Ontological Modelling of Form and Function for Architectural Design

Mehul Bhatt, Joana Hois, and Oliver Kutz

Spatial Cognition Research Center (SFB/TR 8)
University of Bremen, Enrique-Schmidt Strasse 5
Bremen 28359, Germany
{bhatt, joana, okutz}@informatik.uni-bremen.de

Abstract. Form, function and the relationship between the two serve a crucial role in design. Within architectural design, key aspects of the anticipated *function* of buildings, or of spatial environments in general, are supposed to be supported by their *structural form*, i.e., their shape, layout, or connectivity. Whereas the philosophy of form and function is a well-researched topic, the practical relations and dependencies between form and function are only known implicitly by designers and architects. Specifically, the formal modelling of *structural forms* and resulting *artefactual functions* within design and design assistance systems remains elusive.

In our work, we aim at making these definitions explicit by ontologically modelling respective domain entities, their properties and related constraints. We interpret "(structural) form" and "(artefactual) function" by specifying modular ontologies and their interplay for the architectural design domain. A key aspect in our modelling approach is the use of formal conceptual requirements and qualitative spatial calculi as a link between the structural form of a design and the differing functional capabilities that it affords or leads to. We demonstrate how our ontological modelling reflects types of architectural form and function, and how it facilitates the conceptual modelling of requirement constraints in architectural design.

Keywords: Ontology, Architectural Design, Structure and Function, Modularity, Qualitative Spatial Modelling, Multi-Perspective Semantics

1. Introduction

The doctrines associated with the notions of Form and Function have been the cornerstones of the Modernist tradition in engineering design [51, 71]. In architectural design, their broad interpretation is that the *structural form*, i.e., *shape*, *layout*, *connectivity*, of and within a building should be primarily (or more rigidly: *solely*) determined by its practical *function* or *purpose*. Much of the philosophical literature about design and architecture and the ensuing debates have focused on the semantics of *functions* with respect to design artefacts and the causal link between *form* and *function*. The emphasis has been on the question whether form should, or indeed does, follow function wholly or in part.

Despite these philosophical debates and the acceptance of these topics as being important and mainstream within the philosophical discourse on design thought, these concepts are understood by designers only implicitly. Contemporary architectural design and assistive technologies—with their methods, tools, systems, and design paradigms—lack a formal characterisation of *semantics, structure, function, behaviour* and *user-centred design* [12]. The world of Computer Aided Architectural Design (CAAD) rests on primitives such as points, line-segments and polygons – abstractions that are all important and even necessary, but they only provide a limited perspective that cannot comprise the rich ontological understanding of form and function in the philosophical discourse.

Our work is motivated by the practical concerns surrounding the formal interpretation of the terms "*structural form*" and "*function*" with respect to their applicability in spatial and architectural design assistance

Fig. 1. A Malfunctioning Structure; Mutually Interfering Functions

systems. Consider the scenario in Figure 1, which aims at illustrating the core themes addressed in this paper, namely, structural form, function and design semantics. The scenario consists of two key objects types, Bench and Bin. Independently of each other, both entities may be regarded as achieving their intended function, and also afford precisely the kind of actions for which they have been conceived. However, their relative placement shown in Figure 1 is not optimal, as users of the Bench may not want to use it when the Bin is full with garbage, or possibly (considerate) users of the Bin may not prefer to use it when the Bench is occupied by people. This situation can be paraphrased by common-sense knowledge about the placement of these objects as follows:

> *Bins may typically not be placed <u>directly next-to</u> or in <u>close proximity</u> to Benches. Bins, when full, are likely to stink, and this may dissuade people from utilizing Benches in <u>close vicinity</u>.*

Although this example is rather simple and may not require any complex reasoning, it points out the main issue of this paper, namely that common-sense conceptual reasoning, e.g., facilitated by ontological inference, about the semantics of a design, structure and function is a capable and promising feature for design (assistance) systems of the future. In this relatively simple example alone, an informed reader may already notice the relevance of several forms of logics and patterns of ontological inference, e.g., pertaining to topological and distance logics, and common-sense conceptual reasoning about the properties of objects.

The focus of this paper is on the strictly spatial aspects concerning the identification of a particular functionality, i.e., economic, aesthetic, social and cultural constraints are not investigated in this paper. Addressing all these diverse aspects together is a Herculean task that cannot be fully addressed by the spatially-driven objectives of this paper. We put structural form, function and design semantics into practice by formalising ontological specifications accordingly, in particular, by using modularly constructed ontologies for the domain of architectural design. A key aspect of our modelling approach is the use of formal qualitative (and semi-qualitative, e.g., for absolute distance) spatial calculi and conceptual requirements as a link between the structural form of a design and the differing functional capabilities that it affords or leads to. In essence, certain structural forms are inherently (in)capable of producing desired effects with respect to a pre-specified set of requirements conceptually expressed by an architect or a designer. The main objective of formally modelling aspects pertaining to form and function is to ensure that automated reasoning within design assistance systems becomes possible. Our research encompasses the following facets, as identifiable within the broader context of *spatio-linguistic, formal, and computational* aspects of 'space' within spatial assistance systems [12] in general, and spatial computing for design [9] in particular:

Formal Ontological Modelling in Architectural Design. The ontologies we develop particularly for the architectural domain are based on different standards and methodologies from ontological engineering as well as architectural design tools. Those ontological specifications that describe the structural aspects of a design, i.e., the floor plan and its relevant information, build on a standardised format for building
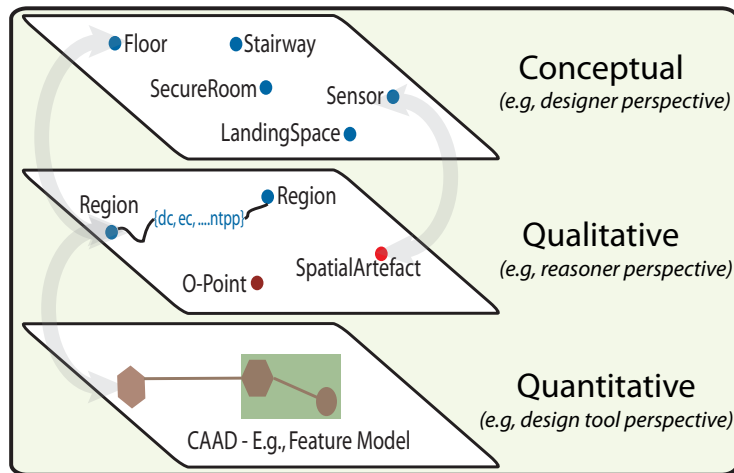
Fig. 2. Multi-Perspective Semantics. (Source: [9])

designs, namely the *Industry Foundation Classes* (IFC) [27]. Ontological specifications that are related to (qualitative) spatial information are based on different formalisms for qualitative spatial representation and reasoning [25, 20]. Terminological (conceptual) ontologies for specifying high-level design constraints formulated by designers, architects or engineers are grounded in foundational ontological engineering methods [53, 78]. Substantially, the ontological modelling of these different aspects, i.e., design, form, function, architectural parts and requirements, are tailored to the architectural design domain and relate form and function in terms of their spatial constraints.

Qualitative Spatial Modelling. A crucial aspect that is missing in contemporary design tools is support for explicitly characterising functional requirements of a design. Especially when considering the new generation of building automation systems and smart environments, this is very limiting [5]. For instance, although it is possible to model the spatial layout of an environment at a fine-grained level, it is not possible in the currently available tools for architectural design to model *spatial artefacts*, such as the *range space* of a sensory device (e.g., a camera or a motion sensor). Although this is not strictly a spatial entity in the sense of having a material existence it needs to be treated as such. For instance, consider the following constraint: '*A motion sensor should be placed such that the door connecting room A and room B is always within the sensor's range space*'. The capability to model such a constraint is absent from even the most state-of-the-art design tools. Furthermore, conventional design expertise is often driven by experience and intuition. It is concerned with spatial and structural aspects of the design rather than its functional characterisation. Here, we augment precisely these structural aspects by using qualitative spatial constraints to model their functional characteristics.

Multi-Perspective Semantics and Modularity. Consider the illustration in Figure 2: an abstraction such as a Room or Sensor may be identified semantically by its placement within an ontological hierarchy and its relationships with other conceptual categories. These different categories are used by a designer during the initial design conceptualisation phase. However, when these types are transferred to a Computer-Aided Architecture Design (CAAD) tool, the same concepts acquire a new perspective, i.e., the designer has to interpret design concepts in terms of points, line-segments, polygons and other geometric primitives available by the design tool. Such primitive concepts are necessary yet in conflict with the mental image and qualitative conceptualisation of the designer. Given the lack of design semantics within contemporary design tools, no solution is available for a knowledge-based system to make inferences about the conceptual design and its geometric interpretation within a CAAD model in a unified manner. We model these different perspectives and their respective spatial semantics by using modular ontologies that individually comply with one of the perspectives. As we can distinguish mainly conceptual, qualitative, quantitative and requirements-specific perspectives on architectural designs, our ontological modules

have been developed on the basis of this distinction. These modularly specified ontologies not only reflect the thematically different perspectives in a more adequate way but they also provide a clear ontological representation of the interplay and exchange between form and function and their spatial constraints and they can also directly support modular and spatial reasoning. Technically, we apply different combination techniques for modular ontologies, in particular the theory of $\mathcal{E}$-Connections [47] and refinements [49]. The concept of multi-perspective semantics may be elaborated on with additional details pertaining to its interpretation within a formal and computational framework [9, 12]. In this paper, we focus on a high-level overview.

The paper is organised as follows: Section 2 illustrates the basic concepts of structural form and artefactual function. We employ simple though real design scenarios and requirements in order to exemplify the relationship between form and function. Section 3 focuses on the role of different thematic perspectives on domains and their modular ontological representation, in particular, the applicability of modularity to model multiple perspectives in architectural design. Section 4 elaborates on the use of spatial logics and description logics to formalise the 'design semantics', i.e., high-level conceptual knowledge and requirements for architectural design. Section 5 builds on Section 3 and presents technical issues for handling multi-perspective representations using the theory of $\mathcal{E}$-connections. Section 6 then presents in detail an exploratory study of utilizing these modelling constructs to capture the real-world examples introduced in Section 2 by using modular ontological specifications for the different perspectives. Finally, Section 7 provides discussion and outlook for the work described in the paper.

2. Structural Form and Artefactual Function in Architectural Design

A crucial element that is missing in conventional architectural design systems pertains to formal modelling, i.e., representation of and reasoning over 'architectural structures'. Formal modelling of the structural form of an environment, and common-sense reasoning about the different functional capabilities that it affords or leads to is necessary to ensure that design objectives are met when the design is deployed in reality. As all architectural design tasks are concerned with a spatial environment, formal representation and reasoning along conceptual and spatial dimensions are essential to ensure that the designed model satisfies key requirements to enable and facilitate intended functions.

2.1. Structure and Function: an Abstract Characterisation

A designer or an architect envisions a structure with respect to its anticipated behaviours to satisfy desired functions. This view of the *functional* aspect of a design bears close relationships to the ontological model Function-Behaviour-Structure (FBS) [30, 31] for the design process. For the purposes of this paper, we interpret *structural form* and *artefactual function* as follows:

Structural Form.   The *structural form* of an environment is an abstraction mechanism generally corresponding to the layout, shape, relative arrangement and composition at the common-sense level of spatial entities, artefacts and other abstract or real elements that are modelled geometrically, interpreted or derived within a design system. The only conceivable premise underlying this notion is that it should be possible to communicate the conception of the structural form using one or more spatio-linguistic modalities – e.g., spatial prepositions, path and pattern descriptions, region and point-based abstractions – that may be wholly or partially grounded in an underlying physical structure either in metric space or in an abstract qualitative space.

Artefactual Function. Artefactual functions (also referred to as '*functions*') correspond to the overall behaviour or set of behaviours that a particular structural configuration or arrangement affords, produces or leads to. For the purposes of this paper, functions essentially correspond to behaviours associated with high-level design requirements that are ontologically interpreted as sets of constraints within a task-specific design requirement ontology. From the viewpoint of an ontological terminology, they may also be interpreted as categories with specific relationships and properties specified in an architectural requirement ontology.

This abstractly presented interpretation of *structural form* and the resulting *artefactual function* is applicable beyond the domain of architecture design, although architecture remains the focus area in this paper. Structural forms may also entail functionalities that may be interpreted in the context of affordances, design aesthetics, subjective emotional reactions, etc., as applicable in a wide-range of domains, such as architectural design assistance, creative assistance in media design[1] or ambient intelligence—broadly, the class of systems referred to as *spatial assistance systems* [12].

A discussion of these abstractions also inevitably requires a discussion of the engineering aspects related to CAD / CAAD models, which is not necessary for this paper. We refer readers to [12] for our definitive take on the *conceptual*, *formal*, and *computational* aspects of '*space*' within a wide-range of (spatial) assistance systems beyond the architecture design domain. For this paper, the abstract interpretation of structural form and artefactual function is sufficient. The notion of *structural form* is formally specifiable via a detailed characterisation involving graph-theoretic and qualitative formalisations. A technically detailed overview of the multi-modal characterisation of the *structural form* of (indoor) spatial environments in the context of design systems is available in [64, 65].

An additional note in this context is that this interpretation of functions only refers to those aspects that emanate directly from structural form; specifically, this paper is concerned with functions that are identifiable directly via semantic, physical, and logical constraints. Functions encompassing social, cultural, and economic constraints are beyond the scope of this article.

In the rest of the paper, we illustrate these concepts using concrete examples for the domain of spatial design for architecture. Furthermore, we provide practical examples how these aspects are formalised by using modular ontological representations.

2.2. A Design Task

Consider an architect specialising in the design and development of any general building environment as a basic use case. An example of a typical design challenge is to:

> *Design the layout of an office environment to satisfy structural and functional requirements that collectively aid and complement (and never hinder) the building's automation systems (monitoring devices, sensors, etc.), and which, by implication, facilitate the intended smartness of such automation systems.*

From the viewpoint of the overall design requirements, aspects of this problem explicitly pertain to the functional aspects (e.g., security, privacy, building-automation, accessibility) of the space being modelled, structural code-checking with respect to building regulations, and also possibly specialised client demands. It is worth noting that often different expert or interest groups tend to refer to similar functional expectations in a different language, e.g., a designer may express a high-level feature such as privacy or security in a qualitative manner, whereas a statutory body or a structural engineer would express the same requirement in precise quantitative terms, e.g., referring to exact visibility and proximity parameters. Although our work does not deal with the semantic similarity underlying such overlapping specifications of form and function, this capability is an important area of the ongoing and future outlook of our research.

