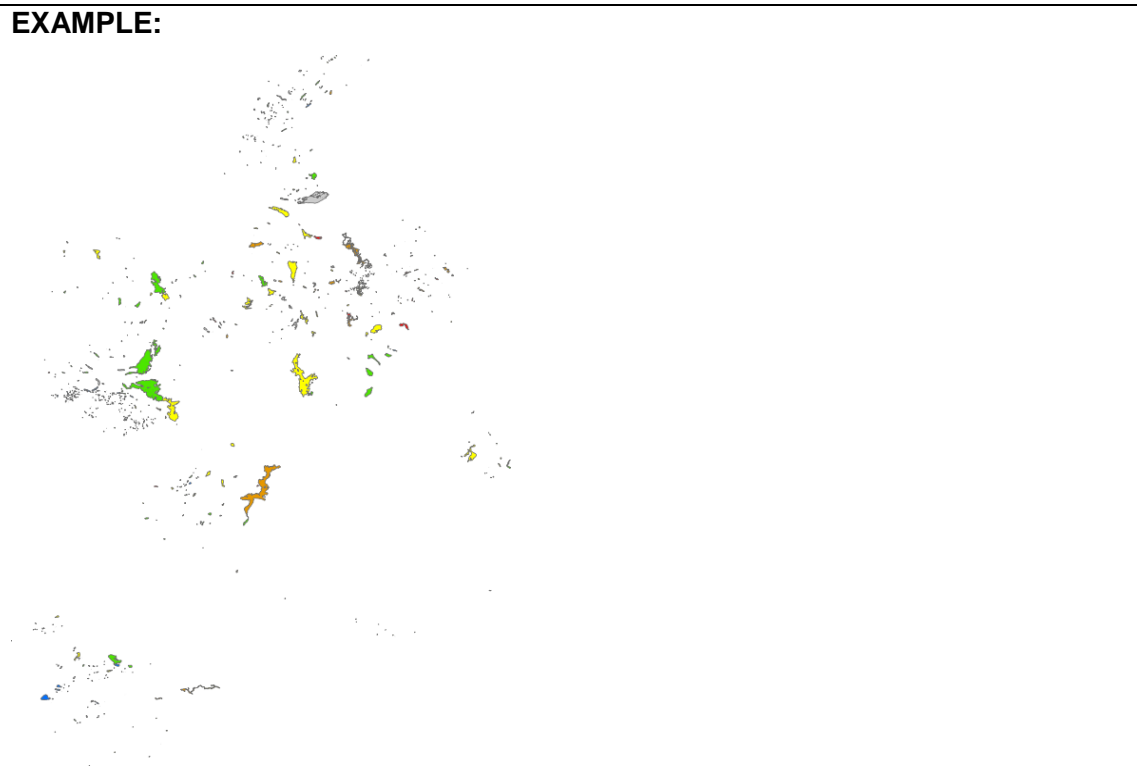


Table 8.2-29: Historic data on River Waterbody Status in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_RWFD_20102015	CORK_ENV_EPA_RWFD_20102015	URL	Shapefile	PiCork-Env
DESCRIPTION:				
River Waterbody Status geometry.coordinates: Array nx2 of floats geometry.type: string properties.AREAKM2: float properties.BASIN_CD: string properties.Change: string properties.DateChange: date properties.EU_CD: string properties.Ms_CD: string properties.NAME: string				

```
properties.ORDER_: integer  
properties.RBD: string  
properties.Shape_Area: float  
properties.Shape_Leng: float  
type: string
```

METAINFORMATION:

The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: <http://www.opengis.net/def/crs/EPSG/0/29902> (TM65 / Irish Grid)

The data for River Waterbody status show information regarding geometry (coordinates, type), properties (AREAKM2, BASIN_CD, Change, DateChange, EU_CD, Ms_CD, NAME, ORDER_, RBD, Shape_Area, Shape_Leng) and type.

```
geometry.coordinates – format [[(261880.00000042003, 277879.9999445081)..]]  
geometry.type – possible values “Polygon”, “MultiPolygon”  
properties.AREAKM2 – (km2) example 10.488450  
properties.BASIN_CD – example “159 Boyne”  
properties.Change – descriptive text  
properties.DateChange – format YYYY-mm-dd  
properties.EU_CD – European code, format “IE_EA_07A010020”  
properties.Ms_CD – code, format “EA_07A010020”  
properties.NAME – format “ATHBOY_010”  
properties.ORDER_ – example 3  
properties.RBD – River Basin Districts example “IEEA”  
properties.Shape_Area – (m2) example 1.048845e+07  
properties.Shape_Leng – (m) example 19270.000146  
type – “Feature”
```

EXAMPLE:

