Interoperability and collaboration in Computational Design: A Grimshaw case study of distributed authoring environments for fast pace design iterations

Authors : Aurélie de Boissieu (ULiège), Andy Watts (Grimshaw)

1. Introduction

While the architectural design process is sometimes romanticized through the starification of a small number of architects individuals and their narratives, the reality of designing a building remains a complex collective process [1, 2]. On-going controversies (in Bruno Latour definition [3]), negotiations and iterations are key in the design activities. Supporting such aspects of the design process is difficult, and digital toolsets dedicated to it are put to hard work. There is no "one-size-fits-all" solution, and each software has its expected uses, strength and weakness. Consequently, in their everyday work, design teams often have to deal with numerous toolsets. Ensuring the quality and the consistency of project's digital models across diverse software and platforms raises multiple difficulties and risks [4–6]. Data loss and failures in transfer, format incompatibility, as well as data inconsistency and tedious manipulations are some of these difficulties, most of which have been ongoing in the industry for decades [6]. These issues have been widely interrogated in terms of data interoperability and data management through BIM (Building Information Modeling) practices [5, 7]. However, whilst interoperability continues to gain attention in academic research, professional practice and software development, the specific interoperability needs and characteristics for early design stages remain poorly documented.

Exchanging data between different design systems usually requires bespoke workflows. These get difficult to implement as the tools multiply, increasing the risk of data loss and tedious modeling tasks. Software limitation can be highly constraining for the design teams, even if the practice intends to streamline processes through working standards, training and best practice recommendations. In particular, Computational Design (CD) is a practice which relies on advanced computation processes to design, optimize and evaluate performances, and relies most of the time on multiple toolsets [8, 9]. Ensuring the consistency and the quality of data exchanges across CD distributed design environments is a challenge for practices, and currently is one of the factors limiting wider adoption. Collaborative CD practices remain scarce, as traditional data exchange and data interoperability toolsets and processes are not fully addressing the inherent challenges [10–13].

This paper proposes an analysis of interoperability processes during fast-paced design iterations in distributed environments. Especially, it interrogates the interoperability requirements and characteristics observed in Computational Design processes. Based on literature review and participant observations, an analysis tool is proposed to explore and describe five case studies from UK-based architecture studio, Grimshaw.

Section 2 of the paper presents the research context in the field of interoperability and computational design. Section 3 exposes the methodology followed, and especially the definition of the proposed analysis matrix. Section 4 presents the application of the analysis matrix on the selected case studies. The results and observations issued from this analysis are described in section 5. Finally, the conclusions and summary of the main scientific contributions of the paper are presented in section 6.

2. Research context

2.1 Computational Design (CD)

CD practices are very diverse, but they share a common foundation: they rely on the power of computation as well as on computational thinking [8]. Through the ever-increasing speed and capacity of computers, computation allows designers to deal with geometries and tasks which were traditionally either very time consuming or too challenging to be accomplished by a human alone [8, 14, 15]. However, CD is not just automating existing traditional processes or tedious tasks; it's about shifting the way designers think and design [8, 9, 14, 16, 17]. Negroponte [18] and Terzidis [17] especially distinguish clearly "computational" practices from "computer-aided" ones (figure 1).

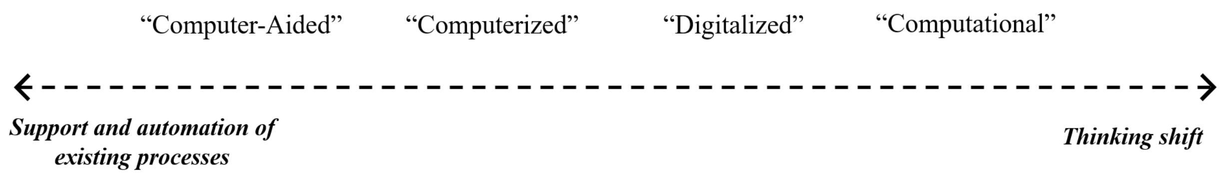


Figure 1 : "Computational" versus "Computer aided » as defined by Negroponte [18] and expended by Terzidis [17]

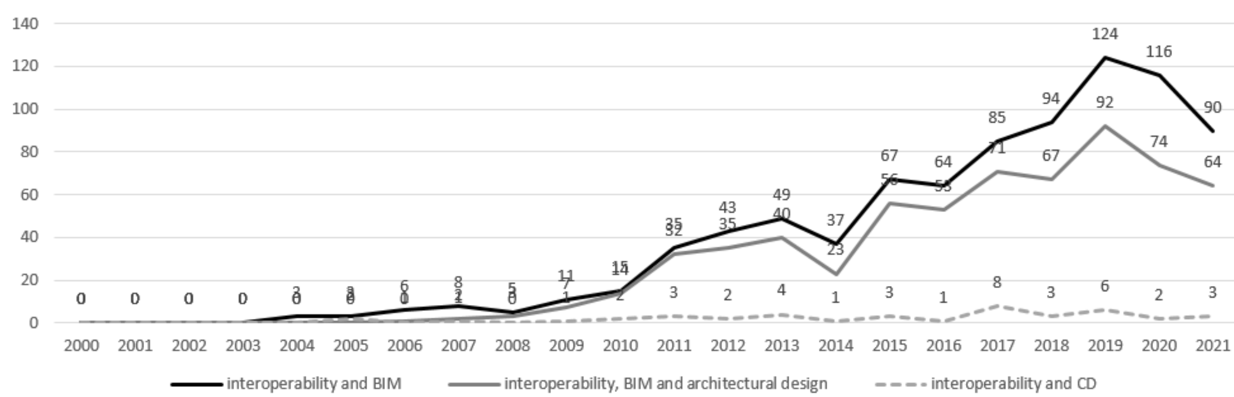
At the foundation of CD is the power of computation, which relies on algorithms. An algorithm is no more than a sequence of instructions[1] [8]. In the context of CD practices, algorithmic technologies and thinking are used to define and manipulate both the flow of data and complex geometries of projects. Designers focus on defining a series of instructions, rules and relationships that gives them better control of the design, enabling the analysis, modification or generation of a wide range of data and geometries. These significantly powerful capabilities open up new fields of possibilities for the AECO[2] industry and its stakeholders [8, 14, 16]. Computational Design is often described through three main subsets of practices: *parametric design* [19–22], *generative design* [23, 24] and *algorithmic design* [8, 17]. The flow of data and geometries implemented in CD processes often relies on different softwares, and has to communicate from one toolset to another, generating interoperability and collaboration issues.

2.2 Interoperability

Interoperability can be defined as : "*The ability of diverse systems (and organizations) to work together seamlessly without data loss and without a special effort."[3]*. Interoperability is not only about data interoperability (ie. sending/receiving readable data from different systems), but also about collaboration [5, 11, 12, 25]. The data exchanged in AECO needs to be expected, agreed, readable and usable [10, 11, 26]. In parallel, the method of interoperability needs to be just as usable, in terms of accessibility, complexity and directionality [5].

