



ECOLOPES

ECOLOGical building envelOPES: a game-changing design approach for regenerative urban ecosystems

H2020-FET-OPEN-2021-2025

Action number 964414

D5.1

Development Process for ECOLOPES Algorithm

Dissemination level:	Public
Contractual date of delivery:	Month 12, 31 March 2022
Actual date of delivery:	31.03.2022
Work package:	WP 5
Task:	T5.1
Type:	Report
Approval Status:	Final Version for Submission
Version:	V0.2
Number of pages:	69
Filename:	D5.1_Ecolopes_DevelopmentProcessforECOLOPESalgorithm.docx

Abstract

Deliverable 5.1 outlines the approach to the development of the ECOLOPES algorithmic design process with the following objectives: (1) identification of a conceptual approach to urban form, (2) systematic approach to related types of data sets; (3) steps towards identifying a framework for a generative algorithmic design process; and (4) identification of principal approaches for linking the ECOLOPES algorithmic design process to the ECOLOPES voxel (WP5) model and EIM Ontology (WP4) and ensuring the required alignment with other parts of the ECOLOPES computational and user workflows.

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.



Funded by the European Union

HISTORY

Version	Date	Reason	Revised by
1	29.03.2022	1st Version	Shany Barath, Defne Sunguroğlu Hensel
2	31.03.2022	2 nd Version	

AUTHOR LIST

Organization	Name	Contact Information
TU Wien	Michael Hensel	michael.hensel@tuwien.ac.at
TU Wien	Tina Selami	tina.selami@tuwien.ac.at
TU Wien	Jakub Tyc	jakub.tyc@tuwien.ac.at
TUM	Defne Sunguroğlu Hensel	defne.hensel@tum.de



EXECUTIVE SUMMARY

This report outlines the development of the ECOLOPES algorithmic design process. This entails the following objectives: (1) identification of a conceptual approach to urban form, (2) systematic approach to related types of data sets; (3) steps towards identifying a framework for a generative algorithmic design process; and (4) identification of principal approaches for linking the ECOLOPES algorithmic design process to the ECOLOPES voxel (WP5) model and EIM Ontology (WP4, *D4.1 Preliminary EIM Ontology*), ensuring the required alignment with other parts of the ECOLOPES computational workflow (WP3 *D 3.1 Prototype technical requirement report*) and user workflow (WP1 *D1.3 Report of Year 1*). This work constitutes the first part of the algorithmic process that leads up to the detailed design process.

In the introduction we locate the items outlined in this report in the context of the overall ECOLOPES computational and user workflow. We then outline relevant datasets, tasks concerning linkage of the algorithmic process with the ECOLOPES Voxel model and EIM Ontology (*D4.1 Preliminary EIM Ontology*), as well as specific methodological aspects especially introducing research-by-design. The development of the ECOLOPES algorithmic design process in WP5 involves master-level architectural design studios that serve as testbeds for the development of the algorithmic approach. Design projects and feedback from the students serves to orient and illustrate specific aspects of the research activities in WP5. The identification of relevant datasets is based on a shift away from urban form and architectures understood as discrete systems and objects. Instead, we pursue urban form as continuous and continuously differentiated landform, to enable a more direct relation between ecological aspects, for instance species distribution and related dynamics, and the articulation of the “urban terrain”. Sections 2, 3 and 4 focus on outlining four types of related datasets (1) *terrain*, (2) *maps*, (3) *networks*, and (4) *volumes* that are elaborated systematically and algorithmically. The dataset *terrain* builds on and adapts an existing systematic approach and method to describe landform that is entitled “geomorphons”. We describe this method, outline its advantages and shortcomings, as well as necessary steps towards adapting this method for the algorithmic process. The datasets *maps* and *networks* are correlated with the dataset *terrain*. Context-specific *maps* describe for instance microclimatic conditions that exist prior to design interventions or that result from design interventions, i.e. terrain modifications. Context-specific *networks* describe, for instance, intended building program relations and distribution on site, intended human-nature or stakeholder (human, plants, animals) interactions, etc. Since the datasets *terrain*, *maps*, and *networks* are correlated, design can commence in different ways by changing priorities between different datasets, thereby ensuring a robust design framework that can facilitate different approaches to given sites and design briefs. Section 5 outlines steps towards identifying a generative algorithmic design process that serves the purpose of variety generation in design outputs and their evaluation and ranking. Section 6 focuses on how the algorithmic process can be linked with the



ECOLOPES voxel model (WP5) and the EIM Ontology (WP4). The studio brief for the current master-level studio ECOLOPES Kindergarten Wienerwaldrand 2 is appended as an example of a studio that serves as a testbed and input for the development of the algorithmic design process.



Abbreviations and Acronyms

CAD	Computer Aided Design
DEM	Digital Elevation Model
DTM	Digital Terrain Model
EA	Evolutionary algorithms
EIM	ECOLOPES Information Model
GH	Grasshopper
GIS	Geographic Information System
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
LiDAR	Light Detection and Ranging
LTP	Local Ternary Patterns
ML	Machine Learning
MRI	Magnetic resonance imaging
MSPA	Morphological Spatial Pattern Analysis
NURBS	Non-uniform rational B-spline
OWL	Web Ontology Language
RDF	Resource Description Framework
PCA	Performative Computational Architecture
SQL	Structured Query Language
UGI	Urban Green Infrastructure
WP	Work-package



Table of Contents

History	2
Author list	2
Executive Summary	2
1. Introduction	8
1.1 ECOLOPES workflow and architectural design cluster	8
1.2 ECOLOPES User Design Workflow	10
1.3 D5.1 Aims and Objectives	11
1.4 Urban Form as Continuous Terrain	12
1.5 Terrain, Maps, Networks, Volumes	13
1.5.1 Premise	13
1.5.2 Approach	13
1.6 Algorithmic Design Process, Voxel Model and Ontology	15
1.7 Research by Design & Studio as Test-bed	15
2 Dataset Terrain	17
2.1 Current Approach to Urban Form and Architectures	17
2.2 Systematic Approach to Terrain Features	18
2.2.1 Analysis of the Algorithm for generating Geomorphons	18
2.2.2 Advantages of the Geomorphon Approach	20
2.2.3 Shortcomings of the Geomorphon Approach	21
2.3 Algorithmic Approach to Terrain Features	22
2.3.1 Extending the Geomorphon Approach	22
2.3.2 Open Questions	24
2.4 Next development steps for terrain modeling	26
3. Dataset Maps & Networks	28
3.1 Systemic Approach to Maps and Networks	29
3.1.1 Design method development	29
3.1.2 Maps - Relations between multiple datasets	31
3.2 Algorithmic Approach to Maps & Networks	33
3.2.1 Analytical methods in urban design - historical review	34
3.2.2 Analytical methods in urban design - state of the art	34
3.2.3 Tool Evaluation	35
3.2.3 Open questions	36
4. Dataset Volumes	37
4.1 Systemic Approach to Volumes	38



4.2 Operational & Algorithmic Approach to Volumes	39
5. Generative Design	39
5.1 Systemic Approach to Generative Design	39
5.2 Algorithmic Approach to Generative Design	39
5.2.1 Definition of metaheuristics	40
5.2.2 Metaheuristic Algorithms	40
5.2.3 Evolutionary Algorithms	41
5.2.4 Next steps	41
6. Algorithmic Approach in relation to the Computational Workflow	42
6.1 Algorithmic Approach & Voxel Model	42
6.1.1 Outline of the algorithmic approach to the voxel models	42
6.1.2 Initial implementation of the algorithmic approach to the voxel models	43
6.2 Next steps towards linking the algorithmic process with the voxel model	45
6.3 Algorithmic Approach & EIM Ontology	48
6.3.1 Dataset Terrain & EIM Ontology	48
6.3.2 Dataset Networks & EIM Ontology	49
References	49
Source Code Citation	54
Web Pages	54
Appendix - Design Studio Brief	54



1. INTRODUCTION

1.1 ECOLOPES workflow and architectural design cluster

As specified in the grant application, the ECOLOPES project aims at developing a data-driven design recommendation system to assist architects and planners in the design of *ecolopes*, which entails the ecological adaptation of building envelopes in the design process. The overall goal is to provide the technology that enables an iterative design process based on the simulation of the dynamic development of the *ecolope*.

This technology involves five specific objectives (SO) that focus on:

SO1: development of the ECOLOPES computational platform (WP3);

SO2: development the ECOLOPES Information Model (EIM) Ontology (WP4);

SO3: development computational tools for modeling and visualizing the *ecolope* (WP5);

SO4: development a computational simulation environment (WP6)

SO5: demonstration of the effectiveness of the ECOLOPES design platform (WP7);

In order to progress towards this end the consortium outlined the ECOLOPES computational workflow (Fig. 1) (*D 3.1 Prototype technical requirement report*) and the ECOLOPES user design workflow (Fig. 2) (*D1.3 Report of Year 1*). As described in the *D 3.1 Prototype technical requirement report* the computational workflow consists of five primary parts: (1) open and expert databases (WP3, WP4), (2) ecological model (WP4), (3) EIM ontology and knowledge base (WP4), (4) the design generation environment (WP5) and (5) optimisation environment (WP6) (Fig. 1).

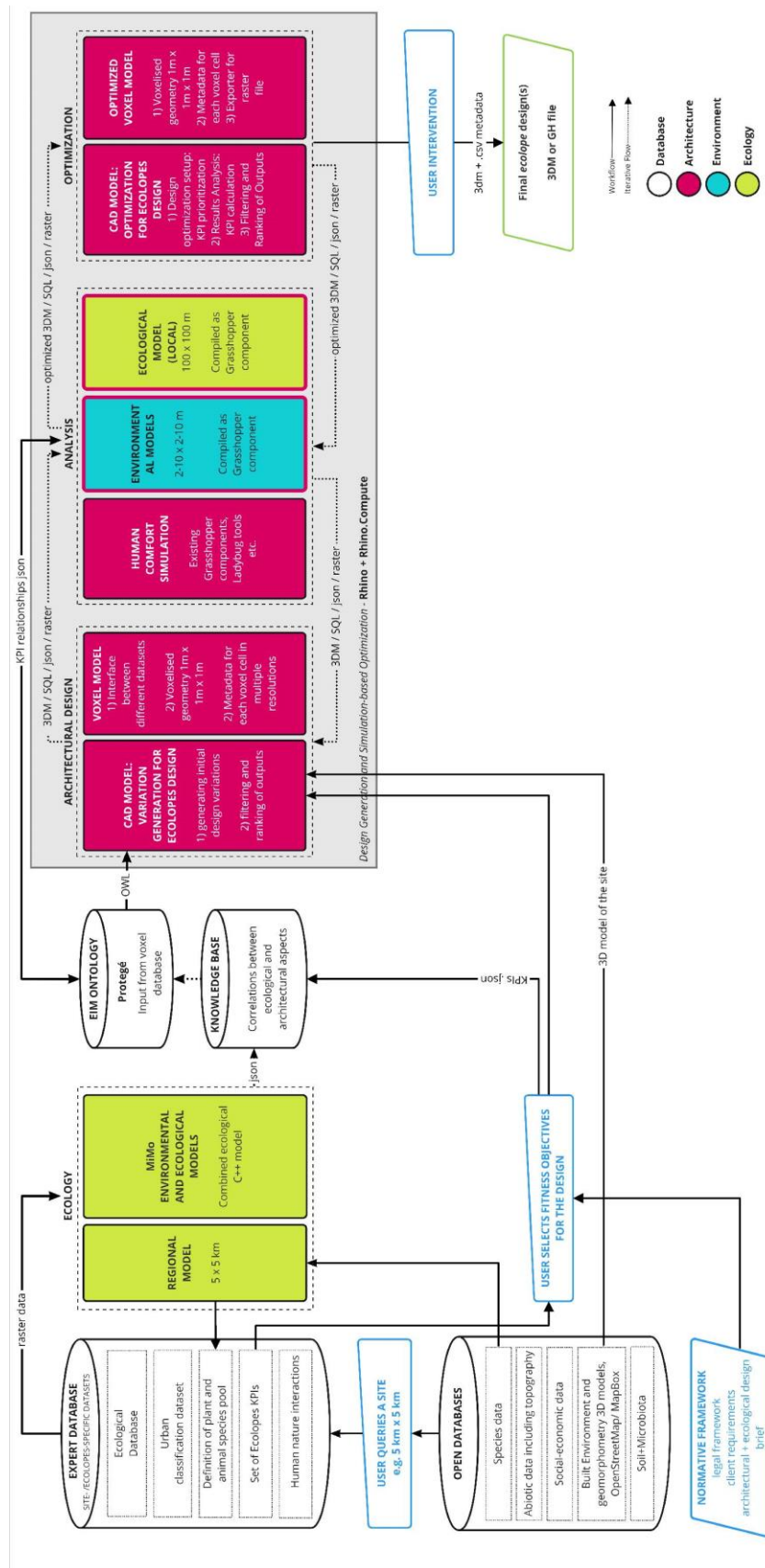


Fig. 1: Computational workflow. (Source: WP3 D 3.1 Prototype technical requirement report)



WP5 develops the design generation environment and has three key objectives: 1) development of the ECOLOPES Voxel model that links the EIM Ontology (WP4) with the computational model (CAD model); 2) development and integration of the ECOLOPES generative algorithmic design process within the Rhinoceros and Rhino.Compute framework implemented by McNeel; 3) validation of the algorithmic process that delivers the basis for the work in WP6 and WP7. The ECOLOPES computational workflow (Fig.1) (*D3.1 Prototype technical requirements report, section 3 the computational framework in ecolopes*) locates the deliverables of WP5 in the “Architectural Design” cluster shown in Figure 1.

Objective 2 of WP5 focuses on the development of the ECOLOPES algorithmic design process for 1) generating initial design variation and 2) filtering and ranking the different design outputs. D5.1 describes the first part of the development process of the ECOLOPES algorithmic process, which includes: 1) identification of relevant data sets (*terrain, maps, networks, volumes*); (2) identification of an approach to a generative algorithmic design process; (3) identification of approaches to linking the algorithmic process to the ECOLOPES voxel model and the EIM Ontology (WP4, *D4.1 Preliminary EIM Ontology*). The selected types of datasets facilitate the first part of the algorithmic process up to the detailed design process. The ECOLOPES Voxel model operates as an interface between different datasets that can incorporate expert information. The Voxel data will be written in an SQL database in different resolutions.

WP5 focuses on the generation of a variety of design options, from which a selection will be further optimized in WP6. The main inputs for the design generation process are EIM Ontology (WP4) and Expert Databases (WP4) and through those the ecological models (WP4), 3D models of built environment and topography from open databases, normative framework, and design brief. Georeferenced datasets and relevant variables for local building features have been identified in WP4. For the full list of variables and their respective descriptions we refer to *D4.1 Preliminary EIM Ontology*. A detailed description of the normative considerations in an *ecolopes* design can be found in the *D1.3 Report of Year 1*.

1.2 ECOLOPES User Design Workflow

The ECOLOPES user design workflow (Fig. 2) describes the approach to design from the user perspective. The user is conceived of as an interdisciplinary design team, consisting of different disciplines including architecture, landscape architecture, and ecology. The workflow is described in detail in *D1.3 Report of Year 1*.

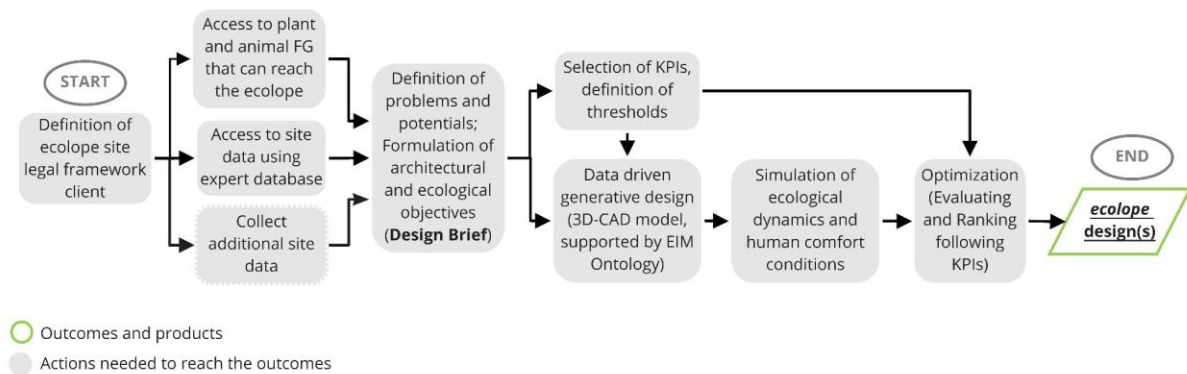


Fig. 2: User Design Workflow. (Source: Figure 2.5 in the Report of the first year)

WP5 focuses on the algorithmic design process that enables the user to apply generative design to create initial design variations in a data-driven way. Setting out a meaningful computational design workflow necessitates detailed consideration of the intentions of the user / designer and entails both maintaining an overview over the design process and intervening in it in a targeted manner. The design brief constitutes a major input by the user for the algorithmic processes. Early-stage designs are created as a combination of modeling in CAD, by selecting the required algorithmic tools and components and querying the EIM Ontology (WP4, *D4.1 Preliminary EIM Ontology*). The deployment of an ontology in an early design phase is an innovative approach and not common practice. As stated in the ECOLOPES Grant Application, the EIM Ontology defines the fundamental relationships between architecture, the abiotic environment, soil, plants, animals, and microbiota. These relationships serve as decision support and guide the design process. More detailed descriptions of the EIM Ontology are available in *D4.1 Preliminary EIM Ontology*. The connection to the EIM Ontology occurs through a query processed in the semantic web language Web Ontology Language (OWL) (W3C OWL working group. 2012). In this context, the selection of evaluation and ranking criteria and definition of thresholds and the preferred solution that is further developed, constitute major user decisions.

1.3 Aims and Objectives

As outlined above WP5 has three objectives: 1) development of a Voxel model that links the EIM Ontology from WP4 with the computational model (CAD model); 2) development and integration of a generative algorithmic process within the Rhino and Rhino.Compute framework implemented by McNeel; 3) validation of the algorithmic process that delivers the basis for the work in WP6 and WP7.

This report portrays the first stage of the development and integration of a generative algorithmic process within the Rhino and Rhino.Compute framework implemented by McNeel. More specifically this entails the first stage of the development process of the ECOLOPES algorithmic design process with the following objectives: (1) identification of a conceptual and systematic approach and related types of data sets; (2) steps towards identifying a framework for a generative algorithmic design process; (3) identification of



principal approaches for linking the ECOLOPES algorithmic design process to the ECOLOPES voxel model (WP5) and EIM Ontology (WP4, D4.1). The description of the identified types of datasets constitutes the first part of the algorithmic process leading up to the detailed design process.

The design generation components consist of the CAD model, the ECOLOPES algorithms and the ECOLOPES Voxel Model. The ECOLOPES Algorithms are subdivided into a set of algorithms for (1) data exchange between the CAD and the Voxel model, (2) design variety generation algorithms and (3) output filtering algorithms. Ecological models as well as environmental analysis models can be connected in an iterative loop within the generative design process. In the next steps include evaluation as to which specific models will be included in this process. Solutions that fulfill required conditions will be filtered and ranked according to defined thresholds. The user intervenes by selecting the solutions that will be further developed in the optimization process (WP6).

The development of an algorithmic approach for the purpose of urban and architectural design requires clarification as to *what* is being designed and *why*. This needs to take place prior to addressing the *how* and necessitates a systematic approach to *what* is being subjected to transformative action, i.e., planning and design. For the purpose at hand, this concerns the understanding of (1) urban form, and the identification of a related system by which to describe different instances of urban form, and (2) the defining traits of architectural objects, i.e., buildings. For this reason, the currently prevailing understanding of urban form needs to be examined in order to establish whether the current approach is adequately suited for the task at hand, or whether it is necessary to change the conceptual approach such that it is better suited for the purpose of the ECOLOPES project.

1.4 Urban Form as Continuous Terrain

As mentioned above it is necessary to establish *what* is being modeled, that is how to address urban form and architecture. We opted for a framework for understanding urban form and architectures as a continuous yet differentiated terrain to enable a useful exchange between ecological model data and architectural model data. In this approach individual architectural interventions constitute well-defined instances of the continuous urban terrain.

An understanding of urban form based on discrete systems and objects, that is on cumulative partitioning and segregation, runs evidently counter to the overlapping and extensive territories and multi-species *dwelling spaces*. Operating on a pre-established system of nested boundaries will foster an increasing number of conflicts. Negative impacts can hence mainly be addressed through seeking out best possible trade-offs for an increasing number of conflicts. Rethinking the overarching approach can result in a different perspective for which the requirements of ecosystems and involved stakeholders form the basis, and from which the traits of non-discrete urban form and architectures can be derived. Such an approach eventually requires rethinking essential aspects of current governance, ownership, rights,



access, etc. Still, tackling the consequences of reorientation towards a coherent approach justifies the effort when compared with the continuous struggle to negotiate the contradictory tension between perpetual partitioning of urban form on the one hand, and the need for continuous and extensive territories on the other hand.

