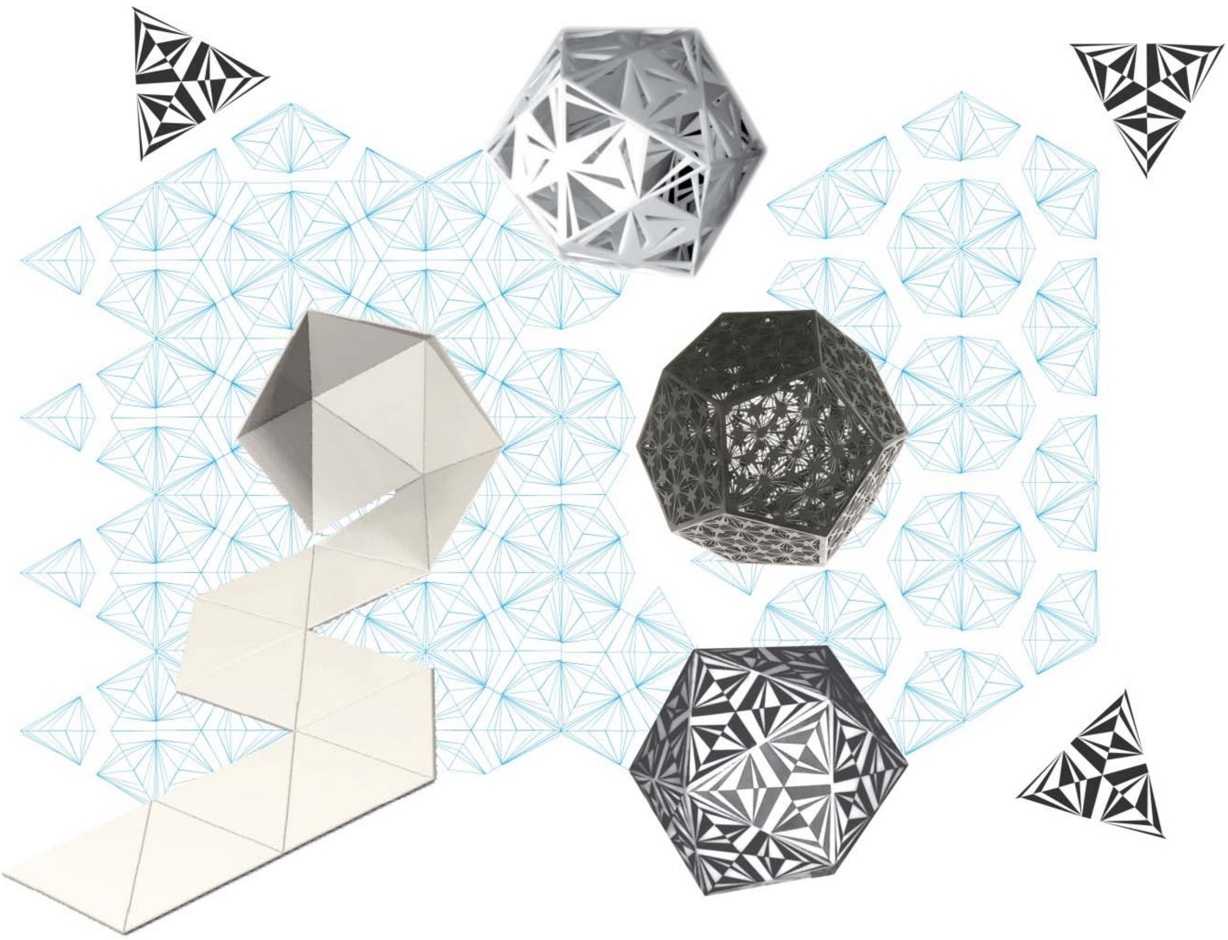
UNIVERSITY OF LEEDS

Design: Shape & Structure
(decomposition / description)
A white paper resulting from a University of Leeds
Fund for International Research Collaboration (FIRC) project



This white paper was edited by Alison McKay with input from the workshop participants.

The visual materials on the front cover were produced by Briony Thomas.

February 2016

Editor: McKay, Alison
Place of Publication: Leeds
Publisher: University of Leeds, School of Mechanical Engineering
Date of publication: 2016
DOI: http://doi.org/10.5518/wp/1

1 Abstract

Shortcomings in the cognate body of knowledge that underpins the discipline of engineering design result in slow incremental progress based on a series of ad hoc, idiosyncratic solutions with few if any high impact breakthroughs. This is problematic because engineering design is a critical early stage of the innovation processes that deliver new products to markets where societal challenges are addressed and wealth generated.

This white paper reports results of a meeting that brought together a group of international research leaders. The working title for the meeting was "Design: Shape and Structure (decomposition/ description)" and the participating disciplines included architecture, computing, engineering and industrial design, manufacturing, and philosophy. The meeting took an existing research area, one characterised by experience-based frameworks and approaches, and explored the feasibility of establishing an integrated theory that draws together design with grammatical and mereological theories of shape and structure. Applying such theories to the synthesis of new designs could elevate the work of a designer above the detailed nuances of individual designers without diminishing their creativity and ingenuity. Benefits would include a well-founded theoretical and computational infrastructure on which the next generation of design systems could be built.

The discussions focussed on aspects of engineering design that such an integrated theory might cover, how it might be used and who would benefit. With respect to coverage, three areas were identified: design process, functional aspects of designed artefacts and physical aspects of designed artefacts. The long-term impact of an integrated theory for engineering design would include coherent underlying principles for design definition and synthesis that could be used in design practice and education and as foundations for the development of future design systems. This, in turn, could lead to step-change improvements in the performance of the innovation processes that depend on effective and efficient engineering design processes, so resulting in the faster delivery of improved products to market using fewer (eg, financial and natural) resources.

This white paper reports discussions and key conclusions from the meeting. The importance of distinguishing between design as a noun and designing as a verb was recognised. Given the broad range of areas and sectors such a theory would need to cover, a collection of related theories was regarded as more feasible and valuable. When considering design processes, a key requirement for any design theory lies in supporting zig-zagging processes between physical (for 3D products) and functional aspects of a design. There are well-established mechanisms for the definition of 3D products, developed through the development of 3D CAD systems and associated applications, but further work is needed to create a similar depth of understanding for functional aspects. 'Capacity' was put forward as a general concept that covers both function and behaviour. For designing as an activity, the critical unmet need in today's computational design support systems lies in support for the ambiguity that designers need during creative episodes. Two areas for further development were identified: embedding[1] and the need for computational tools that can operate on visual objects (so-called "visual machines"). In the shorter term, general purpose implementations of embedding will support design ambiguity by allowing visual and other emergent objects to be superimposed on existing design definitions. In the longer term, visual machines will be designed to support ambiguity by operating visually rather than symbolically. A roadmap for the development of these mechanisms is provided towards the end of this white paper.

[1] https://en.wikipedia.org/wiki/Embedding and see Section 6.1 for more details

2 Purpose and rationale

The meeting was convened in response to discussions on engineering design research and the observation that, despite pockets of theoretical underpinnings in some areas, there is not yet a coherent body of knowledge on which a theory for engineering design can be built, extended and tested. Without such a body of knowledge, the discipline of engineering design is at risk.

The purpose of the meeting was to explore the feasibility of an integrated theory for engineering design. If realised, such a theory could lead to a wide range of opportunities. For example, a major challenge for engineering design research lies in finding affordable, efficient and effective ways to support families of engineering products through life. This is especially challenging because the lives of many products are expected to extend beyond the working lives of the engineering designers who created them and design information is often inaccessible because it has embedded within it idiosyncrasies of the approaches used by the people who created the design or the computational systems they used.

Our theoretical goal was to bring 3D design (including functional and aesthetic aspects) under a single umbrella so that different approaches can be brought together and commonalities leveraged through design education and research programmes. This will create opportunities for learning across design disciplines that will last well beyond the end of this programme. Our practical goal was to create an underlying theoretical framework for engineering design. Such a framework will be of value to design educators and practising engineers.

3 Method

The selection of participants was based on research interests and the potential formation of consortia for EU, White Rose and RCUK funding. To improve the immediate impact of the workshop we invited three EU research students to participate. A list of attendees is given in Appendix A and the meeting agenda in Appendix B. Before the meeting, each participant identified a paper to share with the group; references for these papers are given in Appendix C. The workshop was structured around the following process that is reflected in this white paper.

1) Day 1: Introductions and initial perspectives based on shared papers
2) Day 1: Walk and talk at Yorkshire Sculpture Park (www.ysp.co.uk/)
3) Day 2: Reflection on Day 1, primarily in break-out groups with student rapporteurs
4) Day 2: Road mapping

Participants: (left to right) Pieter Vermaas, Laura Harrison, Alison McKay, Chris Earl, Peter Simons, Saeema Ahmed-Kristensen, Susan Stepney, Alan de Pennington, Briony Thomas, Corinna Königseder, Marta Perez Mata, Eduardo Castro e Costa, Keith Ridgway, George Stiny, Jose Duarte

4 A summary of the discussions

The content of this section is a blend of notes provided by multiple participants.

4.1 Introductions and initial perspectives based on shared papers

The workshop was opened by Prof David Hogg, Pro-Vice Chancellor for Research & Innovation at the University of Leeds. The purpose of this session was to share ideas and thinking on the goal of the workshop and its relationship to the papers participants had selected before the meeting. The workshop was introduced using the mind-map given in Figure 1. This was followed by a discussion on why we selected the papers we did and what we learnt from reading each other's papers.

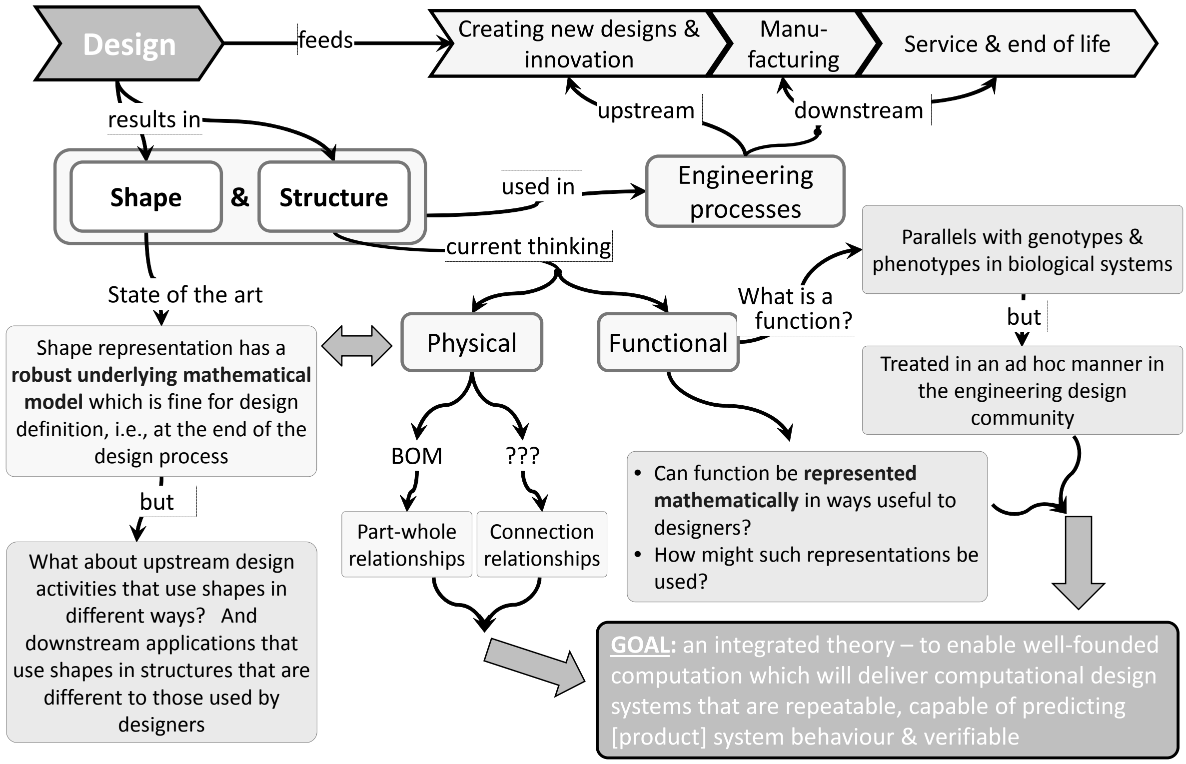


Figure 1: Mind map used in introduction

A key area for discussion lay in the feasibility of a single integrated theory for engineering design and questions around whether design is best regarded as a process or the result of a design process.

4.1.1 Design as a process

Designed artefacts result from product development processes that include engineering design and feed into downstream processes such as manufacturing, support services and end of life processes. Some of the papers selected draw from biology and highlight two areas for theoretical developments (i) describing what designers do and (ii) selecting appropriate degrees of prescriptiveness for different activities within engineering design. Currently available models of design processes cover both of these dimensions through descriptive and prescriptive models of design.

Prescriptive models tend to be used in engineering design education, where students are guided through an iterative process that includes requirements definition, divergent and convergent thinking and design development, analysis (which, for the purposes of this paper, includes traditional design analysis, and design optimisation and simulation). There are many prescriptive models in the literature; the majority include iterative cycles and the following process steps: design definition, synthesis, analysis and communication. A drawback of prescriptive models of design is that they tend to be based on observations of what designers currently do, staying close to current practice, which poses the risk that innovation in the design process itself, for example, doing something new or designing in new ways, which could limit the capacity to create radically new kinds of design solutions.

