



UNIVERSIDAD NACIONAL DEL LITORAL
FACULTAD DE INGENIERÍA Y CIENCIAS HÍDRICAS



Tecnicatura en diseño
y programación de videojuegos

UNL VIRTUAL



Diseño de videojuegos I

Unidad 3

Level 7, Profundidad: La organización del tiempo y la forma del gameplay

Docente
Francisco Cascallares

CONTENIDOS

| | |
|--|----|
| Organizando el tiempo de gameplay | 2 |
| 1. Detrás de la primera impresión..... | 2 |
| 2. Tiempo..... | 6 |
| 3. Niveles y dificultad | 7 |
| Organizando la forma del <i>gameplay</i> (Parte I) | 9 |
| 1. La forma del juego..... | 9 |
| 2. Flujo (Flow): Organizando la forma del juego | 10 |

Organizando el tiempo de gameplay

1. Detrás de la primera Impresión

Para empezar a examinar el concepto de *profundidad*, imaginemos un paisaje natural realmente increíble. La primera impresión de lugares como estos suelen dejarnos absolutamente asombrados y sin aliento.



Como diseñadores, ojalá pudiéramos causar la misma clase de impresión con nuestros juegos, ¿no? No hablo en términos de la estética, sino del *asombro* que nos causan los conceptos, las temáticas, o las propuestas de *gameplay* que ciertos juegos nos ofrecen desde el momento en que entramos en contacto con ellos.

Pero volvamos al ejemplo de nuestro paisaje, en este caso una playa. Podríamos estar minutos o días observando este lugar, hasta que en algún momento nuestro asombro inicial se empieza a consumir, y nuestro interés por el paisaje se desvanece de a poco. Tiene que pasar algo más, por el simple hecho de que ahora queremos algo más de este paisaje. Por ejemplo...



...explorar un poco más. Es decir, interactuar con el entorno. Verlo desde otra perspectiva. ¿Qué sucede ahora con nuestro interés agotado? Se renueva. El escenario nos vuelve a interesar porque, sin haber cambiado (el mar sigue siendo el mar), nos ofrece una experiencia nueva. No hemos necesitado irnos a la montaña para interesarnos por otro tipo de paisaje: no hizo falta cambiar de experiencia. El lugar sigue siendo el mismo, pero descubrimos que hay algo más allá de la superficie. Hemos descubierto que existe una *profundidad*.

Ahora, imaginemos que, por el siguiente par de horas de buceo, nuestra experiencia continúa exactamente igual a lo que muestra la imagen de arriba. ¿No llega un momento en que nos empezamos a aburrir, en que empezamos a pensar en cambiar de actividad, buscar otra nueva que nos *sorprenda*, tal vez incluso fuera del agua? Quizás sea hora de pensar seriamente en ir a la montaña. Hemos tocado el fondo de la profundidad que esta experiencia tenía para ofrecernos, hemos visto todo lo que había para ver y, con la posibilidad de nuevas experiencias *agotada*, tarde o temprano emprendemos el regreso a tierra.

Sería muy distinto si, después de descubrir lo que existe debajo de la superficie del mar, hubiera aún otras nuevas sorpresas...



...como esta.

Si encontramos nuevas cosas para hacer y nos enfrentamos a peligros...



Si de vez en cuando estamos al borde del fin...



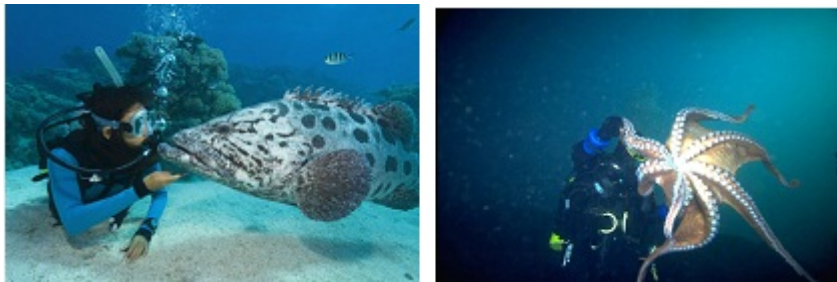
...y escapamos justo a tiempo.

