AGILE X UNISA PAVILION: AGILE PRINCIPLES AND THE PARAMETRIC PARADOX

TIM MCGINLEY¹, BRETT ABROE², DAVID KROLL³, MATT MURPHY⁴, TESSA SARE⁵ and NING GU⁶
¹,²,³,⁶ University of South Australia, Australia
¹,²,³,⁶ {tim.mcginley|brett.abroe|david.kroll|ning.gu}@unisa.edu.au
⁴ MassBespoke Ltd, Australia
⁵ mattm@baumanlyons.co.uk
⁶ Office for Design and Architecture South Australia, Australia
⁵ tessa.sare@sa.gov.au

Abstract. The world is experiencing an ever increasing pace of change and yet our design processes typically follow a waterfall model that make can make change and adaptation difficult. Digital design approaches provide an opportunity to develop agile solutions that are more open to change in the design process. This paper proposes the development of immaterial architectures wherein the material expression is left to later in the design process when there is greater certainty. We describe a series of 3 workshops that employ aspects of agile software development methodologies into architecture. The workshops proposed 3 immaterial pavilions for Delft, Adelaide and Tianjin. This first cycle of three workshops resulted in the design, fabrication and installation of the Agile X UniSA Pavilion in Adelaide. This paper discusses the applicability of agile development methodologies to this process and details a series of adaptations to provide a set of appropriate agile development principles for architecture.

Keywords. Agile; Architecture pavilion; Immaterial architecture; Parametric paradox; Collaboration.

1. Introduction

Modernism dictates that design should be tailored to its materials. This belief has largely continued unchecked into contemporary digital design processes. This paper proposes an alternative approach that supports flexibility in the choice of material and fabrication systems. Parametric design systems provide flexibility in the design of buildings (Aish & Woodbury 2005). This should be extended to include flexibility in material and fabrication. However, designers of parametric systems typically lock in a specific material type (material system) into the design
system, which ensures reliance on a specific supply chain. This is a challenge because changes in the availability of the material can impact the cost and time that it takes to deliver an architectural project.

![Diagram showing common project planning paradoxes](image)

*Figure 1. Common project planning paradoxes after Samset & Volden (2016) diagram by authors.*

There are many similar paradoxes to be found in project planning (Samset & Volden 2016) (figure 1), whilst it is not possible to address all of them, this paper seeks to identify approaches to help designers to manage them. This paper argues that digital meta design systems could support a more ‘agile’ approach to material specification and procurement in architecture that could respond to material availability and, more relevantly, resource scarcity (McGinley 2015b). There are many paradoxes of front end decision making in project management (Samset & Volden 2016). These include that tactical success of a project does not automatically mean a strategic success. This can be because the tactical approach may be misaligned with the project strategy, or the success assessment focuses on tactical rather than strategic success. This can be seen in news media items that focus on failure as running over on time or budget, more than if it meets the needs for which it was intended. A project can therefore be a tactical success, i.e. on time and budget but a strategic disaster.

Samset & Volden reveal other paradoxes in project management that are relevant to architecture. These include a concern about an overflow of information in the early stages of the project that can make effective cost estimation, tactical and strategic decision difficult due to ‘analysis paralysis’. This is compounded by the potentially myopic perspectives of the planners, predict and provide models that are designed only to satisfy the potentially poorly predicted future demand. Further investigation of the opportunity space may enable stakeholders to consider how to reduce demand rather than simply provision for it for instance. Some of these challenges may be able to be explored by expanding the opportunity space, however to achieve this the path dependence of the stakeholder’s design decisions also needs to be challenged. Ultimately even with a robust conceptual model generated in the opportunity space, the paradox of strategic alliance says that ‘the internal logic of causalities and the probabilities of realization’ will result in erroneous solutions (Samset & Volden 2016). It therefore seems sensible that identifying design technologies and methods that are flexible and support exploration of the opportunity space as far into the project as possible would be beneficial to strategic success in architecture projects.
2. Parametric Paradox

Parametric models define and encode relationships in digital design components that enable designers to control the parametric design through the manipulation of parameters. The parameters that the designer of the parametric design system selects determines the opportunity space that the parametric design system explores. Parametric models are require additional work to be done at the start of a project and are reliant on the appropriateness of the parameters selection at the front end of the project (Yu et al. 2015). In this way are subject to the ‘paradox of the opportunity space’ described by Samset & Volden (2016) (figure 1). The argument for parametric systems is that they can augment a designer’s ability to explore an opportunity spaces that can be predicted in advance. Furthermore, the development of robust parametric design systems could be used to adapt the design proposals in response to a required change in strategy from late emerging information. However, parametric design systems are frequently developed in an ad hoc manner that means we end up with ‘spaghetti parametrics’ that can be difficult to reconfigure to an adjusted design strategy beyond the scope of the parameters defined in the system (Davis et al. 2011). To address this, we need to apply robust and flexible development methodologies that produce high quality outputs and are appropriate to the design process.