Descriptive models tend to be more abstract and focus on the underlying cognitive processes of design: building understanding of how human designers do designing. For example, (Suh, 1990) provides a design cycle that can be applied to any kind of designed thing and Schon & Wiggins (Schon, 1992) describe a seeing-moving-seeing process that can be applied to the design of things that can be visualised graphically. These kinds of model tend to be used in engineering design research where the research process aspires to generate new methods and tools to improve the activity of designing. For example, research in shape grammars and their application to the design of products is often contextualised using Schon and Wiggin's model because 'seeing' requires visual representations and transformation processes that are a core aspect of shape grammars. As digital technologies become more sophisticated, the integration of computational and human processes will become closer. From a socio-technical systems perspective, where this transition could be regarded as a change process, means of ensuring people are free to maintain or improve their creativity and ingenuity will be critical to improving design productivity and realising the benefits of digital technologies (Brynjolfsson and Hitt, 1998).

Regardless of the kind of process model, a key question surrounds what a process is (and, therefore, is not). In an everyday sense, processes can be regarded as tapestries of events. This implies open-endedness and change over time. Computationally, if a process could be captured as a form of calculation then new kinds of tapestries could be dynamically reconfigured and new structures could emerge. Benefits of this perspective include that the process can always continue and new calculation rules can be imported. As a result, there may never be a complete description of a process; the process design (and so the description of the design) would be more a way of doing the process than a series of specific steps; this aligns with Suh's and similar models of design processes that focus on human processes rather than specific tasks. In computer science, process algebras[2], are used as the basis of some programming languages. They cover different kinds of relationships between process elements, for example, composition, sequence and parallelism. From a philosophical perspective, Alfred North Whitehead's book on Process and Reality and its untimely review (Simons, 2013) highlight the philosophical importance of process. However, current education systems tend not to school people in thinking about process. For Whitehead, process covers many different types: ranging from concrete physical processes (such as chemical processes) through to abstract natural processes (such as dream processes). In computing, functions can be seen as process definitions and their execution as process operations.

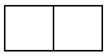
Reflecting on these disciplinary perspectives, prescriptive engineering design process models that are widely used in engineering education programmes (such as, (Pahl et al., 2007), (Ulrich and Eppinger, 2007) and (Cagan and Vogel, 2012)), reflect the mathematical perspective where processes are series of computable functions. This perspective may be well suited to engineering education, where the majority of teachers and students are well versed in mathematics and the physical sciences and the goal is for students to work in systematic processes. However, it may be less well suited to underpinning theories of engineering design and ways of designing which, as with any design process, include requirements definition, synthesis, analysis, decision-making and [process] definition. For engineering design, a unifying theory needs to cover design process and designed product. For the design of physical products, shape is critical because it is inextricably linked to functionality until software, which can be used to create its own functionality, is involved. For example, in a mechanical design shape may relate to the parts of a machine whereas in a drug design the molecular shape is related to how it interacts with other molecules.

Functionality, in turn, can be related to users and their goals, which leads to usage processes. Users include downstream processes ranging from those close to design, such as manufacturing, and later through life processes such as operation and maintenance. In large, make-to-order systems, design and manufacture occur at the same time and the [design] Bill of Materials (BOM) is very flat and fed

[2] Baeten, J.C.M. (2004). "A brief history of process algebra". Rapport CSR 04-02 (Vakgroep Informatica, Technische Universiteit Eindhoven)

into manufacturing planning [conceptual and physical]. In engineering design, where through life support is increasingly important, a consistent theory for process could support the systematic incorporation of user processes, from different life cycle stage and stakeholder perspectives, into design processes and the computational tools that support them. Activity theory[3] could be a way of doing this because it supports the definition of links between artefacts and human activity.

As with any design, the design of a process needs to accommodate trade-offs between strong structure (which improves the likelihood of repeatability in process delivery and offers the potential for automation) and weak structure (which enables flexibility in process delivery and contributes to the resilience that is needed to accommodate turbulence such as that which arises from unanticipated change). It is important to remember that turbulence is a metaphor and a model is an approximation to reality. The idea of turbulence in design leads to a need for design support systems (in their widest sense, i.e., including computational, theoretical, and human systems and behaviours) to be able to handle surprises. Could we have a process where we could vary its structure[4] depending on purpose and situation? For example, synthesis and divergent thinking needs emergence that comes from turbulence whereas analysis and convergent thinking benefits from structure, different structures for different analyses. What would happen if we took a similar perspective for decision making processes? For example, strong structure may be better for decisions that are towards the selection end of a spectrum whereas weaker structure may be more appropriate when decision makers need the freedom to explore different ways of thinking and new ways of seeing possibilities. Emerging simulation tools make this a possibility. Grammars work when things go well, a theory is also needed for when things go wrong. Structure can introduce turbulence in the first place. The ability to derive structures as and when they are needed in the tapestry of things can provide such a capability. For example, two partial descriptions that are derivable from the following are one rectangle and two squares.



4.1.2 Design as an artefact – Shape & Function

The design of physical products results in shape, structure and material specifications. Shape typically results from synthesis processes whereas material specifications result from selection processes. As new manufacturing processes emerge and computational support for materials science develops, it is expected that material specifications will also result from synthesis processes that take into account functional requirements, shape and potential manufacturing processes. Designs also include structure, for example, the BOM on a general assembly drawing or the assembly mating conditions in a Computer Aided Design (CAD) model. However, other structures, such as function structures, are used through the design process and other structures, such as disassembly structures, are used through the life of the product. In addition, the rationale for the design structures created during a design process is often related to convenience to their creators in a particular design activity rather than a wider analysis of the need for particular design structures and their intended users (human or computer)[5].

Processes of designing involve zig-zagging between the functional and physical domains (Suh, 1990). However, while examples of relationships between functions and shapes are straightforward to

[3] https://en.wikipedia.org/wiki/Activity_theory

[4] The structure of a process can be varied by changing the process definition itself or through its meta-structure: a meta-process that can be used to drive computations. If a bootstrap meta-process was available then tools used to vary a process definition could also be used to change the process structure and so the operation of process-related computations.

[5] Discussions with four Yorkshire-based SMEs after the meeting highlighted the quality of structures created during design as an issue for downstream automation and efficiencies. One company referred to the need for discipline during design and trained designers on the importance of well-defined design structures whereas another avoided automating the link from design to manufacture because the product structures created during design were not suitable or of sufficient quality for downstream use.

identify, there is no general theory for this. Grammatical approaches can provide general ways of establishing relationships between structure, shape and functions through description rules. Structure can then be regarded as a derivative. For example, shape grammars are not "stuck" with a given structure because they operate on visual objects. In design, where shapes are synthesised, it is especially important to keep the designed shape open. Function is less well understood, in part, because function includes semantics, c.f., denotational and operational semantics in computer programming[6]. How one links shape and structure is a part of design and firmly entrenched in history.

The discussion on function considered whether function is inherently a part of an object and whether differences arise from different application areas. For example, a teacup has many functions. To determine its potential functions we need to know its form and material but when does function appear? From philosophy of the dual nature of artefacts, products have capacities (physical properties and behaviours) that govern the range of possible uses to which the product can be put and affordances which determine how users 'naturally' try to use them. A product's function is the specific capacity for which the product was designed to be used. Drawing parallels with shape, which has a range of manifestations through the product lifecycle, capacity may well be the functional equivalent to shape, where, for example, the 'as designed' capacity would be function whereas users can still employ the product for its other capacities. Given this, a theory of capacity would inevitably link into usage processes and so be a part of an integrated theory of design.

Considering design representations, there is no one right representation for a given design. What constitutes the "best" representation depends of the purpose for which it is intended. From a philosophical perspective, the number of design structures that can be associated with a given product or its design is unlimited. In computing, structure can be superimposed on any design definition. This leads to the multiple bill of materials problem where traceability is essential if work is organised, as it typically is, according to various design representations. Discrepancies between representations are managed, and translations made, by people using methods and processes that tend to be slow and prone to errors. The management of multiple design representations is not, in itself, a value-adding activity that requires human ingenuity and creativity. Current design systems, in the form of product data and lifecycle management systems, can highlight multiple representations of a given design but not resolve conflicts and discrepancies or manage, in any detail, traceability across such representations. Current CAD systems only support the definition of one each of different types of structure (e.g., one BOM and one set of mating conditions in 3D CAD) and so limit the range of structures it is possible to define and the extent to which such structures, and the definitions they are associated with, can be integrated with one another. In computational support for designing, [data] representation is very important. Computer programs can be created to move between representations, for example, as used in Model Driven Engineering; there is no one right representation and code refactoring can be used to change form but maintain a given function.

From this discussion, two dimensions of a design theory were identified: process and product. For process, representations for different purposes are needed: product definition (for communicating design results with others), design synthesis, design analysis (including computational analysis, simulation and optimisation tools), design decision-making, and design iteration. From a product description perspective, and drawing parallels with natural systems, there are two aspects covering capacity and affordances, and shape and material. In addition, there is a third aspect, structure, which is used to tailor design descriptions for different activities and processes. These two dimensions are used as axes in Table 1, to provide a structure for key points that arose in the discussion.

[6] In computer programming, denotational semantics provide the mathematical meaning of a function whereas operational semantics relate to how a programme is executed to deliver the function's required behavioural properties.

Table 1: Summary of key discussion points

	Design description: Definition / Representation	Design process: Synthesis, Analysis & Decision support	Notes/Examples/Questions
Capacity & Affordance	There is no one functional language and there is no agreement on what constitutes a function. Functional structures tend to be created on an ad hoc basis. It is not clear what is and is not a function structure. Some contain part-whole relationships, so mereology may apply though the purpose of part-wholes for functions would benefit from further consideration. Other function structures describe how functions are connected. A function of an artefact can be regarded as a manipulation of information and/or material (e.g., calculation, movement, assembly) that provides affordances for a user (e.g., a building provides shelter by being a barrier: manipulating the environment to prevent parts of it coming inside). A key issue lies in the transition from functions that are primarily passive, e.g., barriers, to active functions, such as cogs and wheels and software controlled functions such as the manipulation of information in addition to manipulation of material.	If designing from scratch, could we have a single model of function? The benefit of a single model for anything remains to be determined. If models are built for purposes (which govern assumptions, etc), and given the potential for multiple purposes, it is reasonable to suppose that there will be need multiple models, representations and representation schemes. Shape grammar alone is not enough. It may be possible to construct rules that include function but functional ambiguity (which is a desirable feature of design synthesis processes) emerges from interactions between shapes. For example, from architecture, a kitchen/dining area may be one space, which purpose depends on what is wanted: different interpretations and different uses. In design analysis, are these boundary conditions and parameters on design analyses? Function is captured in objective functions in optimisation methods and tools. Design analyses and simulations can be used to study anticipated behaviours. These can be regarded as a form of capacity. These could provide examples of functional representations and further analysis of these could be informative.	What is the function of a product? What is the function of an object? For example, it is possible to find a completely new use for most objects, e.g. an umbrella, but doing this does not result in a new object? From philosophy, capacity is promising as a concept to underpin the notion of function. Like objects and their shapes, capacities can vary across an object's life. For example, function may be regarded as an 'as-designed' capacity whereas behaviour may be an 'as-used' capacity. Different capacities may be of interest life. For example, the capacity to support itself while being made may be a capacity of interest during manufacturing, whereas the capacity to survive shipping may be more important during distribution and the capacity to be recycled may be important at end of life but less so earlier.
Physical	There is no one language for product definition, but core elements for 3D parts are shape and material. 3D piece parts (or components) are defined through their shape and material. 3D assemblies, where parts are related to each other through aspects of their shapes, are defined through their assembly mating structures. The shape and material of a 3D assembly comes from its parts. Shape grammars can be used to design of static and moving parts, including physical constraints, earlier in the process than using techniques (such as today's CAD) where the shape is defined directly. In urban design, an ontology, such as a city information model (Montenegro and Duarte, 2009), that captures squares, streets and other recurrent features, needs different kinds of ontologies for different uses. There would be a benefit in an unifying theory that could be used to support interoperability between systems and models that use different ontologies.	When designing functional components, there is a need for functional and physical descriptions to be used simultaneously. In architecture, a specific problem occurs in large developments of diverse customised housing. Shape grammars may be derived from architects but, from experience, different architects have different grammars for a given development. At a more abstract level of housing, generic grammars built on common concepts, from which specific grammars can be derived, are feasible. These allow the use of patterns and supports grammars capturing different levels of abstraction. Brian Cantwell Smith identifies the need to be able to handle "surprises". For example, putting two squares together and getting a rectangle with a line at its centre. What the surprises are depends on which rule "fires" when it happens. As well as within a design, emergence occurs across using, making and operating processes. When translating between function and shape, changing one, changes the other and vice versa.	Whether a given part is regarded as a component or an assembly depends on the purpose. For example, a ball bearing could be regarded as a component for purchasing but drawing conventions often show them as assemblies ... which communicate functionality. Montague grammars from linguistics enable multiple grammars to work at the same time.
	For, shape and material, representation schemes are available (such as those in STEP standard). These are intended for data exchange ... a means of achieving interoperability between systems.	Designs come from a range of sources. Some are better defined than others. Generating designs from arbitrary sources is akin to throwing a plate up in the air, dropping and breaking it, and using that pattern as your starting point for design.	STEP (ISO 10303) and the like are suitable for post hoc definitions where the need is to capture designs after the designing process has ended and when ambiguity is undesirable. They are not well suited to design synthesis where ambiguity is desirable. Grammatical approaches can provide such ambiguity. For example, it is possible to have different grammars during/after design, where the latter imposes more constraints, and the former allows these to be relaxed during the process. Is interoperability, e.g., across processes, such as, design and manufacture where definitive models improve efficiency of production processes, needed in design synthesis? As new, more flexible, manufacturing approaches emerge or in certain disciplines, could there be benefits in ambiguity, e.g., as in 'refactoring' from computing?
Structure	Mereology provides a mathematical foundation for the definition of composition structures. There is a need for a comparable foundation for how things are parts (e.g., physical parts with shapes or functions) are connected. There are some examples of connection structures that capture relations and connections, such as connections in topology. Topology could offer new insights over graphs and hierarchies. Mereotopology (en.wiktionary.org/wiki/mereotopology) has potential but work would be needed to make it accessible to non-philosophers. Need more than shape/form to do 3D designing. For example, there is a need for material properties (cup has to hold liquid, maintain temperature) when designing functional products.	A need to support shape calculations where things are changing, for example, changing topology, was identified. Structure can introduce the turbulence in the first place. Typically, multiple shape models are needed to support different kinds of engineering analyses.	Washing machine repair was used as an example of topological closure; a small nick requires replacement of entire substructure. The Mereos report ((Dement et al., 2001)) provides a thorough analysis of, and framework for, engineering structures.

