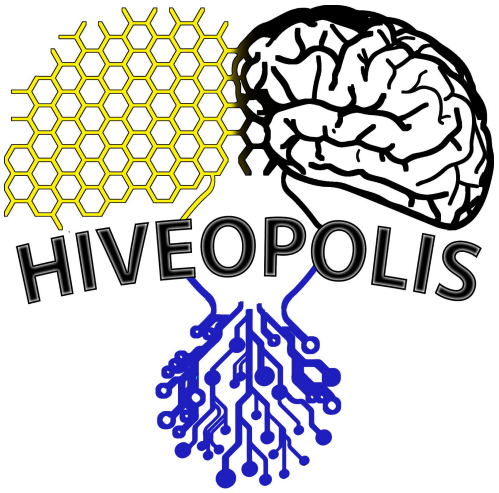

FUTURISTIC BEEHIVES FOR A SMART METROPOLIS

Deliverable D7.1

Data Management Plan (DMP)

Lead Beneficiary	UBER
Delivery date	30.09.2019
Dissemination Level	PU
Version	1.0
Project website	www.hiveopolis.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824069

DELIVERABLE SUMMARY SHEET

Project number	824069
Project Acronym	HIVEOPOLIS
Title	FUTURISTIC BEEHIVES FOR A SMART METROPOLIS
Deliverable No	D7.1
Due Date	Project month M6
Delivery Date	30.09.19
Name	Data Management Plan (DMP)
Description	As part of the Open Research Data Pilot we specify in this deliverable what data the project will generate, whether and how it will be exploited or made accessible for verification and reuse and how it will be curated and preserved.
Lead Beneficiary	UBER
Partners contributed	UNIGRAZ, EPFL, FUB, ULB, BST, LLU, UBER
Dissemination Level	Public

Introduction

Purpose and scope of the document

As part of the Open Research Data Pilot we specify what data the project will generate, whether and how it will be exploited or made accessible for verification and reuse, and how it will be curated and preserved. As general approach we will work towards making as much data open as possible and to allow for third party use as soon as possible. In this document, we will introduce the first version of the Data Management Plan (DMP) elaborated for the HIVEOPOLIS project. This DMP will be updated during the project lifetime.

Overview of the document

The document is structured into six chapters with accordance with the official Horizon 2020 DMP guidelines[1]:

Chapter 1: Data Summary
Chapter 2: FAIR Data
Chapter 3: Allocation of Resources
Chapter 4: Data Security
Chapter 5: Ethical Aspects
Chapter 6: Others

For easier readability of the document, the main points to be addressed according to the template were added at the beginning of the sections as cursive bullet points.

[1] https://ec.europa.eu/research/participants/docs/h2020-funding-guide/cross-cutting-issues/open-access-data-management/data-management_en.htm

Chapter 1: Data Summary

- *State the purpose of the data collection/generation*
- *Explain the relation to the objectives of the project*
- *Specify the types and formats of data generated/collected*
- *Specify if existing data is being re-used (if any)*
- *Specify the origin of the data*
- *State the expected size of the data (if known)*
- *Outline the data utility: to whom will it be useful*

HIVEOPOLIS technology will be integrated in a way that it provides a synergistic added value to the colony, to its owner and to society in general. It will be sustainable from the ecological point of view and also from the intellectual point of view (open software, open hardware, open data, citizen scientists), so that it will be accessible from the beginning.

For a better overview the data produced within the HIVEOPOLIS project is assigned to the following categories which can partially overlap:

1. Design files: This category covers documents describing the hardware & software design and the design of the experimental set-up.

2. Source code: Source code of programs and scripts developed within the project.

3. Collected data: This category covers data on bees, hives and environment collected with different methods (directly with sensors, crowdsourcing, publicly available data).

4. Created data: This category covers data generated by the computational models and robotic systems.

5. Results: This category covers the analysis and evaluation of the data.

6. Media files: This category contains pictures, videos, and audio recordings that document the project for presentations.

7. User data: This category contains all data collected from the users and the community.

The most relevant categories in the scope of this document are collected and created data, source code and design files.

1.1 File formats and software

This table shows the main data types that will be used in HIVEOPOLIS, their purpose, and the software to access them.