2.3 Interoperability for Computational Design

An exploratory bibliometric study shows that if BIM related interoperability researches are relatively well documented, CD related interoperability is more scarcely studied (figure 2).



---

[1] *To be more specific, Menges and Alquist define an algorithm as "a finite sequence of explicit, elementary instructions described in an exact, complete yet general manner"(Menges & Ahlquist, 2011, p. 13).*
[2] Architecture, Engineering, Construction and Operation
[3] https://bimdictionary.com/en/interoperability/1

Figure 02: Evolution of scientific publications on the scientific publication database Scopus[4] about: 1-BIM and interoperability[5], 2-BIM, interoperability and architectural design[6], as well as 3- interoperability and computational design[7]

Interoperability in general remains problematic in the AECO industry [26], and its issues for computational design are rarely studied in scientific researches (figure 2). Current researches interrogates in particular CD authoring tools interoperability with simulations tools [12, 27], CD interoperability in educational context [27] or specific new interoperability solutions or strategies [10, 11, 26, 28]. While these researches are either very specific to niche contexts or very prospective, there is no analytical proposal to describe existing CD interoperability process in order to properly identifies its issues, its strengths and weaknesses.

3. Methodology

3.1 Definition of the analysis matrix

a. Objectives

To address the question of interoperability in CD processes, we propose here an exploratory analysis matrix to allow us to interrogate these processes, to characterise them as well as to identify the strengths and weakness of the technologies and processes implemented. As discussed previously, this research focuses especially on interoperability processes during fast-paced iterative design. In these processes, the data exchanged is principally considered as “work in progress” [29], enabling fast and agile iterations. The focus is also set on interoperability processes for authoring tools[8]. Whilst they are important in fast-paced design processes, exchanges between authoring tools and analysis tools (such as for structural analysis or, carbon impact for example) are not included here. They would have to be studied in further researches.

b. Matrix definition methodology

An analysis matrix has been proposed based on a literature review as well as on exploratory case studies gathered over several years of operational experience of design team needs in architecture practice. These are as follow:

- Exploratory participant observation toward the analysis matrix:

Through several years of experience in the industry and especially four years of CD and BIM practice in architecture studios (from 2013 to 2021), some recurring issues were observed. They have informed the first hypothesis at the origin of the proposed matrix.

- Literature review toward the analysis matrix

We have identified references relevant to CD interoperability analysis in professional practice, those are either: 1- Focused on CD interoperability specific context: [10–12], or 2- General to the AECO interoperability issues but relevant for fast-pace design stages: [5, 7, 26, 30, 31].

c. Criteria

Three categories of criteria are proposed. They are focused on:

- The process’s general **suitability for collaboration** : is a key criteria as identified in literature review [5, 7, 10–12, 26] and participant observations;
- The process’s **reliability** (especially in terms of data richness and stability) : is extensively interrogated in the literature, even if not often for design processes [5, 7, 26, 30, 31], as well as a key observation.

---

[4] https://www.scopus.com/

[5] Research key used : TITLE-ABS-KEY ("interoperability" AND "BIM")

[6] Research key used : TITLE-ABS-KEY ("interoperability" AND "BIM" AND "architectural design")

[7] Research key used : TITLE-ABS-KEY ("interoperability" AND "computational design" OR "parametric design" OR "generative design" OR "algorithmic design")

[8] An “authoring tool”, as in ISO 19760, is a software application which can author a model. It can be distinguished from software application dedicated to specialized simulation, model review or model analysis.

- The process's **agility** (especially in terms of openness and reactivity): is not really tackled in literature review [5, 7, 11, 26], but key in practice as in identified during the explorative participant observations.

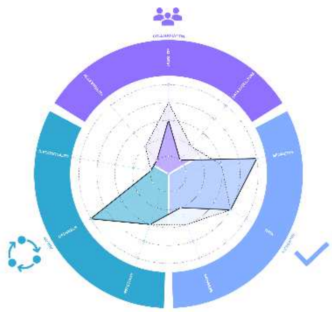
Each of these categories has three targets. For each target a scoring scale enables the assessment of the different case studies, whether they reach the target or not. In these scales, 1 is a low achievement and 5 is a high one. The targets and their scoring scales are described in the table 1 below.

	Target	*Definition*	*Scoring*	*Ref.*
Collaboration	***Accessibility***	Suitability of the process to be reached by users easily	1 = One person at a time can use the tool 5 = Multi-person, multi-site	[5, 7, 10–12, 26]
	Usability	Suitability of the process to be understood and mastered easily	1 = Using the process requires very expert knowledge 5 = The process is user friendly	
	Data structure	Possibility to structure exchanged data to ease on collaborative interactions	1 = Raw data only 5 = Possibility to organise the data into sets with metadata and specific access rights	
Reliability	***Reliability - Geometry***	Suitability of the process to ensure the interoperability of reliable geometric objects	1 = Recurring inconsistencies, loss and distortion 5 = High fidelity	[5, 7, 26, 30, 31]
	Reliability - Data	Suitability of the process to ensure the interoperability of reliable alphanumeric data	1 = Recurring inconsistencies, loss and distortion 5 = High fidelity	
	Data richness	Ability of the process to deal with a wide range of data types, specific to the AECO industry	1 = Poor range of data types 5 = Multiple types of data exchanged, metadata, etc.	
Agility	***Reactivity***	Ability of the process to enable close to real time up-to-date data	1 = No live link (single use data) 5 = Real time data updates	[5, 7, 11, 26]
	Openness	Openness of the API of the technologies used in the process for further analysis or for custom development	1 = Proprietary tool, no modifications 5 = Accessible API, can be developed further	
	Directionality	Ability of the process to enable bi-directional data flows	1 = Unidirectional data flow 5 = Bidirectional data flow	

Table 1: Interoperability process analysis matrix

d. Scoring

For each interoperability process, the scoring was done by the researchers in collaboration with the expert architectural team in charge of implementing the interoperability process studied. The presence of the authors during all the scoring studied ensured its relative homogeneity by ensuring the criteria and their evaluation were properly understood, although this exploratory process will have to be refined by testing the matrix in further case studies.



This scoring was done twice: - once to evaluate the expected capability of the process in general, and - once to evaluate the process as it happened during the project studied. These double evaluations per interoperability processes allow to highlight the team expectations as well as their possible circumstantial difficulties. Visual representations of the evaluation were then produced, with the expected performances in dotted line (figure 3) and with the effective performance in bold line (figure 3). The gaps between expected and effective performances as well as the blanks areas of the matrix would then be easily highlighted and allow for further analysis.

