Explore Data, Enjoy Yourself - KUbism, A Playful Approach to Data Exploration

Bruno Cardoso, Neil Cohn, Frederik Truyen, Koenraad Brosens

This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Explore Data, Enjoy Yourself - KUbism, a Playful Approach to Data Exploration

Bruno Cardoso[1], Neil Cohn[1], Frederik Truyen[2] and Koenraad Brosens[2]

[1] Department of Communication and Cognition, University of Tilburg, Tilburg, Netherlands
{bruno.cardoso, n.cohn}@tilburguniversity.edu
[2] Faculty of Arts, KU Leuven, Leuven, Belgium
{koen.brosens, fred.truyen}@kuleuven.be

Abstract. The increasing relevance of information in today's world has grown the challenge of accessible data exploration. Most existing approaches are primarily utilitarian, where the motivation for exploration remains mostly extrinsic, correlated to the value of the outcome. In light of our natural drives for exploration, we argue that data exploration can be made more rewarding and enjoyable via interfaces that leverage the intrinsic motivations of users. To that end, we present KUbism, a prototype emphasizing the hedonic value of data exploration through a playful interface built with elements of Minecraft gameplay. Implemented as a Terasology module, KUbism invites users to explore data in voxel worlds populated with data blocks. We conducted a user study (n=41) to validate our approach in both hedonic and utilitarian dimensions. Our results demonstrate that KUbism allows effective data exploration while arousing curiosity and joy, thereby reinforcing the extrinsic value of data exploration with intrinsic rewards.

Keywords: Data Exploration Interfaces, Playfulness, Gameplay, User Study

1   Introduction

In a world where information is progressively more digital, it is paramount to empower diverse people, with varying levels of technological expertise and divergent interests, to explore that information effectively [1]. We can find many different tools and approaches in the literature focusing on different aspects of data exploration, like steering the user towards important "areas" of datasets [2], or leveraging the visual capabilities of users to foster data comprehension [3] (see section 2.1 for examples). Notwithstanding the flexibility and potential of these approaches [4], they often have a considerable learning curve [5] and require a degree of domain knowledge that may pose difficulties for less experienced users [6–10]. Furthermore, most of these approaches remain utilitarian in nature, meaning the rewards of the exploration correlate primarily with the value of the outcomes and less with the experience of exploration itself. In light of how naturally people engage in intrinsically motivated exploratory behavior [10–13], we believe that additional value can be found in promoting the intrinsic rewards of the experience offered by data exploration tools, making them more enjoyable, playful and fulfilling of users' needs in a more comprehensive manner [13].

The approach of fluid interaction [14] is well aligned with this point of view. As proposed by Elmqvist et al. [14], fluid interfaces are designed to leverage the psychological concept of flow [15] – an elusive concept that, in short, refers to a mental state of total immersion in an activity. Placing special emphasis on the interaction with the data, fluid interfaces adhere to principles like the promotion of flow [15], support of direct manipulation [16] and minimization of the gulfs of action [17].

In this line of reasoning, we posit that the complex and intrinsically rewarding experience of gameplaying may prove to be an interesting approach to build truly fluid interfaces for data exploration that are playful, immersive, useful and enjoyable.

To the best of our knowledge this concept has not yet been the focus of significant research, but several reasons motivate its value. First, both data exploration [4] and gameplay can be motivated by curiosity [18, 19], an intrinsically motivating factor that suggests that a synergy may be found in combining the two activities. Second, playing fosters the cognitive processes involved in learning [20] – which often motivates data exploration. Third, videogames are known to create high levels of immersion and devotion, and they appeal to broad and diverse audiences [21, 22]. Therefore, datasets made available for exploration in interfaces that harness the playfulness of videogames have the potential of becoming more appealing to the general public. In other words, the self-rewarding nature of gameplay may draw people without special interest in given datasets to explore them, if only because of the enjoyable experience. Even people already interested in the data may find additional value in the intrinsically rewarding form of exploration we are proposing. Fourth, the approachability of videogames may empower people of varying data analysis experience to contribute to noteworthy findings. Indeed, these observations align with Elmqvist et al.'s argument that fluid interfaces for data visualization are, "in many respects, similar to computer games" [14].

Given the exploratory nature of our proposal, we developed KUbism, a prototype platform that aims to let users explore data via an interface built as an extension to the gameplay of Minecraft [23]. It was implemented as a Terasology [24] module, a Java game engine inspired by Minecraft. There are several reasons why the gameplay of Minecraft is well suited to our hypothesis of playful data exploration. First, as an open world, sandbox game, Minecraft's gameplay is hinged on exploration and free gameplay. It offers players expansive worlds to explore without the limitations of linear storylines or predefined goals and a considerable degree of creative freedom to experiment with gameplay [25, 26]. We expect that Minecraft's curiosity-arousing gameplay may predispose, or prime, users towards exploring the data. Second, the voxel worlds that make for one of the most distinctive features of Minecraft's gameplay can be considered tridimensional canvases on which all sorts of immersive data visualizations can be drawn with blocks (see Fig. 1 for simple examples, or [27] for a more complex one). Finally, we draw on another of Minecraft's most distinctive aspects, arbitrary structure building, as a metaphor for data manipulation. For example, by representing information elements (e.g., words) as blocks, structures built with such blocks can represent collections of those elements (e.g., bags-of-words or sentences depending on attributing meaning to the positioning of our hypothetical word blocks).

In line with these observations, KUbism represents information as a special type of block in the game world, and users are empowered to explore that information via in-

game tools and game mechanics. Because it is based on Minecraft's first-person gameplay, KUbism's interface is immersive by design: both users (through the avatars) and data (through the blocks) become actors in the game world. Data blocks interact with one another and share information when attached together. By allowing these blocks to be "mined", structures built with them acquire additional meaning – walls and houses become more than barriers or shelters: they become in-game datasets. Furthermore, players have a number of functional blocks at their disposal that allow them to query those datasets. By providing these tools, and allowing players to ask questions and get answers without leaving the game world, Minecraft's gameplay becomes a de facto interface between the data and the user.
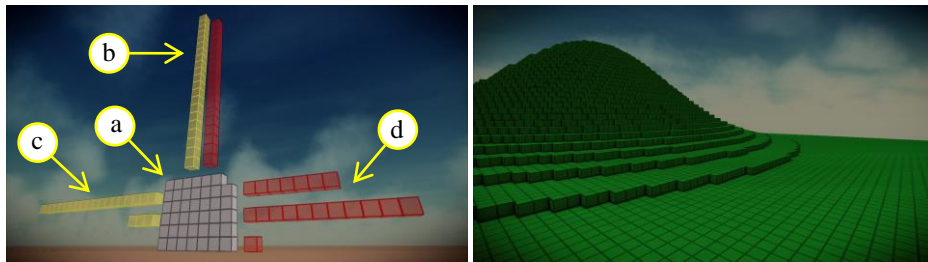

Fig. 1. Data representations in voxel worlds. Left: a representation of our study's demographics: (a) one gray block for each of our 41 participants; (b) a vertical bar chart for participant gender (yellow for females); (c and d) horizontal bar charts for participant area of expertise (yellow for females; top: humanities; middle: science and technology; bottom: middle school). Right: a world generated from a two-dimensional Gaussian function, exhibiting the characteristic bell curve.

We validated out approach by conducting a user study with 41 participants that allowed us to assess use patterns and subjective user experience in two complementary perspectives – as a utilitarian and a hedonic-motivation system. Prior work [28] has shown that KUbism could help art historians gain a sense of situational awareness while conducting their research. In contrast, the present work explores the human-computer interactions of KUbism, ensuring our findings can extrapolate relevantly to the HCI community. The contributions of the work presented herein are: (1) we describe our approach of gameplay-embedded data exploration fluid interfaces; (2) we present KUbism, our proof-of-concept platform; and (3) we present and analyze the results of our user study, drawing conclusions in favor of future research.

