

Estos términos le obligarán a concebir la aplicación como una película o como una narración. Es decir, deberá conocer y aplicar los recursos del lenguaje audiovisual y conjugarlos con los propios de la informática (interacción con el usuario, gestión de la información, tratamiento de la imagen digital, etc.). Explicarle cómo conjuntarlo todo es también uno de los objetivos de este libro.

Además, ya que en cierto modo deberá usted cambiar de mentalidad hemos elegido una estrategia contundente: le hablaremos de principios. Recuerde, pues, la palabra "principio" tal como la entendía en matemáticas o filosofía: es un enunciado que viene dado y no se pone en cuestión. De estos principios (postulados, axiomas) usted podrá deducir sus propios "teoremas" (o "reglas de aplicación") para sus proyectos multimedia.

En este primer capítulo, por tanto, le daremos a conocer los principios generales que han de regir una aplicación. Una vez los hayamos explicado, le expondremos algunas de las consecuencias que se derivan de ellos, aparentemente dispersas, pero que nos llevarán al apartado con el que terminaremos el capítulo y del cual debe tomar nota a partir de este momento: en una aplicación multimedia cada pantalla es un problema. ¿Qué queremos decir exactamente con esto? Para averiguarlo empiece por observar los principios que siguen.

Principio de la múltiple entrada

Las aplicaciones multimedia normalmente son diseños con perfil de destino, es decir, se conciben para ser utilizadas por un tipo determinado de usuario⁹. Por ello, al enfrentarse al diseño de una aplicación, deberá indagar sobre las características de su "cliente". Para facilitarle esta labor le será útil conocer el principio a que nos referimos en este apartado.

Para hablar de psicología cognitiva a no psicólogos, le podemos decir a *grosso modo* que en el almacenamiento de la información del ser humano intervienen tres parámetros: el cognitivo, el afectivo y el factor de la experiencia previa. Ello significa que la forma en que grabamos la información en nuestra memoria depende de:

- a) la estructura de la información (es decir, de si su complejidad es asumible por nuestras destrezas cognitivas).

⁹ En contraposición encontramos los paquetes de software de amplio espectro (como un procesador de textos, por ejemplo). Estos se diseñan pensando en la ergonomía, pero no en el perfil (conocimientos, expectativas, sentimientos...) del usuario.

- b) el impacto afectivo que esta información tiene en nosotros (los sentimientos con los que ha sido recibida).

- c) nuestra experiencia previa (o de cómo hemos reaccionado anteriormente ante información similar y en qué constructos cognitivos¹⁰ vamos a integrarla).

En consecuencia, al diseñar una aplicación debemos tener siempre presente que no nos estamos limitando a la simple transmisión de información. El mundo multimedia nos permite crear una especie de *profesor*¹¹ que considere estos tres factores. El grado de utilización de ellos marcará la diferencia entre las diferentes aplicaciones del mercado, del mismo modo que recordamos, de nuestra experiencia educativa, aquellos profesores que nos gustaban y aquellos que no.

Así, suponiendo que usted tenga resuelto el problema que le plantea el primer factor (es decir, que ha estudiado bien cómo dividir en unidades la información que va a presentar, cómo van a relacionarse entre sí y cómo las va a aprender el usuario), tendrá que asumir que todavía le queda por hacer lo siguiente:

- a) cuidar que la aplicación cree lazos afectivos con el usuario en todo momento (cómo hacerlo lo irá aprendiendo a medida que lea este libro).
- b) vigilar que la aplicación esté en consonancia con lo que se supone que sabe su usuario modelo (ello le obliga siempre a un estudio del destinatario)¹².

Ahora bien, todo cuanto nosotros podamos introducir en una aplicación multimedia pensando en estos tres factores, "viajará" por lo que se llaman los canales de comunicación. Es decir, en último extremo, se traducirá a texto, imagen o sonido.

Y aquí es donde se hace imprescindible recordar que las personas tienen diferente facilidad de percepción para los diferentes canales. El principio multicanal establece, por consiguiente, que para lograr una buena comunicación hay que utilizar todos los canales.

¹⁰ Para los no psicólogos podemos decir que constructos cognitivos son esquemas mentales, diagramas de conceptos, ideas formadas sobre un tema, etc.

¹¹ Evidentemente, y sin ninguna discusión, limitado.

¹² Lo cual se traduce en la siguiente regla en el caso de multimedia para la formación ocupacional: todo cuanto acontezca en la aplicación debe parecerse al máximo a lo que el usuario se encontrará en su puesto de trabajo. La aplicación debe ser *familiar* al usuario y deberá parecerle, ya a primera vista, *tremendamente útil*; en ausencia de estas dos características (y algunas más) se hunde la efectividad de un plan de formación asistido por ordenador.

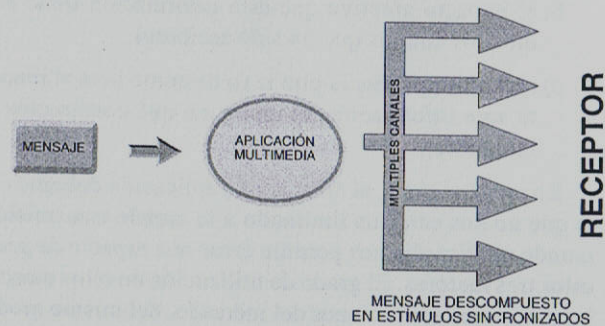


Figura 1.2. El principio multicanal. Una aplicación multimedia es la que manda un mensaje que viaja en diferentes canales perceptivos, pero de forma sincronizada.

Es tan simple como admitir que hay gente que tiene más facilidad para recordar (y entender) algo que ha leído mientras que otra gente necesita haberlo visto en imágenes. Volviendo al ejemplo de la enseñanza, el profesor "democrático"¹³ es aquel que se preocupa de explicar utilizando *todos* los canales, ya que sabe que una explicación oral simple (sin apoyo de pizarra o diapositivas, por ejemplo) perjudicaría a los fuertes en percepción visual y beneficiaría a los fuertes en percepción textual. Por ello (y del mismo modo, en la medida que puedan, deben comportarse las aplicaciones multimedia) el profesor habla, escribe en la pizarra, muestra imágenes, gesticula para reforzar el mensaje¹⁴ y entrega bibliografía donde se encuentran explicaciones alternativas a la que ha presentado en clase¹⁵.

Pues bien, en este planteamiento se sustenta uno de los argumentos educativos (y comunicativos) de más peso a favor de las aplicaciones multimedia: un sistema multimedia es el que transmite una información mediante imagen, sonido y texto de forma sincronizada, y que hace uso adecuado de la capacidad de usar los diferentes canales de comunicación¹⁶. Por tanto, potencial-

¹³ En el sentido de que no privilegia a unos alumnos sobre otros.

¹⁴ Si no se cree lo que estamos diciendo, fijese en cómo los políticos españoles son buenos profesores. Tome nota de cómo enfatizan cada idea con un gesto y una expresión característica; llegará a descubrir el código de gestos que han establecido sus asesores de imagen.

¹⁵ Aunque no se trate estrictamente del uso de un canal en sí, el hecho de entregar información adicional con un enfoque de exposición del tema diferente al que se ha hecho en clase, podrá ayudar mucho a los alumnos que no hayan seguido la explicación. Por tanto, el profesor que suministra material de este estilo es preferible al que plantea la asignatura en torno a sus apuntes "sagrados" (no tanto por el acierto o no de éstos, sino porque no ofrece alternativa a los alumnos cuyo estilo cognitivo no se adapta al de la fuente de información). Por lo que a multimedia se refiere, si es posible (y conseguimos que no sea redundante) podrá ser positivo explicar lo mismo de varias formas completamente diferentes.