Si luego de los momentos más intensos, tenemos momentos de calma...



...para explorar a nuestras anchas....

Si volvemos a encontrarnos con más sorpresas y más peligros...



Y si siempre hay más para explorar y más para descubrir...



Y hasta ciertas reglas, que al principio parecían fijas, cambian...



...y renuevan toda nuestra manera de experimentar ese lugar.

Si a cada paso que nos adentramos en este mundo hay algo que renueve nuestro sentido del asombro...



...hasta el último momento que pasamos en ese lugar.

Entonces, cuando emergemos de esta experiencia, cuando la hayamos vivido hasta el final sin agotar nuestro interés...



...saldremos aún asombrados, y tal vez hasta cambiados para siempre. (O al menos tendremos una buena historia que contar sobre nuestras vacaciones en el mar).

Es cierto que habría sido mucho más breve limitarnos a decir que la profundidad de un juego es la capacidad que le hemos diseñado para mantener al jugador asombrado o al menos interesado en el *gameplay* (es decir, la experiencia de juego) a lo largo de cierto período de tiempo, para que la primera impresión del juego no sea la última que se lleva.

Pero espero que de este modo hayamos podido experimentar o imaginar más a flor de piel lo que realmente ocurre cuando esto sucede. La diferencia entre:



Emerger al final de un juego de gran profundidad



Entrar de cabeza en un juego sin la profundidad que necesitaba

2. Tiempo

El tiempo no es sólo dinero, por más que un jugador lo invierta en un juego u otro. El tiempo es sobre todo juego, es *gameplay*. ¿Qué significa que el tiempo sea juego?

Un juego, cuente o no una historia, se desarrolla a lo largo del tiempo, como la música o la literatura. Es un arte hecho de tiempo, a diferencia de las artes plásticas o el diseño, que son artes espaciales –por más que podamos pasar horas recorriéndolas con la mirada, se estructuran a través de su manejo del espacio.

Distintos juegos *necesitan* de distintos tiempos para expresar la experiencia que se proponen expresar. *Moon Stories*, de Daniel Benmergui, necesita de unos quince minutos para lograrlo; *World of Warcraft* ha necesitado hasta ahora de 10 años para hacerlo (y no ha terminado todavía). Ambos juegos tienen en común el hecho de que encontraron la medida de tiempo justa para alcanzar la profundidad y la experiencia de *inmersión* (el estado de interés continuo en el juego) que cada uno necesita.

La profundidad de un juego está determinada por su estructura –y la estructura está definida inicialmente por tiempo. El tiempo y la forma que adopta un juego se diseñan en base a lo que un juego necesita realmente, en base a lo que potencialmente podría ofrecer su planteo, en base a nuestras expectativas y necesidades sobre ese juego puntual, en base al género y a las expectativas de la audiencia a la que el juego está dirigido.

Entonces, ¿cuánto tiempo de *gameplay* necesita un juego? Existen juegos de 4 horas. Juegos de 15 minutos. Juegos de 80 o más horas. Juegos que duran 10 años (un tercio de la historia completa de los videojuegos aproximadamente) y juegos infinitos, que jamás terminarán.

En la industria, hay un cierto (flexible) estándar de duraciones a las que distintas audiencias están acostumbradas. Juegos AAA como *Assassin's Creed* suelen tener una duración de entre 20 y 40 horas de *gameplay*, mientras que los juegos casuales son mucho más breves –de 20 a 4 horas, e incluso a veces menos. Los juegos web, por su parte, rara vez superan las 2 horas de *gameplay* y suelen durar 1 hora o menos. Los MMORPG suelen durar muy por encima de 40 horas de *gameplay*, incluso llegando a las 500 horas.

Pero medidas como estas resultan arbitrarias, ya que se basan en expectativas externas (de marketing y audiencia) y no en el potencial interno de un concepto de juego en particular. Podemos, a veces, designar la duración de un juego en base a la historia o mecánica que tenemos para presentar. Aquí no aplican reglas ni guías más concretas que el criterio personal, la experiencia jugando y creando juegos de distintas duraciones, y el hecho de entender el juego que estamos diseñando lo suficiente como para intuir el tiempo de juego que necesitará para expresar al máximo potencial posible la experiencia que tenga para expresar.