---

[1]Examples of media design assistance include tasks such as automatic story-boarding for film and comic design, virtual cinematography for film and animation; the relationship this class of creative design work is available in [8, 12].

Our illustrations and formalisations in this paper distinguish three types of functional requirements:

I. Client Specifications, Expert Knowledge

Client specifications or expert knowledge are general requirements or basic characteristics of architectural designs. For example, certain areas within a building, floor, or room should (not) be trackable by sensing devices such as cameras or motion sensors. As much as possible, the operation of doors should be non-interfering with the functionality of nearby utilities. Another example is:

Example 1 (A Sunny Counter) *"Place the main part of the kitchen counter on the south and south-east side of the kitchen, with big windows around it, so that sun can flood in and fill the kitchen with yellow light both morning and afternoon"*

(A Pattern Language (p. 916–918) [1])

II. Statutory Requirements

Regional statutory requirements that stipulate structural constraints and other categorical specifications, e.g., as stipulated by disability access codes. An example follows:

Example 2 (Staircase) *"Steps of a staircase may not be connected directly to a door that opens in the direction of the steps. There has to be a landing between the staircase steps and the door. The length of this landing has to have at least the size of the door width".*

(Bremen (Germany) Building code [16]: Staircase (§35 (10), p. 24))

III. Specialised Requirements

Specialised requirements correspond to those aspects that arise as a direct result of the specialised nature of the environment being designed, e.g., the design of museums, courthouses, airports, train stations. An example follows:

Example 3 (US Courts Design Guide) The US courts design guide stipulates an elaborate set of requirements, ranging from precise structural specifications to imprecise, fuzzy, and sometimes rather vague guidelines involving cultural, aesthetic, and legal dimensions. Examples are:
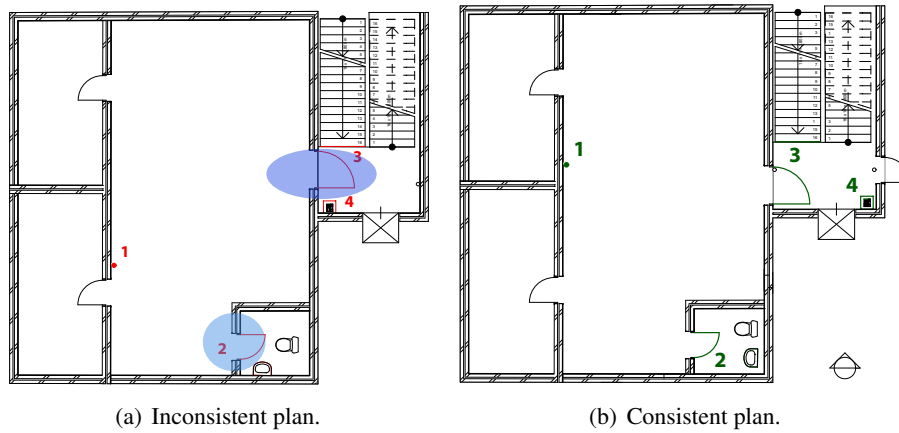
Witness-Box Placement: *"Witnesses must be able to see and hear, and be seen and heard by, all court participants as close to full face as possible".*

Barrier-Free Accessibility: *"Courtroom areas used by the public must be accessible to people with disabilities. Private work areas, including the judge's bench and the courtroom deputy, law clerk, bailiff, and court reporter stations, must be adaptable to accessibility. While all judge's benches and courtroom personnel stations do not need to be immediately accessible, disabled judges and court personnel must be accommodated".*

Judge's Bench Placement: *"The height and location of the judge's bench expresses the role of the judge and facilitates control of the court. Generally, the judge's bench should be elevated three or four steps (21-24 inches or 525-600 mm) above the courtroom well.".*

Visibility: *"The entrance or entrance vestibule should be clearly visible and recognisable as such from the exterior of the building. The vestibule should be a minimum of 7 feet in depth and able to handle the flow of traffic at peak times.".*

(US Courts Design Guide 2007 [76])

(a) Inconsistent plan.   (b) Consistent plan.

Fig. 3. Two designs of a floor plan: it is not identifiable directly from their raw metrical data if they are consistent with certain design requirements.

Figure 3 is an example schematisation to illustrate the different categories of requirement constraints discussed above. It consists of a consistent and inconsistent model of the example requirements. The aspects marked by the numbers 1–4 in Figures 3(a) and 3(b) determine the plans to be (in)consistent with respect to the following requirements:

– The camera sensor is placed at a position where a private area (the wash room) is within its range (No. 1)

– The operating space of the door of the wash room interferes with the functional area of the sink; this arrangement is also not conducive to disability access requirements (No. 2)

– The operation of the main entrance door interferes with the function of the telephone next to it; from a structural viewpoint, it is also not an ideal placement given its proximity to the staircase (No. 3 and 4)
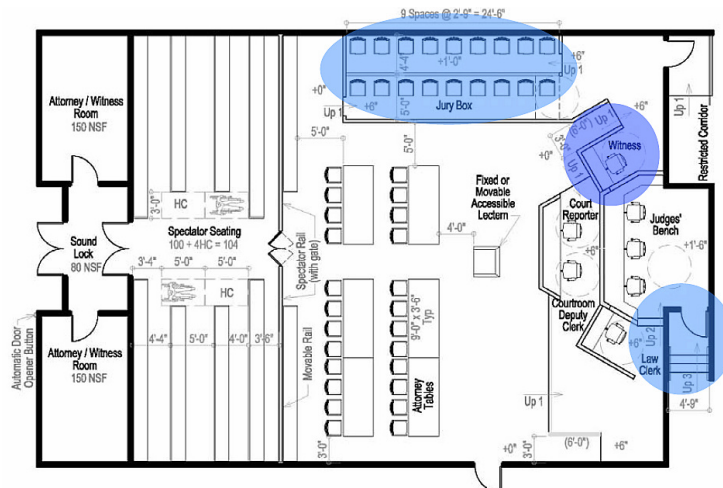

Fig. 4. A courtroom sample, in which the marked areas have to satisfy particular functional requirements for optimal locations. (Source: US Courts Design Guide 2007 [76])

Figure 4 is an example design for a courtroom adapted from the US Courts Design Guide, which includes this design to illustrate how the main court proceedings area should be designed. The marked regions in Figure 4 highlight the requirements from Example 3, namely the Witness-Box Placement, the Judge's Bench Placement and Visibility. The spatial structure of these requirements may be interpreted basically in terms of topological and orientation-based constraints, which is discussed in more detail in Section 6.2.

3. Ontological Modularity and Multi-Perspective Representation

Modularity as well as semantic heterogeneity have become key issues in ontology engineering in recent years [70].[2] Research into aspects of modularity in ontologies covers a wide spectrum. Applied to ontology engineering, modularity is central not only to reduce the complexity of understanding ontologies, but also in maintaining, querying and reasoning over modules. Here, distinctions between modules can be drawn on the basis of structural (e.g. syntactic, graph-theoretical), semantic, purely logical, functional or pragmatic aspects [6, 7, 13, 70]. A collection of already constructed ontological components can then be exploited by composing or combining such modules, or by establishing '*links*' between such ontologies. The main research question here is how to define the notion of *module* and how to re-use such modules. An extensive discussion of such methods can be found in [70, 49].

In particular, the re-use and sharing of information and resources encoded in different ontological (or logical) modules depend on purpose-dependent, logically versatile criteria. In our present context, such purposes include 'tight' logical integration of different ontologies (wholly or in part), 'loose' association and information exchange across different representation formalisms, and alignment of vocabularies.

The problem of semantic heterogeneity is particularly relevant and inherent in the problem of formal modelling in architectural design. Concerning heterogeneity in ontology engineering, we endorse the pragmatic version of *logical pluralism* as advocated in [49]. In a nutshell, this maintains that different levels of expressivity, reasoning modes and envisioned application areas ask for a variety of logical languages being used in formal ontology design. This position carries over to the spatial modelling and reasoning scenarios that we describe in this paper, and it is principally supported by two main arguments: the first derives from the position that heterogeneous and modular modelling is cognitively more adequate, i.e. easier and more useful for the modeller, and the second maintains that distributed and heterogeneous reasoning has computationally great advantages over monolithic reasoning.

Concerning the first argument, universal modelling languages such as first-order logic, although expressive enough for most purposes, exhibit severe drawbacks when formalising heterogeneous information. First of all, their syntax is not fine-tuned for describing spatial scenarios, which negatively impacts their usability as modelling languages. For instance, some spatial calculi use linguistic terms to define spatial relations (e.g. the Region Connection Calculus RCC-8 discussed in Section 4). The aim is to be more cognitively adequate in the sense that the formal terms introduced reflect distinctions humans (in our case, especially architects and designers) would typically make and find relevant. RCC-8 distinguishes relations such as 'overlap' and 'disjointness', but also 'tangential proper part'. The simpler RCC-5, on the other hand, does not consider the boundary of a region and is thus seen as more cognitively adequate in situations where humans would not consider this distinction to be relevant (see [19] and compare [43] for such an analysis concerning Allen's temporal interval calculus). Another case are modal and temporal operators (compare the logics of distance introduced in Section 4), whose usage could be seen as cognitively more adequate than first-order logic as these operators more directly reflect natural language (and its semantics)—quantifying over points in a region and explicitly specifying distance relations, for most people, perhaps is not the most straightforward way of writing down a 'distance' statement statement such as '10 meters around' or 'closer to A than to B'.[3]

Concerning the second, there is a vast literature on logical formalisms fine-tuned to specific modelling domains and scenarios and computationally optimised for such reasoning tasks with corresponding specialised reasoning engines (see, e.g., [28, 77] for overviews). Using an expressive modelling language such as first-order logic does provide a uniform formalisation, yet very quickly yields situations in which con-

---

[2]Additionally, the workshop series on Modular Ontologies [46] gives a good overview of the breadth of this field. `www.informatik.uni-bremen.de/~okutz/womo5/`.

[3]Cognitive adequacy is also related to the *succinctness* of logical formalisms. For instance, many first-order statements can be equivalently formalised in various modal or temporal formalisms, but the first-order formulae may be exponentially shorter (and thus more intricate).

sistency checks, which are the main reasoning task that we are concerned with here, become impossible. Evidence for this claim can be given by an attempt at proving the consistency of the foundational ontology DOLCE, the 'Descriptive Ontology for Linguistic and Cognitive Engineering' [29, 54]. It contains several hundred axioms formulated in first-order logic.[4] The complexity of the DOLCE ontology stems from the fact that it combines several (non-trivial) formalised ontological theories into one theory, i.e., the theories of essence and identity, parts and wholes (mereology), dependence, composition and constitution, as well as properties and qualities. Trying to prove the consistency of DOLCE using standard first-order theorem provers or model finders proves impossible even for small fragments of the overall ontology. Indeed, the difficulties already arise for the rather tiny sub-theories 'classical extensional parthood' (CEP) and 'constitution' (CON) of DOLCE. CEP is a theory of mereology, and it is straightforward to see that finite models for it can be obtained by powersets of finite sets, where the empty set has to be excluded. The singleton sets are then just the atoms of the mereology. Standard techniques do not find models with more than four atoms for these theories. Moreover, several weeks of computation time do not suffice to find a model for the whole of DOLCE (compare [48] for technical details).

A cure for this is given by a deep modularisation of the consistency problem (again see [48]), which can be achieved by structured specification, i.e., keeping reasoning problems 'independent' and 'local' as far as possible, and delegating reasoning problems to specialised reasoners whenever possible, which is exactly the modelling paradigm that we are following in this paper. Tool support for developing such heterogeneous ontologies (or, more generally, theories) is available via the Heterogeneous Tool Set HETS (see [58] for a system description). This system provides parsing, static analysis and proof management for heterogeneous logical theories, visualising the module structure of complex logical theories. Moreover, HETS is able to prove intended consequences of theories, prove refinements between theories or demonstrate their consistency.[5]

Following this analysis, we introduce several relevant formalisms for spatial design in Section 4, and then go into greater detail on how these formalisms can be combined with ontological modelling on the basis of the theory of $\mathcal{E}$-connections in Section 5.[6]

We next briefly summarise some of the general aspects concerning ontological modularity by giving a birds eye view of problems and approaches, and discuss which of these approaches we use in our formal modelling for architectural design in Section 3.2.

3.1. Aspects and Dimensions of Ontological Modularity

The main dimensions of ontological modularity and respective (automated) reasoning challenges are:

The Language Layer and Semantic Heterogeneity. Whenever we want to combine two ontologies (or formal theories), we run into the problem of syntactic and semantic heterogeneity. Indeed, even if we stay in the same formal logic, we run into the problem of reconciling the joint vocabulary of the ontologies. The most general solution to this problem is to provide a family of logic translations that allows to seamlessly move from one logic to another along the translation, based on a general definition of logic and logic translation as provided by institution theory [32, 57]. Tool support for such translations is, for instance, provided by the HETS system [58].

---

[4]There are also versions of DOLCE including some axioms using modal logic, but they do not concern the heart of DOLCE, and leaving them out does not in any way trivialise the consistency problem [54].

[5]This is achieved by integrating several reasoning tools, amongst them first-order provers and model-finders (SPASS, DARWIN, VAMPIRE and others), the higher-order prover ISABELLE, the DL reasoners PELLET and FACT++, as well as SAT solvers.

[6]For a detailed analysis of $\mathcal{E}$-connections understood as heterogeneous theories, compare [49].

Structuring, Extension, and Refinement.   The mere size of ontologies can make the design process quite hard and error prone (at least for humans). This issue has been only partly cured in OWL by the `imports` construct, which essentially copies the axioms of one ontology into another. Natural operations are, for instance, union, intersection, 'hiding' certain symbols, and extension. However, the semantics of such operations is in general non-trivial. For example, methods developed for (algebraic) specification can be applied to ontology engineering, as they provide systematic structuring techniques [49].