1.5 Terrain, Maps, Networks, Volumes

1.5.1 Premise

Understanding urban form as continuous *terrain* and architectures as instances in this terrain delivers the basis for an ecosystem-focused approach to planning and designing multi-species territories and exigencies in conjunction with the transformation of environments necessitated by the need for construction. In this context several questions arise: (1) How can a continuous terrain be systematically described and organized as identifiable instances that can be transformed by design, while at the same time not defining a new system of physical segregation? (2) What can constitute meaningful design input / datasets for transforming the continuous urban terrain through design interventions? (3) How can a step-by-step and iterative design process be developed that is based on the approach to 1. and 2.

1.5.2 Approach

Systematizing urban form as continuous terrain can serve to set out a trans-scalar approach across spatial, temporal, and functional scales. Terrain features, such as saddle, valley, or ridge, etc., can vary in scale and size and be nested within one another. Nested configurations can, for instance, yield heterogeneous microclimatic conditions, and soil, water, and biomass distribution, thereby providing the range of differentiated conditions that are needed to provide for ecosystem exigencies. In turn this can facilitate a broad range of possible human-nature interactions.

The question that follows from this concerns the selection of relevant types of datasets. The latter constitutes “a collection of similar data, sharing a structure, which covers a fixed period of time” (OECD Glossary of Statistical terms 2001). For the algorithmic process up to the detailed design phase four types of datasets are necessary. The first type concerns landform and terrain articulation. A second type is required that represents conditions such as microclimate in terms of spatial and temporal dimensions. A third type of datasets is required in which desired relations between items (terrain and microclimate, building programs, stakeholders, human nature interactions, etc.) can be defined. Finally, a fourth dataset is required that describes the required spatial volumes for architecture, biomass, and soil. Furthermore, it is necessary to establish how these four types of datasets can inform one another in a design process. This entails elaboration of a systematic, operational, and algorithmic level.



For the second and third type of dataset we considered *maps* and *networks* respectively. *Maps* can describe for instance intensity, as well as spatial and temporal extent of microclimatic conditions. *Networks* can describe sets of items and their relations such as building program relations, species relations (i.e., food webs), existing or intended human-nature interaction or architectural program distribution, as well as across in combined networks, such as building program and stakeholder relations, or map and network combinations, such as building program and microclimatic conditions.

The correlation between the datasets (1) *terrain*, (2) *maps* and (3) *networks* can form the basis for an iterative design process in which changes in one dataset can yield corresponding changes in another. A first anticipated outcome of this algorithmic process is the specific arrangements of correlated architectural, biomass and soil *volumes* that can be evaluated and subsequently form the basis for a more detailed design process.

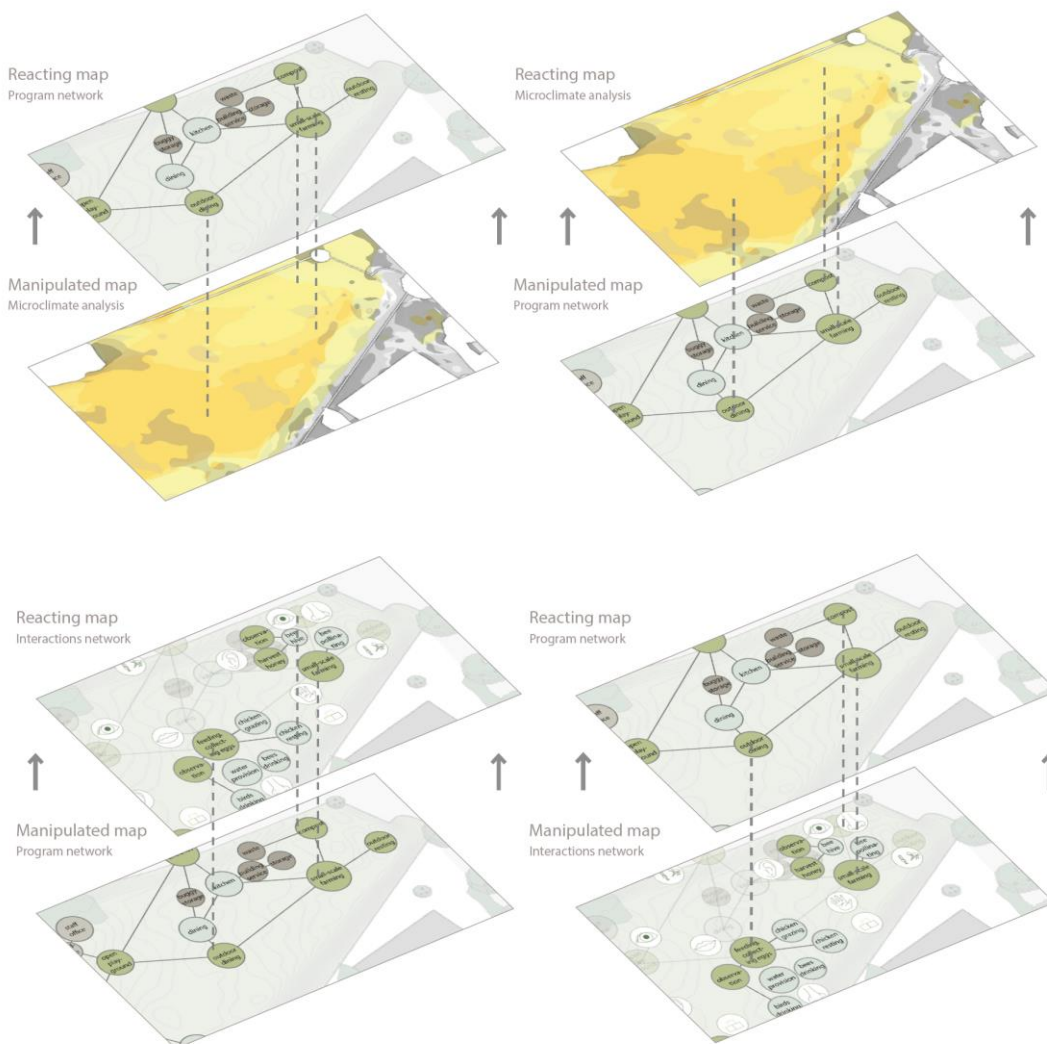


Fig. 3: 2021-22 Master Studio ECOLOPES Kindergarten Wienerwaldrand (Students: Juliana Schuch, Filip Larsson): switching input and output relations between different maps and



networks, i.e. microclimate and program network (top) and intended human-nature interaction and program distribution.

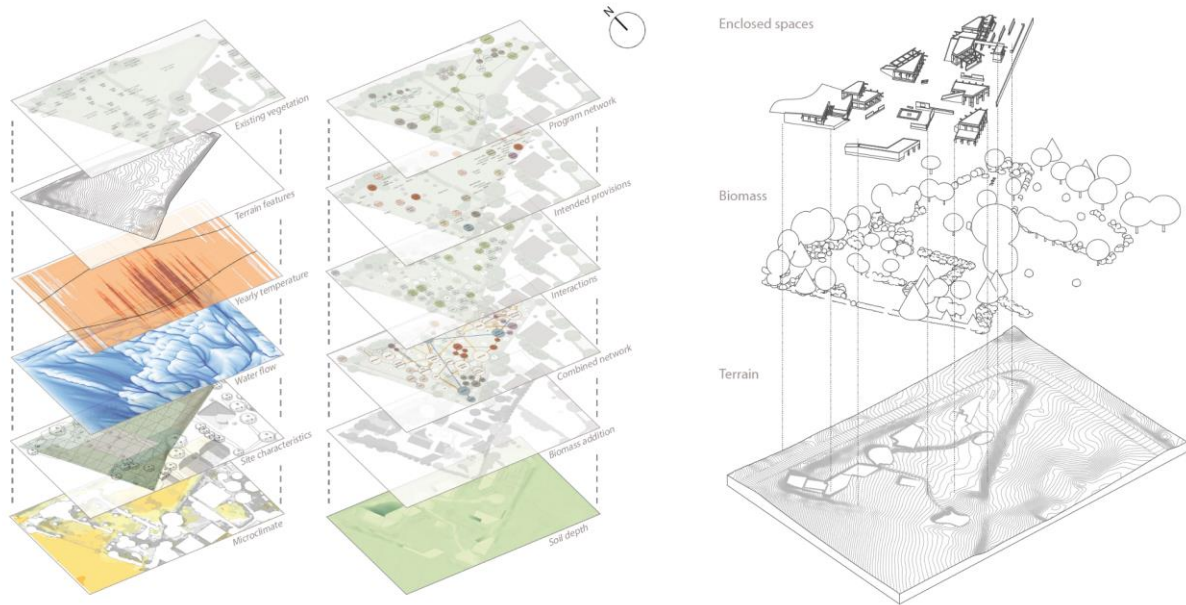


Fig. 4: 2021-22 Master Studio ECOLOPES Kindergarten Wienerwaldrand (Students: Juliana Schuch, Filip Larsson): selected maps and networks datasets (left) and developed architectural volumes, biomass volumes, and terrain / soil volume.

1.6 Algorithmic Design Process, Voxel Model and Ontology

Based on the above outlined approach (Fig. 1) we pursue a threefold communication between algorithmic process, voxel model and EIM ontology (WP4, D4.1): (1) The ontology suggests relevant datasets in relation to a specific site and design brief. (2) Relevant datasets are loaded into the voxel model, from where they are provided to the generative design algorithm. (3) The ontology can be queried by the designer to configure relevant networks and their interaction, as well as to uncover potential co-benefits or trade-offs between different requirements and related datasets. The algorithmic design process serves the purpose of generating a variety of possible designs and evaluates them in relation to specified benchmarks (Rittel 1970). More detailed descriptions of the EIM Ontology are available in *D4.1 Preliminary EIM Ontology*.

1.7 Research by Design & Studio as Testbed

In WP5 we employ different modes of research, including *research by design*. The context for this is a series of master-level design studios at Vienna University of Technology. These studios serve as testbeds for the development of the algorithmic approach and design intentions and



feedback from the students serves to orient related research activities outside the studio context.

The question as to what research in the creative disciplines is and what are domain-specific modes of research has occupied numerous thinkers over several decades (see for instance Frayling 1993). In this context various modes have been described including *research by* or *through design*.

Research by design or *research through design* is a method of inquiry that has become an established mode of research in architecture, urban and landscape architecture (Lenholzer et al. 2017). In this context different research strategies namely *qualitative*, *quantitative*, and *mixed* have been proposed as well as four different overarching types of inquiry that include *(post)positivist*, *constructivist*, *transformative*, and *pragmatist* (Creswell 2014). The development of the ECOLOPES algorithm requires *qualitative*, *quantitative*, and *mixed* strategies due to the need to correlate different datasets in an instrumental manner. Lenholzer et al stated that the output of what they termed research through design is “new knowledge that is applicable in design practice or in further research” (Lenholzer et al. 2017, 59). Furthermore, Lenholzer et al described the four types of inquiry as follows:

The *(post) positivist* type is predominately used in engineering-oriented inquiries, and frequently based on computer simulations or mock-ups, that are evaluated by employing predominantly quantitative feedback loops to reach an optimized solution. The *constructivist* type focuses on generation of designs predominantly through qualitative feedback loops. This involves evaluation with focus on cultural, aesthetic, ethical or other social values, that can include different stakeholders ranging from experts to citizens. The inclusion of the latter can give rise to the *transformative* type, which aims at co-creation and co-ownership of knowledge and process. However, Lenholzer et al suggested that the *pragmatist* type may be the most significant type of research through design and focuses on the research problem and “pluralistic approaches to derive knowledge about the problem and its solution” (Lenholzer et al. 2017, 60).

In the context of the development of the ECOLOPES algorithm we operate in the main on the *(post)positivist* and the *pragmatist* type, with the former aiming for deductible knowledge and verified theory / design guidelines via design hypothesis testing, and the latter mixing research methods to derive new practice-oriented knowledge including new design knowledge (Lenholzer et al. 2017). To undertaking research by design we are running a series of six consecutive master-level design studios in the context of the Department of Digital Architecture and Planning at Vienna University of Technology. The first studio took place during the winter semester 2021-22. This studio is completed and has significantly contributed to the development of the algorithmic design approach thus far. We are now running the



second studio to further develop this approach through research by design (see appended studio brief).

2 DATASET TERRAIN

2.1 Current Approach to Urban Form and Architectures

Latour proposed a new actor that “brings together the opposing figures of the soil and the world”, which he termed the *Terrestrial* (Latour 2017, 92). Latour stated that *soil* refers to materiality, heterogeneity, strata, etc., while the *world* refers to the different forms of existence. For Latour this necessitates a redefinition of *dwelling space*, “as that on which a terrestrial depends for its survival, while asking what other terrestrial also depend on it” (Latour 2017, 95). Furthermore, Latour points out that “it is unlikely that this territory will coincide with a classical legal, spatial, administrative, or geographic entity. On the contrary, the configurations will traverse all scales of space and time” (Latour 2017, 95).

The examination of the prevailing understanding of urban form and the architectural object entails a closer look at urban morphology as the field of study of urban form(s) (Cowan 2005, Araújo de Oliveira 2016).

Araújo de Olivera elaborated that “urban form refers to the main physical elements that structure and shape cities - streets (and squares), street blocks, plots, and common and singular buildings, ...” (Araújo de Olivera 2022, p. 2). The understanding of urban form as a set of discrete items and systems, evolved from a tradition of surveying (as can, for instance, be seen in the work of Giambattista Nolli, e.g., in his survey of Rome (1736-1748)). This survey is shown as a so-called *figure-ground* map, which separates a figure (building) from a background (urban surface), and thereby built space (object) from unbuilt space. This separation of figure from ground is based on systematic dissociation and consequently leads to an understanding of the architectural object as a discrete item (Hensel 2013), i.e., building, that is clearly set apart from its surroundings. In this context the question arises whether an approach based on dissociation of items and hence emphasis on boundaries and discontinuity is best suited as a framework for planning and design when the design task requires correlating and operating on overlapping and extensive territories and exigencies of different actors in urban ecosystems. This question can be formulated as follows: *What is the most adequate framework for understanding urban form and architectures in the context of the objectives of the ECOLOPES project?*

Araújo de Olivera pointed out that urban morphology concerns “urban forms, and the agents and processes responsible for their transformation” (Araújo de Olivera 2022, p. 2). This implies that urban form is defined as a set of discrete items and systems, prior to identifying the agents and processes that act upon and transform these items and systems. In this context the second question arises, which concerns the identification of agents and processes that act



upon and transform exigencies and relations between different actors in urban ecosystems. This question can be formulated as follows: *What are the agents and processes that act upon urban ecosystems given an alternative framework for understanding urban form and architectures that is not predicated on the separation of discrete systems and objects?*

2.2 Systematic Approach to Terrain Features

The above-described approach to urban form as continuous terrain necessitates a systematic way of defining and identifying terrain features and can be based on the field of geomorphometry (Lobeck 1939), which is the scientific analysis of *land surface* (Pike et al. 2009). Land surface is understood as continuous. General geomorphometry is used to analyze landforms that are “bound segments of a land surface and may be discontinuous” (Evans 2012). Specific geomorphometry entails the analysis of geometric and topological traits of landforms (Evans 2012). A clear definition and delineation of landforms can pose ontological challenges (Smith and Mark 2003). However, geomorphometric parameters (Mark 1975) can describe the morphology of the land surface and geomorphons can be used to classify and map landform elements. Recent research outlined a “*novel method for classification and mapping of landform elements [...] based on the principle of pattern recognition*” (Jasiewicz and Stepinski 2013). The landform elements described in the geomorphons approach are shown in Fig. 4. We aim to extend and adapt this method to the study and development of urban form understood as continuous terrain.

2.2.1 Analysis of the Algorithm for generating Geomorphons

Recent research characterized *geomorphons* as a pattern-recognition based approach to classify and map landforms (Jasiewicz and Stepinski 2013). Geomorphons are based on Local Ternary Patterns (LTP) that describe the relation of cells on a 3x3 grid (defined by degrees rather than metric units), consisting of a central cell (conventionally numbered 0) and 8 neighboring cells (numbered from 1 to 8) (Fig. 3). Local Ternary Patterns in geomorphons encode information that describes whether neighboring cells are located higher, lower or at the same height as the central cell. Small differences in height are filtered with the application of the threshold angle parameter (t), set to one degree by default. For graphical representation, refer to Figure 2 in Jasiewicz and Stepinski, 2013.

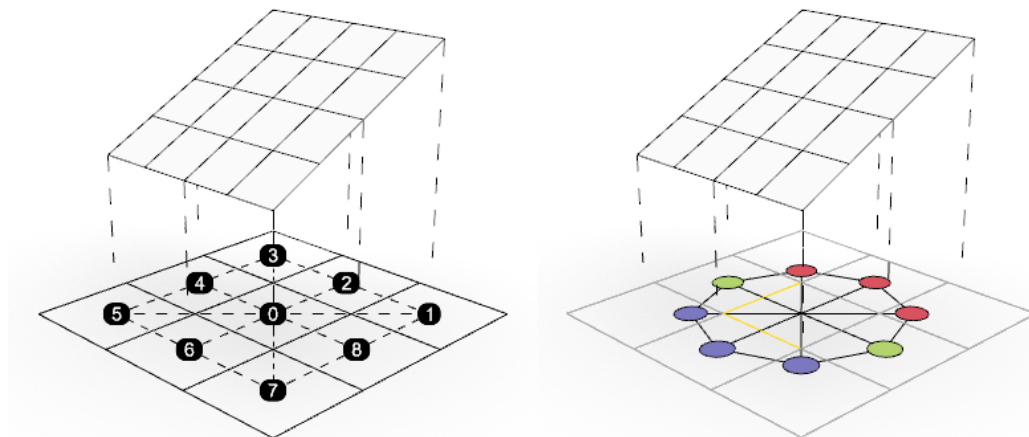


Fig. 5: 3x3 cell neighborhood of a single geomorphon shown on the left side. Cell numbering convention taken from source code (Jasiewicz & Stepinski [2017] 2022). Ternary patterns describing geomorphons (left) are often represented as octagons with colored nodes that are related to the 8 neighboring cells (right). The red color indicates points that are higher than the central cell, green refers to the points that have the same height and blue points that are located lower than the central cell. Geomorphons represent 3D objects or landforms that can be visualized as a 3D mesh, as shown above each of the symbolic representations in this figure.

For each of the eight neighboring cells three relative height values are possible. This results in 6561 possible combinations. Many combinations are reflections or rotations of the same pattern, and the removal of such duplicates results in 498 unique patterns. However, this number of different patterns is still of limited applicability. Jasiewicz and Stepinski stated that “for a general purpose the number of landform elements in a morphometric map needs to be reduced” and proposed a lookup table (Fig. 5) (Jasiewicz and Stepinski 2013, 149). Geomorphons are represented in the algorithmic implementation as a sequence of eight symbols, labeled either as 0, or as + or -. These symbols represent the location of each of the 8 neighboring cells in relation to the central cell (see the example in Fig. 4). Both ‘+++00000’ and ‘00+++000’ sequences represent the “footslope” geomorphon. The difference is only related to the rotation. The de-duplication procedure assigns the name “footslope” to both of those sequences and related landforms. However, there are some sequence combinations that are not trivial to filter since they depart slightly from the archetypical geomorphon form.

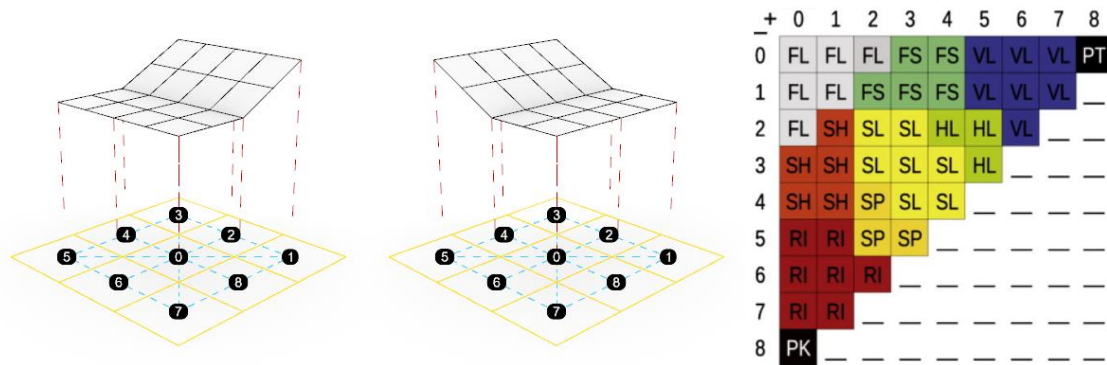


Fig. 6: Two sequences encoding the same ‘footslope’ geomorphon, differing in rotation (left), lookup table (right) (Jasiewicz & Stepinski 2013).

2.2.2 Advantages of the Geomorphon Approach

Geomorphons gained wide recognition in the field of geomatics, when the implementation (Jasiewicz and Stepinski 2013) was contributed to the open-source GIS community, thereby making it available in all major open-source GIS packages. Geomorphons represent *tangible 3D objects* (Fig. 6) that “constitute a comprehensive and exhaustive set of idealized landform elements” (Jasiewicz and Stepinski, 2013, 149).

Alternative landform classification methods that are, for instance, based on maximum and minimum curvatures or on profile and tangential curvatures do not cover such a wide range of landforms and require detailed understanding of concepts related to geomorphometry. In comparison geomorphons constitute a useful and easily comprehensible method for describing surface topology, especially in the context of interdisciplinary research.