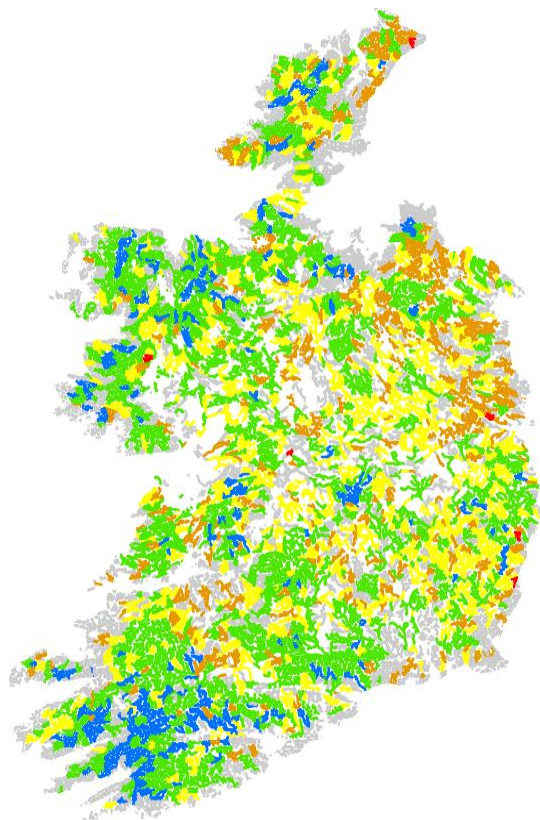


Table 8.2-30: Historic data on Coastal Waterbody Status in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_CWFD_20102015	CORK_ENV_EPA_CWFD_20102015	URL	Shapefile	PiCork-Env
<p>DESCRIPTION:</p> <p>Coastal Waterbody Status geometry.coordinates: Array nx2 of floats geometry.type: string properties.AdjacentHy: string properties.AreaHectar: float properties.AreaKm2: float properties.Artificial: string properties.Category: string properties.Change: string properties.DEPTH_CAT: integer properties.DIST_CD: string properties.DateChange: date properties.DonorWater: string properties.EDENEntity: integer properties.EDENLACode: integer properties.EU_CD: string properties.Easting: float properties.INS_BY: string properties.INS_WHEN: date properties.Intercalib: integer properties.LAT: float properties.LON: float properties.LocalAutho: string properties.MS_CD: string properties.Modified: string properties.NAME: string properties.Northing: float properties.Processing: string properties.ProtectedA: string properties.REGION_CD: integer properties.SALINITY: character properties.SYSTEM: character properties.SubsitePre: string properties.TIDAL: properties.Type: string properties.WISERefere: string type: string</p>				
<p>META INFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSSG/0/29902 (TM65 / Irish Grid) geometry.coordinates – example (((326359.7548000002, 164223.9308000002)) geometry.type – values “Polygon” properties.AdjacentHy – example “Ovoca-Vartry” properties.AreaHectar – (ha) example 4694.715921 properties.AreaKm2 – (Km2) example 46.947159 properties.Artificial – values “Yes”, “No” properties.Category – descriptive string; example “Coastal Waterody” properties.Change – descriptive string; example “Topology verified” properties.DEPTH_CAT – integer; example 0 properties.DIST_CD – District Code example “Eastern”</p>				

properties.DateChange – date format YYYY-mm-dd
properties.DonorWater – example “IE_EA_100_0000”
properties.EDENEntity – integer; example 1020
properties.EDENLACode – integer; example 3400
properties.EU_CD – example “IE_EA_140_0000”
properties.Easting – example 328560.88
properties.INS_BY – string example “EPA”
properties.INS_WHEN – date format YYYY-mm-dd
properties.Intercalib – integer; example 0
properties.LAT – float; example 52.809630
properties.LON – float; example -6.112080
properties.LocalAuth – string “WICKLOW COUNTY COUNCIL”
properties.MS_CD – string; example “EA_140_0000”
properties.Modified – string, values “Yes”, “No”
properties.NAME – string; example “Southwestern Irish Sea - Brittas Bay (HA 10)”
properties.Northing – float; example 175045.21
properties.Processing – string; example “Active”
properties.ProtectedA – string; values “Yes”, “No”
properties.REGION_CD – integer; example 1
properties.SALINITY – character; example “E”
properties.SYSTEM – character example “B”
properties.SubsitePre – string; example “AV1”
properties.TIDAL – always empty
properties.Type – Strings separated by commas; example “Euhaline, Mesotidal, Moderately Exposed”
properties.WISERefere – value “Unknown” string
type – string

EXAMPLE:

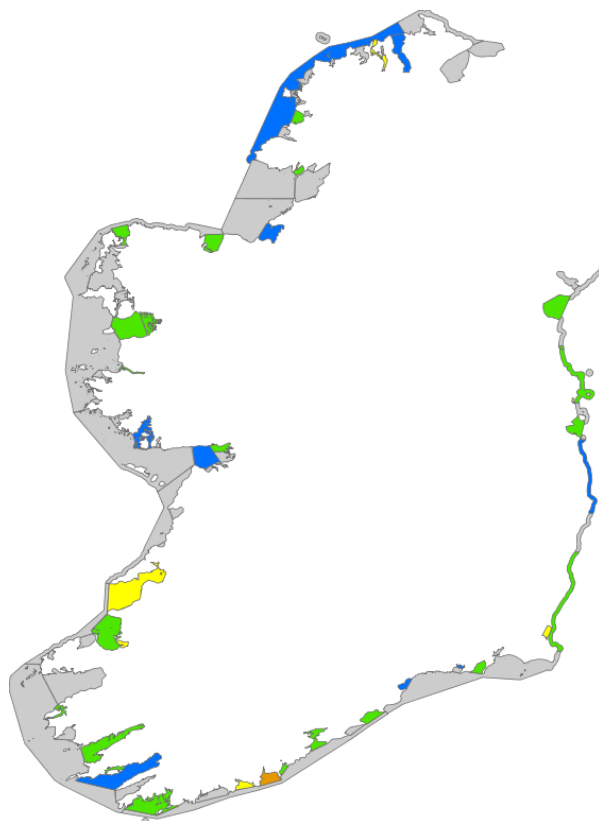


Table 8.2-31: Historic data on Transitional Waterbody Status in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_EPA_TWFD_20102015	CORK_ENV_EPA_TWFD_20102015	URL	Shapefile	PiCork-Env
<p>DESCRIPTION:</p> <p>Transitional Waterbody Status geometry.coordinates: Array nx2 of floats geometry.type: string properties.ARTIFICIAL: string properties.AdjacentHy: string properties.AreaHectar: float properties.AreaKm2: float properties.CATEGORY: string properties.Change: string properties.DEPTH: float properties.DIST_CD: string properties.DateChange: date properties.DonorWater: string properties.EDENEntity: integer properties.EDENLACode: integer properties.EU_CD: string properties.Easting: float properties.INS_BY: string properties.INS_WHEN: date properties.Intercalib: integer properties.LAT: float properties.LOCALAUTHO: string properties.LON: float properties.MODIFIED: string properties.MS_CD: string properties.NAME: string properties.Northing: float properties.Processing: string properties.ProtectedA: string properties.REGION_CD: integer properties.SALINITY: character properties.SYSTEM: character properties.SubsitePre: string properties.TIDAL: string properties.TWCategory: string properties.TYPE: string properties.WiseRefere: string type: string</p>				
<p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSG/0/29902 (TM65 / Irish Grid)</p> <p>geometry.coordinates – example (((325187.67310000025, 252530.99990000017)...)) geometry.type – values “Polygon” properties.ARTIFICIAL – values “Yes” “No” properties.AdjacentHy – string properties.AreaHectar – (ha) example 304.642626 properties.AreaKm2 – (km2) example 3.046426 properties.CATEGORY – descriptive string example “Transitional Waterbody” properties.Change – descriptive string example “Topology verified” properties.DEPTH – example 0.0</p>				

properties.DIST_CD – District Code example “Eastern”
properties.DateChange – format YYYY-mm-dd
properties.DonorWater – always empty
properties.EDENEntity – example 1006
properties.EDENLACode – example 0900
properties.EU_CD – “IE_EA_050_0100”
properties.Easting – 323403.11
properties.INS_BY – example “EPA”
properties.INS_WHEN – format YYYY-mm-dd
properties.Intercalib – example 0
properties.LAT – example 53.506090
properties.LOCALAUTHO – example “FINGAL COUNTY COUNCIL”
properties.LON – example -6.147980
properties.MODIFIED – values “Yes”, “No”
properties.MS_CD – format “EA_050_0100”
properties.NAME – example “Rogerstown Estuary”
properties.Northing – float example 252134.02
properties.Processing – string; example “Active”
properties.ProtectedA – values “Yes”, “No”
properties.REGION_CD – integer; example 1
properties.SALINITY – character; example “P”
properties.SYSTEM – character; example “A”
properties.SubsitePre – string; example “RG1”
properties.TIDAL – value “No Data”
properties.TWCategory – value “Unknown”
properties.TYPE – Strings separated by commas; example “Meso or Polyhaline, Strongly Mesotidal, Sheltered”
properties.WiseRefere – value “Unknown”
type – value “Feature”

EXAMPLE:



Table 8.2-32: Flood maps for Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_OPW_FLOODS_2016	CORK_ENV_OPW_FLOODS_2016	URL	Shapefile	PiCork-Env
<p>DESCRIPTION:</p> <pre> geometry.coordinates: Array nx2 of floats geometry.type: string properties.AEP: float properties.EXT_ID: string properties.ModelCode: Integer properties.RBDCCode: char properties.RunType: char properties.Scenario: Char properties.SourceCode: Char properties.Status: Char properties.TypeCode: string properties.UoMCode:integer properties.goid: string type: string) </pre> <p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSSG/0/29902 (TM65 / Irish Grid)</p> <pre> geometry.coordinates – format (((36419.41000000015, 72901.60999999994)..)) geometry.type – value “Polygon” properties.AEP – Annual Exceedance Probability example 0.001 properties.EXT_ID – alphanumeric string which is a combinations of the AEP and the following codes and types; example “I2241CC001” properties.ModelCode – example 41 properties.RBDCCode – example “I” properties.RunType – example “D” properties.Scenario – example “C” properties.SourceCode – example “C” properties.Status – example “F” properties.TypeCode – example “EX” which means Data/Map type “Extent” properties.UoMCode – example “22” properties.goid – alphanumeric; example “57E9F9A0EDBDCDE0235BB36A70C5E248” type – value “Feature” </pre> <p>Notes: Source Types - c for coastal, p pluvial, f for fluvial and w wave overtopping, Scenarios- c for current, m for mid range and h for high end</p> <p>EXAMPLE:</p>				

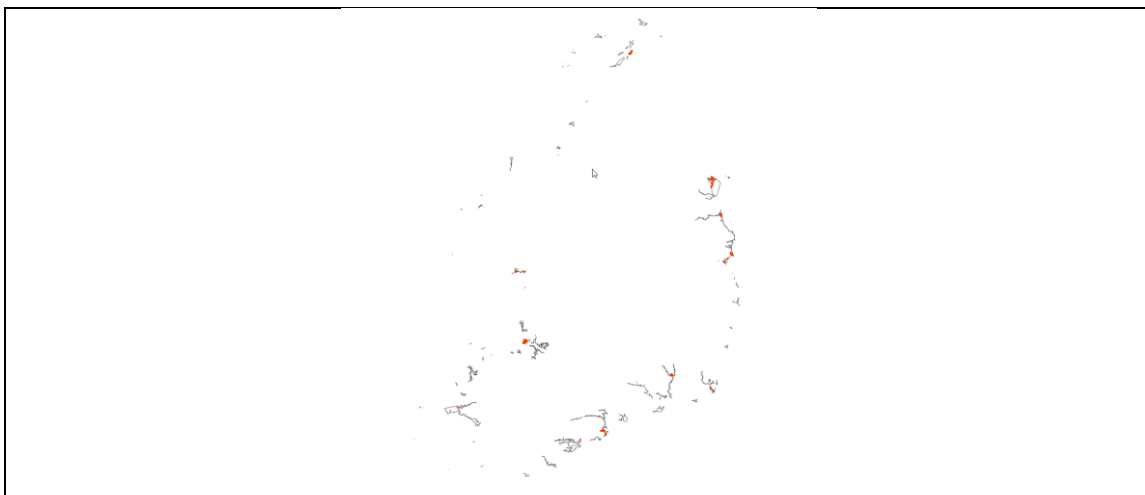


Table 8.2-33: Landscape character in Cork

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_CCC3_LAND_2014	CORK_ENV_CCC3_LAND_2014	URL	Shapefile	PiCork-Env
<p>DESCRIPTION:</p> <pre> geometry.coordinates: Array nx2 of floats geometry.type: string properties.ID: string properties.Sub_ID: string properties.TYPE: string type: string </pre>				
<p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSSG/0/29902 (TM65 / Irish Grid)</p> <pre> geometry.coordinates – format (((131096.83954366442, 118097.43850825573)..)) geometry.type – value “Polygon” properties.ID – alphanumeric; example CT14a properties.Sub_ID – alphanumeric; example 14a properties.TYPE – Descriptive string; example “Fissured Marginal and Forested Rolling Upland” type – value “Feature” </pre>				
<p>EXAMPLE:</p> <p>A map of Ireland showing the outlines of all counties, with a focus on the southern region where the data is located.</p>				

