6. The Use of Gameplay Design Patterns “It’s like a project in itself, really”

The Use of Gameplay Design Patterns “It’s like a project in itself, really”

Word count currently 11,000

Research Questions- March 2024

1. What pedagogical tools and processes are available to support novices to overcome

barriers to participation in game coding processes?
2. How can game design patterns support the development of coding practices with novices?
3. How can learners build agency in an evolving community of game makers?

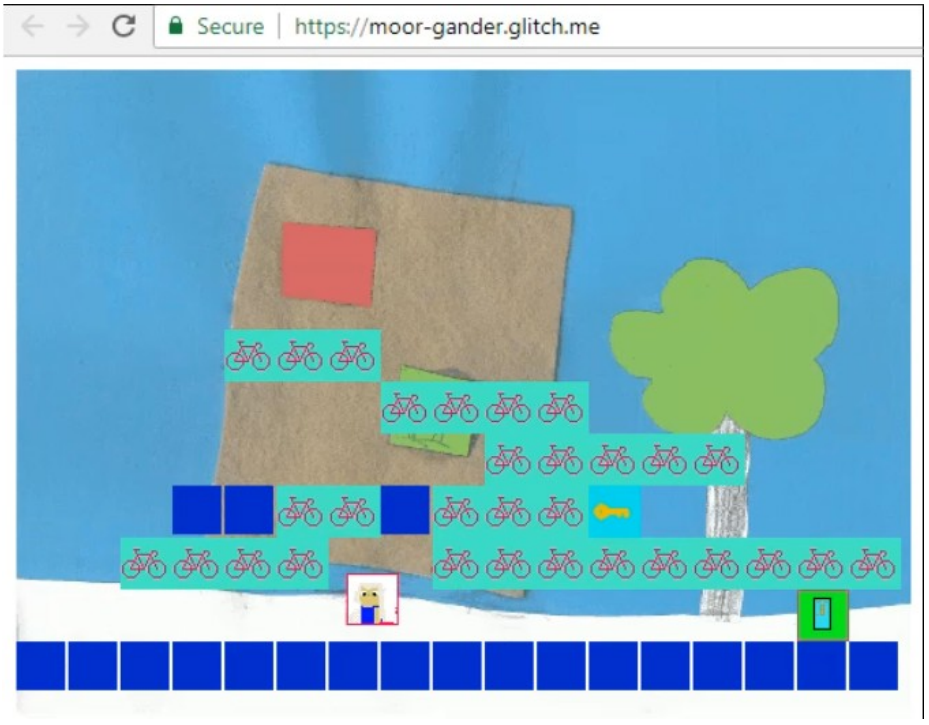
Introduction

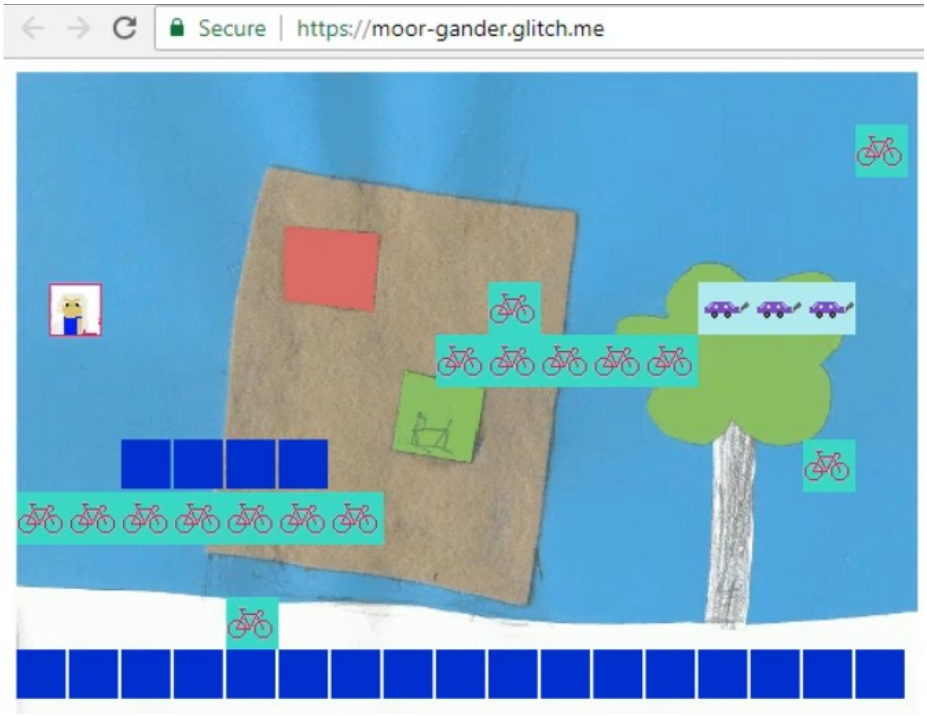
In the previous chapter, I outlined the evolution of tool use, emphasising emerging tensions in activity. This chapter shifts focus to the perspectives of participants, using recorded data to interpret the varied applications of gameplay design patterns (GDPs). In doing so, it addresses the research question: How can game design patterns support the development of coding practices by novices? Chapter 2 summarised software design patterns as a grouping of a description of target project behaviour, with a suggested community solution of code structure often accompanied with worked examples. While existing research explores the use of game design patterns in educational settings, for instance to aid the transfer of science simulation concepts to other contexts (Basawapatna, Koh and Repenning, 2010; Repenning *et al.*, 2015) or as a foundation for co-design work (Eriksson *et al.*, 2019), there is limited work addressing their role in overcoming challenges faced by emerging communities of game creators using text coding. The last chapter investigated iterative activity involving alterations to games at the level of gameplay design patterns as part of the mutual evolution of learning design. This evolution encompassed various forms, including supporting resources such as code examples, instructional tutorials and quick-start activities proposing micro-level code modifications. This chapter builds on that analysis, focusing on the use of GDPs in diverse mediational strategies, not only within facilitator-provided resources and scaffolds but also as part of emerging community practices and interpersonal repertoires. During data analysis, GDPs appeared in such diverse and significant forms that they became candidates for a germ cell concept.

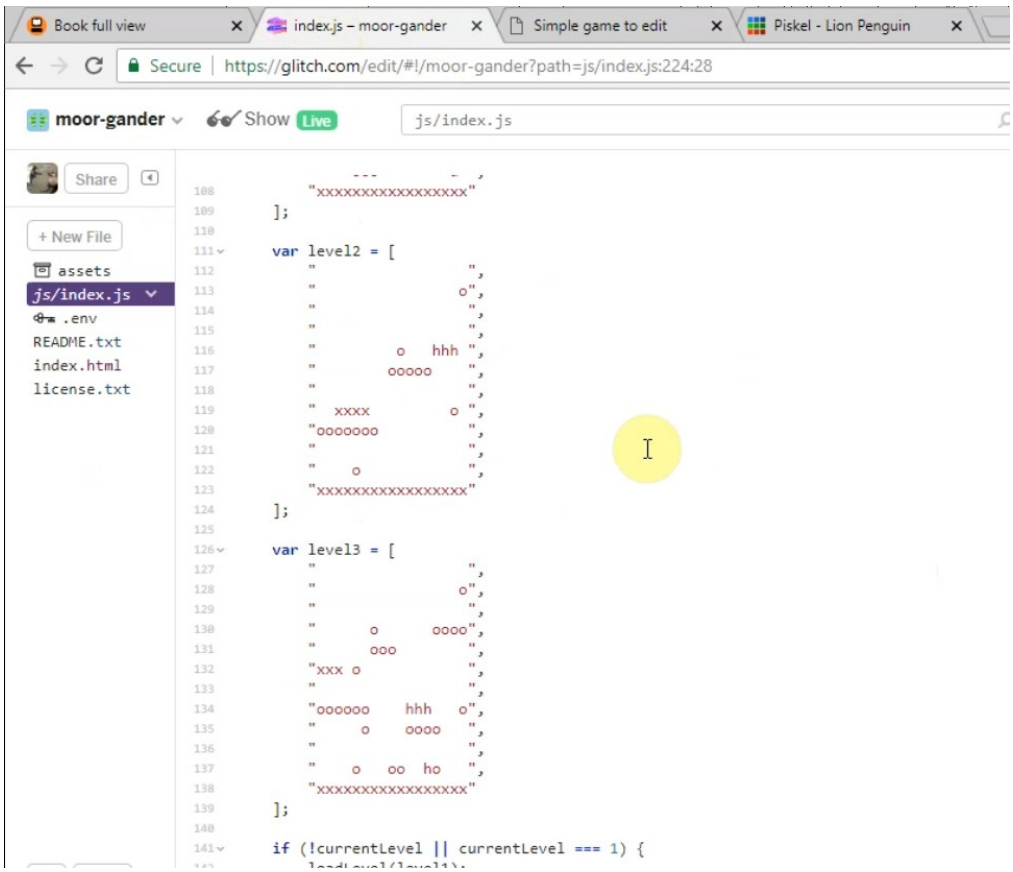
Regarding the structure of this chapter, it begins with a vignette of parent-child interaction during the game-making process to contextualise the findings that follow. The chapter then presents a systematic analysis of participants' varied uses of game design patterns, drawing on detailed observations of recorded video data. For this analysis, I employ Rogoff's three planes of analysis (Rogoff, 1995), as introduced in Chapter four. Recognising that these observations do not neatly align with the three planes, I discuss the use of GDPs across cultural activities, interpersonal activity and personal knowledge in three distinct sections. These sections focus on how gameplay design patterns are utilised by parents, children and facilitators, each through the lens of the respective analytical plane. An analogy can be drawn between these three planes of analysis and the examination of activity systems at varying scopes. Specifically, the cultural plane aligns with the community scope of activity described in Figure 5.x in the previous chapter, while the interpersonal and personal planes correspond to smaller scopes depicted in Figure 5.x. A discussion section explores the implications of these findings in relation to existing research and theoretical concepts. This discussion considers the examples of GDP use in practice to analyse GDPs as a mid-level construct, demonstrating their utility for both educational designers and researchers.

