

Studying an Author-Oriented Approach to Procedural Content Generation through Participatory Design

Rui Craveirinha, Licinio Roque

► **To cite this version:**

Rui Craveirinha, Licinio Roque. Studying an Author-Oriented Approach to Procedural Content Generation through Participatory Design. 14th International Conference on Entertainment Computing (ICEC), Sep 2015, Trondheim, Norway. pp.383-390, 10.1007/978-3-319-24589-8_30 . hal-01758420

HAL Id: hal-01758420

<https://hal.inria.fr/hal-01758420>

Submitted on 4 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Studying an Author-Oriented Approach to Procedural Content Generation through Participatory Design

Rui Craveirinha and Licinio Roque

Universidade de Coimbra
rui.craveirinha@gmail.com
lir@dei.uc.pt

Abstract. The paper describes the design research process of a procedural content generation tool aimed at supporting creative game design processes. An author oriented approach to procedural content generation tools is used where these tools can be manipulated so as to let authors define the design space they want to explore and the design solution they wish to find, therefore maintaining their creative agenda intact. We present two Participatory Design exercises where game designers were tasked with creating a complete Interface Design for an implementation of this approach. Content Analysis from participants' discourse during these design exercises showed two important results. First, designers have trouble understanding how this procedural content generation works, and how to express their design problem within its conceptual framework. Second, subjects were averse to a pure optimization led approach to content generation and suggested the need for an exploratory phase, where content is created only to grasp the design landscape, without having to specifically define the desired solution.

Keywords: Creativity Support Tools, Participatory Design, Prototyping, Human-Centered Design

1 Introduction

Procedural Content Generation (PCG) tools promise enormous potential for game design activities, as they may actively empower designers to create artifacts more effectively and efficiently, as well as expand their ability to explore the creative space. However, PCG has been criticized for being confined to the goal of efficient production of assets (scenery, levels, maps), and for producing uninteresting, repetitive and/or unoriginal results [5]. And despite significant investment into researching new methods and algorithms to improve the quality of generated artifacts, there is little research done on how game designers can meaningfully interface with PCG tools in a way that potentiates their creative design process. This paper describes the two-stage design research process of a PCG tool interface focused on realizing an authors' design agenda. At this stage, the main objective was to clarify needs/requirements for an interface for creators

to interact with PCG algorithms. Towards that end, we involved game designers and researchers in a Participatory Design (PD) process, where they could design the user interface for the PCG approach. This design process served both for: a) designing an interface proposal from the perspective of its prospective users; b) providing a context to study how design practitioners would interact with this novel PCG approach. The outcome of this work was a series of findings related to how prospective users envision their interaction with a PCG-enabled game authoring system and, an interface model to build and user-study a prototype for such a system.

This paper goes on to describe the Participatory Design sessions, and the content analysis of the designers' discourse during these. Background section gives a brief overview of procedural content tools for games and the approach we are using, while the Methodology section outlines the PD exercises and how we analyzed results. Next we present the results from the PD sessions 1 and 2, with the content analysis of collected designers discourse. Finally, Discussion details our reactions on these sessions.

2 Background

In the field of game design, the past years have seen the rise of research into Experience-Driven Procedural Content Generation (EDPCG)[15], a family of computational methods that allow for automatic or semi-automatic generation of videogame content aimed at improving player-experience. The aim is to strive for methods that can dynamically create content that is tailored to specific models of player experience. While results from these EDPCG methods are very promising, from our research positioning, they are limited in two regards: a) they are confined to experimental contexts, and are now only taking their first steps into becoming fully realized in actual game design processes (as an example of an early attempt, see [14]); b) most importantly, several of the reviewed methods ([7][13]) are focused on generating content that improves player-experience models, i.e., optimizing user reported aspects such as 'interest', or 'fun'. These models are built by correlating features of the game's content with player reported evaluations of the experience, so that features from levels that players find 'fun' are used when generating levels that are meant to optimize 'fun'. Hence, these models overlook authorial definition and intentional exploration of the player-experience spectrum (at best, an author can choose which user preference to optimize). While there is much value in this model-based approach, we have proposed an alternative, author-centric approach [2]. It works by transforming the design problem into a search-space mapping problem. The search-space is defined by designers choosing certain key elements of the game's design (henceforth called artifact features) to which they will forfeit control within set boundaries. These are the elements which a search algorithm then proceeds to vary until it finds the set of artifact elements that consistently elicit the target player experience. The target player experience is defined as a quantitative goal, established by setting optimal values or boundaries, for player experience indicators as de-