Figure 3 : Sample application of the discussed analytical matrix on project, for each target of the three category, a score is defined from low (0 at the centre of the graph) to high (5 at the periphery of the graph)

3.3 Corpus definition

Fast-paced computational design practices remain niche, but can be observed in studios like Grimshaw. Grimshaw is an architectural studio where digital practices are involved in the everyday design work, especially through BIM [6, 29] and Computational Design (CD) [8, 14]. The studio mainly works on large scale infrastructure projects with demanding performances in terms of structure and energy efficiencies. To address the different challenges faced by the designers, multiple toolsets[9] are used, either out of the box or developed in-house. It raises multiple difficulties and interoperability processes are implemented in order enable complex distributed design environments.

In order to test the relevance of the analysis matrix as well as to identify trends, strengths and weaknesses of interoperability processes in a distributed design environment, several projects were selected. The criteria of selection were for including an interoperability processes which is:

- actually used on project and appropriated by the design teams,
- significant of recurring needs across the practice or to have been implemented in several projects in the studio,
- dealing with a large scale or a high design complexity in terms of geometry or performances.

Five interoperability processes were selected across the pool of Grimshaw projects, each of them identified as representative of specifics design needs and technological environments. Each of these processes will be described shortly in the next sections. Because CD interoperability toolsets are evolving very quickly and the case studies analyzed here have been taken place at different times, the year of each case study indicate when the design and the interoperability process was developed and implemented.

3.3 Data collection methodology

The data collection methodology implemented for the case study rely on a double approach:

- Participant observations during the projects studied [32, 33];
- Participatory data collection at the end of the project [34] involving participatory discussions of the analysis matrix with the design team.

[9] Some of the authoring tools used at Grimshaw are: Rhinoceros, Grasshopper (McNeel), Revit, Dynamo (Autodesk), Aecosim, Generative Components (Bentley). To these authoring tools must be added visualization toolsets, environmental analysis, model review, collaboration platforms etc.

4. **Case studies**

4.1 Oman Botanic Garden façade: From Rhino to Revit through Grasshopper and Mantis Shrimp (2016)

The Oman Botanic Garden is a monumental infrastructure project in Middle-East, composed of two large greenhouses, called biomes. The façades of the biomes are panelized with planar panels addressing the overall non-regular geometry of the envelops (figure 4). Each of these façades is composed of several hundred panels, grouped into repeating types, allowing a cost effective and efficient construction solution. The facade explorations and the paneling were led by the architects using models and scripts developed with Rhinoceros-Grasshopper[10] software, known for efficient NURBS modeling and easy visual programming[11]. The overall coordination of the project was done through models developed using the BIM authoring tool Revit[12].

The interoperability process was led within the design team between different sub-teams, organized by building and construction packages (envelope, interiors design, landscape, etc). For the purpose of this study, the façade package will be the focus. While façade options were mainly explored in Grasshopper, they were communicated to the overall team through Revit modelling using a bespoke tool developed in-house called Mantis Shrimp. Mantis Shrimp[13] is a plugin developed by Konrad Sobon for Grasshopper and Dynamo for Revit. The vertices of the facades panels as well as the code for their type were extracted from the Rhinoceros-Grasshopper modeling (figure 4 on the left) and sent through Mantis Shrimp to Dynamo[14] for Revit, were Mantis Shrimp could read them back and feed the information to a Dynamo routine to model the panels as relevant native objects (figure 4 on the right).

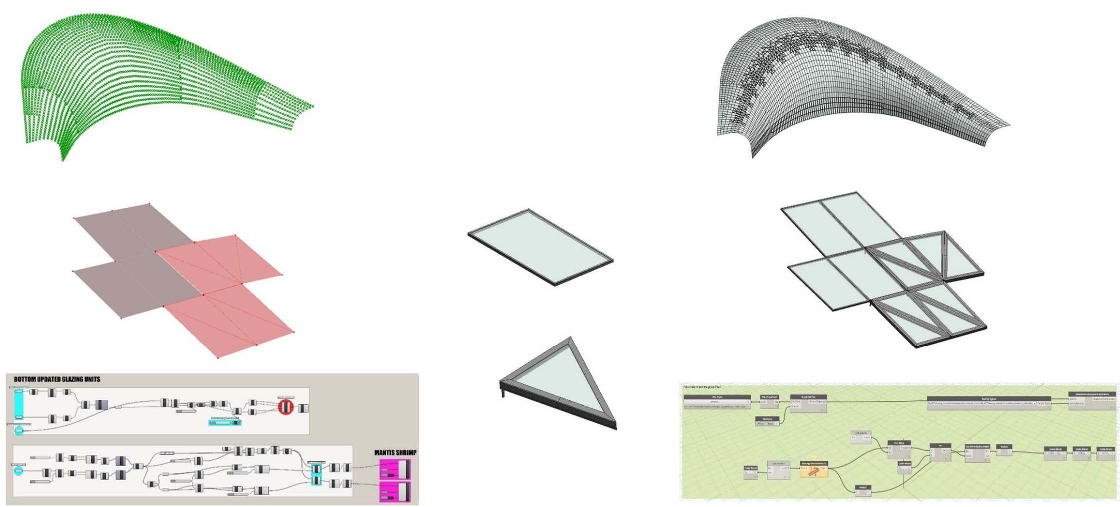


Figure 4: Oman Botanic Garden façade, from Rhinoceros-Grasshopper (on the left) to Dynamo-Revit (on the right) thanks to Mantis Shrimp developed by Konrad Sobon (credits: Grimshaw)

---

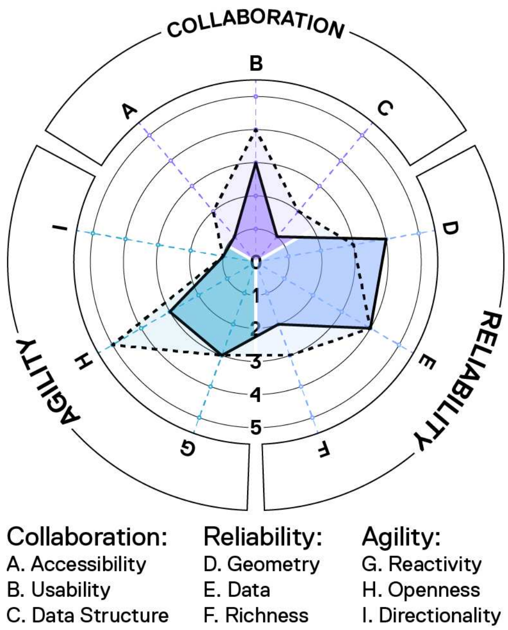
[10] Rhinoceros and Grasshopper are two different authoring tools which can be used in conjunction. When used as such, they will be named here as “Rhinoceros-Grasshopper”. They are both developed by McNeel: https://www.grasshopper3d.com/

[11] Visual programming lets users create programs by manipulating program elements graphically rather than by specifying them textually. It is widely used in Architectural Design practice as enables a relatively easy access to computer programming to non-experts. Grasshopper (McNeel) and Dynamo (Autodesk) are two widely used visual programming applications used by architects.