3. Agile Development Methods

In response to an increase in the control of project managers and creativity crippling documentation requirements in software development at the turn of the 21st century, programmers developed a series of methodologies which were united under the manifesto of agile software development (Beck et al. 2001). The story of the development of Agile is well documented as are the techniques to employ it (Martin 2003). Agile is often compared to the more traditional ‘waterfall’ model (figure 2) wherein each of the main tasks are completed sequentially and are all planned up front. In Agile development the design progresses in sprints and the tasks are reviewed on a daily basis. These ideas were incorporated in the four values of the Agile Software Development Manifesto (Beck et al. 2001).

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

Agile development covers a range of approaches including, XP (extreme programming), Scrum, (Erickson et al. 2005) as well as others. Droste (2010) describes a process of applying agile development methodologies from computer science in architecture. The limitation of Droste is that it focuses on XP rather than more broadly on the principles of Agile development. This is important because there is much discussion in computer science about the most appropriate method of agile development. Therefore this paper focuses on the principles of agile development rather than a particular methodology.

The principles are summarised here as: Satisfy clients by early and continuous delivery of valuable designs (P1); Welcome changing requirements, even in late development (P2); Deliver working designs frequently (P3); Facilitate close, daily cooperation between stakeholders and designers (P4); Build projects around motivated individuals, and trust them (P5); Encourage face-to-face conversation as the best form of communication (co-location) (P6); Measure progress on working design systems (P7); Maintain a constant sustainable pace (avoid burn out) (P8); Focus on technical excellence and good design (P9); Maximise the amount of work not done (P10); Support teams to self-organise (P11); Reflect continuously on how to become more effective and adjust behaviour accordingly (P12).

4. A Hybrid Approach

There have been approaches previously to map the Agile development processes into design and design research (Conboy et al. 2015). Agile project management has also been proposed as a method to deal with resource scarcity and fluctuating availability in construction in the form of an ‘Agile fabrication methodology’ (McGinley 2015b). These approaches could be applied to frameworks to support more open architectural design processes by applying methods from computer science into digital architectural design (Schneider et al. 2010; McGinley 2015a). To answer this question we embarked on a series of design workshops to explore the application of agile development principles in architecture. The mapping was used to inform three workshops which proposed an ‘immaterial’ approach to digital design that decoupled the design from its potential material and fabrication systems. The following sections of the paper provide descriptions of the workshops followed by a reflection on how they performed in relation to the agile principles defined previously.

5. Agile X Workshops

The first workshop attempted to create immaterial pavilions that could adapt to the diverse contexts of Delft, Adelaide and Tianjin. The second workshop explored a parallel design process wherein students worked on the conceptual; design, detail and materiality systems in parallel. This resulted in a pavilion for Adelaide that could be conceptualized, materialized and fabricated in many different combinations. The final workshop incorporated representatives from practice with architecture and engineering students from UniSA. It resulted in the fabrication and
installation of the ‘Agile X UniSA Pavilion’ (figure 5) at ODASA (the Office for Design and Architecture South Australia). The three workshops are discussed in the following sections from the perspective of the Agile principles to identify how well the agile process fitted and was applied to the design context.

5.1. AGILE X 1 (AGILE FAB DELFT)

Figure 3. Overview of Agile X 1.