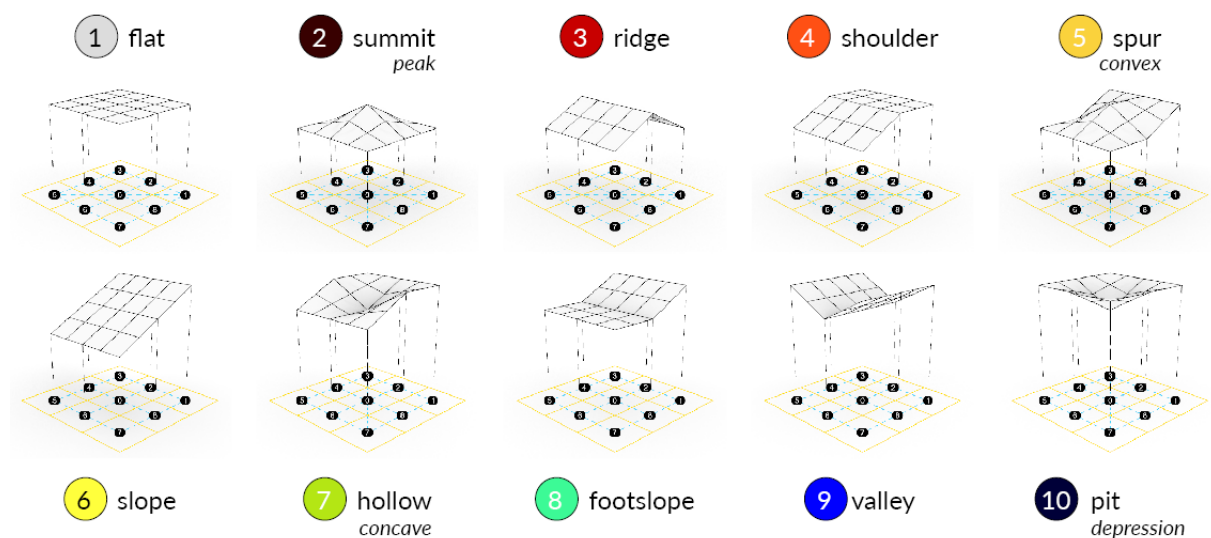


Fig. 7: Symbolic representation of the 10 most common landform elements (Jasiewicz and Stepinski, 2013). This visualization and related color coding and numbering became the de



facto standard for describing geomorphons. Based on the original implementation contributed by Jasiewicz and Stepinski to the open source GrassGIS codebase, geomorphons are currently implemented in QGIS, GrassGIS, SAGA GIS, etc. Colors, numbering, and alternative names are based on the original code (Jasiewicz & Stepinski [2017] 2022)

2.2.3 Shortcomings of the Geomorphon Approach

A common misconception relates to the fact that the values analyzed computationally in the 3x3 grid are commonly sampled from cells that are not directly located next to each other. This feature is related to the lookup distance parameter (L) that enables the algorithm to classify terrain features of different scales in one run. For example, there might exist small “ridges” on the two sides of a “valley” landform. The lookup distance can be fine-tuned to filter out the small “ridges” to successfully identify the “valley” landform. On the implementation level, the lookup distance is combined with zenith and nadir angles of the terrain to identify intersections of the zenithal and nadiral lines-of-sight or rays with the terrain, within the range defined by lookup distance. This property of geomorphons is crucial for its applicability in the context of terrain analysis, but limits direct applicability in the context of terrain generation.

Geomorphons are based on a 2.5D definition of the terrain surface. This is a common property of terrain models used in geoscience, referred to as Digital Elevation Models (DEM) or Digital Terrain Models (DTM). A heightmap is a concept that is similar to the 2.5D terrain model that is widely adopted in architecture. Heightmaps, a term originating from computer graphics, are established in architecture, because of the widespread adoption of specialized 3D modeling tools used for architectural visualization.

Heightmaps and displacement maps are often applied to a subdivided planar surface to create detailed three-dimensional surfaces that can be geometrically complex. However, topological changes are not possible, which implies that discontinuities and undercuts are not describable. 2.5D terrain models share these characteristics. However, topological changes might be necessary when three dimensional solutions are required that would alter connectivity within an urban fabric. A second shortcoming of heightmaps, displacement maps, and geomorphons alike is that it is not possible to describe vertical surfaces. This is a major shortcoming in relation to modeling urban form and buildings. Furthermore, there is only one height value for each point or cell of the terrain model or pixel in the heightmap.



2.3 Algorithmic Approach to Terrain Features

2.3.1 Extending the Geomorphon Approach

We seek to extend the geomorphon approach with the aim to enable full computational analysis and design of urban form and architectures as continuous terrain. In reference to surface topography, we refer to the proposed extension of geomorphons as topographic patterns. In the following part we discuss how the identified limitations can be addressed. Subsequently we outline how the proposed topographic pattern method can be constructed. To address the limitation related to the scale adaptation mechanism, we anticipate that topographic patterns would need to exclude the logic that incorporates lookup distance combined with zenith and nadir angles. Instead, fixed 3x3 cell neighborhoods in different scales are proposed for the generation of urban form as continuous terrain. The 3x3 cell neighborhoods are based on the original LTP algorithm related to the field of image pattern recognition and the size of the 3x3 neighborhood cannot be changed. At the same time the cells are not constrained to any physical dimension and can therefore be potentially applied at different scales.

The limitations related to the 2.5D representation of the terrain are implicitly addressed in the ECOLOPES project given the 3D environment of the Rhinoceros and GH software. The logic of Rhinoceros and GH is different from the original context in which geomorphons are implemented.

We consider four groups of topographic patterns to address the shortcomings,

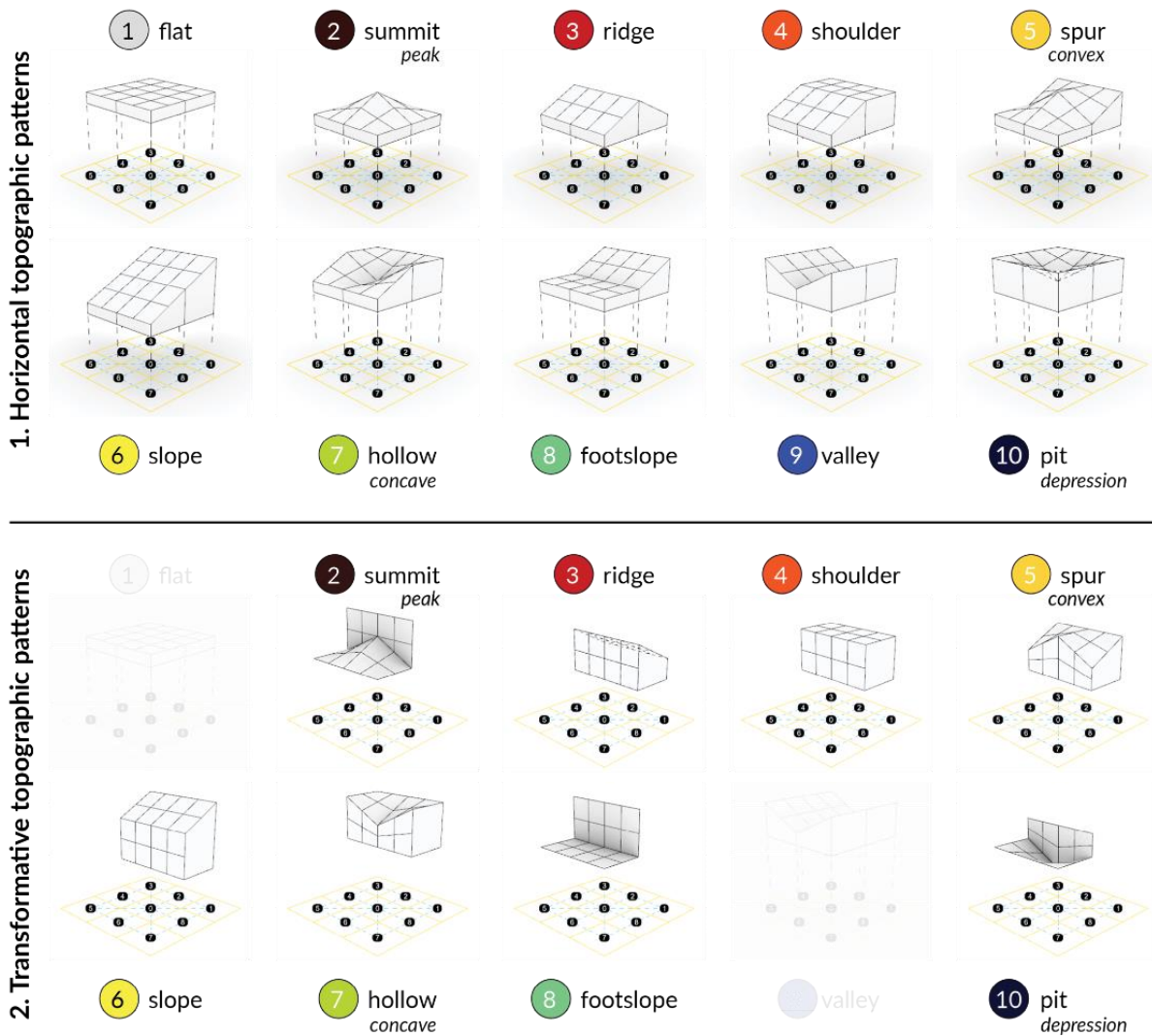
1. Horizontal topographic patterns, which are equivalent to the original 2.5D geomorphons, with the exception that the 3x3 neighborhoods are created from cells directly neighboring each other.
2. Transformative topographic patterns, which allow for transformation from the horizontal plane (2.5D surface) to the vertical direction.
3. Vertical topographic patterns, which translate the language of the horizontal topographic patterns to the vertical surfaces of buildings.
4. Inverted horizontal topographic patterns, which are equivalent to the horizontal topographic patterns, but applied to the bottom sides of the undercut volumes. The underside of a bridge is one example of an inverted horizontal topographic pattern.

We anticipate that adopting vocabulary related to landforms, such as ‘ridge’ or ‘spur’, would make it easier to communicate the formal expression of intended design. We propose architectures as distinct instances in the continuous terrain. Architectural intervention in the continuous terrain can be expressed as a collection of volumes described by edges that are an integral part of the terrain. Word “ridge” can describe at the same time an architectural



element of a gable roof and a specific landform. Designing within the proposed paradigm will accommodate geometric continuity of the *ridge geometry* between the landform and the architectural form. Hence, instrumentalizing abstract geometries and related naming conventions, (Fig. 7) will serve to align the language of landform and architectural form.

Geomorphons can be mapped to equivalent topographic patterns in each of the four groups described above (Fig. 7). For the proposed groups of topographical patterns further research is needed to establish whether all possible combinations are exhausted, and no redundant patterns are created. This is important, since our approach adopts the vocabulary related to landforms. Commencing from geomorphologic forms delivers clear criteria for evaluation and validation of the findings. The listed shortcomings are addressed in the following section.



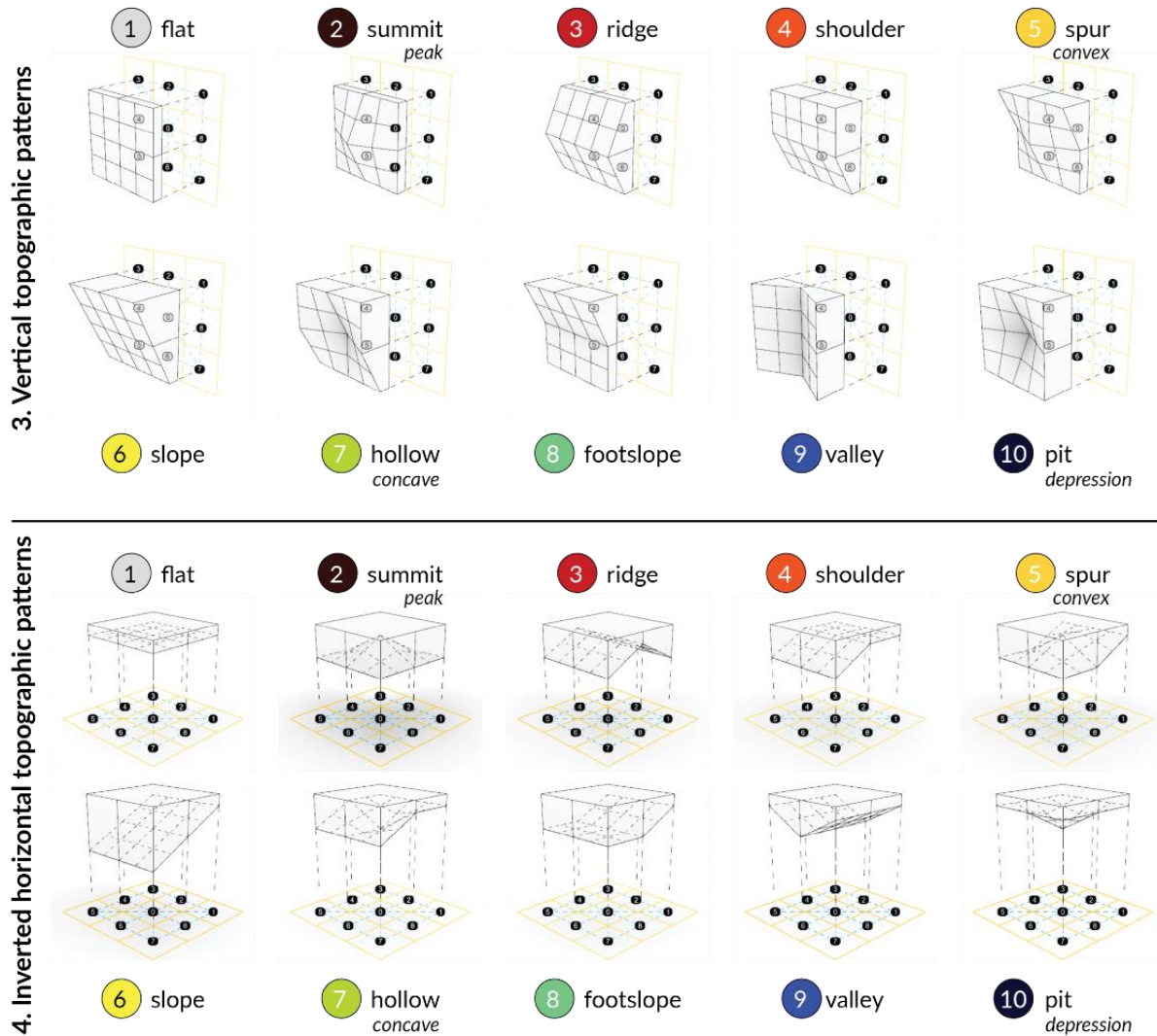


Fig. 8: Overview of the topographic patterns that can be derived from the original geomorphons assigned to the four topographic groups.

2.3.2 Open Questions

Based on the analysis of the geomorphons algorithm, identification of shortcomings and discussion related to the possible extensions, the following open questions arise:

- What are the methods for translating designs incorporating habitat provisions on vertical surfaces into a representation that is compatible with the georaster based ecological models?



- How can analysis and generation of complex patterns inherent in natural landscapes be enabled through the inclusion of topographic patterns in multiple scales and within a single location?
- What is the best strategy for testing the applicability of the Docofossor plugin in this context? Can it combine 3D NURBS curves and surfaces with continuous, raster-based terrain models utilizing a user-defined slope angle parameter?

In the ECOLOPES project we are developing a computational framework that links computational ecological models and design processes. The ecological models configured in WP4 operate on a raster-based representation of the terrain and features such as walls or bottom sides of the undercut volumes cannot be incorporated. However, Animal-Aided Design guidelines for instance, feature strategies to integrate provisions for wild animals on or in vertical surfaces. The application of currently available ecological models to simulate such environments would require methods to translate designs incorporating habitat provisions on vertical surfaces into a representation that is compatible with the georaster based ecological models.

A further challenge is related to the multi-scalar character of urban form, especially in its understanding as continuous terrain that can feature nested landforms. The aim is to enable analysis and generation of complex patterns inherent in natural landscapes through the inclusion of topographic patterns in multiple scales and within a single location. This logic applies not only to relations between objects in different scales but also to the design of an element in a single scale. For example, in the case of a facade design, the main pattern might be defined by the horizontal “spur” pattern, transformations between the vertical surface might be described with “summit” and “ridge” transformative patterns and the vertical part of the surface might contain multiple, vertical “hollows” and “ridges” with different dimensions.

We have identified a relevant tool in the Rhinoceros and GH software ecosystem, the Docofossor plugin (Hurkxkens and Bernhard 2019). This tool establishes the initial interface between 2.5D terrain models and the mesh and NURBS based modeling capabilities of Rhinoceros and GH tools. The Docofossor plugin was made for a different purpose related to large scale terrain modeling in the context of landscape architecture and digital fabrication. Its applicability for the envisioned functionality will be tested and evaluated.

Jasiewicz et al pointed out that “while traversing a LTP in order of principal directions a ‘transition’ is a change in ternary element” (Jasiewicz et al. 2013, 150). This implies that some of the 10 common geomorphons have specific “transitive equivalents”. Jasiewicz et al further elaborated that “shoulder, spur, hollow, and valley have 2 transitions, while ridge, valley, and slope have 4 transitions. Geomorphons with more complicated morphologies, for example, a ‘saddle’, have more transitions (up to 8) and are relatively rare” (Jasiewicz et al. 2013, 150). Therefore, the 3D representation of ‘transitive geomorphons’ will be studied and evaluated regarding their relevance for the envisioned application in the context of topographic



patterns. The need for the transitive elements might depend on the method used to integrate the elements in the Rhinoceros environment. Existing implementation of distance functions for terrain modeling is available in Rhinoceros through the Docofossor plugin (Hurkxkens and Bernhard 2019). The Docofossor plugin makes it possible to combine 3D NURBS curves and surfaces with continuous, raster-based terrain models utilizing a user-defined slope angle parameter. In result, transitions between NURBS geometries and terrain models are solved by the application of signed distance function. Initial experiments with the GHPython based Docofossor plugin have shown constraints related to computational performance, particularly when working with high-resolution data. This performance constraint might be hard to address within the constraints related to the GHPython technology, based on the IronPython implementation. At the same time, we experienced compatibility issues related to Docofossor and personal computers in the context of the master-level design studios. This concerns the compatibility of McNeel's internal implementation of IronPython and the Apple M1 hardware platform, which rendered Docofossor and several other plugins unusable.

2.4 Next development steps for terrain modeling

We are proceeding with the application of horizontal and transformative topographic patterns in the Rhinoceros and GH environment and continue to evaluate the Docofossor plugin (Hurkxkens and Bernhard 2019). Based on student feedback from the master-level design studio, the application of Docofossor outside of the original context related to large scale landscape modifications is challenging for the students. It was challenging for some of the students to grasp advanced concepts related to computational terrain modeling that were originally developed for a computational landscape design curriculum. To address this challenge, we are preparing software workshop sessions that will introduce key concepts related to computational terrain modeling and geomorphology (geomorphons). Furthermore, we will align these topics to concepts that are understood in the field of architectural design. For example, geomorphons will be introduced based on a rich graphical content, foregrounding the tangible 3D character of landforms, and referring to analogies with building construction elements such as ridge and gable roof.

Secondly, the application of the Docofossor plugin is challenging on the computational level in relation to architectural objects and their geometric details. To some degree, this is due to the resolution required to represent such objects is considerably higher than the resolution required to design large scale terrain modifications. The Docofossor plugin reduces the resolution of terrain models (dfGridFilter) to select fragments of the terrain for processing and display (dfGridRegion). These components can help to reduce the resolution of the terrain mode for the "reason of speed gains", as suggested by the authors of Docofossor (Hurkxkens and Bernhard 2019, p.224). However, we need to test and evaluate whether the terrain model preprocessing, resampling, and cropping can be done efficiently in GH and Docofossor, or if



separate terrain models in different resolutions will need to be prepared in advance in GIS software.

More importantly, building up on the discussion about the geomorphons as the analytic component and topographic patterns as the generative component, different resolutions can be used to address the multi-scalar character of urban form in relation to the proposed generative processes. We argue that progressing from the terrain analysis to continuous terrain generation requires the ability to construct complex patterns present in natural landscape forms often consisting of patterns in different scales, overlaid in one position in space. This hypothesis will be tested in relation to the horizontal topographic patterns with the Docofossor plugin and 3D surface-based representations of geomorphons generated for different scales or grid resolutions.

Rhinoceros software is combined with GH starting from version 6. The capabilities of GH can be extended with many community-created plugins that are hosted by McNeel. As we progress with the research, we will continuously monitor the food4rhino platform in relation to new software developments that can be relevant for the functionality required by our terrain modeling approach.

Progressing beyond the horizontal topographic patterns requires going beyond the functionalities available in the Docofossor plugin. Docofossor is licensed on MIT license and distributed as a compiled .ghpy assembly. However, the source code is not available. Only the information about the algorithms that are implemented in this tool is currently available (Hurkxkens and Bernhard 2019). Distance functions are used but limited to 2D raster-like data. One way of progressing is to implement similar distance function logics but applied to three dimensions. Another possibility is to study similar algorithms implemented in different open-source GIS packages (e.g., buffer and distance raster functions in GRASS GIS) and to analyze in how far the computational logics of Docofossor can be generalized to terrain modeling methods available in the open-source GIS software, both in relation to 2.5D and 3D data. A third possibility is to research algorithms that can be applied for the proposed terrain modeling and generation approach. These approaches would need to be evaluated in relation to their applicability within the Rhinoceros NURBS based modeling interface.

