Chapter 8
Interview with Nicholas Francis and Thomas Hagen from Unity Technologies

Alessandro Canossa

Unity 3D is an integrated authoring tool for creating 3D interactive environments, such as games and real-time 3D visualizations. Unity consists of both an editor, for designing content, and a game engine, for executing the final product. Over the recent years since its inception in 2006, Unity was given several awards, including Wall Street Journal 2010 Technology Innovation Award and Gamasutra 2009 Top 5 Game Companies. Additionally, the company was held partly responsible for the democratization of the game development process.

This is an interview with two key figures behind the Unity engine. Nicholas Francis – Chief Creative Officer at Unity Technologies, responsible for product and feature design. Nicholas is one of the founders of Unity Technologies and has been working in the company for more than 10 years, before that he was just a dreamer conjuring ways to make game development a smooth, fun process. He has been credited as the grandfather of the Asset Store, just an example of Unity's approach to problem solving: building tech so that people can sell stuff to each other, optimizing small teams' resources and skills.

Thomas Hagen – He worked at EIDOS and Square Enix as Online Development Manager to build their metrics infrastructure for a number of years. He started writing software in the early 1980s at age 11; today he is an independent contractor and works with Unity Technologies to develop analytic systems.

Q: How do you think telemetry can be useful for the industry?
Nicholas: We at Unity Technologies believe that telemetry and metrics are going to be crucial for making better, more fun games and that is the core proposition of our business model. We are not interested in revenue optimization; we just want to help people make fun games.

A. Canossa, Ph.D. (✉)
College of Arts, Media and Design, Northeastern University,
Boston, MA, USA

Center for Computer Games Research, IT University of Copenhagen,
Copenhagen, Denmark
e-mail: a.canossa@neu.edu

Most of the time what we see is developers trying to minmax, micro-optimize and balance their designs, but when addressing metrics as a tool to achieve that goal, they acknowledge the relevance of the tool and at the same time their eyes wander, looking for someone else to talk to. I believe it is a human trait: when we don't know exactly how to do something, we will do anything else, procrastinating the blurry task indefinitely. Mobile game developers, unless they have a past as web developers, they have no idea how to set up a server, and that is enough to begin the cycle of procrastination. They would do the most amazing things in order to avoid doing something they don't perfectly understand. And that applies even more when open-ended thinking with no concrete question is involved, such as in exploratory analyses (see Chap. 12).

Q: What can Unity do to help in that regard?
Nicholas: I believe that most of the time some very simple analytics will get 90% of the job done. We want to deliver basic solutions that allow anybody, with no previous knowledge, to include simple analytics from the moment they begin authoring a game. Due to the fact that we provide a somewhat controlled environment, there is a ton of tools that we can provide out of the box, for example, tracking performance metrics such as frame rates, slowdowns, etc. We can also provide end-to-end hosting services integrated in the editor.

Another aspect is the presentation: in order to win over the initial hurdle and begin using analytics tools, we want to invest considerable resources in visualizing the data to provide an environment where developers can feel free to play around with their data. The key concepts are: simplicity of out-of-the-box analyses, accessibility of tools, and playfulness of visualizations. Complexity can be added via APIs where advanced developers can expand the basic solution. We tried it on our own skin: playing with analytics is addictive; we believe that developers will easily get hooked and will soon start toying around with the API.

Thomas: I expect the Unity Asset Store to be soon crowded with analytics extensions developed with the API.

Q: Do you think that a lack of clarity on potential benefits of analytics is also part of the problem for developers?
Nicholas: Not really; every developer knows that Zynga is racking in cash and everybody suspects that it has something to do with analytics. So there is a basic desire and positive bias towards it, but it is difficult to take it further than that. It is our intent to help developers turn raw metrics data into meaningful insights and actionable information.

Thomas: Game directors, CEOs and studio heads are very aware of the potential of the tool, they are even aware of the different costs and benefits for each type of stakeholder (see Chap. 3) The more business oriented you are, the more you require data to make decisions. However, if you come from the creative side of development, you are taught to trust in your vision. Creative directors and designers are more resistant to the idea of using metrics; they often feel challenged in their creative vision. It seems easier to convince top management, but at the same time it is impossible to deploy telemetry solutions without having the whole team on board.

A viable solution could be to remove any entry barrier for all levels of development: if for example level designers find analytic tools enabled and ready out of

the box, they will have no reason not to experiment with the tools, and they may eventually become advocates for the practice and help convincing colleagues. It is our vision to turn every step of game development into a game itself, and that applies to analytics as well.

Q: What are the metrics you plan to include initially as built-in features in the editor?