The following were key conclusions from the discussion.

- Complex systems mean multiple models; design support systems need to embrace multiple models. This need for and the potential value of multiple descriptions of a given design has been identified for many years by several authors (e.g., (Stiny, 1990, 1998)), (McKay et al., 1996), (Andreasen, 2011) and (Dement et al., 2001)).
- Embedding allows for multiple perspectives simply by being superimposing one description on another. Comparable concerns arise in object-oriented programing but no solutions are yet available. Shape, shape rules, and the like provide a way out for engineering design when related to physical products. A more general implementation of embedding may provide a solution to this wider group of problems.
- In engineering, the problem of multiple BOMs is a consequence of the need for multiple design descriptions. Achieving traceability across BOMs, where discrepancy could be up to 40% based on some participants' experiences, is a key challenge with high potential benefits. Human beings currently do the translation across BOMs. From an engineering perspective, we talk about structure, but it is not always clear what this means. McKay, Stiny & de Pennington (2015) identify six principles for defining design structures that are applicable at a meta-level, for example, in underlying representation schemes. Many BOM problems relate to phantom parts and duplication across BOMs. BOMs can be defined using a common meta-model but problems such as phantom parts and duplication arise from the instantiation process.
  An example raised during the meeting was that of turbine generators. Design and manufacture occur at the same time; designers, manufacturers and planners use different models and structures. There is no translation between structures. The BOM tends to be very flat and then, through life, endless design structures are created. This highlights the fact that the need to relate across structures does not mean that all structures need to be related to all other structures in a given design process. Creation, validation and verification requirements, benefits and costs vary across sectors and need to be taken into account.
- From the design computation papers, there was a consistency of presentation and depth of thought that spanned a broad spectrum from ontology to practical design via symmetry.
- In urban design, there is a need to describe environments: city information models (and underpinning ontologies) that create the possibility of a <u>unifying</u> theory. Previous work on architectural design captured the work of architects. A generic grammar for housing, described overall concepts and was used to develop specific grammars at different levels of abstraction, going from generic rules to specific rules.
- In reviewing the papers that were used as a backdrop to the workshop, the following were missing:
  - Understanding how human designers design; shape grammars alone are not sufficient.
  - The importance of function and structure; this was recognised as a major problem for 3D parts.

4.2 Walk and talk at YSP

By the nature of the walk and talk there are no notes on these. However, the small group discussions informed the Day 2 discussions.

4.3 Day 2: Group discussions & road mapping

The purpose of the discussions in Day 2 was to reflect on Day 1, in breakout groups with student rapporteurs, and to produce a road map for future research in this area.

In this section, the discussions are reported as a synthesis of the notes taken by the student rapporteurs; complete sets of notes are provided in Appendix D. The topics for the group discussions are provided in the section headings.

4.3.1 Session 1: Important learning/message for white paper

The first discussion identified a series of key areas to be highlighted in this white paper.

<u>Historical contexts</u>

Historical perspectives, drawn from histories of engineering design, could provide opportunities for learning. For example, designers have always used models; technology makes creating them faster and easier which, based on anecdotal evidence, has tended to lead to larger numbers of models. Valuable learning could be gained from studies of how engineers' uses of models have developed over time. This is likely to need collaboration with researchers in appropriate disciplines. For example, at Leeds there are researchers in Philosophy studying the history of technology and the School of Law using patents to study how the design of locks has developed and been driven by criminal behaviours such as the unpicking of locks.

<u>Philosophical perspectives</u>

Mereology provides a theoretical foundation for how physical parts relate. Part representations can be extended to function but this demands more detailed understanding of what function is, and is not. Capacity is a general concept that includes function where function relates to the uses that a designer intended for a product. Questions around a mereology of functions were discussed and connections between function and process highlighted.

<u>Human perspectives</u>

Human designers use two kinds of process: stage-gated *project/programme management processes* that are used to manage new product development projects, and *creative design processes* where ideas (including shape and structure) are generated and developed with human creativity and ingenuity. Many IT vendors provide computational systems, in the form of Product Data and Lifecycle Management [PDM and PLM] systems to support the former, management, processes. These are not the focus of this white paper but their existence is important because they are built around BOM structures that can inhibit creative processes by forcing product structures that are not well suited to particular design activities. Instead, these structures are necessary for breaking down design tasks into manageable chunks and for downstream processes such as manufacturing where ambiguity in a design definition is undesirable because it is likely to lead to waste such as increased levels of scrap.

This white paper focusses on the second of these processes: creative design processes. Artists, who create shapes and structures but tend to be less constrained by management processes, could be a useful subject to study if we wish to learn more about creative design processes and ways in which shape and structure, and their representations, develop. A key difference lies in the importance of functionality and constraints to engineers, but for designs such as sculptures seen at Yorkshire Sculpture Park, properties such as manufacturability and the ability of a sculpture to stand and support itself in its intended orientation were apparent, indicating that constraints also have some relevance to artistic design.

The consensus within the group was that computational tools to support designers designing are more likely to be effective if they align with designers' cognitive processes, such as Schon & Wiggins' seeing-moving-seeing model. As such, important design requirements for such tools are the ability to tolerate (or better still, capitalise on) emergence and ambiguity, including inconsistencies across different views and multiple design representations. Within the context of industrial engineering processes, there is also a requirement to support multiple stakeholders involved across subjects, sharing common knowledge and aligning design results (which may be in the form of multiple not wholly consistent models) so that they feed effectively into project/ programme management processes and systems.

Models and tools: design descriptions and representations

The goals of design activities change as product development proceeds. For example, early in such processes designers may be exploring alternative options whereas later they will be developing and evaluating selected solutions. Other differences occur because of the kinds of product being developed (e.g., in some cases aesthetics is more highly prioritised) and constraints to which product development processes are subjected (such as regulatory frameworks for some products). These differences inevitably lead to multiple models and tools.

A model is a form of design description. Both the description itself and its underlying representation have varying degrees of structure. Design structures (such as BOMs and function structures) occur in design descriptions. In choosing a representation scheme, the forms of possible design structure in the design description become limited. Since many design descriptions are created to drive software analysis and simulation tools, the permissible structures are typically related to the requirements of the tools. A key benefit of such design descriptions is that they can used to generate design visualisations that, in turn, can be validated and verified in a number of ways, e.g., with experts and through experimental work and physical testing. Given the capabilities of today's tools, relationships between structure, shape and function are limited because very few operate on these three concepts. An exception is emerging simulation and optimisation tools where relationships between shape and function are captured in mathematical functions. However, these are highly restrictive because of the computational implications of even low levels of complexity. It is, of course, possible to create more general design descriptions that are not related to specific design tools. However, such models are difficult to verify and validate in a systematic way, in part because the purpose for which they are created is often not clearly specified.

As more models emerge through a product development process, the potential benefits of interoperability across models grow. The debate over whether the goal of interoperability is best achieved through the integration of design descriptions, tools or representation schemes began in the 1970s with the advent of data exchange standards such as IGES and STEP. Embedding, from the shape computation community, has the potential to support the definition of relationships across design descriptions but needs further exploration. A key problem with current CAD systems is that the CAD paradigm contains sufficient information to create unambiguous design descriptions for downstream applications such as manufacturing but not enough information to support designing itself. To support design activities, tools and design descriptions need to support the definition of relationships between function and shape, in general.

Design synthesis: creating designs that can be described

Shape systems used in product development processes include CAD and, in later stages, systems such as Shape Grabber (http://www.shapegrabber.com/) which operate on 3D scan data. Parametric capabilities of CAD systems can be used to support the definition of new designs when the solution can be defined through a parametrized shape, but, support for early stages of design, before a shape has been established, is less well supported.

The shape grammar approach has been shown to be effective in supporting architectural and visual design but it needs to be integrated into wider development processes. To support engineering design, where shape and function are often inextricably linked and designs are collections of discrete parts, more, non-shape related, semantics are needed. A given grammar can have multiple semantics. However, grammars can be used to restructure a design description and this structuring could include design structures and other non-shape information. The question of whether shape grammars restrict innovation was raised. In response it was noted that, from a theoretical perspective, there is the flexibility to define whatever rules are needed as and when they are needed. However, exploiting this flexibility requires substantial development in the capability of shape grammar-based design tools and in the competencies (and so education) of designers.

Theoretical perspectives

The way forward is likely to be a collection of theories with some interrelationships. While recognising the benefits of a single coherent theory, a more feasible approach would be to look for a core that would contain the fundamentals of design and with specific extensions for different subjects. The core would cover the central and abstract knowledge about design and its axioms in some way. This core would have properties such as abstraction, reusability and cohesion (i.e., providing the possibility to translate across domains where appropriate), and would be a solid base on which domain-specific knowledge could be built. However, it was not clear that such a core exists or is achievable.

Before proceeding, more clarity is needed on the aim of the theory. For example, is this theory aiming to capture the knowledge and innovation in engineering design and resulting design descriptions, and/or is it aiming to support people doing design? Key targets for such a theory fall into two areas: design description and design process. A theory for design process would make design processes faster and smoother, by relating theories for generative, descriptive and prescriptive perspectives. For design description, a theory for describing and working with shapes, processes such as those embedded in software, and designs is needed. This could lead to a unified design language that could be used to express translations between information from different knowledge areas and representations. The current lack of effective, general methods of communication was regarded as one of the most important issues to address.

4.3.2 Sessions 2 & 3: Vision of where we want to be, issues and actions

Areas discussed in these two sessions fell under three of the headings from Session 1: theoretical perspectives, models and tools, and human perspectives (which included design synthesis).

Theoretical perspectives

It was argued that benefits would include a well-founded theoretical and computational infrastructure on which the next generation of design systems could be built. Discussions across the groups focussed on two areas: design processes and design descriptions. Permeating both areas was the need to preserve and understand a core (covering processes and representations) that is common to multiple areas (e.g., models and multiple views).

It was recognised that there already are numerous descriptive and prescriptive models for and of design processes, ranging from general models of design such as Suh to specific models of engineering design such as Pahl & Beitz. Figure 2 resulted from discussions focussed on what the scope for a design theory might be. A conclusion was that there may be too many different inputs to such a broad 'theory'. Restriction of the scope, more focus on specific problems (e.g., interactions and interoperability between design methods and tools) and reuse of existing theories where appropriate would be beneficial. From a software development perspective, the need was for 'a theory to help the design *process*, not the design' – as this would help with the programming aspect of design.

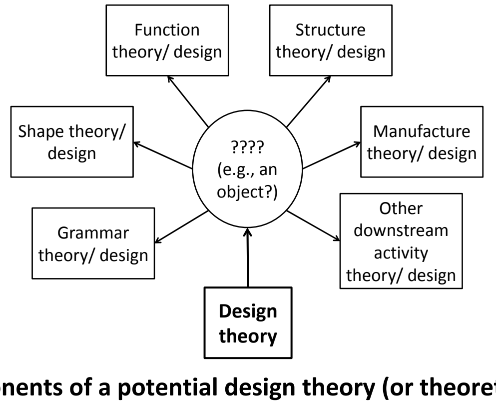


Figure 2: Components of a potential design theory (or theoretical framework)

A key limitation of current process-focussed design theories in supporting seeing-moving-seeing type design activities is that they are not connected with sufficiently detailed theories of design descriptions that include shape and transformations between shapes. Existing theories of design description (such as Constructive Solid Geometry (Requicha and Voelcker, 1983)) are symbolic. These serve their intended purpose, the provision of unambiguous shape definitions to support 3D visualisation and downstream shape-based applications such as manufacturing, very well. As a result, they are widely used to underpin commercially available CAD systems and standards to support interoperability between such systems. However, although they can be used to generate visualisations of designs, such as rendered images, these renders need to be interpreted by humans. When designing 3D products, there is also a need to support processes of transformation that include composing -decomposing-composing, while moving across different levels of a design structures containing part-whole relationships and while zigzagging between functional and physical domains.

Seeing-moving-seeing can be regarded as a form of composing-decomposing–composing where the things being decomposed are visual objects. Given this, but drawing in the opposite direction, different design structures can be regarded as being emergent from others. Embedding seems to be necessary in any design process where seeing-moving-seeing is necessary. Recent computer technologies, such as pattern matching on large data sets (as used by Google Translate), are creating new solution principles with the potential to support the computer implementation of embedding.

In conclusion, a need to seek relationships between shape, structure and function was identified. To achieve this, the following questions need to be addressed.

- Are there some things that all areas of design represented by the group at the meeting need?
- How should function be defined? The notion of function is currently ambiguous. Regarding function as a form of capacity seems promising, since this would allow function to be considered, like shape, as something that is used in different ways through the product lifecycle. To achieve this, an analogue to Constructive Solid Geometry for capacity is needed that includes both syntax and semantics. In addition to semantics, a way of validating what a design described through a given design definition does (and its affordances) against what it should do will be needed. An initial task to develop this question would be a literature review of current methods for defining and working with capacity (including function and behaviour).
- Where does material fit into this? Does this vary depending on whether material is selected or designed?
- Is a unified language for communication between design methods and tools feasible? The current lack of an effective method of communication was regarded as one of the most important issues to address. In practice, one universal language may not be feasible, but a process to design such a language, given the components, may be.
- Two types of engineering design were identified: '*Optimisation*' and '*Disruptive/Radical*'. The scope of any theory would need to accommodate both of these.