Apart from such structuring concepts, another natural relationship between ontologies is a refinement: An ontology O_2 refines an ontology O_1 if all of O_1' s axioms are entailed by O_2 (possibly along a translation). Essentially, this means that we need to provide a theory interpretation of O_1 into O_2 [44]. Another kind of 'extension' is provided by the idea of *concrete domains*. They extend an ontology language by constructs that allow to 'import' computations in specific structures, such as the natural numbers or time intervals [35].

Logical Independence.   One of the most important logical concepts of modularity is given by *conservativity*. An ontology O_2 is a conservative extension of O_1 if all assertions made in the language of O_1 that follow from O_2 already follow from O_1. Essentially, this means that O_1 completely and independently specifies its vocabulary, with respect to O_2. This concept can be used to extract logically independent modules from a large ontology. While this notion of module is thus important, it is also computationally difficult. Although proving conservativity is undecidable for first-order logic and many expressive description logics (DL) [70], general algorithmic solutions exist for less expressive DLs [44]. The simplest case of a conservative extension is a *definitional extension*, as it extends the vocabulary of an ontology O by new terms, whose meaning is entirely determined by the axioms given in O.

Matching and Alignment.   Matching [24] and aligning [79] ontologies focus on the identification of (thematically) overlapping parts of two ontologies (*matching problem*) and on systematically relating terms across ontologies that have been identified as, for instance, synonymous (*alignment problem*). As opposed to structuring and conservativity, such relationships are often established by using statistical methods and heuristics, e.g., employing similarity measures and probabilities.

Integration and Connection.   Informally, an *integration* of two ontologies (O_1 and O_2) into a third ontology $\mathcal{O}$ is any operation by which O_1, O_2 are 're-interpreted' from the (global) point of view of $\mathcal{O}$. This has been utilised in the approach of [63] (called *semantic integration*), which integrates two ontologies by mapping (or translating) them into a common reference ontology. The main feature here is that semantic consequence is preserved upwards to the reference ontology.

Intuitively, the difference between *integrations* and *connections* is that in the former, we combine two ontologies O_1 and O_2 using an often large and previously-known reference ontology $\mathcal{O}$, where the models of $\mathcal{O}$ are typically much richer than those of O_1 and O_2. In the latter, we connect two ontologies by keeping the respective theories, signatures and models disjoint, and a (usually small and flexible) *bridge theory* is formulated (in a bridge language) over a signature that *goes across* the sort structure of the components to *link together* the two ontologies. Connections, in the form of $\mathcal{E}$-connections, are introduced in detail in Section 5 and employed for modelling architectural design in Section 6.

3.2. Modularity in Architectural Design Specifications

The aspects of ontological modularity that we employ in order to realise the envisioned application to architectural design are manual alignments, conservative (definitional) extensions, $\mathcal{E}$-connecting thematically different ontologies, and global extension.

Thematic module. A *thematic module* for a domain D is an ontology that covers a particular perspective on D. Its main impact is that we assume two thematically different modules for D to be interpreted by disjoint domains. An example, that we elaborate on later, is the conceptual space of materials of objects and qualitative representations of topological relationships between such objects: these interpretations clearly should not overlap. We specify such thematic modules as modular ontologies that are part of a certain architectural perspective, e.g., conceptual or qualitative spatial perspectives that are introduced in Section 6.1.

Definitional Extensions. New concepts are, for instance, added to the DOLCE-Lite ontology in the *conceptual* ontology module (see Section 6.1) by a definitional signature extension. Moreover, we add new concepts to the spatial relations in the ontology of structural building entities, again, in a definitional manner.

Linking thematic modules. Alignments are given by the human expert (the architect or designer), identifying certain relationships between thematically different modules. An overall integration of these thematic modules is achieved by $\mathcal{E}$-connecting the aligned vocabulary along newly introduced link relations and appropriate linking axioms.

Global extensions of integrated representations. New constraints are added on top of the integrated representation by $\mathcal{E}$-connections. Moreover, the process of building integrated representations might be iterated at a later stage of the specification process. This allows the integration of further ontologies, which treats the previously built representation as a new 'unified' building block (see Section 6.2).

We have presented some of the key aspects of ontological modularity in general and summarised the aspects of multi-perspective modelling that we employ for the domain of architectural design presented in Section 6. We next turn to the concrete logical formalisms that we use in our multi-perspective modelling.

4. Spatial Logics and Conceptual Modelling for Architectural Design

The examples in Section 2.2 have shown that different types of spatial information are used when describing architectural functions or requirements. Among them are **cardinal directions** (*the south-east side of the kitchen*), **regions** (*a landing between the staircase steps and the door*), **shape** (*a window around it*), **orientation** (*a door that opens in the direction of the steps*), and **distance** (*the bench should be elevated three or four steps above the well*). Specific spatial logics that formalise these types of spatial information can thus be used for modelling architectural design. More importantly, spatial logics can be used for reasoning over design models, i.e., for analysing model and requirements consistency, and we have argued in the last section that the use of several special purpose formalisms (and reasoners) is preferable over using a single very expressive formalism (and reasoner or model-checker). Spatial logics include variants and fragments of classical first- and second-order logics, such as Description Logics and Common Logic, and of course many qualitative and (semi-)quantitative spatial logics. The most important aspects that need to be covered by such spatial logics are topology, distance, shape and orientation.

For the purposes of this paper, we focus on modelling aspects and define in some detail DLs (Section 4.1), standard qualitative spatial calculi such as RCC-8 and logics for reasoning over distances, similarities, orientations (Section 4.2) and combinations of such calculi based on the theory of $\mathcal{E}$-connections (defined in Section 5).

4.1. *Description Logics and* OWL

The Web Ontology Language (OWL) has been specifically designed for the 'Semantic Web'. It builds on existing web standards such as XML and RDF whilst being semantically grounded in the formal rigour of expressive *Description Logics* (DL) [41, 40], which we describe in some more detail in the following.

Signatures of the description logic $\mathcal{ALC}$ consist of a set $\mathcal{A}$ of atomic concepts, a set $\mathcal{R}$ of roles (relations or properties) and a set $\mathcal{I}$ of individual constants (instances). Models are single-sorted first-order structures that interpret concepts as unary and roles as binary predicates.

Constructor	DL Syntax	Example
Intersection	$C_1 \sqcap \ldots \sqcap C_n$	Door ⊓ MainEntrance
Union	$C_1 \sqcup \ldots \sqcup C_n$	SwingDoor ⊔ SlidingDoor
Complement	$\neg\, C$	¬ EmergencyExit
Universal Restriction	$\forall\, R.C$	∀ has_material . Material
Existential Restriction	$\exists\, R.C$	∃ has_material . Wood ⊔ Aluminium
Max Cardinality	$\leqslant nR.C$	⩽ 5 has_level . Floor
Min Cardinality	$\geqslant nR.C$	⩾ 1 has_element . Door

Table 1

Examples for Description Logic Concept Constructors.

Sentences are subsumption relations $C_1 \sqsubseteq C_2$ between concepts, where concepts follow the grammar

$$C ::= \mathcal{A} \,|\, \top \,|\, \bot \,|\, C_1 \sqcup C_2 \,|\, C_1 \sqcap C_2 \,|\, \neg C \,|\, \forall R.C \,|\, \exists R.C$$

These kind of sentences are also called TBox sentences. Sentences can also be ABox sentences, which are membership assertions of individuals in concepts (written $a : C$ for $a \in \mathcal{I}$) or pairs of individuals in roles (written $R(a, b)$ for $a, b \in \mathcal{I}, R \in \mathcal{R}$). For example, OfficeDoor308 : Door can be specified in an ABox of an architectural design ontology.

Axiom	DL Syntax	Example
Subsumption	$C_1 \sqsubseteq C_2$	Wood ⊑ Material
Equivalence	$C_1 \equiv C_2$	Door ≡ SlidingDoor ⊔ SwingDoor ⊔ RevolvingDoor
Disjointness	$C_1 \sqsubseteq \neg C_2$	SlidingDoor ⊑ ¬ SwingDoor
Inverse	$R_1 \equiv R_2^-$	has_conceptual_structure ≡ has_metrical_structure$^-$
Functional Role	$\top \sqsubseteq\ \leqslant 1R$	⊤ ⊑ ⩽1 has_metrical_structure
Inverse Functional Role	$\top \sqsubseteq\ \leqslant 1R^-$	⊤ ⊑ ⩽1 has_metrical_structure$^-$

Table 2

Examples for Description Logic Axioms.

The logic $\mathcal{SROIQ}$ [40], which is the logical core of the Web Ontology Language OWL 2 DL[7] extends $\mathcal{ALC}$ with the following constructs: (i) complex role boxes (denoted by $\mathcal{SR}$): these can contain complex role inclusions such as $R \circ S \sqsubseteq S$ as well as simple role hierarchies such as $R \sqsubseteq S$, assertions for

---

[7]See also `http://www.w3.org/TR/owl2-overview/`

symmetric, transitive, reflexive, asymmetric and disjoint roles (called RBox sentences) as well as the construct $\exists R.\mathsf{Self}$ (collecting the set of 'R-reflexive points'); (ii) nominals (denoted by $\mathcal{O}$); (iii) inverse roles (denoted by $\mathcal{I}$); (iv) qualified and unqualified number restrictions ($\mathcal{Q}$). For details on the rather complex grammatical restrictions for $\mathcal{SROIQ}$ (e.g., regular role inclusions, simple roles) compare [40]. Apart from some exceptions[8], description logics can be seen as fragments of first-order logic via the standard translation [2] that translates both the syntax and semantics of various DLs into untyped first-order logic.

Tables 1 and 2 illustrate the various complex class constructors and TBox axioms respectively provided by basic Description Logics. All of them are supported by the present version of OWL 2 DL. The examples in the right hand side of the tables illustrate some of the usages of DL expressivity in the modularly defined ontologies specified in Section 6.1.

4.2. Qualitative and Quantitative Spatial Logics

Topology. The Region Connection Calculus (RCC) [59] is heavily being used in qualitative spatial representation and reasoning, and we give examples in Section 5 how this kind of reasoning can be combined with ontological reasoning. Figure 5 displays the 8 basic relations of RCC-8, which are mutually exclusive, pairwise disjoint and exhaustive in describing the possible overlap and touching relationships between two (well-behaved[9]) regions in space.
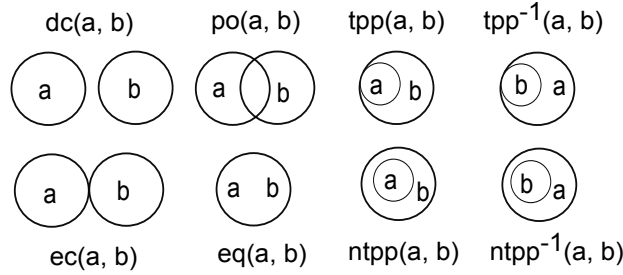

Fig. 5. The RCC-8 base relations.

To have a slightly more expressive logic at hand in which we can use the RCC-8 relations, which is more straightforwardly used in an $\mathcal{E}$-connection setting, it is convenient to encode the RCC-8 relations in a topological logic. The modal logic $\mathbf{S4_u}$, i.e., Lewis' modal system $\mathbf{S4}$ with the universal modality added, can be used for this purpose. $\mathbf{S4_u}$ is complete with respect to the semantics based on topological spaces as the intended interpretation. Here, the propositional variables are interpreted as subsets of a topological space, the necessity operator $\Box$ is interpreted as the interior operator $\mathbf{I}$, the possibility operator $\Diamond$ as the closure operator $\mathbf{C}$, and the universal quantifier $\blacksquare$ as universal quantification over all points of the topological space (and its defined dual $\blacklozenge = \neg\blacksquare\neg$ as corresponding existential quantification). Sentences are built using propositional variables and these four unary modal operators [73, 3, 66].[10]

For instance, the relation of partial overlap $po(a, b)$ between regions a and b is defined in $\mathbf{S4_u}$ as follows:

$$\blacklozenge(\mathbf{I}a \wedge \mathbf{I}b) \wedge \blacklozenge(\mathbf{I}a \wedge \neg b) \wedge \blacklozenge(\mathbf{I}b \wedge \neg a)$$

saying that a point is in the intersection of the interiors of a and b, a point in the interior of a not belonging to b, and a point in the interior of b not belonging to a, and similarly for the other relations.[11]

---

[8]For instance, adding transitive closure of roles or fixpoints to DLs makes them decidable fragments of second-order logic [14].

[9]This is typically taken to mean regular-closed subsets of a topological space, i.e., regions X such that $X = \mathbf{CI}X$.

[10]When interpreted over standard Kripke semantics, models are based on frames with reflexive and transitive relations and $\blacksquare$ is the universal quantification over worlds.

[11]See [52] for modal logics that explicitly introduce modal operators for the eight RCC-8 relations.

Distance.   Being able to specify metrical constraints about absolute and relative distance, or more qual-
itative constraints about relative 'closeness' of objects, are relevant requirements in a spatial design task.
We here introduce one such family of logics, namely the distance and similarity logics of [45, 68, 69]. The
basic idea is to augment a structure W with a distance (or similarity) measure $d : W \times W \mapsto \mathbb{R}_0^+$, which
maps pairs $\langle a, b \rangle$ of elements of W to a positive real number (including zero), called the *distance* between
the points a and b. In the context of working in the euclidean plane, d will typically be assumed to be a
metric, i.e., satisfying, for all $x, y, z \in W$, the following axioms:

$$d(x, y) = 0 \text{ iff } x = y \qquad d(x, z) \leq d(x, y) + d(y, z) \qquad d(x, y) = d(y, x)$$

Here, a distance of zero means that the objects x and y are actually located at the same position. In the
following we sketch the syntax and semantics of basic distance logics as well as their interpretation as
logics for similarity.[12] We apply these logics in Section 6.

The basic distance logics are syntactically defined just like standard modal logics such as $\mathbf{S4_u}$, i.e.,
we have a family of propositional variables $\{p_i : i < \omega\}$, Boolean connectives, $\wedge$ and $\neg$, and a list
$\{\mathsf{A}^{\leq a}, \mathsf{A}^{>a}, \dots : a \in M\}$ of (unary) modal operators depending on a set M, called the *parameter set*,
of non-negative real numbers that we allow as parameters a in formulae. Well-formed formulae in this
language are now constructed in a the standard way.