Table 8.2-34: Cork Car parking

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_CAR_PARKING	CORK_ENV_CAR_PARKING	URL	Shapefile	PiCork-Env
<p>DESCRIPTION:</p> <pre> geometry.coordinates: Array nx2 of floats geometry.type: string properties.Area: integer properties.Descriptio: string properties.ID:integer properties.Potential: integer type: string </pre>				
<p>METAINFORMATION:</p> <p>The data is retrieved from an URL. Format: ESRI Shapefile, representation type: Vector, Scale: 1/50000, Coordinate reference system: http://www.opengis.net/def/crs/EPSG/0/29902 (TM65 / Irish Grid)</p> <pre> geometry.coordinates – format (((580584.5403438584, 561945.8300102253)...)) geometry.type – value “Polygon” properties.Area – value 0 properties.Descriptio – No value properties.ID – integer (1,2 ...) properties.Potential_ – value 0 type – value “Feature” </pre>				
<p>EXAMPLE:</p> <p>The map shows 3 possible locations in the CrossHaven area.</p>				

Table 8.2-35: Weather historical information for Ireland

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_MET_W_DAILY	CORK_ENV_MET_W_DAILY	URL	CSV	PiCork-Env
<p>DESCRIPTION:</p> <pre> date: String rain: float temp: float wetb: float dewpt: float vappr: float rhum: integer msl: float wdsp: integer wddir: integer ind: float </pre>				
<p>METAINFORMATION:</p> <p>.CSV Each downloaded file contains dataset, as:</p> <pre> date: - Date and Time (utc) format DD-MMM-YYYY HH:mm rain: - Precipitation Amount (mm) temp: - Air Temperature (C) wetb: - Wet Bulb Temperature (C) dewpt: - Dew Point Temperature (C) vappr: - Vapour Pressure (hPa) </pre>				

<p>rhum: - Relative Humidity (%) msl: - Mean Sea Level Pressure (hPa) wdsp: - Mean Wind Speed (kt) wddir: - Predominant Wind Direction (deg) ind: - Indicator</p>
<p>EXAMPLE:</p> <pre>date,ind,rain,ind,temp,ind,wetb,dewpt,vappr,rhum,msl,ind,wdsp,ind,wddir 01-jan-1989 00:00,3,0.0,0,9.8,0,9.5,9.2,11.6,96,1036.5,2,12,2,170 01-jan-1989 01:00,3,0.0,0,9.8,0,9.6,9.4,11.8,97,1036.4,2,12,2,180</pre>

Table 8.2-36: Historic water levels data provided by the OPW for station 19069 Ringaskiddy NMCI (Cork)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ENV_OPW_WL_15MIN	CORK_ENV_OPW_WL_15MIN	URL	CSV	PiCork-Env
<p>DESCRIPTION:</p> <p>Date: String Value: float Quality: integer</p>				
<p>METAINFORMATION:</p> <p>.txt Each downloaded file contains dataset, as: Date: - Date and Time (utc) format YYYY/MM/DD HH:mm:ss Value: - Water Levels (metres) Quality: - Numeric value</p>				
<p>EXAMPLE:</p> <pre>Date Value Quality 2011/12/31 00:15:00 -0.535 *</pre>				


Table 8.2-37: Flood Maps in Antwerp

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ant_env_cityofant_maps	ant_env_cityofant_maps	URL	Shapefile	PiAntw-Env
<p>DESCRIPTION:</p> <pre>geometry.coordinates: Array nx2 of floats geometry.type: string properties.DEPTH2D: float type: string properties.X: float properties.Y: float</pre>				
<p>METAINFORMATION:</p> <p>Maps named floodMaps_2050_T05, floodMaps_2050_T20, floodMaps_2050_T100, floodMaps_2100_T20, floodMaps_2100_T05, floodMaps_2100_T100, floodMaps_current_T05, floodMaps_current_T20, floodMaps_current_T100 have the following information: geometry.coordinates – format (((149167.7607956122, 200004.05524744894)..))</p>				

geometry.type – value “Polygon”
 properties.DEPTH2D – (m) possible value 0.027
 type – value “Feature”

Map named floodMaps_current_T20, has also the following information
 properties.X – 154513.154786
 properties.Y – 206830.713052

EXAMPLE:



8.3 Economic data

This subsection presents new available economic data sources.

Table 8.3-16: Duden waterfall recreation area otopark (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_OTOPARK_MONTHLY	ANTA_ECO_CITYOFANTALYA_OTOPARK_MONTHLY	FILE	EXCEL	PiAnta-Eco
DESCRIPTION:				
No_field_name (Year):int No_field_name (Month):String Otopark Bileti: decimal Otopark Ücreti:decimal				
METAINFORMATION:				
No_field_name – Year with the format YYYY No_field_name – Month (name in Turkish Ocak, Subat, Mart...) Otopark Bileti – Number of Otopark Ticket Otopark Ücreti – Parking Fee revenue (Turkish Lira)				

EXAMPLE:

		Ziyaretçi Sayısı	Giriş Ücreti	Otopark Bileti	Otopark Ücreti	Dükkan Kirası	Elektrik Faturası	Pompa Elektrik Faturası	Çalışan Maaş
2018	Şubat	19,831	98,155	1,776	5,328		1,258.47 €	5,870.38 €	10,770.62 €
	Mart	30,134	150,870	2,865	8,595		1,212.22 €	2,599.41 €	10,770.62 €
	Nisan	49,427	247,135	5,875	17,625		1,696.83 €	6,029.75 €	11,570.71 €
	Mayıs	43,912	219,580	4,439	13,317		2,647.46 €		22,365.64 €
	Haziran	67,811	338,055	6,754	20,262		3,438.47 €	33,421.88 €	12,862.03 €
	Temmuz	59,551	297,755	14,141	42,423		5,591.53 €	36,049.41 €	11,957.94 €
	Ağustos	73,940	389,700	15,642	46,926		5,653.73 €	53,120.77 €	25,362.48 €
	Eylül	61,924	309,820	9,887	29,661		4,481.09 €	36,489.30 €	12,618.64 €
	Ekim	18,000	90,000	4,492	13,476		2,204.58 €	41,054.10 €	23,185.31 €
	Kasım	11,887	59,435	2,324	6,972		1,908.06 €	32,815.60 €	11,778.50 €
	Aralık	8,447	42,235	1,697	5,091		2,291.94 €	25,415.03 €	12,303.59 €

Table 8.3-17: Duden waterfall recreation area visitors (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_VISITORTICKET_MONTHLY	ANTA_ECO_CITYOFANTALYA_VISITORTICKET_MONTHLY	FILE	EXCEL	PiAnta-Eco

DESCRIPTION:

No_field_name (Year):int
No_field_name (Month):String
Ziyaretçi Sayısı: decimal
Giriş Ücreti:decimal

METAINFORMATION:

No_field_name – Year with the format YYYY
No_field_name – Month (name in Turkish Ocak, Subat, Mart...)
Ziyaretçi Sayısı – Number of Visitors Tickets
Giriş Ücreti – Entrance Fee revenue (Turkish Lira)

EXAMPLE:

		Ziyaretçi Sayısı	Giriş Ücreti	Otopark Bileti	Otopark Ücreti	Dükkan Kirası	Elektrik Faturası	Pompa Elektrik Faturası	Çalışan Maaş
2018	Şubat	19,831	98,155	1,776	5,328		1,258.47 €	5,870.38 €	10,770.62 €
	Mart	30,134	150,870	2,865	8,595		1,212.22 €	2,599.41 €	10,770.62 €
	Nisan	49,427	247,135	5,875	17,625		1,696.83 €	6,029.75 €	11,570.71 €
	Mayıs	43,912	219,580	4,439	13,317		2,647.46 €		22,365.64 €
	Haziran	67,811	338,055	6,754	20,262		3,438.47 €	33,421.88 €	12,862.03 €
	Temmuz	59,551	297,755	14,141	42,423		5,591.53 €	36,049.41 €	11,957.94 €
	Ağustos	73,940	389,700	15,642	46,926		5,653.73 €	53,120.77 €	25,362.48 €
	Eylül	61,924	309,820	9,887	29,661		4,481.09 €	36,489.30 €	12,618.64 €
	Ekim	18,000	90,000	4,492	13,476		2,204.58 €	41,054.10 €	23,185.31 €
	Kasım	11,887	59,435	2,324	6,972		1,908.06 €	32,815.60 €	11,778.50 €
	Aralık	8,447	42,235	1,697	5,091		2,291.94 €	25,415.03 €	12,303.59 €

Table 8.3-18: Duden waterfall recreation area operation employee salary (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_OPERATIONEMPLOYEESALARY_MONTHLY	ANTA_ECO_CITYOFANTALYA_OPERATIONEMPLOYEESALARY_MONTHLY	FILE	EXCEL	PiAnta-Eco

DESCRIPTION:

No_field_name (Year):int
No_field_name (Month):String
Çalışan Maaş:decimal

METAINFORMATION:

No_field_name – Year with the format YYYY
No_field_name – Month (name in Turkish Ocak, Subat, Mart...)
Çalışan Maaş – operation employee salary (Turkish Lira)

EXAMPLE:

		Ziyaretçi Sayısı	Giriş Ücreti	Otopark Bileti	Otopark Ücreti	Dükkan Kirası	Elektrik Faturası	Pompa Elektrik Faturası	Çalışan Maaş
2018	Şubat	19,831	98,155	1,776	5,328		1,258.47 €	5,870.38 €	10,770.62 €
	Mart	30,134	150,870	2,865	8,595		1,212.22 €	2,599.41 €	10,770.62 €
	Nisan	49,427	247,135	5,875	17,625		1,696.83 €	6,029.75 €	11,570.71 €
	Mayıs	43,912	219,560	4,439	13,317		2,647.46 €		22,365.64 €
	Haziran	67,611	338,055	6,754	20,262		3,438.47 €	33,421.88 €	12,862.03 €
	Temmuz	59,551	297,755	14,141	42,423		5,591.53 €	36,049.41 €	11,957.94 €
	Ağustos	73,940	369,700	15,642	46,926		5,653.73 €	53,120.77 €	25,362.48 €
	Eylül	61,924	309,620	9,887	29,661		4,481.09 €	36,489.30 €	12,618.64 €
	Ekim	18,000	90,000	4,492	13,476		2,204.58 €	41,054.10 €	23,185.31 €
	Kasım	11,887	59,435	2,324	6,972		1,908.06 €	32,815.60 €	11,778.50 €
	Aralık	8,447	42,235	1,697	5,091		2,291.94 €	25,415.03 €	12,303.59 €

Table 8.3-19: Duden waterfall general electric bill (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_GENERALELECTIRICBILL_MONTHLY	ANTA_ECO_CITYOFANTALYA_GENERALELECTIRICBILL_MONTHLY	FILE	EXCEL	PiAnta-Eco

DESCRIPTION:

No_field_name (Year):int
No_field_name (Month):String
Elektrik Faturası: decimal

METAINFORMATION:

No_field_name – Year with the format YYYY
No_field_name – Month (name in Turkish Ocak, Subat, Mart...)
Elektrik Faturası – General Electric bill (Turkish Lira)

EXAMPLE:

		Ziyaretçi Sayısı	Giriş Ücreti	Otopark Bileti	Otopark Ücreti	Dükkan Kirası	Elektrik Faturası	Pompa Elektrik Faturası	Çalışan Maaş
2018	Şubat	19,831	98,155	1,776	5,328		1,258.47 €	5,870.38 €	10,770.62 €
	Mart	30,134	150,870	2,865	8,595		1,212.22 €	2,599.41 €	10,770.62 €
	Nisan	49,427	247,135	5,875	17,625		1,696.83 €	6,029.75 €	11,570.71 €
	Mayıs	43,912	219,560	4,439	13,317		2,647.46 €		22,365.64 €
	Haziran	67,611	338,055	6,754	20,262		3,438.47 €	33,421.88 €	12,862.03 €
	Temmuz	59,551	297,755	14,141	42,423		5,591.53 €	36,049.41 €	11,957.94 €
	Ağustos	73,940	369,700	15,642	46,926		5,653.73 €	53,120.77 €	25,362.48 €
	Eylül	61,924	309,620	9,887	29,661		4,481.09 €	36,489.30 €	12,618.64 €
	Ekim	18,000	90,000	4,492	13,476		2,204.58 €	41,054.10 €	23,185.31 €
	Kasım	11,887	59,435	2,324	6,972		1,908.06 €	32,815.60 €	11,778.50 €
	Aralık	8,447	42,235	1,697	5,091		2,291.94 €	25,415.03 €	12,303.59 €

Table 8.3-20: Duden waterfall pumps electric bill (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_WATERPOMPS_MONTHLY	ANTA_ECO_CITYOFANTALYA_WATERPOMPS_MONTHLY	FILE	EXCEL	PiAnta-Eco

DESCRIPTION:

No_field_name (Year):int
No_field_name (Month):String
Pompa Elektrik Faturası: decimal

METAINFORMATION:

No_field_name – Year with the format YYYY
No_field_name – Month (name in Turkish Ocak, Subat, Mart...)
Pompa Elektrik Faturası – Water pump electric bill (Turkish Lira)

EXAMPLE:

		Ziyaretçi Sayısı	Giriş Ücreti	Otopark Bileti	Otopark Ücreti	Dükkan Kirası	Elektrik Faturası	Pompa Elektrik Faturası	Çalışan Maaş
2018	Şubat	19,831	98,155	1,776	5,328		1,258.47 ₺	5,870.38 ₺	10,770.62 ₺
	Mart	30,134	150,870	2,865	8,595		1,212.22 ₺	2,539.41 ₺	10,770.62 ₺
	Nisan	49,427	247,135	5,875	17,625		1,696.83 ₺	6,029.75 ₺	11,570.71 ₺
	Mayıs	43,912	219,580	4,439	13,317		2,647.46 ₺		22,365.64 ₺
	Haziran	67,811	338,055	6,754	20,262		3,438.47 ₺	33,421.88 ₺	12,862.03 ₺
	Temmuz	59,551	297,755	14,141	42,423		5,591.53 ₺	36,049.41 ₺	11,957.94 ₺
	Ağustos	73,940	389,700	15,642	46,926		5,653.73 ₺	53,120.77 ₺	25,362.48 ₺
	Eylül	61,924	309,820	9,887	29,661		4,481.09 ₺	36,489.30 ₺	12,618.64 ₺
	Ekim	18,000	90,000	4,492	13,476		2,204.58 ₺	41,054.10 ₺	23,185.31 ₺
	Kasım	11,887	59,435	2,324	6,972		1,908.06 ₺	32,815.60 ₺	11,778.50 ₺
	Aralık	8,447	42,235	1,697	5,091		2,291.94 ₺	25,415.03 ₺	12,303.59 ₺

Table 8.3-21: Duden waterfall public transportation passenger number (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE								
ANTA_ECO_CITYOFANTALYA_CITYZONEPULICTRANSPORTATIONPASNGERNUMBER_MONTHLY	ANTA_ECO_CITYOFANTALYA_CITYZONEPULICTRANSPORTATIONPASNGERNUMBER_MONTHLY	FILE	EXCEL	PiAnta-Eco								
DESCRIPTION:												
DATE:date ZONE NAME:String NUMBER OF TOUR ,FREE1, FREE2, FREE3, TICKET, STUDENT, PERSON, KREDI KART PERSON, TEACHER, RETRIED, S.KART INDIRIMLI, AIRPORT EMPLOYER, TAX AUDIT CARD, TOTAL SUM: decimal BUS NUMBER CODE: String												
METAINFORMATION:												
Information in several sheets of an .excel file Date – Date with the format mm/dd/ YYYY ZONE NAME – empty NUMBER OF TOUR ,FREE1, FREE2, FREE3, TICKET, STUDENT, PERSON, KREDI KART PERSON, TEACHER, RETRIED, S.KART INDIRIMLI, AIRPORT EMPLOYER, TAX AUDIT CARD, TOTAL SUM – value for number of different type of users CODE– String, name of each sheet												
EXAMPLE:												
Yolcu Sayısı												
ZONE NAME	NUMBER OF TOUR	FREE1	FREE2	FREE3	TICKET	STUDENT	PERSON	EDI KART	PERSO	TEACHER	RETRIED	S.KART INDIR
	33	163	0	0	6	206	658		4	19	133	
	48	396	0	0	13	628	1,036		16	44	253	
	47	262	0	0	10	631	988		7	37	214	
	47	297	0	0	19	583	903		8	39	224	
	47	295	0	0	19	638	1,015		9	54	243	
	48	294	0	0	19	490	947		11	31	251	
	32	185	0	0	24	297	628		3	19	129	
	47	275	0	0	17	558	1,027		6	56	232	
	47	349	0	0	19	619	1,029		7	48	292	
	47	289	0	0	18	619	946		9	50	184	
	45	138	0	0	9	522	834		6	52	166	
	45	337	0	0	20	638	1,112		14	71	246	