¹⁶ Si se trata de aplicaciones para la formación, deberíamos añadir a esa definición que esta sincronización de recursos técnicos está siempre al servicio de un objetivo educativo.

mente, es el que puede hacer llegar un mensaje a un mayor número de usuarios, dada su eficiencia en el aprovechamiento de los canales. En consecuencia, si una imagen no acompaña adecuadamente al texto, si la música va por su cuenta, si un texto compite con otro o desplaza a una foto, si la inserción de una animación interrumpe el discurso en vez de darle continuidad... ¡esto no es multimedia! Todo lo más será un derroche de recursos o una aplicación vistosa que a duras penas se sostiene. El principio multicanal establece las siguientes reglas a respetar en el diseño de aplicaciones:

- a se usarán diferentes canales para transmitir (incluso es lícito acompañar una aplicación con material auxiliar como libros o videos si nos hemos preocupado de establecer un enlace consistente entre ellos).
- b la sincronización de todos los canales utilizados está al servicio de la transmisión (e integración por parte del usuario) de un mensaje.

Por lo tanto, la regla práctica para aplicar en el diseño de aplicaciones es preguntarse sobre cada pantalla (es decir, ¡en todas!), en primer lugar, si lo que percibe el usuario no podría además recibirlo por otra vía. En segundo lugar, si todos los estímulos que se han puesto en juego funcionan de forma acompasada, formando un todo unitario. Si usted consigue pensar constantemente en este principio cuando diseñe, estará sentando las bases para conseguir una buena aplicación multimedia.

Principio de interactividad

La interactividad es un recurso propio de los sistemas informáticos especialmente importante (de entrada, constituye la ventaja principal de las aplicaciones actuales sobre los productos de vídeo tradicional). Por tanto, hablar del principio de interactividad es tanto como decir que *siempre que pueda haber interacción debe haberla*. Ahora bien, no porque el destinatario pueda interactuar se consigue un aumento de la calidad del proyecto¹⁷ **Es más bien**

¹⁷ Una reflexión sobre la interacción para llegar a una conclusión importante: el cine aventaja al teatro en la posibilidad de ofrecer diferentes planos de la acción. De ello se dieron cuenta rápidamente los cineastas y pusieron en práctica toda una serie de enfoques y movimientos de cámara que ayudaron a transmitir el mensaje (le dieron fuerza dramática). En contrapartida, el teatro puede utilizar como recurso la participación del público. Este recurso, bien aprovechado, sirvió como elemento innovador y vitalizador de un género que todavía hoy necesita protección institucional. Sin embargo, a los que nos remuerde la conciencia por asistir poco al teatro nos viene bien la excusa de "... y encima me molestan haciéndome intervenir en la obra!", lo cual refleja una observación importante: la interactividad es un recurso, pero no garantiza que la obra teatral guste al público. Esta conclusión es igualmente aplicable a las aplicaciones multimedia: sin interactividad no hay calidad, pero la interactividad por sí sola no es garantía de nada (incluso puede convertirse en un impedimento).

al contrario: debe planificarse cuidadosamente cada interacción (entrada de datos, elección, forma de señalar, etc.) del usuario con la aplicación. Dicho de otro modo: el diseño de la interacción debe asumirse como una tarea diferenciada (aunque no separada o independiente del resto) dentro de una aplicación multimedia.

En consecuencia, el diseño de la interacción en una aplicación multimedia debe regirse por unas reglas genéricas que deberán considerarse. Las resumimos en seis directrices prácticas:

1. La interacción, como todo recurso, tiene la misma función última que los demás: reforzar el mensaje.

En consecuencia, una aplicación no hace que el usuario entre datos porque sí: hay un motivo para cada intervención. Una planificación seria de la interacción se nota porque el usuario interactúa con la máquina cuando es estrictamente necesario. Si el diseño es correcto, se logrará que el mensaje se haya transmitido mejor gracias al establecimiento de un buen diálogo entre el programa y la persona que lo utiliza. Un ejemplo ilustrativo (y frecuente) de diseño desafortunado de la interacción se da en aquellas aplicaciones que, nada más empezar, piden el nombre del usuario porque sí. Esta decisión es más importante de lo que parece: sólo debe pedirse el nombre del usuario si el diseño recomienda un trato personalizado¹⁸. En caso contrario, se está empezando la aplicación con una redundancia (en el sentido que es una dato innecesario, sobran) que a la larga la perjudicará, ya que el usuario espera saber durante la ejecución por qué se le ha pedido su nombre.

2. El ordenador ofrece la posibilidad de aplicaciones altamente interactivas. Por tanto, cada vez que se entra en un proceso no interactivo se desperdicia la potencialidad del medio.

¹⁸ Algunas aplicaciones se enfocan al trato personal (por ejemplo, las que persiguen un cambio de actitud en el usuario, las que pretenden concienciarle de algún problema social, las que necesitan de una implicación emotiva fuerte, etc.). En todas ellas tiene sentido que el usuario introduzca su nombre (la aplicación se "personaliza") y durante el diálogo que se establezca con él se utilice el nombre introducido. Sin embargo, ello a veces no es en absoluto recomendable. Por ejemplo, a veces la respuesta del usuario es más sincera si el trato es anónimo.

Un caso particular en el cual hay que discutir la personalización es el de las aplicaciones cuya finalidad es evaluar individualmente el rendimiento del usuario. A este respecto, hay que aclarar que individualizar no es personalizar. Si la aplicación se utiliza para calificar el rendimiento de una persona lo único que hace falta es identificarla: con que se le pida el DNI, o un código de identificación es suficiente. Como hemos dicho antes, si esta evaluación se enmarca en un programa cuya intención es lograr un cambio de actitudes de un usuario (otros ejemplos: concienciación sobre el problema del medio ambiente, promoción del gusto por el trabajo bien hecho, promoción de la solidaridad y la tolerancia, etc.) sí que puede ser conveniente pedirle el nombre, para conseguir que se implique afectivamente en el discurso de la aplicación.

En consecuencia, se deben evitar los períodos de tiempo excesivamente prolongados en los que el usuario no interviene: lectura de textos extensos en pantalla, secuencias prolongadas de sonido e imagen animada, etc. Introducir uno de estos fragmentos de larga duración en una aplicación (aunque usted piense que son de gran calidad) es como apostar para que el usuario la abandone¹⁹.

3. La interacción implica participación activa, no repetición de gestos.

Con ello queremos decir que toda interacción conlleva una decisión entre alternativas (o, formulado en negativo con un ejemplo: la intercalación de pulsaciones de teclado en una secuencia de imágenes raramente puede ser considerada como interacción). En resumen, no hay nada más contraproducente que el usuario tenga la sensación que continuamente "le está dando al Intro para pasar de una pantalla a otra". Es decir, si usted está empeñado en que el usuario lea un texto largo y, para evitar lo que decíamos en el apartado anterior, lo descompone en fragmentos y los muestra secuencialmente, no conseguirá nada. Mejor evite que el usuario tenga que leerse un texto tan largo: obliéguese a diseñar su presentación de forma auténticamente interactiva.