The first workshop ran over 5 days, hosted by Hyperbody Research Group at TU Delft. Students from University of South Australia (UniSA), Hyperbody, Tianjin University and Beijing Jiaotong University worked in 6 groups of 6. Each group had participants from each institution. 2 groups focused on the Beijing pavilion, 2 on the Adelaide pavilion and 2 on the Delft immaterial pavilion. On the last day of the workshop the intention was to materialize a tectonic detail within the pavilion. Due to time constraints this was not possible, but all groups did manage to adapt their immaterial design so it could be applied to the other 2 sites. This meant that we ended up with a set of 6 designs that could each be applied in the three contexts. In terms of agile principles, this workshop performed well in the facilitation of close daily collaboration (P4) as shown in figure 3. The groups were also self-organizing and maintained a constant sustainable pace (P8). The challenge with reviewing the agility of the workshop is that the agility was not extensively tested in this workshop. For instance, few changing requirements (P2) were placed on the participants beyond starting with a focus on a single site and then in the last days being asked to adapt their immaterial architecture so that it could adjust to all three contexts. In the end AX1 provided the ‘front end’ work that we needed to the support the other 2 workshops, but in itself was not very agile and produced a limited number of working design systems. That said there was constant and stimulating informal reflection on the process (P12) which helped to inform the following workshops.

5.2. AGILE X 2 (AX2)

The second Agile X workshop ran over 2 weeks in Adelaide, South Australia. 12 participants divided into 6 teams of which: 2 teams focused on the material systems of the pavilion; 2 focused on the design system; and 2 focused on the fabrication system. The teams had between 1 and 3 people in them. For those with 3,
they could nominate a Scrum master. However this designation proved tricky as the Scrum masters were typically participants who had less technical skills. This meant that it was difficult for them to understand the tasks that the rest of the team internalized. It was therefore not possible for the Scrum masters to enforce the Scrum. Therefore, in reality it became more of an eXtreme programming pair programming approach supported by a third participant who was involved in documentation of the solution and reflection (P12).

The teams self-organised (P11) into three teams to support the material, design, tessellation and fabrication (numbering) systems. AX2 showed that whilst the extreme programming (a subset of agile development) concept ‘pair programming’ worked well, that ‘Scrum managers’ were less well received by the architecture students. The participants produced working systems (P7) although they struggled to link these systems together. This should be a focus in future workshops. The team performed well on adaptation to changing requirements and where able to significantly adapt the final design ‘on the fly’ in the final presentation in response to the developing requirements of the client (P2). The teams self-organized (P11) around motivated individuals (P5) were co-located (P6) collaborated closely (P4), maintained a constant, sustainable and fast pace (P8) to deliver design frequently (P3) that were valuable to the client (P1). Interestingly in this workshop some of the technical excellence was provided to the participants by the tutors, however we feel this is important as it means that the tutors are part of the team (P9).

5.3. AGILE X 3 (AX3)

The third Agile X workshop was hosted at UniSA in Adelaide, and was structured around the requirements and input of the clients (P1). This communication was facilitated with frequent prototypes and reports to the client (P3, P4). The 32 participants were divided into 10 teams that covered a number of design, fabrication and installation tasks including: checking the design against the Building Code of Australia (BCA); production of the single coordination drawing; Structural analysis; Fire safety; drawings for landlord approval; arranging the cutting sheets for CNC milling; Painting; 1:6 Prototype production; and finally 2 groups focused on producing instructions for the construction sequence. In the first of the 2 weeks of AX3 we fabricated and assembled the structure based on the design from AX2.

Two significant changes that were made included a change to height of the pavilion and a change to the materiality of half of the pavilion during fabrication due to a late merging issue on site (P2). In the second week participants designed the panels that would infill the structure and assembled and installed the final pavilion. This process required a lot of coordination and collaboration which was difficult due to the distributed locations of the workshop (stretched between the digital and physical workshops). That said the experience, expert advice and support of the physical workshop staff were invaluable to the production of the prototypes and the fabrication of the pavilion (P9). There were a number of key features that defined the workshop including: an insistence that we only have one drawing, and that that drawing is correct.
In this way the participants were able to maximize the amount of work not done (P10), although the large number of participants meant that there was inevitably some redundancy in tasks, which should be addressed in future workshops. All participants reflected on their experience and performance in daily blogs and journals, some of these reflections include comment on improving efficiencies in the process (P12).

Table 1. Comparison of each workshops performance against the 12 principles of Agile Development.