[12] Revit is a BIM authoring tool developed by Autodesk : https://www.autodesk.com/products/revit/overview

[13] https://archi-lab.net/downloads/mantis-shrimp/

[14] https://dynamobim.org/



While the process was expected to be relatively accessible and usable, its implementation in the team remained limited to CD experts and CD aware team members. The usability and reliability of the process proved adequate for the project.

The process was mainly aimed at transferring native panels elements from one authoring tool to the other. This context of limited expectations made the process relevant, even though it was limited in terms of capabilities for: bi-directionality, data richness and data structure (figure 5). In the end, data was exchanged more efficiently than expected, but not all the existing possibilities related to semantic data were used. Finally, the toolset used in this process ended being difficult to maintain once the developer left the company. Today this process is no longer used at Grimshaw.

Figure 5 :Analysis of the Oman Botanic Garden facade interoperability process (Expected process performance in dotted line, actual process in bold line. Evaluation method from 0 to 5 detailed in table 1)

The interoperability of complex geometries, especially for façade, is a recurring need for design team. The uses of curated data stream have revealed a very efficient strategy in Grimshaw experience. Indeed, it is not the whole geometry of the panels which is exchanged across the two authoring tools, but only the relevant data needed to recreate native objects (see the panels in figure 4Figure, from left to right).

4.2 Water park pavilions roofs: Rhinoceros and Revit consistency through Excel (2017)

During the design explorations of the infrastructure of a water park in China, three pavilions were proposed to host the different hospitality facilities needed for the public (ticket office, shop, etc) as well as different educational activities related to the theme of the park. Due to fabrication and cost constraints, the possibility of three identical pavilions with varying roof patterns was explored. The standardization of the pavilions through modules would then still address the specifics of the site and the program through varying transparencies of the roof, allowing different light and space qualities. As in the previous case study, Rhinoceros-Grasshopper was used to explore design options for elements with complex geometries (such as the pavilions roof), while Revit was used to design the internal layout of the building, to consolidate and coordinate the design. Due to the dimensions of the pavilions and the numerous panels of the roof, the exploration of patterns options and their evaluations in terms of solar performances revealed tedious, in Grasshopper as well as in Revit.

A bespoke interface was then developed by the design team. An excel spreadsheets would "map" the roof with one cell per panel, where the designers could easily annotate the different types of panels to place (such as: solid, glazed, serigraphic of different types and so on). These simple maps of the roofs were then read through a CSV format by Rhinoceros-Grasshopper or Dynamo-Revit, to create native panels (figure 6). From these native modeling further solar analysis, renders and other studies were undertaken, both in Rhinoceros-Grasshopper and Revit.

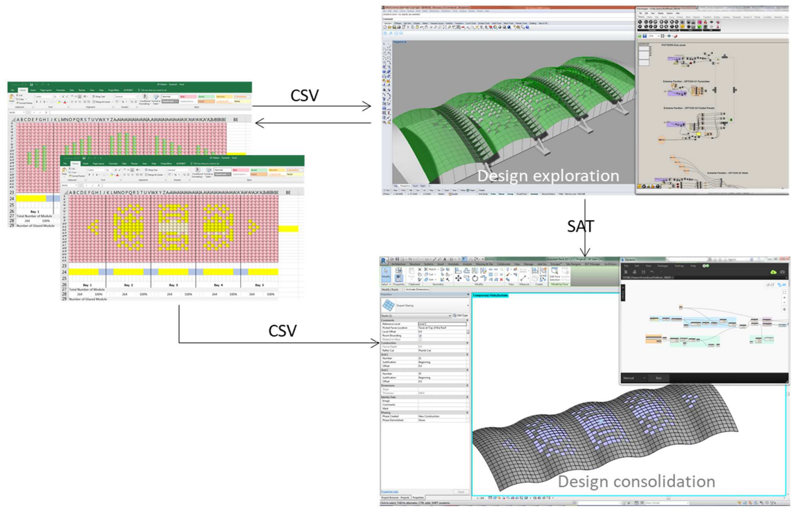
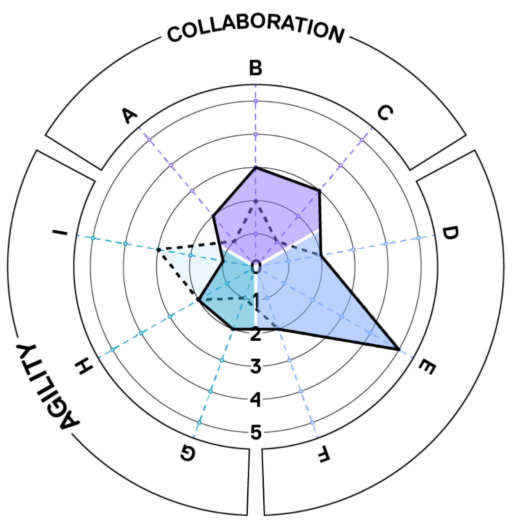


Figure 6: Roof paneling study led by "low tech" maps in excel. Left: roof pattern in excel, Top Right: Interoperability to Rhinoceros-Grasshopper, Bottom Right: interoperability to Dynamo-Revit (credits: Grimshaw)



Through this non-conventional use of CSV data exchange, more accessibility and usability was enabled than expected. The reduction of design information to the minimum for the exchange between the two authoring tools enabled a satisfactory level of reliability (figure 7).

Although the process was simple enough to efficiently focus on the only data needed and compensate the limitation of the toolset, the process still had a very reduced level of data-richness, openness and bi-directionality.

While this process shows a relatively unexpected use of the toolset, the versatility of the CSV formats makes it an interesting interoperability tool in the practice. However, the process studied here remains very bespoke and its transfer to other design contexts would mean several transformations which would have to be studied in further case studies.

Figure 7: Analysis of the Water park pavilions roofs interoperability process (Expected process performance in dotted line, actual process in bold line. Evaluation method from 0 to 5 detailed in table 1)

4.3 *Curzon Street Station structure: From Grasshopper to AECOsim through GeometryGym's IFC generation (2018)*

The Birmingham train station designed for the new HS2 line in UK is featuring a large open space enabled by a roof with large span capacity. The main design authoring environment was AECOsim developed by Bentley[15] (today called Openbuildings Designer). The geometry and the performances of the roof structure were explored using Rhinoceros-Grasshopper. To enable both fast design iterations as well as suitable modeling in AECOsim compliant with the BIM requirements, it was decided to use the IFC format. Each new option of the roof structure was automatically

[15] https://www.bentley.com/en/products/product-line/building-design-software

attributed relevant IFC specifications in Rhinoceros-Grasshopper using the plugin GeometryGym[16] (figure 8). The options were then exported in IFC[17] 2x3 and federated in the AECOsim model.