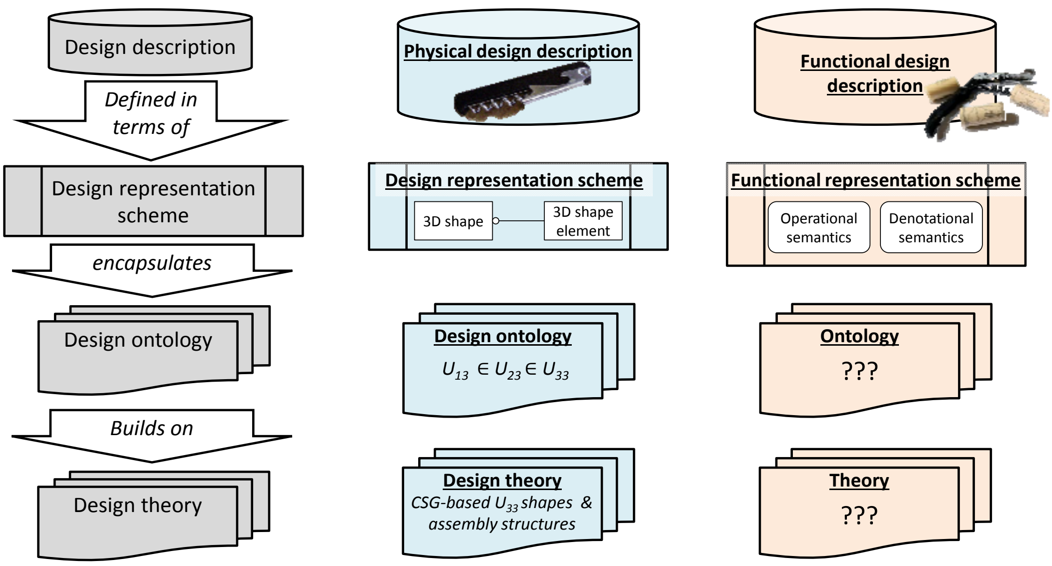
In answer the question, "What is needed?" people need tools to support its use, and a killer application. Given current interest in emerging manufacturing technologies and robotics, such an app could link design and 3D printing, of parts and assemblies, possibly of designs that are difficult to produce, such as 3D printing of folding patterns. Such an application would need a limited vocabulary and descriptive set of rules for processes - design/move/assemble/use.

<u>Models and tools: design descriptions and representations</u>

The diagram in Figure 3 provides the structure that has been used for this section. A corkscrew is used as an example because its physical description is passive but its functional description requires a user process perspective and relationships with other products such as corks and bottles. Design descriptions are used in engineering design processes and defined in terms of an underlying representation scheme. Value could be added to current representations by giving more meaning to designed shapes and their functionalities. With data such as this, which could be metadata associated with a given design description or extensions to the underlying representation scheme, the potential of computational tools in facilitating the design process could be improved.

Design representation schemes specify the format of design descriptions and, in turn, the kinds of operation that can be performed on the description itself. Typically, different representation schemes are associated with different design tools. Since new product development processes inevitably incorporate multiple tools, there is a widely recognised need for interoperability between representation schemes. Ways in which such interoperability might be achieved remains an open question. The aspiration to build an abstract model from which multiple representations, for different computational tools, could be extracted was discussed. While architecturally elegant and popular within the wider community, the practicality of such a solution, often referred to as a "universal database" or the like, may be limited. For example, substantial effort in the international data exchange and associated standards areas have made little progress in achieving a common representation scheme from which other, application views, can be extracted. For example, the STEP interpretation process for tailoring application protocols from the integrated resources includes the potential to add relationships, meaning that the information that would need to be extracted does not exist in the integrated resources which are the underlying representation scheme of the STEP standard.

Design ontologies provide the conceptual frameworks on which design representations, and so descriptions, are built. They are realisations of theories of design information. The Mereos report (Dement et al., 2001) is the most developed ontology for design definition that was known within the group. It provides support for design definitions, their multiple structures and change information. As a result, Mereos-based design definitions are well-suited to support downstream applications, especially in highly regulated sectors where it is imperative (from both regulatory frameworks and business drivers) that change be managed effectively and efficiently. A different kind of design ontology would be needed for creative design tasks where visual representations of designs and design transformations are needed. Whilst design changes and design transformations may be regarded as the same kinds of concept, and so learning from Mereos may be appropriate, the concepts upon which they operate are different. More work is needed on understanding non-shape related design transformations such as product structures, materials and system architectures, and creative processes that result in less tangible products such as those of software engineering, which are core functional elements of many of today's engineered products.



First two columns adapted from: McKay A; Kundu S. *A representation scheme for digital product service system definitions*. **Advanced Engineering Informatics**, vol. 28, pp.479-498. 2014.

Figure 3: Framework for design descriptions (left), physical example (middle), functional example (right)

Human perspectives

Symbolic representations of shape are highly effective in supporting downstream shape-based activities. For design synthesis, however, visual representations are important because effective computational support for human designers working in seeing-moving-seeing type ways embraces and supports the ambiguity (from emergence in visual representations) that is needed to drive creativity. In conclusion, ways of bridging the gap between visual and symbolic information are required. This led to the idea of a 'visual machine' that computes on components that people see, visually. To deliver such a machine, the following questions would need to be addressed.

- Is a visual machine feasible? Such a machine would represent shapes visually. Since the first class objects in such a machine would be visual, anything that could be seen could be operated on. This contrasts with symbolic representations where visual objects can only be manipulated if they have a symbolic representation, e.g., multi-dimensional geometry in CAD systems.
- Could early versions of visual machines be created by embedding visual information into symbolic representations?
- How might information from the visual machine be translated into symbolic representations for downstream applications?
- Special focus on the areas where designers experience the biggest difficulties could be a good starting point for the development of a visual machine. The key challenge was identified as finding ways of implementing a general notion of embedding. Embedding is about relationships: input of computing in design and vice versa. In the longer term, research in this area could identify requirements for new kinds of computational device that operate on visual rather than symbolic information.
- In conclusion, we need a visual machine that does both: computes visually and symbolically.

A framework to support the realisation of such a machine would need to allow shifting descriptions and [designers] moving within and between the levels of abstraction. Key elements of such a framework would include support to move between and within levels, a focus on shape and support for shifting design descriptions in both physical and functional domains. Rules of some kind, such as shape and other grammars covering both syntax and semantics, would govern permissible transformations and so the shifts in design description. A flexible core, based on such a framework, would support people during their designing processes. This is where innovation would happen. Innovation itself could be moulded through changing the rules and applying them at different levels of abstraction. Within this discussion group, it was concluded that shape grammars could deliver such a visual machine. However, more shape grammar knowledge, wider understanding of its potential and accommodation of functional perspectives are needed. Hence the conclusion that what is needed is a "killer application" of shape grammars. This application would be easy to use and applied to a problem that would inspire people to use it. 3D printing could be used as part of the application allowing users to communicate directly with makers. The emphasis of this paragraph on shape grammars is linked to the interests of the group at the meeting; an early step in creating a visual machine would be to consider a wider range of solution principles.

4.3.3 Technology Road Mapping

The roadmap in Table 2 draws together the results of the three discussion groups' technology road mapping activity. The map is divided into three colour-coded areas: general items are in black, design definition-related items are in **bold blue** and items related to design generation are in *italic red*.

Table 2: Overall roadmap

Timescale	Past	Now	Near Future	Far future
Vision	- Still frustrations with systems although they have improved a lot over the last years depending on the industrial setting you are in. - Ability to move between systems and tweak descriptions is good but the path back to reconstruction doesn't always work		- considering human - machine interactions - satisfying the needs of the user in better ways - general notion of embedding - stitching these together, improve transformation, increase flexibility	- An application that can vary how a programme follows design theory depending on what the user wants to get out - General implementation of embedding - Uptake of the design theory in different scenarios
	- **A common model ... general model and technologies**	- **Applications and models for different contexts**	- **Ability to embed different structures into shape**	- **Database vision: all knowledge is compatible and stored somewhere (long way off)**
	- *In limited domains, ability to generate designs in a domain (usually by hand)*	- *In limited domains, ability to generate designs in a domain computationally*	- *Develop translation tool of realistic demonstration - developing the design support system*	- *A visual machine that can compute both visually and symbolically*
Challenges/ Drivers	- Lack of scientific basis for research proposals	- Components, shapes	- To have a virtual machine which translates symbolic info about the shape into visual.	- A core issue is to have a process that can integrate shape, and mechanical and software functionality, in a single design language/process, to enable co-design across domains (e.g., "passive walkers", that exploit structure and obviate need for controllers).
	- **What's broken is that there are lots of small specific tools that work well but moving between them does not work well they cannot talk or work together** - **Do not have a way to regulate or automate multiple representation often has translation problem**	- **Dimensional tolerances are fundamental, if components do not fit it adds delay and cost** - **It would be best to go for a relatively lean cut down common core of ontology that managed to do most of the translation enough to persuade people to invest but also not too specific people couldn't relate to it**	- **How to modify an existing design to take into account new factors and do it in a way that's traceable and error free (or verifiable)?**	
	Improved computational support for design generation activities	- *Improved computational support for designing in visual [and possibly other] visual styles* - *Stitching applications within different contexts together improve transformation increase flexibility*	- *Do you need universal grammar between? It is easier if you restrict the domain if you can do fundamentals on prototype.*	- *Implementing vision-based computations on symbolic machines is challenging – a more feasible solution may be to use a different kind of machine*
Output/ Deliverables		- Point solutions for design systems with some interoperability	- Architecture has had some success in applying shape grammars and could inform a pre-killer app to guide the development visual machines	- Killer app
Research		- Interdisciplinary group discussion		
		- **Use two models/tools often used and may give conflicting results and iterate between them to see results and figure out how to include them try and integrate in two ways and e.g. dimension and tolerances could be one tool used**	- **Design systems interact with each other to hook potential clients. Need to find a couple of important systems and integrate in a more automated way: needs to be driven by industrialists**	
		- *Seeing and reasoning*	- *More knowledge on the cognitive processes fed to the designers* - *Look for a code for the visual representation, analogous to the DNA code.*	- *developing visual machine that is easy to use since there is no point having a great tool set if it is not used* - *It is important it is scalable: works for small scale things but then scale it up so that next generation can be assisted by it*
Opportunity		- incremental improvements to product development processes	- Step improvements to product development processes - 3D printing	- More general way to apply design theory in other contexts - 4D printing
		- **With respect to time, cost, quality, responsiveness**		
		- *With respect to creativity & innovation*		

5 Concluding remarks

At the end of the meeting, each participant was invited to make a closing comment. This section provides an overview of these comments.

- Regarding the whole group as a design, and each individual as a design description, all talking, not always consistently, how could you handle this plurality, flexibility, change and open-endedness formally? The group did not have a common core but was still talking. Need to design the theory to start on the journey of this design process. For example, what is a function? Designers are people, and only one kind of stakeholder in the process. The common core is a domain.
- What is needed is a "go anywhere" ontology that can be applied in different contexts. The ability to embed different structures of the same shape would lead to step improvements in product development processes. The wider impact would be a more general way to apply design in different contexts. The task is feasible, but needs theory behind it.
- Does developing a design support system require a general model of design? Visual machines would be able to compute visually rather than just symbolically. Their development needs some theoretical guidance; there are many questions about design theory that get stuck and keep going round. Theoretical guidance, rather than a theory per se, may be more feasible.
- Whether we could do this feasibly [i.e., create a visual machine] is an open question. Could such a machine be created incrementally?
- A common theme from the discussions is that tools need to integrate. What is the research question?
- A design theory alone risks being too lightweight. We need to include more design disciplines (e.g., industrial design) and look outside the design sector for inspiration and user pull. Manufacturing conferences could be a useful starting point. For example, Factory 2050 (http://factory2050conference.com/) addressed questions to which the discussions in this workshop could contribute.
- Do not think that it is feasible to create a specific enough theory to be of any benefit because the theory would be too broad. While it may not be possible to have one theory we could end up with multiple theories. A key challenge would lie in how different perspectives are brought together.
- It would be feasible to create a design theory but the question is which theoretical base to build on? Should it be multiple theories?
- A design theory would definitely be very useful and valuable, but unsure if it is feasible. It would be more feasible to develop incrementally one over the next 30 years or so. Thoughts from a visual workshop could be interesting.
- Do not subscribe to one theory. A group of separate but connected theories and solutions would add more value. More than one theory of design could end up with different theories, which is fine, do not think we need to see all connections.
- It would be feasible to have a group of theories that could add value. We need to start on a design journey to get to the theory. Having a theory will give benefits such as giving a baseline. If we get examples and can be self-consistent within ourselves around what we are describing then that would be a step forward. Use road-mapping structure to create draft white paper, not touched on the cognitive process much.
- Reasoning under conditions of uncertainty is doable in principle. Better ways to talk to each other are needed.
- Philosopher: 'Likes to think it could be feasible, but questions whether you can ever come up with something that will keep everybody satisfied'.
- Talk around integrated theory or frame. This speaker would have liked to see bits of theories to help some of the doing in computing, engineering and other disciplines. Not sure about whole but pieces of it, feel in principle it is possible but need to be very careful about the way the steps are mapped out.

- Some sort of theoretical framework seems feasible but there may be many within one, common language between subjects, still needs to be further work on defining terms, still a lot of questions being asked and incremental progress to move forward.
- Some sort of framework seems feasible. A common language and further work on defining our terms would be a good starting point.
- The idea of a visual machine is exciting. Establishing a working methodology would be a useful starting point. Question is can we actually achieve this ambitious goal? It is feasible as it can be done incrementally, lay down basic design then address more detail; visual machine sounds good, a workshop to getting it working and made would be of interest.

6 Conclusions

The need to accommodate the existence of multiple models and representations of designs and designing stood out through the workshop. The discussions highlighted three areas where there were seen to be significant opportunities for further work: embedding as a means of relating shape, structure and function, the idea of a visual machine to support designers working with visual objects, and theories developed with a need /end users in mind. These, along with opportunities and areas for further development, are detailed further in this section.