2 RELATED WORK

2.1 Data Exploration and Fluid Interaction

Given the growing importance of data in today's world, the research community has already proposed a number of promising approaches for data exploration, concentrating on different facets of this challenge. AIDE [2], for instance, is an interactive data exploration framework that steers users towards interesting areas of the data, while

predicting queries that would retrieve relevant information. Another interesting approach is Queriosity [7], an automated and personalized data exploration system that, instead of just retrieving data by request, aims to discover what is interesting in datasets and offer those insights to users. We can also find many approaches based in visualization, aiming to leverage users' proficiency for processing visual information. Perhaps the best known examples are interactive environments like Tableau [29], that enable users to design visualizations while replacing code-writing with resources such as visual grammars and data pipelines. While highly configurable, the process of specifying visualizations in these platforms requires a degree of knowledge and experience that may impact the experience of new users. Addressing that challenge, we can find proposals like SeeDB [30], a visualization recommendation engine that, given a subset of data to be studied, suggests the visualizations ranked the most useful. Inspired by the *small multiples* paradigm, Elzen and Wijk's [8] approach offers multiple simultaneous perspectives over a dataset based on smaller, easier to understand, visualizations.

Placing interaction design at the forefront, fluid interfaces [14] offer a different perspective over data visualization which is closely related to our work. While acknowledging the elusiveness of the concept of *flow*, Elmqvist et al. [14] proposed fluid interfaces for information visualization to be characterized by one or more of the following properties: (1) promotion of flow [15]; (2) support of direct manipulation [16]; and (3) minimization of the gulfs of action [17]. We can find many proposals in the literature that recognize the value of offering an experience that helps users stay immersed in the current activity and not be distracted by the user interface (i.e., *in the flow*) [1, 14]. For instance, ImAxes [31] is an interactive multidimensional visualization tool that relies on the arrangement of data axes to create collections of data visualizations in immersive, virtual spaces. Keshif [5] proposes an easy-to-use platform for exploring tabular data by summarizing it, aggregating records by value and visualizing aggregate aspects of the data according to the type of the data. Vizdom [1], the front end of the Northstar system [1], is a data exploration environment designed for pen-and-touch interfaces that relies on prompt feedback and approximate results to offer fast response times to users.

Though there are many similarities between KUbism and the approaches mentioned in this section – especially those that offer fluid experiences to users – perhaps the main distinction to be made lies in the utilitarian/hedonic continuum. Even though some of these examples do emphasize the intrinsic rewards offered by their systems, they are fundamentally utilitarian, with the value of the exploration effort positively correlated to the value of the retrieved information. By contrast, KUbism's data exploration experience is embedded in the gameplay of Minecraft and that makes our approach rather unique. It is as much a utilitarian-motivation system as a hedonic-motivation one. Users can use the elements of KUbism to query the dataset from within Minecraft worlds; they can just play classical Minecraft bearing no mind to the information contained in KUbism's blocks; or they can do both and use KUbism elements to add further meaning to elements of Minecraft's gameplay, like building structures that double as datasets (anecdotally, one of our study participants daydreamed of building a "temple with columns composed of blocks of data").

It is also interesting to note the conceptual analogy between KUbism and tangible interfaces, which have been studied to enhance the experience of interacting with digital

information like programming languages [32] or bar charts [33]. KUbism's data exploration interface actually builds on a sort of tangibility "by proxy" – both users (via the avatars) and the data (via the data blocks) co-exist and are agents in the same space (the game world), while abiding to the same set of rules (the game mechanics). From the user's perspective, avatars are arguably as tangible as the data: they share the same world. This is very much what happens with tangible interfaces, where digital information is given physical form (avatars) so as to share the (real) world with users.

2.2 Exploration in Videogames

Many videogames maintain exploration as a core gameplay activity [34]. For instance, exploration is one of No Man's Sky [35]'s gameplay pillars. No Man's Sky offers an open virtual universe including over 18 quintillion procedurally generated planets, including unique biomes. Allowing for multiplayer gameplay, the game informs players of unique discoveries they made. A companion website allows players to share interesting in-game discoveries with the player community. Similarly, Elite Dangerous [36] is a space-flight simulation game that allows players to explore a realistic 1:1 scale representation of the Milky Way. The game world contains some 400 billion-star systems and circa 150,000 are modeled after real astronomical data, complete with planets and moons that rotate and orbit in real time. How well this game appeals to players' drives for exploration is well illustrated by its (at the time of writing) 38,241-subscriber Reddit community [37] of explorers. In a different approach, Planetarium [38] is a "planet generator with creatures and secrets" that invites players to "discover" planets generated out of textual seeds (the planet names). Although it is possible to change parameters like planet temperature or land/water ratio and thereby change the aspect of the planets, the most original form of exploration in Planetarium lies in discovering planet names, an endeavor with its own community of name-sharing enthusiasts [39].

Although these examples illustrate how some videogames are using the allure of exploration and player's curiosity to keep them engaged, we found only one other example in the literature exploring the use of videogames to enhance the experience of exploring data: Onto-Frogger [40, 41]. Using elements of the classic game Frogger [42], Onto-Frogger focuses on communicating the emergent semantic structures of datasets to users, in a way that is both understandable and enjoyable. While both Onto-Frogger and KUbism offer users an interface to data exploration based on videogames, there are also some notable differences. For instance, Onto-Frogger is focused on exploring the structure of graph-based data repositories whereas our approach emphasizes the exploration of the data itself. Also, Onto-Frogger is, for all effects, a videogame by itself; a reimagination of Frogger's gameplay to allow players to learn about the structure of graph-based data repositories. KUbism, in turn, is an extension to Minecraft's gameplay. It does not require the removal or repurposing of any of the existing gameplay elements; instead, it extends Minecraft with an interface for interacting with data.

2.3 Minecraft in Research

Minecraft is a multiplayer, open world, sandbox game [43] whose default gameplay invites players to explore procedurally generated worlds of blocks that can be picked up and placed in order to build arbitrarily complex structures. It was inspired by Infiniminer [44], a 2009 independently created videogame that laid the foundations of the well-known Minecraft game mechanics, including the procedural generation of worlds composed of blocks that players can pick up and place. More than just a successful means of entertainment, Minecraft's gameplay lends a flexible medium that has been used in research, while keeping true to its intrinsically motivating playful nature [19, 45–47]. Examples come from fields as diverse as education [46, 48–50], health [51], art [52], productivity and planning [53], creative experimentation [54] and technology [55–57], among many others (see Rey et al.'s work for a more complete review [46]). Emphasizing the creative potential of this game, OPERAcraft [54] is a Minecraft-based arts and technology education platform that promotes live performances combining gaming, telematics, machinima and opera. As highlighted above, applications of Minecraft for purposes other than entertainment are diverse and plentiful. However, with the exception of MetaboCraft [27] where Minecraft structures are used to visualize metabolomics information, to our knowledge KUbism is the first application of a Minecraft-like game engine to data exploration.

3   KUBISM

3.1   Implementation and Architecture

Although the mechanics and overall look-and-feel of KUbism are akin to Minecraft's, we implemented our platform as an extension to Terasology [24]. The Terasology project started in 2011 as an LWJGL-based[1] Minecraft demo before later developers guided it to its current state. Terasology is an open source, Minecraft-like, voxel-based game engine implemented in Java. It is a community effort, developed and maintained by software developers, designers, game testers, graphic artists and musicians [58]. KUbism was implemented in Java as a Terasology module – an independent modular container for code and assets, including game systems, logic and content, with restricted access to the core Terasology engine.