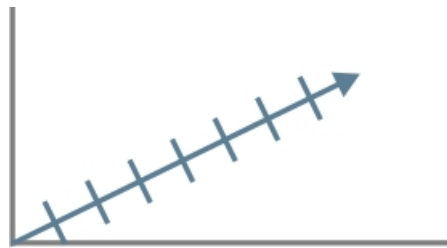
En la práctica, la negociación entre expectativas externas y necesidades internas de un juego es permanente. Y muchas veces, esta negociación nos ayuda a definir un tiempo

mínimo de *gameplay*, que es el primer paso hacia organizar la experiencia que deseamos transmitir al jugador. Todo el *gameplay* de un juego se basa sobre esta primera línea de tiempo que dibujamos. Lo único que falta luego de esto es llenarla con detalles.

3. Niveles y dificultad

Los *niveles* y la *dificultad* organizan el tiempo de juego que nos proponemos a lograr. Suelen ser el primer paso que damos para diseñar la profundidad del juego. Lo que hacen ambos es delimitar con mayor precisión el tiempo de *gameplay* total, y ayudarnos a distribuirlo en unidades claras y manejables.

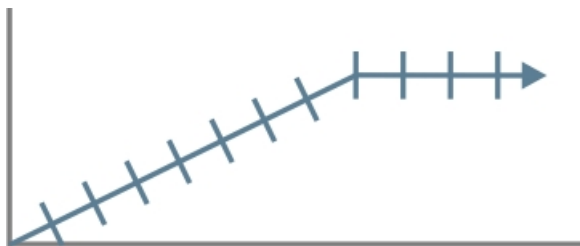
El método más directo es el incremento mecánico. Imaginemos los niveles y la dificultad expresados en una tabla: estiramos una línea ascendente (tiempo) hacia arriba (la altura es la dificultad creciente) y luego dividimos esa línea en segmentos más o menos equivalentes entre sí (los distintos niveles de un juego).



Sin saber aún de qué trata el juego, empezamos ya a ver cómo funciona su estructura –cómo se desarrollará en el tiempo. A medida que se suceden los niveles de juego, la dificultad incrementa mecánicamente y sin saltos abruptos. Por ejemplo, en el nivel 1 los enemigos podrían tener una velocidad de 1 (lentos) y una frecuencia de disparo de 1 (lenta). Incrementando mecánicamente estas variables de a +1 por nivel, conseguimos un incremento constante y parejo de la dificultad desde el principio hasta el final del juego.

De un modo u otro, este sistema es típico de los juegos “retro”: casi cualquier juego de Atari 2600 lo emplea hasta cierto punto. En *Space Invaders* los invasores empiezan cada vez más rápido y más abajo, y su frecuencia de tiro aumenta nivel a nivel. Y dentro de cada nivel, existe un aumento de la dificultad también: cuanto menos enemigos quedan, más rápido se mueven. Hay que tener en cuenta otra variable: la dificultad emergente del juego, en este caso dada por cuántas defensas nos van quedando nivel a nivel (imposible de determinar por diseño, porque es distinto en cada partida). En líneas generales, el incremento mecánico de la dificultad aumenta en cada nivel hasta que el juego se torna imposible de jugar porque sobrepasa nuestras habilidades... o nuestra paciencia con la misma mecánica, que ya se torna repetitiva.

Una variante de este modelo de incremento mecánico de la dificultad es la de congelar la dificultad en el momento precisamente anterior a que el juego se torne imposible de jugar, y mantener la dificultad en una “meseta” desde ese nivel en adelante.



Esta variante se puede aprovechar para juegos con niveles infinitos. El punto en que se congela la dificultad corresponde al tiempo de *gameplay* que se había planificado para el juego. Desde ese punto en adelante, el juego ya habrá mostrado al jugador todo lo que tenía para mostrar, y los jugadores lo irán abandonando gradualmente.