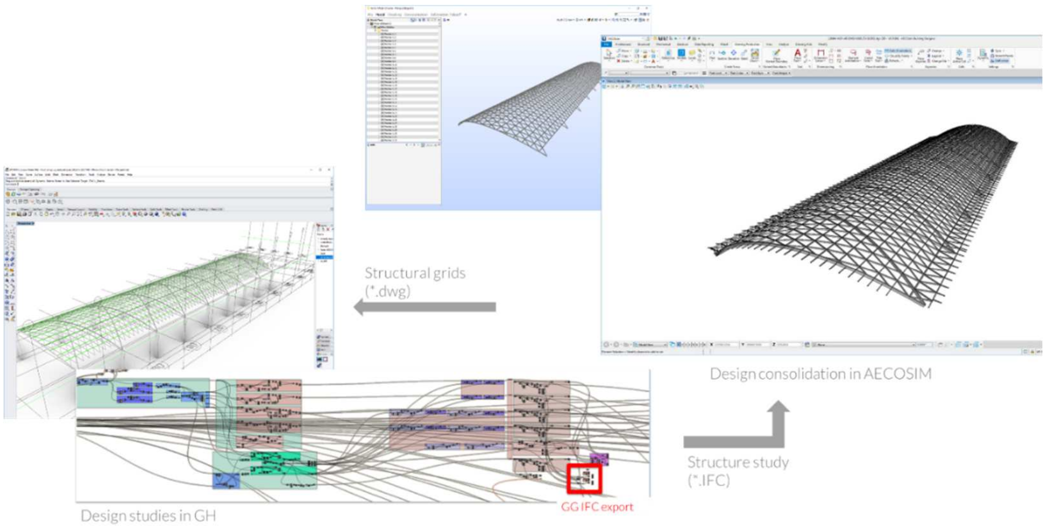
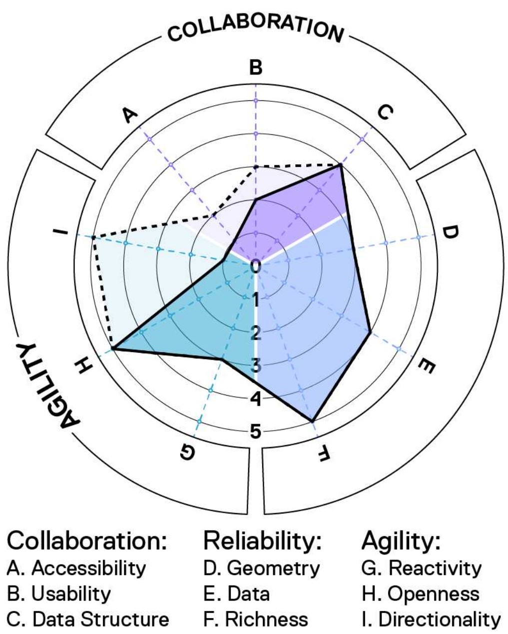


Figure 8: Curzon Street Station roof design, from Rhinoceros-Grasshopper (on the left) to IFC and AECOsim (on the right) thanks to GeometryGym developed by John Mirtschin (credits: Grimshaw)



This process had to address specific BIM modeling requirements, which was enabled in terms of data structure, data richness and overall reliability (figure 9).

Although the process was very relevant for the design context and its requirements, it proved to be relatively complex to implement, as it required both scripting skills and knowledge of the IFC schema. The process ended up being difficult to access for the design team, and instead was implemented mainly by the Computational Design Specialist of the project, in collaboration with the team BIM manager and the team lead. So whilst this process proved efficient in terms of reliability, its performances in terms of collaboration support and agility was reduced (figure 9).

Figure 9: Analysis of the Curzon Street Station roof's interoperability process (Expected process performance in dotted line, actual process in bold line. Evaluation method from 0 to 5 detailed in table 1)

4.4 Heathrow Masterplan study using Speckle (2019)

The Heathrow Airport Expansion was a challenging project involving hundreds of stakeholders from numerous expertise, from urbanisms to zoning and architectural design. During this project hundreds of references and design datasets were exchanged across multiple platforms and formats, especially during the masterplan design. The

[16] https://geometrygym.wordpress.com/

[17] IFC (Industry Foundation Classes) is a standardized open format for Building Information Modeling. https://technical.buildingsmart.org/standards/ifc

masterplan was designed across Civil3D[18], Revit and Rhinoceros-Grasshopper. Numerous datasets were exchanged in multiple formats, including CAD formats such as DWG, BIM formats like IFC and RVT, as well as GIS (Geographic Information System) formats, like SHP and GEOTIFF.

During this Masterplan design process, a combined system of parts (components) was developed based on the given brief. These components were related to both on-airport activities (such as airfield, terminals or satellite buildings) and off-airport activities (such as road & rail network, parking infrastructure or utilities). These components were distributed in a series of zones across the available land, through an iterative process of extensive optioneering and spatial and organization optimizations, which when put together form the document called the preferred masterplan. These airport components were exchanged and consolidated within the design team using the Speckle toolkit [19] [11, 35] as in Figure 10 and 11. Speckle is a cloud based collaboration tool which allow designer to send and receive data streams from most of their authoring tools, including Rhinoceros, Grasshopper, Revit and Dynamo. Speckle is used to improve data interoperability between these tools. During the Masterplan Design iterations, Speckle was used to exchange and consolidate the various components information between the different formats and authoring tools (figure 11).

Figure 10: Example of a component's Data Stream in the Speckle Viewer (credits: Grimshaw)

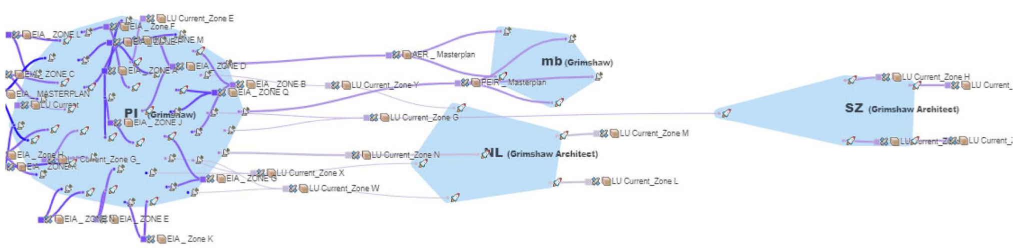
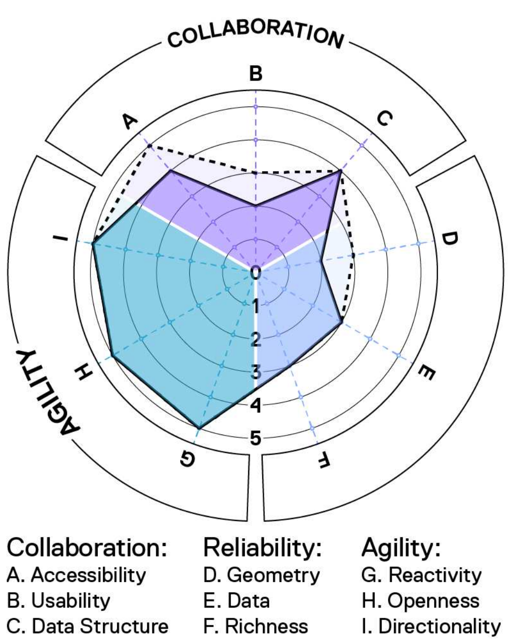


Figure 11: User-based visualization of the exchanges using Speckle-Viz: Each blue area is design team member, each connection is a data stream (sent and received between or several team members) (credits: Grimshaw)

[18] Civil3D is an Autodesk tool for infrastructure modeling.