Architecturally, the KUbism prototype was developed as a desktop application that communicates with a database server, acting as a client in a standard client-server architecture. This way, KUbism stands as an interface between the user and the data contained in a remote database (see section 4.1).

3.2   Data Source Format

KUbism assumes its supporting data to be modeled after the Entity-Relationship (ER) model [59], and therefore to be formatted as a graph of nodes (entities) and vertices

---

[1]   LightWeight Java Game Library (https://www.lwjgl.org)

between them (relations). Given the ER model's more "natural" view over data [59], the mapping of data elements (entities) to game elements (blocks) in KUbism becomes more straightforward, facilitating user understanding and reasoning about the underlying data model (see section 4.1). True to the ER model, KUbism entities and relations have attributes that characterize them.

3.3 A Gameplay-based Interface to Data Exploration

KUbism may be understood as a mapping between elements of the domain of data exploration to the domain of gameplay. As such, the entities and relationships of a data source must have a representation in the game world that allows users to interact with them, as well as the necessary tools to navigate the graph of data.

As "things that can be distinctly identified" [59], entities are represented in the game world as a special type of block, the *entity block* (Fig. 2, a), which can contain one or more entities of the same type.
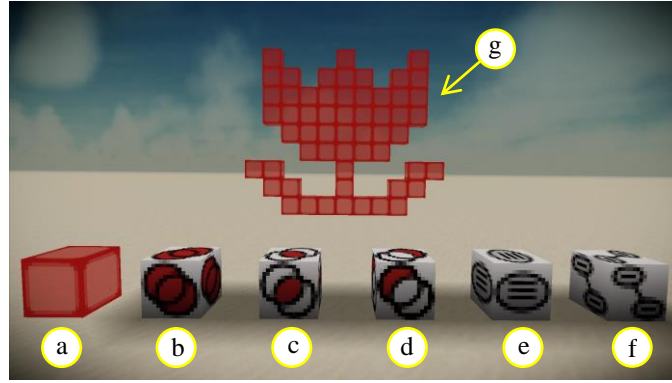

Fig. 2. KUbism's data exploration tools: (a) entity block; (b) set union block; (c) set intersection block; (d) set difference block; (e) entity filter block; (f) relation filter block; and (g) an entity structure, composed of many attached entity blocks sharing the same entities.

Users can interact with these blocks in the same way they would with other Minecraft blocks: they can be activated, picked up, carried, and placed anywhere in the world. Entities are shared between attached entity blocks, thereby allowing players to create in-game datasets with any shape or format by attaching entity blocks together. We designate groups of attached *entity blocks* sharing a common set of entities as *entity structures*, playing on both their in-game block structure aspect, and the role they play for data exploration as datasets (Fig. 2, g).

In turn, because relationships are "associations between entities" [59], their operationalization in KUbism is less straightforward. As the existence of relationships depends upon the existence of the related entities, their manifestation in the game world does not have the same independent and individual "presence" as that of entities. As such, players are only able to explore relationships by attaching a dedicated functional

block, the *relation filter block* (Fig. 2, f) to entity blocks (Fig. 2, a) or structures (Fig. 2, g) containing related entities.

We now present the full list of blocks that comprise KUbism's data exploration interface. For the sake of simplicity, in the following examples we will refer to a hypothetical and minimalist data source composed of (1) one type of entity, *person*, with a single property, *name*; and (2) one relationship, *parenthood*, which assigns the roles of *parent* and *child* to related entities. Finally, assume there are only three entities in the data source: "Lenny" and his parents "Mom" and "Dad".

Entity Block (Fig. 2, a). The red entity blocks act as containers for one or more entities of the same type. Activating an entity block allows the player to see the entities it contains – a popup window will display the entities in a list. Entities are shared, without repetitions, between entity blocks attached together (entity structures). Imagine two entity blocks, EB_1 and EB_2, such that $EB_1 = \{Lenny, Mom\}$ and $EB_2 = \{Lenny, Dad\}$. Attaching these two blocks together would result in a 2-block entity structure containing the entities of both blocks without repetitions, i.e., $\{Lenny, Mom, Dad\}$.

Set Union, **Intersection and Difference Blocks** (Fig. 2, b, c and d). These blocks implement the three basic operations of the algebra of sets, the union ($\cup$), intersection ($\cap$) and difference ($\setminus$). They operate virtually in the same way: once attached to different structures they will produce a new entity block containing the entities resulting from the application of the corresponding operation to the structures' entities. For EB_1 and EB_2, that would mean a new entity block containing $\{Lenny, Mom, Dad\}$ for the set union, or just $\{Lenny\}$ for the intersection. In turn, because the set difference operation is not commutative by convention, the set difference block is configurable – when activated, it will display a form requesting the user to indicate which of the attached structures is the "main" one. Depending respectively on which block EB_1 or EB_2 was set as the "main", that would mean a new entity block containing either $\{Mom\}$ or $\{Dad\}$.

Entity Filter Block (Fig. 2, e). This block operates like a filter, retrieving all entities of an attached structure that meet specified criteria. Users can express the criteria as predicates using a syntax similar to the where clause of SQL, in a window that pops up when the block is activated. E.g., to extract the entity named "Lenny" from the structure $\{Lenny, Mom, Dad\}$, we could attach an entity filter block to said structure and write an expression like person.name = 'Lenny' (where the "person.name" notation states the type and property of the entities we are interested in). Since there is only one matching entity, we would obtain a new entity block for $\{Lenny\}$.

Relation Filter Block (Fig. 2, f). This block works like the previous one but, instead of properties, the entities are filtered according to the role they play in relationships. If we wanted to find the parents of the entity Lenny, we could attach one of these blocks to a pair of structures, $S_1 = \{Lenny, Mom, Dad\}$ and $S_2 = \{Lenny\}$, configure it to focus on the *parenthood* relation between the entities of S_1 and S_2, set the former as *parents* and the latter as *children*, and thus obtain a new entity block containing $\{Mom, Dad\}$.

4 METHODS

4.1 Experiment Data

The data we used in our user tests comes from Project Cornelia's database [60], a data repository containing archival historical data about the 17th-century creative communities of Brussels and Antwerp [60]. Because the Cornelia database contains over 30 types of entities and many more relations, and since our study is focused on KUbism as an interface between users and datasets, we needed to minimize the potential inadvertent effects brought by factors such as data complexity. As such, to keep our model simple yet complete enough for the purposes of our study, we limited the data model to two types of entity, *actor* and *place*, and two relations, *parenthood* and *baptism*.

To provide some context, the actors in the Cornelia database correspond to people mentioned in historical records like painters, gold-beaters, stained-glass makers and their respective families, while the places stand for locations mentioned in said archives. Regarding the relationships between these entities, parenthood is a relation between two actors (*parent* and *child*), and baptism is a relation between an actor and the place where they were baptized (*baptized* and the *place of baptism*).

Communication between KUbism and the Cornelia database was implemented via the latter's API, so all queries built by our participants ran against a live database.

4.2 Experiment World

Because KUbism's interface is built as an extension to Terasology's default gameplay, our participants could engage in a range of playful activities instead of the data exploration required by our study, like exploring the world or mining. We thus needed to constrain our virtual world so as to keep participants focused in our experiment instead of just playing. Therefore, we devised a very simple world, featuring only a sky, and an endless flat ground (Fig. 3).
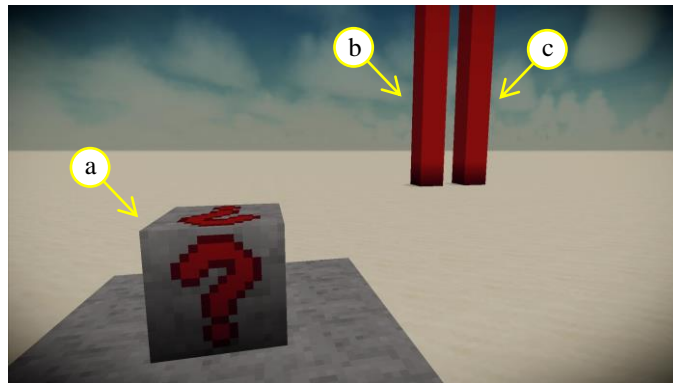

Fig. 3. The Experiment World. This was the setting of our study, featuring (a) the study block used to guide participants through the study tasks and (b and c) two column-shaped entity structures containing a block for each entity of type *place* (b) and *actor* (c).