Table 8.3-22: Duden waterfall shops rent earn (Antalya)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
ANTA_ECO_CITYOFANTALYA_SHOPSRENTEARN_YEAR	ANTA_ECO_CITYOFANTALYA_SHOPSRENTEARN_YEAR	FILE	EXCEL	PiAnta-Eco
DESCRIPTION:				
No field name (Year):int				

TL: decimal	
METAINFORMATION:	
No_field_name – Year with the format YYYY	
TL – Shops rent (Turkish Lira)	
EXAMPLE:	
	TL
2015	137000
2016	164000
2017	176000
2018	196000
2019	

Table 8.3-23: Number of visitors (Cork)

DS_ID	D_ID	ORIGIN	FORMAT	CODE
CORK_ECO_VISITORS_DAILY	CORK_ECO_VISITORS_DAILY	FILE	EXCEL	PiCork-Eco
DESCRIPTION:				
Date :date				
Number of visitors: int				
Number of visitors (pay): int				
METAINFORMATION:				
Date – Date with the format d-mmm-YYYY, month in string format				
Number of visitors – Total number of visitors				
Number of visitors (pay) – Number of visitors that actually pay entrance fee				
EXAMPLE:				
Date	Number of visitors	Number of visitors (pay)		
1-Jan-18	100	50		
2-Jan-18	110	55		
3-Jan-18	23	13		
4-Jan-18	45	22		
5-Jan-18	67	33		
6-Jan-18	23	11		
7-Jan-18	12	6		
8-Jan-18	2	1		
9-Jan-18	16	8		
10-Jan-18	30	15		
11-Jan-18	44	22		
12-Jan-18	59	29		

8.4 Social data

No new social data were included in this version. Recently, Cork and Antalya prepared questionnaires addressed to visitors of the pilot sites and launched surveys to collect answers. The structure of the collected data will be presented in D3.5.

9. Appendix 2 - Assessment of confirmed data sources by DS_ID

The data attributes corresponding to the index numbers mentioned in the Tables below can be found in https://mklab.iti.gr/cutler/doku.php?id=ds_attributes. This assessment is presented only for confirmed data sources. Please refer to D1.1 and D1.2 for further details on legal taxonomy and legal requirements.

9.1 Data sources for Antalya

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
1,2,7,9,16,17	ANTALYA_ENV_CITYOFANTALYA_PERMINUTE (D3.2 8.2-20)			X				X	X	
4,5,6,10,11,15,425,426,427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,443,444	ANTA_ENV_CITYOFANTALYA2_MONTHLY (D3.2 8.2-22)			X		X		X	X	X
19	PIALL-SOC_NEWS (D3.2 8.4-4)	X			X	X				
20	PIALL-SOC_TWITTER (D3.2 8.4-3)	X			X	X				X
32,34,36,38,39,47	PIANTAL_ECO_OECD_REGION_ECO (D3.2 8.3-9, 8.3-10,8.3-6,			X	X	X				

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
	8.3-7,8.3-8)									
44,45,46,345,346,52,53,349,350,347,348,351	PIANTAL_ECO_OECD_REGION LABOUR (D3.2 8.3-3,8.3-4,8.3-5)			X	X	X				
445	ANTA_ECO_CITYOFANTALYA_OTOPARK_MONTHLY (D3.2 8.3-16)			X		X				
446	ANTA_ECO_CITYOFANTALYA_VISITORTICKET_MONTHLY (8.3-17)			X		X				
447	ANTA_ECO_CITYOFANTALYA_SHOPSRENTYEAR_YEAR (8.3-22)	X				X				
448	ANTA_ECO_CITYOFANTALYA_OPERATIONEMPLOYEESALARY_MONTHLY (8.3-18)	X				X				
449	ANTA_ECO_CITYOFANTALYA_GENERALELECTIRICBILL_MONTHLY (8.3-19)			X		X			X	
450	ANTA_ECO_CITYOFANTALYA_WATERPOMPS_MONTHLY (8.3-20)			X		X			X	
452	ANTA_ECO_CITYOFANTALYA			X		X				

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
	_CITYZONEPUBLICTRANSPORTATIONPASSENGERNUMBER_MONTHLY (8.3-21)									

9.2 Data sources for Antwerp

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
57	ANT_ENV_IMEC_PREC2018 (8.2-17)			X		X			X	X
59	ANT_ENV_IMEC_OPENWATER (8.2-18)			X		X			X	X
64	PIALL-SOC_NEWS (D3.2 8.4-4)	X			X	X				
67	PIALL-SOC_TWITTER (D3.2 8.4-3)	X			X					X
72	ANT_ENV_CITYOFANT_HISTPREC (D3.2 8.2-15)			X				X	X	X

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
75	ANT_ENV_CITYOFANT_GWL (D3.2 8.2-16)			X				X	X	X
76,77, 108,109, 110,81, 82,93,84, 91,116, 117	PIANTW_ECO_OECD_REGIONL ABOUT (D3.2 8.3-3,8.3-5,8.3-4)			X	X	X				
96,98, 100,102, 103,111	PIANTW_ECO_OECD_REGION ECO (D3.2 8.3-9,8.3-10,8.3-6,8.3-7,8.3-8)			X	X	X				
343	ant_env_imec_sewer2018 (8.2-19)			X		X			X	X
424	ant_env_cityofant_maps (8.2-37)			X		X		X	X	X

9.3 Data sources for Cork

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
120	PIALL-SOC_NEWS (D3.2 8.4-4)	X			X	X				