Other Booleans as well as the dual modal operators $\mathsf{E}^{\leq a}$ and $\mathsf{E}^{>a}$ are defined as abbreviations (e.g.,
$\mathsf{E}^{\leq a} = \neg \mathsf{A}^{\leq a}\neg$, $\mathsf{E}^{>a} = \neg \mathsf{A}^{>a}\neg$). Models for this logic are of the form $\mathfrak{B} = \langle W, d, p_0^{\mathfrak{B}}, p_1^{\mathfrak{B}}, \dots \rangle$, where
$\langle W, d \rangle$ is a distance space and the $p_i^{\mathfrak{B}}$ are subsets of W. The *truth-relation* $\langle \mathfrak{B}, w \rangle \models \phi$ for this language
is completely standard except for the distance operators. They can be used to define 'complex regions' as
follows:

$$(\mathsf{A}^{\leq a}\phi)^{\mathfrak{B}} = \{w \in W \mid \text{ for all } u \in W \text{ with } d(w, u) \leq a \text{ we have } u \in \phi^{\mathfrak{B}}\}$$
$$(\mathsf{A}^{>a}\phi)^{\mathfrak{B}} = \{w \in W \mid \text{ for all } u \in W \text{ with } d(w, u) > a \text{ we have } u \in \phi^{\mathfrak{B}}\}$$

Note that this language already allows to define standard modal operators such as the universal modality,
the difference operator, as well as nominals [23, 33]. In an obvious way we can also define more complex
operators, e.g., $\mathsf{A}^{>a}_{<b}$.

Rather than interpreting the measure $d(a, b) = x$ as a metrical distance, it can also be understood as a
similarity measure between a and b, where a is more similar to b the smaller the measured distance x is.
This interpretation suggests in particular the following binary operator:

$$(C \Leftarrow D)^{\mathfrak{B}} = \{w \in W \mid d(w, C) < d(w, D)\}$$

where the distance $d(w, C)$ is defined as the infimum of the distances $d(w, u), u \in C^{\mathfrak{B}}$ between w and
members of C. This operator (e.g., studied in [67]) can also define Voronoi tessellations using prototypes
(i.e., any point in the cell of a prototype a is closer to a than to any other prototype a').

Consider the example in Figure 6: we might want to introduce a region Safe_Area into a floor plan by
defining it as an area whose points are closer to the fire exit than to any regular door leading to another
room. This can be achieved by introducing two concepts: (1) Regular_Door enumerating all and only the
(points representing) doors leading to other rooms, and (2) {fire_exit} containing the point that represents
the exit. Formally, a location x can be considered as being *safe* if it belongs to the concept Safe_Area
defined as:

$$\text{Safe\_Area} \equiv (\{\text{fire\_exit}\} \Leftarrow \text{Regular\_Door})$$

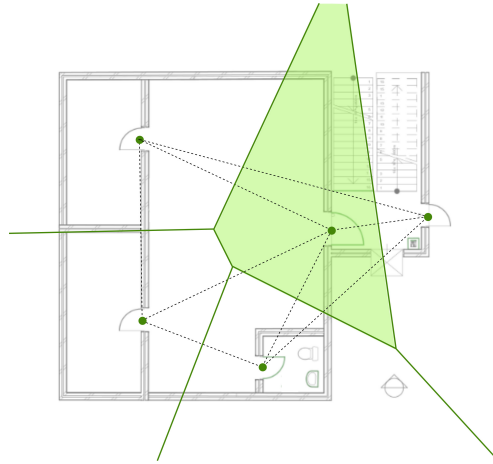[12]The basic system is taken from [50] which contains various axiomatisations of these logics.

Fig. 6. Distances between access points yielding a Voronoi tessellation of a floor plan.

Although metrical information is ubiquitous in architectural floor plans, designs that are still in progress are a good example for a mix of open and closed world reasoning, i.e., some distances may be fixed though others are left completely open (for instance, by using variables for the distances). In a floor plan, some details might be specified entirely (and database reasoning can be applied), while other aspects are left open. In general, when a floor plan is seen as a model for a distance logic, it corresponds to a map-like representation with fixed extensions of regions and distances. Such a scenario can be formally realised by using object and region constants with a fixed interpretation (rather than variables)—see [47] for such a scenario applied on a city map.

Orientation. For representing relative orientation in recent years many different calculi have been presented, e.g., the DCC [26] and the Dipole Calculus [62]. Here, we apply the $\mathcal{OPRA}_m$ approach [56] because of its expressiveness. The calculi in this family are designed for reasoning about relative orientation relations between *oriented points* (points in the plane with an additional direction parameter), and they are well-suited for dealing with objects that have an intrinsic orientation. An oriented point $\vec{O}$ can be described by its Cartesian coordinates $x_O, y_O \in \mathbb{R}$ and a direction $\phi_{\vec{O}} \in [0, 2\pi)$ with respect to an absolute frame of reference. With the parameter m the angular resolution can be influenced, i.e., the number of base relations is determined.
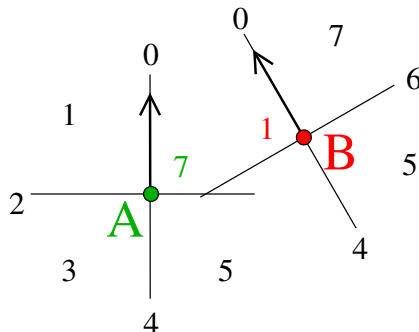

Fig. 7. The $\mathcal{OPRA}_2$ relation $\vec{A}\,_2\angle_7^1\,\vec{B}$.

In the case of $\mathcal{OPRA}_2$, the orientation calculus we apply in our examples, 2 lines are used for each pair of oriented points to partition the plane into 4 planar and 4 linear regions (see Figure 7). The orientation of the two points is depicted by the arrows starting at $\vec{A}$ and $\vec{B}$, respectively. The regions are numbered from 0 to 7, for which 0 is the intrinsic orientation of the point. An $\mathcal{OPRA}_2$ base relation is a pair (i, j),

where i is the number of the region, seen from $\vec{A}$, that contains $\vec{B}$ and j vice versa. These relations are written as $\vec{A}\,_2\angle_i^j\,\vec{B}$. Additional base relations describe situations in which both oriented points are at the same position but may have different orientations ($_m\angle i$).

Size, Shape, Morphology, and Spatial Change.   While the most important aspects of space are topology, orientation, and distance, other aspects of space include size, shape, morphology and spatial change (over time) [61]. For the purpose of this paper, we use spatial logics primarily for topology, orientation and distance, as they are mostly sufficient for our modelling examples. Here, theories formulated in a suitable spatial logic figure as high-level constraints for the respective spatial aspect. Moreover, ontological classifications of shapes are a relatively new and unexplored area, and are thus omitted; we refer to [34] for a first-order axiomatisation of 'shapes' in the engineering domain and to [36] for shape specifications combining ontology and topology in the domain of chemical molecules.

In the next section, we present in some detail the theoretical foundations for multi-perspective modelling using $\mathcal{E}$-connected spatial logics and conceptual ontologies.

5. $\mathcal{E}$-Connecting Conceptual and Spatial Dimensions

Heterogeneous knowledge representation is a major motivation also for the design of 'modular ontology languages', such as distributed description logics (DDLs) [15] and $\mathcal{E}$-connections [47]. Here, we concentrate on the latter. $\mathcal{E}$-connections were originally conceived as a versatile and computationally 'tame' technique for combining logics, but were subsequently also adopted as a framework for the combination of ontologies in the Semantic Web [22].

The general idea behind this combination method is that the interpretation domains of the connected logics are interpreted by disjoint (or sorted) vocabulary and interconnected by means of *link relations*. The language of the $\mathcal{E}$-connection is the union of the original languages enriched with operators capable of talking about the link relations. Just as DLs themselves, $\mathcal{E}$-connections offer an appealing compromise between expressive power and computational complexity: although powerful enough to express many interesting concepts, the coupling between the combined logics is sufficiently loose for proving general results about the transfer of decidability. But as follows from the complexity results of [47], $\mathcal{E}$-connections in general add substantial expressivity and interaction to the components. Here, the transfer of decidability as well as the expressiveness of the obtained $\mathcal{E}$-connection depend not only on the component logics but, essentially, on the employed connecting link language.

In $\mathcal{E}$-connections [47] a finite number of formalisms that typically talk about distinct domains or distinct views on the same domain are *connected* by relations between entities in the different domains. This allows us to capture different aspects or representations of the 'same object'. For instance, the following link relations are relevant for the logics that we employ to model architectural design, namely logics for topology, distance and orientation that we introduced in Section 4: (1) a concept C of a conceptual module specified in a DL_1 related via a link relation to a corresponding concept in another conceptual module specified in another DL_2 (see Example 4 and Figure 8); (2) an 'abstract' object o of a description logic DL_1 related via a relation R to its spatial extension in a spatial logic such as RCC-8, i.e., to a regular closed set of points in a topological space (see Example 5); (3) two points a and b of a description logic related via a link relation *has_orientation* to two oriented points in a model of $\mathcal{OPRA}_2$. For this to work coherently, we need to make link relations functional, which can be achieved by adding number restrictions on links (see below).

Essentially, the language of an $\mathcal{E}$-connection is the (disjoint) union of the original languages enriched with operators capable of specifying the link relations. The possibility of having multiple relations between
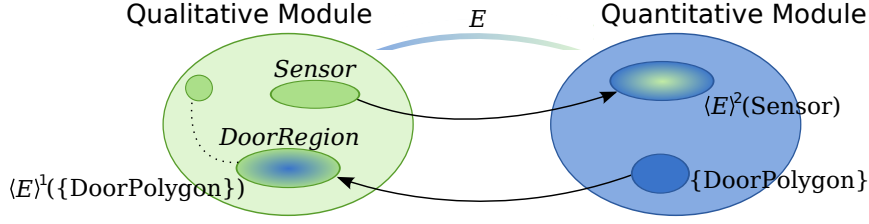
Fig. 8. A two-dimensional $\mathcal{E}$-Connection with examples from the qualitative and quantitative modules.

domains is essential for the versatility of this framework, the expressiveness of which can be varied by allowing different language constructs to be applied to the connecting relations. Figure 8 displays an example of the connection of two ontologies, with a single link relation E.

We first sketch the formal definitions for the 2-dimensional case. The reader is referred to [47] for involved examples and technical results on the computational properties of various specific $\mathcal{E}$-connections. To formulate a 2-dimensional $\mathcal{E}$-connection between two ontologies O_1 and O_2 formulated e.g., in two different DLs DL_1 and DL_2 (here, an ontology is a set of axioms in the respective DL), we assume that the **signatures** $\mathcal{L}_1 = \mathsf{Sig}(DL_1)$ and $\mathcal{L}_2 = \mathsf{Sig}(DL_2)$ of the two DLs, i.e., their sets of atomic concepts, roles, and object names, are pairwise disjoint.

To form a connection $\mathcal{C}^{\mathcal{E}}(DL_1, DL_2)$, fix a non-empty set $\mathcal{E} = \{E_j \mid j \in J\}$ of binary relation symbols. The **basic $\mathcal{E}$-connection**, then, has as signature the disjoint union of $\mathcal{L}_1, \mathcal{L}_2$ and $\mathcal{E}$; its concept language is two-sorted with sorts s_1 and s_2 defined by simultaneous induction as follows.

- (i) If C is a concept in DL_1, then C is of sort s_1; (ii) if D is of sort s_2 then $\langle E_j \rangle^1 D$ is of sort s_1; (iii) s_1 is closed under the concept-forming operations of DL_1.
- (i) If D is a concept in DL_2, then D is of sort s_2; (ii) if C is of sort s_1 then $\langle E_j \rangle^2 C$ is of sort s_2; (iii) s_2 is closed under the concept-forming operations of DL_2.

Here, the $\mathcal{E}$-connection-operators $\langle E_j \rangle^1$ and $\langle E_j \rangle^2$ are new concept-formation operators that are interpreted as the added link relations. The formal semantics for the class of models of $\mathcal{C}^{\mathcal{E}}(DL_1, DL_2)$ comprises all structures of the form

$$\mathfrak{M} = \left\langle \mathfrak{W}_1, \mathfrak{W}_2, \mathcal{E}^{\mathfrak{M}} = (E_j^{\mathfrak{M}})_{j \in J} \right\rangle,$$

where $\mathfrak{W}_i = (W_i, \cdot^{\mathfrak{W}_i})$ is an interpretation for DL_i for $i \in \{1, 2\}$ and $E_j^{\mathfrak{M}} \subseteq W_1 \times W_2$ for each $j \in J$. Given concepts C_i of ontology DL_i, for $i = 1, 2$, denoting subsets of W_i, the semantics of the basic $\mathcal{E}$-connection operators is

$$(\langle E_j \rangle^1 C_2)^{\mathfrak{M}} = \{x \in W_1 \mid \exists y \in C_2^{\mathfrak{M}} \ (x, y) \in E_j^{\mathfrak{M}}\}$$

$$(\langle E_j \rangle^2 C_1)^{\mathfrak{M}} = \{x \in W_2 \mid \exists y \in C_1^{\mathfrak{M}} \ (x, y) \in E_j^{\mathfrak{M}}\}$$

Note, the requirement of disjoint domains is not essential for the expressivity of $\mathcal{E}$-connections. What is essential, however, is the disjointness of the *formal languages* of the component logics. What this boils down to is the following simple fact: although more expressive $\mathcal{E}$-connection languages allow to express various degrees of *qualitative* identity, e.g., by using number restrictions on links to establish partial bijections (which we use below), they lack means to express 'proper' *numerical* trans-module identity.

It remains to clarify what the **sentences** of a basic $\mathcal{E}$-connection are. They follow the same grammar as the component logics (in the case of DLs concept subsumptions, ABox and RBox statements) but respect

the enriched concept language with the obvious semantics interpreted in the local models. Moreover, we have ABox-like sentences for the link relations:

$$\mathfrak{M} \models (a,b) : E_j \iff E_j{}^{\mathfrak{M}}(a^{\mathfrak{M}}, b^{\mathfrak{M}}).$$

Figure 8 displays the connection of two ontologies by means of a single link relation E. Here, the concept $\langle E \rangle^1(\{a\})$ of O_1 'corresponds' to the nominal $\{a\}$ of ontology O_2: it collects the set of all those points in O_1 that 'can be seen' from a (in O_2) along the relation E.