We arranged the entities of our simple data model in two columns in this rather dull world, conveniently at hand for participants to interact with (Fig. 3, b). We also included a study block (Fig. 3, a) in the experiment world that guided participants through our study from within the game world. When participants activated this block, a popup window would be displayed with varying content: before starting the study, it presented participants with a demographic questionnaire; during the session, the block would either present users with the current task (Table 1) or verify their answers; finally, after the last task participants were asked to complete two questionnaires intended to assess their subjective experience with KUbism (see section 4.4). Note that this block is a resource of our experiment and not part of KUbism's data exploration interface.

4.3   Study Design

Our study sessions followed a task-based protocol requiring participants to perform 7 tasks involving different aspects of our interface (see Table 1).

Table 1. The 7 challenges of our task-based user study, along with possible solutions.

Task 1. Find all actors named "Marcus"
1. Attach an entity filter block to the structure of actors;
2. Activate the block with the condition actor.name = 'Marcus'.

Task 2. Find all actors with surname "Auwercx"
1. Attach an entity filter block to the structure of actors;
2. Activate the block with the condition actor.surname = 'Auwercx'.

Task 3. Find all actors with first name "Marcus" or surname "Auwercx"
1. Attach a union block to the blocks produced in tasks 1 and 2;
2. Activate the union block.

Task 4. Find all actors named "Marcus Auwercx"
1. Attach an intersection block to the blocks produced in tasks 1 and 2;
2. Activate the intersection block.

Task 5. Find all actors named "Marcus" without the "Auwercx" surname
1. Attach a difference block to the blocks produced in tasks 1 and 2;
2. Activate the difference block, setting the block of Task 1 as the main block.

Task 6. Who were the children of actor Marcus Auwercx?
1. Attach a relation filter block to the structure of actors;
2. Attach the entity block produced in Task 3 to the relation filter block;
3. Activate the block focusing on the parenthood relation between the entities in the Task 3 block (parents) and those in the structure of actors (children).

Task 7. Where were the children of actor Marcus Auwercx baptized?
1. Attach a relation filter block to the structure of places;
2. Attach the entity block produced in Task 6 to the relation filter block;
3. Activate block focusing on the baptism relation between the entities in the block of Task 6 (baptized) and those in the structure of places (places of baptism).

Since KUbism's data source is a graph (see section 3.2), we designed these tasks based on Lee et al.'s graph visualization taxonomy [61]. In that sense, all tasks involved the retrieval of values, tasks 1 and 2 required filtering entities by attribute value; tasks 3, 4

and 5 assessed the use of set operations; and tasks 6 and 7 focused on node adjacency, i.e., the retrieval of entities related to other entities.

Before starting the experiment, we asked participants to read a short document explaining the ideas behind KUbism (data exploration via gameplay embedded interfaces), the KUbism world (data is represented by blocks and can be explored via other blocks), and the controls. After reading the document, we invited participants to have a first-hand experience with KUbism, try the controls freely, and ask any questions. Although we did not impose restrictions on how long this first experience lasted, it generally took less than a couple of minutes. Because most of our participants did not report playing videogames frequently (see section 5.1), this phase served as a "crash course" that allowed them to gain familiarity with the controls. To start the experiment, we informed participants that they had to activate the study block (Fig. 3, a) whenever they felt ready. Upon activation, the study block presented participants with a questionnaire intended to gather demographic data, like date of birth, gender, education level, and gameplay frequency. Once the demographic questionnaire was answered, participants were presented with the first task of the study. Each of the seven tasks was formulated as a "challenge", a question about the entities or the relationships in the experiment's data model. All tasks followed the step sequence below:

1. Participants activated the study block;
2. The study block presented participants with a popup displaying the current challenge (see preceding list);
3. Participants proceeded to execute operations on data freely, following whatever course of action they felt appropriate, until obtaining an entity block with the entities, they thought to be the answer to the challenge;
4. Participants placed the response block on top of the study block; the response was verified automatically:
   a. If it was incorrect, participants were informed and asked to try again (step 3);
   b. Otherwise, participants were either informed of their success and presented with the next challenge (step 2), or the experiment would end if this was the final challenge.

In order to complement our understanding of the user experience, KUbism automatically kept a log of participant actions inside the game, like block-placing or activation, that we analyzed through process mining procedures.

4.4   Subjective Experience Evaluation

After completing all tasks we asked participants to answer two questionnaires to assess their overall user experience with the system: the well-known System Usability Scale (SUS), which is widely used and has become an industry standard [62] and the more recently proposed Hedonic-Motivation System Adoption Model (HMSAM) [18]. The SUS is a technology agnostic model for assessing system usability composed of 10 5-point Likert items. The SUS scale ranges from 0 to 100, with higher scores indicating generally more usable systems. In turn, the HMSAM is a more recent proposal aimed at evaluating hedonism-oriented systems such as videogames, and focuses on the

inherent factors influencing said systems. To that end, the HMSAM models user experience based on eight constructs (definitions in [18, 63]): *joy*; *control*; *focused immersion*; *temporal dissociation*; *curiosity*; *perceived ease of use*; *perceived usefulness*; and finally *behavioral intention to use*.

Given that KUbism aims to enhance data exploration via gameplay, we applied both the SUS and HMSAM questionnaires to assess both the utilitarian and the hedonic facets of KUbism, to draw a more complete picture of the overall user experience.

5 RESULTS

Our study took place in the second semester of 2018, in a quiet room at the university, with only the presence of the participant and an observer.

5.1 Demographics

Participants were recruited via direct invitation and snowball sampling, resulting from communications through our university's student and departmental mailing lists. Participation was voluntary and not compensated. This procedure yielded a total of 41 participants (19 male, 22 female; mean age: 29, range: 14 to 58). Participant's education range included middle school (1), undergraduates (15), Bachelor's (3), Master's (12) and Ph.D (10). Other than the middle-schooler participant, 26 reported studying Humanities (e.g., Cultural Studies and Art History) and 14 Science and Technology (e.g., Computer Science and Biology). On a 5-point scale (0 – "never" to 5 – "daily"), reported gameplay frequency was: 17 participants "never" played videogames, 11 played "rarely", 3 played "monthly", 6 did so "weekly" and 4 participants played "daily". With the exception of 3, none of the participants had previous contact with the data source we used in our study (see section 4.1). All participants gave their informed consent, and express consent was also given by the parents of our middle-school participant.

Participant Groups – Player vs. Non-Player. Given the relevance of videogames to our research, we divided the study participants in "Player" and "Non-Player' groups to control for the effect of gameplaying habits in the experience with our tool. Participants reporting "daily", "weekly" or "monthly" playing habits were considered "Players" ($n = 13$), and the rest, who "rarely" or "never" played videogames, were considered "Non-Players" ($n = 28$).