finned by the game designers. Experience quality indicators are defined based on quantifiable aspects of logged user behavior, e.g. action rhythm, based on any count/frequency of specific actions within the game. General search algorithms can then try to solve the problem, iteratively generating artifacts, automatically measuring their mediated player experience with human players' participation, until a desirable solution is found. In this way, authors can use PCG to study/solve their design problems. In EDPCG terms, this is equivalent to letting authors design their own player-experience 'model'. For more on this approach, please refer to [2, 3].

3 Methodology

To design the interface, our strategy was to do use Participatory Design. Participatory design is an approach to design where end-users are given a lead role in the process [8]; given that the goal was to provide game designers with a tool that they would find useful for their creative processes, providing them a key role in the design of this tool seemed an obvious fit, as their needs and wants should guide the design, and they are apt designers in the first place. Out of all strands in PD techniques [1], because we were considering how to develop an artifact for supporting PCG-based game design, we wanted designers themselves to engage in Prototype Sessions, to envision the interaction with this new technology. The purpose of these prototype sessions is to provide an environment where the end-users can create, interact with, and discuss low-res prototypes of the intended artifact. The advantages in employing a Participatory Design method are two-fold: one, to quickly adjust this approach's prototype interface to game designers mindset and practices so that, as a tool, it can actually benefit their design process; second, to start a preliminary study on the nature and impact that using such a tool could have in a game design context, namely as to adjusting game design activities. Because our research concerns the study of how design processes can use PCG tools, it is only natural that a design technique be used as a means to obtain data, given that one of the forefront subjects of design research concerns design praxeology[4].

Besides the participants designing of the interface by means of paper prototyping, audio recording from the exercises was used to support revisions and discourse analysis, allowing further insight into users' perceptions. While the main participant's focus was the creation of an interface prototype, hints on how designers would appropriate this technology should surface, as this would be intimately related with the mindset of participants and the way they describe and analyze interactions with the interface in respect to their design problem. Hence, so as to inspect these Participatory Design sessions, we used Content Analysis applied to participants' discourse during the design exercise, by means of open coding by a subjective coder [6]. An initial pass of all audio recordings was done in search of key concepts, after which a new pass was done for the coding proper. A third pass was then done to correct any coding errors. Naturally, forms of qualitative analysis are susceptible to biased interpretations

and subjective manipulations [6]; in this case, given that only one author performed the analysis, one should temper the finality of any conclusions herein drawn. Since the purpose of this analysis was to complement users own design, by way of providing further insight into the rationale behind their expressed needs and specifications, seemed not to require further guarantees of objectivity. Furthermore, Relational Analysis was done by searching for occurrences of disparate concepts in time windows of one minute. Conclusions were drawn from the results of these analyses.