Recent developments of Rhinoceros and GH resulted in the release of the Hops component (Baer, Davidson, and Payne 2021) that makes it possible to interactively link native Rhinoceros objects with an external Python interpreter. This is based on an interface that utilizes web technologies, for details refer to D3.1: Prototype technical requirements report - 2.4.3 The Grasshopper Hops component. Hops component is available as a GH node that can communicate with dedicated web servers that are hosted online or that run on the same computer. This integration is based on the Python Flask library and open-source McNeel gh_hops_server component (Robert McNeel et al 2021). The component utilizes widely used Python web server technology that exposes GET and POST methods of the HTTP protocol. This approach enables bidirectional data exchange between the Python interpreter and local



Rhinoceros application. Data exchange is made possible through the Hops component within the Rhinoceros application and through the open source rhino3dm Python library developed by McNeel (Robert McNeel et al 2022). Geometric operations conventionally computed by the local Rhinoceros application, can be solved through a headless Rhino.Compute component that can be hosted locally or in the cloud. Rhinoceros, ghhops_server, rhino3dm and Rhino.Compute are all developed by McNeel, thereby ensuring interoperability. Initial tests of the components described above were conducted in relation to the voxel model SQL interface and delivered promising results. Algorithms developed for this application will be hosted on the Rhino.Compute platform set up by McNeel (WP3).

The further development depends to some extent on advances in other work packages that will influence how the described terrain modeling approach will develop. For example, the development of ecological models in WP4, such as the local model (D4.1 - 2.2.4) and their integration with the computational infrastructure developed by WP3 and the EIM Ontology (D4.1 - 3) will influence the scope in which the urban form expressed as continuous terrain can be evaluated with the application of ecological models. The components described above are planned in accordance with WP3 recommendations and technologies actively developed at McNeel.

Next steps:

- Further systematization of horizontal and transformative topographic pattern;
- Evaluation of the applicability of Docofossor plugin in the context of horizontal topographic pattern;
- Evaluation of modeling approaches to represent horizontal and transformative topographic patterns in regard to the platform architecture proposed by WP3;
- Evaluation of possibilities to alter terrain form through algorithmic data input related to maps and rasters (WP5 and WP4), and networks (WP5 and WP4);
- Developing the link to the EIM ontology (WP4, D4.1) to establish the way in which the ontology can aid geometric modifications in an algorithmic process based on an adapted version of the geomorphons approach.

3. DATASET MAPS & NETWORKS

The understanding of urban form as a continuous terrain described as an extended set of geomorphons, offers a useful inroad to the analysis of the existing context, as well algorithmic design generation. This identifies one aspect of *what* is being modeled in the first step but needs to be extended and correlated with other datasets. For this purpose, we develop two types of correlated context-specific datasets: (1) *maps* and (2) *networks*. *Maps* describe conditions *prior* to design intervention and conditions *resulting* from design interventions. Examples include for instance a broad range of microclimatic conditions (solar exposure, wind exposure, etc.). For this reason, *maps* contain data that can support an iterative design process



by serving as in-put and out-put for a generative algorithmic design process. *Networks* describe diverse design intentions such as the distribution and relation of building programs, human-nature interactions, ecological relations, food webs, stakeholder provisions, etc. As mentioned above, *networks* can serve as in-put and out-put for a generative algorithmic design process. In the following part we outline a systemic and algorithmic approach to defining these datasets for the first stage of the generative design process.

3.1 Systemic Approach to Maps and Networks

Given the large amount of input data and sources for a multi-species design approach, there is a need for a design method that operates on correlations between data and datasets. The first stage of the development of the algorithmic approach focused on identifying and developing relevant context-specific datasets. This was done both in WP5 research and the linked master-level design studio during the winter semester 2021-22.

3.1.1 Design method development

In the context of the master-level design studio we pursued the design of a kindergarten for children in the age group of three to six years. The site for the project is in the 23rd district of Vienna in Wienerwaldrand-Liesing near the Maurer Wald national forest.

The first set of data that was introduced to the students was a CSV file of the kindergarten's required building program and the related spatial requirements (Fig. 8).

<i>parameters</i>	number of groups	1	3	4
	children/group	25	25	25
	total number of children	25	75	100
	area / child [m2]	7,4	7,4	7,4
<i>basic functional program</i>	minimum space Kindergarten [m2] (basically consisting of:)	185	555	740
	1 - group area: group room, wardrobe, sanitary rooms, storage rooms [m2]	105	315	420
	2 - office and staff area: management office, meeting room, social room, staff wardrobes, staff bathroom, staff restroom [m2]	17,5	52,5	70
	3 - activity area: exercise room, multifunctional area, dining room, kitchen with food storage and refrigerated storage [m2]	27,5	82,5	110
	4 - other space: wheelchair-accessible restroom, buggy storage, laundry room, cleaning supplies storage, general storage room, storage room for garden equipments, garden restrooms [m2]	15	45	60
	5 - circulation space: proportion of traffic area to the sum of usable area and sanitary area is approx. 12-15 % [m2]	20	60	80
<i>utility rooms</i>	minimum space [m2]	35	35	35
	8 - building services: electricity supply, heating, boiler, etc. [m2]	20	20	20
	9 - garbage room [m2]	15	15	15
<i>extended program (optional/expandable)</i>	6 - therapy/relaxing area [m2]	20	20	20
	7- creativity space: atelier, handicraft studio, etc. [m2]	20	20	20

Fig. 9: Kindergarten program - ECOLOPES Studio 2021



The first step consisted of listing and locating the required building programs and areas on site. This task was executed in the Rhinoceros and GH environment with a GH definition that we prepared for the studio. The values were imported from the CSV file (Fig. 9).

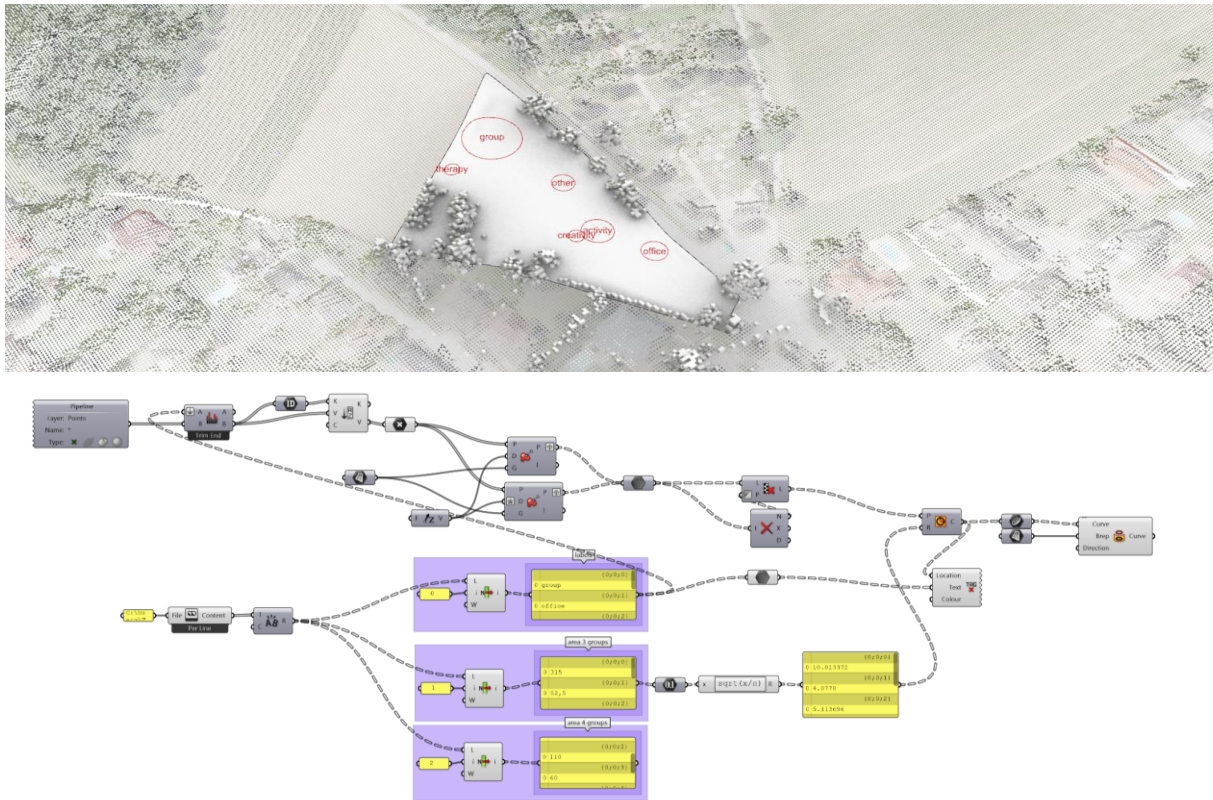


Fig. 10: GH definition for program area distribution on site.

This step served to initiate a program *network*, which denotes the distribution of the required programs of the kindergarten. Generally, networks consist of nodes and edges. The nodes represent data, e.g., functions or locations, while the edges represent the relation between them. The types of relations between building programs (e.g., required, or desired visual or physical connections) were defined in the following step. Connections between nodes were color-coded to denote different types of relations. In subsequent steps further networks were defined, e.g., a stakeholder network that describes desired relations between the human and non-human stakeholders on site.

Combined networks can describe relations between different datasets. Students combined, for instance, building program, stakeholder, and human-nature interaction networks (Fig. 10) that define:

- Distribution of building programs on site;
- Stakeholder distribution on site;
- Types of interactions;
- Benefits of the proposed relations (e.g., ecosystem services);



Network overview

Kindergarten program,
interaction of stakeholders,
skills and senses, 1:500

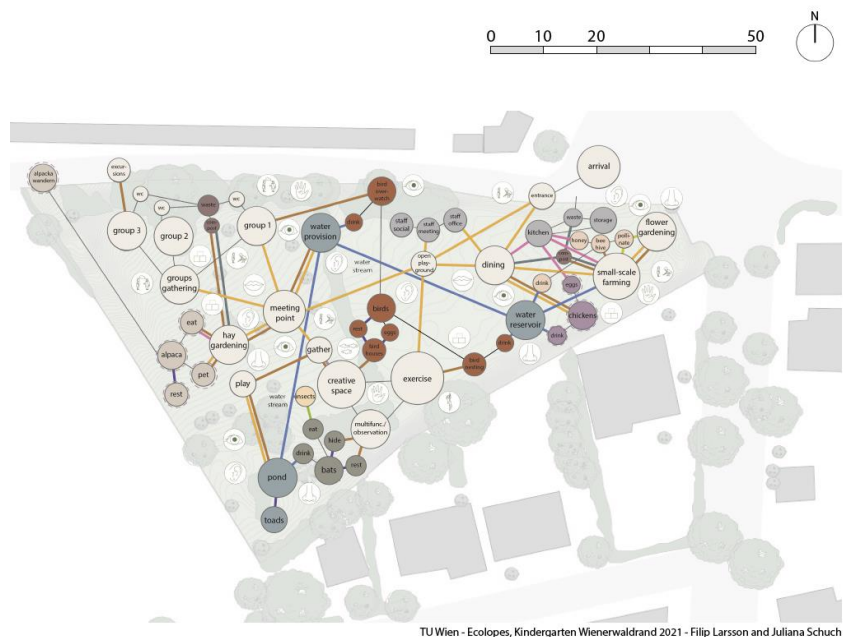
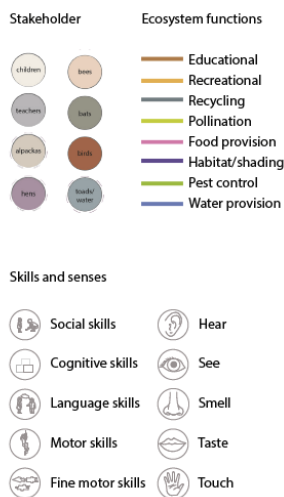


Fig. 11: Example of a combined network. (Students: Juliana Schuch, Filip Larsson)

3.1.2 Maps - Relations between multiple datasets

Site analysis and related simulations resulted in datasets for the design of the kindergarten. This included, for instance, analyses of context-specific microclimatic conditions of the site (*maps*) and context-specific stakeholder analysis (*networks*). The term *map* refers here to a georaster, which is a common concept related to GIS software. Georasters consist of numerical values aligned to a grid (raster) positioned precisely on the earth surface using geographic coordinates. Numerical values aligned on the grid can be represented with a color scale in most common GIS software packages to visualize change of the numeric value (e.g., microclimatic variable) in different locations within the grid. Locations on a map with the same color represent similar conditions regarding a specific indicator (e.g., solar radiation, shading, wind exposure, etc.) and therefore can be classified in zones. The transition between different microclimatic conditions appears often gradually, hence the visualization of the values of the georasters in a fluent color gradient style supports the understanding of changes between zones.

Context-specific maps are a descriptive method for site analysis; however, such maps can also be an input to the design process. For instance, microclimate maps can indicate areas on site that are suitable for specific building programs or intended human-nature interactions. If areas display unsuitable conditions, e.g., due to degree of solar or wind exposure, this entails that either building programs need to be placed somewhere else or, alternatively, that microclimatic conditions have to be changed through interventions that create suitable conditions. This can be achieved for instance through biomass related interventions such as



placing trees or hedges for solar or wind protection. This example shows how different datasets can influence and inform each other. One way to display such a correlation is through layers of maps (Fig. 11) and a relationship matrix (Table 1).

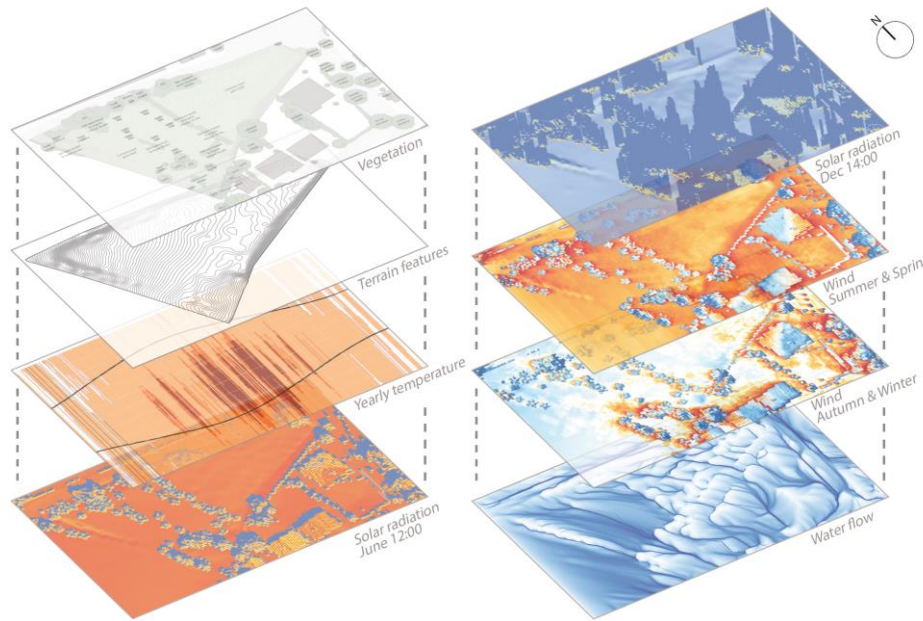


Fig. 12: Example of a series of maps resulting from site analysis that are in part based on using different computational tools. (Students: Juliana Schuch, Filip Larsson)

	Solar exposed	Shaded	Sequential shading	Wind exposed	Wind protected
Small-scale farming area	X		X	X	
Outdoor dining area		X			X
Compost area		X			X
Program elements distribution					
Microclimatic conditions					

Table 1: Example of a relationship matrix that identifies different building program and microclimate relations (Students: Juliana Schuch, Filip Larsson)

The context of the relationship matrix has not been validated, since the focus of this exercise was on the method, *how* to work with different datasets, and not the correctness of the data used to make these relations.



Context-specific networks and relationship matrices thereby serve as design supporting methods to operate on aspects of the design problem definition and solution space that cannot be addressed through common architectural drawings. During the previous semester students developed such networks manually. Currently we are developing methods to create such networks through an algorithmic computational process to advance the workflow. This approach is described in the following section.

3.2 Algorithmic Approach to Maps & Networks

Developing and instrumenting the correlations and interactions between different datasets that inform the algorithmic design process is a challenging task as complexity is added to the design process. Therefore, the integration of the ontology is necessary to manage the complexity. Visualizing these datasets as different layers of correlated *maps* and *networks* in CAD, makes it possible to custom-configure the computational data-driven workflow in relation to a given site and project brief. These datasets therefore constitute a key input for the algorithmic design process. As mentioned before, we aim to utilize *networks* and *maps* as a design support tool, therefore a method is needed to construct these in CAD. Existing approaches, algorithms and tools have been reviewed to develop a method for generating *networks*,

We suggest developing the *networks* based on graph theory, which describes the term “graph” as a geometrical figure which consists of nodes (also called points or vertices) and edges (also called links or lines) (Lauwerier et al. 1971). To construct a graph, different parameters must be defined (Fig. 12). Important parameters are the number of nodes which represent items and the number of edges which describe their connections. Furthermore, the type of edge can also vary as they can be directed or undirected, weighted, or unweighted. The degree of a node describes the number of its connected edges (Chai et al. 2019).

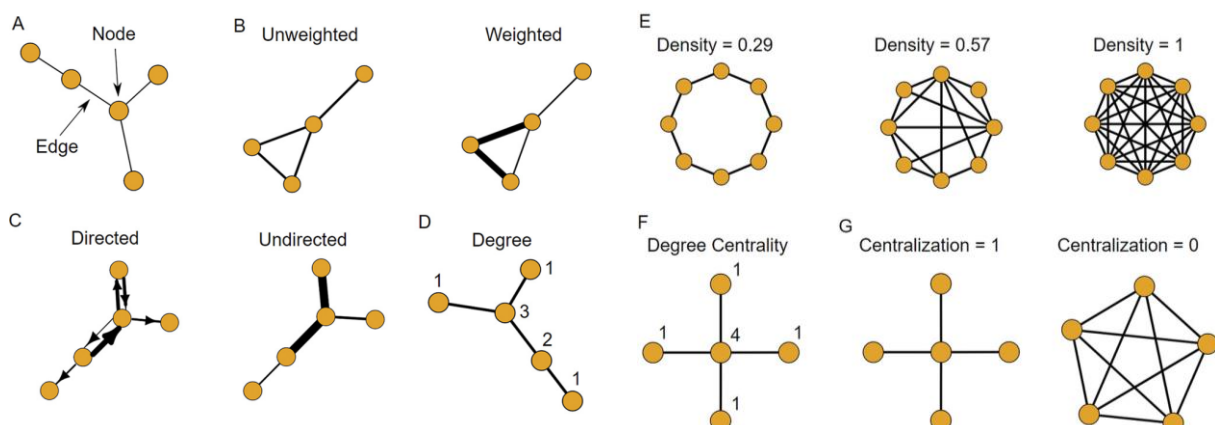


Fig. 13: Graph Theory Parameters (Source: Chai et.al, 2019. Figure 3)



The following sections describe the development and application of graph theory as part of analytical methods in urban design and outline the intended implementation for creating *networks* in the design generation process in WP5.

3.2.1 Analytical methods in urban design - historical review

A broad range of different analytical methods for urban form have been developed in the 1960s. Kevin Lynch developed a mapping method that considers paths, edges, districts, and landmarks as main urban components to analyse a city's structure based on perception of these items (Lynch, 1960). Christopher Alexander analyzed the urban grid through graph representation and graph analysis (Alexander, 1968). In the 1970s criticism was raised against these methods because the complexity of design that cannot be solved with pure rationality was not considered (Karimi 2012). Rittel proposed to differentiate between *tame problems*, such as quadratic equations or chess problems, and *wicked problems* (Rittel 1972). For the latter there exists no definitive formulation as these are unique and their solutions cannot be assigned as *correct* or *false*. Horst Rittel pointed out a shortcoming in applying the first-generation system approach in urban design since "all essential planning problems are wicked" (Rittel 1972, 392). One principle of the systems approach of the second generation deals with *wicked problems* by treating planning as an argumentative process and creating a framework that supports a basis of decision and judgment, rather than selecting between *correct* and *false*. Still, in a planning process design solutions must be evaluated, and this process is not entirely objective because of varying personal preferences. The system approach of the second generation addresses this issue by "objectification" (Rittel 1972, 394). This implies that the information on which a judgment is based is exchanged to make the decision-making process transparent and comprehensible (Rittel 1972). Based on this argument, we suggest implementing analytical methods and tools for design and decision support. Traceable information exchange is especially of high importance within design tasks that involve a variety of disciplines, as is the case in the ECOLOPES project.

Generally, the potential of analytical methods in architectural and urban design lies in the possibility to display complex issues in a tangible manner and in facilitating an iterative approach for computational generation and evaluation of different possible design outcomes.