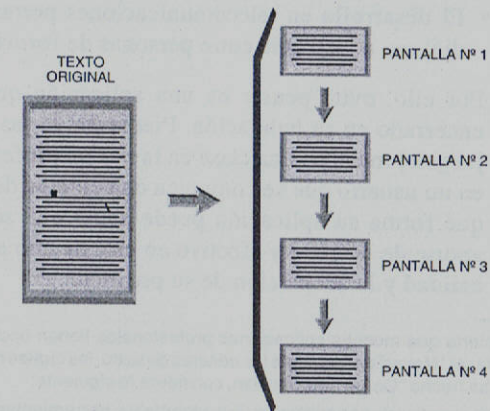


Figura 1.3. La división de un texto largo en partes más reducidas no logra que el usuario lo perciba como una estructura interactiva, pese al cambio de pantalla.

¹⁹ Aunque alguien pueda acogerse a la calidad de las animaciones introducidas para argumentar en contra de este principio, no debe olvidarse la siguiente regla: una animación cuanto más corta, mejor. Piense que cada vez que se inserta una trama de vídeo, o bien es interesantísima y capta la atención del usuario o éste tendrá la sensación de perder el tiempo (y a este respecto es importante señalar que al autor de la aplicación siempre le parece interesantísima la trama de vídeo, por lo que deberá aprender a ser muy crítico con su trabajo o el de su equipo).

4. No es aconsejable recordar al usuario que no puede interactuar.

Como regla general evite la aparición en pantalla de zonas inertes pero aparentemente sensibles. En particular, a veces se encontrará con que temporalmente es necesario desactivar un determinado botón o punto sensible de la pantalla. En estos casos, a ser posible, es mejor ocultar de la escena aquellas opciones inactivas (¡porque no se imagina cuánto frustra al usuario pulsar sobre un sitio en el que no pasa nada!). La única limitación a este criterio es el peligro de convertir la aplicación en un baile de objetos que aparecen y desaparecen²⁰.

5. La interacción no se limita al esquema usuario-máquina.

Debe concebir este concepto en un sentido más amplio. Por ejemplo, también forma parte de la *interacción* de una aplicación el prever que varias personas participen en ella a la vez (y, por tanto, ofrecer unas pautas de discusión, unos roles a desempeñar, etc.). Tenga en cuenta los siguientes factores:

- Se valora siempre que, lejos aislarse, el ordenador promueva que las personas dialoguen y cooperen.
- El desarrollo en telecomunicaciones permitirá, a partir de ahora, el diálogo a distancia entre personas de forma habitual.

Por ello, evite pensar en una aplicación que utiliza un solo usuario encerrado en su habitación. Piense en varias personas que se reúnen y juegan juntas, en una clase en la que un profesor habla con los alumnos, en un usuario que se comunica con el tutor de su formación... piense de qué forma su aplicación puede hacer que se relacionen. Todo lo que aporte de original y efectivo en este campo aumentará notablemente la calidad y la aceptación de su proyecto²¹.

²⁰ Es cierto que muchas aplicaciones profesionales tienen opciones inactivas. No hay más que recordar el "Mover" o "Pegar" de los editores de texto, los cuales no están en negrilla si previamente no se ha hecho "Cortar". Ahora bien, considere lo siguiente:

- Estas aplicaciones suelen ser instrumentales: herramientas que sirven para dibujar, procesar textos, gestionar, etc.
- La utilización de las opciones de edición citadas anteriormente es sobradamente conocida por los usuarios. Es decir, todos saben que no se puede hacer "pegar" si antes no se ha hecho "cortar", por lo que no intentarán pulsar sobre la primera opción cuando esté inactiva.
- Pero, para aquellos en que el usuario no conozca la mecánica de uso de las diferentes opciones, piense en usted mismo: recuerde lo mal que le sienta cuando en un programa un botón está inactivo y usted no entiende por qué.

²¹ Si al leer el párrafo anterior se le ha ocurrido que:

- 1) en sus aplicaciones puede incluir preguntas abiertas.

6. La interacción permite obtener un registro de datos descriptivos de la conducta del usuario.

Este es uno de los valores añadidos de la interacción: permite el estudio de las reacciones del usuario ante las situaciones que le plantea una aplicación. La elaboración de *diarios de respuestas*, generados por la propia aplicación y que registran las decisiones del usuario, abren las puertas a estudios posteriores sobre la efectividad de la aplicación, la conducta del usuario o el proceso general de utilización (todas estas cuestiones se enmarcarán en un diseño general que expondremos en el capítulo 5, al hablar de aplicaciones para la formación de los usuarios).

Principio de libertad

Una vez que se ha logrado un diseño interactivo, donde el usuario no es un mero espectador de los acontecimientos, se ha conseguido uno de los principales objetivos de la aplicación: convertirle en actor de la misma.

Ahora bien, el principio de libertad debería llamarse de otra forma, ya que su enunciado es contradictorio: el objetivo del diseñador de una aplicación multimedia es que el usuario piense que navega libremente, mientras que en realidad está inmerso en un esquema de etapas predeterminado.

El objetivo del guionista es ocultar este esquema: Es decir, una aplicación mal diseñada es la que aparece a la vista del usuario como una secuencia lineal de contenidos o etapas. Por el contrario, un buen diseño será el que consiga una impresión totalmente diferente: el usuario percibe la aplicación como un mundo en el que se mueve sin ninguna ruta prefijada (y precisamente en este tránsito acumula información y experiencias).

Veamos un ejemplo: supongamos que una aplicación necesita forzosa-mente que el usuario reciba las informaciones A, B, C y D (piénsese que son cuatro imágenes concretas o cuatro textos que hay que leer). Aunque se economice tiempo y esfuerzo de diseño exponiendo cuatro pantallas consecutivas (el usuario pasa de una a otra pulsando el ratón) el resultado suele ser contraproducente. Mucho mejor es presentar la información inmersa en un

- 2) el usuario podrá consultarlas con un tutor o con otros compañeros (modelos de *aprendizaje entre iguales*).
- 3) de esta discusión debidamente dirigida se puede conseguir la mejora del proceso de formación y una mayor eficiencia de la aplicación, entonces...

...enhorabuena, ¡es usted un genio atrasado! Porque estos mecanismos son los que ya emplean las empresas de aplicaciones telemáticas, apoyadas en correo electrónico o en material audiovisual por correspondencia.

grafo de escenas²² y provocar que el usuario la descubra (¿Cómo hacer que la descubra en su totalidad? ¡Este es el problema del guionista!).

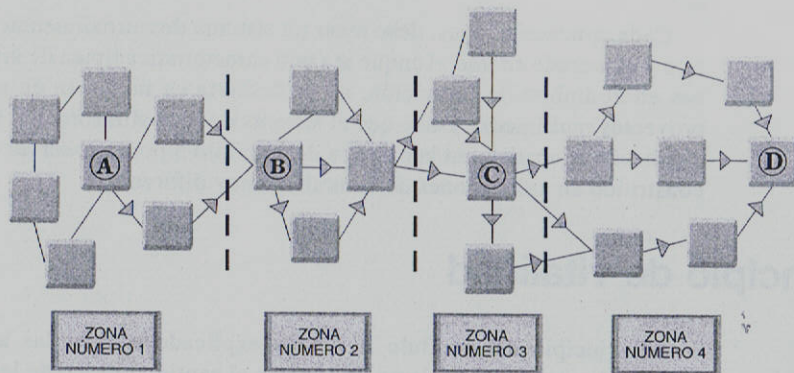


Figura 1.4. La libertad aparente del usuario. Cuatro escenas de paso obligado (A, B, C y D) pueden sumergirse en un grafo de escenas mayor, de forma que el usuario crea que las descubre en su navegación por la aplicación.²³

La estrategia más simple consiste en sumergir esta *secuencia lineal* dentro de un grafo mayor, de manera que el usuario no se dé cuenta que se está moviendo por una estructura que en el fondo le obliga a seguir una *secuencia*²³. De todas maneras, debe tener presente que para construir una aplicación que respete el principio de libertad es necesario conocer otras reglas narrativas (que se verán en los capítulos 3 y 4). Mientras tanto, lo que sí puede hacer ahora mismo es anotar cómo no debe ser su aplicación: debe evitar a toda costa la sucesión determinista de pantallas (es decir, la percepción de que la aplicación es un pase de diapositivas).