4 PD Session 1

The first PD session was to design an interface that would make the EDPCG approach usable to game designers. One designer and a computer engineer were developing a simple platform game, and their prototype was at the stage where this PCG approach could be used. They were told to draw the interface on paper whilst describing its workings orally. By grounding the exercise on their design process, we expected them to provide the best solutions for their work context. Coding for this session was divided into 3 major categories: a) data sources, b) UI elements and c) games design references. In terms of data, several gameplay experience measurements were mentioned, and so we subdivided counts in respect to the specific data type which they referred to. So, whenever a participant mentioned a gameplay metric, we counted one reference, the same for biometrics and subjectively determined qualities. Also, because the platform is focused on varying game features, every mention of these was counted. ‘UI’ related terms were divided into sub-categories: a) ‘platform configuration’, how data is inserted/edited/visualized in the application), b) ‘application structure’, how the program should be structured in terms of components and screen flow, c) how experience data should be processed visually when shown to users, and d) User Experience, references to usability and other user experience considerations). Game Design is the area we were most concerned with analyzing. We extracted 3 main categories in discourse, relating to the approach: a) ‘optimization’ of given experience goals, i.e. searching for the right set of features), b) ‘experimentation’ of different game variations, c) direct reference to ‘player experience’ qualities. We also registered problematic events: ‘terminological misconception’, registered whenever a participant used the wrong word for a given concept, the number of researcher directed ‘queries’ that were made, and expressions of doubt on either the working of the approach – ‘Doubts (Approach)’ – or its intended goals in terms of the creative design process – ‘Doubts (Goals)’.

Figure 1 shows the event count for each topic; we also accounted for a Proximity Analysis metric: the number of co-occurrence of concepts in one minute windows. A high number of ‘App Config’ (and App Config + App Structure pairs, 14 counts) counts means the session was dominated by a focus on screens for inserting/editing data into the PCG tools. This is further emphasized by high counts of two data concepts (‘Metrics’ and ‘Features’) and two very frequent co-occurrence pairs, ‘App Config’+‘Metrics’ (11 counts) and ‘App Con-

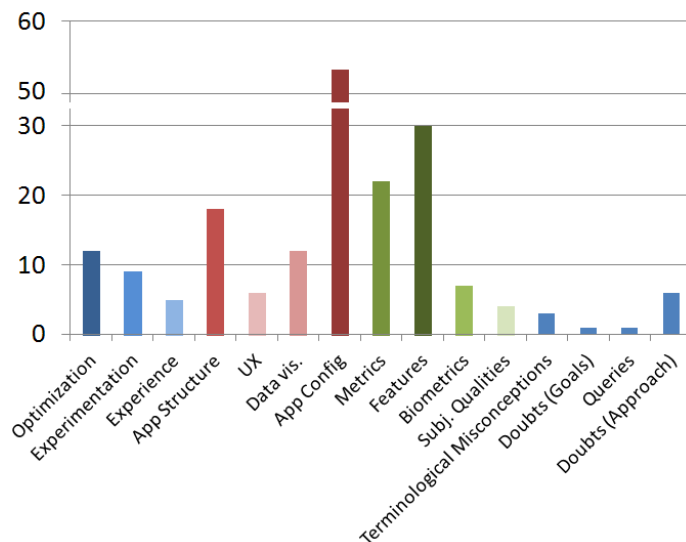


Fig. 1. Counting of references. Blue refers to Design topics, red to User Interface topics, green to Data topics and violet for problematic events.

fig'+‘Features’ (9 counts) where subjects focused discourse on how to get data into the application. We find a significant number of co-occurrences with pairs ‘App Config’+‘Terminological Misconceptions’ (6) and ‘Features’+‘Terminological Misconceptions’ (7); as while describing the process of inserting/editing a specific type of data subjects mistook one type of data with another. Though the main focus of the approach is on finding the artifact that fulfils a certain experience (e.g. ‘optimization’), the case for experimental testing of different cases was put forth, with no mention of target values. The number of events mentioning this approach was significant, and when queried if this experimentation should be a use case to be realized before optimization, one participant replied that “*yes, something along that line*”, an “*exploration*”. This means that this approach needed to be supplemented by a no target experimentation phase, where designers simply test out different variants of the same game. Also, participants struggled with finding appropriate values for target indicators. The pair ‘App Config’+‘Optimization’ (13 occurrences), shows that there was a sequential order in the mindset of the process, starting with editing the data and following with the optimization of these values, “*For all these features, we can establish [here] their boundaries and values, and in that case, we then either optimize the rhythm or...*”. But optimization was not, as would be expected beforehand, a considerable focus of the session, and was only mentioned 12 times, and only in abstract, with no target values associated. This suggests that : designers struggle to find meaningful values for certain indicators *a priori* without data. Also, the focus on data visualization, even though not dominant, speaks of the importance

of supporting visual data analysis, irrespective of whether or not optimization goals are met.