Example 4 (Connecting two ontologies) *Suppose two ontologies O_1 and O_2, formulated in different DLs DL_1 and DL_2, contain the concept* Window. *Ontology O_1 formalises functionalities of objects found in buildings, and ontology O_2 formalises the properties of materials of such objects. The intended relation between the two instances of* Window *might be one of* polysemy *(meaning variation), i.e.,* Window *in O_1 involves 'something with views that can be open or closed':*

$$Window \sqsubseteq \exists has\_state.(Open \sqcup Closed) \sqcap \exists offers.Views,$$

whilst the meaning of Window *in O_2 might be 'something that has a bulletproof glass':*

$$Window \equiv Glass \sqcap \exists has\_feature.Bulletproof.$$

*A systematic integration of these two ontologies could now require a mapping of objects in O_1 to the material they are made from, using a link relation '*consists_of*'. A concept of the form $\langle$consists_of$\rangle^1 C$ then collects all objects of O_1 that are made from something in C, and a concept $\langle$consists_of$\rangle^2 D$ collects the materials in O_2 some object in D consists of. A sensible alignment between the two instances of* Window, *introducing disjoint vocabulary* Window$_1$ *and* Window$_2$, *could now be formalised in $\mathcal{E}$-connections as:*

$$\langle consists\_of \rangle^2 Window_1 \sqsubseteq \exists has\_feature.Transparent$$
$$\langle consists\_of \rangle^1 Window_2 \sqsubseteq Window_1 \sqcap \exists provides\_security.Inhabitant$$

assuming that windows in O_1 might also be made of plastic, paper, etc. $\dashv$

Although this example is heterogeneous in the sense that two different description logics are involved, both dimensions in this $\mathcal{E}$-connection still conform to the same semantic paradigm. More interesting heterogeneous $\mathcal{E}$-connections are obtained when mixing logics with different 'reasoning modes', e.g., when combining conceptual with spatial reasoning, which is necessary for modelling architectural design as sketched in the next example.

Example 5 (Modelling Architectural Design) *We sketch an example of the use of $\mathcal{E}$-connections for modelling architectural design involving both conceptual and spatial dimensions: extending the previous example, let us suppose that we have a third dimension, a knowledge base formalised in the Region Connection Calculus RCC-8 (as encoded in the modal logic $\mathbf{S4_u}$ that we have introduced on Page 13). The following constraint is taken from [39] and illustrates the kind of modelling that can be performed in this setup.*

> *[. . .] sensors have to cover certain regions around doors. These are functional regions that are defined by the doors and instantiated by a qualitative ontology module. The region of the sensor range has to be an inverse proper part of this functional region.*
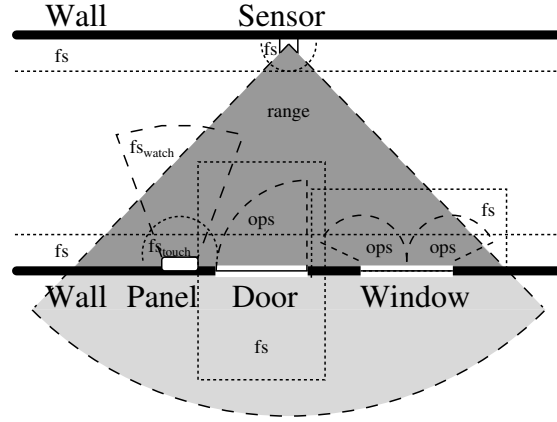
Fig. 9. Spatial extensions of ontology terms in $\mathbb{R}^2$. (Source: [5])

Here, 'door' and 'sensor' are taken as concepts Door, Sensor *that live in ontology O_1 introduced above. Moreover, we introduce two new relations bridging ontology O_1 and the RCC-8 domain, namely* has_functional_space *that relates the instances of* Door *with their functional space, i.e., regions in RCC-8, and* has_range_space, *again giving the regions covered by the sensors (see Figure 9).*

Here, models for an $\mathcal{E}$-connection of ontology O_1 and RCC-8 are of the form

$$\mathfrak{M} = \left\langle \mathfrak{W}_1, \mathfrak{W}_2, \mathcal{E}^{\mathfrak{M}} = (\textit{has\_range\_space}^{\mathfrak{M}}, \textit{has\_functional\_space}^{\mathfrak{M}}) \right\rangle,$$

where $\mathfrak{W}_1$ interprets ontology O_1, $\mathfrak{W}_2$ interprets RCC-8, and the link relations are interpreted as subsets of the Cartesian products of the domains of $\mathfrak{W}_1, \mathfrak{W}_2$. The constraint can be formalised:[13]

$$PP^{-1}(\langle \textit{has\_range\_space} \rangle^3 \, \textit{Sensor}, \langle \textit{has\_functional\_space} \rangle^3 \, \textit{Door})$$

Here $\langle$ has_range_space $\rangle^3$ Sensor defines a region by collecting, for a given model $\mathfrak{M}$ of the $\mathcal{E}$-connection, all points in the RCC-8 model that are 'connected' by the role has_range_space *to an element of the concept* Sensor. *Note that this enforces the constraint "The region of the sensor range has to be an inverse proper part of this functional region." in the following sense: the 'sum' of all functional spaces of all doors are covered by the 'sum' of all range spaces of all sensors. This suffices to guarantee coverage of the functional spaces of doors; however, it allows for non-standard, possibly unintended interpretations where several sensors are needed to cover one particular functional space of a door. To strengthen this constraint, we can pick, for every named instance* d : Door *a named sensor* s_d : Sensor *and postulate, for every such* d:

$$PP^{-1}(\langle \textit{has\_range\_space} \rangle^3 \, s_d, \langle \textit{has\_functional\_space} \rangle^3 \, d)$$

To ensure that the regions thus obtained are 'well-behaved', we might want to enforce that they are regular-closed sets. This is a typical assumption in RCC-8 based reasoning and can be formulated in $\mathcal{E}$-connections as follows:

$$\langle \textit{has\_range\_space} \rangle^3 \, \textit{Sensor} = \mathbf{CI} \, \langle \textit{has\_range\_space} \rangle^3 \, \textit{Sensor}$$

using the closure operator $\mathbf{C}$ and interior operator $\mathbf{I}$ of $\mathbf{S4_u}$. As above, such constraints can be strengthened by postulating them for every named instance, in this case of Sensor. ⊣

---

[13] PP^{-1} is the abbreviation for the union of TPP^{-1} and $NTPP^{-1}$.

Dedicated reasoning support for general $\mathcal{E}$-connections is not available at the moment. However, $\mathcal{E}$-connecting just ontological modules given in OWL, reasoning support can still be realised by a complete encoding of the semantics of $\mathcal{E}$-connections into OWL DL as follows (compare also [21]): (1) disjointness of thematically different domains is enforced by introducing new 'local' top concepts for each ontology (2) domain and range of link relations are accordingly restricted; (3) as $\mathcal{E}$-connection operators we can use DL's existential and universal restrictions for these link relations. Figure 10 below (Page 24) illustrates how the different architectural perspectives are $\mathcal{E}$-connected with each other. In the case of $\mathcal{E}$-connections connecting conceptual and spatial domains, although specialised implementations are not available, reasoning support can still be obtained by encoding the $\mathcal{E}$-connections as heterogeneous theories in the sense sketched above in Section 3 and by *borrowing* a reasonably expressive reasoner (see [57] for discussion). In the case of combining OWL ontologies with RCC-8, the encoding can still be done within OWL, so we can use OWL reasoners rather than first-order logic theorem provers.

More involved $\mathcal{E}$-connection modelling using the other spatial logics introduced in Section 4 is shown in the exploratory study that we turn to next.

6. Modular Representation for Structural Form and Function: An Exploratory Study

In Section 3.2, the different types of spatial information that can be distinguished for modelling the domain of architectural design have been introduced. In particular, we identified three main spatial perspectives [5, 39] that contribute to the overall architectural representation, namely the conceptual, qualitative and quantitative spatial perspective (cf. Figure 2). To achieve a complete ontological specification for an architectural design, ontological modules from these three perspectives have to be instantiated and connected with each other. Each module individually provides information about domain-specific, qualitative or quantitative aspects of the architectural design. For instance, a module for the conceptual perspective may specify courtroom-specific information about witness box, jury box and judges' bench, whereas a module for the quantitative perspective may specify construction-related information about size and position of walls, benches and seats. Hence, these modules define the concept- and form-related criteria of the design. Connecting the individual modules with each other, as demonstrated in Section 5, provides the specification of architectural functions and thus design-specific requirements. For example, the 'Witness-Box Placement'-requirement introduced in Section 2.2 is modelled by connecting the courtroom module with the spatial distance module by specifying constraints on the distance properties in the distance module of the connected witness box instances in the courtroom module.

As a result, the specifications for conceptual and spatial information are kept separate in the ontological modelling framework, whilst the connection of these representations provides the specification of functional requirements. This reflects not only the actual difference of conceptual and spatial information types but it also helps in specifying form and function and their relationship in a more appropriate way. Section 4 has introduced in detail various formalisms available for relating and combining different ontological specifications on a theoretical level. The main aim in our approach is to (1) ontologically specify the different spatial modules necessary for architectural design specifically for the interplay between form and function and (2) apply modular and heterogeneous specification formalisms to reflect the different perspectives appropriately also to allow intelligible re-use and application of these modular ontologies.

The following exploratory study presents our ontological formalisation for the different spatial criteria that contribute to an architectural design. It also explicates the modelling of architectural requirements by specifying connections and constraints between different conceptual and spatial modules. The ontological formalisation demonstrates particularly how the building code requirements for courthouses are represented.

6.1. Heterogeneous Modules for Spatial and Conceptual Information

We introduce the ontology modules developed for modelling architectural design that comply with the thematically different perspectives for conceptual, qualitative and quantitative space according to the framework in Section 3. The connection and relationships among the modules are illustrated in Figure 11.

Ontological Modules for the Conceptual Perspective. Modules complying with the *conceptual perspective* ($M1$) represent entities related to architectural design on the basis of their entity-based characteristics, i.e., they are specified by their properties and axioms without any contextual or spatially embedded aspects. Modules for the conceptual perspective can extend existing foundational or general domain-specific ontologies. This can technically be done by importing and re-using (i.e., conservatively extending) the existing ontologies and refining their categories and relations if necessary. We developed a modular ontology for this perspective that builds on and extends a foundational ontology, which provides an abstract foundation for specifying specific domain entities and relations, namely DOLCE [53]. In particular, we refine its OWL version DOLCE-Lite[14] to provide a categorisation of architectural entities. The resulting ontology refines primarily physical-object and non-physical-endurant of DOLCE-Lite. It introduces physical entities, e.g., PhysicalBuilding, Entrance and WitnessBox, as well as functional non-physical endurants, e.g., the building types Courthouse and Museum and the room types Office and Lobby. For instance, the building types are related to a PhysicalBuilding in the following way (omitting the inferred constraints from inherited DOLCE-LIte categories):

$$BuildingType \sqsubseteq dolceLite{:}non\text{-}physical\text{-}endurant$$
$$\sqcap \, \exists \, dolceLite{:}generically\text{-}dependent\text{-}on \, . \, PhysicalBuilding$$

This class construction specifies that types of buildings, e.g., courthouse, museum or university, are non-physical endurants that depend on an actual (physical) building, which provides the building's type or function. This formalisation is inspired by the modelling of artefacts and roles introduced in [78], in which an entity can be classified as a specific type of a physical endurant (e.g., a *pebble* or a *physical room*) although its artefact may be classified by some other type (e.g., a *paperweight* or a *courthouse* respectively). The namespace 'dolceLite' in the formula indicates the parts that are re-used from the foundational ontology, i.e., the relations non-physical-endurant and generically-dependent-on are specified in the DOLCE-Lite ontology. Parts without a namespace, namely BuildingType and PhysicalBuilding, are specified in the conceptual module that refines DOLCE-Lite. While the categories specified in the architectural entities module can be related to an intended architectural design, modules complying with the conceptual perspective can also include even more abstract types of information, e.g., about costs, environment, user groups or actions, which can be modelled by respective ontology modules [38].

Whilst the category Courtroom is specified by the conceptual perspective module as a physical room type in a courthouse building, its actual qualitative spatial properties and quantitative extent are specified by using connections to the qualitative and quantitative perspective modules. An example of this is illustrated in Figure 10.

Ontological Modules for the Qualitative Perspective. Modules complying with the *qualitative perspective* ($M2$) represent entities related to architectural design on the basis of their qualitative spatial characteristics. For instance, modules for this perspective specify spatial entities with regard to topology, distance, orientation or other qualitative spatial information (discussed in Section 4). In particular, spatial logics or calculi (introduced in Section 4.2) are spatial modules of this perspective, and we use their formalisations primarily to specify qualitative relations and constraints of an architectural design. Modules of this layer are consequently closely connected with modules from the conceptual perspective.

---

[14]DOLCE-Lite: http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl

Information about qualitative spatial relationships are needed to model architectural design not only to define basic relations among entities in an architectural design, e.g., doors are connected with walls or windows, but also to define their functional requirements, e.g., staircase landings are not supposed to overlap with functional regions of doors. Analysing if an architectural design satisfies given requirements can thus be achieved by defining restrictions on qualitative spatial models of an architectural design. The staircase landing example can be formalised by requiring the regional extension of the landing to be (RCC-)disconnected with the regional extension of the functional areas of doors (cf. formulas in Section 6.2). This modelling then uses $\mathcal{E}$-connections to formalise constraints between modules from the conceptual and qualitative perspectives.