Principio de retroalimentación

Cuando se produjo el "tirón" de los lenguajes gestores de bases de datos, su aportación se notó en dos facetas fundamentales del mundo empresarial: la gestión y la consulta. Las aportaciones en la primera faceta se concretaron en

²² En la práctica del capítulo 3 se definen los diferentes tipos de *grafos de escenas*.

²³ Normalmente, en una aplicación por descubrimiento, el usuario se da cuenta de todos nuestros trucos la segunda o tercera vez que la usa hasta el final. Sin embargo, se trata de pensar normalmente en la primera vez que el usuario juega. La inclusión de pruebas aleatorias, obstáculos, juegos, etc. añadirá un grado de indeterminación suficiente para que pueda utilizarla otras veces sin aburrirse.

la informatización de algunos procesos. En la segunda, se incorporó a la documentación interna los datos sobre funcionamiento generados por las propias aplicaciones.

Si bien en un principio se priorizó la informatización de las tareas, enseguida se vio que para la orientación de la empresa se debía contar con un buen sistema de consulta. Consecuentemente, las aplicaciones de gestión se transformaron en herramientas que, a la vez que resolvían los problemas organizativos del presente, generaban documentación para el futuro. En estas aplicaciones se introdujeron rutinas de elaboración de resúmenes e informes, de balances y otros documentos, que serían punto de partida de futuras tomas de decisiones.

La idea de un sistema que genera información y se utiliza para corregir su funcionamiento, se denomina en diferentes idiomas (y con diferentes matices) *retroalimentación*. Para adaptarla a las aplicaciones multimedia hay que tener presentes los cuatro puntos siguientes:

1. ¿Qué información se recoge?
2. ¿Cómo se presenta?
3. ¿A quién se dirige?
4. ¿Cómo se procesa?

Por ejemplo, imagínese que su aplicación enseña un idioma. Los alumnos (3) disponen de un listado (2) que les informa de sus errores (1), les indica cómo corregirlos (4) y les orienta sobre los progresos conseguidos (4) desde que empezaron a estudiar. Este es uno de los casos más simples, en el que la aplicación utiliza la información generada durante su uso para que revierta en el progreso del propio usuario.

Evidentemente, los dos apartados marcados como (4) deben pensarse cuidadosamente, pero ya intuimos que en el primero se presentaría una casuística de diagnósticos y en el segundo algunas mediciones indicadoras del nivel alcanzado en cada momento.

Otra posibilidad sería la de una aplicación para la formación que se instalara en la red de una empresa. Sería posible que ahora la aplicación evaluase los niveles adquiridos por los usuarios en diversas pruebas (cuestionarios sobre conocimientos técnicos, por ejemplo) y que se quisiera recoger unos recuentos globales para ajustar tanto los contenidos de lo que se enseña como los niveles exigidos en la evaluación.

En este último caso fíjese que el destinatario de la retroalimentación (3) es la propia empresa que solicita la aplicación. Por este factor, entre otros, habrá llegado a la conclusión de que se trataría de un sistema con ciertas diferencias

con el anterior²⁴. Sin embargo, tanto en un ejemplo como en otro, debe planificarse cuidadosamente la recogida de la información, ya que deben considerarse las variables más importantes a tener en cuenta para que se facilite el posterior análisis del proceso de formación.

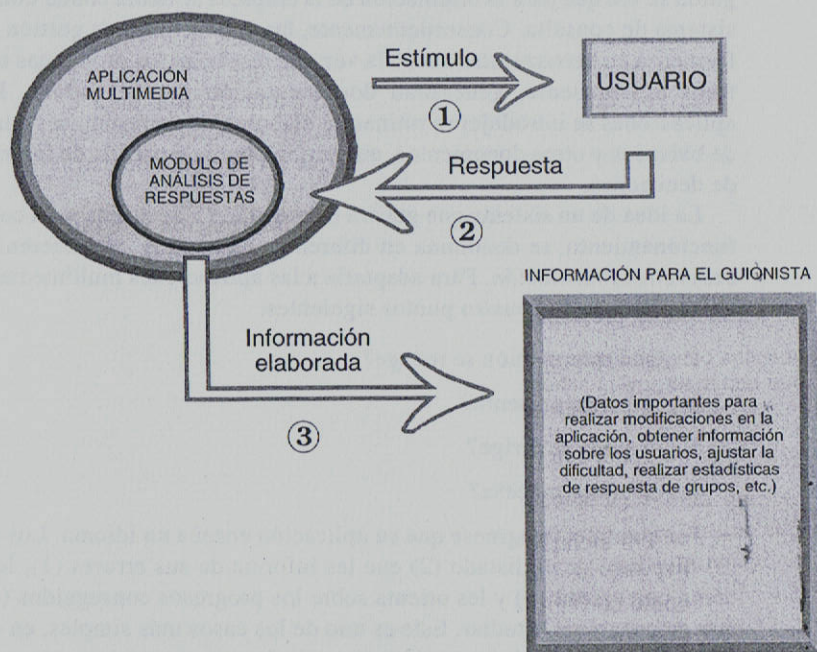


Figura 1.5. Retroalimentación. El módulo de análisis de respuestas obtiene información importante para el propio guionista.

A este respecto, seguro que oírás más de una vez una palabra gastada que está de moda incluir en todos los proyectos de formación²⁵: *feed-back*. Seguramente la verás en un esquema que lleva dibujada una flecha curvada que va del alumno al profesor, en sentido contrario a una flecha recta que va del profesor al alumno y es etiquetada como "acción formativa", "intervención", etc. Si no hay una explicación de cómo se ha diseñado el proceso que

²⁴ Nótese la diferencia fundamental: el primer sistema de retroalimentación intenta corregir la marcha de cada alumno en particular, mientras que el segundo intenta corregir la del sistema global de formación de la empresa.

²⁵ También se usa el concepto en aplicaciones multimedia en general, refiriéndose al proceso de refinado (mejora, aumento de la eficiencia) de las mismas. No obstante, el campo donde genuinamente aparece un *feed-back* en el proceso suele ser el formativo.

representa esta flecha curvada y qué documentación va a producir la aplicación, quiere decir que el autor del informe no lo ha pensado cuidadosamente²⁶.

Cada aplicación, pues, debe tener un sistema de retroalimentación específico y elaborado *ad hoc*. Aunque sea una característica típica de las aplicaciones en el ámbito de formación, no se descarta su inclusión en otro tipo de proyectos multimedia. Dado que el sistema de retroalimentación forma parte del diseño educativo, en la práctica del capítulo 5 podrá observar cómo se ha construido en aplicaciones de naturaleza muy diferente.