Extract and commentary on Vignette 2

In this section, I provide an overview of the specifics of the learning design and participant interactions. Beyond providing the reader with an illustrative example of partner work in game coding, this commentary begins to surfacing uses of GDPs as a meditational strategy, which are then explored using wider data in the rest of the chapter. To achieve this, I use a vignette to capture a snapshot of the interactions within one family, as it includes several aspects of activity that incorporate the use of GDP concepts. A truncated table presents participant dialogue, screenshots and a description of their interactions, including gestures, followed by a summary interpretation of these interactions. A more extended extract of dialogue, activity description and commentary is available in Vignette 2 in the appendices.

4	Suzanna: Thank you. Did you see Olivia?	(Off Screen) Mick cedes computer to Suzanna.
5	Olivia: Did it do it?	
6	Suzanna: Yes	
7	Suzanna: I'll reload it. Here you go	Suzanna presses refresh key to reload the game
8	Olivia: Laughs	Olivia moves laptop to point towards her and takes over using the keyboard
9		

10	Olivia: (Makes excited cat noise)	Olivia is using arrow keys to control game and checks desired behaviour of player progressing to level 2 is working. T progresses to level 2 where there is not a door or key.
11		
12	Olivia: We need a door in this one. I want to make more levels now. So you can put lots of doors and lots of keys (in a rhythmic voice) Key Door Key Door Key Door (in a sing song voice) Key Door Level. Key Door Level, Key Door Level	Suzanna moves computer back to take over the use of the keyboard. She changes browser tab and navigates to the section of the code which deals with the design of levels and placement of different game components.

13		
14	S. Well. Where are you going to put your door in level two? Do you want to change the layout at all?	Suzanna gestures to screen
15	Olivia: What's door?	Suzanna doesn't move laptop but gets closer to take over keyboard
16	Suzanna: d	
17	Olivia: Ok	
18	Olivia: d . Door right there.	Inserts a "d" in an array high above a platform

For the preceding ten minutes, the parent and child dyad had been struggling with a tutorial chapter in the online course manual textbook, included as Appendix.tech.samplechapter, which explains how to add the pattern of keys and doors. In this pattern, players must collect a key and then navigate to a door to progress to the next level. The parent, Suzanna, had been making multiple changes to the code while the child, Olivia, expressed boredom. Suzanna then suggested that Olivia ask the facilitator, Mick, for help in resolving the coding issue. Once the bug was resolved, Suzanna handed the keyboard to Olivia to test the now working game. The implementation of a new game design pattern (GDP) involves adding new code to the game. The

amount of code and changes required varies depending on its complexity. Larger changes typically require more careful manipulation of code and subsequent debugging. The key and door pattern is relatively technically challenging to implement. Olivia is one of the younger participants, and the parent, Suzanna, has taken on the role of solving harder coding problems and, in doing so, has built important proficiency in coding practice.

The parent takes action to involve her child in the design process when technically possible and to incorporate her child's interests. Suzanna overcomes her own hesitancy about asking for facilitator help by encouraging her child to seek assistance. When the coding problem is resolved, Suzanna signals to Olivia that it is her turn to return to the design process. This indicates that Suzanna recognises the greater potential for effective input from her child at these moments. After the blockage caused by debugging is removed, Suzanna immediately re-engages Olivia, prompting her to play-test the game. Suzanna skilfully navigates the process, eliciting and incorporating her child's interests in a variety of ways.

The child also shows initiative, particularly in relation to level design. There is an awareness of and playfulness surrounding the conventions of a platformer game as a dynamic system, which drives their work on the design and coding mechanics of the game. An understanding of the game as a dynamic system becomes especially evident in the parent's alarm when, shortly after this interaction, the child deletes all elements of hazard in one level. The pair use the names of the game design patterns they are working on (Adding Levels and Keys and Doors) (Key 44 in Vignette 2). Evidence suggests that Olivia (child) is building a connection to the game-making pattern. In Key 12 in Vignette 2, abbreviations of the GDP names, Key and Door and Levels, are repeated in a sing-song voice. Additionally, my description of the Key and Door pattern as a challenging pattern during the session's starting orientation, outlined in Vignette 11, provokes a strong reaction. The use of GDP terms as a lingua franca, a concept that underpins the broader utility of design patterns (Erickson, 2000), to facilitate designing for others is explored in the next section. More broadly, the use of GDP concepts as an evolving design language for providing feedback, sharing knowledge, making requests and other purposes during community activities, pair interactions and individual work is explored in the following sections.

Use of GDPs in the cultural plane / community level of activity

Following Rogoff's (Rogoff, 1995) three planes approach to analysis, the following section explores the use of GDPs at the community plane of activity. While Rogoff's earlier analysis of activity on a cultural plane emphasised the metaphor of apprenticeship, focusing on an already existing community, her later work with Gutiérrez (Gutiérrez and Rogoff, 2003) places greater emphasis on cultural activity as dynamic. This perspective examines existing and emerging norms and repertoires, adapted from participants' engagement in other communities and contexts.

The role of GDPs to facilitate learners to design for others

The use of GDPs, particularly during playtesting, can support the process of imagining end-user experiences. This section examines how the concept of, and experience with, implementing

GDPs encouraged participants to envision the experience of end users of their games. Chapter two explored the proposition from professional and participatory design processes that design should be informed by end-user experience (Redström, 2006), as well as the challenges involved in undertaking such an "*operationalization of empathy*" (Surma-aho and Hölttä-Otto, 2022, p. 1). From a CHAT interpretation, this process involves shifts in perspective by participants as they engage with objectives across different scopes of activity. For instance, in the vignette above, Olivia (child) pursues a quirky design goal during her paired design work, which Suzanna (parent) resists. Imagining a shift in perspective to the intended audience at a community level of interaction, the parent aims to ensure a sense of challenge for the imagined player.

Suzanna shares "*Must be quite hard to get through that door.*" when Olivia places the exit door high above a platform. She then continues, "*It's no fun having a game without any hazards to avoid.*" Olivia (child) seems determined to remove all hazards. "*It is for me!*" she counters. She may be aware of the implications for game balance but takes pleasure in this destruction of the key challenge of the game as an act of disruptive play. However, a later interaction with a peer shows that Olivia (child) is indeed imagining the experience of the immediate audience of fellow game makers and supporting students. "*I like making it frustrating. That other people find it frustrating!*" Olivia (child) notes the persistence of a student helper who pushes past her frustration to complete the game. Her remark, "*If people tried hard they would get to my level,*" shows her awareness that not all players will persist in the same way to reach her final level, which has only rewards and no hazards. This being a "*secret, special*" experience, which plays against the norms of platform game design, is thus intended to provoke player surprise.

There are other examples of how concepts of game challenge and other aspects of gameplay experience evolved through informal feedback during playtesting and served to influence peers to modify their games to increase the enjoyment of peer players. The discussion of game challenge, specifically comments about how 'hard' participant games were, was a particularly common interaction during playtesting. Vignette.challenge outlines a key interaction which demonstrates norming behaviour towards Molly(p) who has concentrated on the graphics of her game to the detriment of the level of challenges. In particular, the controls of the game are frustrating. The vignette shows varied attempts to influence Molly to change the game variables to make the gameplay less frustrating. They praise the look of the game but offer feedback on the experience of the game mechanic of jumping. While the players do not tell Molly directly to change the game, these comments appear to direct the direction of the design to comply with an emerging community norm of how a player jumps should feel, stemming from the participants' feedback on the feeling of lack of control over the player's character in the game. These behaviours are some of the informal norming behaviours that are less directive, seen in the work of Rogoff and colleagues, as explored in the literature review (Rogoff, 2003). DEVELOP THIS POINT.

In interpreting data there were other examples of pair partners and peers either commenting on or suggesting to others that they should imagining others user experience to suggest game design alterations. BRIEFLY LIST HERE

INSERT VIDEO DATA SUPPORT E.g. The use of GDPs to facilitate learners to design for others was present in 6 of the 12 video capture sessions analysed.