Nicholas: We expect all the variables currently tracked in the Profiler (CPU usage, rendering, memory, audio, animation, etc.) to be available as metrics for the developer. These include memory usage, CPU and GPU usage, streaming in and out of memory, etc. In addition to these basic performance processes, we already track automatically many variables from the cameras and the viewports, such as position relative to world coordinates, when players log in and log out, when players start a play session, when and which levels are loaded, when players stop a play session, and how long time passes between play sessions.

Thomas: We have two main target users: designers and coders. We want to provide a solution that caters to both: providing player metrics for the first group and performance metrics for the second group.

Q: Ideally, when do you see analytics becoming a part of the development process?

Nicholas: As it stands now usually developers devote resources to metrics collection and analytics very late in production, mostly at the beta stage. This is usually due to resource shortages. It is indeed difficult to put in place a system that tracks variables when there is no game at all. However, the appearance of metrics tracking that late in the process is a missed opportunity for both monitoring the evolution of software performance and stability, and charting the change of gameplay dynamics through iterations. We plan to change that by enabling out-of-the-box automatic tracking of several basic variables.

Thomas: We plan to develop solutions for rapid creation of heatmaps based on any game variable, both as traditional 2D visualization and as 3D particle clouds, integrated natively in the editor. Furthermore, we want to have a filter inspector to perform simple Boolean operations on sets of features to be visualized and compose basic spatial queries inspired by Geographic Information Systems; a sort of complex query compositor.

We might also go the Apple way and offer a very limited number of basic queries that can be applied in most cases, and later in time expand the offering with additional queries. This fits the development philosophy at Unity: ship few features very polished and expand constantly. Approaching the problem in this manner softens a potentially daunting learning curve, minimizes risks and facilitates early adoption by beginners.

Q: How many of your customers, which are relatively small in size, have user research departments? How will your tool benefit them?

Thomas: Only the most experienced game director understands and values the benefits of user research to the point of insisting to have a specialist on board. Otherwise, most game directors tend to enjoy being left on their own taking drastic, risky decisions about their games. A second factor is the size of the development teams: only medium to large teams have enough stakeholders, who may be interested

in keeping an eye on player behavior; in smaller teams, say less than ten people, only the lead designer pays attention to player experience, and often functions as a user researcher. No matter what, even in the best case scenario, user researchers have to agree with the game director on the variables to track. Usually, it is the game director who gives the final authorization to programmers to enable hooks in the code to tracking certain variables. Right now most of our customers are studios with a staff of around five people, and thus do not have dedicated analysts, but I see the situation changing soon: a streamlined environment instantly empowers even one-man teams to adopt analytics practices.

Nicholas: I believe that our tool, ideally, would empower user researchers to open the game in the editor, define variables and start tracking independently from the wishes of the game director. User researchers no longer have to depend on the game director greenlighting the tracking of specific variables; user researchers can decide what to track and implement it by themselves, at least during development; what to track from a finished, shipped game is still a big decision that should be agreed upon by all stakeholders involved.

Until now, due to the extra burden put on programmers and the fact that user research departments do not usually include technical people, it has not been possible for user researchers to have that kind of independence, but if our tool is easy enough to use, that scenario could change allowing user researchers, who have some or no technical expertise, to constantly tweak variables and metrics almost at a whim. This will also allow user researchers to seamlessly setup AB testing (see Chap. 4) both before and after release. They may also setup dynamic metrics, maybe not in the first iteration of the tool. However, it is definitely our dream to enable users to define and change metrics even after the game is deployed.

Q: Why do you think metrics collection and analysis seems to be in the Zeitgeist?

Thomas: Telemetry is currently perceived as a difference-making element. Due to the extreme competitiveness of the game industry, it is becoming more and more important. Unity has democratized games' development: the tools are affordable and the necessary skills fairly easy to obtain, so much so that games, like *Angry Birds* (Chillingo/Rovio Entertainment, 2009), can be created by a small team of students in a very short time. Creating successful games is no longer just a technical achievement, it becomes vital to know how to tweak and polish your creation to make it different from your competitors. And that's Zynga's real strength; most of the employees are engineers, some are from marketing, but almost no designers. They are not doing anything revolutionary in terms of design. However, whatever they are doing, they are doing it really well and they know precisely what their players are doing at any given time.

Nicholas: Another key factor is how much people are getting used to the concept of analytics. Taking any introductory web design class will expose you to the basic principles of Google Analytics, something simple enough for anybody to grasp and pervasive enough. People are getting comfortable with the concept of hits, and care enough about that to check more than once a day. Furthermore in pretty much every aspect of the business, it is necessary to justify spending and decisions need to be

backed up by facts. This brings analytics to the spotlight as it is a fast and cheap way to obtain facts. I will not be surprised if, within 5 years, game data analytics becomes an integral part of game education curricula.

Q: What are the main benefits of telemetry and analytics for the game industry?