6.1 Embedding as a means of relating shape, structure and function

Embedding has been documented since the 1930s in the mathematics literature. Descriptions of concrete applications are less common but do occur, e.g., in shape grammars. Methods to enable the robust implementation of embedding for use in real-world applications remains an open research issue. Embedding could be a mechanism by which structures could be derived. Taking the example of one rectangle or two squares provided earlier, both definitions (in red with a grey fill pattern and blue with a hatch fill pattern in Figure 4) could be created by embedding either one in the given shape.

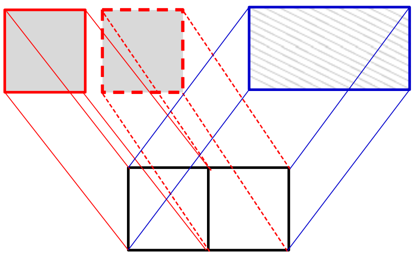


Figure 4: Embedding example

A general-purpose implementation of embedding could revolutionize the way people think about engineering descriptions and their dynamic relationships, and could lead to opportunities for using shape as a means of mediating across design descriptions and structures. For engineering design researchers, embedding could be used to embed design structures, e.g., bills of materials, function structures and service process structures, in design definitions. More widely, there are potential applications of embedding in other disciplines where there is a need to create relationships across scales. For example, in medical physics there are opportunities for embedding in creating cross-scale models of molecules, and opportunities lie in the "big data" area where the ability to embed structure could provide new ways of interpreting large data sets. From Figure 3, if representation schemes are regarded as a form of meta-data, and so themselves data, then embedding could provide a way of defining new meta-models on the fly. This would be an interesting capability to have in that it would enable the repurposing of design descriptions (and other data) defined using one representation scheme for use by applications that need a different representation scheme. For example, could a wireframe representation be embedded into a solid model and used by applications designed to operate on wireframe models?

6.2 Visual machines

Engineering design needs visual shape computation possibilities. The idea of creating a new kind of computer, one that operates visually rather than symbolically, came from group discussions on Day 2. The idea arose from issues related to current CAD systems that operate symbolically and produce visual outputs. However, these visual outputs are presentations (including both explicit and emergent elements) that can be viewed and manipulated by human users but only in a very limited way, computationally. As a result, current CAD systems do not support, and often inhibit, designers' creative flows. This led to the identification of a need for computational design systems that can cope with ambiguity and emergence. A visual machine will be one that can work with emergence, e.g., machines that can "see" when working with designed shapes. Embedding[7] is likely to be a prerequisite for such a machine if implemented using conventional computing infrastructures but not necessarily if such machines were built on new kinds of (currently unconventional) computational infrastructure.

Given this ambitious goal, establishing a working methodology would be a useful starting point to determine the technical feasibility of such a machine and demonstrate its potential value. If it is feasible then an incremental methodology, to lay down basic design principles and engage with potential end users and supporters, would be an effective approach. The content of Figure 5 was extracted from an email received from one of the participants soon after the meeting. It provides a schema for design, implementation and use that would be part of a collection of wider cycles of redesign and improvement. Table 3 provides a roadmap for the development of such machines, developed in writing this paper.

- Step 1: Formulate requirements. What is to be done?
- Step 2: Trace requirements to features: how the features of the item to be designed (thing or process) accomplish what is required (modus operandi)
- Step 3: Work out how to produce or implement or embody those features
- Step 4: Manufacture or realize the items to do that
- Step 5: Operate it as required
- Step 6: Maintain it as required (fuel, repair, etc.)
- Step 7: Retire it when no longer required

Figure 5: Method for the establishment of a visual machine

Table 3: A roadmap for the development of visual machines

Timescale	Past	Now	Near Future	Far future
Goal	- CAD, advanced engineering simulations & design visualisations	- CAD – symbolic and visual output but output is a presentation that cannot be manipulated computationally	- Embedding of shapes and structures into design visualisations	- Visual machines that can embrace emergence & ambiguity
Challenges/ Drivers	- Only human users can operate on design visualisations to generate new designs	- Need to demonstrate the potential of embedding in ways that are understandable - Need for critical mass of researchers, both developers and users		- Embedding of the visual into the symbolic
Output/ Deliverables	- CAD, design analysis and simulation systems	- White paper (this document)	- Publications based on White Paper - Demonstrator solutions	
Research	- Computational geometry - Simulation/ optimisation - Engineering information systems		- EPSRC Designing the Future	
Opportunity	- Dramatically improved computational support for humans designing			- Dramatically improved computational support for humans designing

[7] For example, if two squares are butted together, the system would be able to "see" the emergent rectangle, and allow users to manipulate that, also.

6.3 Requirements for design theory

There are likely to be multiple theories covering different perspectives of engineering design. In the meeting, two broad perspectives were identified: design process and design description.

Two process perspectives were discussed. Prescriptive and descriptive models of design processes, of which there are many in the literature, highlight the development of design descriptions from needs and requirements through to detailed design definitions to support downstream processes such as manufacturing. In engineering practice, these kinds of processes are supported by product data and life-cycle management systems. Design activity models focus on how designs are created. Computational tools to support these models are at the early stages of their development; spatial grammar-based systems (including shape and set grammars) are examples of such systems. With respect to requirements for design theory, key needs are support for zig-zagging between functional and physical domains, and accommodation of commonalities and variation across human designers, industries, and problem domains.

Design definition perspectives need to cover shape, capacity (of which function is a kind) and, for physical products, material. A theory of capacities could build on activity theory to support the definition of use plans and embedding to relate these plans to shape descriptions. Design synthesis requires means of dealing with emergence and the description of design structures needs to include both hierarchies and lattices. Designers need to be free to work with whatever they can see in their designs and there is a need to recognise that engineering designers design processes and supply networks as well as physical products.

7 Progress against Anticipated Outputs

Four key outputs were identified in the proposal that funded this meeting. Progress against each of these is outlined in this section.

1) *A white paper that included (a) technical requirements and goals that could inform developments in current design practice and education and (b) a roadmap for the development of future research opportunities. Funds were requested for four Leeds Masters students to act as rapporteurs in the workshop.*
   This document is the white paper. The rapporteurs (one from each of the following courses: design, product design, mechanical engineering and computing) attended on Day 2. Their reports form Appendix E of this report.

2) *Use outcomes from the workshop to inform the development of an MDes module on Design Research and in the development of new discovery modules.*
   The workshop had an immediate impact through a workshop ("Alberti's landscapes: calculating with insight in art and design") delivered by Prof George Stiny to students on the MDes module the day after the meeting. The discussions are also informing the design of a new undergraduate module on designing and making; current thinking is that this will be a Level 2 discovery module and so available to all Leeds undergraduates.

3) *Dissemination of results*
   Outcomes from the meeting have been disseminated in two international research fora.
   - McKay, A. (2015) "Design Research Spectrum from Transformational Fundamentals to Design Innovation". invited presentation to the International Design Symposium. http://www.design.kyoto-u.ac.jp/idskyoto-2015/. Kyoto, Japan. 13-14 March 2015.
   - Thomas, B.G. (2015) at Bridges 2015. http://bridgesmathart.org/bridges-2015/. Baltimore, USA. July 2015 while presenting the following two papers:
     - Thomas BG; Twarock R; Valiunas M; Zappa E (2015). *Nested Polytopes with Non-crystallographic Symmetry Induced by Projection*. Proceedings of Bridges 2015: Mathematics, Music, Art, Architecture, Culture. pp.167–174. <http://archive.bridgesmathart.org/2015/bridges2015-167.html>

- Arstall A; Thomas BG; Twarock R; Zappa E (2015). *Expandohedra: Modeling Structural Transitions of a Viral Capsid*. Proceedings of Bridges 2015: Mathematics, Music, Art, Architecture, Culture. pp. 471–474. <http://archive.bridgesmathart.org/2015/bridges2015-471.html>

4) *Joint research*

   A research grant has been funded through the EPSRC Design the Future call.
   2015-17. "Embedding Design Structures In Engineering Information". Alison McKay (Mechanical Engineering), Mark Robinson (LUBS), David Hogg (Computing), Chris Earl (Open University), Rolls-Royce, four Yorkshire-based SMEs. An EPSRC Design the Future programme project.
   Funding for 2 x PDRAs x 18 months.

   Other proposals are under development. In addition, several opportunities for further work were identified including a workshop to develop the idea of a visual machine, establishment of a White Rose design research community and more interactions with manufacturing industries to create a user pull for design research in this area.

8 Acknowledgements

The meeting was funded by the University of Leeds Fund for International Research Collaboration and the project web site (http://ris.leeds.ac.uk/international/irca/design-shape-and-structure) was established with support from the University of Leeds International Office. Prof Isobel Pollock provided facilitation for the discussions on the second day along with four student rapporteurs (Diarmaid Clery (Mechanical Engineering), Lucia Gomez Alvarez (Computing), Rowena Madar (Product Design) and Joanna Wilk (Design)) whose notes informed the writing of this report. Administrative support for the meeting was provided by Michelle Byrne (Mechanical Engineering). An invitation and travel support from Prof Tetsuo Sarawagi, School of Design at Kyoto University, to present preliminary findings from the workshop at the 2015 Kyoto International Design Symposium were instrumental in the design and development of this white paper.

Appendix A: Participants

Name	Discipline	Organisation
Prof Saeema Ahmed-Kristensen	Engineering Design	Technical University of Denmark
Eduardo Castro e Costa	Architecture	University of Lisbon
Dr Hau Hing Chau	Engineering Design	University of Leeds
Diarmaid Clery	Student rapporteur	University of Leeds (Mech Eng)
em Prof Alan de Pennington **	Engineering	University of Leeds
Prof Jose Duarte	Architecture	University of Lisbon
Prof Chris Earl	Design	Open University
Lucia Gomez Alvarez	Student rapporteur	University of Leeds (Computing)
Laura Harrison	Design	Open University
Prof David Hogg	Computing	University of Leeds
Corinna Königseder	Engineering	ETH Zurich
Rowena Madar	Student rapporteur	University of Leeds (Product Design)
Marta Perez Mata	Engineering Design	Technical University of Denmark
Prof Alison McKay **	Engineering	University of Leeds
Prof Isobel Pollock	Facilitator	University of Leeds (Visiting Professor)
Prof Keith Ridgway	Manufacturing	University of Sheffield
Prof Peter Simons	Philosophy	Trinity College Dublin
Prof Susan Stepney	Computing	University of York
Prof George Stiny **	Architecture	MIT
Dr Briony Thomas **	Design/Visual Arts	University of Leeds
Dr Pieter Vermaas	Philosophy	TU Delft
Joanna Wilk	Student rapporteur	University of Leeds (Design)
NOT ATTENDING		
Prof Johan Malmqvist	Engineering	Chalmers University
Prof Kristina Shea	Engineering	ETH Zurich

** Members of the organising committee

Appendix B: Meeting agenda

Day 1: Tuesday 18th November 2014

09.15	COFFEE available	
09.30	Welcome	
09.45	Rationale for the meeting (10 mins)	Alison McKay
	Discussion on why we selected the papers we did and what we learnt from reading each other's papers	All
	Response (10 mins)	George Stiny
11.00	COFFEE	
11.15	Depart (11.30am, latest) for trip to Yorkshire Sculpture Park	www.ysp.co.uk/
	LUNCH (13.00)	
	Walk & talk (Details to be provided during the meeting)	
16.00	Depart Yorkshire Sculpture Park	
17.00	Return to campus	
18.00	Dinner at University House (18.00 for 18.30)	

DAY 2: Wednesday 19th November 2014

09.00	Rationale, structure, goals and ground rules for the day	Prof Isobel Pollock, Facilitator
09.15	Group 1	
10.00	2 minute feedback sessions by rapporteurs & switch groups	
10.15	COFFEE/BREAK	
10.30	Group 2	
11.15	2 minute feedback sessions by rapporteurs & switch groups	
11.30	Group 3	
12.15	2 minute feedback sessions by rapporteurs & switch groups	
12.30	LUNCH & informal discussions on development of white paper	
13.30	Interactive structuring/ road-mapping for the development of the white paper	
14.00	Identify next steps & opportunities	
15.00	Reflection on the White Paper design & development WORKING COFFEE BREAK	
15.30	Wrap-up - Agree next steps and timescales	Alison & George
16.00	CLOSE	

Appendix C: Papers shared and provided in advance of the meeting (in chronological order)

Stiny, G. (1982). “Spatial relations and grammars”. Environment and Planning B. Vol. 9, 113-114

Stiny, G. (1998). “New ways to look at things”. Environment and Planning B. Anniversary Issue, 68-75

Earl, C.F. (1999). “Generated designs: Structure and composition”. AI EDAM, 13, 277-285

Prats, M,, Earl, C., Garner, S., Jowers, I. (2006). “Shape exploration of designs in a style: Toward generation of product designs”. AI EDAM, 20, pp 201-215.

Thomas, B.G., Hann, M.A. (2007) “Patterned Polyhedra: Tiling the Platonic Solids”. Bridges Donostia: Mathematical Connections in Art, Music and Science Tarquin Publications.: 195-202.

Thomas, B.G., Hann, M.A. (2008) “Patterning by Projection: Tiling the Dodecahedron and other Solids.” Bridges Leeuwarden: Mathematical Connections in Art, Music and Architecture Tarquin Publications.: 101-108.

Achiche, S., Ahmed, S. (2009) “Investigating the Relationships between Quantitative and Qualitative Properties of 3D Shapes using Fuzzy Logic Models”. In *International Conference on Research into Design, ICoRD’09.*

Montenegro N., Beirão J.Duarte J. (2012) “Describing and locating public open spaces in urban planning”. International Journal of Design Sciences and Technology, 19:2, 91-104.

Beirão, J., Duarte, J., Stouffs, R., Bekkering, H. (2012) “Designing with urban induction patterns: a methodological approach”. *Environment and Planning B: Planning and Design* 2012, volume 39, 665 – 682.