GDPs facilitate the use of wider funds of knowledge and interest

GDPs can allow participants to share and explore funds of knowledge and interest in the emerging learning community. One premise driving my exploration of the use of gameplay features in pedagogy is that the tacit knowledge of gaming conventions among most family members is extensive, even if they are not avid gamers, due to the extensive influence of video game culture in mainstream culture, including contexts of family life (Itō *et al.*, 2010; Livingstone *et al.*, 2018), particularly retro gaming (Heineman, 2014). As such, the process of facilitating ways to surface and work with such tacit knowledge aligns with Moll's concept of funds of knowledge, and, potentially, funds of identity (Esteban-Guitart and Moll, 2014; Fasso and Knight, 2020).

Within a CHAT framework, these funds of identity and knowledge can be seen as tools that facilitate the transfer of potentially useful concepts and practices across the boundaries between activity systems, in this case, between home activity and that happening in this game-making community. The associated inclusive benefits of working with participants' home interests are explored by Barron (Barron, 2006, 2010) and similar work by Gutiérrez on third spaces (Gutiérrez, 2008). Existing research outlines the motivations for social making incorporating gaming cultural elements (Gee, 2003; Itō *et al.*, 2010) and the ability to make personalised game assets and narratives, which sustain engagement in digital making projects (Sefton-Green, 2013).

In the last chapter I outlined processes facilitating participants could to create and incorporate graphical, audio assets and narratives elements into their games and, referencing Appendix.tech.gameframework briefly outlined how these elements were themed as *Game Polish* items within a typology of GDPs which aligned with the MDA (mechanics, dynamics and aesthetics) game element framework explored in Chapter 2. Addressing the aspect of patterns aesthetics, in Vignette 5 Molly and Nadine showing identification with graphical making process and the process of bringing it the game. Beyond their internal family interaction, they share this artistic flair and attention to detail into the emerging learning community.

In the last chapter, I outlined processes facilitating participants in creating and incorporating graphical, audio assets, and narrative elements into their games, and, referencing Appendix.tech.gameframework, briefly outlined how these elements were themed as *Game Polish* items within a typology of GDPs, which aligned with the MDA (mechanics, dynamics, and aesthetics) game element framework explored in Chapter 2. Addressing the aspect of pattern polish/aesthetics, in Vignette 5, Molly and Nadine show identification with the graphical-making process and the process of bringing it into the game. Beyond their internal family interaction, they share this artistic flair and attention to detail with the emerging learning community.

In addition to issues of aesthetics explored in the section above, home knowledge of game mechanics and dynamics, the M and D of the MDA framework also contributed to emerging cultural repertoires. Addressing experiences of game dynamics, as explored in the previous section, Vignette 4 gives several examples of participant feedback on the dynamics of Molly's game during playtesting. Turning to game mechanics, in part two of Vignette 7, Bertie comments on Toby's game, which has a dominant game experience of timed jumping: "*It's like parkour in Minecraft but timed. It's like playing the game Wipeout. Have you ever played Wipeout?*"

Parkour in Minecraft and Wipeout are both game experiences whose main gameplay mechanic is about judging jumps and landing accurately. Bertie makes links to his existing experience of games, making comparisons between Toby's game, commercial games, and his own. In doing so, Bertie is able to show his knowledge and analysis of gameplay patterns.

The different types of GDPs available to participants to choose from in the menu of GDP documentation and code snippets allowed for a choice in participant making paths, enabling alignment with home interests. While this study is exploratory and not designed to be representational, observations from video data in this study support existing research on the use of funds of knowledge and interest in digital projects as motivating factors and as a way to overcome barriers to alien cultures (Gutiérrez *et al.*, 2019).

SUPPORTING DATA FROM VIDEO ANALYSIS

Evolution and propagation of GDPs concepts during playtesting

Within the emerging idioculture of the game-making sessions, and during playtesting in particular, the implementation of popular or novel GDPs by participant pairs or individuals is often spread through peer activity. Both documented (those part of provided resources) and novel GDPs (those introduced by participants) were transmitted between participants, enriching the games of participants. At times, participants were influenced by playing the games of others, and at times, they would request the direct help of peers to implement GDPs in their own games. In part two of Vignette 1, Toby's work adding 21 levels to his game is noticed by Bertie.

Bertie: Why's that enemy in every level

Toby: He's not.

Bertie: Can you show me how you add more levels on to yours?

Toby: Yeah sure.

Pause

Toby: I'm just going to have one go of beating this (refering to his own game which he is playtesing). It's 21 levels in it. So .. Yeeeeah.

Pause

Bertie: It's like parkours in Minecraft but timed. It's like playing the game Wipeout. Have you ever played Wipeout?

Toby: Er not really.

Bertie: Or seen it?

Bertie: That's like my second level.

Toby: Ah so hard (Toby fails at a high level on his game and starts to move off)

Toby: (To someone else calling for attention) No I'm helping Bertie.

*_(Toby then follows Bertie to his workstation to help him implement more levels.)_*

This exchange shows the propagation of GDPs emerging from the process of community playtesting through a direct request. It is possible that Bertie's request to Toby is prompted by Bertie noticing Toby helping Nadine and Harper add levels to their games. Toby's emerging role as a specialist that the community can draw on for practical help, and similar examples (see also that of Nadine in Vignette 5), were welcomed by me as a way to reduce dependence on myself as a facilitator, or to provide an alternative to the instruction-based support documents. Other examples of propagating patterns include placing hazards in tricky places like a lava pit, the use of moving enemies and changes to jump dynamics.

In addition to the propagation of GDPs offered in the menu of supporting documentation, other gameplay patterns and related design concepts emerged organically from the community. For example, Olive's concept of a level that eschews challenge and offers only rewards, a type of bonus stage pattern, emerged through playful experimentation (see Vignette 2). Similarly, the concept of a safe zone in the game of Pearl and Clive arrived as a direct result of adding a moving enemies GDP. The extensive use of that pattern dominated the game challenge to such an extent that it became essential for players to quickly identify and use 'safe zones'. These novel concepts may have emerged spontaneously or have a source in part from participants' funds of gaming knowledge. In either case, once the concept of safe zone or bonus level enters the linguistic and coding repertoire of this community of game makers, they may spread if other participants pick up on them. Even if other participants only noticed the use of GDPs and used approximate terminology to comment on them during playtesting, this form of activity can be seen as legitimate peripheral participation (Lave and Wenger, 1991; Guzdial and Tew, 2006), and thus a helpful practice contributing to the emerging ecosystems of the learning community.

This section has explored the use of GDPs within strategies and processes that occur at a community level in a way that demonstrates the transfer between different activity systems, supporting Gutiérrez's perspective (Gutiérrez, 2008) regarding the importance of transcending borders and linguistic practices between sites of learning. Playtesting served as a medium for a shared idioculture to emerge and, as such, became a significant part of the pedagogy of the learning design. While the design processes directly linked to implementing GDPs were among the most significant in the video data, others were present. The role of playtesting and other community interactions, including warm-up games and session reflections, are explored in relation to the development of agency in the following chapter.

GDPs used in the process of guided participation (Target 2500)

This section explores data on the use of GDPs from the perspective of interpersonal interaction. Here, the process of being guided into planned activity or peer work is exploratory, contributing to interpersonal processes that serve the completion of the activity. Based on observations of video data, I describe four categories of how GDPs are used in diverse meditational strategies, most frequently through pair participants' interactions.

GDPs and their grouping as a collection or menu of suggested options, offer the potential to assist project navigation, acting as a framework for code examples and step-by-step tutorials (600)

The use of a catalogue of design patterns addresses the tension between participants' choice and the need for documentation to support the technical implementation of code structures. However, not all participants used the provided instructions and developed diverse methods to access support for implementing patterns. At the cultural level of activity analysis, the previous section outlined how providing a menu of game design patterns enabled participants to select patterns aligned with their personal interests, potentially reducing feelings of alienation from coding or game-making culture. This section further examines how a curated collection of game patterns, along with supporting documentation, created opportunities and influenced participants' experiences on the interpersonal plane of activity.

The previous chapter explored, from my perspective as a facilitator, how the use of a menu of GDPs helped address the tension between participant choice and the need for supporting materials on code implementation. Vignette 9 shows one of the few short episodes where I addressed the whole class, outlining the webpage listing the menu of patterns: "*It's almost like this is our control panel. (you decide) what do you want to do next and we jump off from there.*". In addition to using written documentation as a framework, in P2, I guided supporting student facilitators to ask what patterns participants were currently working on or had recently completed as a way of provoking informal reflection on project progress. Given the integration of these elements into the learning design from P2 onwards, it would seem natural that participants would use the provided framework, at least in some capacity. Despite this, the evolving use of this documentation, the variations in its use, and the interpersonal interactions surrounding it merit further attention due to the diversity, evolution, and fundamental importance of GDP use at this interpersonal scope.