3.2.2 Analytical methods in urban design - state of the art

There is a wide field of applying graph theory as analytical methods in urban design. Among others, the fields of application are spatial structures, urban infrastructure optimizations, indications of economic wealth, social capital, residential mobility, and public health control (Liang et al. 2021). In the field of spatial analysis weighted graphs are used to identify spatial structures of city hubs and to support the understanding of dynamic urban interactions (Chen



et al. 2014). Graph theory-based methods can also be combined with other analytical methods for example applying pattern recognition and analysis. As an example, there are studies which investigate methods to construct Urban Green Infrastructure (UGI) networks to evaluate connectivity of urban green spaces by combining Morphological Spatial Pattern Analysis (MSPA) with graph theory-based landscape metrics (Liu et al. 2022).

Parametric design methods and GIS have expanded the scope of analytical and generative methods and tools in architectural and urban design. For a comprehensive review of the state-of-the-art for data driven urban planning and analysis tools we refer to *WP3 D1.3: Prototype technical requirements report - section 2*.

Implementing analysis as an input in the early design stage can support a deeper understanding of the design problem definition, design constraints, conditions, and goals. However, specifications need to be set out and met (Karimi 2012) to apply an analytical method that can directly be used in the design process particular. Kayvan Karimi (2012) describes these characteristics as follows: “Any analytical approach that could be used in design has to be a spatial one. (...) The spatial analytical approach should be able to link directly space with people and users. (...) Analytical approaches have to be capable of dealing with different scales. (...) A spatial analytical model should be able to investigate a system as a whole or in its parts” (Karimi 2012, 304).

Among the wide range of analytical approaches which are graph-theory based is Space Syntax. Space Syntax refers to a set of theories regarding the relation between spatial configurations and social patterns. The two core concepts refer to space as (1) indispensable for human activity and (2) “fundamentally a configurational entity” with configurations defined as “simultaneously existing relations” (Karimi 2012, 304). Understanding space through configurations and relations resonates with the above-described approach to networks. For this reason, we started with the search for a suitable algorithmic approach by examining Space Syntax. In this context we investigate whether some principles of Space Syntax can be applied to operating with networks such as, for instance, building program and human-nature interaction networks and their correlations.

3.2.3 Tool Evaluation

In the next step, we selected evaluation criteria (Table 2) for querying tools that are based on graph-theory in the GH environment.



Evaluation criteria	Description
Interoperability/ Compatibility	<ul style="list-style-type: none">● compatibility with Rhino Version 7● possibility of data exchange for inputs and outputs with common data formats for graphs (e.g., GraphML, JSON, RDF)?● possibility of data exchange with GIS network modeling ESRI Shapefile (.shp) (important for connecting to ecological models in WP4)● storage of output (graph) SQL database
Implementation	<ul style="list-style-type: none">● possibility of contribution to network approach, in a manner of expanding the use of graph-theory on other (non-spatial) functional networks (e.g. human-nature interaction)?
User-friendliness	<ul style="list-style-type: none">● availability of user instructions (tutorials, descriptions, sample files, etc.)

Table 2: Evaluation criteria for tool query

3.2.3 Open questions

The following questions emerged regarding the integration of the EIM ontology and the context-specific networks.

- In which way can the algorithmic approach towards context-specific networks potentially be linked to as well as supported and informed by the ontology and what are the limits of its contribution in the algorithmic design process?
- Which context should the networks visualize? Are the relations described in the networks (e.g., ecosystem services) a direct input from the ontology?

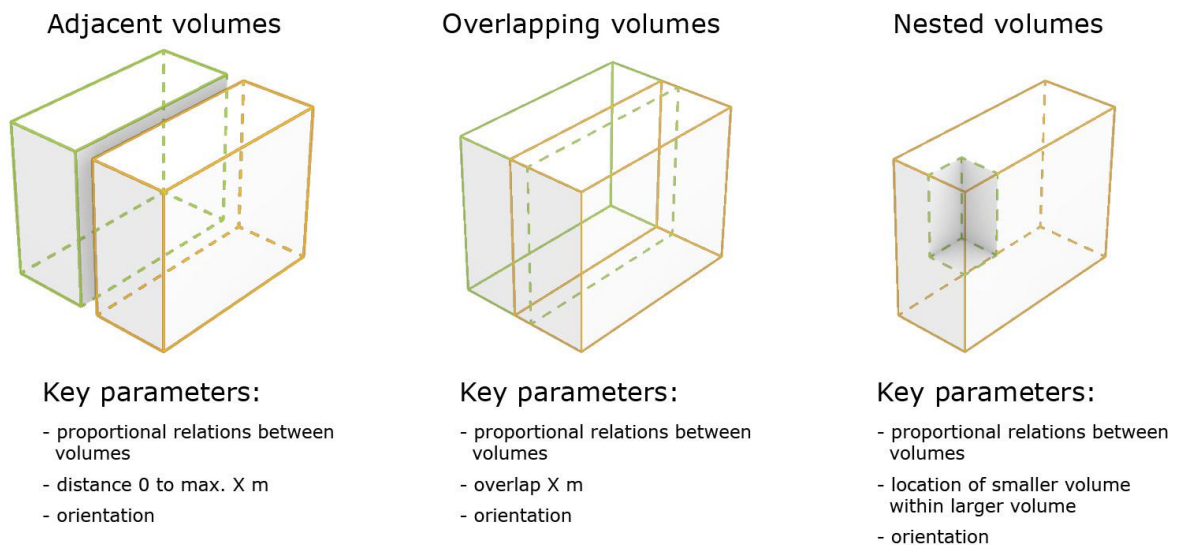
These questions will be discussed and evaluated in the further process. Further next steps related to this topic include:

- Literature review of algorithmic approaches to modeling networks;
- Identification and detailed description of relevant networks for the design process;
- Outline of selected algorithmic approach to modeling networks;
- Query of tools which apply Space Syntax in the GH environment to outline potentials and shortcomings for application in WP5;
- Development of an algorithmic approach to modeling networks (first version);
- Identification of potential connection points and integration with the EIM ontology;



4. DATASET VOLUMES

While the datasets *terrain*, *maps* and *networks* constitute key components as in- and outputs of the algorithmic design process, they are not yet indicative of the spatial extent of the various items including the architectural body, biomass body, and soil body and the way these interrelate. Therefore, the iterative process that defines terrain in relation to conditions (*maps*) and items and their relations (*networks*) is incomplete without the dataset *volumes*. The latter is required to inform and delimit the second stage of the generative algorithmic process in which the detailed design commences.



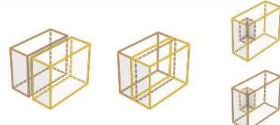
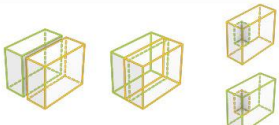
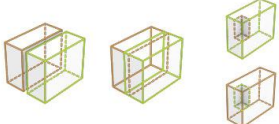
	Architectural volume	Soil volume	Biomass volume
Architectural volume	/		
Soil volume	/	/	
Biomass volume	/	/	/

Fig. 14: The dataset volumes concerns systematic relations between architectural, biomass and soil volumes that can be adjacent, or overlap or be nested. Top: diagrams showing adjacent, overlapping, and nested volumes and related key parameters. Bottom: table showing different combinations of volumes.



The datasets terrain, maps and networks inform the distribution of (1) architectural volume(s) as required by the brief, (2) existing and projected biomass volume(s), and (3) existing and projected soil volume(s). Architectural volume(s) need to comply with local regulations, such as permitted building footprint, floor area ratio, etc. Soil volumes include those that form the projected terrain and those that are added onto or into the architectural volume. Biomass volumes are those that describe the volume of existing and projected plants.

While this dataset describes the spatial extent and relations of architectural, biomass and soil volumes, it does not contain design instructions, which need to be added.

To take this further it is necessary to establish a systematic approach to differentiate between different types of spatial and hierarchical relations of the volumes. Spatial types include exterior, interior, and transitional spaces. Spatial relations include (1) proximity (volumes next to one another at varying distances), (2) overlapping, and (3) nested (e.g., one or several small biomass volumes nested within a larger architectural volume) (Fig. 14). The latter also implies hierarchical relations. Larger volumes establishing the primary level of hierarchy and the smaller nested volumes establish secondary and tertiary levels of hierarchy. Placement and systematic description of the volumes is followed by assignment of essential design parameters for each volume (i.e., biomass volumes require instruction regarding solar exposure, such as sun angles etc., as well as supporting aspects such as water sources and structural requirements, provision of adequate soil bodies, etc.). Once the datasets *terrain*, *maps*, *networks*, and *volumes* are clearly defined detailed design can commence.

4.1 Systemic Approach to Volumes

As stated above we defined three types of spatial volumes: (1) architectural volumes, (2) biomass volumes, and (3) soil volumes. Architectural volumes define the spatial extent of architectural interventions. Biomass volumes define the spatial extent of plant species distribution. Soil volumes define the placement and amount of soil across the site and in and around the architectural volumes. The latter indicates that these different types of volumes can be situated adjacent to one another, they can overlap or be nested within one another. Overlap of volumes indicates areas in which different volumes generate a hybrid condition, while nested conditions indicate a hierarchical ordering of volumes, for instance a biomass volume integrated in the architectural volume.



4.2 Operational & Algorithmic Approach to Volumes

Current efforts involve clarification of operational aspects before identifying or configuring a related algorithmic approach. The necessary steps are outlined below.

Next steps:

- Alignment of generic “form” of architectural volumes with building regulations;
- (System for describing existing building(s) generalized as architectural volume(s))
- System for describing projected architectural volume(s);
- System for describing existing cluster(s) of plants generalized as biomass volume(s);
- System for describing projected biomass volume(s);
- System for describing and distinguishing between different existing soil volume(s);
- System for describing and distinguishing between projected soil volume(s);
- Criteria for governing proportional relations between the three types of volumes;

5. GENERATIVE DESIGN

5.1 Systemic Approach to Generative Design

Receiving different design alternatives and variations as an output of the form generation process is important when dealing with complex architectural problem definitions, because there might exist more than one suitable solution. Moreover, single-solution approaches pose the risk of overlooking suitable design solutions in the early design stage. It is therefore beneficial to generate a range of outcomes and to narrow these down through a filtering and ranking process. In a generative design process, the design is guided through the setting of rules and algorithms in an iterative operation. Therefore, generative design is suitable for creating design variations by changing parameters and reinforcing algorithms through a feedback loop (Agkathidis, 2016; Bohnacker et al., 2009).

5.2 Algorithmic Approach to Generative Design

It is important to establish a fundamental comprehensive understanding on the overarching theme to inform the development of an algorithmic approach in a systematic way, it is important to first establish a fundamental comprehensive understanding on the overarching theme. For this reason, we are in the process of undertaking a broad literature review on the application of “metaheuristics” in architectural design. This is done with the aim to gain a detailed understanding of the variety of algorithms that can generate a range of outcomes and possible solutions based on specified input parameters within a reasonable amount of time. The aim is not to search every possible solution or combination, but to generate a range of feasible solutions that can be evaluated and ranked. It can be expected that through this



process several design outcomes can be identified that fulfill the evaluation criteria as defined objectives and can be moved on to the next design stage (Gandomi et al. 2013).

The understanding of the algorithmic processes in WP5 and WP6 is synchronized, as both WPs utilize “metaheuristic algorithms”. The different applications include (1) the generation of initial design variations (WP5), and (2) the optimization of design solutions (WP6). For a detailed description of strategies and approaches in WP6 we refer to *D1.3 Report of Year 1 - Work Package 6*.

5.2.1 Definition of metaheuristics

Gandomi et al. (2013) explained that there exists “no agreed upon definition of heuristics and metaheuristics in the literature. Some researchers use ‘heuristics’ and ‘metaheuristics’ interchangeably. However, the recent trend tends to name all stochastic algorithms with randomization and global explorations metaheuristic” (Gandomi et al. 2013, 1.) Sörensen et al. (2018) proposed the following definition: “A metaheuristic is a high-level problem-independent algorithmic framework that provides a set of guidelines or strategies to develop heuristic optimization algorithms. The term is also used to refer to a problem-specific implementation of a heuristic optimization algorithm according to the guidelines expressed in such a framework” (Sörensen et al. 2018, 4). Makas et al explained that “metaheuristics can be defined as iterative methods that mimic the exploitation and exploration behaviors of some agents. Most of them are neighborhood search methods that are inspired by nature. They constitute a large and important class among improvement algorithms. A metaheuristic explores the problem space globally and searches in the neighborhoods of the existing solutions locally to get new and better solutions. There should be a fine balance between local intensive exploitation and global exploration” (Makas et al. 2016, 4936; Yan et al. 2007).

5.2.2 Metaheuristic Algorithms

There exists a wide range of algorithms that can be categorized as “metaheuristic algorithms” (Sörensen et al. 2018, 4). Figure 13 shows the categorization in population-based algorithms, which start with an initial population (finite set of elements) and compute a new population at each iteration, and trajectory-based algorithms, which evolve towards a single solution (Dreo et al. 2007). In general, metaheuristic algorithms that apply randomization are suitable for generating a variety of solutions (Gandomi et al. 2013). Since the aim in WP5 is to produce a solution space with initial design variations, we exclude in our selection trajectory-based algorithms that only output single solutions. Furthermore, we exclude metaheuristic algorithms created for optimization purposes, known as “optimization algorithms”, as the aim is to generate design variations and not to optimize solutions as is the case in WP6.

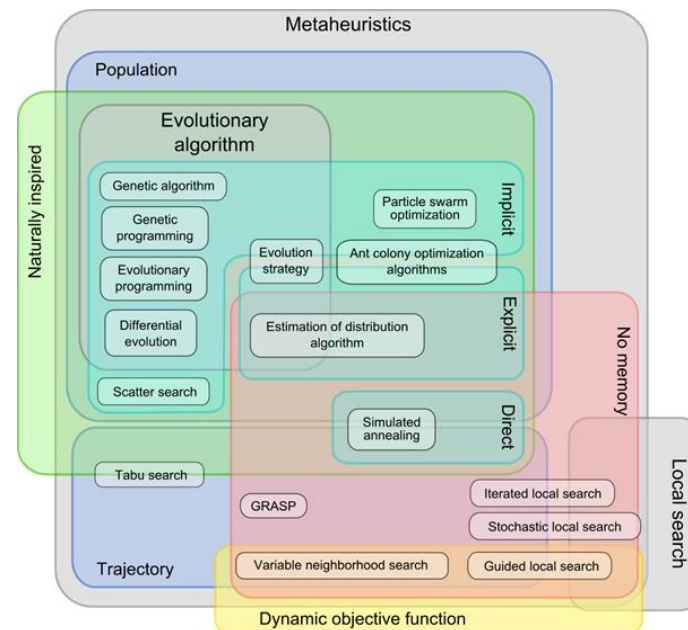


Fig. 15: Classification of metaheuristics (Source: Beheshti et al. 2013, Figure 1)

5.2.3 Evolutionary Algorithms

We are in the process of undertaking a detailed review of different types of generative design approaches based on evolutionary algorithms (EA) that can serve as a template for the intended algorithmic process. EAs offer an alternative to standard problem solving in which solutions are evolved instead of directly solved (Encyclopedia of Physical Science and Technology 3rd Edition 2003). EAs generate a variety of design solutions that fulfill different objectives. Conflicting objectives usually occur, and their fulfillment is affected by decisions that are taken during an early design phase. Inspired by biological processes, EAs generate variety through random modifications of individual solutions, also referred to as “mutations” (Dreo et al., 2007). In conclusion, EAs are a likely candidate for the algorithmic processes in WP5. In the next steps it needs to be specified which type(s) of EA will form the basis for the initial design generation.

5.2.4 Next steps

In the next steps, the literature review of generative design approaches will be finalized to establish a basis for decision on selecting and possibly adapting a suitable algorithmic approach. Furthermore, the ranking and filtering of the outputs requires the definition of scenarios which describe conditions and objectives to fulfill them. A focus for the next project phase will also be the elaboration of the interfaces between algorithmic processes in WP5 and the EIM ontology developed in WP4, the optimization processes developed in WP6 as well as



the implementation and integration of the algorithmic processes in the computational framework which is developed in WP3 and provided by McNeel.

6. ALGORITHMIC APPROACH IN RELATION TO THE COMPUTATIONAL WORKFLOW

6.1 Algorithmic Approach & Voxel Model

A voxel model is introduced in the ECOLOPES project to establish an interface between the EIM Ontology (WP4) and the algorithmic design processes described in the preceding paragraph. Voxel model can incorporate expert information and integrate spatial data in different resolutions. Shchurova (2015) described a voxel as a "three-dimensional space region limited by given sizes, which has its own nodal point coordinates in an accepted coordinate system, its own form, its own state parameter that indicates its belonging to some modeled object and has properties of modeled region" (Shchurova 2015). Based on this definition we can define voxel models as collections of data-points assigned to a three-dimensional grid with precisely defined resolution. In the context of the ECOLOPES project, the voxel model will provide an interface between different datasets that can incorporate expert information. We are currently undertaking a detailed review of the use of voxel models in a broad range of disciplines. The development of voxels is closely linked to the invention of MRI scanners in the 1970s. Early examples of voxels were linked with pioneering efforts to digitize the MRI imagery (Drebin, Carpenter, and Hanrahan 1988) and to establish methods to generate discrete representations of three-dimensionally scanned objects (Kaufman and Shimony 1987).

6.1.1 Outline of the algorithmic approach to the voxel models

Applications of voxel models range from volumetric particle simulations (e.g., OpenVDB) to frameworks for data integration, augmentation and automated ML-based analysis (e.g. NiBabel, MedPy). Efforts aimed at the integration of different geospatial datasets describing objects that are considerably different in scale are sparse. However, such approaches are gradually emerging (Lamprecht 2019) due to the increasing availability of digital mapping technologies. Voxel models have been applied in different disciplinary contexts, for example in the context of soil science both in geomorphometry (Hofierka, Mitášová, and Neteler 2009) and in the scale of soil architecture, or via the application of voxel automata to simulate root growth processes in relation to soil conditions (Mulia, Dupraz, and Noordwijk 2010). In geospatial science, a similar concept has been applied to simulate different geo-spatial processes (Jjumba and Dragičević 2016). Voxel-based modeling can be combined with LiDAR based datasets to study relations between land-use, three-dimensional vegetation structure

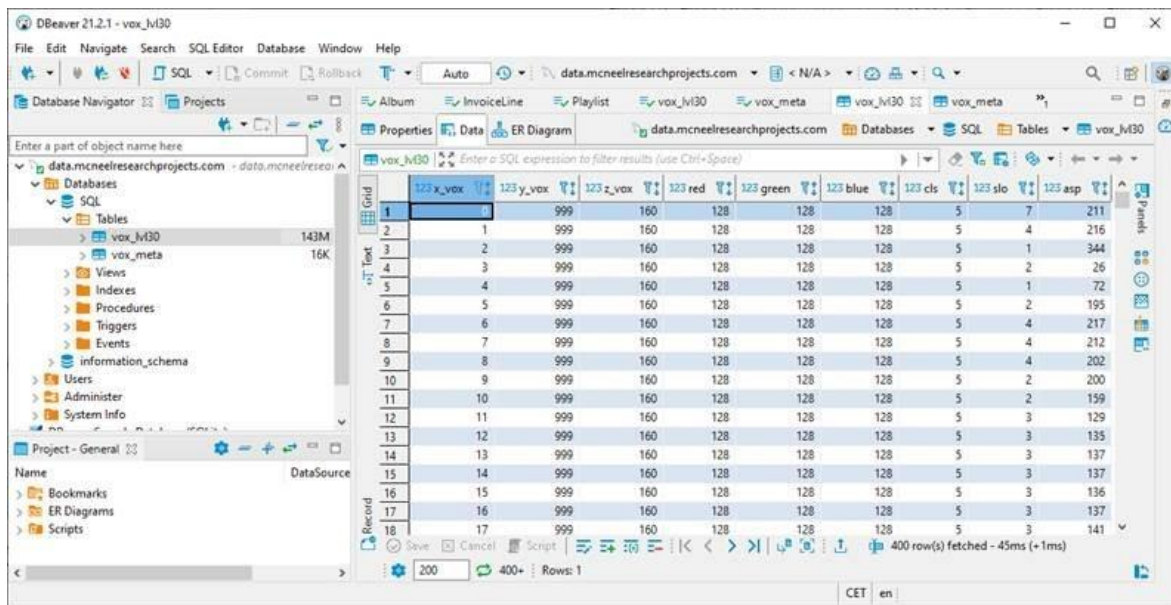


and vegetation density. Several domain-specific uses of voxel models align well with the focus of the ECOLOPES project.

The possible application of SQL databases for storing voxelized data was evaluated in anticipation of requirements related to the EIM Ontology. The relation between EIM Ontology and the voxel model are further elaborated in the deliverable *D4.1 Preliminary EIM Ontology* in the paragraphs *3.1 General description of the EIM Ontology (task 4.7)* and *3.2 Representation of domain knowledge (task 4.7)*. In the context of SQL databases, the storage of point-based geospatial data in a database has been studied (Martinez-Rubi et al. 2015) that primarily serves compression or real-time data visualization. A further elaboration of the voxel model is available in the *D3.1: Prototype technical requirements* report in the section *3.5.1 Architectural design components* and in the context of a related experiment in the section *4.3 MiMo inputs*, in particular the text describing conversion from 3D CAD to raster data and geometry parameters.