[19] Initiated during Dimitrie Stefanecu's Innochain PhD research, Speckle is now developed by the Speckle System Company, founded by D. Stefanescu and M. Cominetti. https://speckle.systems/



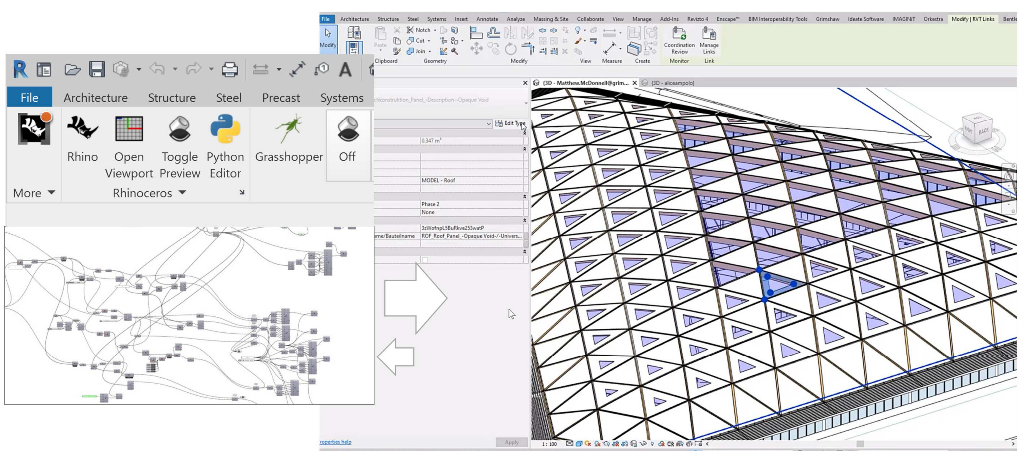
The design context of this process was especially demanding. It had to deal with large set of data from multiple sources to be consolidated and managed in a very bespoke manner. The Speckle toolset proved extremely versatile and agile (figure 12). However, it's versatility comes with a need for good expertise, and made the process accessible to only specifically trained team members.

Figure 12: Analysis of the Heathrow Masterplan interoperability process (Expected process performance in dotted line, actual process in bold line. Evaluation method from 0 to 5 detailed in table 1)

4.5 Swiss Airport Competition: Roof design from Rhinoceros-Grasshopper to Revit using RhinoInside (2020)

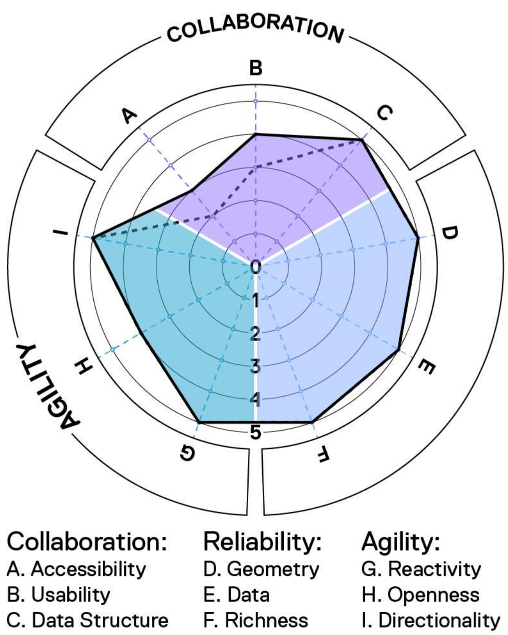
BIM requirements are becoming more and more important, to the point of being included in competition briefs. This was the case for an airport competition in Switzerland, for which specific IFC specifications were requested. The overall design team was working in Revit, while the roof design was explored in Rhinoceros-Grasshopper. While this project has similarities with the Oman Botanic Garden case study covered previously, several years separate them. For this latest case study, RhinoInside for Revit[20] recently developed by McNeel and in beta version at the time, was used. The tool has since been officially launched as part of Rhino 7.

RhinoInside for Revit allows designers to use Rhinoceros-Grasshopper within Revit, to script with Revit-aware components in Grasshopper that can query, modify, analyze, and create native Revit elements[21]. By using it, Grimshaw designers were able to design the airport roof components (panels and structure) in Grasshopper, and to create native Revit Elements from the same script (figure 13).



[20] https://www.rhino3d.com/inside/revit/beta/
[21] idem

Figure 13: Swiss Airport roof design, from Grasshopper (on the left) directly to Revit (on the right) without any middle-man thanks to RhinoInside (credits: Grimshaw)



This last process was very reliable (figure 14) for its expected use: a strong connection between Grasshopper and Revit. This process streamlined very efficiently the creation of BIM native elements from the Grasshopper Computational Design environment.

Although the accessibility was better there than for previous tested interoperability processes, the implementation of the process required specific training of some members of the design team, as well as targeted collaboration between CD and BIM experts.

Figure 14: Analysis of the Swiss airport roof interoperability process (Expected process performance in dotted line, actual process in bold line. Evaluation method from 0 to 5 detailed in table 1)

5. Discussions

5.1 Corpus limitations

These case studies are focused on processes involving mainly two authoring tools and their visual programming extension: on Rhino with Grasshopper, as well as Revit with Dynamo. These authoring tools are the most commonly used in practices that embrace Computational Design, but other processes involving different authoring software (such as ArchiCAD[22] or BlenderBIM[23] for example) or other interoperability solutions (such as Beams[24] for example) and strategies would have to be studied.

Also, as Computational Design practices remain niche, the access to “real world” case studies remain challenging. This research overcame this limitation and presented several real-life case studies, but its corpus remains focused on only one studio practice, Grimshaw. Further research will be needed to confirm and generalize the results.