File format	Data type	Software	Purpose	Comments
JPEG, PNG	Bitmap images	Image viewer, e.g., Gimp	Any images taken for documentation purposes; Images of honeycomb for detection of bees, broodnest, etc.	Compressed images
MPEG, MKV, AVI	Videos	Video viewer, e.g., VLC	Observation Videos of beehives	Compressed video
WAV	Audio	Audio Player, e.g., VLC	Sound observations of beehives	Uncompressed audio
FLAC, MP3, OGG	Audio	Audio Player	Sound observations of beehives	Compressed audio
HDF5, netCDF	Structured binary scientific data	MATLAB, Python, Scilab, Julia, R	store and organize large amounts of scientific data	Hierarchical Data Format, designed for large amount of data
JSON	Structured scientific data	MATLAB, Python, Scilab, Julia, R	Annotation of data, interoperable way of storing models and data	Human readable, stored in text files
YAML	Structured, human-readable data	Javascript, Python, R	Configuration files, small data files	Human readable, stored in text files
CSV	Value tables	Excel, openoffice, MATLAB, R	Readable data storage	Human readable, very easy format

TEX	LaTeX - text with formatting markup	Text Editor	Writing documents such as journal articles	
TEXT/UTF8	text	Text editor	Unstructured text	Plain text
PDF	Documents	PDF viewer	Articles, reports, and other documents	
MD	Markdown	Text editor, Markdown viewer	Structured notes, tutorials, technical documentation to be presented online, e.g., on GitHub	Raw input is human-readable plain-text, but can be rendered into html, pdf etc.
RST	Restructured text	Text editor	Structured notes, technical documentation.	Raw input is human-readable plain-text, but can be rendered into html, pdf etc.
Source code, including .c, .cpp, .kt, .py, .R, .js, .ino, .h, .m	Programs and scripts	Text editor	Coding for system software and firmware, data analysis & visualisation	Stored in text files
SVG	Vector images	Image viewer, e.g. Inkscape	Vector graphics, like diagrams, schematics	Open source format for vector graphics
OBJ, STL, STEP, IGES, 3DS, MODEL	3D models	3D model viewer, e.g. Blender, Meshlab	Design of physical objects (3D models)	Would be nice to have cross-software format.
SLDDRW, SLDPRT, SLDASM, 3DM, DWG, DXF, CATDRAWING, CATPART, CATPRODUCT, SESSION, SCAD	Structured binary data	E.g. SOLIDWORKS, 3d Builder, Rhino3D, Creo, CATIA, OpenSCAD	CAD for mechanical designs	

STL, GCODE, DXF, DWG	Structured text	software specific for each equipment	Rapid prototyping (3d printing, CNC milling, laser cutting, etc.)	
Sch, brd, lib, kicad_pcb, pro, schdoc, pcbdoc, prjpcb, pcblib, schlib	Structured binary data	E.g. KiCad (Open Source), EAGLE, Altium designer, PADS, Orcad	Electrical schematic capture and PCB layout	
Text file with many extensions (e.g. .GTO, .GBO, GBR, etc.)	Structured text data	E.g. Text editors or CAM/Gerber viewers	PCB fabrication files (CAM and Gerber files)	Raw input is human-readable plain-text

1.2 Main storage locations

The project follows a distributed storage strategy, as data volumes can become very large and therefore centralized storage during the project is not feasible. Each partner is hence responsible to provide adequate institutional storage and ensure backups for the local data generated by that partner during the project. In the following we outline the projected data to be generated by each project partner and their local storage infrastructure.

University of Graz (UNIGRAZ)

- Testhive Full HD Videos: ~ 30TB (compressed; per experimentation season; single camera)
- Testhive .jpg-pictures: ~ 40TB (uncompressed; per experimentation season; single camera)
- .wav sound data: ~ 400GB (per experimentation season; one single comb with 4-channel audio)
- .csv data log: < 1GB (per experimentation season)
- Source code: <1GB

These approximations apply for a single observation hive and could multiply drastically within the course of the project.

Local Storage	
Storage	NAS system (scalable to a capacity of 384TB)
Backup	NAS system (identical to the above) with daily incremental backups.