Principio de vitalidad

Al principio del capítulo le hemos explicado que en las aplicaciones multimedia no puede hablarse de *vista* en el sentido clásico de las aplicaciones de gestión. La evolución de este concepto nos ha llevado a establecer que las aplicaciones multimedia deben ser, ante todo, *dinámicas* y, por tanto, la construcción de la "vista de usuario" ha adquirido el suficiente protagonismo como para distinguir entre trabajos bien hechos y mal acabados o poco profesionales.

Más tarde, al exponer el principio de interactividad, le hemos advertido que la entrada de datos también debe obedecer a unas reglas. El seguimiento de ellas dará cuerpo a la aplicación y la dotará de lo que se llama un sistema de interacción. Es decir, el usuario apreciará que en la producción de la aplicación se han invertido horas en pensar cómo iban a ser las relaciones entre él y la máquina.

Y, finalmente, hemos señalado que la retroalimentación es un mecanismo que precisa ser pensado celosamente en las aplicaciones de formación. También este aspecto servirá de criterio para apreciar aquellas aplicaciones en las que se ha conseguido que los datos recogidos reviertan en la mejora de la aplicación (de sus objetivos formativos, de la gestión de información generada, etc.).

Con estos antecedentes, no le sorprenderá lo que ahora le digamos en este apartado sobre la *vitalidad*. Para ser concisos intentaremos resumirlo en una frase: **toda pantalla está viva. Es decir, el usuario tiene que percibir la aplica**

²⁶ Es más: seguramente se trata de una persona o una empresa que no ha desarrollado el proyecto y que está incluyendo este gráfico porque queda bien. Probablemente no tenga ni idea de cómo se lleva a la práctica. En el peor de los casos, resultará que no es consciente de que el *feed-back* constituye un proceso en sí, y le explicará que se trata de unas entrevistas que mantienen los alumnos con el profesor y que dicha información revierte en la mejora del proceso formativo.

ción como algo que funciona autónomamente, como un mundo al que se asoma. Con ello se va más allá del principio de interactividad: en la aplicación siempre sucede algo... Aunque el usuario no haga nada!

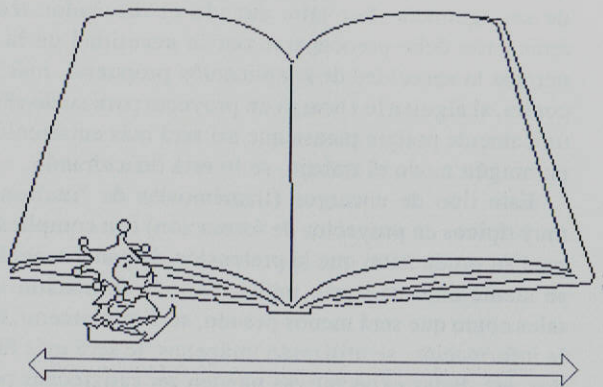


Figura 1.6. Vitalidad de la aplicación. Basta que, mientras el usuario piensa su elección, una mascota recorra la pantalla en uno y otro sentido para dar la sensación de que la aplicación está viva.

Un ejemplo bastante ilustrativo, fuera del mundo multimedia, de lo que llamamos "aplicaciones de pantalla viva" lo constituyen los sistemas de control de la producción a tiempo real. Estos consisten, básicamente, en un ordenador (o sistema informático) conectado a una planta industrial, de la que recoge cada cierto período de tiempo (normalmente cada uno o dos segundos) la cantidad de material producido por cada máquina. También recoge las incidencias de interés (la máquina 23 se ha parado, la 134 exige recargar materia prima, etc.). El usuario (un jefe de planta, un gestor, o un operario) acude al sistema para recoger datos acumulados, como la cantidad producida perteneciente a cierto pedido, para poder pronosticar cuándo podrá ser entregado.

No es por casualidad que en el diseño exterior de estos sistemas aparezcan dibujadas en pantalla siete u ocho máquinas (animadas si están trabajando y fijas si se han parado) con sus correspondientes gráficos o datos sobre rendimiento (por ejemplo, la velocidad de funcionamiento que es "refrescada" cada segundo).

No puede imaginarse la satisfacción que supone para el usuario observar cómo aquellos datos varían a cada instante (y no digamos, si una máquina pierde velocidad y en seguida se observa en la pantalla). Aunque el hecho de

mostrar estos estados puntuales sea totalmente gratuito²⁷, contribuye a que el usuario tenga la sensación de que el sistema "está trabajando" y no desperdicia ni un instante para contribuir a la mejora de la producción.

Si extrapolamos la idea anterior al campo de la formación (por poner una actividad en la que se produce ya una fuerte demanda de aplicaciones), el hecho fundamental es que la aplicación *nunca deja de enseñar*. Por ejemplo, si se trata de un programa para aprender idiomas, deberá aprovechar cuando el usuario no hace nada para intercalar frases, canciones, observaciones, etc. Tanto da si se escriben como texto en pantalla o se oyen por el altavoz, la cuestión es que suceda algo que contribuya al aprendizaje mientras el usuario está decidiendo qué tecla pulsa.

En el caso particular de los idiomas, es muy fácil conseguir que la pantalla esté viva en las pausas (precisamente por ello, podrá apreciar cuántas aplicaciones mal diseñadas hay en este campo). En otro tipo de formación también puede conseguirse respetar este principio, pero le llevará más esfuerzo pensar qué puede contribuir realmente a mejorar el proceso de aprendizaje²⁸.

En las aplicaciones multimedia en general (ocio, cultura, catálogos, etc.) el que aparezcan pantallas vivas será altamente determinante del éxito de las mismas. Por poner un ejemplo, basta que nos fijemos ante qué tipo de aplicaciones (independientemente de su calidad) se paran los visitantes de las exposiciones de software.

La próxima vez que visite una de ellas (una feria de informática, un encuentro de editoriales en CD-ROM, etc.) dedique parte de su tiempo a observar cuáles son los *stands* que, sin una presencia considerable de empleados, consiguen atraer la atención del público. Descubrirá aplicaciones muy dinámicas, llenas de pantallas vivas que se mueven mientras desfilamos ante ellas. Entonces verá cómo la gente se para y termina por curiosear. ¿Cree usted que, hoy en día, es suficiente para atraerlos el dejar una foto fija digitalizada en un monitor?

Una visión más detallada del principio de vitalidad

Para terminar la exposición de lo que contribuye a la *vitalidad* de una pantalla, será útil considerar tres observaciones importantes sobre los elementos que se colocan en ella:

²⁷ Y funcionalmente delicado, ya que el sistema sacrifica parte de su tiempo para preparar el refresco de la imagen.

²⁸ Le será de ayuda consultar la práctica del capítulo 2, donde se habla del *lote de tareas de fondo* de la escena.

1. Resultan agradables a los usuarios los iconos animados que se mueven aunque no se *cliqué* sobre ellos.

Esto es lo que intentamos que entienda del principio de vitalidad. Intente que las pantallas no estén llenas sólo de botones planos de color metalizado. Intente situar alguna mascota que se mueva (sin desplazarse) y reaccione al *clic* del usuario.

2. Resultan agradables los iconos que responden instantáneamente al usuario.

Es decir, lo que atenta más intensamente contra la apariencia "viva" de una aplicación es la inercia en la respuesta. En este sentido, si por fuerza el usuario tendrá que esperar después de *clicar* sobre un icono, haga todo lo posible para distraerle (que se dispare la música, que cambie el cursor, que el icono baile...) antes de entrar en la operación lenta.