Vignette 2 in this chapter shows the parent Suzanna's use of a step by step tutorial to address the child's objective of implementing chosen GDPs. Olivia is limited in her technical skills but leverages the concept of two GDPs in particular to persist in problem solving using the tutorial and trouble shooting using facilitator help. Given Olivia's limited computer literacy (explored in the Vignette commentary) the idea of the GDP is likely to be from a cultural source, most likely playtesting. By way of contrast, Toby in Vignette 1 engages in lone working, copying a code from code snippet example of the game design pattern which he chose from a graphical menu of patterns. His independence in adding new features using supporting documentation without facilitator support helps him take on a mantle of a mentor inspiring and helping others add the same GDPs to their games. (EXPLORED ABOVE?)

OTHER DATA HERE - Examples from the data of participants using GDPs as a framework for organisation and project navigations.

These varied forms of documentation, and their varied adoption and propagation represent different approaches to scaffolding the process of becoming familiar with and then modifying to the code in often increasingly ambitious ways inline with the stages and ethos of a use-modify-create (UMC) approach. [source]

Key characteristics of structuring support around quick start activities and a subsequent menu of GDPs which align with existing research include:

- supporting the value of informal approaches to learning technology [ITO etc] whilst avoiding the lack of support in P1 where participants called out for more support with technical and organisational processes.
- providing shorter, just-in-time tutorials to reduce barriers associated with more extensive instruction-based approaches including barriers of literacy (Resnick and Rusk, 2020; Dietz *et al.*, 2021).
- The use of a menu / collection of resources can be considered an inclusive educational practice and offer a rich description of a concrete example which aims to align to inclusive, practice-based frameworks like UDL and PBL. While this may involve more work in advance, the benefits to inclusive practice merit it.

GDPs used to scaffold ideation and prioritisation processes (800)

GDP concepts and supporting tools can guide ideation and organisation processes in diverse ways including use as a lingua franca, as a visual prototyping tool, and within participant gestural interaction. The design chapter explored the tension between reduced choice of genre of game. In my journal notes on the shift from P1 to P2, I note that the provision of a menu of GDPs significantly decreased in time spend in ideation phase by providing scaffolding and a restriction of choice as outlined in Vignette 1.

Video date surfaced frequently use of GDPs terms as a lingua franca to organise and prioritise game making activity. The following exchange from Vignette 3 between participants Ed and Mark shows the use of GDP concepts to help resolve a tension between a more chaotic style of working jumping from one goal to another and a parental motivation to prioritise work.

```
Mark: I've brought the music, and also we could just concentrate on one thing
and just change that. You know, keep working through.
Ed: Yeah. I think I want to get an enemy in - oh no - my person animated.
Mark: So you want to get your person animated that's the main thing.
Mark: Shall we concentrate on that and changing the platforms into something
different?
```

The pair's initial listing of features is a brainstorming technique uses of the approximate names of of game design patterns (*get the person animated, get an enemy in, changing the platforms, make a theme tune*). The vignette analysis sees the parent *overwhelmed* with the child's lack of focus on one pattern "that's what I mean, you can't just skip around like that".

In previous section explored the use of home FoK within cultural interactions. In Vignette 7 home-base knowledge of GDP mechanics is utilised on an interpersonal plane, with Dave(p) helping Toby(c) via guided faciliattion to help shape new design ideas. Dave() draw on the game playing experiences to promote innovation in the design of the existing template: "*the previous style of game was a platform (makes shape with hands) game wasn't it? You went along and there was gravity pushing down (points down). There are other types of games aren't there?*" Dave invites thinking outside of the constraints of the suggested design early in this first session. In this example the family bring their knowledge of game play patterns and genre conventions

and more broadly experience and a perspective of coding as a creative challenge to the activity.

In the same vignette, we see the use of GDPs to allow the spacial exploration of design in a visual and intuitive way that suits being mapped onto paper, or onto graphical software which allows for a similarly intuitive and rapid prototyping. The pair alter their use of the tilemap tool to create a maze game instead of a platformer game. Toby is able to map existing knowledge of tools and home knowledge of the target game genre to rapidly make revisions.

```
Toby: Oo. Shall we try to make it (unintelligible). Cos in pac man you can go
off the edge.
Dave: and you wrap round the other way?
Dave: Yeah, yeah. We can do that. Save that for version 1.1
Toby continues making changes to the code design.
Dave: What's the theme? What are you drawing?
Toby: What? I'm trying to make like a pac-man type thing.
Dave: Alright. What if you sketched it on paper first? Or have you got it in
your head?
Toby: I'm just kinda going for it it. (laughs)
Dave: Ok go for it, see what you get up to.
Toby: I'll leave a hole there.
```

Changes in the code which Toby is altering to impact on the new design pattern of a top down game, are immediately apparent in the preview window. As such, Toby does not feel the need to prototype on paper. The use of GDPs to support visual prototyping is explored in more detail in Appendix.tech.prototyping

To end this section, it is of value to link this use of GDPs terminology and visual prototyping techniques, to existing research on the development of shared language to support joint work and guided participation. INSERT RESEARCH on joint attention, and joint problem space and the importance language in this process from a sociocultural perspective.

GDPs aiding the process of division of labour (1400)

Addressing division of labour on an interpersonal plane, participants worked mostly as pairs or individuals, alternating between community playtesting and pair/individual design work. Turning to the use of GDPs specifically in DoL as an organisational strategy, structuring work processes through implementing modular sub-projects in the form of GDPs can aid division of labour.

GDPs feature in processes division of labour in a variety of ways: working knowledge of different types of GDPs can help participants to divide work by working on different patterns or taking on project elements within patterns.

Returning to the example of Mark and Toby from Vignette 3, Mark appears to be keen to quickly pick one pattern, then then work through the documentation on that pattern, a process he later refers to in interview data a *plodding*. At a different stage Mark is engaged puzzling over documentation on how to add animation to a character for some time. This results in Ed being blocked from progressing. In an exchange listed in Vignette 4, the child proposes dividing their labour informally, appearing to make a tactical decision allowing the father to specialise in GDPs that involve deciphering technical instructions, whilst he engages with a pattern that involves

creating audio and graphical assets in a more experimental process. The use of GDPs concepts here is used in the young person's circumvention of a tension via his informal division on labour.

Analysis of video data uncovered flexible way in which the participant divided the process of game making, often showing traces of home collaboration practices. For example, In vignette 2, Suzanna(p) and Olivia(c) relied on the parent to do the majority of code implementation and shared one computer. The pair also alternated between use of keyboard and mouse to give the child hands on experience when possible. In interview data the parent notes "*I resist the urge to fix things immediately when she struggles.*" In an illuminating exchange, Olivia uses the name of a GDP within a request for her parent to take on a specific task within their making process.

```
 Olivia: Go on then. Key - Door - Person.
 Suzanna: Person?
 Olivia: Key Door Person.
 Olivia gestures with her hands to indicate that her mother is the person she
is referring to.
```

The utterance by the child "Key Door Person" work on the game design pattern called Keys and Doors to the adult. The child appears to consider the level of complexity needed to add a new pattern into the code to be beyond her ability and thus directly delegate the task to her mother. While, the parent took driving seat to overcome code blockages they were careful to then involve the child to input their design choices when solved.

After the child has delegated a coding task to her mother she undertakes other activities. At times her activities directly contribute to the main goal of game making. At times the parent asked the child to seek help from facilitator. On another occasion when the child appeared bored of waiting for parent to solve a code problem, she approached the facilitator to ask for help on behalf of the adult without prompting. At other times she engages more peripheral activities such as watching older children playtest each others games, or observing community activity from under the table.

While in P2 when supported by grandparents Toby had worked mostly alone, in P3, he works as a pair with his father (Dan) in a more guided practice. In several interchanges the father starts as a facilitator taking a lead from the direction of the child. As the child reaches the limits of their ability he begins to be more directive, by asking leading questions and testing existing knowledge. Finally, in order to complete the programming or research tasks beyond of the child's knowledge, the father is more direct in instruction, directing the research and proposing a coding solution for their new game design pattern. Dan's experience as a software engineer and volunteer at Coder Dojo (Glossary), in an interview extract he outlines being present to focus on facilitating learners to develop independently.

```
I try never to touch the keyboard of who's there. If they are stuck on
something I always tell them what to do. Even if it's then taken me five
minutes to explain what a semicolon is. And point. _It's that key_. Because it
was just, I could do it so effortlessly. I think I'm sure I put people off
very quickly by "Dan did something really quickly. I don't know what it was.".
```

In the second part of vignette 7, Dan(p) outlines a similar approach to Suzanna(p) in resisting *jumping in*. "I try never to touch the keyboard of who's there. If they are stuck on something I always tell them what to do." This approach is guided influenced by Dan's experience as a software engineer and volunteer at Coder Dojo (see glossary). In the first part of Vignette 7,

Dan(p) shift the kind of helping role as he progresses through different stages of applying a GDP. MORE DETAIL?

In Vignette 5, Molly continues to do solo design using the Piskel graphical too. She encounters a design problem. When erasing a part of the design she gets rid of background colour. Molly asks for help from partner but receives misleading advice which does not help her progress.