5.2 Task Time

All of our participants completed the 7 tasks of our study successfully, though time taken varied considerably. From the start of the first task to the completion of the final one, each testing session took an average of 886 seconds (00:14:46). The shortest session took 170 seconds (00:02:50) and the longest a total of 1992 seconds (00:33:12). Fig. 4 and Table 2 provide an overview of the time taken for each task, across participants, and for the Player and Non-Player groups.
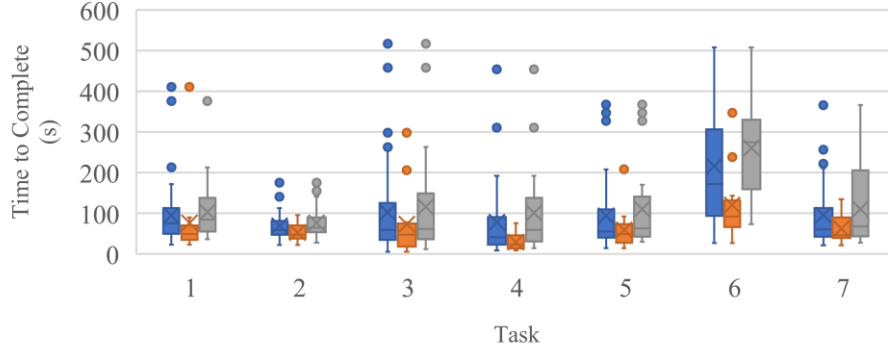
Fig. 4. Boxplots depicting the times for each of the 7 tasks of our study, for all participants (■), and the Player (■) and Non-Player (■) groups.

A Friedman test was run to understand the differences between the task times, across all participants (Table 2, first line). Pairwise comparisons were performed with a Bonferroni correction for multiple comparisons, yielding statistically significantly differences $\chi^2(6) = 79.251, p < .0005$. Post hoc analysis revealed those differences to be between tasks 1 and 4 ($p = 0.32$), and between the times of task 6 and all the other tasks' ($p < .0005$ for all comparisons; refer to Table 2 for the median times).

Table 2. Minimum, median and maximum task times, in seconds, for each of the 7 challenges, across participants (n=41) and for the Player (n=13) and Non-Player (n=28) groups. Table cell format: minimum-**median**-maximum.

Task	1	2	3	4	5	6	7
All	23-**76**-411	22-**59**-175	6-**59**-517	9-**41**-454	14-**55**-367	27-**172**-508	21-**61**-366
Player	23-**49**-411	22-**47**-95	6-**47**-298	9-**23**-76	14-**49**-208	27-**92**-347	21-**46**-135
Non-Player	36-**84**-376	28-**65**-175	12-**62**-517	14-**59**-454	30-**63**-367	73-**275**-508	28-**68**-366

To explore the effects of playing videogames frequently on task execution time, we ran a Mann-Whitney U test on the times of each task comparing the Player and Non-Player groups (Table 2, second and third lines). We found statistically significant differences in the times of task 1 (Player mean rank = 14.93 vs. Non-Player mean rank = 24.15; U = 274, z = 2.338, p = .019), task 4 (13.50 vs. 24.89; U = 294, z = 2.888, p = .004) and task 6 (13.57 vs. 24.85; U = 293, z = 2.859, p = .004).

5.3 Task Activity

Process mining procedures can bring insight into the way users interact with systems. To understand the first contact of our participants with KUbism we applied a process discovery technique to the event logs of Tasks 1 (Fig. 5, left). The relative simplicity of the process map, in which all actions occur a relatively similar number of times while progressing towards the final "Correct Answer" box, reveals that participants found

their way towards the solution in a rather straightforward manner. In turn, Task 6 required participants to explore database relationships for the first time and was the one that took the longest median time to complete across participants (Table 2). The process map for this task (Fig. 5, right) reveals that, instead of following roughly the same sequence of actions while searching for a solution, our participants performed a more diverse range of actions than for Task 1 (lighter colored boxes) in varied order (relatively intricate web of thin arrows). This is consistent with a trial-and-error approach suggesting further research is needed to facilitate the exploration of database relations.
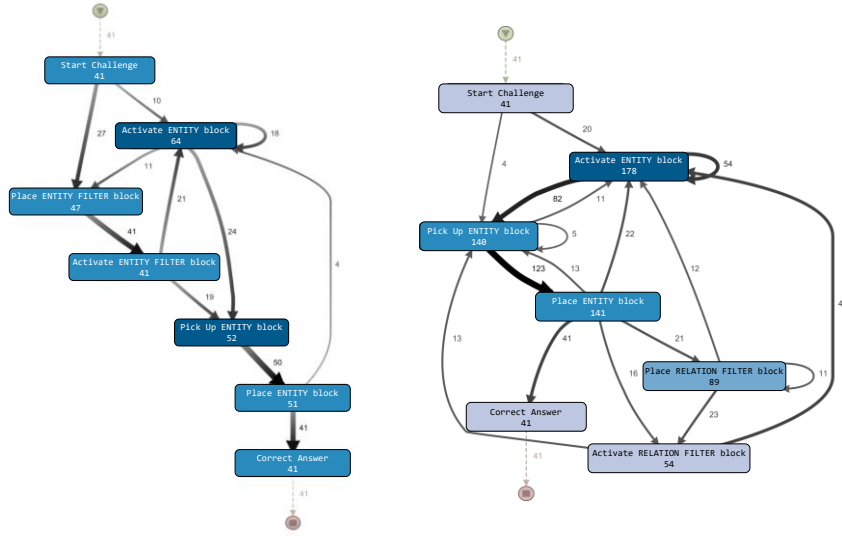

Fig. 5. Process Mining Map for the event logs of tasks 1 (left) and 6 (right), for all participants, showing the most frequent events and transitions. Transition frequency is indicated by the boxes' background darkness and thickness of the arrows.

It is also worth noting that there is a considerable number of consecutive repetitions of the "Activated Entity" action, meaning that participants repeatedly activated entity blocks to see the entities contained therein. Given that (1) the most straightforward solution for Task 6 involved reusing the entity block previously used to complete Task 3; (2) participants generally did not destroy the entity blocks as they progressed through the tasks; and (3) the only way to see the contents of entity blocks is by activating them, this finding likely reflects our participants searching for the entity block of Task 3.

Another insightful perspective over the many ways our participants used KUbism can be found in the way they left the world after the last task (Fig. 6). Some participants were methodical, keeping the blocks with the answers to previous tasks in a systematized way, easy to find and reuse in following tasks (Fig. 6, left). Others adopted a more ad hoc approach, focusing on solving the tasks one at a time (Fig. 6, right). Ultimately, all were able to complete the tasks, evidencing that our platform is flexible enough for participants to experiment with data and find their own way to explore the data.
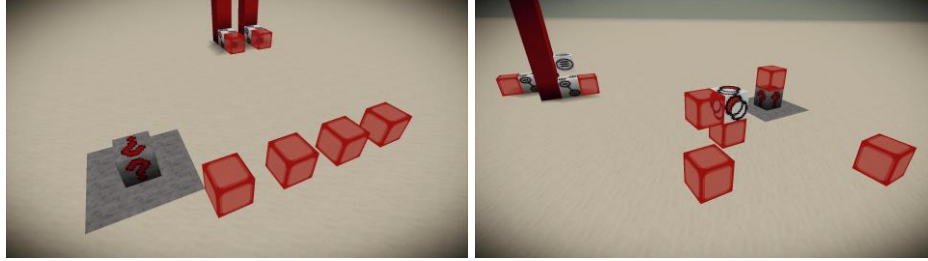
Fig. 6. Screenshots of the study world after two different participants completed the study. Some participants were methodical and organized (left), others found their answers in a more ad hoc approach (right). Ultimately, all participants completed the seven tasks of our study.