École Polytechnique Fédérale de Lausanne (EPFL)

- Video data for validation purposes < 300GB
- multichannel audio (raw), for prototype validation ~120GB/prototype
- Hive sensing, in operational-mode sensing ~ 2GB/season/hive
- Source code for firmware, software, simulation and modelling <1GB
- Simulation model output <10GB
- Design files (electronics and mechanical designs) <2GB
- Total anticipated <1TB

Local Storage	
Storage	Google apps for education (unlimited quota)
Backup	local backups on physical storage

Université Libre de Bruxelles (ULB)

- Source code for firmware, software, simulation and modelling <1GB
- Simulation model output <10GB

Local Storage	
Storage	Internal servers
Backup	local backups on physical storage

Freie Universität Berlin (FUB)

- High-res video data of full colony: ~ 50 TB per year
- Low-res video snippets of bee dances: 100 GB per year
- Marker detection data and trajectories: 200 GB per year
- Video snippets of detected bees at the hive entrance and feeders: ~ 100 GB per year
- Photos of markers used for marking age cohorts (~ 100 images per year)
- Design files and related data (CAD, circuit boards, BOMs): 2 GB per year

Local Storage	
Storage	storage space at HLRN (supercomputing facility at Zuse Institute Berlin) and CURTA cluster (internal compute and storage cluster)
Backup	tape storage at HLRN

Pollenity (BST)

- High resolution hive thermal images: <1GB
- Beehive prototype 3d designs and technical drawings: ~30GB
- Sensor data records from gate module: ~25GB
- Source code for developed systems: <1GB
- Schematics and board designs: ~500MB

Local Storage	
Storage	self hosted cloud storage
Backup	local backups on physical storage

Latvia University of Life Sciences & Technologies (LLU)

- Source code for modeling and simulations: < 200MB, locally and public cloud based repositories
- Source code of the developed systems, solutions: < 500MB, locally and public cloud based repositories
- Sample data for modeling and simulation with different formats: <10GB, locally and university storage
- Data of the individual HIVEOPOLIS units: < 1TB,

Local Storage	
Storage	local storage, managed by the faculty IT administrator; cloud storage on Google Drive
Backup	local backups on physical storage

Humboldt-Universität zu Berlin (UBER)

- Implementation for predictive models and simulations / source code / < 200MB
- Evaluation of the performance of the models / analysis results / < 1GB
- Visualization of results / media files / < 30GB
- Data generated by models and simulations / collected & created data / < 1TB

Local Storage	
Storage	https://www.cms.hu-berlin.de/de/dl/speicherdienste/HU-Box
Backup	The service is provided by the University and includes backup by default.

1.3 Reused Data

This project will make use of previously generated data from project “BeesBook” at FU Berlin (Wario et al. 2015). High resolution image data of barcode-tagged bees was recorded over the course of several weeks. Datasets from 2014, 2015, 2016, 2018 and 2019 are available. In detail, we will use:

- HR: High resolution image data, 3 - 6 Hz frame rate, 15 - 30 pixels / mm spatial resolution, 6 - 10 weeks continuous recording duration
- LR: Low resolution image data, 60 - 125 Hz frame rate, 1 pixel / mm spatial resolution, video snippets 50 x 50 px for up to 10 sec when detected vibrational activity, 6-10 w recording duration
- Trajectories of individually marked bees, result of processing HR
- Focal behaviors: location, identity and time interval of waggle dances, dance following and food exchange behaviors, result of processing HR and LR datasets

Chapter 2: FAIR Data

2.1 Making data findable, including provisions for metadata

- *Outline the discoverability of data (metadata provision)*
- *Outline the identifiability of data and refer to standard identification mechanism. Do you make use of persistent and unique identifiers such as Digital Object Identifiers?*
- *Outline naming conventions used*
- *Outline the approach towards search keyword*
- *Outline the approach for clear versioning*
- *Specify standards for metadata creation (if any). If there are no standards in your discipline describe what metadata will be created and how*

For the discoverability of representative core data of HIVEOPOLIS that is intended for the validation of results in publications and for releases of source code the Zenodo repository (https://zenodo.org) will be used. The zenodo service provides adequate capacity (50GB per dataset at the time of writing) and a long-term retention period of more than 20 years. In addition, the publication of HIVEOPOLIS research data at Zenodo ensures the FAIRness of the data by providing DOIs, a landing page with metadata including keywords and clear licensing for re-use. At the end of the project relevant selected project data - including public data sets and media files - that have not yet been published will be archived at Zenodo. A description, presentation and link of the produced data will be presented on the HIVEOPOLIS website as well.

Files (excluding software) will be structured in hierarchical folders that state the subproject and data category:
`[subproject]_[data category]`

File names are created according to the following convention that allows for versioning by date:
`HIVEOPOLIS_[data type]_[date].[file format]`

The source code will be made available through industry-standard repositories such as GitHub. The repository will be documented with clear descriptions and keywords which makes the code easily discoverable. Documentation on how to install and use the software will be part of the repository. Standard versioning practices and tools like Git, will be used to keep trace of the stable versions of the code.