5.2 Analysis matrix limitations and future evolutions

While the matrix proposed here allowed to meaningfully analyze most of each processes characteristics in regard of collaboration, reliability and agility; some further criteria could be improved. Especially, to describe more precisely the “success” of the process in practice, through the evaluation of its implementation in the office (if it’s rarely used or if it’s used on most projects) and its transferability from on project to another.

Also, criteria need to be improved in terms of refining the description of the accessibility and usability within the design team, as the current matrix don’t allow to describe in detail the diffusion of the process in the design team itself as well as the skills and roles of the people involved (Are they only experts? Are they BIM or CD aware? Etc.). Indeed, the current scoring does not allow the identification of the exact nature of accessibility. For instance, when considering simultaneous access, RhinoInside is somewhat limited as, by default, it only allows one user to input into

[22] https://graphisoft.com/solutions/products/archicad
[23] https://blenderbim.org/
[24] https://simplyrhino.co.uk/3d-modelling-software/beam

the process at a given time, whereas Speckle, through its distributed nature, allows an entire team to input into the interoperability process. The current scoring system could evolve to accommodate this through a more precise definition of each category and score, or through the addition of a “concurrency” category.

The possibility of prioritization of some of the matrix criteria could also be investigated. Currently, the criteria are treated as equal within the assessment. However, it may be that key users (designers, CD specialists, BIM specialists, etc.) may place more importance on one aspect than another. Through investigating this further, a process of applying a weighting to each score could be applied, either as a broad overall approach or as a per-user approach.

Finally, the scoring methodology itself could be reviewed, due to its current somewhat subjective nature. Rather than relying on the viewpoint of the author, albeit an experienced viewpoint, the scoring could be altered to a more defined quantitative scoring mechanism. Certain conditions could be met within each criterion to achieve, for instance, a score of 4 over a score of 3. This would involve further development of the analysis matrix as well as wider industry review and acceptance of criteria scoring.

5.3 Lessons learnt on weaknesses: “Why we all hate interoperability”

This research was initiated at Grimshaw in 2018 under the title “Why we all hate interoperability” [36]. Indeed, both data interoperability and collaboration processes across several authoring tools is challenging. There are no “one-size-fits-all solutions”, and interoperability technical capabilities are limited and restrained by proprietary technology. In the overall cases studied, visual programming was needed to implement the interoperability processes. Thus, these processes focused on recurring computational design challenges, such as optimizing of façade elements or ensuring BIM modeling compliance of non-standard elements, bespoke processes and scripting were needed for each of them, requiring specific and advanced skillsets.

Usability and accessibility of the toolsets is especially poorly supported in some of the processes analyzed (figure 15, left diagram), making interoperability a very specialist practice.

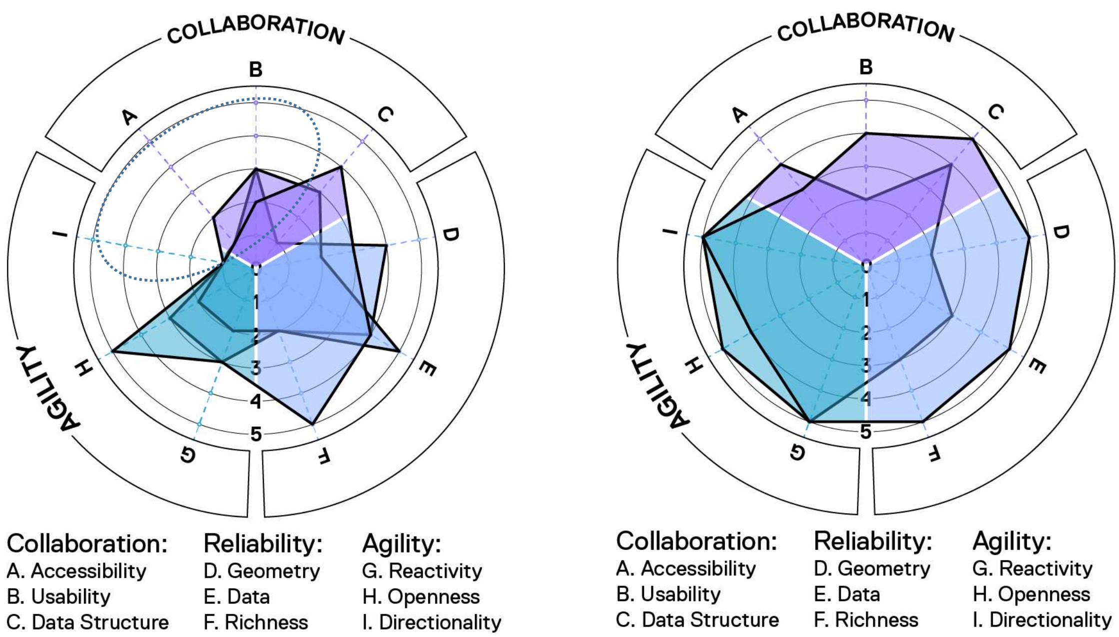


Figure 15: Synthesis from the case studies. On the left: case studies previous to 2019, on the right: case studies after 2019

However, the interoperability toolsets available to Computational Designers are evolving significantly, especially through:

- The ongoing improvement of existing plugins and software such as GeometryGym ones for example;
- The discontinuation of some solutions such as Flux.io in 2018 [12];
- The release of new solutions, such as Speckle or RhinoInside.

As a result, interoperability practices have been changing. It is especially the case at Grimshaw with the release of stable versions of RhinoInside in 2020. To this day, interoperability processes relying on RhinoInside are becoming dominant. This is the case as well in other practices relying heavily on Computational Design, such as at BIG-Bjarke Ingels Group[25], where RhinoInside is used on the majority of projects that necessitate advanced interoperability processes [37]. RhinoInside processes are perceived as highly usable and reliable (Figure, right diagram). The processes are streamlined through connecting authoring software (Rhinoceros-Grasshopper and Revit) directly and natively.

5.4 Lessons learnt on strengths: "Share the data, not the file"

While information exchange and interoperability in AEC is mainly a file based process, the ISO 19650 describes information exchange as "container-based" [29]. During design, fast-paced exchanges are needed to allow multiple iterations of design and fast communications between the team stakeholders as well as between multiple authoring tools if needed. In these contexts, it is shown that sharing meaningful and carefully curated data (such as minimal reference geometries like in Figure and Figure) through alternative means (such as data stream through web based services) is enabling powerful design processes. Thus, the expression "Share the data, not the file" started to gain exposure in the 2010's, especially with the development of flux.io [12]. And as Dimitrie Stefanescu states: "*smaller (and thus faster) data transactions increase the velocity of the overall communication process*" [11]. These data transactions enabling faster paced information exchanges across design stakeholders as well as across diverse authoring tools are indispensable for Computational Design.

However, the research identified how processes involving the toolset RhinoInside tend to differ from others. Indeed, its high usability and its strong takeoff in terms of frequency of usage in practice clearly distinguishes it.