Molly: Oh no it's not done that has it? Molly calls the name of her child across room with theatrical gesture and loud whisper voice Molly: "Nadine !" Molly then makes face, wiggles head and shrugs at parent peer. The other parent laughs. Nadine arrives to help.

```
<!-- The implementation of some GDP involved the use of different tools and activities. As learners build the familiarity with the component actions needed to implement design patterns, some start to specialise as they divide labour between pairs or Rather than promising the transfer abstract concepts to other domains, we see learners build competency in participation in replicable processes. These processes which aid future iterations of the GDP implementation design cycle. The process of operationalisation of these sets of actions contributes to the creation of an informal, complex networked resource of operations which complement the more visible curated catalogue of GDPs.
peers. -->

<!-- Structuring code implementation on relatively small code stages of GDP implementation steers the production process towards relatively frequent iterations of design cycle. Specifically the design stages here typically involve; ideation, planning, implementation, testing and modification and sharing via playtesting. THIS IS EXPLORED IN THE NEXT CHAPTER SO DROP OR ONLY BRIEFLY REFERENCE. -->
```

Discussion on division of labour in the video data

Leveraging the possibilities of emerging divisions of labour: While the context of the participants as families involved in home education makes any general claims difficult, these observations support exist in other research in this domain. For example, research shows that children have the potential to help parents as technology brokers (Correa *et al.*, 2015). In joint technology work parents can fulfil several,reciprocal roles including, collaborator, resource provider, learner, non-technical consultant and emotional support (Barron *et al.*, 2009).

Thus, in response to the creative support that parents and siblings provide, facilitators should design learning environments to facilitate these possibilities. The work of Roque provides guidance for helpers in the process to support parents to value and feel confident these roles (Roque, Lin and Liuzzi, 2016) in a way which mirrors the use of helpers this phase of my study.

While these examples involve GDPs others are explore in relation to development of agency in the following chapter. The examples above illustrate, part some of the varied strategies to divisions of labour adopted by participants at times involving complex tensions in activity. For example Molly and Nadine reversed traditional helping relationship, and the child would reluctantly implement the parent's requested technical elements of GDPS but would pointedly not explain the changes made, seemingly taking pride in knowing something that her mother

didn’t. Similarly, while Olivia’s activity away from the screen while non-productive within the scope of technical progress, can be characterised as legitimate peripheral activity of observation of community activity (Lave, 1991 ; Rogoff, 2014). Indeed the possibility for children to *not* engage in community activities is seen by Rogoff (2003; 2016) as an important characteristics in participation based models of learning.

The variety of repertoires of helping practice depends in part on the different funds of knowledge the parent has access to. Toby and Dan’s pair process is more guided and focused than many other participants and includes accessing professional documentation and exploration of computational thinking concepts.

While the helping pattern of accessing the repetoire of researching and accessing technical documentation is available to all pairs, due to the level of skill and experience involved, other technical processes are developed by most participants and are explored in the next section.

The role of GDP used in emerging technical processes (900)

Turning from the division of labour to how GDPs interact with the development of use of technical processes in game making. The implementation of GDPs can stimulate participants to adopt or to share technical processes.

The above example from Vignette details Molly getting help in exporting graphical files to be imported into the game code project is example of GDPs taking a role in the development of technical processes. Here the parent needs to learn or at least to draw on peer knowledge to complete the implementation of her chosen pattern. Similarly later, Molly is helped in the use of the cropping tool by a peer child Ed(c)

A further example is the the process in *code patching* (see glossary for terms in italics) in Appendix 1.

Code Patching as a guided technical process and the use of GDPs to nurture tactical responses to coding errors

Code patching, the process of copying and pasting fixes into on-going projects, is outlined in Appendix.codepatching. The process of code patching is similar in that the documentation is structured around GDPS, and thus similar aspects of an appropriate organising unit apply.

Code patching is a processes which suits being organised around GDPs aligns with the UMC process.

The patches, and patching process became an important process in documentation. While code was provided, in practice mistakes were still made and additional customisations were made.

Technical processes were sometimes introduced in a basic form through scaffolded activities, supporting documentation or evolved from informal interaction between peers. Even introduced processes evolved, being modified by the community.

Thus while use and propagation of new technical processes were often motivated by a drive to implement or complete a GDPs, the following observations bear in mind that the relationship between GDPs and emergent technical processes is complex and interdependent rather than assuming that GDPs directly facilitate emerging tool use.

The following paragraphs address x, y and z

Increasing technical fluidity of general computer use and software tools

Whilst pursuing the motivation of seeing their GDP-related change in the game many participants become fluid in certain aspects of general computer use and more specific technical processes involved with software use of game making tools.

Examples of processes being adopted by participants through the implementation of GDPs including the fluidity of navigation between playing and coding window shown by all younger participants and many adults and the development of keyboard and mouse coordination to facilitate navigation within the code environment and external support resources to facilitate the code patching process.

GDPs encouraging sharing of technical practices

In Vignette 5.a, we can see that the process of changing pen colours on the graphical tool is one which the child has been able to translate into a effortless process whereas the parent is still consciously building her competency. Nadine has operationalised the process and it becomes part of the toolset of practices that she can draw on [CITE]. Molly also benefits, the expertise of her child to undertake that process is added to her distributed toolset. However, she is also keen to develop her own competency as indicated by her asking child to explain the process.

References to such emerging technical practices were present in many exchanges during informal playtesting and pair interactions. In the following example a reference to a GDP is used to explain the need for the technical process.

In Vignette 5.e, after giving feedback on the jump speed Molly's game Ed sharing a process to redundant space at the edges of design sprite characters.

```
"Ed: for people with background like yours You can use the cramping tool."
```

Ed leads Molly to his workstation and involves his father Mark in the process. Mark uses an example of a GDP in his explanation of why to undertake that process, to not "set things off when you are close to them".

Thus rather than GDPs helping the graphical cropping process, it provokes Ed to notice and suggest it to Molly, and Mark to evoke it as a reason to undertake cropping.

Solving / Debuggging code problems that responses are helped by the high level of context present in the use of GDP as a structural design framework.

In critique of online tutorials researchers note few provide common errors (Kim and Ko, 2017). While it seems reasonable not to include all possible errors, the challenge of supporting participants to build skills and effective repertoires involving debugging is widely recognised [SUPPORT]. My observations suggest that the close coupling of code implementation and target behaviour aids participants in debugging and also be profitably can be drawn on by facilitators.

In analysis of journal notes and recorded screens I began to identify different kinds of coding problems that blocked participants from progressing and build proficiency and flexibility in addressing them. Reflections on participant experiences of kinds of errors are explored in Appendix.tech.2. In the appendix I reflect on syntax errors which stopped the game from working entirely, bugs where code changes had no effect on the game and glitches which don't crash the game instead creating an unintended effect. Glitches merit a summary here as a particularly interesting site of guided participation. In my reflective notes I propose that glitches are more motivating to correct than other errors.

Aspects of the design, use of a starting template and supporting resources based on code snippets and thus code patching process increased appear to encourage glitch bugs. See Vignette 1 for an example of a glitch bug which provoked further investigation and hand on tinkering. A diversity of helping strategies evolved, my own and those of others, in response to errors. For my part, these altered when I judged when participants to be receptive to different forms of input. Glitch errors were often embraced as curiosities and as a learning opportunity to understand the related code and associated abstract concepts using a concrete example afforded by the mechanics of the game design pattern. Mirroring the proposition of half baked games theory that a incomplete / templated game are motivates initial participation, my observation support a position that glitch bugs, which move game into a broken state also provide motivation. Additionally, as the bug relates to a targeted GDP which is closely coupled with a suggested code solution and tutorial, this process incorporates the benefits of a just-in-time approach to supporting documentation

The debugging process at times surfaced and allow the exploration of surfacing of computational thinking and other curricular concepts, a topic explored as part of the next section. This area opens some interesting lines of questioning which are beyond the full remit of this research but are included in Appendix.tech.2 as promising areas for future work. When addressing issues of motivation and efficacy a different research methodology would be needed.

The role of GDPs in the personal appropriation of concepts and processes (500 - Target 750)

The previous section began to touch on the personal appropriation of coding skill via the debugging process.

Rogoff's third plane of analysis, one which mostly closely aligns with much educational research is that of personal appropriation. Rogoff is careful to underline the bi-directional nature of the appropriation of personal knowledge, emphasising Vygotsky's foundational view of knowledge as being mutually constituted as participants feed their interpretation of knowledge back into the communities of which they are a part.

As outlined in Chapter 2, the focus on skills and knowledge developed by participants is one of the strongest threads in related research on digital coding and game making, and particulary in that of constructionist school (Papert, 1980; Kafai and Burke, 2015). Papert's research on Microworlds saw one of the roles learning designers to shape the affordances of playful learning environments to steer learners to explore scientific concepts, mathematical concepts, for example geometry in Turtle computing. In this research, detailed study of young coders evolving code creations and accompanying interview data allowed a close study of the personal appropriation, including elements of understanding, application and sharing back this knowledge via created and shared digital projects. Much subsequent research on the use of coding to develop both concrete and more abstract computational thinking skills, has neglected this social and perspectives [find source], instead focusing on a personal dimension of learning as a curricular skill to be absorbed and tested. By way of contrast, this study is aligned with the development of the constructionist movement towards socio-cultural approaches to transcend individual personal conceptions of knowledge, exemplified by the issues associated with the UK computing curriculum. In this informal learning context, there was scant desire from participants to explore concepts aligned with computing curriculum, instead driven more by practical desire to make a game using the tools provided.