Stepney, S., Diaconescu, A., Doursat, R., Giavitto, J-L., Kowaliw, T., Leyser, O., MacLennan, B., Michel, O., Miller, J.F., Nikolic, I., Spicher, A., Teuscher, C., Tufte, G., Vico, F.J., Yamamoto, L. (2012) “Gardening Cyber-Physical Systems”. In *Unconventional Computation and Natural Computation 2012, Orleans, France, September 2012*. LNCS 7445:237-238. Springer, 2012

Simons, P. (2013) “Varieties of Parthood: Ontology learns from Engineering”. *Philosophy and Engineering: Reflections on Practice, Principles and Process.* Diane Michelfelder, Natasha McCarthy, David E. Goldberg (eds.) Dordrecht: Springer, 151–163.

McKay, A., Stiny, G.N., de Pennington, A. (2015) “Principles for the definition of design structures”. International Journal of Computer Integrated Manufacturing.

Vermaas, P.E. (2014) “Functional Decomposition and Functional Parthood: Design Research Colliding with Mereology”. Philosophy Department, Delft University of Technology. Available on request from p.e.vermaas@tudelft.nl

Appendix D: References identified during the meeting

Andreasen, M. M. 2011. 45 Years with design methodology. Journal of Engineering Design, 22, 293-332.

Brynjolfsson, E. & Hitt, L. M. 1998. Beyond the productivity paradox. Commun. ACM, 41, 49-55.

Cagan, J. & Vogel, C. M. 2012. Creating Breakthrough Products: Revealing the Secrets that Drive Global Innovation, FT Press.

Dement, C. W., Mairet, C. E., DeWitt, S. E. & Slusser, R. W. 2001. Mereos. Available from: http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA397558: Ontek Corporation.

McKay, A., Erens, F. & Bloor, M. S. 1996. Relating Product Definition and Product Variety. Research in Engineering Design, 2, 63-80.

Montenegro, N. C. & Duarte, J. P. 2009. Computational Ontology of Urban Design: Towards a City Information Model eCAADe 27: Education and research in computer aided architectural design in Europe. Istanbul, Turkey.

Pahl, G., Beitz, W., Feldhusen, J. & Grote, K.-H. 2007. Engineering Design: A Systematic Approach Springer-Verlag.

Schon, D. A. W., G. 1992. Kinds of seeing and their functions in designing. Design Studies, 13:2, 135--156.

Simons, P. 2013. Alfred North Whitehead’s Process and Reality. Topoi, 1-9.

Stiny, G. 1990. What is a design? Environment and Planning B, 17, 97-103.

Suh, N. 1990. The Principles of Design, Oxford University Press.

Ulrich, K. T. & Eppinger, S. D. 2007. Product Design and Development, McGraw-Hill.

Appendix E: Rapporteurs’ reports

Four student rapporteurs were recruited to participate in the second day of the workshop. Their role was to take notes in the breakout groups.

- Diamaid Clery, Engineering, PhD Researcher, EPSRC Doctoral Training Centre in Bioenergy
- Lucía Gómez Álvarez, Masters student in Advanced Computer Science (Intelligent Systems)
- Joanna Wilk, part-time PhD student in Graphic Design
- Rowena Madar, Final year MDes student in Product Design

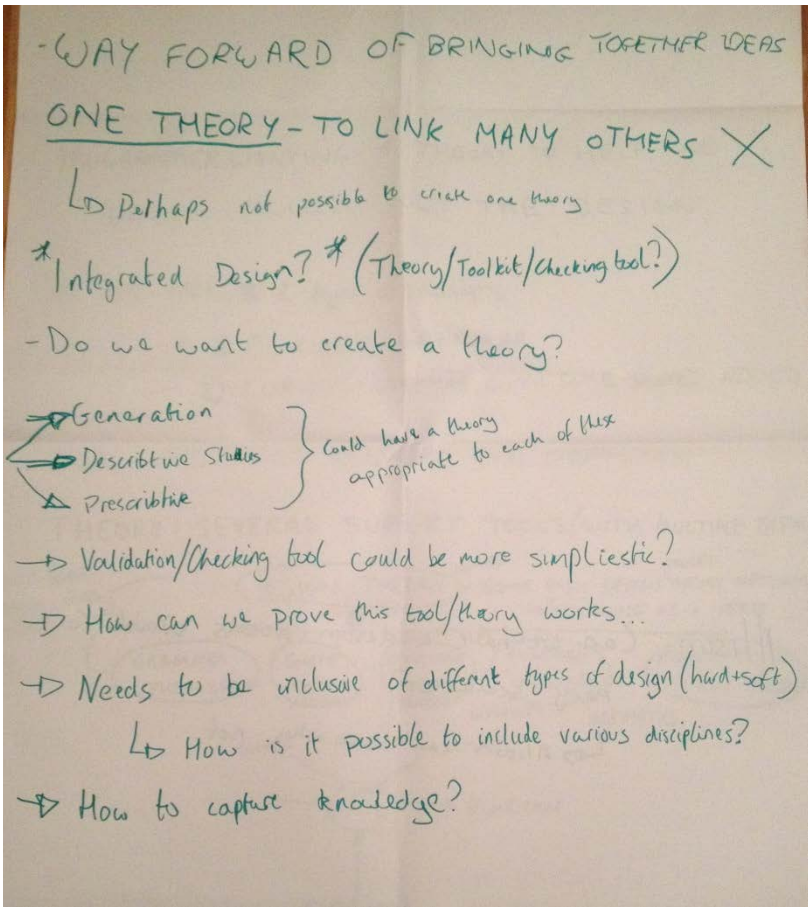
The reports they produced are provided in the following pages of.pdf versions of the document.

Diarmaid Clery, Mechanical Engineering

Engineering Design Theory Meeting: Shape & Structure

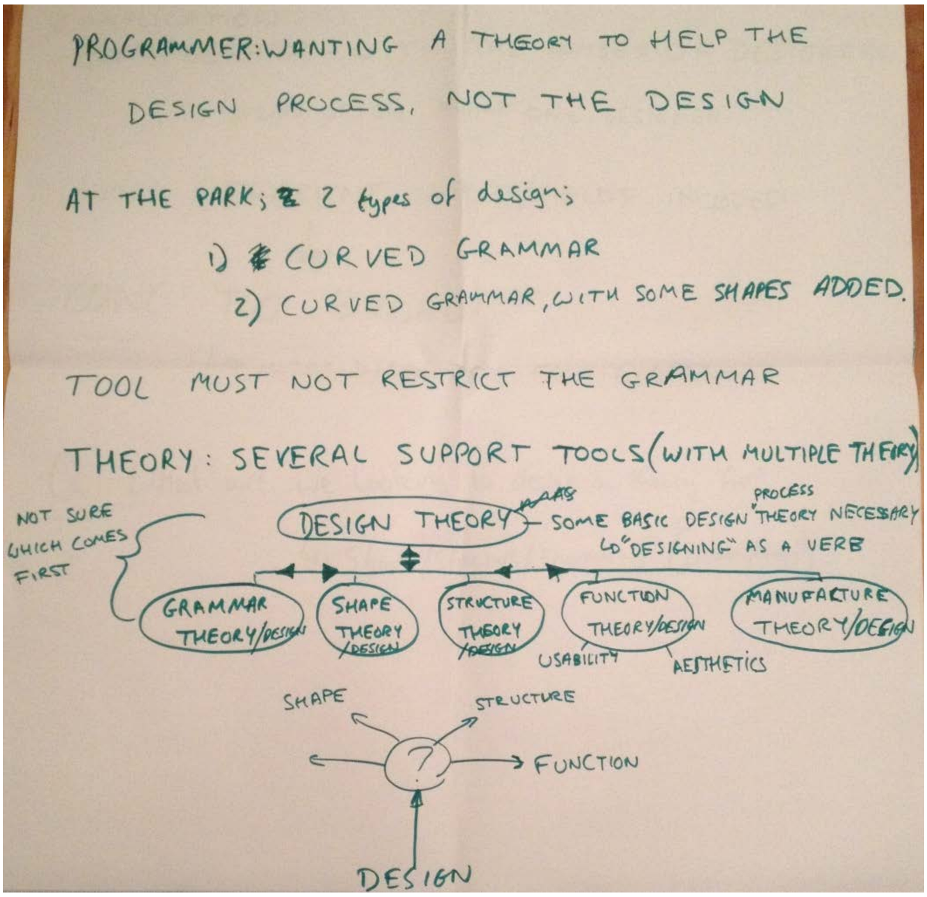
Location: Cloberry & St George rooms in University House
Date: Wednesday 18th November 2014
Notes: Taken and typed by Diarmaid Clery

Group 1 Discussion: Most important message from the white paper? And why?

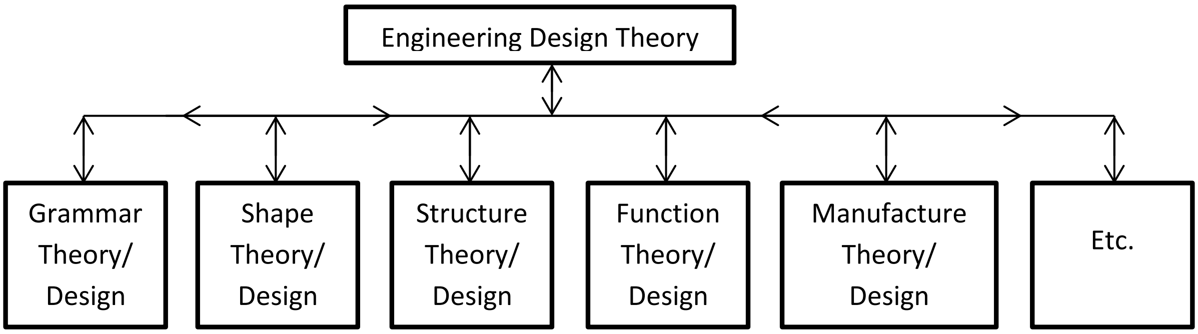


- Questioning whether it is a 'theory' that we want to create?
- Could it be a toolkit/validation checking tool instead? – As this may be simpler.
- The group agreed that there are several different fields and contexts that use design – these all need to be incorporated in this design theory.
- Also decided that the theory needs to include hard and soft engineering design – but unsure how best to do this.
- Questions were raised regarding how the 'theory/tool' would be proved to work.
- Final point questioned if this theory was aiming to capture the knowledge/innovation in engineering design.

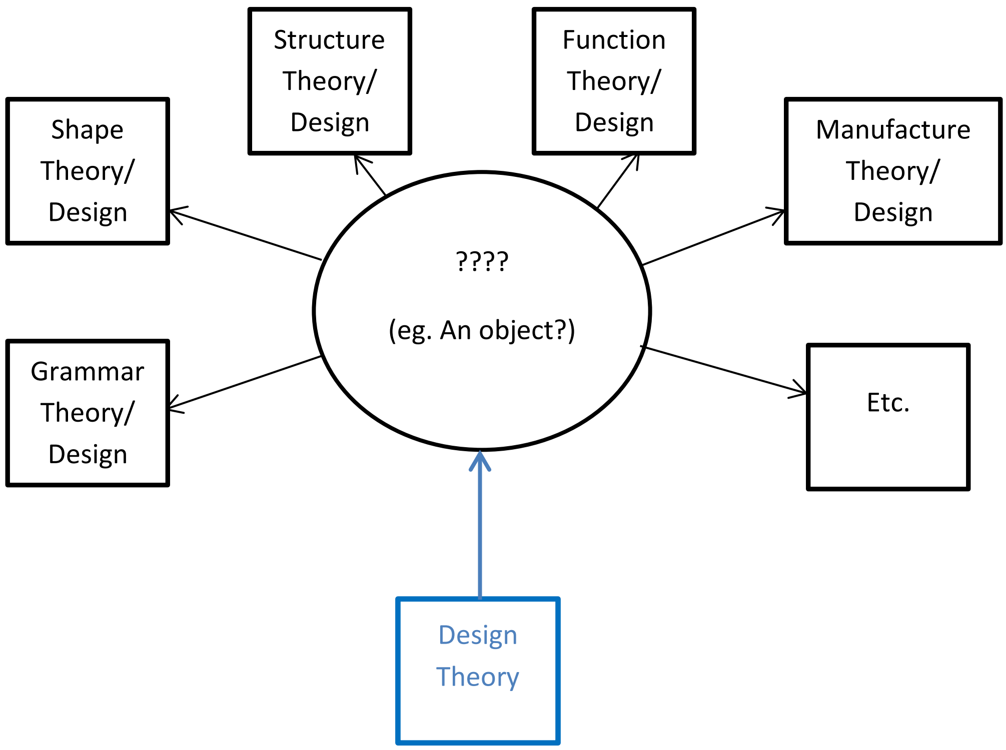
Group 2 Discussion:



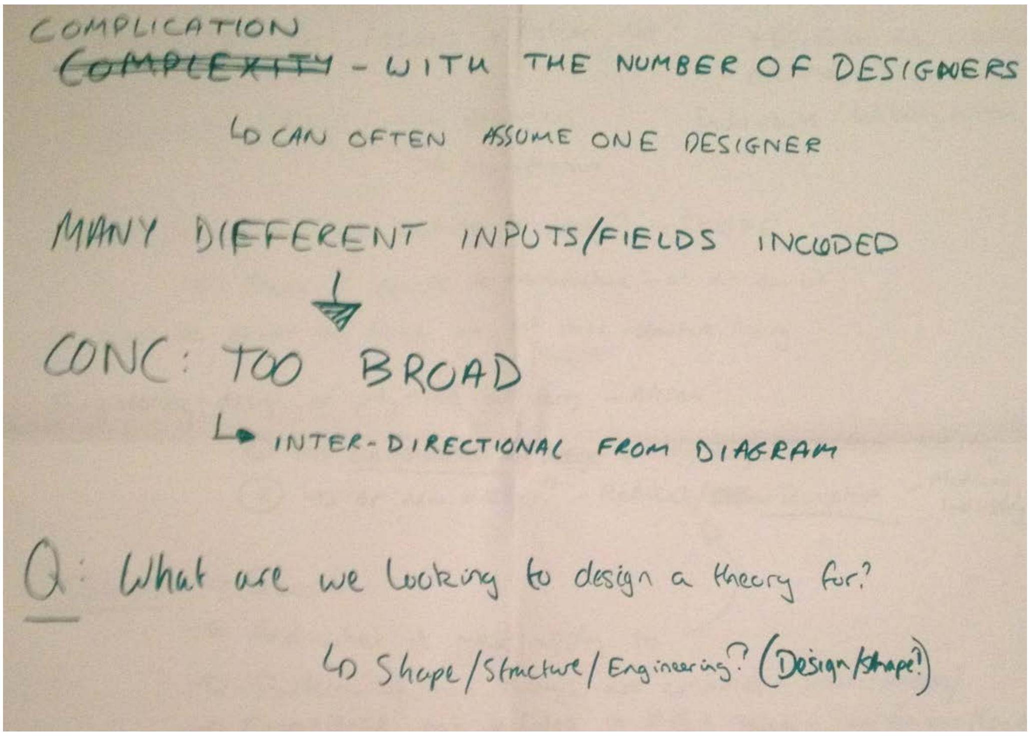
- Conversation began with an individual programmer within the group explaining that they wanted ‘a theory to help the design process, not the design’ – as this would help with the programming aspect of design.
- The conversation developed into discussions on the shapes and designs seen at Yorkshire Sculpture Park the previous day – observed as either curved grammar, or curved grammar with something added.
- It was agreed that the task in hand cannot restrict grammar.
- Conversation changed to focus on the question in hand: It was identified that the ‘theory’ created could consist of several support tools, with multiple theories underneath.
- This developed into the tree shaped diagram seen below (with the conflicting arrows representing the cross over between different discipline theories):



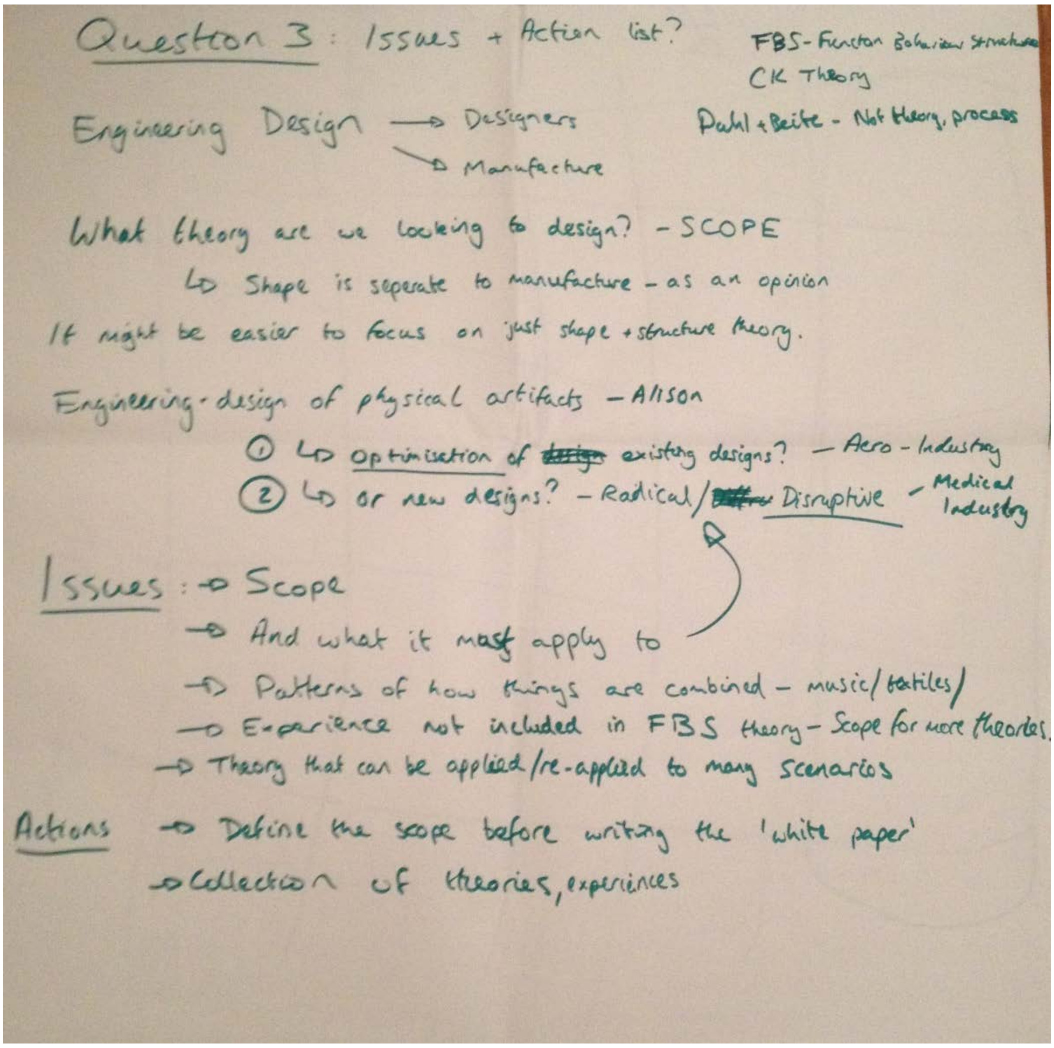
- Confusion arose around which should come first, resulting in the following diagram being suggested as an alternative:



- Conclusion: There may be too many different inputs to this broad ‘theory’.
- Complications may arise with the number of designers – given design theories are normally aimed at an individual.
- Questioned what *exactly* ‘the design theory' scope was? – Shape/Structure/Engineering/ Design?



Group 3 Discussion: List some Issues and Actions that may occur



- The Scope of the paper was discussed in further detail.
- Alison: ‘Engineering design of physical artefacts’.
- Two types of Engineering design were identified: ‘*Optimisation’* and ‘*Disruptive/Radical’*
- Decided that the scope would need to include both of those above.
- Musical and textile combining of patterns were discussed.
- Existing theories were identified such as:
    - FBS theory – Function, Behaviour, Structure
    - CK Theory
    - Pahl and Beitz – Manufacturing Process Theory
- Experience was identified as something that is missed by the above theories.
- The scope for more theories was identified.
- Actions included:
    - Collecting existing theories, experiences, etc.
    - Defining the scope of the paper prior to writing.

Post-Lunch - Interactive structuring/ road-mapping:

- The team used the roadmap below to identify challenges, outputs, reseach, oppourtunity and visions that may be applicable to the suggested theory.

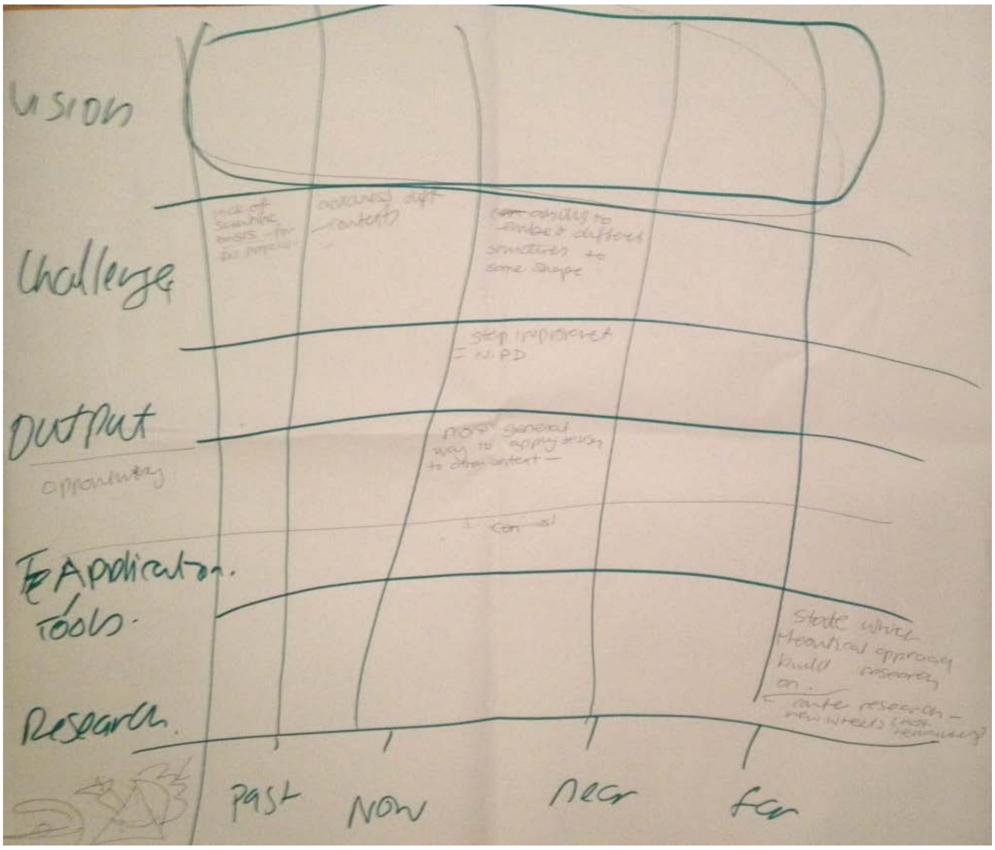


Table 1. Details of the drawing above are transposed into this table for clarity

Timescale	Past	Now	Near Future	Far future
Challenges/Drivers	Lack of scientific basis – for research proposals	Applications within different contexts	Ability to embed different structures to some shape	Uptake of the design theory in different scenarios
Output/Deliverables		White paper	Step improvement N.P.D.	
Research				State which theoretical approach we can build research on, or follow
Opportunity			More general way to apply design theory in other contexts	
Vision				An application that can vary how a programme follows design theory depending on what the user wants to get out

Reflection on the day and feasibility of the task in hand:

Thoughts as we went around the table:

- ‘The task is feasible, but needs theory behind it’
- ‘A design theory would definitely be very useful and valuable, but unsure if it is feasible’
- ‘Feasible to incrementally develop over the next 30 years or so. Thoughts from a visual workshop could be interesting’
- Manufacturer (Keith?): ‘Doesn’t think that it is feasible to create anything specific enough to be of any benefit’ i.e. the theory would be to broad
- ‘Feasible, but the question is which theoretical base to build on, should it be multiple theories?’
- Philosopher: ‘Likes to think it could be feasible, but questions whether you can ever come up with something that will keep everybody satisfied’
- ‘A theoretical framework is my favourite, with several other theories under that’
- George: ‘Impressed with the different backgrounds of the people present, and there respect for each other. Sees everyone in the room as a key part of the design for this task’

Joanna Wilk, Design

Group discussion GROUP 1

What, for you, is the most important learning (message), that you would like to see in the White Paper, and why?

Discussed issues:

1. Relationship: Function - GENERAL SHAPE
2. Differences in design processes: CAD vs ShapeGrabber
3. Software limitation, introducing required adjustments.
4. Embedding the information into the design/ shape structure/model

Key targets:

- Describing the shape
- Making process faster, smoother
- Sharing the description, information about the shape, not only the final shape
- Adding attributes

Problem:

Not enough information in the CAD paradigm, which is more constraining that releasing.

Group 2

Express a vision of where we would like to be, and what could the theory do for you.

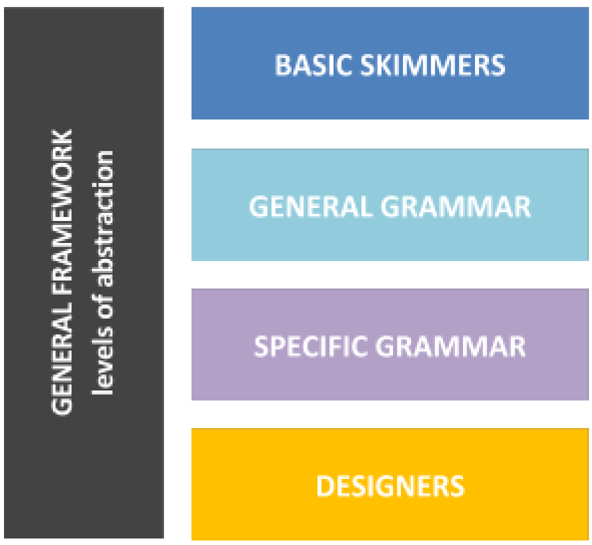


Figure 1. General framework - levels of abstraction

General framework - levels of abstraction

1. Basic Skimmers
2. General Grammar
3. Specific Grammar
4. Designers

Questions:

- What are critical problems on specific levels?
- Why is the visual representation important?
- How to transform the shape?
- How we can bridge the gap between the visual and symbolic information?
- How to translate this information for the shape machine?

CORE PROBLEM: TO BE ABLE TO IMPLEMENT GENERAL NOTION OF EMBEDDING

Discussed issues:

- Embedding is about the relationship
- Input of computing in design and vice versa
- SHAPE MACHINE - Representing shapes on symbolic machine embedding information about the relationship.
- Science - design

KEY ISSUE: Process of transformation (composing - decomposing - composing) while moving across the levels.

Group 3

'What's broken?'

What are the issues and what is on the action list?

CORE PROBLEM

To be able to implement the general notion of embedding.

1. Develop shape machine that uses visual information.

2. Finding way of visual representation of shape on the symbolic machine.

Conclusion: We need the shape machine that does both; computes visually and symbolically.

KEY: Framework - Levels of abstraction

- Moving between the levels
- Moving inside the levels
- Focussing on the shape
- Shifting descriptions

FINAL CONCLUSION:

WE NEED THE FRAMEWORK THAT ALLOWS SHIFTING DESCRIPTIONS AND MOVING WITHIN AND BETWEEN THE LEVELS.

1. The CORE that is FLEXIBLE and supports people during designing process.
2. Where the innovation happens through changing the rules and moving between the levels.
3. Solving the representation problem
4. Shape + information; the same shape with different meanings attached (information about its character).