En ciertos juegos, los niveles se piensan de otra manera: son, por ejemplo, eventos (en una aventura gráfica) o *quests* (en un juego de RPG). Sin embargo, siempre organizan el tiempo de juego subdividiéndolo en unidades más manejables, y por eso mismo son comparables a niveles.

De todos modos, los juegos que emplean estos modelos necesitan de mucho más que un aumento constante y parejo de dificultad nivel a nivel para alcanzar el tiempo de *gameplay* que buscan. Estas dos variables (niveles y dificultad) plantean espacios vacíos para rellenar con contenido de calidad, y a la vez sugieren un criterio general con el que rellenar cada uno de ellos –tomando en cuenta cuánto deseamos que dure cada nivel y qué dificultad deseamos que presente al jugador. Todo esto es lo mismo que decir que hay *otra* serie de elementos que aplicar dentro de esta estructura de tiempo aún vacía. Todavía necesitamos definir la *forma* que va a tomar cada uno de estos espacios y, por sumatoria, la forma que organizará el *gameplay* completo de un juego –decisiones que afectarán incluso este modo de organizar el tiempo para agregarle fluctuaciones y cambios que se adapten a un tipo de juego más enriquecido.

Organizando la forma del *gameplay* (Parte I)

1. La forma del juego

Existen distintos modos de organizar la *forma* de un juego –es decir, diseñarle una estructura apropiada. Es importante aclarar algunos puntos antes de continuar con este tema:

1. *La forma ideal de un juego la dicta el mismo juego que tenemos entre manos.*

No existe una forma única de organizar todos los juegos. Las herramientas (como la que veremos en un momento) pueden ser generales, pero debemos usarlas a conciencia para mejorar la experiencia puntual que necesitamos para el juego específico que estamos haciendo –y no convertir nuestro juego en una serie de generalizaciones. Y de esto se desprende que:

2. *Cada tipo de juego, y dentro de esos tipos cada juego en sí, presenta sus propios problemas de diseño a resolver. No resolvamos otros problemas que no sean esos.*

Sería imposible, en el espacio de una o dos unidades, cubrir todo este espectro. También sería ingenuo pretender que las herramientas más básicas alcanzan para solucionar todos los problemas que enfrentamos, a pesar de todos los *cookbooks* que dicen lo contrario. (Lo hacen, sí, pero sólo una vez que vamos ganando experiencia al emplearlas).

Aquí introduciremos algunas cuestiones que atraviesan a muchos juegos, pero es importante que, mientras diseñan, estén muy atentos para captar aquellos aspectos de sus juegos, como ciertos problemas muy específicos que presentan, que no resultan demasiado generalizables. Por ejemplo, los que surgen del tipo de juego puntual en el que están trabajando. Entre un juego del estilo *Tower Defense* y una aventura gráfica (como *Syberia I y II*, o la saga *Monkey Island*) existen seguramente puntos en común, problemas que se trasladan de un tipo de juego a otro porque son comunes a los juegos en general, pero a medida que diseñemos *profundidad* en ellos encontraremos más y más problemas únicos a un tipo de juego, que otros géneros ni siquiera necesitan enfrentar. La secuencia, expresada en *quests*, del tipo y complejidad de los puzzles, o el mejor modo de ordenarlos narrativamente, es un problema que no afecta en gran medida a los juegos de *Tower Defense*¹ pero que existe en el centro del género “aventura gráfica”, ya que a través de puzzles de ingenio, el jugador va construyendo o revelando una historia. Vemos que hay “territorios” (por así llamar a estos espacios conceptuales) que han sido explorados ya muchas veces, y que presentan series de problemas que han sido identificados una y otra vez, perfeccionando la lista de problemas de cada “territorio”. El “mapa” que incluye estos “territorios” se fue cartografiando una y otra vez por las experiencias sucesivas de muchos diseñadores y desarrolladores que exploraron uno u otro territorio con sus juegos, guiándose por las versiones anteriores de estos mapas y agregando o puliendo o rectificando lo que iban encontrando de novedoso.