3. Resultan desagradables a los usuarios los botones que no van a responder.

Aunque ya se lo indicamos en el apartado sobre interactividad, se lo recalcamos ahora bajo el punto de vista de este principio. Si deja un icono visible e insensible en una pantalla está contribuyendo a que su aplicación pierda en dos aspectos: está dejando una pieza que no responde y que contribuye a dar a la pantalla una apariencia muerta. Aunque lleve más trabajo de programación, por regla general será mejor hacer desaparecer los iconos o botones que inactive²⁹.

Principio de necesidad

A excepción de notables casos particulares, todas las aplicaciones deben regirse por el principio de necesidad: deben ser *necesarias*. Esto quiere decir que, para su diseño, se debe partir de dos *a priori*:

1. La aplicación sirve para algo (necesidad de la existencia de la aplicación).

2. La aplicación debe ser multimedia (necesidad de ser diseñada, precisamente, bajo este enfoque).

²⁹ Sólo en algunos casos hacemos caso omiso de esta regla. Por ejemplo, si estamos en un proceso en el que el usuario ha de realizar una serie de trabajos, se podría discutir si hace falta que los botones inactivos estén ahí para recordárselos.

Es decir, la aplicación viene a resolver un problema (llenar el tiempo de ocio con entretenimiento también es un problema) cuya solución percibimos inmediatamente que requiere de un diseño multimedia. Toda la producción que no nazca de estas dos condiciones es gratuita y, por tanto, corre el riesgo de ser ignorada. Por ello, cuando el diseñador recibe el encargo de una aplicación debe preocuparse por la necesidad de la misma. Cuanto más se perciba la necesidad de la aplicación propuesta, más fácil será diseñarla. Por contra, si alguien le encarga un proyecto para su diseño en soporte multimedia únicamente porque piensa que así será más entretenida, no le está facilitando en ningún modo el trabajo, se lo está dificultando.

Este tipo de encargos (llamémosles de "multimediatizar" aplicaciones, muy típicos en proyectos de formación) son complicados si no hay otra causa para su producción que la pretensión de ganar amenidad. Cuando un usuario se sienta ante un curso multimedia de formación suele tener expectativas tales como que será menos pesado, será más ameno, tendrá acceso más fácil a la información, se utilizarán imágenes, le será más fácil recordar los contenidos, etc. Estas expectativas pueden ser satisfechas por una presentación vistosa y, durante un tiempo, por una secuencia de pantallas diseñadas con acierto. Pero llegará el momento en que el usuario se preguntará si era realmente necesario sentarse ante un ordenador para aprender de la forma en que lo hace. Esta conducta explica el fracaso de muchos proyectos que no han hecho más que "informatizar" productos editoriales ya existentes, sin aportar nada nuevo. Cuanto más famoso sea el producto original, más peligrosa es su informatización. Por ello, antes de asumir un proyecto deben detectarse las ventajas que supone para el usuario la informatización del mismo.

Si un guionista consigue dar la vuelta a este principio se encuentra con una aportación tremendamente positiva. Es decir, si alguien consigue acertar cuál es la causa significativa que justifica el proyecto y que hace necesaria la existencia de la versión multimedia, entonces ha encontrado una aplicación con futuro.

Veamos un ejemplo: ¿Es necesario "multimediatizar" un control de stock?, o lo que es lo mismo: ¿Hace falta que una empresa encargue la reprogramación en formato multimedia de su sistema informático de pedidos y escandallos?

La respuesta sería afirmativa en los casos siguientes:

- en el sistema actual se comenten errores ya que se confunden artículos.
- los usuarios de las terminales deben, además, recoger físicamente los artículos de un almacén complejo.

En estos supuestos, sí podría ser útil que las referencias se acompañaran en pantalla de imágenes gráficas (para evitar confusiones) o que antes de la

Por tanto, la forma más fiable de poner en marcha un buen control de calidad empieza por la recogida de datos sobre el uso de la aplicación... ¡hecha por la propia aplicación! Incluso, con una adecuada planificación del sistema de recogida de respuestas, se puede obtener mucha información susceptible de ser tratada estadísticamente (pruebas de correlación, de detección de tendencias, de impacto por grupos, etc.). De este tratamiento pueden obtenerse resultados realmente interesantes.

Principio de atención

Sin ser psicológicamente demasiado rigurosos, podemos entender por atención la apertura selectiva del individuo al entorno, es decir, la postura de selección de información que presentamos y sobre la que el individuo actuará.

Si el guionista se deja de preocupar por la atención, todo se desmorona: los errores en los programas provienen muchas veces de fallos debidos a que la información importante no se lee (ni se escucha, ni se mira) si no llama la atención.

El objetivo de las aplicaciones es mantener la *atención sostenida*, es decir, conseguir que el receptor mantenga una actitud continua de expectación ante la aplicación. Para ello disponemos, de entrada, de dos factores que nos pueden ayudar a conseguirla: la naturaleza misma de la aplicación y la apariencia de la aplicación. A la generada por el primer factor la identificamos con la atención cognitiva y a la generada por el segundo, con la atención afectiva³⁵.

Atención cognitiva

Es la que se basa en el valor de la información suministrada. Es típica de las aplicaciones profesionales o de contenidos muy particulares. Se hace especialmente atractiva para los usuarios especializados a los cuales va dirigida (y, por tanto, que perciben la importancia de la información que se transmite). Para conseguirla hace falta que:

- en efecto, la información sea relevante.
- la información esté bien organizada.

³⁵ Estas definiciones no proceden de la psicología. Son una forma de hablar para referirnos a la forma de captar la atención en las aplicaciones multimedia.

En consecuencia, cuanto más especializados sean los contenidos de una aplicación, más podremos basar el diseño de la aplicación en los principios de la atención cognitiva.

Una muestra típica de esta clase de proyectos son los de enseñanza de la medicina. Supongamos que realizamos una aplicación que prepara al usuario para el reconocimiento de tomas microscópicas. Se entiende que tiene que llegar a identificar los elementos importantes, el tipo de tejido, la tinción utilizada, etc. Pues bien, si se dispone de buenas tomas digitalizadas, si el tratamiento de la información está bien organizado, si hay un sistema de evaluación y si no se cometen errores importantes, la aplicación será bien recibida por los futuros usuarios. Es decir, se juega con la ventaja que nos da el interés del receptor: si el médico (o médico en formación) juzga que la calidad de la imagen es la misma que en el microscopio y, además, toda la organización es coherente, el captar su atención está garantizado.

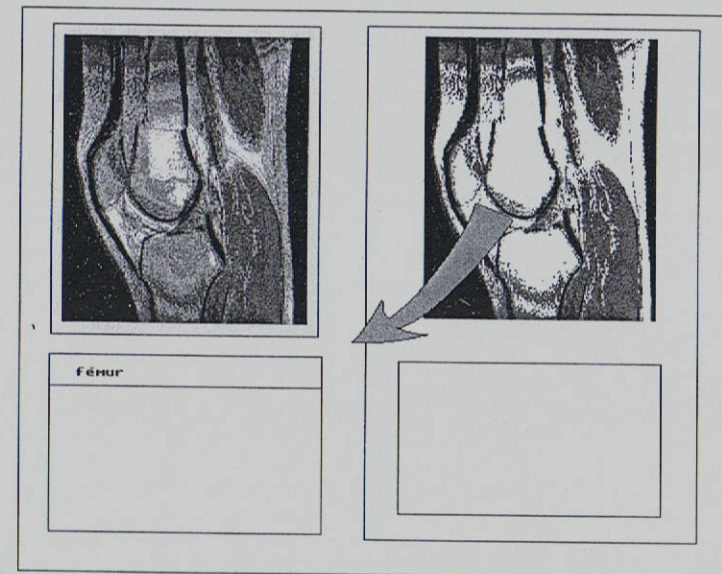


Figura 1.9. Atención cognitiva. Una aplicación médica llama la atención del usuario gracias a la valía de la información que aporta.