5.4 Subjective Experience Evaluation

As previously mentioned, we asked our participants to answer two experience evaluation questionnaires after the testing sessions, the SUS and the HMSAM (section 4.4). These results are illustrated of these two assessments are illustrated in Fig. 7, for all participants and the "Player" and "Non-Player" groups.
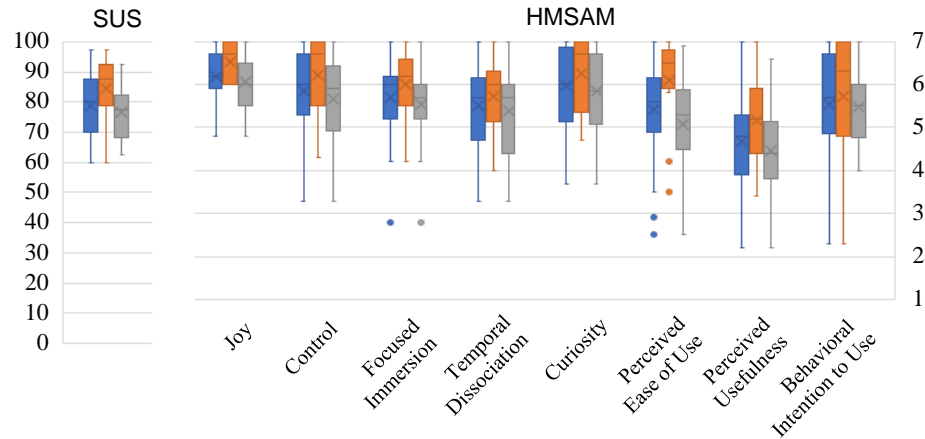

Fig. 7. Boxplots depicting the results of two subjective experience questionnaires for all participants (■), and the Player (■) and Non-Player (■) groups: (left) the System Usability Scale (SUS, 0-100 scale) and (right) the Hedonic-Motivation System Adoption Model (HMSAM, 8 dimensions, 1-7 scale).

The median SUS score was 80 (min: 60, max: 97.5) across participants, 87.5 (min: 60, max: 97.5) for the Player group and 77.5 (min: 62.5, max: 92.5) for the Non-Player group. We ran a Mann-Whitney U test on the results of the two groups revealing a statistically significant difference in their SUS scores (Player mean rank = 28 vs. Non-Player mean rank = 17.75; U = 273, z = 2.559, p = .011).

The HMSAM results may be found in Table 2, again for all participants and the Player and Non-Player groups. To understand the difference in the subjective

experience of the two groups, we ran a Mann-Whitney U test on the scores of each HMSAM construct. These tests revealed differences between the two groups in three of the eight dimensions: joy (27 vs. 18.21; $U = 260$, $z = 2.207$, $p = .027$), control (26.58 vs. 18.41.; $U = 254.5$, $z = 2.038$, $p = .042$) and perceived ease of use (29.81 vs. 16.91; $U = 296.5$, $z = 3.212$, $p = .001$).

6   DISCUSSION

In this work we propose KUbism, a data exploration interface embedded in Minecraft's gameplay. Through our user testing we showed that people can use our tool to successfully explore data while having an enjoyable experience. This supports the overall viability of our vision of data exploration interfaces embedded in gameplay.

All participants successfully completed all tasks of the study, indicating that people can effectively use KUbism to explore data and find answers. This observation becomes even more important in light of our participants' heterogeneity (see section 5.1), showing that our approach is accessible and flexible enough to allow people of diverse backgrounds and experience to explore data effectively. This is further corroborated by the process maps of tasks 1 and 6 (Fig. 5) that illustrate the many different, sometimes unique, ways our participants devised to complete the study tasks (Fig. 6).

The analysis of the task execution times for all participants also emphasizes the overall accessibility of our approach: with the exception of tasks 1 and 6, the median time of completion was approximately 1 minute for all tasks (Table 2, first row). It is also insightful that Task 6 took significantly longer to complete than all the other tasks, but the following Task 7, which also involved exploring relationships between entities, did not. With the exception of the relationship involved, both tasks were similar in all aspects. This suggests that there is a narrow learning curve involved in the exploration of relationships that is quickly overcome. In other words, exploring relationships becomes as straightforward as exploring entities after the first time.

Comparing task times between groups, the Player group had systematically lower median times for all tasks than the Non-Player group, but this was only statistically significant for tasks 1, 4 and 6. To help position this finding: (1) Task 1 was the first time participants interacted with KUbism without supervision, so it was a first contact scenario with the novelty that entails; (2) Task 4 was generally the quickest for everyone and, with the exception of the used block (*set intersection* instead of *set union*), the steps were the same as in the preceding Task 3 (Table 1); and finally, (3) the longest-taking Task 6 was the first time participants had to explore a relationship. In light of these observations, the increased familiarity of users with gameplaying experience seems to make them grasp new mechanics faster, but this difference tends to become less evident after the first contact with said mechanics.

Regarding the assessments of subjective experience, the overall scores for both the SUS and the HMSAM questionnaires are encouraging. The median SUS score of 80 obtained across participants placed KUbism's general usability between "good" and "excellent" [64] for our experiment, and this is also the case for the median SUS scores for the Player (87.5) and Non-Player groups (77.5). The statistical difference between

these groups, however, suggests that people with more frequent gaming habits tend to perceive KUbism's usability more positively.

In turn, all eight HMSAM dimensions also received high median scores across all participants, with the constructs of joy and curiosity receiving the highest scores (6.2 and 6, respectively). The joy construct refers to the hedonic quality of the interaction experience (see 4.4), and curiosity is a known motivation of exploratory behavior. We consider this observation to be a particularly positive result since one of our main goals is to enhance the intrinsically motivating factors of data exploration, making it not only a useful activity but also a positive and exciting experience.

Higher scores for HMSAM's joy, control and ease of use found for the Player group indicate that they generally felt more in control of the interaction and had a more pleasurable overall experience than Non-Players. This observation is in line with the higher SUS usability score also found for this group, and sheds some light on the rationale. Indeed, the relatively higher appeal that KUbism seems to have for people with more frequent gameplaying habits appears to be linked to factors related to *control* and *ease of use*. Given the influence that *control* seems to have in the overall user experience [18, 62, 63], it is reasonable to conclude that better gaming skills lead to a more pleasurable experience with our interface. Interestingly, the remaining constructs of HMSAM, namely *focused immersion*, *temporal dissociation*, *perceived usefulness*, *behavioral intention to use* and – more importantly for our study – *curiosity*, did not yield statistically significant differences between the two groups. This indicates that playing habits did not appear to influence these dimensions strongly – which were all rated highly. In light of the alignment between these dimensions and factors of flow [14] like *transformation of time*, *sense of control*, *intrinsically rewarding interaction*, *concentration* and *loss of self-consciousness*, we have reasons to believe that the experience of KUbism does indeed promote user flow, the main concept behind fluid interaction.

Though KUbism is a proof-of-concept, these encouraging results support that gameplay offers an interesting stage for data exploration. All participants were able to explore our study's data and enjoyed doing so, meaning that neither the utilitarian value of data exploration nor the hedonic rewards of gameplay were lost in our hybrid platform. That is the promise of our approach: an accessible, self-rewarding, fun, and compelling way to find valuable information – playful data in a literal sense.

7   CONCLUSIONS AND FUTURE WORK

Our work yielded evidence supporting the value of gameplay as a promising stage for building effective and playful interfaces for data exploration, leveraging both the utilitarian aspects of data exploration and the immersive, self-reinforcing nature of gameplay. All participants of our study successfully completed all study tasks while evaluating the experience positively. Overall, our approach scored highly in HMSAM's curiosity and joy. This is an encouraging finding as curiosity is a well-known motivator of exploratory behavior, and joy is a known predictor of behavioral intention to use [18]. Furthermore, in light of the diversity of our participants' backgrounds and the lack of familiarity most had with the study's dataset, the relatively low time taken to

complete the study tasks and the several different paths participants devised to find their answers are also important evidence towards the overall accessibility of our approach.