Ya al mezclar dos mecánicas, dos temáticas o dos conceptos (¡o lo que sea que mezclamos, realmente!) y generar algo híbrido, el mapa empieza a desdibujarse. Los juegos basados en *Guitar Hero* sirven como base u orientación para diseñar juegos como *Patapon*, pero no alcanzan: *Patapon* enfrenta y da alguna clase de solución a problemas que tienen que ver también con otros géneros. Lo mismo con los juegos

¹A menos, claro, que el proyecto sea diseñar alguna clase de *Tower Defense* narrativo, que va contando una historia que se bifurca a medida que resolvemos los niveles de un modo u otro; en ese caso, para inventar esa clase de *gameplay*, les vendrá bien aplicar estructuras de los géneros más narrativos de juegos, como el de la aventura gráfica.

kinéticos², que podrían ser híbridos simplemente porque cuando el controlador cambia, cambia la experiencia: afectan qué se puede hacer en el juego por sus limitaciones técnicas pero sobre todo por las limitaciones físicas de un jugador.

Así, los problemas que plantea el diseño de un FPS tradicional (¡“territorio” explorado, si los hay!) cambian radicalmente al plantearse como juego kinético: se abre una serie de problemas nuevos y muy específicos a los juegos kinéticos, que el género FPS jamás había necesitado responderse hasta ese momento.

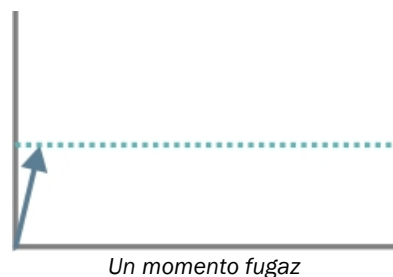
Y, en juegos muy novedosos y experimentales, se encontrarán (con suerte) con problemas que rara vez habrían sido planteados antes; estarán navegando en aguas desconocidas, orientados por su propio criterio, experiencia, sensibilidad e intuición –y por un poco de suerte también, por qué no. Los mapas de estos últimos “territorios” están prácticamente en blanco, esperando a que alguien empiece a delinear sus contornos.

Existen artículos, presentaciones, papers académicos, estudios, posts en devblogs y postmortems sobre todas estas cuestiones. Son cuestiones puntuales, lo que explica por qué muchas veces las mejores charlas (las que más nos servirán eventualmente) en una conferencia de videojuegos tienen nombres de situaciones de diseño tan específicas que a veces pensamos que jamás la enfrentaremos en nuestras carreras. Llegará un punto en que, para resolver problemas, necesitarán empezar a evitar la generalización y adentrarse en cuestiones puntuales –en todas las que puedan. Por el momento, sería imposible reunir todo ese material en este espacio y en una primera materia de diseño de juegos, pero no que se mantengan desde ahora al tanto de la existencia de estas cuestiones. Las investigaciones y proyectos que encontrarán en universidades de diseño y de otras disciplinas, los portales con columnas periódicas de diseño como Gamasutra, o bien los pensamientos que miles de diseñadores independientes o de enormes compañías vuelcan en sus páginas personales o corporativas, les ofrecerán cuestiones puntuales nuevas continuamente para explorar, investigar y continuar formándose.

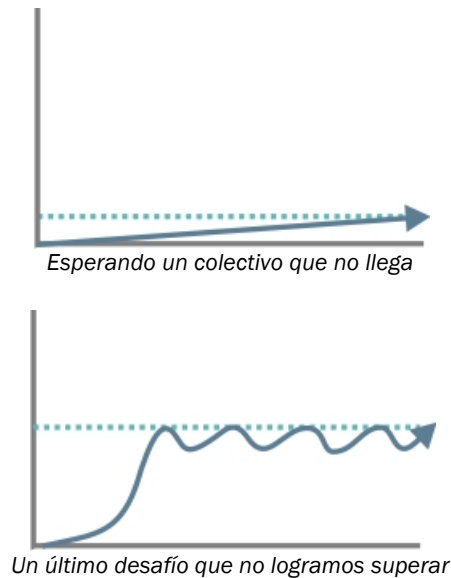
En cuanto a proponer un primer acercamiento a este tema (o sea, generalizar por necesidad): sea como sea el juego que se proponen, existe una herramienta que a mi entender nos puede servir como orientación en una enorme cantidad de casos para diseñar (organizar una forma específica) el *gameplay* de un juego. Ya que hablábamos de navegación, esta herramienta se llama “flujo” y es más conocida como “flow”.