Despite this positioning, given the potential to adapt this learning design to a more formal environment to address needs outlined in the introductory chapter, from P2 onwards I began to augment the learning design to suit curriculum oriented teaching and assessment of computational thinking concepts and skills, a process which resulted in a learning map of curricular concepts included here as Table 6.x .

Coding Concepts	**Systems Patterns**	**Design Practices**
Sequences	Systems Elements	Goal Setting
Variables	Systems Dynamics	Being Incremental and Iterative
Logic	Reinforcing Feedback Loops	Developing Vocabulary
Loops	Balancing Feedback Loops	Web Navigation
Arrays		Problem Solving
Creating Functions		Version Control
Change Listener		Debugging
Input Event		Reusing and Remixing

Table 6.x. Learning Dimensions of the 3M Game Making Model

Appendix.learningmap explores in more detail the evolution of this map, which acknowledges significant tensions in integration of concepts into a informal context, and draws on the process and rationale of Bevan and Petrich's (2013; 2015) work to bring a similar learning map to a seemingly chaotic tinkering and making process in museum contexts. A summary of my book chapter (Chesterman, 2023) exploring these issues with recommendations for practitioners is included in Appendix.learningmap

This exploration focuses on the potential of the 3M learning design rather than making claims on the suitability to develop CT via game making which is explored extensively in other research.

Conceptions of personal appropriation should not be limited to only the curricular concepts included in the learning map, the many of the behaviours explored above show both the appropriation and feeding back of varied understandings of game design patterns and related concepts and skills. While the design of this study does not suit full exploration of the personal plane of activity, the repeated, solo, incremental changes of the details of implementation of game design patterns indicate a personal appropriation of concepts like game feel and challenge and design processes.

Given that this dimension is not explored in the thesis which has a greater focus on community and social aspects of use of GDPs, examples of personal appropriation are included in Appendix.learning

The ongoing reflection provoked by the process of privileging social and cultural aspects of the game making activity has yielded more philosophical questions in relation to the learning process which are explored in the following discussion section.

Discussion

The majority of this chapter has focused on findings and immediate interpretation of participant and facilitator use of GDPS as a mediational strategy / repertoire. This section takes a step back to explore the wider perspectives of the findings beyond the immediate context to digital making and project based approaches and understandings of design patterns in educational contexts.

In this discussion section, I:

- explore the uses of GDPs in relation to varied conceptions of abstraction, chiefly CHAT and CT
- propose that GDPs provide a suitable way to engage with coding practices partly due to their position between abstract computational concepts and concrete implementations of code structures.
- explore potential implications for facilitators and researchers on use of GDP collection as an intermediate pedagogical construct.

Discussion of findings in relation to CHAT conceptions of abstraction

The previous section began to explored, through the lens of personal appropriation, to variety of skills suitable for development via this learning design. The final section surfaced a distinction between curricular contents commonly associated with a more abstract understandings of personal knowledge, including more abstract elements of computation thinking. This section continues an interpretation of this chapters findings and an socio-cultural understanding of learning using activity theory as a framework. This investigation has required a careful investigation of abstract-concrete dimensions of emerging knowledge happening in the game making and digital making more generally. To start, let us consider the nature of the abstraction of gameplay design patterns via Sannino' (2011) distinction between empirical and theoretical

abstraction.

"While observation and categorization are actions at the root of empirical abstraction, practical transformation, change, and experimentation are actions at the root of theoretical abstraction. Theoretical abstraction allows one to generate and project complex, theoretically mastered concrete manifestations and developmental forms of the reality under scrutiny. A theoretical abstraction is based on a functional relationship, also called a germ cell." [@sannino_activity_2011].

In this research, the empirical level of abstraction is present in P1 and in P2 where features to be added to the game were classified gameplay features and categorised according to to the existing MDA framework. Moving beyond the empirical towards the theoretical aspect, is the early positioning of each feature as a gameplay design pattern which can be described and supplemented with a suggested code structures and examples. ANY OTHERS

The key, guiding conception here becomes the pedagogical utility of a strong coupled link between a feature of an observable game product (in this case gameplay design patterns) and scaffolding resources available within a community. This concept becomes the root of the varied approaches and forms of support outlined in this chapter. Some instances based on this kernel concept were created by myself in response to user requests in varied forms of support documentation outlined in Chapter 5 (e.g. quick start cards, a home menu of GDPs, code snippets, stepped tutorials). Many other examples, as outlined above were more spontaneous and informal, spreading from peer to peer in the community often during playtesting.

The process of identifying and working with GDPs in this research process can be mapped as a dialectical process of moving (rising) to the concrete (Sannino, 2011). It is this movement from a kernel idea into varied concrete instances that is the root of the power of the approach according to CHAT scholars. While, the varied uses as meditational strategies and their integration into varied concrete forms of creative cultural and interpersonal repertoires many be particular to this setting, the germ cell concept described above is a theoretical and generalisable abstraction not particular to this context.

While the educational use of design patterns and has been explored in other settings via research, the focus has been on professional rather than novice learners.

While I address areas for future research in the conclusion, a proposed benefit of this research is While understanding the process is more obviously useful for facilitators - it may be helpful to participants as well as they build familiarity with frameworks of features of multimedia projects.to underscore the potential and to uncover a starting point for pedagogies that can be transferred to diverse design projects.

Design patterns as a intermediate-level, pedagogical framework compared to computational thinking

It is productive to ask: how can the use of GDPs as a framework be compared to that of computational thinking given CT’s is a dominance in this area?

In chapter two I described the characteristics and limits of computational thinking as a

pedagogical framework, specifically concepts include: varied interpretations; the role of levels of abstraction for teachers and learners; and the value of epistemological pluralism as a way to value concrete approaches.

Waite's interpretation of levels of abstraction and semantic waves in a computing education context is potentially relevant as example of how can contribute to the debate in this area. In terms of semantic waves, the movement between abstract and concrete is helpful to address some of the issues concerning inclusion and accessibility of overly abstract approaches (Papert and Turkle, 1990). However, if we return to Papert and Turkle's (1990) challenge to the legitimacy of the primacy of abstract, top-down approaches to teaching coding, such an approach would also challenge Waite rationale of alternating between abstract and concrete poles.

While, none of the author's above are clear about what exactly is meant by the abstract in their explorations, a synthesis of relevant literature supports an spectrum of abstraction, from Wing's take on CT at the most abstract, and specifics coding constructs or even syntax at the most concreted. Such a spectrum is in is broadly in line with LOA interpretation, although the purpose is shown as the higher level of abstraction. ANY DIFFERENCES?

PERHAPS BRING UP UP TO DATE ISSUE OF TRANSFER. - BUT PERHAPS LEAVE FOR CONCLUSION. NOTE ALSO THE BALANCE BETWEEN CT CONCRETE SKILLS AND TRANSFER IN (Kynigos and Grizioti, 2020)

This direction of the findings of this chapter, support findings on design patterns in other contexts as a middle ground between abstract CT concepts and more concrete techniques.

```
In order to be useful, patterns must present an abstraction of good practice
at a meaningful level of granularity. Formulations that are too abstract will
be impractical in real design use; those that are too specific will be
difficult to re-use in new scenarios.[@dearden_pattern_2006, p. 20]
```

Such an interpretation is adopted by Eriksson and colleagues using gameplay design patterns with young people (2019) which drew inspiration from the value of design patterns as a form of "intermediate-level concept" as advocated by fellow researchers as a way of sharing results of research (Barendregt *et al.*, 2018).

Turning to alignment with PBL research in digital making, as explored in the literature review a key challenge to project based learning is finding appropriate conceptual and practical frameworks to scaffold domain specific working practices to ameliorate the dialectical tension between engagement via tinkering and requirements to promote "principled understanding" (Barron *et al.*, 1998, p. 63). Waite and colleagues propose that abstraction should be explicitly taught even to younger pupils (Waite *et al.*, 2016). Waite and colleagues (Waite *et al.*, 2018) also propose the utility of awareness of LOA in terms of learners knowing at what level of abstraction they are operating, although the evidence is not representative and authors call for further research. As a challenge these findings, my own notes and reflections in appendix.learning map cautioned me against explicit teaching of abstraction and imposing shifts in perspectives for fear of interrupting the flow of participant making.