The basic requirement that doors in an architectural design are necessarily connected with either walls, windows or doors, is formalised by constraints in the spatial extension of these concepts. This requirement is related to region-based information thus the region-based extension of any door needs to be (RCC-)externally connected with walls, windows or doors. The following formula defines this particular requirement in the $\mathcal{E}$-connected theory of a module from the conceptual and the qualitative perspectives. For clarity, we define the following regions in the qualitative module:

$$
\begin{aligned}
\mathsf{OpSpaceDoor} \quad &= \quad \langle \mathsf{has\_operational\_space} \rangle^{M2}\,\mathsf{Door} \\
\mathsf{OpSpaceWall} \quad &= \quad \langle \mathit{has\_operational\_space} \rangle^{M2}\,\mathsf{Wall} \\
\mathsf{OpSpaceWindow} \quad &= \quad \langle \mathit{has\_operational\_space} \rangle^{M2}\,\mathsf{Window}
\end{aligned}
$$

The index $M2$ indicates the qualitative module. Informally, the category OpSpaceDoor specifies the region that is composed by a door and its opening angle, the category OpSpaceWall specifies the region that is composed by the external boundary of a wall, and the category OpSpaceWindow specifies the region that is composed by a windows and its opening angle (cf. Figure 9). The following constraint encodes the requirement that all doors have to be externally connected with walls, windows or doors:

$$
EC(\mathsf{OpSpaceDoor}, \mathsf{OpSpaceDoor} \sqcup \mathsf{OpSpaceWall} \sqcup \mathsf{OpSpaceWindow})
$$

Such constraints can formulate basic requirements architectural designs have to satisfy. They primarily reflect the kinds of spatial relations that can be specified by using a spatial calculus. As shown above, legal issues and building codes require even more abstract and complex constraints on a particular design. While the basic constraints ensure that a building complies with primitive spatial or physical standards, the complex constraints ensure that a building conforms to particular design standards. Therefore, primitive kinds of constraints are formalised for the $\mathcal{E}$-connected conceptual and qualitative perspectives, complex kinds of constraints are formalised by an additional perspective for particular building code requirements.

For example, for the witness box requirement introduced in Section 2.2, several spatial qualitative constraints have to be satisfied by a design: qualitative distance information is indicated by the closeness between witnesses and court participants; qualitative orientation information is indicated by facial direction of witnesses; qualitative region-based information is implied as witnesses and participants have to be located in the same courtroom. Analysing if an architectural design satisfies given requirements can thus be achieved by defining restrictions on qualitative spatial models of an architectural design. The witness box example can be formalised by requiring that the regional extensions of witness and participants be (RCC-)proper parts of the courtroom, that the intrinsic orientations of witness and participants are oriented towards each other, and that the distance between witness and participants is in close proximity (cf. formulas in Section 6.2). Technically, the modelling of qualitative spatial extensions of conceptual entities applies $\mathcal{E}$-connections as defined above to formalise constraints between modules from the conceptual and qualitative perspectives.

Proving if an architectural design satisfies a set of given requirements, which are formulated in a logical module or an $\mathcal{E}$-connected theory, is based on ontological reasoning, in particular, ABox reasoning for the

conceptual modules and spatial reasoning for the qualitative modules. The first is achieved by standard DL ABox consistency checks, the second is achieved by using spatial model checking with specialised spatial reasoners, e.g., the constraint logic programming based declarative spatial reasoner CLP(QS) [11] or the integrated spatio-terminological approach in [5]. If no constraints are violated, the architectural design (instantiated as an ABox) complies with the given requirements.

Ontological Modules for the Quantitative Perspective. Modules complying with the *quantitative perspective* ($M3$) represent entities related to architectural design on the basis of their metrical information. They particularly reflect metrical data of construction elements in building plans, e.g., heights of ceilings, positions of walls, widths and heights of windows or opening angles of doors. Modules of this perspective are closely related to standards and tools for architectural design. We developed a quantitative module that resembles such a standardised architectural design format.

In previous work [5, 39], we applied the Industry Foundation Classes (IFC) [27], which is a data model for architectural design and design tools. It aims at interoperability in the building industry by providing a non-proprietary data exchange format that reflects constructional information about buildings. IFC can be related to 3D CAD models, though its data model is more expressive than CAD. IFC defines not only geometric primitives, such as points, lines and polygons, and raw metrical data about these entities, it also defines primitive semantics by relating them to structural elements: it defines concrete building components like walls, windows or roofs as well as abstract entities like actions, spaces or costs. As the data format is supported by commercial as well as free software design tools, which also allow exports into other XML and binary formats, for modelling, visualising, or syntax checking, its applicability is guaranteed.

In particular, as we use IFC in our approach, datasets from other IFC compliant design tool can easily be used and integrated. IFC specifies different types of *building entities* that provide a basis for an ontology module of the quantitative perspective. This module resembles relevant IFC classes necessary for formulating functional requirements of a design. IFC provides the different architectural entities of a design and their basic properties. For instance, a door in IFC is defined as *IfcDoor* that has a width, a height and an opening direction. This information and more is encoded accordingly in the ontological module of the quantitative perspective as follows:

$$\text{Door} \sqsubseteq \text{StructuralBuildingElement} \sqcap = 1 \text{ openingAngle . float}$$
$$\sqcap = 1 \text{ doorknobType . ENTITY} \sqcap = 1 \text{ height . float}$$
$$\sqcap = 1 \text{ length . float} \sqcap = 1 \text{ width . float}$$

Modules complying with the quantitative perspective are primarily related and linked to modules with the qualitative perspective. They can metrically ground the qualitative spatial relations between entities. Furthermore, the entities defined by quantitative modules also relate to information from the conceptual perspective (an example is illustrated in Figure 10). As for the complex requirement in the witness box example, the individual quantitative elements are specified by the ontological module from the quantitative perspective. Metrical information about the overall courtroom design, which is illustrated in Figure 4, is thus available by the respective IFC model used in CAAD tools.

Connecting Modular Ontologies for Architectural Design. The three thematically different perspectives for architectural design (conceptual, qualitative and quantitative) are connected with each other in order to formulate their correspondences and define basic architectural requirements and functions, namely client specifications and expert knowledge (see Section 2.2). As the modules represent aspects and elements that model different parts of an architectural design, we can identify equivalent or related entities that can be connected with each other. Conceptual entities can be interpreted in terms of their qualitative relations and quantitative extent, qualitative spatial relations can be inferred from quantitative data and related to conceptual information, quantitative spatial aspects provide the metrical information of the architectural design that is equipped with abstract conceptual elements and that indicates qualitative spatial

properties. Thus, the connection between the different spatial perspectives results in a three-dimensional
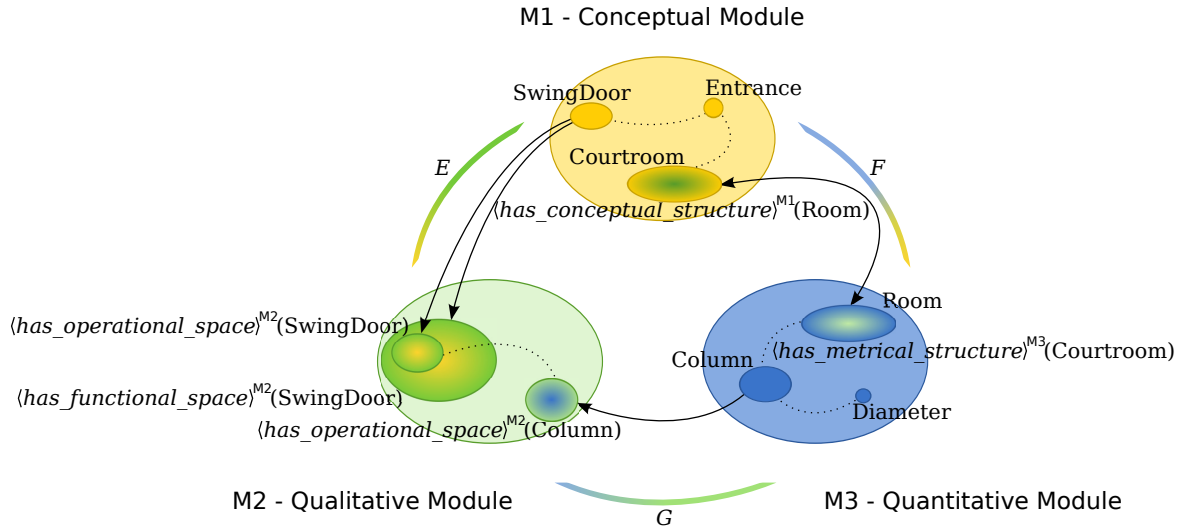$\mathcal{E}$-Connection, illustrated in Figure 10.

Fig. 10. A 3-Dimensional $\mathcal{E}$-connection, relating the conceptual, qualitative, and quantitative modules.

The figure shows an example part of an architectural model of a design. The links between the modules identify basic relationships among entities and their connections across modules. In the conceptual module, a particular type of a door (a swing door) is related to an entrance point that refers to the place that can be entered by the door. In the example, the swing door is the entrance to a courtroom. The qualitative region that spatially extends the swing door is defined in the qualitative module. This entity is related by an $\mathcal{E}$-Connection between the modules. As introduced in Section 5, the link relation has_operational_space connects the swing door with a region that reflects its operational space and the link relation has_functional_space connects the door with a region that reflects its functional space. The indices in Figure 10 indicate which modules are related by the respective link relations. Within the qualitative module, regions can spatially be related with each other. For instance, the operational space of the door is a proper part of the functional space of the door and it is disjoint with another region that is related to a column. In the example, the column is given by a quantitative model of the architectural design, which is linked from the quantitative model to the qualitative module. The quantitatively specified column is based on its metrical information, e.g., its diameter. Its position is also located in a room (the courtroom) in the architectural floor plan, and the room is connected with the conceptual courtroom by an $\mathcal{E}$-Connection between the quantitative and the conceptual perspective. The link relation F between these modules are specified as being (logically) functional (cf. Table 2), as each concept only refers to exactly one metrical entity, and similarly, each metrical entity (a polygon, line, etc.) in the floor plan refers to only one conceptual entity (a specific door, window, etc.).

The $\mathcal{E}$-Connections E, F, G depicted in Figure 10 between the different perspectives are used to link the various types of information for a complete representation of an architectural design. Whilst the example shows the use of the link relations for some basic spatial requirements between the three modules, the next section presents specific functional requirements of specific building code standards that were introduced in Section 2.

6.2. Modelling Functional Requirements in Architectural Design with Spatial Ontology Modules

Architectural design requirements are usually given by a set of (natural language) descriptions and by describing their functional specifications. In our ontological modelling of architectural design, these design requirements are specified by ontological modules from the *requirement perspective* that is based on the

three-level formalisation for conceptual, qualitative, and quantitative spatial information. These requirement modules enhance and refine the specification among components that are more specific and complex than the primitive building constraints. Hence, they technically extend the $\mathcal{E}$-Connection theory, illustrated in Figure 11. The extensions define particular requirements or constraints that have to be satisfied by an architectural design, e.g., to fulfil legal requirements or building codes.

Fig. 11. Functional requirements of a particular architectural design (e.g., a courthouse design guide) can be defined by (conservatively) extending the 3-dimensional formalisation for conceptual, qualitative and quantitative spatial information. Each perspective can consist of several logical theories or modular ontologies, e.g., the qualitative perspective consists of spatial logics and calculi for regions, orientations and distances.

The German building code referred to in Section 2 is one example of such a requirement module that specifies building functionalities and constraints, e.g., the landing example. As presented above, one of its official regulations is the following:

> "*Steps of a staircase may not be connected directly to a door that opens towards the steps. There has to be a landing between the staircase steps and the door. The length of this landing has to have at least the size of the door width.*" (Bremen (Germany) Building code [16])

Given this (natural language) constraint, its interpretation in conceptual, qualitative and quantitative spatial categories has to be specified in the related building code ontology module of the requirements perspective for the respective building code. It is primarily reflected by region-based constraints on the entities in the qualitative module. These entities are related to their qualitative counterparts describing staircases and landings in the quantitative module, i.e., in the actual floor plan. These entities may also be connected with conceptual elements that reflect particular aspects of the staircase, such as material or tread types.

The landing requirement is specified by the ontological requirement that all operational spaces of staircase landings should not overlap with operational spaces of doors (cf. the definition of operational spaces in the qualitative perspective module). The requirement constrains the link relation between the quantitative and the qualitative module and is formalised in the building code module in the following way:

$$\text{OpSpaceStaircaseLanding} = \langle \textit{has\_operational\_space} \rangle^{M1} (\text{Landing} \sqcap \exists \text{landingOf.Staircase})$$
$$\text{OpSpaceDoor} = \langle \textit{has\_operational\_space} \rangle^{M1} \text{Door}$$

$$DC(\text{OpSpaceStaircaseLanding}, \text{OpSpaceDoor})$$

Again for clarity, we introduced the regions OpSpaceStaircaseLanding (the operational space of a landing of a staircase) and OpSpaceDoor (the operational space of a door) in the qualitative module. The index $M1$ indicates that the categories Landing, Staircase and Door are defined in the quantitative module. These regions have to be disjoint with each other, which models the building code requirement. A given architectural design can be proven to satisfy this requirement by its spatial model consistency. This way, regulations from a building code can be specified in the requirement module. A design that satisfies these requirements consequently satisfies the building code.

The landing example basically constrains structural building aspects of a design (statutory requirements), however, requirements can be more complex and abstract (specialised requirements) as introduced in Section 2. The following example from the courthouse design guideline constrains the requirements of a witness box in a courthouse, which has been used throughout the examples for presenting the different perspectives:

> *"Witnesses must be able to see and hear, and be seen and heard by, all court participants as close to full face as possible. The witness box must accommodate one witness and an interpreter (...). Witnesses in the box receive, examine, and return exhibits." (US Courts Design Guide 2007 [76])*

This requirement contains conceptual information about witnesses, interpreters, judges and participants as particular persons or user groups and about seeing, hearing, receiving, examining and returning exhibits as specific actions. Spatial qualitative constraints require that certain persons are in close distance with each other, that they face each other, and that the witness box provides enough space for certain persons. Hence qualitative spatial relations about distance, orientation, and size or region have to be specified to provide the anticipated function of the respective architectural entities. The interrelations between the different spatial types of information for qualitative and conceptual aspects are specified in terms of $\mathcal{E}$-Connections between the ontological modules, which is formulated by a courthouse module in the requirements perspective.