Keywords and Metadata will be created within the Zenodo Repository that relies on inter-disciplinary standards from DataCite and DublinCore. This is necessary, because no disciplinary standards are applicable. The respective project members will publish their data during the project phase for verification of results. Three months before project end the project partners upload selected research data that might be of broader interest to the public and that has not yet been published.

2.2 Making data openly accessible

- *Specify which data will be made openly available? If some data is kept closed provide rationale for doing so*
- *Specify how the data will be made available*
- *Specify what methods or software tools are needed to access the data? Is documentation about the software needed to access the data included? Is it possible to include the relevant software (e.g. in open source code)?*
- *Specify where the data and associated metadata, documentation and code are deposited*
- *Specify how access will be provided in case there are any restrictions*

Collected and created data

The amount of data created and collected within the project HIVEOPOLIS is expected to be too large to be made publicly accessible as a whole. We will select and curate data sets of special broader interest collected in our experiments. The datasets will be archived in an open-access repository with guaranteed long term accessibility, like https://zenodo.org, and linked in an “open data” section of the project website. We also plan to publish some data sets of special interest as “data papers”, e.g. in the journals “Int. J of Robotics Research”, “Ecology”, “Data in Brief” or “Scientific data”. Sample analysis and some basic analysis/filtering tools using an open source standard tool, like R or a python notebook will be provided with each dataset, demonstrating a basic workflow with the data as to lower the threshold for starting the work for interested researchers. Other large data sets, which were not published, will be archived in the respective partner institution and made available upon request.

Source code

Our efforts towards making the source code findable and reusable will result in the source code being openly accessible. Further details are provided in sec 2.1 and sec 2.4.

2.3 Making data interoperable

- *Assess the interoperability of your data. Specify what data and metadata vocabularies, standards or methodologies you will follow to facilitate interoperability.*
- *Specify whether you will be using standard vocabulary for all data types present in your data set, to allow inter-disciplinary interoperability? If not, will you provide mapping to more commonly used ontologies?*

Collected and created data

Widely used file formats will be used (see 1.1). If data will be produced in proprietary formats, it will be exported to a more open standard, if possible, stored and made available in addition to the original file. Data will be documented and described with discipline-specific metadata schemata like DarwinCore (http://rs.tdwg.org/dwc/index.htm), ABCD (https://abcd.tdwg.org/) or AgMES (http://aims.fao.org/standards/agmes) for biological and agricultural data. Corresponding thesauri and ontologies are used like e.g. AGROVOC (http://aims.fao.org/vest-registry/vocabularies/agrovoc). If the data is classified as interdisciplinary, a mapping will be provided or free keywords and generic controlled vocabulary like DDC (https://www.oclc.org/en/dewey.html) will be applied.

Source code

The software will be made to run at least on Linux, an industry-standard and free operating system. Efforts will be made to ensure that all the source code is portable to other platforms, in particular MacOS and Windows. For the Apps, we will aim to provide solutions for common mobile devices. Metadata will be created by applying the CodeMeta standard (https://github.com/codemeta/codemeta). In addition, the ACM Computing Classification System (https://www.acm.org/publications/class-2012) will be used as an ontology for metadata and documentation purposes.

2.4 Increase data re-use (through clarifying licenses)

- *Specify how the data will be licenced to permit the widest reuse possible*
- *Specify when the data will be made available for re-use. If applicable, specify why and for what period a data embargo is needed*
- *Specify whether the data produced and/or used in the project is useable by third parties, in particular after the end of the project? If the re-use of some data is restricted, explain why*
- *Describe data quality assurance processes*
- *Specify the length of time for which the data will remain re-usable*

HIVEOPOLIS will perform an open data publication plan and will work towards making as much data open as possible and to allow for third party use as soon as possible. An embargo period might be needed for publication reasons, e.g. doctoral thesis written within the project context.
To ensure reusability of the project outputs, we will use appropriate free and open, permissive licencing: for source code developed from scratch, MIT licence; for source code

that extends other open-source projects, we aim to adopt the most permissive license possible given the constraints of the software being extended; for other contributions we will use either the Creative Commons CC BY-ND or CC BY 4.0 International licenses, depending on whether it is deemed appropriate to permit derivative work or not.