Despite this, in terms of participant experiences, it is clear from this chapters findings that learners *are* shifting between LOA without explicit in the process of the use of and contribution

to this learning design. Participant perspectives shift between the wider goal of making an engaging game for a real and imagined audience to a narrower goal of implementing a game design pattern and the technical actions and operations needed to complete it. This chapter has outlined the potential of GDP concepts, especially used within playtesting, to support these participant shifts in perspective. In this context I judged the imposition of an additional goal of understanding abstract curricular concepts as potentially counter productive to the overall aim of acculturation to game making tools and processes and developing an inclusive idioculture. I had a reluctance to shift learners away from the practical implementation of repeated game design patterns to focus on more abstract, de-contextualised conceptions of the knowledge. My concern also reflected potential disorientation and reduced experiences of agency from imposed shifts of focus rather than self-initiated ones. I also propose that to further research to would be needed to justify the utility take the time to interrupt flow to explicitly teach abstraction, and abstract CT concepts, apart from if understanding of such concepts are required by curriculum constraints.

On the other hand, my reflections support a proposition that there is utility for facilitators to be aware of the level of abstraction both that present in different elements of project learning design and that of the concepts that learners are working with. Different purposes and contexts of learning may require different interpretations, for example greater awareness of abstract concepts for reasons of assessment. According to (Barendregt *et al.*, 2018), in their identification of design patterns as intermediate level construct, identifying a suitable the level of abstraction is recommended to best support the goals of learners FIND REASON. Waite and colleagues (Waite *et al.*, 2018), also identify the intermediate design level of the LOA model as that suitable for supporting grounded abstraction in learner coding. In computing eduction pattern based instruction has benefits of making complex problems more modular and concrete (Muller, Haberman and Averbuch, 2004; Waite and Sentance, 2021). The evidence and examples surfaced in this paragraph support a view of GDPs, as an productive intermediate-level framework and contributes a series of grounded pedagogical instances via examples. It also proposes GDPs as potentially more productive and accessible framework in terms of pedagogical uses than that of computational thinking when making games.

Implications for practitioners

Returning to issues of inclusive approaches to pedagogy, the findings of this chapter resonates with Papert and Turkle's celebration of the process of concrete tinkering. They also suggest that it is potentially augmented, in terms of wider accessibility, with the use of design patterns as a germ cell concept. This approach is flexible and potentially responsive whilst still retaining a unifying pedagogical approach.

In many of the outlined uses of GDPs in chapter we can see processes at play that help bridge shifts in goals between activity systems and navigation between one sub-goal and the next. Any other analysis on the part of the participant can wait until the process becomes more familiar.

This approach is aligned to inductive research processes. Finding useful abstractions only after immersing ones-self in the detail. Previously abstract concepts or processes become concrete through familiarisation via direct use and indirect observations through community participation.

Table of varied GDP use with supporting evidence

The following table outlines the principle different uses of GDP in the data analysed. The next to last column lists example behaviour and notes vignettes or appendices where the behaviour can be found in situ with more detailed analysis. The final column indicates how many of the 12 in-depth session transcriptions the behaviour was found within.

GDP usage in game making	**Description (pitched to facilitators?)**	**Examples in data**
Cultural activity focus		
GDPs can encourage learners to design for others	Referring to the impact of implementing or changing GDPs can help scaffold the process of imagining the end user experiences.	In Vignette 2 Suzanna (p) and Olive (c) show awareness of designing for others. Present in 8 out of 12 coded sessions.
GDPs facilitate the use of wider funds of knowledge	Three dimensions of GDPs (mechanics, dynamics and aesthetics) can be used by participants to share and explore funds of knowledge and interest within the emerging learning community.	Dan's enthusiasm for coding in Vignette 7. Bertie's sharing of knowledge of game mechanics in Vignette 1 (part two) Present in x out of y coded sessions.
GDPs are propagated into the community through playtesting	The implementation of popular or novel GDPs by participant pairs or individuals is often spread through peer activity.	Toby's focus on level building becomes a subject of community activity Vignette 1 Present in x out of y coded sessions.
Guided Participation / Interpersonal focus		
GDPs as a framework for supporting resources and navigation	The restricted set of GDPs can serve as a framework for code examples and step-by-step tutorials. Presentation of documentation of GDP implementation as a menu or collection can facilitate flexibility of learner pathway and diversity of end design and without restricting access to resources beyond programme based support	Present in x out of y coded sessions.
GDPs used to scaffold ideation processes	GDP concepts can guide ideation processes in diverse ways including the use of visual code structures, and experimentation with core dynamics via rapid prototyping techniques.	Use of GDPs concepts by participants during ideation in Vignettes 2,3 & 7 Via visual prototyping in Vignettes 2 and 7 Discussion of GDP dynamics facilitated by template game structure in Vignettes …

		Present in x out of y coded sessions.
GDPs used to aid the process of division of labour	Participants can split work and resolve working blockages by allocating different GDPs or parts thereof. The use of GDP concepts as a lingua franca and the sometimes modular structure of GDPs help the division of labour as participants to divide work by working on different patterns or taking on project elements within patterns, sometimes building specialisms in the process.	Dividing work between pairs by pattern based on difficulty or aptitude. E.g. Vignette 3 Alternating between coding tasks by x in Vignette Parents shifting from supportive role to a more directive at times e.g. Vignette 7 Present in x out of y coded sessions.
The role of GDP used emerging technical processes	The implementation of GDPs can stimulate participants to adopt or to share technical processes.	Debugging in Vignettes 2, 1, Code patching and code snippets in Vignette 1 Present in x out of y coded sessions.
Personal appropriation:		
GDPs used in appropriation of personal knowledge: e.g. CT, design and systems concepts	Facilitators can use existing and planned GDPs to surface computational, design and systems concepts embedded in games. Facilitators should be cautious of overloading learners with shifting and competing goals.	Present in x out of y coded sessions.

It is important to recall the purpose and limitation of these figures outlined in Chapter 4, that this is not designed to claim efficacy of the learning design or to be representative of an average participant (which would require follow up studies of different design).

Conclusion

The observations of this chapter show the advantages of GDP as an intermediate, relatable design framework, occupying a the space between too concrete to be repeated and too theoretical to be grasped by novice game makers. The use of GDP concepts can create a tangible link between concrete player experience and the affordances of a guided creative process. Learners use of GDP as relatable and flexible constructs that facilitate communication, sustaining engagement, planning and division of labour. On a cultural plane, using GDPs can aid the propagation of technical and social processes game making practices within an emerging community of learners. GDPs served both researchers and participants by *providing a common language* to clarify first learner expression and researcher's analysis of gameplay experience. Gdps functioned as an *inspirational structured design tool* The utility of a patterns and their collection as a form *intermediate-level knowledge* by both researchers and participants is shown in part by in a

growing sense of mastery towards technical processes becoming second nature and the resulting ability to share them with family and other peer groups.

On a theoretical plane, this chapter has outlined the evolving use of GDPs as a meditational strategy to develop coding repertoires that are often shared. CHAT frames diverse use of shared resources and concepts, in this case GDPs, as meditational strategies which can involved into repertoires of practice (Lecusay, 2015; Gutiérrez *et al.*, 2019). These concepts and this framing of learning as participation in community repertoires, are explored in the next chapter in relation to participant agency using examples that extend beyond the use of GDPs to other cultural processes. While the development of agency has not been the primary focus of this chapter, the processes of community adaptation of GDP use can be directly linked to conceptions of transformative agency explored in Chapter 4. The next chapter explores cultural aspects in greater detail continuing to explore emergent agency in this research.

Barendregt, W. *et al.* (2018) 'Intermediate-level knowledge in child-computer interaction', in *Proceedings of the 17th ACM Conference on Interaction Design and Children*. *IDC '18: Interaction Design and Children*, Trondheim Norway: ACM, pp. 699–704. Available at: https://doi.org/10.1145/3202185.3205865.

Barron, B. (2006) 'Interest and Self-Sustained Learning as Catalysts of Development: A Learning Ecology Perspective', *Human Development*, 49(4), pp. 193–224. Available at: https://doi.org/10.1159/000094368.

Barron, B. *et al.* (2009) 'Parents as Learning Partners in the Development of Technological Fluency', *International Journal of Learning and Media*, 1(2), pp. 55–77. Available at: https://doi.org/10.1162/ijlm.2009.0021.

Barron, B. (2010) 'Conceptualizing and Tracing Learning Pathways over Time and Setting', *Teachers College Record: The Voice of Scholarship in Education*, 112(13), pp. 113–127. Available at: https://doi.org/10.1177/016146811011201308.

Barron, B.J.S. *et al.* (1998) 'Doing With Understanding: Lessons From Research on Problem- and Project-Based Learning', *Journal of the Learning Sciences*, 7(3–4), pp. 271–311. Available at: https://doi.org/10.1080/10508406.1998.9672056.

Basawapatna, A.R., Koh, K.H. and Repenning, A. (2010) 'Using scalable game design to teach computer science from middle school to graduate school', in *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education - ITiCSE '10*. *the fifteenth annual conference*, Bilkent, Ankara, Turkey: ACM Press, p. 224. Available at: https://doi.org/10.1145/1822090.1822154.

Bevan, B. *et al.* (2015) 'Learning Through STEM-Rich Tinkering: Findings From a Jointly Negotiated Research Project Taken Up in Practice', *Science Education*, 99(1), pp. 98–120. Available at: https://doi.org/10.1002/sce.21151.