Group 4

Technology roadmapping

NOW	NEAREST FUTURE	FUTURE
COMPONENTS SHAPES	STITCHING THESE TOGETHER IMPROVE TRANSFORMATION INCREASE FLEXIBILITY	DEVELOPING VISUAL MACHINE

WHAT is NEEDED:

- Interdisciplinary group discussion
- Seeing and reasoning
- More knowledge on the cognitive processes fed to the designers
- Looking for a code for the visual representation, analogous to the DNA code.

- Increase the framework flexibility
- We need to understand how it works. (Would it work? Is the knowledge on how it constructs enough for it to work)
- general model and technologies

GOAL:

Developing translation tool of realistic demonstration:

- developing the design support system

Core issue in developing it, is to have a virtual machine which translates symbolic info about the shape into visual.

- considering human - machine interactions
- satisfying the needs of the user in better way

Lucía Gómez Álvarez Computing
Alison McKay
14 December 2014

Design: Shape & Structure (decomposition / description)

Brief summary of the discussions undertaken the day 26 November.

Retrieving Knowledge

Study the design history

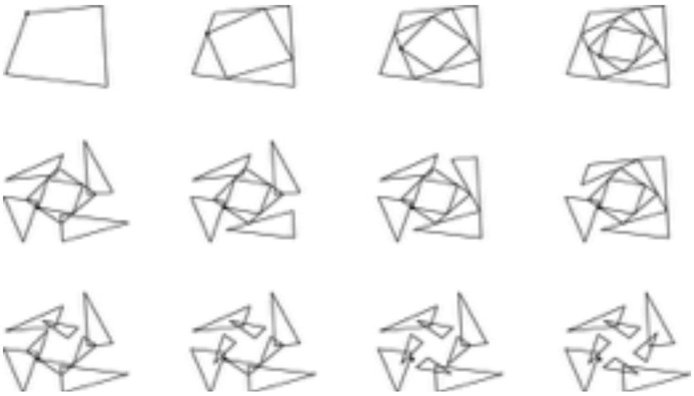
One of the sources of knowledge to take in account and described as a key for understanding design is the analysis of the design history, as well as its origins (art, mechanics, …). Furthermore it would be interesting to look across domains and to take inspirations from other areas like fashion, architecture or computing and the way that they design.

The innovation process

A clear understanding of the innovation process may prevent us, as well, on creating rigid theories that wouldn't promote creativity. In this connection an analogy with music is purposed, where innovation came when the preferred code sequences where cut. In that sense, if we have a grammar of shape we must assure that unexpected combinations can happen, rather than favour uniformity.

Exporting existing grammars

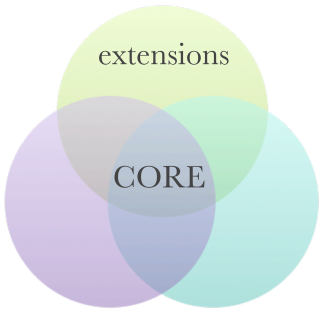
Beyond the general knowledge, there are already existing formal grammars that are working in specific areas such as *shape grammar* in architecture. In the process of building a core it could be interesting to consider exporting these grammars or part of them, abstracting them and making them global.



A focus on the Design Research

Look for a core that would contain the fundamentals of design and that would have specific extensions for different subjects

In order to undertake a research on design it seems to be desirable to search for something like a core that would cover the central and abstract knowledge about Design, its axioms in some way.



This core would have properties like abstraction, reusability and cohesion, and should be a solid base which will be used to build domain-specific knowledge on.

Nevertheless we must precise that we wonder even if such a core actually exists and if it is findable.

Look for a unified language

The natural output of that core should be a unified language that could express it as well as allow a potential translation of information placed on different knowledge areas or expressed on different representations. The current lack of a proper method of communication seems to be one of the most important issues to address.

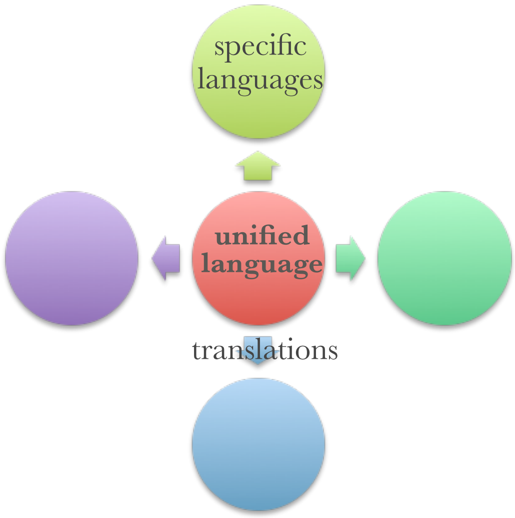
Need for defining specific goals, short term and long term

It doesn't seem to be obvious which are the major challenges neither for the academic world nor for the industry, and a clear understanding of that could be a key to succeed. The firsts proposals are:

- Creation of a slim language that fits everything.
  - Special focus on the areas where designers experiment bigger difficulties.
  - Research on implementing translations to this common core.

- Reutilisation of existing useful tools
- Restriction of the area to simplify the translation problem.

- Selection of a couple of important coding systems that could be improved if they interact between them
  - The selection would be better if it covers part of the common core, instead of a particular problem
  - Try to address the main detected dissatisfaction: the lack of interaction between tools.



The need & possibilities of new tools

Adding value to the current representations/Giving meaning to design

Valuable improvements can be made by adding meaning to the current representations. With this metadata, for example stating the material of some part of 2D model, the computational tools can go lot further facilitating the design process.

Supporting different representations

In order to build a useful tool there is a major need of previous understanding on the multiples models and views that a complex system can have.

If we achieve to build an abstract model from which to extract multiple representations, computational tools will enable the use any of them in an equivalent way, reducing the choice to a design decision.

Ontologies

In addition to the computational solutions, a discrete ontology could facilitate the establishment of the mentioned core, by providing a formal set of its concepts and definitions of the types, properties, and interrelationships of its entities.

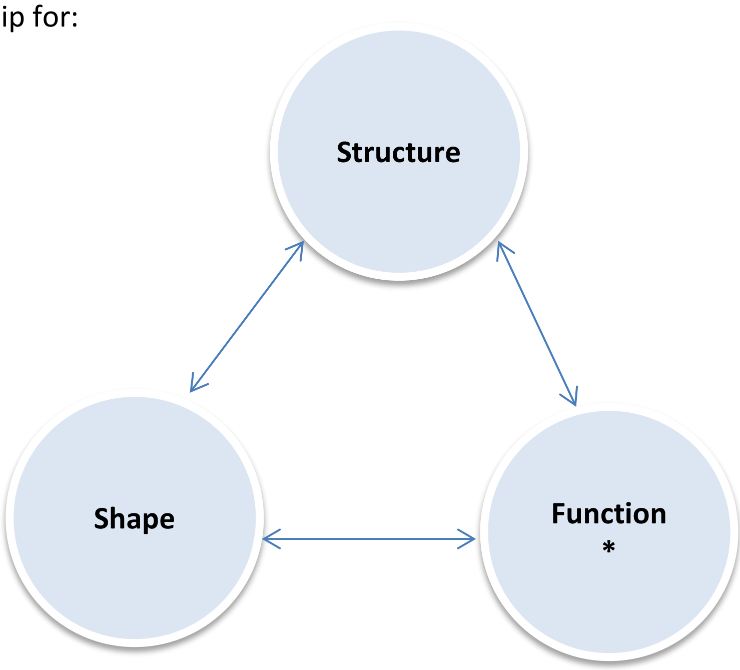
Rowena Madar, Product Design

Discussion 1:

- The approach works for architecture and visual design
- Shape grammar perspective is hard, you use memory for task chunks of recyclable behaviour, needs to be integrated
- Does thinking grammar leave out semantics? Different disciplines have different meanings of the words
- Knowledge needed but not all in the same place, if they are put in rules to be utilised
- Range of parameters wide, - what are the set of values?
- Theory about grammar and use of design against theory of design are different, what are we focusing on?
- Supporting designers with tools needs to stick to cognitive process, descriptive grammar
- Levels applicable, different levels to work on
- Tolerating inconsistency, different views so there probably will be a degree of inconsistency with multiple representations
- Multiple stakeholders involved across subjects showing common knowledge- how many perspectives can you have? How much consistency accepted? A way to share and agree on knowledge aligning is needed

Discussion 2:

Seek relationship for:



- Function needs to be defined; it is ambiguous and not always so clear
- Scope is needed, what defines?
- Define application, will need to be scaled
- Where does material fit into this?

Discussion 3:

Issue 1: Don't know the end result and don't know how to get there

Action 1: there is a want for a universal database that contains function/shape/ that can be applied to domain

Issue 2: The formal aspect is separate from representation aspect

Issue 3: Understanding the core and representation wanted; need to know this before a tool is made

Action 2: **(response to issue 3)** rather than getting hung up on words can maybe use a different representation system for different areas?

Issue 3.1: a common core is a nice idea but is it possible?

Action 3: **(response to issue 3.1)** model with multiple views is still possible, complex systems have multiple models so when talking about a common core it can have different parts with multiple views

Issue 3.2: **(response to action 3)** How do you go about system with various models? Trying to use a single model is easier

Issue 4: More shape grammar knowledge is needed, there is a major gap not enough people know about it or understand it

Action 4: **(In response to issue 4)** formulisation of apparatus, a sort of marketing campaign introducing shape grammar to encourage people to use it. It needs to be turned into a game, something that inspires people to use it. It takes a while to learn to use grammar so using a game means user will invest time in it.

What is really needed to get people using shape grammar is a killer application. If you could have an application that goes straight to retailers it would let people access it easily. 3D printing could be used as part of the application allowing it to go straight to users.

Issue 5: Part of the problem is that anything suggesting creativity being turned into a mechanism is a problem for people

Technology road mapping (afternoon discussion)

Is the system broken or can we do something better?

- One level there is a definite dissatisfaction with functionality of present design systems depends what field it is in (can't talk about architecture) areas usually have subsystems that can be defined with 3D systems the ability to move between systems and then carry out analysis to see if it's strong enough to withstand certain things a range of analyses
- Still frustrations with systems although they have improved a lot over the last years depending on industrial setting you are in, ability to move between them and then tweaking and wanting to go back reconstruction doesn't always work
- Dimension tolerancing pretty fundamental, if components don't come together it adds delay and cost
- What's broken is that there are lots of small specific tools that work well but moving between them doesn't work well they can't talk or work together
- Don't have a way to regulate or automate
- multiple representation often has translation problem

Tool integration:

- How to modify an existing design to take into account new factors and do it in a way that's traceable and as error free as possible?
- Usability, no point having a great tool set if not used
- Need some guidance
- Need to be able to code it up so it is machine implementable
- It is important it is scalable- works for small scale things but then scale it up so that next generation can be assisted by it
- Stands increasing complexity

Deliverables output products:

1. Gigantic computer tool
   - Incremental add ons which makes visualise improvement

2. Research project
   - Use two models/tools often used and may give conflicting results and iterate between them to see results and figure out how to include them
   - try and integrate in two ways and e.g. dimension and tolerancing could be one tool used and then
   - Design systems interacts with each other to hook potential clients find a couple of Important systems which existing customers tell us and improve if work together in a more automated way need to ask Leeds industrialists
   - Looking for scalable pair of applications

3. Database vision:
- all knowledge is compatible and stored somewhere (long way off)
- Common core is a domain e.g. Football
- Do you need universal grammar between? It is easier if you restrict the domain
- If you can do fundamentals on prototype
- If it is two different tools they need to change together so they can both work together in the future
- It would be best to go for a relatively lean cut down common core of ontology that managed to do most of the translation enough to persuade people to invest but also not too specific people couldn't relate to it

Final comments from everyone about the conference:

- Feasible, needs some theoretical guidance many questions about design theory which gets stuck and keeps going round

- Gone highly philosophical and theoretic, need to link sections better

- More interactive theory on structure

- Question is can we actually do this ambitious goal? It is feasible as it can be done incrementally, lay down basic design then address more detail visual machine sounds good a workshop to getting it working and made

- Larger period it should be feasible and has been useful hearing commonalities between different principles that apply to design and tools needed to develop support for theory, some questions are still confusing actual question still uncertain

- We all do design but have no opportunity to look out, maybe add someone else like an industrial designers , not possible to have one theory as it would be so lightweight would end up into multiple theories

- Interesting to hear different perspectives long term probably possible not very sure where to start more discussion is necessary but you have to start somewhere

- Many viewpoints, similar things different words, it is feasible. Integrate more disciplines, define scope, good idea to draft white paper

- More than one theory of design could end up with different theories which is fine, don't think we need to see all connections

- Progress is feasible timescale I'm uncertain about, need theoretic guidance

- Could be a single connected theory of all parts of design, optimistic if it is actually doable what is encouraging is that not only have we found better ways to talk but it has become gradually clearer to what steps can be taken

- Talk around integrated theory or frame (not entirely sure right meaning) I'd like to see bits of theories help some of the doing in computing engineering and other disciplines Not sure about whole but little pieces of it, feel in principle it is possible but need to be very careful about way the steps are mapped out

- Some sort of theoretical framework feasible may be many within one, common language between subjects, still needs to be further work on defining terms, still a lot of questions being asked and incremental progress to move forward

- Think of this group as a design and each as a design description, often talking to each other, think about flexibility in the way they are linked, being connected is a common theme, visual theme is interesting is doable, it can be done without a common core, putting descriptions together talking to people from so many different background disciplines and it has had everyone talking and a design has been made its open ended and always changes, it's great for seeing everyone here discussing

- Visual map would be hard tricky to go from analogue to ontology

- Feasible to have a group of theories that could add value, we need to start on a design journey to get down the theory having a theory will give benefits gives a baseline, if we get examples and can be self-consistent within ourselves around what we are describing would be a step forward. Use road mapping structure to create draft white paper, not touched on the cognitive process much