The novelty of KUbism warranted a conservative approach to the design of our study. Specifically, a controlled, task-based design allowed us to focus on each of our platform features one at a time. Following these initial results, a less restricted study should be considered, with participants being allowed to explore a data world without limitations or predefined goals. Such study is bound to yield a rich set of data and enable us to acquire a broader perspective on the way people experience KUbism's open-ended, sandbox approach to data exploration, which could include, for instance, affective dimensions [65].

Ultimately, KUbism is but a first approach towards data exploration intertwined with gameplay, and many possibilities for further research lie ahead. For instance, multi-player gameplay modes could offer great potential, especially if paired with persistent worlds. Rather than just exploring the results of procedurally generated worlds for fun, like what happens with many titles nowadays, we can imagine scenarios in education where communities of students explore relevant datasets together. If needed, minigames could be used to frame the exploration efforts – e.g., the task-based experiment of the study we present in this paper may be considered a minigame where players seek to answer questions in order to advance through the game.

On another perspective, we built the KUbism data exploration interface focusing mainly on structure building and block interactivity. However, another promise of our approach lies in the generation of worlds informed by the underlying data. As suggested in Fig. 1, there is clear potential for further research along this line. We can imagine whole worlds built in ways that reflect the dataset in some form or another, using pertinent aspects of the data as seeds for procedural generation. E.g., taking the term "seed" literally, a family's genealogical data could be used to generate trees of blocks with growth patterns hinting at important aspects of the data, like branch quantity and length to indicate the amount and lifespan of descendants. While observing such a tree would convey interesting information about a given family, a whole genealogy database could generate a forest for players to explore and learn about populations. On a different take, just like we used a Gaussian function to generate the terrain in Fig. 1, we could also use functions fitted to data points to create worlds with unique and meaningful features – e.g., wave-shaped mountains for skewed data distributions. Ultimately, the challenge of generating meaningful worlds, alluring yet useful, will be about finding the right balance between gameplay and data exploration, learning and fun.

References

1. Kraska, T.: Northstar: An interactive data science system. Proc. VLDB Endow. 11, 2150–2164 (2018). https://doi.org/10.14778/3229863.3240493.
2. Dimitriadou, K., Papaemmanouil, O., Diao, Y.: Explore-by-Example: An Automatic Query Steering Framework for Interactive Data Exploration. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data (2014). https://doi.org/10.1145/2588555.2610523.

3. Ferreira de Oliveira, M.C., Levkowitz, H.: From visual data exploration to visual data mining: A survey. IEEE Trans. Vis. Comput. Graph. (2003). https://doi.org/10.1109/TVCG.2003.1207445.

4. Yalçin, M.A., Elmqvist, N., Bederson, B.B.: Cognitive stages in visual data exploration. In: Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization. pp. 86–95 (2016). https://doi.org/10.1145/2993901.2993902.

5. Yalçin, M.A., Elmqvist, N., Bederson, B.B.: Keshif: Rapid and Expressive Tabular Data Exploration for Novices. IEEE Trans. Vis. Comput. Graph. (2018). https://doi.org/10.1109/TVCG.2017.2723393.

6. Mayr, E., Federico, P., Miksch, S., Schreder, G., Smuc, M., Windhager, F.: Visualization of Cultural Heritage Data for Casual Users. IEEE VIS Work. Vis. Digit. Humanit. (2016). https://doi.org/10.1109/TVCG.2016.2598468.

7. Wasay, A., Athanassoulis, M., Idreos, S.: Queriosity: Automated Data Exploration. Proc. - 2015 IEEE Int. Congr. Big Data, BigData Congr. 2015. 716–719 (2015). https://doi.org/10.1109/BigDataCongress.2015.116.

8. Van Den Elzen, S., Van Wijk, J.J.: Small multiples, large singles: A new approach for visual data exploration. Comput. Graph. Forum. (2013). https://doi.org/10.1111/cgf.12106.

9. Wongsuphasawat, K., Qu, Z., Moritz, D., Chang, R., Ouk, F., Anand, A., MacKinlay, J., Howe, B., Heer, J.: Voyager 2: Augmenting visual analysis with partial view specifications. In: Conference on Human Factors in Computing Systems - Proceedings (2017). https://doi.org/10.1145/3025453.3025768.

10. Idreos, S., Papaemmanouil, O., Chaudhuri, S.: Overview of data exploration techniques. In: Proceedings of the ACM SIGMOD International Conference on Management of Data (2015). https://doi.org/10.1145/2723372.2731084.

11. Maddi, S.R., Hoover, M., Kobasa, S.C.: Alienation and exploratory behavior. J. Pers. Soc. Psychol. (1982). https://doi.org/10.1037/0022-3514.42.5.884.

12. Loewenstein, G.: The Psychology of Curiosity: A Review and Reinterpretation. Psychol. Bull. (1994). https://doi.org/10.1037//0033-2909.116.1.75.

13. Korhonen, H., Montola, M., Arrasvuori, J.: Understanding Playful User Experience Through Digital Games. In: International Conference on Designing Pleasurable Products and Interfaces. pp. 274–285 (2009).

14. Elmqvist, N., Moere, A. Vande, Jetter, H.C., Cernea, D., Reiterer, H., Jankun-Kelly, T.J.: Fluid interaction for information visualization. Inf. Vis. 10, 327–340 (2011). https://doi.org/10.1177/1473871611413180.

15. Csikszentmihalyi, M.: Flow: The Psychology of Optimal Experience. Harper & Row (1990).

16. Hutchins, E.L., Hollan, J.D., Norman, D.A.: Direct Manipulation Interfaces. Human-Computer Interact. (1985). https://doi.org/10.1207/s15327051hci0104_2.

17. Norman, D.A., Draper, S.W.: User Centered System Design: New Perspectives on Human-Computer Interaction. Am. J. Psychol. 101, 148–151 (1988). https://doi.org/10.2307/1422802.

18. Lowry, P.B., Gaskin, J.E., Twyman, N.W., Hammer, B., Roberts, T.L.: Taking

"Fun and Games" Seriously: Proposing the Hedonic-Motivation System Adoption Model (HMSAM). J. Assoc. Inf. Syst. 14, 617–671 (2013). https://doi.org/10.17705/1jais.00347.

19. Mayes, D.K., Cotton, J.E.: Measuring Engagement in Video Games: A Questionnaire. Proc. Hum. Factors Ergon. Soc. Annu. Meet. 45, 692–696 (2012). https://doi.org/10.1177/154193120104500704.

20. Steinkuehler, C., Squire, K.: Videogames and learning. In: The Cambridge Handbook of the Learning Sciences, Second Edition (2014). https://doi.org/10.1017/CBO9781139519526.023.

21. Jegers, K.: Pervasive game flow: understanding player enjoyment in pervasive gaming. Comput. Entertain. (2007).

22. Entertainment Software Association: 2020 Essential Facts About the Video Game Industry, https://www.theesa.com/esa-research/2020-essential-facts-about-the-video-game-industry/, last accessed 2020/08/25.

23. Persson, M.: Minecraft. Game [Windows]. (18 November 2011). Mojang, Stockholm, Sweden, (2011).

24. Glatzel, B., Kireev, A., Praestholm, R.: Terasology. Game [Windows]. Open-source, community-based development and maintenance. (2011).

25. Min, W., Mott, B., Rowe, J., Liu, B., Lester, J.: Player goal recognition in open-world digital games with long short-term memory networks. In: IJCAI International Joint Conference on Artificial Intelligence (2016).

26. Tavinor, G.: The Art of Videogames. (2009). https://doi.org/10.1002/9781444310177.

27. Megalios, A., Daly, R., Burgess, K.: MetaboCraft: building a Minecraft plugin for metabolomics. Bioinformatics. (2018). https://doi.org/10.1093/bioinformatics/bty102.