The functional requirement of the witness box is specified in the requirement module for the Courthouse Design Guide as follows. The spatial calculus $\mathcal{OPRA}_2$ [56] is used for modelling orientations. The index $O1$ in the formulas encode the qualitative module for this orientation calculus. Participants and witnesses are here required to face each other at least within $90°$ to both sides. The spatial distance logic [45] is indexed with $D1$. The $\mathcal{E}$-connection links *has_orientation*, *has_distance_space* and *has_distance* are specified as functional in order to guarantee a reasonable semantics, i.e., to ensure that every point in the distance model (whenever defined) is related to exactly one *oriented point* in the orientation model, and vice versa.[15] Domain and range of the three $\mathcal{E}$-connections are also restricted, i.e., *has_orientation* relates conceptual entities from the requirement module to the oriented points in the orientation module, *has_distance_space* relates conceptual entities from the requirement module to points in the distance module, and *has_distance* relates integer nominals from the requirement module to distances in the distance module.

(1) "Witnesses must be able to see and hear, and be seen and heard by, all court participants as close to full face as possible"

For any named individual w classified as Witness and p classified as Participant in the ABox of an architectural design model that specifies the respective parts of a courtroom in a floor plan, the witness constraints are modelled as follows:[16]

$$\text{frontW} = \langle \textit{has\_orientation} \rangle^{O1} (\text{w})$$
$$\text{frontP} = \langle \textit{has\_orientation} \rangle^{O1} (\text{p})$$

---

[15]Technically, we need number restrictions on link relations in order to do this declaratively, see [47].

[16]In any given model, statements of the form $\mathcal{OPRA}_2 \angle_i^j(x, y)$, where $i, j \in \{0, \dots, 7\}$ and x, y are fixed oriented points, behave like Boolean variables, i.e. are either true or false. We can therefore without further complications allow Boolean operations on such atoms.
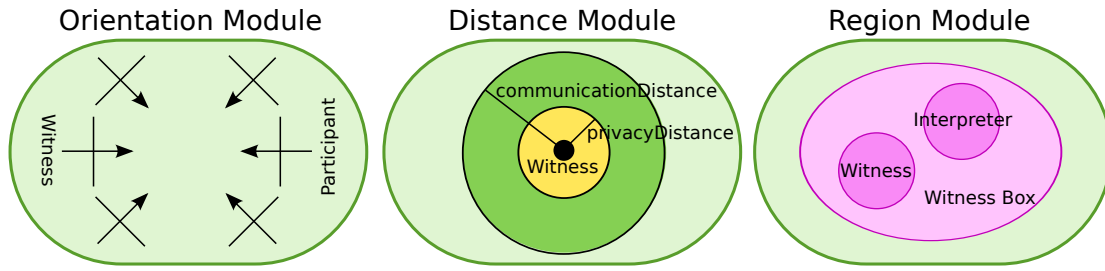
Fig. 12. Courtroom building requirements for different qualitative spatial models for orientation, distance and region.

$(\ \mathcal{OPRA}_2 \angle_0^0 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_2 \angle_0^1 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_2 \angle_0^7 \ (\text{frontW, frontP})$
$\sqcup \ \mathcal{OPRA}_2 \angle_1^0 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_2 \angle_1^1 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_1 \angle_7^0 \ (\text{frontW, frontP})$
$\sqcup \ \mathcal{OPRA}_2 \angle_7^0 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_7 \angle_1^0 \ (\text{frontW, frontP}) \sqcup \mathcal{OPRA}_2 \angle_7^7 \ (\text{frontW, frontP}) \ )$

posW	$=$	$\langle \textit{has\_distance\_space} \rangle^{D1} (\text{w})$
posP	$=$	$\langle \textit{has\_distance\_space} \rangle^{D1} (\text{p})$
pd	$=$	$\langle \textit{has\_distance} \rangle^{D1} \text{privacyDistanceValue}$
cd	$=$	$\langle \textit{has\_distance} \rangle^{D1} \text{communicationDistanceValue}$

$\text{posW} \rightarrow \text{A}_{<cd}^{>pd} \ \text{posP}$
$\text{posW} \rightarrow \text{A}_{<pd} \neg \ \text{posP}$

The region-based spatial calculus RCC-8, given by the index $R1$, is used to model the requirements that are interpreted in terms of region-related constraints. The $\mathcal{E}$-connection link *has_oprational_space* is specified as functional. In the witness box example, witnesses and Interpreters are supposed to be located inside the witness box:

(2) "The witness box must accommodate one witness and an interpreter, and the preferences of the presiding judge."

For any named individual w classified as Witness, i classified as Interpreter, and b classified as WitnessBox in the ABox of an architectural design model, that specifies the respective parts of a courtroom in a floor plan, the witness box constraints are modelled as follows:

opSpaceB $=$	$\langle \textit{has\_operational\_space} \rangle^{R1} (\text{b})$
opSpaceW $=$	$\langle \textit{has\_operational\_space} \rangle^{R1} (\text{w})$
opSpaceI $=$	$\langle \textit{has\_operational\_space} \rangle^{R1} (\text{i})$

$NTPP(\text{OpSpaceI} \sqcap \text{OpSpaceW}, \text{OpSpaceB})$

An illustration of the spatial constraints in the different qualitative modules is shown in Figure 12. Functional and informally defined design requirements described in the courthouse design guide are formally specified by using the three-dimensional representation of the different spatial perspectives. Individual requirements are interpreted in terms of their conceptual, qualitative and quantitative characteristics, and they are axiomatised in the requirement module particularly by using constraints on the connections between the conceptual, qualitative and quantitative modules. If an architectural design satisfies these (inter-)ontological requirements specified in the design guide module, it also satisfies the design guide, i.e., the ABox of a concrete design of a building, which can be instantiated automatically by using its CAAD data model, satisfies the constraints and class restrictions formulated by the TBox of the requirement module. Spatial model checking and ABox consistency support reasoning for these different modules. The overall

ontological framework that distinguishes between the different aspects of spatial information (conceptual, qualitative, quantitative, basic and complex requirements) thus allows a direct translation from (natural language) functional requirements in design guidelines to ontological modular specifications by re-using and extending the three-dimensional view on architectural design. The individual modules also support modular reasoning, particularly for the different spatial qualitative representations.

7. Discussion and Outlook

In this paper, we have focussed on the ontological formalisation of the structural form of spatial designs and high-level specification of functional design requirements. Our modelling approach aims to make the representation accessible for automated reasoning capabilities concerned with providing an analytical function during the initial design conceptualisation and iterative refinement phase, as identified in the context of *spatial computing for design* [9]. Here, spatial computing (for design) is defined as:

– *"the body of work that is concerned with the use of formal methods in knowledge representation and reasoning in general, and terminological and spatial representation and reasoning in specific, for solving problems in modelling (e.g., spatial semantics, modularity, requirement constraints) and validation (e.g., diagnosis, hypothetical reasoning) in the domain of spatial design"*

– *"that body of work whose aim is to develop the generic apparatus— application framework, methodology, tool-sets —that may be used as a basis of providing assistive design support within a conventional CAAD-based spatial design workflow"*

Situated within this AI-centred view of spatial computing for design, this paper has addressed the need to formally represent and reason about (structural) form and (artefactual) function. Specifically, the paper focussed on *semantic modelling, spatial abstraction* and *multi-perspective representation*. These aspects contribute to a conventional 'iterative refinement by automated design assistance' workflow, and are identifiable with respect to the *modelling–evaluation–re-design* phases in intelligent design assistance, for instance, as interpreted within the ontological framework of the Function-Behaviour-Structure (FBS) model [30, 31, 74, 75] of the design process. An overview of the design work-flow is illustrated in Figure 13: first, a work-in-progress design is modelled in an architectural design tool. Then, for every module, the data model formulated in a CAAD model is instantiated by the respective ontological modules. Given certain task-specific or functional requirements for the environment being modelled, spatio-terminological reasoning supported by different reasoning components can prove the consistency of the work-in-progress design. The results are incorporated in the iterative refinement phase, repeating this process until certain design objectives are satisfied, i.e., no requirement inconsistencies occur. A more detailed description of this work-flow and the design assistance are available in [9, 12].

In this paper, we have concentrated on the high-level modelling problems that are encountered in the domain of architectural design. We aimed at providing a solution for the following two major aspects in architectural design:

1. *Design Semantics.* The formal modelling of design requirements can be achieved by using modular ontologies for architectural design that specify the individual perspectives and guidelines. Combinations of modules determine functionality in terms of required spatial forms.
2. *Spatial Abstraction.* The modular representation also provides an abstraction from CAAD-based metrical information to qualitative spatial information and conceptual design requirements. Modular reasoning supports the use of ontological and spatial reasoners.

In order to account the need to incorporate heterogeneous multi-perspective representations in design, we have adopted the formal theory of $\mathcal{E}$-connections. This theory not only facilitates modularity and multi-perspective representations but also supports automatic reasoning by ensuring the decidability of the global
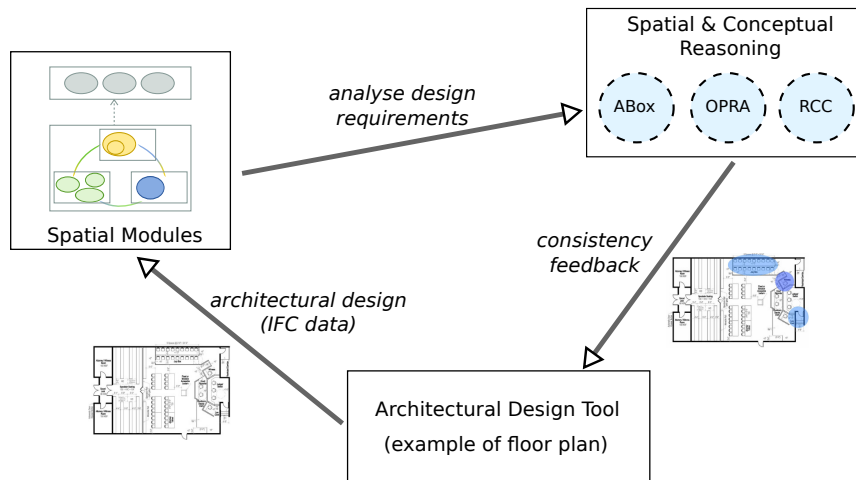
Fig. 13. Reasoning for Design Assistance

reasoning problems with diverse logical reasoning modes, such as reasoning over topological relations, distances, etc. As discussed in Section 3, modularity in ontological design has many facets. In [60], for instance, modularity is motivated by the complexities of designing large biomedical ontologies: they utilise normalisation—as in a database sense—as the underlying approach to achieve modularity for arbitrary DL ontologies, including OWL ontologies. Such an approach to modularity could potentially also serve as a basis of modularisation in the architectural domain if the ontological knowledge involved was expected to undergo continuous evolution, distributed design and development. In our case, however, the taxonomic knowledge is more or less static since the IFC is a de facto design standard and not subject to regular modifications. Nevertheless, comparing the different approaches to modularity and their impact on the reasoning capabilities is an important topic for further consideration. Non-classical inference, of course, is of particular interest in a domain such as architectural design, in particular *hypothetical reasoning* [9, 4] and *para-consistent inference* [72]. Reasoning support for $\mathcal{E}$-connections that combine DLs with RCC-8 is under active development, and the extension to non-classical reasoning problems remains a very promising direction for future research.

The work in [17, 18] presents a detailed formalisation and analysis of functions, the goals they may provide for an agent, and the roles they may provide for an entity. In a similar way, the architectural requirements discussed in this paper can be regarded as goals that should be achieved by an architectural design in order to provide a certain functionality. In comparison with existing foundational and upper level ontologies that describe functions, for instance, introduced in [37, 42, 55], the type of function we present in this paper is reflected closely by the concept of *requirement function* in those ontologies. Here, "function is a role played by a behaviour specified in a context" [55]. In the context of architectural design, however, we adopt a notion of functions that is strongly interpreted in terms of its possible qualitative and quantitative spatial models formalised by different perspectives on space and their connections. Hence, no upper-level ontology of functions was re-used explicitly in the conceptual modelling of architectural design. However, ontologically axiomatising the relationship between abstract or loosely defined requirements for architectural design and their modular formalisation across spatial perspectives is another important future direction.

Acknowledgements

References

[1] C. Alexander, S. Ishikawa, and M. Silverstein. *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, 1977.

[2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.

[3] B. Bennett. Modal logics for qualitative spatial reasoning. *Journal of the Interest Group on Pure and Applied Logic*, 4:23–45, 1996.

[4] M. Bhatt. Reasoning about space, actions and change: A paradigm for applications of spatial reasoning. In *Qualitative Spatial Representation and Reasoning: Trends and Future Directions*. IGI Global, USA, 2010.

[5] M. Bhatt, F. Dylla, and J. Hois. Spatio-Terminological Inference for the Design of Ambient Environments. In K. S. Hornsby, C. Claramunt, M. Denis, and G. Ligozat, editors, *Proceedings of the Conference on Spatial Information Theory*, pages 371–391. Springer, 2009.

[6] M. Bhatt, A. Flahive, C. Wouters, J. W. Rahayu, and D. Taniar. Move: A distributed framework for materialized ontology view extraction. *Algorithmica*, 45(3):457–481, 2006.

[7] M. Bhatt, A. Flahive, C. Wouters, J. W. Rahayu, D. Taniar, and T. S. Dillon. A distributed approach to sub-ontology extraction. In *AINA (1)*, pages 636–641. IEEE Computer Society, 2004.

[8] M. Bhatt and G. Flanagan. Spatio-temporal abduction for scenario and narrative completion. In Bhatt et al. [10], pages 31–36. in [10].

[9] M. Bhatt and C. Freksa. Spatial Computing for Design: An Artificial Intelligence Perspective. In *NSF International Workshop on Studying Visual and Spatial Reasoning for Design Creativity, Springer.*, Aix-en-Provence, France. 2010.

[10] M. Bhatt, H. Guesgen, and S. Hazarika, editors. *Spatio-Temporal Dynamics (STeDy 10)*. ECAI Workshop Proceedings; SFB/TR 8 Spatial Cognition Report Series, 2010.

[11] M. Bhatt, J. H. Lee, and C. Schultz. CLP(QS): A Declarative Spatial Reasoning Framework. In M. J. Egenhofer, N. A. Giudice, R. Moratz, and M. F. Worboys, editors, *COSIT*, volume 6899 of *Lecture Notes in Computer Science*, pages 210–230. Springer, 2011.