Atención afectiva

Se basa en el lazo afectivo que se establece entre el usuario y la aplicación. Para conseguir crear este lazo, dedicaremos el capítulo cuatro a explicar los

dad a considerar por el guionista es una "página virtual y carente de orden" que se compone de la escena maestra y del polígono de viñetas.

Tomemos como ejemplo la escena de la tienda de comestibles. Supongamos que, en primer lugar, el usuario puede dirigirse a la estantería de los refrescos y allí elegir uno.

Además, también habrá otro cambio de decorado para ir a comprar productos lácteos. Y, finalmente, podrá visitar una tercera sección donde comprará comida precocinada. No es obligatorio realizar las compras en este orden, el usuario se moverá libremente por la tienda de comestibles entrando en la sección que quiera.

Todo ello, en el grafo de escenas (tanto el general como el exhaustivo), se empieza a construir de la siguiente manera:

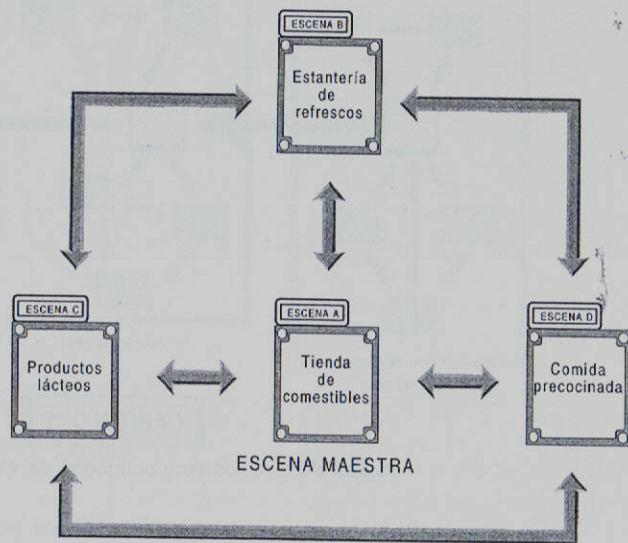


Figura P3.3. Escena maestra y sus adyacentes.

Las aristas entre las escenas B, C y D de la figura P3.3 indican que se puede ir de una a otra sin pasar por la escena maestra. El guionista podría haber obligado al usuario a pasar por A en cada cambio de escena; pero tal como hemos dibujado el grafo de la figura P3.3, no hay ninguna restricción de este tipo. Si el guión impusiera unas condiciones de prioridad en las escenas adyacentes, entonces se señalaría con números de orden de la siguiente forma:

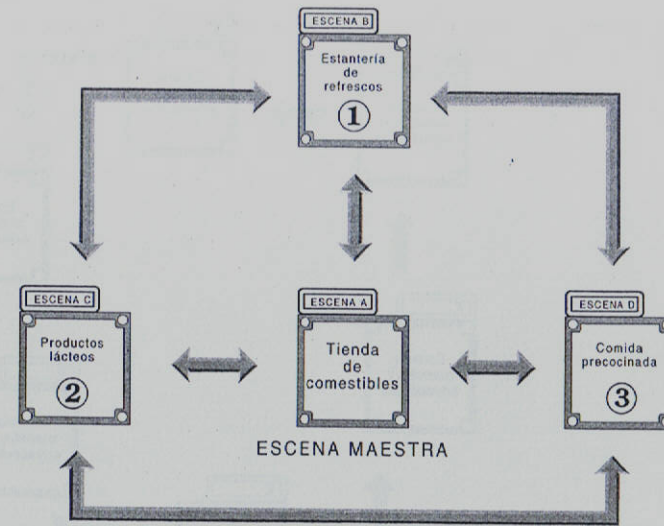


Figura P3.4. Escenas adyacentes ordenadas.

Tal como está la figura, el juego exige que el usuario visite primero la sección de refrescos, después la de lácteos y, finalmente, la de comida precocinada.

Sigue siendo opcional el pasar o no por la escena maestra, pero los objetivos del guión hacen que se siga el orden que se ha detallado.

Grafo general de la aplicación

Hay una excusa argumental para pasar de un decorado al siguiente en las escenas de la figura P3.3. Sin embargo, estas apreciaciones locales no se detallan en el grafo general de la aplicación.

En él aparecen las escenas formando un grupo compacto y se señalan, exclusivamente, las enlaces para salir de dicho grupo (otros elementos a nivel general que también se anotan en el grafo se explicarán en el apartado siguiente).

En el ejemplo que nos ocupa, las diferentes escenas maestras se visitan en el orden que se señala en el grafo general y, al final, se vuelve a la escena inicial donde empezó el juego. Dicho grafo, a simple vista, nos revela esta estructura circular en la que el usuario se mueve libremente dentro de cada uno de los 6 polígonos de viñetas.

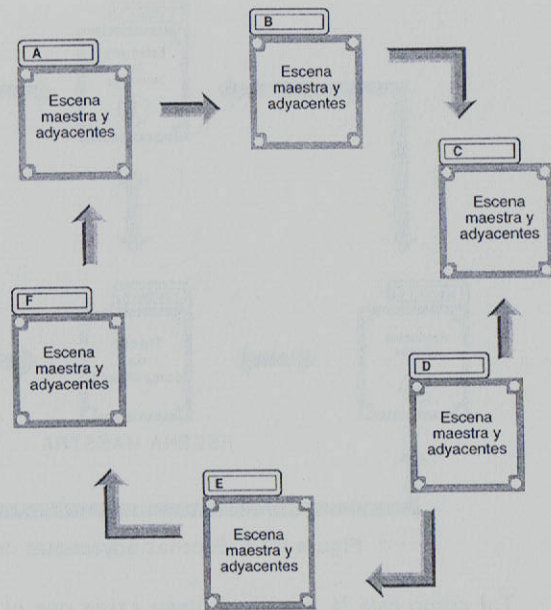


Figura P3.5. Boceto del grafo general de la aplicación.

Elementos del grafo general de escenas

Uno de los objetivos principales del grafo general de la aplicación es reflejar el tránsito entre grupos de escenas. Para detallar con más precisión los posibles movimientos del usuario entre estos grupos, además, se marcan en el grafo los siguientes elementos:

- a) los capítulos.
- b) las escenas.
- c) las puertas globales.
- d) las rutinas.

Algunos de ellos ya han sido introducidos en prácticas anteriores. En la práctica actual definiremos estos conceptos con rigor a fin de situarlos adecuadamente en el grafo general.

Capítulos

Es útil asociar los *capítulos* de una aplicación educativa con las partes correspondientes a las unidades de contenido. Por ejemplo, si se trata de un

curso multimedia de inglés hablaremos del capítulo de los pronombres personales, del capítulo del genitivo sajón, etc.

Si la aplicación no es educativa, los capítulos serán entonces las unidades temáticas. La cuestión es que, siempre que la estructura de la aplicación tiene un diseño por capítulos (grupos de escenas), su construcción es más fácil.