28. Brosens, K., Cardoso, B., Truyen, F.: Slow Digital Art History And KUbism. In: The Routledge Companion to Digital Humanities and Art History (2020).

29. Stolte, C., Tang, D., Hanrahan, P.: Polaris: A system for query, analysis, and visualization of multidimensional relational databases. IEEE Trans. Vis. Comput. Graph. 8, 52–65 (2002). https://doi.org/10.1109/2945.981851.

30. Vartak, M., Rahman, S., Madden, S., Parameswaran, A., Polyzotis, N.: SeeDB: Efficient Data-Driven Visualization Recommendations to Support Visual Analytics. Proc. VLDB Endowment. Int. Conf. Very Large Data Bases. (2015). https://doi.org/10.14778/2831360.2831371.

31. Cordeil, M., Cunningham, A., Dwyer, T., Thomas, B.H., Marriott, K.: ImAxes: Immersive axes as embodied affordances for interactive multivariate data visualisation. UIST 2017 - Proc. 30th Annu. ACM Symp. User Interface Softw. Technol. 71–83 (2017). https://doi.org/10.1145/3126594.3126613.

32. Horn, M.S., Crouser, R.J., Bers, M.U.: Tangible interaction and learning: The case for a hybrid approach. Pers. Ubiquitous Comput. 16, 379–389 (2012). https://doi.org/10.1007/s00779-011-0404-2.

33. Taher, F., Hardy, J., Karnik, A., Weichel, C., Jansen, Y., Hornbæk, K., Alexander, J.: Exploring interactions with physically dynamic bar charts. In: Conference on Human Factors in Computing Systems - Proceedings (2015).

https://doi.org/10.1145/2702123.2702604.

34. Heaven, D.: The gamers who only want to explore virtual worlds, https://www.newscientist.com/article/dn27946-buttonmasher-the-gamers-who-only-want-to-explore-virtual-worlds/, last accessed 2019/03/15.

35. Hello Games: No Man's Sky. Game [Windows]. Hello Games, Guildford, England., (2016).

36. Frontier Developments: Elite Dangerous. Game [Windows]. (16 December 2014). Frontier Developments, Cambridge, England. (2014).

37. r/eliteexplorers, https://www.reddit.com/r/EliteExplorers, last accessed 2020/08/28.

38. Linssen, D.: Planetarium. Game [Windows]. (May 2015). (2015).

39. r/planetarium, https://www.reddit.com/r/planetarium/, last accessed 2019/03/14.

40. Kallergi, A., Verbeek, F.J.: Onto-Frogger: the Making Of. In: Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI '10). pp. 691–694. ACM New York, NY, USA (2010). https://doi.org/10.1145/1868914.1869006.

41. Kallergi, A., Verbeek, F.J.: Video games for collection exploration: Games for and out of data repositories. Proc. 14th Int. Acad. MindTrek Conf. Envisioning Futur. Media Environ. - MindTrek '10. 143–146 (2010). https://doi.org/10.1145/1930488.1930518.

42. Konami: Frogger. Game [Arcade]. (5 June 1981). Sega, Tokyo, Japan. (1981).

43. Ekaputra, G., Lim, C., Eng, K.I.: Minecraft : A Game as an Education and Scientific Learning Tool. In: Information Systems International Conference (2013).

44. Barth, Z.: Infiniminer. Game [Windows]. (April 2009). Zachtronics, Washington, USA. (2009).

45. Malone, T.W.: Toward a theory of intrinsically motivating instruction. Cogn. Sci. (1981). https://doi.org/10.1016/S0364-0213(81)80017-1.

46. Nebel, S., Schneider, S., Rey, G.D.: Mining Learning and Crafting Scientific Experiments: A Literature Review on the Use of Minecraft in Education and Research. J. Educ. Technol. Soc. 19, 355 (2016).

47. Shneiderman, B.: Designing for fun: How Can We Design User Interfaces to Be More Fun? interactions. (2004).

48. Schifter, C.C., Cipollone, M.: Constructivism vs Constructionism: Implications for Minecraft and Classroom Implementation. In: E-Learning Systems, Environments and Approaches (2015). https://doi.org/10.1007/978-3-319-05825-2_15.

49. Bebbington, S., Vellino, A.: Can playing Minecraft improve teenagers' information literacy? J. Inf. Lit. (2015). https://doi.org/10.11645/9.2.2029.

50. West, D.M., Bleiberg, J.: Education Technology Success Stories, (2013). https://doi.org/10.1097/MEG.0b013e32833bce39.

51. Villani, D., Carissoli, C., Triberti, S., Marchetti, A., Gilli, G., Riva, G.: Videogames for Emotion Regulation: A Systematic Review, (2018). https://doi.org/10.1089/g4h.2017.0108.

52. Styles, E.B.: Tate Worlds: Art and Artifacts Reimagined in Minecraft. Adv. Archaeol. Pract. (2016). https://doi.org/10.7183/2326-3768.4.3.410.

53. Saito, D., Takebayashi, A., Yamaura, T.: Minecraft-based preparatory training for software development project. In: IEEE International Professional Communication Conference (2015). https://doi.org/10.1109/IPCC.2014.7020393.

54. Bukvic, I., Cahoon, C., Wyatt, A., Cowden, T., Dredger, K.: OPERAcraft: Blurring the Lines between Real and Virtual. In: Proceedings ICMC SMC 2014 (2014).

55. Johnson, M., Hofmann, K., Hutton, T., Bignell, D.: The malmo platform for artificial intelligence experimentation. In: IJCAI International Joint Conference on Artificial Intelligence (2016).

56. Alstad, T., Duncan, J.R., Detlor, S., French, B., Caswell, H., Ouimet, Z., Khmelevsky, Y., Hains, G., Bartlett, R., Needham, A.: Minecraft computer game performance analysis and network traffic emulation by a custom bot. In: Proceedings of the 2015 Science and Information Conference, SAI 2015 (2015). https://doi.org/10.1109/SAI.2015.7237149.

57. Bayliss, J.D.: Teaching game AI through Minecraft mods. In: 4th International IEEE Consumer Electronic Society - Games Innovation Conference, IGiC 2012 (2012). https://doi.org/10.1109/IGIC.2012.6329841.

58. What is Terasology, https://github.com/MovingBlocks/Terasology/wiki/What-is-Terasology, last accessed 2019/03/18.

59. Chen, P.P.-S.: The Entity-relationship Model: Toward a Unified View of Data. ACM Trans. Database Syst. 1, 9–36 (1976). https://doi.org/10.1145/320434.320440.

60. Project Cornelia, https://projectcornelia.be/, last accessed 2021/06/10.

61. Lee, B., Plaisant, C., Parr, C.S., Fekete, J.-D., Henry, N.: Task taxonomy for graph visualization. In: Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization. pp. 1–5. Association for Computing Machinery (2006). https://doi.org/10.1145/1168149.1168168.

62. Brooke, J.: SUS - A quick and dirty usability scale. Usability Eval. Ind. 189, 4–7 (1996).

63. Agarwal, R., Karahanna, E.: Time Flies When You're Having Fun: Cognitive Absorption and Beliefs about Information Technology Usage. MIS Q. (2006). https://doi.org/10.2307/3250951.

64. Bangor, A., Kortum, P., Miller, J.: Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. J. Usability Stud. (2009). https://doi.org/66.39.39.113.

65. Cardoso, B., Santos, O., Romão, T.: On Sounder Ground: CAAT, a Viable Widget for Affective Reaction Assessment. In: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15). pp. 501–510. ACM, New York, NY, USA (2015). https://doi.org/10.1145/2807442.2807465.