6.1.2 Initial implementation of the algorithmic approach to the voxel models

The following describes how the overall approach described in the previous section is implemented utilizing the technologies proposed by WP3 as described in the deliverable *D3.1: Prototype technical requirements* report. We currently pursue a straightforward implementation of a voxel model aligned with the structure of SQL database, to simplify the development of interfaces with relevant tools and methods from different disciplines. At the same time, SQL-based voxel implementation was selected to ensure interoperability with different programming languages (R, Python, Java) and hence easy access for all ECOLOPES project members. McNeel (WP3) currently hosts an SQL server (data.mcneelresearchprojects.com, *D3.1: Prototype technical requirements report - 7.2.1 Data storage*) that can be accessed by the R-scripts of WP4 or viewed with the free HeidiSQL database viewer (Fig. 14). For security reasons, access credentials are available upon request from WP3 and WP5. According to the specifications defined by McNeel, this SQL server is a part of cloud based ECOLOPES infrastructure, where different components such as WP4 ecological models, EIM Ontology, WP5 algorithmic tools, WP6 simulation and optimisation tools can access project related resources.



Grid	123 x_vox	123 y_vox	123 z_vox	123 red	123 green	123 blue	123 cls	123 slo	123 asp
1	999	160	128	128	128	5	7	211	
2	1	999	160	128	128	5	4	216	
3	2	999	160	128	128	5	1	344	
4	3	999	160	128	128	5	2	26	
5	4	999	160	128	128	5	1	72	
6	5	999	160	128	128	5	2	195	
7	6	999	160	128	128	5	4	217	
8	7	999	160	128	128	5	4	212	
9	8	999	160	128	128	5	4	202	
10	9	999	160	128	128	5	2	200	
11	10	999	160	128	128	5	2	159	
12	11	999	160	128	128	5	3	129	
13	12	999	160	128	128	5	3	135	
14	13	999	160	128	128	5	3	137	
15	14	999	160	128	128	5	3	137	
16	15	999	160	128	128	5	3	136	
17	16	999	160	128	128	5	3	137	
18	17	999	160	128	128	5	3	141	

Fig. 16: Tabular view of the SQL data saved on the McNeel server. Voxel dataset can be viewed with such a simplified interface, using free software such as HeidiSQL or DBeaver.

An interface between Rhinoceros and the SQL voxel model has been implemented by using the McNeel Rhino Hops component (see section 2.3 and *D3.1: Prototype technical requirements report - 2.4.3 The Grasshopper Hops component*). The interface gives Rhinoceros users the ability to visualize and interact with datasets available in the SQL voxel model through the Rhinoceros interface (Fig. 15, Fig. 16). Rhinoceros users can interact with the SQL voxel data hosted online by WP3 (McNeel) utilizing the Rhinoceros Hops component. The Hops component is used to integrate cloud-hosted SQL voxel data with cloud-hosted Rhinoceros interface written in Python. Python based Hops applications can be run from cloud-based web server infrastructure, such as Heroku or Amazon Web Services (for more detail see Baer, Davidson, and Payne 2021). WP3 (McNeel) outlined the overall structure of the Ecolopes platform and suggested designing and implementing individual components to be deployed on the cloud-based Heroku platform (“Cloud Application Platform | Heroku” 2022). The purpose of the Heroku platform is described in *D3.1: Prototype technical requirements report - 7.2.3 The algorithm production server*. For Python and Hops based components this implies a container-based development approach. Tools implementing different functionalities need to be packaged into Docker containers (“Empowering App Development for Developers | Docker” 2022) that can be deployed to the Heroku platform. This approach aligns well with the cloud-based SQL voxel model. A first prototype of Python based Heroku application has been created and tested locally with WP3 (McNeel). The next step will focus on the adaptation of the current Python based Hops component running on a local PC to the requirements of the cloud-based Heroku platform.



Fig. 17: 3D View of the SQL based voxel model (from McNeel Rhinoceros)). Voxel nodal coordinates are aligned on a 1m x 1m grid, which are aligned with the analysis grid / raster grid of the WP4 ecological models. It is a fully 3D model, where for a pair of x and y coordinates multiple z coordinates are available. Highest or lowest points can be queried directly from the SQL server, without a need to postprocess the stack of 3D points in e.g., the R-based WP4 models.

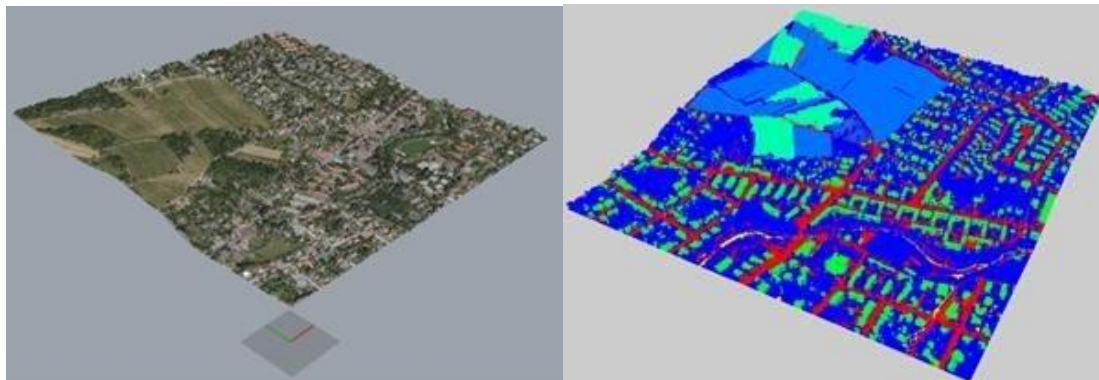


Fig. 18: Full extent of the 1m² patch around the first Vienna site. Smaller extracts can be efficiently queried from any programming language, using the SQL syntax. R-tree based index is constructed to speed up the query on the x and y dimensions. Other parameters, such as slope (steepness) and aspect (direction of the slope) and classification (for classification codes see Excel file in the WP5 Folder) are encoded in the SQL voxel model for now. More data will be added by the project stakeholders according to their needs.

6.2 Next steps towards linking the algorithmic process with the voxel model

The primary role of the voxel model is to expose spatial data generated by WP4 components, such as the ecological models and EIM Ontology to the WP5 algorithmic design processes in different scales. In the scale related to the ecological models, the voxel model related to preliminary results generated by WP4, as presented in Figure 17 below. Initial integration of a



voxel model in the scale related to architectural objects was studied by the participants of the MiMo Experiment and reported in *D3.1: Prototype technical requirements report - 4. The knowledge generation framework and the MiMo experiment.*

The further development of this component is dependent on the intended interactions with both ecological and architectural models. Biological models and data created by WP4 will be integrated with the common grid system of the SQL voxel model. Technically this requires the ecological models to use a common (raster) grid with matching resolution and extent. WP4 has aligned their models to 1m grid, and the ecological models will need to switch from abstract data inputs to the inputs related to the cities chosen for the ECOLOPES project.

Regarding the Heroku platform (*D3.1: Prototype technical requirements report - 7.2.3 The algorithm production server*), managing access and billing of computational resources, in relation to multiple users using the computational infrastructure simultaneously, needs to be addressed. While this may not be of primary importance in the early development stages, it can become relevant in case the developed components need to be used e.g., by students in the master-level design studios that need to prepare deliverables for the weekly meetings.

Currently the SQL voxel model contains an area of 1 km² around the first site in Vienna in 1m resolution. The interoperability between programming tools selected by WP4 (R programming language) has been tested with positive outcomes, (Fig. 17). Further discussion is needed to identify the connections between the datasets created by different WP4 datasets and models, such as the input datasets and local building features (*D4.1: Preliminary EIM Ontology - 2.1.1 Georeferenced datasets and local building features*) and the local model (*D4.1: Preliminary EIM Ontology - 2.2.4 The local model*) in relation to the overall ECOLOPES design process. The role of individual components, such as the EIM Ontology (*D4.1: Preliminary EIM Ontology - 3.1 General description of the EIM Ontology (task 4.7)*), simulation tools and the KPIs to be evaluated in the design process need to be specified to advance the development.

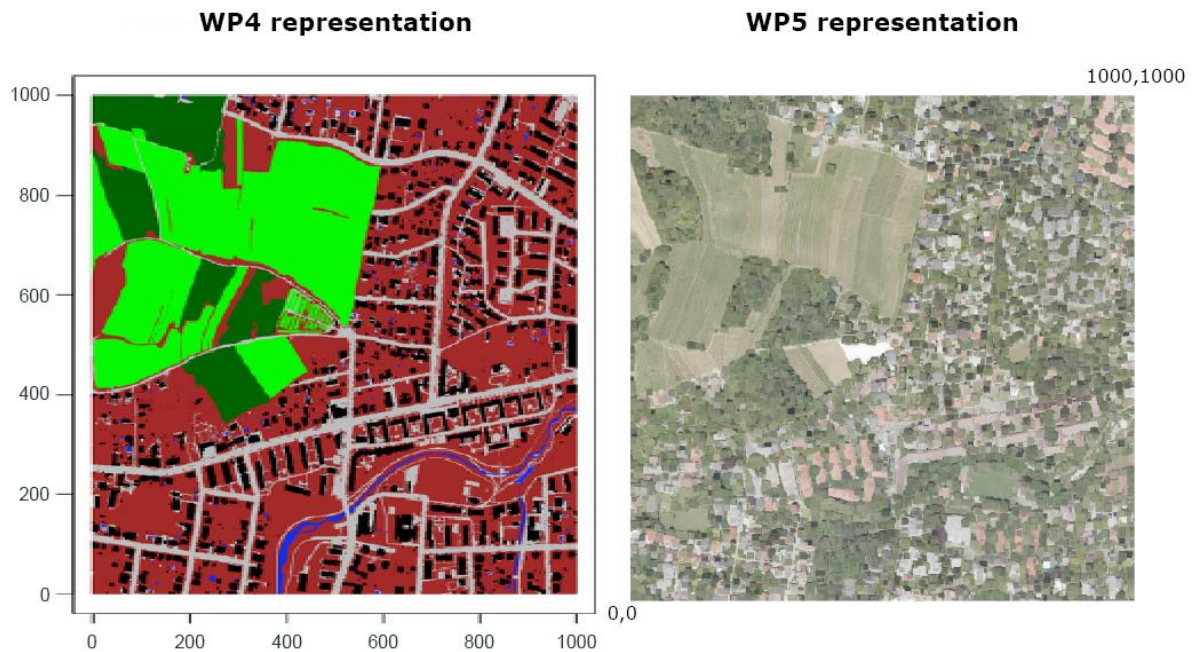


Fig. 19: Different representations of the voxel data written in the SQL database. Left: Illustration generated by WP4 based on the ecological model prototype implemented in the R programming language, which utilizes classification data and 2.5D representation. Right: Illustration generated by WP5 from Rhinoceros utilizing the developed Hops component to query 3D point coordinates and colors.

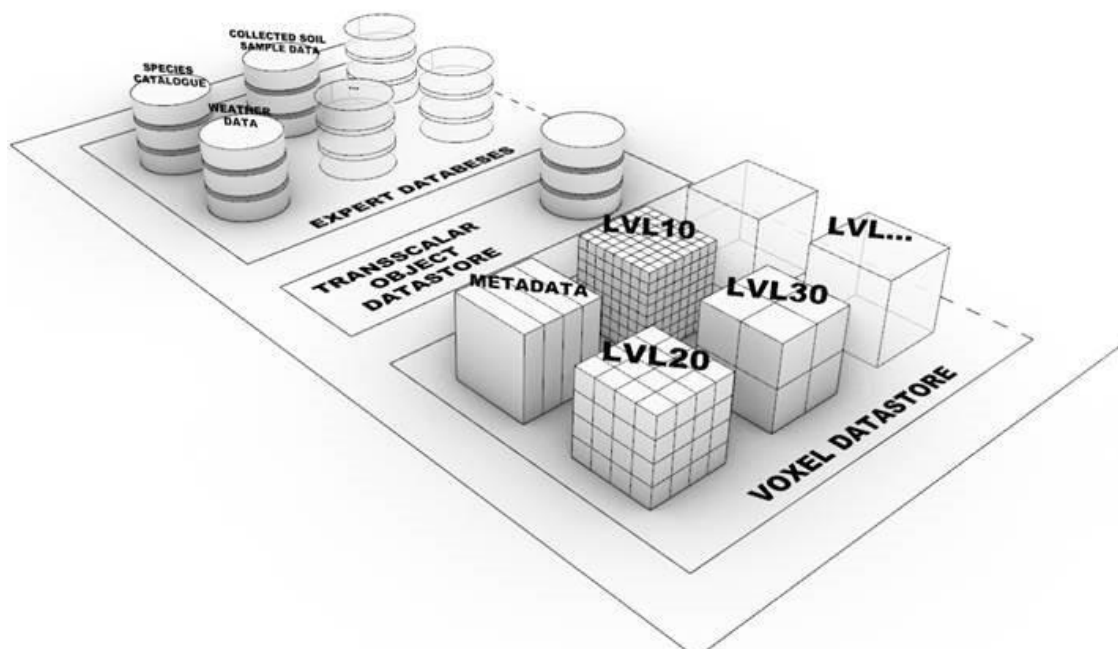


Fig. 20: Conceptual structure of the SQL data store set up by the WP3 and WP5. Further, linked resolutions, referred to as "levels" will be added as data in other resolutions (such as urban habitat classification, building block scale) will be available. Expert databases containing data which do not have spatial representation (weather data, species catalog, plant traits) can also be stored in the SQL database to simplify the integration of different ECOLOPES components.



The integration of different scales will be introduced in the next steps of the development (Fig. 18). Urban habitat classification experiment currently developed by WP4 is utilizing large-scale datasets with international coverage. Urban habitat classification experiment utilizes 2D gridded datasets in multiple resolutions and addresses different locations related to the cities where the consortium participants are located. The outcomes of the urban classification experiment are internally coordinated within WP4 with the components related to ecological modeling and will be available to the other consortium participants at later stages. The integration of the datasets and computational design methods described in this report will be addressed in the next steps. Inputs and outputs of the urban classification and ecological modeling are relevant for multi-scalar data integration. Different scales are already included in the urban classification and ecological modeling data, so initial efforts related to data integration and interoperability with the Rhinoceros and GH interface are possible at the point when data is shared within the consortium.

Next steps:

- Completion of literature review on voxel model use in different disciplines;
- Identification and development of detailed voxel model approach based on the literature review and further analysis;
- Evaluation of the applicability of the voxel based components in the algorithmic design process.

6.3 Algorithmic Approach & EIM Ontology

6.3.1 Dataset *Terrain* & EIM Ontology

As discussed in sections 1.1 and 2 we approach urban form, and architectures as part of urban form, as a continuous yet differentiated terrain. Based on this approach, we identified a systematic and algorithmic approach to defining instances of landform, based on an adapted version of geomorphons. This is aligned with a corresponding approach in the development of the EIM Ontology (WP4) that seeks to inform design solutions that can initiate a novel ecosystem that promotes multifunctionality, biodiversity and human well-being, and that can be used to support, direct, and accelerate the development of a heterogeneous habitat in a timeframe related to construction and habitation of architectures. For this purpose, a computational design workflow with decision support capabilities that can be linked with the algorithmic design process is needed.

The ontology will correlate different sets of information including: (1) urban form (landscape architecture), (2) biodiversity distribution, (3) service delivery and (4) spatial program of an *ecolope* (D4.1). A voxel-based representation will serve to connect these sets of information spatially and is essential for feeding context-specific spatial data into the ontology as input for informing and instructing the ECOLOPES algorithmic design process.



For the EIM ontology an adapted geomorphon-based approach is instrumental for information to be modeled spatially explicitly. Current efforts focus on co-developing the geomorphons-based approach for both the EIM Ontology and the ECOLOPES algorithmic design process.

6.3.2 Dataset Networks & EIM Ontology

Furthermore, focus is placed on coordinating and co-developing the approach to the dataset *networks* with the aim to link the EIM Ontology with the ECOLOPES algorithmic design process. An ontology can be expressed on an instrumental level as a collection of graphs and for this reason ontologies can be written to a database using a graph data store (storage driver). The EIM Ontology information can be represented as volumetrically related datasets that can be mapped onto a database. Features included in the ontology can be visualized as a graph (Fig. 20). Guiding the design process in relation to ecosystem development objectives for a given location depends on relevant information that can be obtained from literature, simulation-based analysis, geospatial and experimental data and simulation-derived data, and the suitability of that information for the ECOLOPES approach. Relevant datasets are filtered and structured in an SQL database for populating the ontology instances. This context-specific information is integrated within the knowledge graph to generate query results that can be used to assist the ECOLOPES algorithmic generative design process.

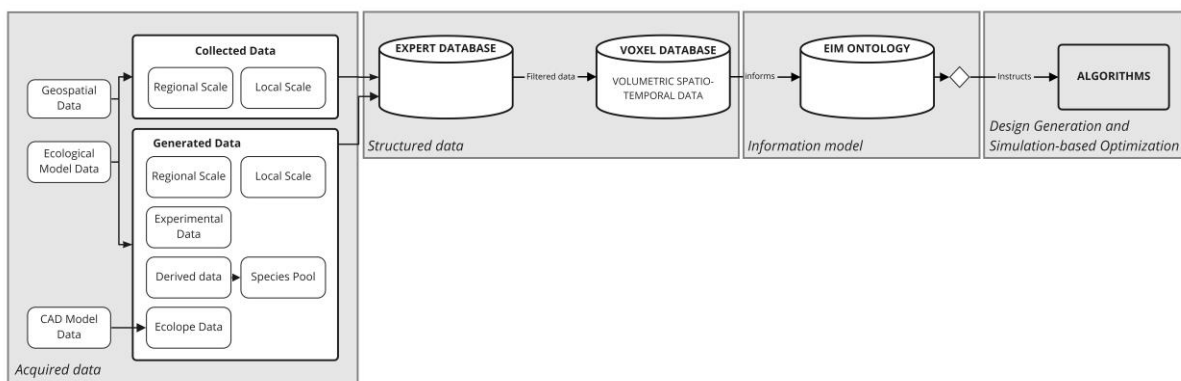


Fig. 21: Partial workflow showing how the EIM Ontology can relate to other components to facilitate information processing and design.

Designers configuring *networks* for the purpose of design require expert input regarding the validity and potential impact of proposed relations expressed as networks (e.g., network of building programs and network of stakeholder interactions). One way to enable interrogation of the EIM ontology by the designer is to provide corresponding networks in the EIM ontology that provide permissible relations between nodes within a network or between different networks, together with specific related co-benefits or trade-offs. This can be made possible by limiting the number of general designer-definable networks to a selected range that will likely frequently occur in design projects.



REFERENCES

- Agkathidis, Asterios. *Generative design*. London: Hachette UK, 2016.
- Alexander, Christopher. *Notes on the Synthesis of Form*. Cambridge, MA: Harvard University Press, 1968.
- Beheshti, Zahra & Siti Mariyam Shamsuddin. "A review of population-based meta-heuristic algorithms". *International Journal of Advances in Soft Computing and Its Applications* 5, no.1(2013): 1-35.
- Bohnacker, Hartmut, Benedikt Groß, Julia Laub and Claudius Lazzaroni. *Generative Gestaltung. Designing, Programming, Visualization with processing*. Mainz: Verlag Herman Schmidt, 2009.
- Chai, Albert, Joshua P. Le, Andrew S. Lee, and Stanley M. Lo. "Applying Graph Theory to Examine the Dynamics of Student Discussions in Small-Group Learning." *CBE—Life Sciences Education* 18, no. 2 (2019): 1-12. <https://doi.org/10.1187/cbe.18-11-0222>
- Chen, Wai-Kai. "Basic Theory". In *Applied Graph Theory: Graphs and Electrical Networks - North Holland Series in Applied Mathematics and Mechanics Vol. 13*. Amsterdam: North Holland Publishing Company, 1971.
- Cowan, Robert. *The dictionary of urbanism*. Tisbury: Streetwise Press, 2016.
- Creswell, John W. *Research Design, Qualitative, Quantitative, and Mixed Methods Approaches* 4th Ed. Thousand Oaks CA: Sage, 2014.
- Drebin, Robert A., Loren Carpenter and Pat Hanrahan. "Volume Rendering." SIGGRAPH '88 Proceedings of the 15th annual conference on Computer graphics and interactive techniques. (Aug 1988) 65-74. <https://doi.org/10.1145/54852.378484>
- Dreo, Johann and Patrick Siarry. "Stochastic Metaheuristics as Sampling Techniques using Swarm Intelligence." In *Swarm Intelligence*, edited by Felix T.S. Chan and Manoj Kumar Tiwari. London: IntechOpen, 2007.
- Ekici, Berk, Cemre Cubukcuoglu, Michela Turrin, I. Sevil Sariyildiz. "Performative computational architecture using swarm and evolutionary optimisation: A review" *Building and Environment* 147 (2019): 356–371. <https://doi.org/10.1016/j.buildenv.2018.10.023>
- Evans, Ian S. (2012) "Geomorphometry and landform mapping: What is landform?". *Geomorphology* 137, no.1 (2019): 94-106. <https://doi.org/10.1016/j.geomorph.2010.09.029>
- Franklin, Steven E. (1987) "Terrain Analysis from Digital Patterns in Geomorphometry and Landsat MSS Spectral Response." *Photogrammetric Engineering and Remote Sensing* 53, no.1 (1987): 59-65.