[12] M. Bhatt, C. Schultz, and C. Freksa. The 'Space' in Spatial Assistance Systems: Conception, Formalisation and Computation. In T. Tenbrink, J. Wiener, and C. Claramunt, editors, *Representing space in cognition: Interrelations of behavior, language, and formal models. Series: Explorations in Language and Space*. Oxford University Press, 2011. (forthcoming).

[13] M. Bhatt, C. Wouters, A. Flahive, J. W. Rahayu, and D. Taniar. Semantic completeness in sub-ontology extraction using distributed methods. In A. Laganà, M. L. Gavrilova, V. Kumar, Y. Mun, C. J. K. Tan, and O. Gervasi, editors, *ICCSA (3)*, volume 3045 of *Lecture Notes in Computer Science*, pages 508–517. Springer, 2004.

[14] A. Borgida. On the Relative Expressiveness of Description Logics and Predicate Logics. *Artificial Intelligence*, 82(1–2):353–367, 1996.

[15] A. Borgida and L. Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *Journal of Data Semantics*, 1:153–184, 2003.

[16] BremLBO. Bremische Landesbauordnung, 2003. `http://www.bauordnungen.de/html/bremen.html`.

[17] P. Burek. *Ontology of Functions. A Domain-independent Framework for Modeling Functions*. PhD thesis, Universität Leipzig, 2006.

[18] P. Burek, H. Herre, and F. Loebe. Ontological analysis of functional decomposition. In H. Fujita and V. Marík, editors, *New Trends in Software Methodologies, Tools and Techniques - Proceedings of the Eighth SoMeT*, volume 199 of *Frontiers in Artificial Intelligence and Applications*, pages 428–439. IOS Press, 2009.

[19] A. G. Cohn and S. Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 46(1-2):1–29, 2001.

[20] A. G. Cohn and J. Renz. Qualitative spatial reasoning. In F. van Harmelen, V. Lifschitz, and B. Porter, editors, *Handbook of Knowledge Representation*. Elsevier, 2007.

[21] B. Cuenca Grau, B. Parsia, and E. Sirin. Combining OWL ontologies using $\mathcal{E}$-Connections. *Journal Of Web Semantics*, 4(1):40–59, 2006.

[22] B. Cuenca Grau, B. Parsia, and E. Sirin. Ontology Integration Using $\mathcal{E}$-connections. In H. Stuckenschmidt and S. Spaccapietra, editors, *Modular Ontologies*, pages 293–320. Springer, 2009.

[23] M. de Rijke. The modal logic of inequality. *Journal of Symbolic Logic*, 57:566–584, 1990.

[24] J. Euzenat and P. Shvaiko. *Ontology matching*. Springer, 2007.

[25] C. Freksa. Qualitative spatial reasoning. In D. Mark and A. Frank, editors, *Cognitive and linguistic aspects of geographic space*, pages 361–372. Kluwer, 1991.

[26] C. Freksa. Using orientation information for qualitative spatial reasoning. In A. U. Frank, I. Campari, and U. Formentini, editors, *Theories and methods of spatio-temporal reasoning in geographic space*, pages 162–178. Springer, 1992.

[27] T. Froese, M. Fischer, F. Grobler, J. Ritzenthaler, K. Yu, S. Sutherland, S. Staub, B. Akinci, R. Akbas, B. Koo, A. Barron, and J. Kunz. Industry foundation classes for project management - a trial implementation. *ITCon*, 4:17–36, 1999. www. ifcwiki.org/.

[28] D. Gabbay, A. Kurucz, F. Wolter, and M. Zakharyaschev. *Many-Dimensional Modal Logics: Theory and Applications*. Number 148 in Studies in Logic and the Foundations of Mathematics. Elsevier Science Publishers, 2003.

[29] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening Ontologies with DOLCE. In *Proceedings of EKAW 2002*, number 2473 in LNCS, pages 166–181. Springer, 2002.

[30] J. S. Gero. Design prototypes: A knowledge representation schema for design. *AI Magazine*, 11(4):26–36, 1990.

[31] J. S. Gero, K. W. Tham, and H. S. Lee. Behavior: A link between function and structure in design. In D. C. Brown, M. B. Waldron, and H. Yoshikawa, editors, *IntCAD*, volume B-4 of *IFIP Transactions*, pages 193–225. North-Holland, 1991.

[32] J. A. Goguen and R. M. Burstall. Introducing institutions. In E. Clarke and D. Kozen, editors, *Proceedings of the Logics of Programming Workshop*, pages 221–256. Springer, 1984.

[33] V. Goranko and S. Passy. Using the universal modality: Gains and questions. *Journal of Logic and Computation*, 2(1):5–30, 1992.

[34] M. Grüninger and A. Delaval. A First-Order Cutting Process Ontology for Sheet Metal Parts. In *Proceedings of the 4th Workshop on Formal Ontologies Meet Industry*, volume 198 of *Frontiers in Artificial Intelligence and Applications*, pages 22–33. IOS Press, 2009.

[35] V. Haarslev and R. Möller. Description logic systems with concrete domains: Applications for the semantic web. In F. Bry, C. Lutz, U. Sattler, and M. Schoop, editors, *10th International Workshop on Knowledge Representation meets Databases*, volume 79. CEUR-WS.org, 2003.

[36] J. Hastings, O. Kutz, and T. Mossakowski. How to model the shapes of molecules? Combining topology and ontology using heterogeneous specifications. In *Proceedings of the Deep Knowledge Representation Challenge Workshop (DKR-11), K-CAP-11, Banff, Alberta, Canada*, 2011.

[37] H. Herre. General Formal Ontology (GFO) : A Foundational Ontology for Conceptual Modelling. In R. Poli and L. Obrst, editors, *Theory and Applications of Ontology*, volume 2. Springer, 2010.

[38] J. Hois. Modularizing spatial ontologies for assisted living systems. In Y. Bi and M.-A. Williams, editors, *Proceedings of the 4th International Conference on Knowledge Science, Engineering & Management*, pages 424–435. Springer, 2010.

[39] J. Hois, M. Bhatt, and O. Kutz. Modular Ontologies for Architectural Design. In *Proceedings of the 4th Workshop on Formal Ontologies Meet Industry*, volume 198 of *Frontiers in Artificial Intelligence and Applications*, pages 66–77. IOS Press, 2009.

[40] I. Horrocks, O. Kutz, and U. Sattler. The Even More Irresistible $\mathcal{SROIQ}$. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning*, pages 57–67. AAAI Press, June 2006.

[41] I. Horrocks, U. Sattler, and S. Tobies. Practical reasoning for expressive description logics. In H. G. an D. McAllester and A. Voronkov, editors, *Logic for Programming and Automated Reasoning, LPAR 99*, pages 161–180. Springer, 1999.

[42] Y. Kitamura. Roles of ontologies of engineering artifacts for design knowledge modeling. In *Proceedings of the 5th International Seminar and Workshop Engineering Design in Integrated Product Development*, pages 59–69, 2006.

[43] M. Knauff, R. Rauh, and C. Schlieder. Preferred mental models in qualitative spatial reasoning: A cognitive assessment of Allen's calculus. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*, 1995.

[44] R. Kontchakov, F. Wolter, and M. Zakharyaschev. Can You Tell the Difference Between DL-Lite Ontologies? In *11th International Conference on Knowledge Representation and Reasoning*, pages 285–295. AAAI, 2008.

[45] O. Kutz. Notes on Logics of Metric Spaces. *Studia Logica*, 85(1):75–104, 2007.

[46] O. Kutz, J. Hois, J. Bao, and B. Cuenca Grau, editors. *Modular Ontologies—Proceedings of the Fourth International Workshop*, volume 210 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2010.

[47] O. Kutz, C. Lutz, F. Wolter, and M. Zakharyaschev. $\mathcal{E}$-Connections of Abstract Description Systems. *Artificial Intelligence*, 156(1):1–73, 2004.

[48] O. Kutz and T. Mossakowski. A Modular Consistency Proof for Dolce. In *Twenty-Fifth Conference on Artificial Intelligence (AAAI-11)*. AAAI Press, 2011.

[49] O. Kutz, T. Mossakowski, and D. Lücke. Carnap, Goguen, and the Hyperontologies: Logical Pluralism and Heterogeneous Structuring in Ontology Design. *Logica Universalis*, 4(2), 2010. Special Issue on 'Is Logic Universal?'.

[50] O. Kutz, H. Sturm, N.-Y. Suzuki, F. Wolter, and M. Zakharyaschev. Axiomatizing distance logics. *Journal of Applied Non-Classical Logic*, 12:425–440, 2002.

[51] A. Loos. Ornament and Crime. *Innsbruck, Reprint Vienna*, 1930.

[52] C. Lutz and F. Wolter. Modal Logics of Topological Relations. *Logical Methods in Computer Science*, 2(2), 2006.

[53] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. Ontologies library. WonderWeb Deliverable D18, ISTC-CNR, 2003.

[54] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. WonderWeb Deliverable D18: Ontology Library. Technical report, ISTC-CNR, 2003.

[55] R. Mizoguchi. Functional ontology of artifacts. In *Proceedings of the First Interdisciplinary Ontology Meeting*, pages 51–60. Keio University Press, 2008.

[56] R. Moratz. Representing relative direction as a binary relation of oriented points. In G. Brewka, S. Coradeschi, A. Perini, and P. Traverso, editors, *ECAI'06*, pages 407–411. IOS Press, 2006.

[57] T. Mossakowski and O. Kutz. The Onto-Logical Translation Graph. In *Proceedings of the 5th International Workshop on Modular Ontologies (WoMO-11)*, Frontiers in Artificial Intelligence and Applications. IOS Press, 2011.

[58] T. Mossakowski, C. Maeder, and K. Lüttich. The Heterogeneous Tool Set. In O. Grumberg and M. Huth, editors, *TACAS 2007*, volume 4424 of *Lecture Notes in Computer Science*, pages 519–522. Springer, 2007.

[59] D. A. Randell, Z. Cui, and A. G. Cohn. A Spatial Logic Based on Regions and Connection. In *Proceedings of the 3rd International Conference on the Principles of Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann, 1992.

[60] A. L. Rector. Modularisation of domain ontologies implemented in description logics and related formalisms including OWL. In *Proceedings of the 2nd international conference on Knowledge capture*, pages 121–128. ACM, 2003.

[61] J. Renz and B. Nebel. Qualitative spatial reasoning using constraint calculi. In M. Aiello, I. Pratt-Hartmann, and J. van Benthem, editors, *Handbook of Spatial Logics*, pages 161–215. Springer, 2007.

[62] C. Schlieder. Reasoning about ordering. In *Proceedings of the Conference on Spatial Information Theory*, pages 341–349. Springer, 1995.

[63] M. Schorlemmer and Y. Kalfoglou. Institutionalising Ontology-Based Semantic Integration. *Journal of Applied Ontology*, 3(3), 2008.

[64] C. Schultz and M. Bhatt. A multi-modal data access framework for spatial assistance systems. In *Proceedings of the Second ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, In conjunction with ACM SIGSPATIAL GIS*, 2010.

[65] C. Schultz and M. Bhatt. Multi-Modal Spatial Data Access for Architecture Design Assistance. In *Proceedings of the Second ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, In conjunction with ACM SIGSPATIAL GIS* [64]. (in review; refer [64] for a short published version).

[66] V. Shehtman. "Everywhere" and "Here". *Journal of Applied Non-Classical Logic*, 9, 1999.

[67] M. Sheremet, D. Tishkovsky, F. Wolter, and M. Zakharyaschev. 'Closer' representation and reasoning. In *International Workshop on Description Logics*, pages 25–36, 2005.

[68] M. Sheremet, D. Tishkovsky, F. Wolter, and M. Zakharyaschev. A Logic for Concepts and Similarity. *Journal of Logic and Computation*, 17(3):415–452, 2007.

[69] M. Sheremet, F. Wolter, and M. Zakharyaschev. A modal logic framework for reasoning about comparative distances and topology. *Annals of Pure and Applied Logic*, 161(4):534–559, 2010.

[70] H. Stuckenschmidt, C. Parent, and S. Spaccapietra, editors. *Modular Ontologies - Concepts, Theories and Techniques for Knowledge Modularization*. Springer, 2009.

[71] L. Sullivan. The tall office building artistically considered. *Lippincott's Magazine*, 1896.

[72] H. Takeda, H. Sasaki, Y. Nomaguchi, M. Yoshioka, Y. Shimomura, and T. Tomiyama. Universal Abduction Studio—Proposal Of A Design Support Environment For Creative Thinking In Design. In *Proceedings of the International Conference On Engineering Design*, 2003.

[73] A. Tarski. Der Aussagenkalkül und die Topologie. *Fundamenta Mathematicae*, 31:103–134, 1938.

[74] Y. Umeda, H. Takeda, T. Tomiyama, and H. Yoshikawa. Function, Behavior and Structure. In *Applications of AI in Engineering (AIENG-90)*, pages 177–193. Computational Mechanic Publications and Springer, 1990.

[75] Y. Umeda and T. Tomiyama. Functional Reasoning in Design. *IEEE Expert: Intelligent Systems and Their Applications*, 12:42–48, 1997.

[76] US GSA. US Courts Design Guide, 2007. Judicial Conference of the United States. US General Services Administration (GSA). http://www.gsa.gov/graphics/pbs/Courts_Design_Guide_07.pdf.

[77] F. van Harmelen, V. Lifschitz, and B. Porter, editors. *Handbook of Knowledge Representation*. Foundations of Artificial Intelligence. Elsevier Science, 2008.

[78] L. Vieu, S. Borgo, and C. Masolo. Artefacts and roles: Modelling strategies in a multiplicative ontology. In C. Eschenbach and M. Grüninger, editors, *Formal Ontology in Information Systems*, pages 121–134. IOS Press, 2008.

[79] A. Zimmermann, M. Krötzsch, J. Euzenat, and P. Hitzler. Formalizing Ontology Alignment and its Operations with Category Theory. In B. Bennett and C. Fellbaum, editors, *Proceedings of the Fourth International Conference on Formal Ontology in Information Systems*, volume 150 of *Frontiers in Artificial Intelligence and Applications*, pages 277–288. IOS Press, 2006.