Thomas: In order to be competitive, it is crucial to know how players are using your game. Obviously this fact translates differently for each of the stakeholders. Designers benefit the most during development, while management and corporate personnel are more interested in data acquired after release. For example, at Square Enix metrics are used to assess and spot piracy – *Tomb Raider: Underworld* (Eidos Interactive, 2008) was played on the Xbox by unauthorized accounts even before the official release. Using metrics, we were able to identify these users and enquired about how they received such early access. Another extremely important layer that can benefit greatly from metrics is the community of players. As a developer it is important to reward players that generate precious data for our production, something beyond leaderboards, like replays, smart match-making or gamer lifecycle information.

Q: Any suggestions to green developers that are interested in including telemetry in their games?

Nicholas: Use Google Analytics: it's completely free, rather simple to use, and, after only 1 day writing code, the rewards are immense. Additionally, the practice is invaluable because doing that can train anybody to the frame of mind necessary allowing them to start tracking data for games. It's a great testing ground before deploying more complex solutions. We are actually using it to track how our customers use the editor to develop their games. For example, we know that in 2011, 30 man-years were spent by our developers waiting for the engine to bake lightmaps. If we had a faster solution, we could have saved a person's professional life.

Thomas: Even developers adopting game telemetry solutions seem to make use of it mostly during production; as soon as the game is shipped they tend to lose interest. However, one thing that is missing from this practice and is sorely needed, is to continue analyzing player behavior data to verify if the design decisions taken during development and based on gut feelings turned out to be accurate guesses or complete misses. This might not bring immediate benefit to the game in question, but it can provide lessons for designers to mature their practice and craft.

Q: Do you think it is possible to mature the current practices in place in the industry? If so, how?

Thomas: I think that the industry at large has not yet leveraged telemetry to its full potential. In the projects I have been involved in, metrics were never gathered from within the editor, so the work process of level designers, for example, has not benefited as much as it could have. The Unreal Development Kit has few features already, and it's a great start, but it's not enough. Ideally anybody could decide to track anything from any game, and be able to do it without involving any programmers, if the tools are developed and integrated well enough within the editor. Relating to what I said before about making better use of analytic practices post-launch of a game: another aspect to be improved could be to harness the passion and commitment demonstrated

by communities of players to investigate how people play the game either by administering questionnaires or employing machine learning, as it was done with the game *Tomb Raider: Underworld* (Eidos Interactive, 2008) where a team of researcher from IT University of Copenhagen classified the behavior of 1,365 players utilizing emergent self-organizing neural networks.[1] It would be precious information for the marketing and management departments when making decisions about future projects.

Nicholas: Recently we tried to hire a number of professional analysts from marketing, they may be proficient with SPSS, but with Google Analytics they can barely hit "print screen". It is hard to qualify a new set of experts who are able to turn data into meaningful knowledge, and provide insights on the reasons behind a certain phenomenon. The end point for any type of analytic enquiry is being able to answer the "why" questions rather than the "what" questions. And in this case, both questions and answers are game-specific. Thus, we, as providers of tools, can enable developers, but cannot provide the answers, as their domain knowledge is essential.

Thomas & Nicholas: we completely agree on one vital outlook for applications of telemetry and behavior profiling and that is: adaptive and predictive game systems. The AI Director in *Left4Dead* (Electronic Arts/Valve, 2008), an automatic AI system that orchestrates difficulty and pacing based on player behavior, proved how this technique has barely scratched the surface of what is possible to achieve when a game system is aware of how it is being utilized by players.

Q: What areas can you identify for future research with game telemetry?
Thomas: If we at Unity manage to identify and standardize the measurement of several game variables across games, such as defining a universal "player death" tag that works on all games where players can die, we can then start looking at differences between games and across different genres. This development together with a permanent player identity could potentially generate a gold mine of data, and analysts can start looking at a long term temporal dimension of play behaviors across different games.

About the Author

Alessandro Canossa, Ph.D. is Associate Professor in the College of Arts, Media and Design at Northeastern University, he obtained a MA in Science of Communication from the University of Turin in 1999 and in 2009 he received his PhD from The Danish Design School and the Royal Danish Academy of Fine Arts, Schools of Architecture, Design and Conservation. His doctoral research was carried out in collaboration with IO Interactive, a Square Enix game development studio, and it focused on user-centric design methods and approaches. His work has been commented on and used by companies such as Ubisoft, Electronic Arts, Microsoft, and Square Enix. Within Square Enix he maintains an ongoing collaboration with IO Interactive, Crystal Dynamics and Beautiful Games Studio.

---

[1] Drachen, A., Canossa, A., & Yannakakis, G. (2009). Player modeling using self-organization in Tomb Raider: Underworld. In *Proceedings of IEEE Computational Intelligence in Games (CIG) 2009* (pp. 1–8). Milan: IEEE Publishers.