Frayling, Christopher. "Research in Art and Design". *Royal College of Art Research Papers* 1, no.1 (1994): 1-5.

Goriainov, Vitali, Richard Cook, Jeremy M. Latham, Douglas G. Dunlop, and Richard O. C. Oreffo. "Bone and Metal: An Orthopedic Perspective on Osseointegration of Metals." *Acta Biomaterialia* 10, no. 10 (2014): 4043–57. <https://doi.org/10.1016/j.actbio.2014.06.004>.

Hensel, Michael. *Performance-oriented Architectures - Rethinking Architectural Design and the Built Environment*. Chichester: Wiley, 2013.

Hofierka, J., Mitášová, H., & Neteler, M. (2009) "Chapter 17 Geomorphometry in GRASS GIS." In *Developments in Soil Science Vol. 33 Geomorphometry - Concepts, Software, Applications* edited by Tomislav Hengl and Hannes I. Reuter, 387–410. Amsterdam: Elsevier, 2019. [https://doi.org/10.1016/S0166-2481\(08\)00017-2](https://doi.org/10.1016/S0166-2481(08)00017-2).

Hurkxkens, Ilmar, and Mathias Bernhard. "Computational Terrain Modeling with Distance Functions for Large Scale Landscape Design." *Journal of Digital Landscape Architecture* 4 (2019): 222–30. <https://doi.org/10.14627/537663024>.

Jasiewicz, Jarloslaw, and Tomasz F. Stepinski (2017) 2022. r.geomorphon - GRASS GIS Repository (version 8.0.0). C. Open Source Geospatial Foundation. https://github.com/OSGeo/grass/tree/releasebranch_7_8/raster/r.geomorphon.

Jasiewicz, Jarloslaw, and Tomasz F. Stepinski. "Geomorphons - a pattern recognition approach to classification and mapping of landforms". *Geomorphology* 182 (2013): 147-156. <https://doi.org/10.1016/j.geomorph.2012.11.005>

Jjumba, Anthony and Suzana Dragičević. "Towards a Voxel-Based Geographic Automata for the Simulation of Geospatial Processes." *ISPRS Journal of Photogrammetry and Remote Sensing* 117 (July 2016): 206–16. <https://doi.org/10.1016/j.isprsjprs.2016.01.017>.

Karimi, Kayvan. "A Configurational Approach to Analytical Urban Design: 'Space Syntax' Methodology." *Urban Design International* 17 (2012): 297-318. <https://doi.org/10.1057/udi.2012.19>

Kaufman, Arie, and Eyal Shimony. "3D Scan-Conversion Algorithms for Voxel-Based Graphics." *I3D '86: Proceedings of the 1986 workshop on Interactive 3D graphics*, (1987). 45-75. <https://doi.org/10.1145/319120.319126>.



Latour, Bruno. *Down to Earth - Politics in the New Climate Regime*. Cambridge: Polity Press, 2017.

Lenholzer, Sandra, Ingrid Duchart, and Adri van den Brink. (2017) "The relationship between research and design". In *Research in Landscape Architecture – Methods and Methodology*, edited by Adri van den Brink, Diedrich Brinks, Hilde Tobi and Simon Bell, 36-41. London: Routledge.

Liang, Xiaofan, and Yuhao Kang. "A Review of Spatial Network Insights and Methods in the Context of Planning: Applications, Challenges, and Opportunities". In *Urban Informatics and Future Cities*, edited by Geertman, Stan S.C.M., Christopher Pettit, Robert Goodspeed and Aija Staffans, 71-91. Springer, Cham, 2021.

Liu, Yang, Ting Ting Huang and Xi Zheng. "A method of linking functional and structural connectivity analysis in urban green infrastructure network construction". *Urban Ecosystems* (2022). <https://doi.org/10.1007/s11252-022-01201-2>

Lobeck, Armin K. *Geomorphology - An Introduction to the Study of Landscapes*. New York: McGraw-Hill, 1939.

Lynch, Kevin. *The Image of the City*. Cambridge, MA: MIT Press, 2016.

Makas, Hasan, and Nejat Yumusak (2016). Balancing exploration and exploitation by using sequential execution cooperation between artificial bee colony and migrating birds optimization algorithms. *Turkish Journal of Electrical Engineering & Computer Sciences*, 24, no. 6 (2016): 4935-4956.

Mark, David M. "Geomorphometric Parameters: A Review and Evaluation". *Geografiska Annaler: Series A, Physical Geography* 57, no. 3-4 (1975) : 165-177. <https://www.tandfonline.com/doi/abs/10.1080/04353676.1975.11879913>

Martinez-Rubi, Oscar, Peter van Oosterom, Romulo Gonçalves, Theo Tijssen, Milena Ivanova, Martin L. Kersten, and Foteini Alvanaki. "Benchmarking and Improving Point Cloud Data Management in MonetDB." *SIGSPATIAL Special* 6, no. 2 (2015): 11–18. <https://doi.org/10.1145/2744700.2744702>.

Mulia, Rachmat, Christian Dupraz, and Meine van Noordwijk (2010) "Reconciling Root Plasticity and Architectural Ground Rules in Tree Root Growth Models with Voxel Automata." *Plant and Soil* 337 (2010): 77-92. <https://doi.org/10.1007/s11104-010-0502-3>.



Olaya, V. (2009) "Chapter 6 Basic Land-Surface Parameters." *Developments in Soil Science Vol. 33*, edited by Tomislav Hengl and Hannes I. Reuter, 141-169. Amsterdam: Elsevier, 2009.

Oliveira, Vítor. *Urban Morphology - An Introduction to the Study of the Physical Form of Cities*. Cham: Springer, 2016.

Oliveira, Vítor. *Urban Morphology - An Introduction to the Study of the Physical Form of Cities*. Cham: Springer, 2022.

Petrasova, Anna, Brendan Harmon, Vaclav Petras, Payam Tabrizian, and Helena Mitasova. *Tangible Modelling with Open Source GIS*. 2nd Ed. 2018 ed. Cham: Springer, 2018.

Pike R.J., Evans I.S. & Hengl T. (2009) "Chapter 1 Geomorphometry: A Brief Guide". *Developments in Soil Science Vol. 33*, edited by Tomislav Hengl and Hannes I. Reuter, 3-30. Amsterdam: Elsevier, 2009.

Rittel, Horst. "Der Planungsprozess als iterativer Vorgang von Varietätserzeugung und Varietätsbeschränkung", edited by Jürgen Joedicke, 17-32. In *Arbeitsberichte zur Planungsmethodik 4: Entwurfsmethoden in der Bauplanung*. Stuttgart: Karl Krämer Verlag, 1970.

Rittel, Horst. "On the Planning Crisis: Systems Analysis of the 'First and Second Generations'". In: *Bedriftsøkonomen* 8 (1972): 390-396.

Shchurova, Catherine Igorevna. "A Methodology to Design a 3D Graphic Editor for Micro-Modeling of Fiber-Reinforced Composite Parts." *Advances in Engineering Software* 90 (December 2015): 76–82. <https://doi.org/10.1016/j.advengsoft.2015.07.001>

Smith, Barry, and David M. Mark. "Do Mountains exist? Towards an Ontology of Landforms". *Environment and Planning B: Planning and Design* 30, no. 3 (2003): 411-427. <https://doi.org/10.1068/b12821>

Sörensen, Kenneth, Marc Sevaux, and Fred Glover. "A History of Metaheuristics". In *Handbook of Heuristics*, edited by Rafael Martí, Pardolos Panos, and Mauricio G.C. Resende, 1-18. Springer, Cham, 2018.

Sörensen, Kenneth, Fred Glover. "Metaheuristics". In: *Encyclopedia of Operations Research and Management Science*, edited by Saul I. Gass, and Michael C. Fu. Boston MA: Springer, 2013.

Yan, Tai-Shan, Yong-Qing Tao YQ, and Du-Wu Cui. "Research on handwritten numeral recognition method based on improved genetic algorithm and neural network".



2007 International Conference on Wavelet Analysis and Pattern Recognition (2007): 1271-1276. <https://doi.org/10.1109/ICWAPR.2007.4421630>

Zhong, Chen, Stefan Müller Arisona, Xianfeng Huang, Michael Batty and Gerhard Schmitt (2014) “Detecting the dynamics of urban structure through spatial network analysis”. *International Journal of Geographical Information Science*, 28, no. 11(2014): 2178-2199. <https://doi.org/10.1080/13658816.2014.914521>

SOURCE CODE CITATION

Robert McNeel & Associates. 2021. Ghhops_server: Grasshopper Hops Server (version 1.4.1). <https://github.com/mcneel/compute.rhino3d/tree/master/src/ghhops-server-py>.

Robert McNeel & Associates. 2022. Rhino3dm: Python Library Based on OpenNURBS with a RhinoCommon Style (version 7.14.1).

WEB PAGES

Baer, Steve, Scott Davidson, and Andy Payne. 2021. “The Hops Component.” December 15, 2021. <https://developer.rhino3d.com/guides/compute/hops-component/>.

Heroku. “Cloud Application Platform | Heroku.” Accessed March 2, 2022. <https://www.heroku.com/>.

Docker. “Empowering App Development for Developers | Docker”. Accessed March 2, 2022. <https://www.docker.com/>

W3C Semantic Web. “Web Ontology Language (OWL)”. Last modified December 11, 2013. Accessed March 8, 2022. <https://www.w3.org/OWL/>

OECD Organization for Economic Co-operation and Development. “OECD Glossary of Statistical Terms”. Last modified February 1, 2006. Accessed March 8, 2022. <https://stats.oecd.org/glossary/detail.asp?ID=542>



APPENDIX

Here we include the studio brief for the summer semester 2022 master-level Ecolopes studio at the Department of Digital Architecture and Planning at Vienna University of Technology. The studio is taught by Prof. Dr. Michael U. Hensel, Tina Selami and Jakub Tyc, and Asst. Prof. Dr. Milica Vujovic, whose field of expertise is *evidence-based design support*.

The studio is part of six consecutive studios that are linked with the Horizon 2020 Future and Emerging Technologies project “ECOLOPES - ECOlogical building envelOPES: a game-changing design approach for regenerative urban ecosystems” (www.ecolopes.eu).

2022 ECOLOPES - Kindergarten Wienerwaldrand 2

Teaching staff:

Prof. Dr. Michael Hensel, Asst. Prof. Dr. Milica Vujovic, Tina Selami & Jakub Tyc

“To design *in* an environment is to design *an* environment.”

Busbea, L. (2019) Foreword: Maldonado’s Environment. In Maldonado T. (2019 [1972]) Design, Nature & Revolution - Toward a Critical Ecology. University of Minnesota Press, Minneapolis, pp. vii-xiii.

Urban areas and urban development are frequently characterized by growing human-nature disconnect, reduction or loss of ecosystems and biodiversity, and negative impact on human health and well-being. We ask therefore whether architectural design can promote and provide intensive human-nature interaction? And is it possible to develop a multi-species design approach?

What kind of human-nature interactions are relevant in a specific design case and what can be their purpose? Ecosystem services (ES) can be a starting point for a design exploration. ES include provisioning (i.e. providing food, materials, etc.), regulating (i.e. purification of water and air, climate regulation, waste decomposition, etc.), supporting (i.e. habitat provision, soil formation, etc.), and cultural services (i.e. recreation, science and education, etc.). However, on a more fundamental level nature can inspire curiosity and exploration and interaction and hence be educational and support child development in various ways. A kindergarten is an interesting context for exploring these questions and for providing nature-based experiences to small children that are not normally part of their urban life.

The studio focuses on the design of a kindergarten for 3 or 4 groups in the age group of three- to six-year-old children. The aim is to explore a synthesis of architecture and landscape, and more specifically the building envelope as a multi-species space. To approach this, we view both city and site as continuous terrain, thereby removing notional boundaries while taking care of soil and water regimes, vegetation, animals, and the intended human-nature interactions and experiences for the children.

The site for the project is in the 23rd district of Vienna in the suburban area Wienerwaldrand-Liesing near the Maurer Wald national forest. The area is well documented and there is data



available on the species and habitat program of Wienerwald-Liesing as part of the Natura 2000 scheme.

A. The role of *evidence & data*

At this stage, students will be introduced to the principles of evidence-based design, that aims to encourage students' scientific thinking and its application during the design process within Ecolopes studio. After introducing themselves to the topic and drafting initial ideas, students will start with the tutorials on the role of evidence and data that ties directly to creation of dataset maps and networks (one of the further stages of studio methodology). Understanding of scientific principles and previous knowledge, useful for the Ecolopes studio will be obtained through the study of existing solutions, principles from practice and knowledge from the literature in the domains of human-nature interaction, kindergarten design, multi-species space, fusion between building and landscape etc.

Students will be provided with guidance on how to answer following questions:

- Which sources are needed to inform design decision making within specific design topics?
- How to relate scientific evidence with physical aspects of space?
- How to create the algorithm of decisions that support design?

Together with the conventional architectural drawings such as plans, sections and perspectives, the algorithm holds an important role in the delivery of the design intent. The algorithmic logic should inform digital models, architectural drawings, and visual representation of *ecolopes* spaces. Although, the biggest challenge is to establish a correlation between an operative algorithm and purposeful architectural strategy, tutorials on evidence-based design will aim to contribute to addressing this challenge. All work related to the evidence-based design will be tied into the student's studio project.

Lectures:

1. Lecture 1(week 4) - Basic principles - Introduction to evidence-based design and its relation to the topic of the Studio [Presentation, Handout]
 - a. Students' exercise 1: Identify sources needed to obtain evidence that would support design decisions in the individual/team projects
 - b. Delivery: 1-page pdf with a diagram containing sources and their selected output information relevant for the design
2. Lecture 2(week 7) - Input-output flowchart: Guide to the meaning of logical operators, their application in architecture and their relation to evidence-based design. Introduction to the algorithms and their use in architectural design. [Presentation, Handout]
 - a. Students' exercise 2: Creating the algorithm that supports architectural design
 - b. Deliveries: 1-page pdf with an algorithm that presents the flow of information related to design decisions; n-page schematic/diagrammatic architectural drawings



DELIVERIES

Exercise 1 (week 6):

* related to Lecture 1

- Design intent supported by evidence (1-page pdf): formulate a connection between collected evidence and design intent. Students should provide a document that combines textual (200-300 words) and diagrammatic presentation (half-page) of selected outputs from resources (research papers, collected data about the site etc.) and aspects of architectural design on which those outputs would influence. Example: findings about children's cognitive development in urban vs rural areas should inform the decision on the amount of specific plant species on the site. The initial content created at this stage can be updated anytime during the semester. (example could be presented during presentation).
- (Connected to Exercise 2 - Networks)

Exercise 2 (week 10 & 12):

* Related to Lecture 2

- (Week 10) Algorithmic design framework (1-page pdf): draft a flowchart to demonstrate the stream of information, showing a design input, logic of decision-making and the (initial) design output. Students are supposed to create a diagrammatic presentation of an algorithm using IF/THEN logical operators. Example 1: IF the average height of a 3-year-old child is 100 cm THEN position specific content or opening at that height. Example 2: IF a specific species of a plant requires lots of sun, THEN consider where to position built objects in order not create shades throughout a year. The initial content created at this stage can be updated anytime during the semester.
- (Connected to Exercise 3 - Networks)
- (Week 12) Schematic architectural drawings: demonstrate the implementation of an algorithm into the design. Students are free to choose the visual language in which they will communicate how an algorithm is translated into the architectural design process.

B. Dataset *Terrain*

Stage 1: Site analysis

Workshop 1: Introduction to SAGA GIS

Introduction to the open-source GIS tools that will be used for large scale analysis in the first weeks and later in the design phases to analyse environmental performance of students' designs. In the following workshops links between GIS based analysis and Rhinoceros + Grasshopper will be presented; this workshop is required to understand those connections.

Tools presented in this workshop would include SAGA GIS (version 7.6.3, not 8.0.1) and QGIS 3, students are required to install this free, open-source software on their personal computers



before, to be able to complete the assignments. Software download links are added at the end of this document.

The workshop covers the following topics:

- Basic terrain analysis - slope, aspect, generating contours (QGIS)
- Basic terrain geomorphology analysis - geomorphons (SAGA GIS)
- Environmental simulations- incoming solar radiation and wind exposure (SAGA GIS)
- Basic hydrologic analysis - water flow accumulation (SAGA GIS)

Basic analysis related to the use of weather data in architectural design will be required to complete the group assignment. Those methods are very well documented in the official software video tutorials:

1. Rhino LadybugTools | How to download and import weather data
| How to draw yearly, monthly and daily plots of e.g. temperature
<https://docs.ladybug.tools/climate-analysis/weather-data>
(Two first video tutorials on this webpage)
2. Rhino Ladybug | How to generate a wind rose diagram for chosen month (Video tutorial): <https://docs.ladybug.tools/climate-analysis/wind-rose>

(Group 3 and 4 will start with the analysis after the workshop).

C. Datasets *Maps & Networks*

Topic C deals with the representation of input data as *maps* and *networks*. Students will be introduced to a design method that enables combining different types of data and understanding of correlations in the site analysis and design process.

Exercise1: Site analysis

In this stage students will work in groups to focus on different aspects of site analysis.

Group 1: Green network

The existing vegetation on the site and the surrounding areas will be analyzed and displayed as a green network map (in district and site scale). Gaps and possibilities for expansions of the network will be explored.

- distinguish different types of land use and related ecological aspects in a certain radius (e.g. 1km) (input for Group 2)
- Borders and connections
- Maps, diagrams, table and text based analysis: complementary and contradictory land use
- Site visit: field study (Plant Identification App: <https://identify.plantnet.org/>)



Group 2: Stakeholder

A matrix of relevant stakeholder on site and in the surrounding (human and non-human eg. plant, animal and insect species) will be created, which will be an important input for the next stages.

- Use land use determination (from Group 1) to define the characteristic species (biodiversity)
- Look for borders for specific animals
- Diagrams, charts
- Site visit: field study (Plant Identification App: <https://identify.plantnet.org/>)

Sources for Analysis:

Environment map of Vienna: <https://www.wien.gv.at/umweltgut/public/>

Land Use map of Vienna: <https://www.wien.gv.at/flaechenwidmung/public/>

Translation plug in:

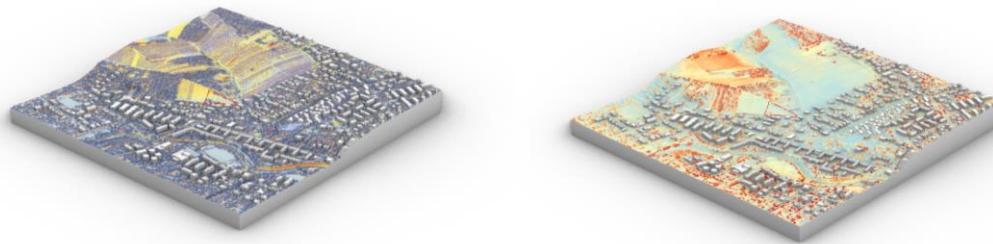
- [Google Translate for Google Chrome](#)
- [Google Translate for Mozilla](#)
- Safari has automatic option for translate
- [Mate Translate for Opera](#)

Result: combined chart of Group 1 and 2

Group 3: Microclimate (based on QGIS and SAGA GIS analysis and online research)

The analysis of the site will include microclimatic conditions related to (e.g., solar radiation, temperature, wind, etc.) in different seasons. Digital analysis tools will be used for large scale solar radiation and wind exposure analysis. Computational analysis will be supplemented with online research to understand seasonal changes related to weather and microclimate in Vienna.

- Research and describe air temperature dynamics in Vienna in a typical meteorological year
- Research and describe wind directions and speeds in Vienna in a typical meteorological year
- Incoming solar radiation analysis in SAGA GIS for characteristic days throughout the year (1 km²)
- Wind Effect analysis in SAGA GIS for characteristic days throughout the year (1 km²)
- Combine results of computational analysis and online research by describing identified, typical microclimatic conditions for different seasons in Vienna. (e.g. winters in Vienna are windy, with winds typically from [direction] reaching [speed] m/s and average temperature of...)
- Prepare graphics and text describing combined results described above. Sum up the work with a few points which can be used as evidence for your future design. Based on the example related to the wind from the point above: On cold Viennese winter days, strong winds are coming from ... direction -> we can design wind protecting elements on site which limit wind impact in winter but not impact the airflow in summer.