6. Conclusions

While interoperability in AECO is widely present in the scientific literature, interoperability for fast paced design and especially Computational Design (CD) remains poorly documented. The scientific contributions of this research are: 1/ an analysis matrix proposed to interrogate CD interop processes and its specificity, and 2/its first tests through real life case studies of computational design in architecture practice. The 5 completed use-cases presented give strong scientific credibility to the interoperability matrix as a way to represent an interoperability problem. The presented matrix and corresponding graphical visualization is a potential useful means for evaluation and comparison of interoperability processes.

Through the processes observed, two different kind of CD interoperability emerged:

- Through integrated authoring environments: where different design toolsets are merged to constitute an augmented environment (as in the case study relying on RhinoInside).
- Through distributed authoring environments: where design toolsets are connected through data transactions (as in the case study relying on Speckle).

While agility and reliability of CD interoperability processes are increasing since 2019 due to the development of new toolsets and methodologies, the collaboration aspect of these processes seems stagnating (as in figure 15).

[25] https://big.dk/#about

The matrix proposed in this paper would have to be further interrogating in future research. Especially, the use-case corpus would have to be expanded to include more diverse projects, firms and toolset, including open-source solutions. The matrix could also be tested against other AECO processes, such as BIM ones or, wider augmented collaboration ones. Furthermore, the project team dynamics, roles and skillsets which are involved in these design processes involving complex authoring environments would have to be interrogated too.

7. References

1. Camus C (2016) Mais que fait vraiment l'architecte ? Enquête sur les pratiques et modes d'existence de l'architecture. L'Harmattan

2. Chadoin O (2013) Etre architecte, les vertus de l'indétermination. Presses Universitaire Limoges

3. Latour B, Woolgar S (1979) Laboratory Life: The Social Construction of Scientific Facts. La Decouverte

4. Sattler L, Lamouri S, Pellerin R (2020) Rétro-BIM ou la question de l'intégration : une revue anachronique. In: SCAN 20, Culture numérique et conception architecturale …. Bruxelles

5. Sattler L, Lamouri S, Pellerin R, Maigne T (2019) Interoperability aims in building information modeling exchanges: A literature review. IFAC-PapersOnLine 52:271–276 .

6. Eastman C, Teicholz P, Sacks R, Liston K (2011) BIM Handbook : A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, 2nd Editio. John Wiley & Sons, Hoboken

7. Ozturk GB (2020) Interoperability in building information modeling for AECO/FM industry. Automation in Construction

8. Menges A, Ahlquist S (2011) Computational Design Thinking, AD readers. Wiley

9. de Boissieu A (2021) Introduction to Computational Design: subsets, challenges in practice and emerging roles. In: Bolpagni M, Gavina R, Ribeiro D (eds) Industry 4.0 for the built environment: Methodologies, technologies and skills. Springer Nature

10. Poinet P (2020) Enhancing Collaborative Practices in Architecture , Engineering and Construction through Multi-Scalar Modelling Methodologies. Aarhus School of Architecture

11. Stefanescu D (2020) Alternative Means of Digital Design Communication. In: Sheil B, Thomsen MR, Tamke M, Hanna S (eds) Design Transactions. UCL Press

12. Ferries B, de Boissieu A (2018) Pour une interopérabilité dynamique. In: EduBIM. Eyrolles

13. Ringley B (2017) The long game, Maintaining complex interoperable workflows through multiple phases and teams. In: BILT

14. Carpo M (2017) The second digital turn : design beyond intelligence. MIT Press, Cambridge

15. Carpo M (2012) The Digital Turn in Architecture 1992 - 2012, AS Reader. Wiley

16. Bernstein PG (2018) Architecture, design, data : practice competency in the era of computation. Birkhauser Architecture

17. Terzidis K (2006) Algorithmic Architecture. Architectural Press

18. Negroponte N (1969) Towards a humanism through Machines. In: Menges A, Ahlquist S (eds) Computational design thinking. AD Readers

19. Aish R, Woodbury R (2005) Multi-level interaction in parametric design. Lect Notes Comput Sci 3638:151–162 .

20. Davis D (2013) Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. RMIT University

21. Janssen P, Stouffs R (2015) Types of parametric modelling. CAADRIA 2015 - 20th Int Conf Comput Archit Des Res Asia Emerg Exp Past, Present Futur Digit Archit 157–166

22. Woodbury R (2010) Elements of Parametric Design. Routledge

23. Thomsen MR, Tamke M (2020) Design Transactions. UCL Press, London

24. Anderson C, Bailey C, Heumann A, Davis D (2018) Augmented space planning: Using procedural generation to automate desk layouts. Int J Archit Comput 16:164–177

25. Calixte X, Ben Rajeb S, Gronier G, Leclercq P (2014) Architecturale Collaborative. In: Psychologie Ergonomique EPIQUE. Arpege Science Publishing

26. Sattler L, Lamouri S, Larabi M, Pellerin R, Maigne T (2020) L'interopérabilité BIM multimétiers: la collaboration par la requête. In: Marques S, Teulier R (eds) Le BIM et l'évolution des pratiques. Editions Eyrolles

27. Holzer D (2016) Design exploration supported by digital tool ecologies. Autom Constr 72:3–8 . https://doi.org/10.1016/j.autcon.2016.07.003

28. Fernando R, Drogemuller R, Burden A (2012) Parametric and generative methods with building information modelling: Connecting BIM with explorative design modelling. Beyond Codes Pixels - Proc 17th Int Conf Comput Archit Des Res Asia, CAADRIA 2012 537–546

29. ISO (2018) EN ISO 19650-1:2018 Information management using building information modelling Part 1: Concepts and principles, bsi

30. BuildingSmart (2018) What is Industry Foundation Classes (IFC). In: buildingsmart.org. https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/. Accessed 15 Feb 2021

31. Lai H, Deng X (2018) Interoperability analysis of ifc-based data exchange between heterogeneous BIM software. J Civ Eng Manag.

32. Van Campenhoudt L, Quivy R (2011) Manuel de recherche en sciences sociales. Dunod, Paris

33. Peretz H (2004) Les Méthodes en sociologie : L'Observation. La Decouverte, Paris

34. Ben Rajeb S, Leclercq P (2016) Analysis of Collaborative Design through Action Research : Methodology and Tools. Int J Adv Intell Syst 9:199–212

35. Poinet P, Stefanescu D, Tsakiridis G, De Boissieu A, Papadonikolaki E (2020) Supporting collaborative design and project management for AEC using Speckle's interactive data flow diagram. In: Maciel A (ed) DCIO - Design Computation Input Output. Design Computation, London

36. de Boissieu A, Watts A (2018) Why We all Hate Interoperability: Tools, Late Nights, and Case Studies. In: BIM Show Live

37. Thomas O (2021) BIM and Computational Design at BIG New York Studio, Interview with Aurelie de Boissieu