123	PIALL-SOC_TWITTER (D3.2 8.4-3)	X			X	X				X
199,200, 231,232, 233,204, 205,216, 207,214, 239,240	PICORK_ECO_OECD_REGIONL ABOUR (D3.2 8.3-3,8.3-5,8.3-4)			X	X	X				
219,221, 223,225, 226,234	PICORK_ECO_OECD_REGIONE CO (D3.2 8.3-9,8.3-10,8.3-6,8.3- 7,8.3-8)			X	X	X				
405	CORK_ENV_EPA_SAC_2015 (8.2-24)			X		X		X	X	X
406	CORK_ENV_EPA_NHA_2012 (8.2-25)			X		X		X	X	X
407	CORK_ENV_EPA_SPA_2015 (8.2-26)			X		X		X	X	X
408	CORK_ENV_EPA_GWWFD_201 02015 (8.2-27)			X		X		X	X	X
409	CORK_ENV_EPA_LWFD_20102 015 (8.2-28)			X		X		X	X	X
410	CORK_ENV_EPA_RWFD_20102 015 (8.2-29)			X		X		X	X	X
411	CORK_ENV_EPA_CWFD_20102 015 (8.2-30)			X		X		X	X	X
412	CORK_ENV_EPA_TWFD_20102 015 (8.2-31)			X		X		X	X	X
413	CORK_ENV_OPW_FLOODS_20 16 (8.2-32)			X		X		X	X	X

414	CORK_ENV_CCC3_LAND_2014 (8.2-33)			X		X		X	X	
415	CORK_ENV_OPW_WL_15MIN (8.2-36)			X		X		X	X	
416,417, 418,419, 420,421, 422,423	CORK_ENV_MET_W_DAILY (8.2-35)			X		X		X	X	

9.4 Data sources for Thessaloniki

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
265	PIALL-SOC_TWITTER (D3.2 8.4-3)	X			X	X				X
269	PIALL-SOC_NEWS (D3.2 8.4-4)	X			X	X				
271	THESS_SOC_IMC_MONTHLY (D3.2 8.4-1)	X			X	X				X
273,274, 305,306, 307,278, 279,290, 281,288, 313,314	PITHESS_ECO_OECD_REGIO NLABOUR (D3.2 8.3-3,8.3- 5,8.3-4)			X	X	X				
293,295, 297,299, 300,308	PITHESS_ECO_OECD_REGIO NECO (D3.2 8.3-9,8.3-10,8.3-6,8.3-			X	X	X				

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR Protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
	7,8.3-8)									
321	THESS_ECO_THESSALONIKI_MUNICIPALITY_BUDGET (D3.2 8.3-1)			X				X		
322,342	THESS_ECO_THESSALONIKI_PARKING_SCANS (D3.2 8.3-11)	X		X		X				X
329,330, 331,332, 333,334, 335	THESS_ENV_CITYOFTHESS_DAILYEARLY (D3.2 8.2-14)			X				X	X	X
389	THESS_ENV_IMET_SPEED_15 MIN (8.2-21)			X		X		X		

9.5 Data sources for all city pilots

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
352,353, 354,355,	EUROSTAT_REG (D3.2 8.3-13)			X		X				

Index number	DS_ID	Personal data	Electronic communications data	Non-personal data	CR protected data	Data protected by the sui generis right on databases	Trade secret	Publicly held documents	Environmental data	Spatial data
356,357, 358,359, 360,361, 362,363, 364,365, 366,367, 368,369, 370,371					X					
372,373, 374,375	EUROSTAT_REG_TYPE (D3.2 8.3-14)			X	X	X				
376,377, 378,379, 380,381, 382,383, 384,385, 386,387, 388	EUROSTAT_URB (D3.2 8.3-15)			X	X	X				

10. References

- [1] Gilman et al. CUTLER deliverable D3.2 “First version of the framework for the collection, cleaning, integration & anonymization of big data”, 2018
- [2] Byrne A. CUTLER deliverable D2.3 “First version architecture for scalable hybrid cloud infrastructure”
- [3] Tsalakanidou, F. et al. CUTLER deliverable D3.1 “Requirements for data crawling, integration and anonymization”, 2018.
- [4] Dewitte, P., Kuczerawy, A., Valcke, P. CUTLER D1.1 “Legal Taxonomy of Datasets”, 2018.
- [5] Dewitte, P., Chelioudakis, E., Kuczerawy, A. CUTLER D1.2 “Legal requirements”, 2018.
- [6] Özkan H. İ. et al. CUTLER deliverable D9.1 “Report on pilot preparations and pilot execution plan”, 2018.
- [7] CUTLER GitHub repository, <https://github.com/CUTLER-H2020>
- [8] CUTLER, data for the second version of the framework, http://mklab.iti.gr/cutler/doku.php?id=framework_v2_data_catalogue
- [9] CUTLER data catalogue, http://mklab.iti.gr/cutler/doku.php?id=data_catalogue
- [10] Yümlü, M.S. et al., CUTLER deliverable D8.1 “Integration Protocol and Technical Verification”, 2018
- [11] Barnett, A. and Byrne A., CUTLER deliverable D2.2 “Threat analysis for policy-supporting hybrid cloud infrastructure”, 2018
- [12] Apache Hadoop <https://hadoop.apache.org/>
- [13] Elasticsearch for Apache Hadoop [7.0], <https://www.elastic.co/guide/en/elasticsearch/hadoop/current/reference.html>
- [14] Apache Flume, User Guide <https://flume.apache.org/releases/content/1.9.0/FlumeUserGuide.html>
- [15] Abedjan, Z., et al. “Detecting data errors: where are we and what needs to be done?” Proc. VLDB Endow. 9, 12 (August 2016), pp. 993-1004.
- [16] Apache Atlas, <https://atlas.apache.org/>
- [17] Apache Ranger, <https://ranger.apache.org/>
- [18] RSA Archer Data Governance, <https://www.rsa.com/content/dam/en/data-sheet/rsa-archer-data-governance.pdf>
- [19] Apache Flume, <https://flume.apache.org/>