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
<th>P12</th>
</tr>
</thead>
<tbody>
<tr>
<td>AX1</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>AX2</td>
<td>✓</td>
<td>✓✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓✓</td>
<td>-</td>
</tr>
<tr>
<td>AX3</td>
<td>✓</td>
<td>✓✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

6. Agile Design Principles for Architecture

P1 seems to apply more in the second and third workshops, this was mainly because there was no explicit client in the first workshop. This also meant that P4 (facilitate, client designer collaboration) was not possible in the first workshop. P2 and P3 similarly only applied in the second and third workshops. These principles were explored in these workshops because the requirements changed as the brief developed with the client. Interestingly the teams did ‘produce working designs frequently’ even though the requirements changed frequently in these last two workshops. P5 was not affected by the lack of a client in the first workshop. The first and second workshops both had large numbers and it appears that P6
was easier to achieve in smaller teams. A future recommendation would be for workshops to be housed in either the same space or spaces that are very close together. P7 needs to be addressed in future workshops. P8 is important not only to the success of the project but the health of the team and is a relevant principle for architectural design.

P9 reflects the professional basis of architecture. Whilst this is relatively simple to impose in a waterfall model it requires further consideration in an agile model. P10 was very helpful in the third workshop once we had explored the opportunity space in the first and second workshops. P11 happened organically in the second workshop. In the third workshop, the teams self-selected their partners but the tasks were prescribed, it might have been better for them to discover the tasks that needed to be done themselves. P12 (reflection) is an essential part of any design process but even more so in an educational context. This was supported in the third workshop by participants writing a daily reflective journal / blog that formed part of the assessment.

Reflecting on the 3 workshops above it seems that the agile principles are relevant to architecture. The second workshop most clearly represented the principle of Agile Development. The first workshop represented the least number of agile principles. The workshops showed that parametrics alone intensify some of the paradoxes of project management. This problem is compounded by the way that parametric systems are frequently developed in ad hoc approaches that do not follow development methodologies from computer science. Therefore, this paper proposes an approach wherein parametric systems are developed using agile development principles. The 12 principles can be quite specific to software and we therefore propose the following 7 adaptions: Agile design principles for architecture.

1. **Maintain a constant sustainable pace (avoid burn out).** It is necessary to take frequent breaks and adjust project constraints and load if required in order to maintain the health of the team.
2. **Encourage face-to-face conversation and colocation wherever possible and facilitate close, daily cooperation between stakeholders and designers.** Barriers to collaboration and communication should be reduced.
3. **Build projects around motivated individuals, trust and support them to self organize.** Encourage and support flat hierarchies where in tutors and participants are on the same team - learning and solving problems together. The flat hierarchy means that any participant can suggest changes to both the design work and the facilitation of the workshop at any point.
4. **Satisfy clients by early and continuous and frequent delivery of valuable designs.** An aim for agile design systems for architecture is to have parallel systems and that we can meaningfully involve the client early in the design process. Agile focusses on using technology to support collaboration with clients. In this way it is the collaboration and the constant testing and exploration of the design tools that enable the design to progress.
5. **Maximise the amount of work not done and measure progress on working interoperable design systems.** This does not mean that everyone needs to be an expert, but some do. This should not be confused with reducing the exploration
of the opportunity space, but focus on minimizing redundancy in the exploration.

6. **Welcome changing requirements, even in late development.** Close collaboration with the clients is not sustainable without high value design solutions to offer. To support this, it is important that the design systems are not only interoperable but can be reconfigured to support alternative design systems.

7. **Focus on technical excellence and good design whilst reflecting continuously on how to become more effective and adjust behaviour accordingly.** This provides an opportunity to capture and reuse valuable design tools and processes to extend exploration of the opportunity space and focus on how to do things better next time.

![Figure 5. The final pavilion following the installation in the 3rd workshop (photo credit: Sam Noonan).](image)

**7. Conclusion**

This paper argues that the underlying cause of many of the intertwined paradoxes presented in (figure 1) is the compression of the opportunity space into the front end of the project. Therefore, from the experience of a series of 3 workshops it appears that the expansion of opportunity space provided by the hybrid agile parametric approach presented here would relieve the compounding impact of the central paradoxes of project management. Expanding Droste (2010), this paper adapts the original principles of agile development to describe 7 agile principles for an architectural design. The potential innovation offered by the hybrid agile parametric approach is to expand the opportunity space offered in traditional approaches (figure 1), which has been observed based on the current preliminary
findings. Further testing in future workshops will be done in order to further verify the effectiveness and explore the potential, in order to critically theorize the approach for wider applications.

Acknowledgements

We wish to acknowledge: the input of the Agile X workshop participants in shaping the research; the support of the Government Office for Design and Architecture South Australia and Woods Bagot, Adelaide as well as Adam Jack for his assistance in the second workshop and Linus Tan in the third.

References