Design files will be made open where possible, but we note here that some design tool vendors place restrictions, including for academic purposes, that may prevent us from releasing the entirety of designs. We have driven some vendors to change their policies in favour of more open access[2], but the issue is far from resolved.

The quality of the collected and created data is ensured by different measures. These include validation of the sample, replication, comparison with the results of similar studies and control of systematic distortion.
Extensive testing and review will ensure the quality of the software code. The software developed within HIVEOPOLIS will be structured as modular as possible to allow for easier testing and reusability. Separate parts of the software can be made public separately, e.g., as libraries. A special attention will be paid to the clarity of the code itself.

Chapter 3: Allocation of Resources

- *Estimate the costs for making your data FAIR. Describe how you intend to cover these costs*
- *Clearly identify responsibilities for data management in your project*
- *Describe costs and potential value of long term preservation*

Source code

Documenting and publishing the source code is part of the established work processes. It incurs no additional costs. The long-term storage will be valuable to researchers working on similar systems as well as for benchmarking their algorithms against ours.

Collected & created data

Each partner is responsible to provide storage and ensure backups for the local data generated by that partner during the project. The necessary storage solutions are provided by the corresponding universities or are part of already existing infrastructure. They incur either no additional costs or only small costs which are covered by the individual partner as part of the running costs. After the end of the project the relevant data will be archived at the https://zenodo.org.
HIVEOPOLIS will perform an open data publication plan. Partner UBER is the responsible open data channel manager and thus will select and publish data sets of special broader interest. Published datasets will be stored in an open-access repository with guaranteed long term accessibility, like https://zenodo.org, which incurs no additional cost.

[2] See, e.g. F. Mondada (2016). “Results held hostage: Hardware design software licenses holding back open science”. *Open Aire Blog*, 2016

The long-term storage will be valuable to researchers working in diverse areas for various activities including: i) developing and benchmarking their algorithms such as in image-based animal tracking; ii) analysis of measurements made of honeybees in diverse locations, at both individual and colony levels; iii) analysis of environmental data measured in diverse locations (without direct relationship to honeybees, per se); iv) develop and test various sensor fusion and modeling approaches.

Chapter 4: Data Security

- *Address data recovery as well as secure storage and transfer of sensitive data*

Source code

During the project, the source code will be versioned using GIT and stored in a redundant way in local repositories as well as in cloud based services like https://github.org. The transfer between the repositories is performed by default through encrypted channels (e.g., SSL, SSH). After the end of the project, the relevant repositories will be archived at an open-access repository with guaranteed long term accessibility, like https://zenodo.org.

Collected & created data

The data collected and created during the project will be stored at the partners local storage solution (see chapter 1.1). Please refer to the chapter 1.2 for detailed information regarding the storage solutions of the particular partners and corresponding backup procedures. The published datasets will be checked for privacy issues and stored at an open-access repository with guaranteed long term accessibility and backup, like https://zenodo.org, which has a retention policy of at least 20 years (at the time of writing).

User data

Data collected from the users will be anonymized and shared only with explicit consent of the users.

Chapter 5: Ethical Aspects

- *To be covered in the context of the ethics review, ethics section of DoA and ethics deliverables. Include references and related technical aspects if not covered by the former*

There are no ethical concerns regarding the data generated by the project that we are aware of at this time.

Chapter 6: Others

- *Refer to other national/funder/sectorial/departmental procedures for data management that you are using (if any)*

We will use the following policies and guidelines of the partner institutions:

- Research Data Management Policy of Humboldt-Universität zu Berlin
- Research Data Policy of University of Graz
- Guidelines on the Handling of Research Data in Biodiversity Research
- Guidelines for research data management at EPFL
- Guidelines on the data privacy in Latvia University of Life Sciences and Technologies

Useful Resources

- https://re3data.org - Registry of Research Data Repositories;
- https://datadryad.org - curated general-purpose data repository.
- https://zenodo.org - a general-purpose open-access repository.
- https://github.com - commercial cloud based hosting service for software development version control using Git.
- https://git-scm.com - GIT is a free and open source distributed version control system.
- https://fairsharing.org - A curated, informative and educational resource on data and metadata standards, inter-related to databases and data policies.

Links to public HIVEOPOLIS resources

- https://www.hiveopolis.eu
- https://zenodo.org/communities/hiveopolis
- https://github.com/hiveopolis