2. Flujo (Flow): Organizando la forma del juego

El tiempo fluye. Ese flujo es la forma básica del tiempo. Y una experiencia, una vivencia, es narrativa: está “montada” sobre el tiempo en la que transcurre. En un cuadro, la extensión y altura que toma la línea que determina el tiempo indica la *manera* en la que el tiempo fluye. Interpretemos algunos cuadros:



²Con controladores o sensores de movimiento, como WiiMote, Move o Kinectic.



Entonces, podemos pensar el tiempo de *gameplay* (el que un jugador experimenta con un juego) como un flujo. ¿Pero un flujo de qué? ¿Con qué materia concreta se rellena este tiempo para que pueda fluir? ¿Cuál es la *forma* de una experiencia, frente a su contenido?

De eso se trataba, realmente, *Flow* –un juego que ya hemos jugado antes en esta materia. *Flow* es la tesis de grado de Jenova Chen, fundador de *thatgamecompany*, y es el principio que mueve a esta manera de estudiar juegos. La tesis explora cómo, a lo largo del tiempo de juego, fluctúa la felicidad del jugador –es decir, cómo el juego lo hace o no feliz en distintos momentos manteniendo su interés o fallando en hacerlo. Con una sencillez muy efectiva, Chen ha explorado este concepto de felicidad del jugador en unos cuantos juegos ya, como los juegos para PS3 *Flower* y *Journey*, así como (en sus inicios) en otros juegos de PC (*Clouds*). Y estas ideas han ingresado al núcleo del *game design* como una manera de pensar en los momentos que componen nuestros juegos –el qué, cómo, cuando de por qué nuestro juego se desarrolla de tal modo y no de otro.

Su propuesta con *Flow* fue un éxito. Al lanzar dicho juego, muchos jugadores experimentados estaban agradablemente extrañados con la fascinación que habían experimentado con este juego: el juego les provocaba algo que no llegaban a poner en palabras. Chen recoge algunas de estas respuestas hacia el final de su tesis:

Tiene que haber algo que está mal en el hecho de jugar toda la mañana con este juego evolutivo... No tiene armas, sangre o explosiones, pero algo me mantuvo pegado a mi silla por mucho, mucho tiempo. Por suerte, mi criatura se quedó sin comida y estuve obligado a abandonar ahí.

[rc.blog\(\)](#)

Flow te absorbe con su física elegantemente sinuosa y te mantiene enganchado en la tarea sutil pero cada-vez-más-desafiante de comer otros bichos marinos. Como un PacMan submarino simple como un ladrillo, Flow consigue la hazaña admirable de sentirse tanto meditativo como adictivo.

[Zen and Art](#)

Por alguna razón no puedo dejar de jugarlo. No tiene mucho sentido, ya que no puedo imaginar por qué debería continuar jugándolo, pero es casi demasiado tranquilizador como para dejarlo.

[Always Beta](#)

Hermoso, relajante y te confunde [...] Es adictivo como si el gobierno hubiera puesto algo en el agua: estás adicto pero no sabés a qué exactamente.

[Something Awful](#)

Jenova Chen tomó el concepto de Flow de los estudios del profesor y psicólogo Mihaly Csikszentmihalyi. Csikszentmihalyi propone este concepto para estudiar cómo podemos ser felices en nuestras vidas, o bien en qué consiste la experiencia de felicidad y cómo la podemos reproducir en los distintos momentos y experiencias. Todo

esto puede parecer medio esotérico, pero en definitiva se aplica a cualquier actividad: si nos hace sentir bien, queremos continuar. Y si no, no.

Antes de resumir la propuesta de Chen en dos o tres párrafos, preferí traducir su tesis para que podamos leerla completa. Leámosla ahora mismo, en inglés o castellano, para poder continuar explorando este concepto en el próximo level.