Chesterman, M. (2023) 'Game Making and Coding Fluency in a Primary Computing Context', in

T. Keane and A.E. Fluck (eds) *Teaching Coding in K-12 Schools: Research and Application*. Cham: Springer International Publishing, pp. 171–187. Available at: https://doi.org/10.1007/978-3-031-21970-2_12.

Correa, T. *et al.* (2015) 'Brokering new technologies: The role of children in their parents' usage of the internet', *New Media & Society*, 17(4), pp. 483–500. Available at: https://doi.org/10.1177/1461444813506975.

Dietz, G. *et al.* (2021) 'Design Guidelines for Early Childhood Computer Science Education Tools', in C. Meinel and L. Leifer (eds) *Design Thinking Research : Interrogating the Doing*. Cham: Springer International Publishing (Understanding Innovation), pp. 291–306. Available at: https://doi.org/10.1007/978-3-030-62037-0_13.

Erickson, T. (2000) 'Lingua Francas for design: sacred places and pattern languages', in *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*. New York, NY, USA: Association for Computing Machinery (DIS '00), pp. 357–368. Available at: https://doi.org/10.1145/347642.347794.

Eriksson, E. *et al.* (2019) 'Using Gameplay Design Patterns with Children in the Redesign of a Collaborative Co-located Game', in *Proceedings of the 18th ACM International Conference on Interaction Design and Children*. *IDC '19: Interaction Design and Children*, Boise ID USA: ACM, pp. 15–25. Available at: https://doi.org/10.1145/3311927.3323155.

Esteban-Guitart, M. and Moll, L.C. (2014) 'Funds of Identity: A new concept based on the Funds of Knowledge approach', *Culture & Psychology*, 20(1), pp. 31–48. Available at: https://doi.org/10.1177/1354067X13515934.

Fasso, W. and Knight, B.A. (2020) 'Identity development in school makerspaces: intentional design', *International Journal of Technology and Design Education*, 30(2), pp. 275–294. Available at: https://doi.org/10.1007/s10798-019-09501-z.

Gee, J.P. (2003) 'What video games have to teach us about learning and literacy', *Computers in Entertainment*, 1(1), p. 20. Available at: https://doi.org/10.1145/950566.950595.

Gutiérrez, K.D. (2008) 'Developing a Sociocritical Literacy in the Third Space', *Reading Research Quarterly*, 43(2), pp. 148–164. Available at: https://doi.org/10.1598/RRQ.43.2.3.

Gutiérrez, K.D. *et al.* (2019) 'Learning as Movement in Social Design-Based Experiments: Play as a Leading Activity', *Human Development*, 62(1–2), pp. 66–82. Available at: https://doi.org/10.1159/000496239.

Gutiérrez, K.D. and Rogoff, B. (2003) 'Cultural Ways of Learning: Individual Traits or Repertoires of Practice', *Educational Researcher*, 32(5), pp. 19–25. Available at: https://doi.org/10.3102/0013189X032005019.

Guzdial, M. and Tew, A.E. (2006) 'Imagineering inauthentic legitimate peripheral participation:

an instructional design approach for motivating computing education', in *Proceedings of the second international workshop on Computing education research*. New York, NY, USA: Association for Computing Machinery (ICER '06), pp. 51–58. Available at: https://doi.org/10.1145/1151588.1151597.

Heineman, D.S. (2014) 'Public memory and gamer identity: Retrogaming as nostalgia', *Journal of Games Criticism*, 1(1), pp. 1–24.

Itō, M. *et al.* (eds) (2010) *Hanging out, messing around, and geeking out: kids living and learning with new media*. Cambridge, Mass.: MIT Press (The John D. and Catherine T. MacArthur Foundation series on digital media and learning).

Kafai, Y. and Burke, Q. (2015) 'Constructionist gaming: understanding the benefits of making games for learning', *Educational Psychologist*, 50(4), pp. 313–334. Available at: https://doi.org/10.1080/00461520.2015.1124022.

Kim, A.S. and Ko, A.J. (2017) 'A Pedagogical Analysis of Online Coding Tutorials', in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. New York, NY, USA: Association for Computing Machinery (SIGCSE '17), pp. 321–326. Available at: https://doi.org/10.1145/3017680.3017728.

Kynigos, C. and Grizioti, M. (2020) 'Modifying games with ChoiCo: Integrated affordances and engineered bugs for computational thinking', *British Journal of Educational Technology*, 51(6), pp. 2252–2267. Available at: https://doi.org/10.1111/bjet.12898.

Lave, J. (1991) 'Situating learning in communities of practice', *Perspectives on socially shared cognition*, 2, pp. 63–82.

Lave, J. and Wenger, E. (1991) *Situated learning: legitimate peripheral participation*. Cambridge [England] ; New York: Cambridge University Press (Learning in doing).

Lecusay, R. (2015) *Telementoring Physics: University-Community After-school Collaborations and the Mediation of the Formal/ Informal Divide*. Available at: https://doi.org/10.13140/RG.2.1.4131.5049.

Livingstone, S. *et al.* (2018) 'In the digital home, how do parents support their children and who supports them?'

Muller, O., Haberman, B. and Averbuch, H. (2004) '(An almost) pedagogical pattern for pattern-based problem-solving instruction', *ACM SIGCSE Bulletin*, 36(3), pp. 102–106. Available at: https://doi.org/10.1145/1026487.1008025.

Papert, S. (1980) *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books.

Papert, S. and Turkle, S. (1990) 'Epistemological Pluralism and the Revaluation of the Concrete', *Signs*, 16(1). Available at:

http://www.papert.org/articles/EpistemologicalPluralism.html (Accessed: 1 November 2017).

Petrich, M., Wilkinson, K. and Bevan, B. (2013) 'It looks like fun, but are they learning?', in *Design, Make, Play*. New York: Routledge, p. 21.

Redström, J. (2006) 'Towards user design? On the shift from object to user as the subject of design', *Design Studies*, 27(2), pp. 123–139. Available at: https://doi.org/10.1016/j.destud.2005.06.001.

Repenning, A. *et al.* (2015) 'Scalable Game Design: A Strategy to Bring Systemic Computer Science Education to Schools Through Game Design and Simulation Creation', *Trans. Comput. Educ.*, 15(2), p. 11:1-11:31. Available at: https://doi.org/10.1145/2700517.

Resnick, M. and Rusk, N. (2020) 'Coding at a crossroads', *Communications of the ACM*, 63(11), pp. 120–127. Available at: https://doi.org/10.1145/3375546.

Rogoff, B. (1995) 'Observing sociocultural activity on three planes: Participatory appropriation, guided participation, and apprenticeship', in *Sociocultural studies of mind*. New York, NY, US: Cambridge University Press (Learning in doing: Social, cognitive, and computational aspects), pp. 139–164.

Rogoff, B. (2003) *The cultural nature of human development*. Oxford [UK] ;a New York: Oxford University Press.

Rogoff, B. (2014) 'Learning by Observing and Pitching In to Family and Community Endeavors: An Orientation', *Human Development*, 57(2–3), pp. 69–81. Available at: https://doi.org/10.1159/000356757.

Rogoff, B. *et al.* (2016) 'The Organization of Informal Learning', *Review of Research in Education*, 40(1), pp. 356–401. Available at: https://doi.org/10.3102/0091732X16680994.

Roque, R., Lin, K. and Liuzzi, R. (2016) '"I'm Not Just a Mom": Parents Developing Multiple Roles in Creative Computing', in *Conference: International Conference of the Learning Sciences*. Available at: https://repository.isls.org//handle/1/177 (Accessed: 30 October 2018).

Sannino, A. (2011) 'Activity theory as an activist and interventionist theory', *Theory & Psychology*, 21(5), pp. 571–597. Available at: https://doi.org/10.1177/0959354311417485.

Sefton-Green, J. (2013) *Mapping digital makers: a review exploring everyday creativity, learning lives and the digital*. Available at: http://www.nominettrust.org.uk/ (Accessed: 18 December 2018).

Surma-aho, A. and Hölttä-Otto, K. (2022) 'Conceptualization and operationalization of empathy in design research', *Design Studies*, 78, p. 101075. Available at: https://doi.org/10.1016/j.destud.2021.101075.

Waite, J. *et al.* (2016) ‘Abstraction and common classroom activities’, in *Proceedings of the 11th Workshop in Primary and Secondary Computing Education on ZZZ - WiPSCE ’16*. *the 11th Workshop in Primary and Secondary Computing Education*, Münster, Germany: ACM Press, pp. 112–113. Available at: https://doi.org/10.1145/2978249.2978272.

Waite, J. and Sentance, S. (2021) ‘Teaching programming in schools: A review of approaches and strategies’.

Waite, J.L. *et al.* (2018) ‘Abstraction in action: K-5 teachers’ uses of levels of abstraction, particularly the design level, in teaching programming.’, *International Journal of Computer Science Education in Schools*, 2(1), pp. 14–40. Available at: https://doi.org/10.21585/ijcses.v2i1.23.