5 PD Session 2

The second session was to iterate on the first prototype. We realized a full PD workshop with 8 members of a game research laboratory, with backgrounds in game design, computer engineering, design (these include the members of the first session). In the second session, we counted both number of events and measured how much time was spent on certain discussion themes. The reason for this is that because of the large number of participants, discourse quickly lead to long winding discussions surrounding a single topic; counting these as singular events would result in a small number of events that could not represent the importance they took in the session. In terms of events we added coding for expressions implying difficulty in understanding some aspect of the platform, ‘Trouble Understanding’, and ‘Functional misconceptions’ regarding the platform working. Discussion themes include ‘UI’, which refers to all discussion of the interface design. ‘Composed Features’, a topic that revolved around the possibility of establishing complex game parameterizations, where each feature could vary in relation to a distinct one. ‘Preset-Features’, the proposal to add of a bundled set of default artifact features. Discussions that concerned how the game-artifact and the procedural platform should be integrated were coded with ‘Code-Platform Integration’. Also, ‘Data Mining’, referring to a proposal for the use data mining techniques for harvesting of meaningful indicator data. Discussions arising from doubts on how the platform should operate were tagged with ‘Platform operation’. Another discussion was referring to when to define Indicators, if before or after running a prototype test, ‘Indicator Definition’. And also, strings of discussion focused on attempting to pinpoint ‘Terminology’ or come up with accurate conceptual definitions to integrate parts of platform.

As expected given the goal of designing the interface, 55.6% of discussion time concerned the ‘UI’ topic and it was also the main focal point of all co-occurrences. More surprising is a non-negligible (9%) portion of time spent discussing how the platform operates, which is symptomatic of the approach’s concepts and functioning being hard to comprehend. This is confirmed by the high number of problematic events: 10 ‘Terminological Misconceptions’, 6 ‘Functional Misconceptions’, 7 ‘UI doubts’, 13 ‘Platform Doubts’ and 22 ‘Researcher Queries’. The almost 15% of time spent on discussing terminology and underlying concepts seems of considerable importance in this respect as they imply a great deal of conceptual confusion. Highlighting this effect, terminology discussions lead to Terminological Misconceptions 8 times. In terms of which concepts subjects struggled more in pinning down, they were either the types of data (features), i.e., *“aren’t these indicators [referring to features]?”*, or its procedural aspects . As one subject said: *“there is a conceptual base here that has yet to be defined, and it is very important”*, and *“the language is not completely defined”*. Nearly 10% was spent on determining how the approach should interoperate

with games' prototype code; the preoccupation in this discussion was how to make the process seamless. Four major design proposals were forwarded during the sessions: 'Complex Features', 'Pre-set Features', 'Data Mining' and 'Indicator'. Apart from the latter, time spent on them was mostly negligible and none translated into new requirements. However, in all these cases, participants betray a desire to have an easy to set up application. Finally, the 'Indicator' discussion (determining when indicators should be defined, whether before starting a test or afterwards) did not take up as much time as other topics (5%), it actually lead to a new specification. In the end of the conversation, there was an agreement that new indicators should be able to be defined both a priori and a posteriori. This reinforces the need for an explorative content generation approach.

6 Discussion

Based on the results of content analysis from the PD sessions, and the resulting paper interface, a number of crucial requirements for the PCG application can be discerned.

Procedural Content Generation needs a metaphor. Before this PCG method can be operationalized in a game design scenario its functioning needs to be easier to understand and apprehend by its users. There was a consistent struggle from subjects to understand the nature of this tool: from the arise of queries and doubts, to outright terminological misunderstandings, there is ample evidence of difficulty in grasping the platform's goals, concepts and (non-technical) working process. The excess of concepts – features, indicators, etc. – without an intuitive semantic framework, as well as the approach's complex mode of operation, presented a high barrier of access to users. Furthermore, a new approach to design requires a new design language. Our approach to solve this is to propose a simple metaphor to encompass this approach, and then make sure both the application's terms and logic are coherent with it, so that users should have less difficulties to understand how this PCG method works. Currently, we are testing one possible solution for this problem.