Left: Incoming solar radiation analysis in SAGA GIS. Right: Wind Effect analysis in SAGA GIS

Resources:

Rhino LadybugTools WeatherData: <https://docs.ladybug.tools/climate-analysis/weather-data>
(Video tutorials: how to download weather data, how to generate daily and monthly temperature statistics)

Rhino Ladybug WindRose (Video tutorial): <https://docs.ladybug.tools/climate-analysis/wind-rose>

SAGA Incoming Solar Radiation https://saga-gis.sourceforge.io/saga_tool_doc/7.1.1/ta_lighting_2.html

SAGA Wind Exposition Index https://saga-gis.sourceforge.io/saga_tool_doc/6.1.0/ta_morphometry_27.html

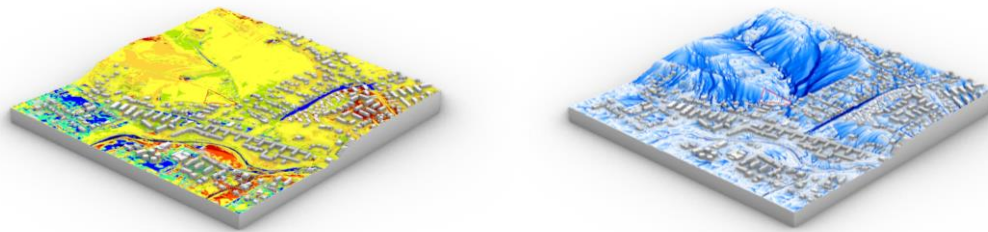
MeteoBlue: https://www.meteoblue.com/en/climate-change/vienna_austria_2761369

WeatherSpark: <https://weatherspark.com/>

Group 4: Terrain & Soil (based on QGIS and SAGA GIS analysis and online research)

Terrain and soil conditions of the site and the surrounding areas will be analyzed. Characteristics of the terrain will be explored with computational analysis tools related to the geomorphometry and hydrological properties of this location (e.g., water flow accumulation).

- Research and describe soil types, their physical and water related properties close to our location
- Research and describe soil types and their properties related to vegetation close to our location
- Basic terrain (slope, aspect, contour lines) and geomorphometric (geomorphons) analysis in QGIS and SAGA GIS (1 km²)
- Basic hydrologic analysis (water flow accumulation, Topographic Wetness) in SAGA GIS (1 km²)
- Combine results of computational analysis and online research by describing identified, typical soil conditions in relation to vegetation and surface water flow found close to our location.
- Prepare graphics and text describing combined results described above. Sum up the work with a few points which can be used as evidence for your future design.



Left: Geomorphons analysis in SAGA GIS. Right: Water flow accumulation analysis in SAGA GIS

Resources:

SAGA Geomorphons: https://saga-gis.sourceforge.io/saga_tool_doc/7.8.2/ta_lighting_8.html

SAGA Fill sinks: https://saga-gis.sourceforge.io/saga_tool_doc/7.8.2/ta_preprocessor_4.html

SAGA Topographic Wetness: https://saga-gis.sourceforge.io/saga_tool_doc/7.8.2/ta_hydrology_15.html

SAGA Flow Accumulation: https://saga-gis.sourceforge.io/saga_tool_doc/7.8.2/ta_hydrology_29.html

Bodenkarte: <https://www.wien.gv.at/umweltschutz/raum/bodentypen.html> and <https://bodenkarte.at/>

<https://www.wien.gv.at/umweltgut/> -> Gewässer und Boden -> Boden -> Bodenkarte, ...

The outcomes will lead to a series of maps and matrices that describe various conditions before the design.

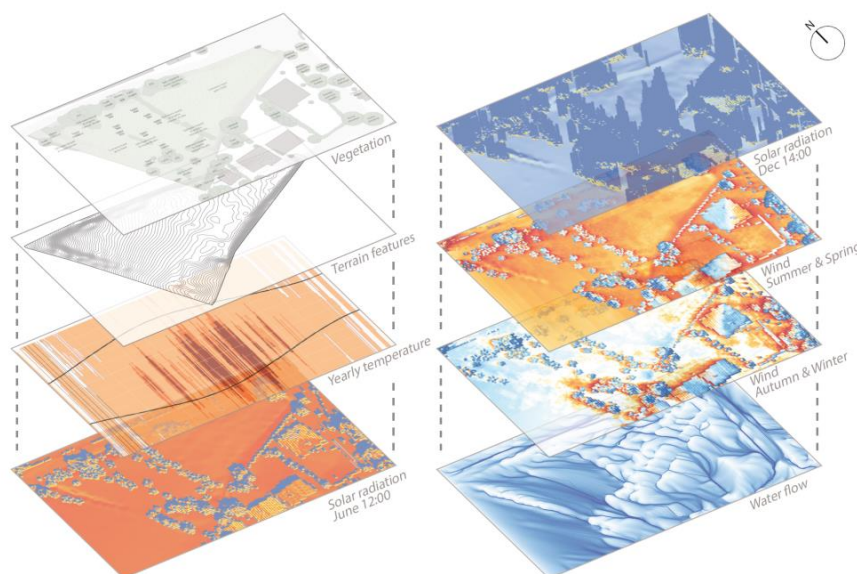


Fig. 1: Example of a series of maps resulting from site analysis that are in part based on using different computational tools.

Exercise 2: Network layers (relates to C)

Maps and *networks* can also be configured to describe design intentions and can serve as an input to the design. At this stage layers of networks will be created in Rhino/GH.



The layers consist of:

- network of building programmes on site
- network of ecosystem functions & services
- network of intended human-nature interactions

It is also possible to add network layers that are not listed above and that are relevant for the design approach. Each network layer will first be created separately. At this stage the correlations between elements of a network will be described by a correlation matrix.

	A	B	C	D	E	F	G	H	I
1		0_Entrance	1_Group1	2_Group2	3_Group3	4_Office	5_Farming	6_Dining	
2	0_Entrance								
3	1_Group1	X							
4	2_Group2	X							
5	3_Group3	X							
6	4_Office	X	X	X	X				
7	5_Farming					X			
8	6_Dining					X	X		
9									
10									
11									
12									

Fig. 2: example of a correlation matrix for the functional program

This information will be written in a JSON file which feeds into a given GH definition using the [SpaceChase](#) plug-in to generate a graph:

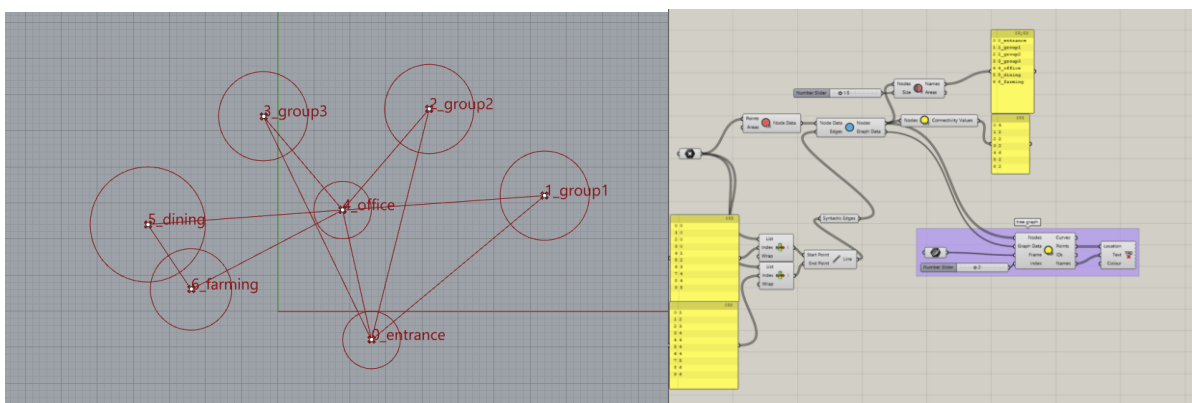


Fig. 3: example of a function network and related GH definition

Exercise 3: Network correlation (relates to topic C)

At this stage, students will focus on the correlation between different datasets. To understand the correlations, at least two networks will first be overlaid and analyzed. If possible, a third network will be included in the next step. In addition the previously generated gradient maps will inform the design process.

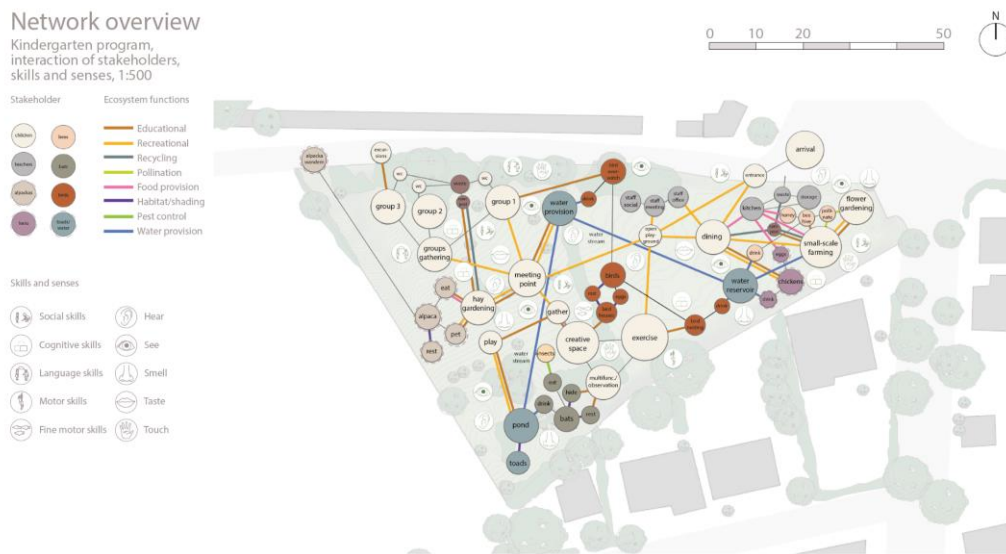


Fig. 4: Example of a combined network of intended program distribution and human-nature interactions laid out on a site plan.

D. Dataset *Volumes*

The datasets “maps and networks” inform the articulation of the terrain, which includes the architecture. More specifically, the maps and networks and the derived terrain articulation inform the distribution of volumes including (1) the architectural volume(s) as required by the brief, (2) the existing and projected biomass volume(s), and (3) the existing and projected soil volume(s). Architectural volume(s) need to comply with local regulations, like permitted building footprint, floor area ratio, etc. However, in the case of the Kindergarten site such regulations do not exist. Soil volumes are those that form the projected terrain and those that are added onto or into the architectural volume. The biomass volume describes the volume of existing and projected plants. These volumes describe where architectural, vegetation and soil interventions will take place, yet do not contain any specific detailed design or design instructions at this stage.

To take this further it is necessary to distinguish between different spatial relations and types and hierarchical relations of the volumes. For spatial relations this can include for instance (1) proximity, i.e., volumes next to one another at varying distances, and (2) overlapping volumes. Spatial types can include exterior, transitional, and interior spaces. Hierarchical relations can include nested volumes, e.g., one or several small biomass volumes nested within a larger architectural volume (see for instance [Bosco Verticale](#)). After placement and systematic description of the volumes follows the assignment of essential design parameters for each volume (i.e. biomass volumes require instruction regarding solar exposure, such as sun angles etc., and supporting aspects such as water sources and structural requirements, i.e., provision of adequate soil bodies). Once this is accomplished the detailed design can commence.

Deliveries: (systemic / algorithmic)

- Location of volume types (architecture, biomass, soil);
- Definition of spatial relations (proximity, overlap, nested, etc.) and spatial types (exterior, transitional, interior);



- Specification of hierarchical relations between volumes;
- Correlation of volumes with requirements as part of a computational process maps & networks (for instance micro-climatic requirements for architectural programs as defined by the brief, or micro-climatic or soil-related for plant species);
- Definition of key design parameters for each volume;

SITE



Fig. 3: Aerial photo of the site from [Google Earth](#)

The site is located in Liesing at the Wienerwaldrand south of the Kalksburger cemetery and at the limit of the area where building is permitted.

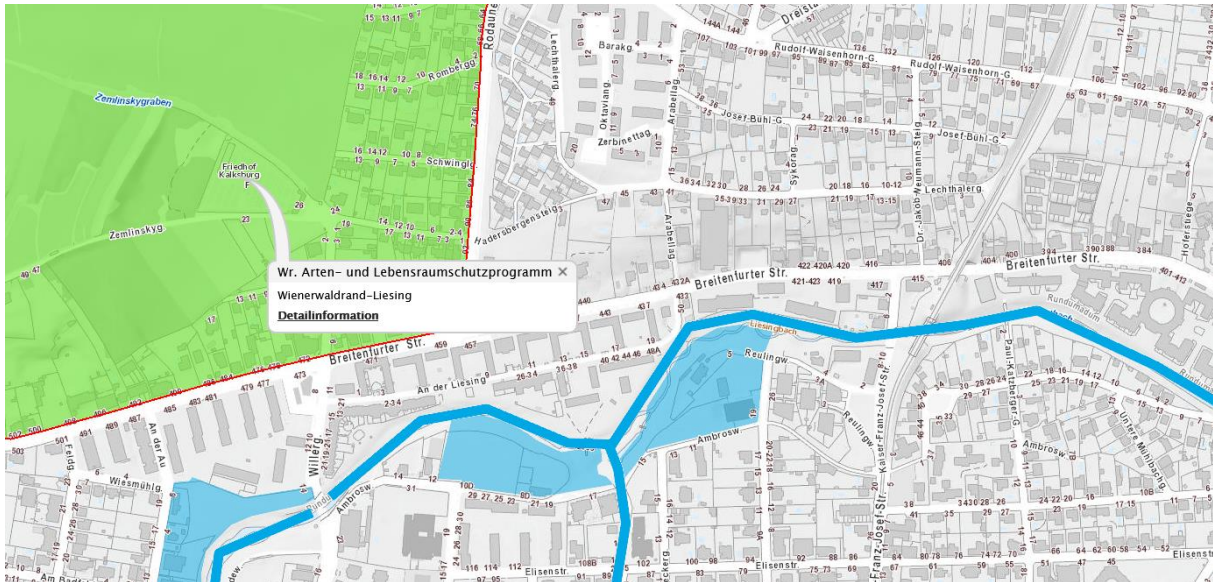
Currently there are [Alpacas](#) on part of the site that can be included into the Kindergarten concept. These are part of [Alpaka Wandern Wien](#). This would provide the possibility of integrating Animal Assisted Therapy with Alpakas "AATLA" <https://www.aatla.de/> However, integrating the Alpacas in the design is voluntary.

Vienna's species and habitat protection program lists two specific areas in the vicinity of the site for the Kindergarten:

1. Wienerwaldrand-Liesing
2. Liesing Creek (Fließgewässer Liesing)



Fig. 4: 3D data of the site in the form of a point cloud is available for the Rhinoceros environment.



Wienerwaldrand-Liesing:

https://www.wien.gv.at/umweltgut/public/identifyNetzwerk.aspx?id=UMWELT.NENA_F.49322&mid=41ee7e8b-79c3-415d-8829-a39dfa6e250b&ftype=vienna:UMWELT.NENA_F&g=3fdaac19-f330-4766-a343-072fa39aadf5&cid=2bae446d-ad77-49f8-9ce0-2d8a5fe3cea



Liesing Creek (Fließgewässer Liesing):
https://www.wien.gv.at/umweltgut/public/identifyNetzwerk.aspx?id=UMWELT.NENA_L.24649&mid=41ee7e8b-79c3-415d-8829-a39dfa6e250b&ftype=vienna:UMWELT.NENA_L&g=6ebff8-779a-41d6-b6cf-712284c492a8&cid=3a40ded3-a5d0-491f-b1db-4a858399313b

The general aim for the Wienerwaldrand area is to keep open and optimise the small land use mosaic from a nature preservation perspective.



Data is available through the portals of MA41 and ViennaGIS. Information on animal species observed on site and plants present in our location can be extracted from those datasources.

KINDERGARTEN

Opening Hours (whole day):



The opening hours of municipal kindergartens in Vienna are Monday through Friday (workdays) from 6:30 am to 5:30 pm all year round. If necessary, the management of the kindergarten can extend the hours to 6:00 am to 6:00 pm.

There are two attendance models for children:

- Full-day attendance: 6:30 am to 5:30 pm (or extended hours as described before)
- Half-day attendance: 6:30 am to 2:00 pm or 12:00 am to 5:30 pm

Closing days:

- The municipal kindergartens in Vienna are closed on public holidays and on December 24th and 31st.

The opening hours and days specify during which time human-nature-interface is taking place on the site.

further information:

Kindergarten AGB

<https://www.wien.gv.at/bildung/kindergarten/pdf/agb.pdf>

Spaces & Requirements

functional program (minimum space requirements):

	number of groups	1	3	4
<i>parameters</i>	children/group	25	25	25
	total number of children	25	75	100
	area / child [m2]	7,4	7,4	7,4
	minimum space Kindergarten [m2] (basically consisting of:)	185	555	740
<i>basic functional program</i>	1 - group area: group room, wardrobe, sanitary rooms, storage rooms [m2]	105	315	420
	2 - office and staff area: management office, meeting room, social room, staff wardrobes, staff bathroom, staff restroom [m2]	17,5	52,5	70
	3 - activity area: exercise room, multifunctional area, dining room, kitchen with food storage and refrigerated storage [m2]	27,5	82,5	110
	4 - other space: wheelchair-accessible restroom, buggy storage, laundry room, cleaning supplies storage, general storage room, storage room for garden equipments, garden restrooms [m2]	15	45	60
	5 - circulation space: proportion of traffic area to the sum of usable area and sanitary area is approx. 12-15 % [m2]	20	60	80
<i>utility rooms</i>	minimum space [m2]	35	35	35
	8 - building services: electricity supply, heating, boiler, etc. [m2]	20	20	20
	9 - garbage room [m2]	15	15	15
<i>extended program (optional/expandable)</i>	6 - therapy/relaxing area [m2]	20	20	20
	7 - creativity space: atelier, handicraft studio, etc. [m2]	20	20	20

Online Resources:

Raum- und Funktionsprogramm für Kindergärten mit 3/4 Gruppen:
<https://www.wien.gv.at/kontakte/ma10/vier-gruppen.html>

Planung und Errichtung - Kindergärten und Horte



<https://www.wien.gv.at/kontakte/ma10/planung.html>

Städtische Kindergärten - Planung

<https://www.wien.gv.at/stadtentwicklung/architektur/oeffentliche-bauten/kindergaerten/index.html>

Raumbuch Kindergarten:

<https://www.wien.gv.at/wirtschaft/auftraggeber-stadt/gebäudemanagement/pdf/raumbuch.pdf>

Raumblätter Kindergarten:

<https://www.wien.gv.at/wirtschaft/auftraggeber-stadt/gebäudemanagement/pdf/raumblatt-kg.pdf>

Spielplatzverordnung:

<https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=LrW&Gesetzesnummer=20000018>

Education:

Bildungsplan im städtischen Kindergarten:

https://www.ris.bka.gv.at/Dokumente/Landesnormen/LWI40004982/Anlage_1.pdf

Animal Assisted Therapy with Alpakas "AATLA" <https://www.aatla.de/>

STUDIO SCHEDULE

1. Thursday 03 March 09:00 - 13:00
2. Thursday 10 March 09:00 - 13:00 Site Visit
3. Thursday 17 March 09:00 - 16:00 Interim Presentation
4. Thursday 31 March 09:00 - 13:00
5. Thursday 07 April 09:00 - 13:00 Workshop
- Thursday 14 April Easter
- Thursday 21 April Easter
6. Thursday 28 April 09:00 - 13:00
7. Thursday 05 May 09:00 - 13:00
8. Thursday 12 May 09:00 - 16:00 Interim Presentation
9. Thursday 19 May 09:00 - 13:00
10. Wednesday 25 May 09:00 - 13:00
- Thursday 26 May Ascension Day
11. Thursday 02 June 09:00 - 13:00
12. Thursday 09 June 09:00 - 13:00
13. Wednesday 15 June 09:00 - 13:00
- Thursday 16 June Corpus Christi
14. Thursday 23 June 09:00 - 13:00
15. Thursday 30 June 09:00 - 16:00 Final Presentation



COMPUTATIONAL TOOLS

QGIS

Homepage: <https://qgis.org/en/site/>

User Guide: https://download.osgeo.org/qgis/doc/manual/qgis-1.0.0_user_guide_en.pdf

SAGA GIS

Homepage: <http://www.saga-gis.org/en/index.html>

Info: https://en.wikipedia.org/wiki/SAGA_GIS

Download link for the 7.6.3 version: https://sourceforge.net/projects/saga-gis/files/SAGA%20-%207/SAGA%20-%207.6.3/saga-7.6.3_x64.zip/download

Manuals: <https://sagatutorials.wordpress.com/training-manual/>

Tutorials: <https://sagatutorials.wordpress.com/about-saga-gis/>

Rhino

Homepage: <https://www.rhino3d.com/>

Rhino User Guide: <http://docs.mcneel.com/rhino/6/usersguide/en-us/index.htm>

Rhino Tutorials: <https://www.rhino3d.com/tutorials>

Rhino Community: <http://v5.rhino3d.com/>

Rhino Discussion Forum: <http://v5.rhino3d.com/forum>

Grasshopper

<https://www.rhino3d.com/de/6/new/grasshopper>

<https://www.grasshopper3d.com/>

Docofossor - Grasshopper plugin

<https://www.food4rhino.com/en/app/docofossor>

<https://github.com/dbt-ethz/docofossor/blob/master/DOCUMENTATION.md>

SpaceChase - Grasshopper plugin

<https://www.food4rhino.com/en/app/spacechase?lang=en>

<https://spacechase.app/>

Ladybug Tools - Grasshopper plugin

<https://www.ladybug.tools/>

<https://www.food4rhino.com/app/ladybug-tools>

<https://www.youtube.com/watch?v=Wxa8k5uZqXY>

Lands Design - Grasshopper plugin

<https://www.landsdesign.com/>

<https://www.food4rhino.com/app/lands-design>

<https://www.landsdesign.com/download/>

<https://www.youtube.com/watch?v=L-NjrLpWRNk>