Exploration before Optimization. The other major difficulty that participants had was in how to define the game design problem and solution according to the metaphor of this procedural approach. It requires a reversal of the traditional game design flux – to decide on experiential qualities before game's material features – and it was never fully incorporated in subjects speech. In the first design session no values for experience indicators were ever discussed and in the second, little to no references were made on actual design cases and agendas that could be fulfilled with this approach. This puts forth the question of whether or not it is feasible for designers to reverse their mental processes in this way and use this tool as was foreseen. In both sessions a use-case was proposed for generating content without defining target experience indicators, so that users could better grasp the design landscape. Complementing this, the UI proposed by participants contains one window solely dedicated to presenting results from gameplay sessions, using plots and tables, and part of the discus-

sion, in both session 1 and 2 was in reference to this topic. Thus, an integral part of this PCG application needs to be focused on how explore gameplay experience data that can inform the design process. In this way, designers can have exploratory phases to map out the design space, before committing to a design agenda that they want to optimize.

References

1. Crabtree, A. Ethnography in participatory design. In *Proceedings of the 1998 Participatory Design Conference*, Computer Professionals for Social Responsibility (1998), 93–105.
2. Craveirinha, R., Santos, L., and Roque, L. An author-centric approach to procedural content generation. In *ACE 2013 the 10th International Conference in Advances in Computer Entertainment Technology* (2013).
3. Craveirinha, R., and Roque, L. Designing games with procedural content generation. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 2015)*, Seoul, South Korea, CHI '15, ACM.
4. Cross, N. Design Research: A Disciplined Conversation. *Design Issues* 15, 2 (1999), 5–10.
5. Hendriks, M., Meijer, S., Van Der Velden, J., and Iosup, A. Procedural content generation for games: A survey. *ACM Trans. Multimedia Comput. Commun. Appl.* 9, 1 (2013), 1:1–1:22.
6. Lazar, J., Feng, J., and Hochheiser, H. *Research Methods in Human-Computer Interaction*. Wiley, 2010.
7. Pedersen, C., Togelius, J., and Yannakakis, G. Modeling player experience for content creation. *Computational Intelligence and AI in Games, IEEE Transactions on* 2, 1 (march 2010), 54 –67.
8. Schuler, D., and Namioka, A., Eds. *Participatory Design: Principles and Practices*. L. Erlbaum Associates Inc., 1993.
9. Sears, A., and Jacko, J. *Human-Computer Interaction Development Process*. Human Factors and Ergonomics. Taylor & Francis, 2009.
10. Shaker, N., Yannakakis, G. N., and Togelius, J. Towards Automatic Personalized Content Generation for Platform Games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, AAAI Press (October 2010).
11. Smith, A. M., and Mateas, M. Variations forever: Flexibly generating rulesets from a sculptable design space of mini-games. In *IEEE Conference on Computational Intelligence and Games (CIG)* (2010).
12. Togelius, J., Nardi, R. D., and Lucas, S. M. Making racing fun through player modeling and track evolution. In *in Proceedings of the SAB'06 Workshop on Adaptive Approaches for Optimizing Player Satisfaction in Computer and Physical Games* (2006), 70.
13. Togelius, J., and Schmidhuber, J. An experiment in automatic game design. In *IEEE Symposium On Computational Intelligence and Games (CIG) 2008*, IEEE (2009), 111–118.
14. Yannakakis, G. N., Liapis, A., and Alexopoulos, C. Mixed-initiative co-creativity. In *Proceedings of the ACM Conference on Foundations of Digital Games* (2014).
15. Yannakakis, G. N., and Togelius, J. Experience-driven procedural content generation. *IEEE Transactions on Affective Computing* 99, PrePrints (2011).