Un capítulo no tiene por qué coincidir con un polígono de viñetas, puede componerse de varios de ellos. La siguiente figura corresponde a un capítulo que consta de dos polígonos:

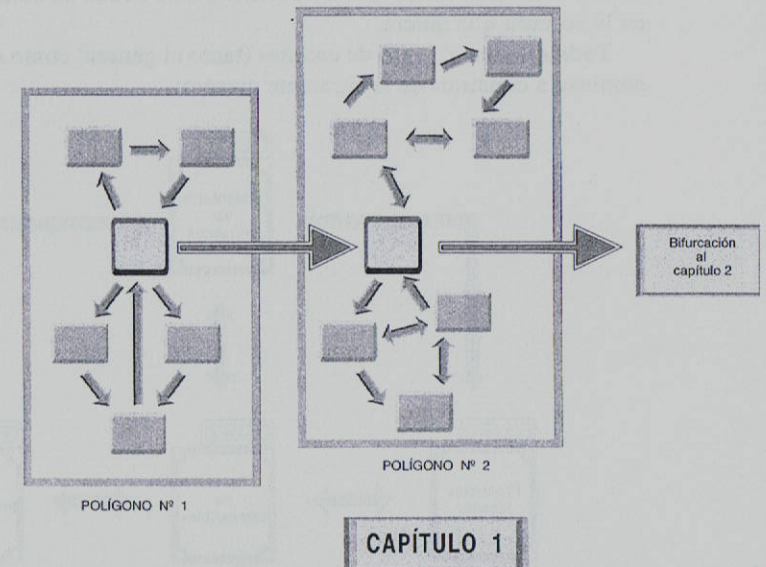


Figura P3.6. Capítulo y polígonos de viñetas.

En este capítulo el usuario navega libremente por un primer grupo de escenas, consigue unos objetivos y puede pasar a un segundo grupo (en el que navega también libremente). Cuando ha conseguido los objetivos de los dos grupos (de los dos polígonos de viñetas) ha completado un capítulo (asociado a una unidad de contenidos) de la aplicación.

Escenas

En las prácticas anteriores las escenas se han definido como elementos unitarios de la aplicación. Constan de un fondo, un lote de tareas de entrada, unas zonas sensibles y otros elementos. Hemos dicho que el fondo de una escena puede cambiar y que muchos de los recursos que se utilizan en ella

(textos, iconos, secuencias...) aparecen y desaparecen, pero no hemos dado un criterio para establecer cuándo es obligatorio cambiar de escena.

Dicho criterio le parecerá de sentido común si usted ha entendido lo que le explicábamos en la práctica del capítulo 1. Necesariamente se cambia de escena cuando cambian las zonas sensibles de la pantalla. Es decir, cuando cambian las respuestas de la aplicación a las acciones del usuario. Si cambian las zonas sensibles, la interacción es totalmente distinta, lo que implica que en el guión se debe empezar una página nueva en la que se describe la nueva escena.

Veamos un ejemplo: tenemos un mapa con diferentes regiones sensibles a las pulsaciones del ratón. Cuando pulsamos sobre una región, se despliega la información que tenemos acerca de ella. ¿Es cada región una escena diferente? Distingamos dos casos:

- Primero: la nueva pantalla consiste en un texto explicativo acompañado de una fotografía y una trama de vídeo.
- Segundo: la nueva pantalla consiste en una fotografía de la región, con nuevas zonas sensibles que, al pulsar encima, despliegan fotografías y textos.

El primer caso no es una escena nueva, se trata de unos elementos que aparecen dentro de la escena principal (la del mapa). El segundo caso sí es una escena diferente, ya que se han cambiado las zonas sensibles de la pantalla.

Puertas

Cuando el salto de una escena a otra es interrumpido por alguna condición u obstáculo a superar, se dice que existe una *puerta*. Las puertas se señalizan en el grafo con el símbolo "//". Si dependen de un objetivo o una condición, estos se marcan a continuación del símbolo.

Por ejemplo, "//\$tiene llave" significa que hay un objetivo consistente en conseguir una llave y que, cuando el usuario la tiene, puede pasar a la siguiente escena.

Hay que distinguir entre las puertas que sirven para pasar de un capítulo a otro de las que enlazan escenas dentro del mismo capítulo. Las primeras son las que se indican en el grafo general de escenas, ya que son las que determinan el flujo global de la historia; se denominan *puertas globales* o *puertas de historia*.

Las segundas son *puertas locales*, ya que determinan el flujo de la aplicación entre escenas pero sin abandonar el mismo capítulo.

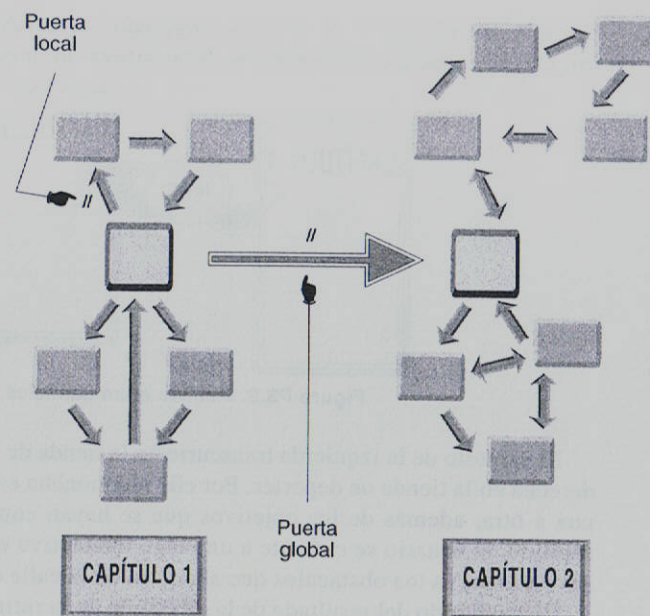


Figura P3.7. Puertas globales y locales.

Cuando se redacta el guión definitivo, en el formato explicado en las prácticas del capítulo primero y en las del segundo, no hay distinción entre puertas globales y puertas locales. Tanto unas como otras se especifican en el guión como condiciones de la escena. No obstante, es necesario marcarlas en los grafos de escenas (general y exhaustivo) para guiar al equipo de producción.

En el grafo general de escenas sólo se marcan las puertas globales, ya que las locales pueden ser muy numerosas. Del mismo modo, sólo se indican en dicho grafo los objetivos que determinan el flujo de la historia (se ignoran los que determinan el comportamiento de las puertas locales). Cuando veamos el *grafo exhaustivo*, introduciremos la notación que permite diferenciar entre un tipo u otro de objetivos y de puertas.

Ejecuciones o rutinas

Las rutinas o ejecuciones², cuando afectan al cambio de capítulo, también deben aparecer en el grafo general de escenas. Se señalan con un nombre envuelto en un círculo para que se puedan identificar a simple vista. Por

²El concepto de rutina o ejecución se ha introducido en las prácticas del capítulo segundo.

ejemplo, en el dibujo de la figura siguiente, el cambio de capítulo no es directo, sino que para realizarlo se debe superar un juego llamado "nave".

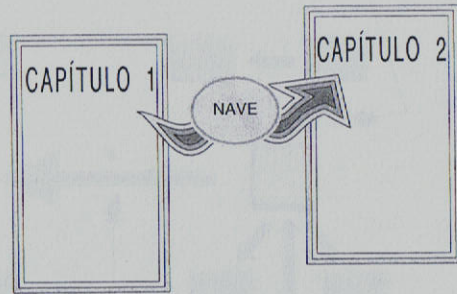


Figura P3.8. Rutinas entre capítulos.

El capítulo de la izquierda transcurre en la tienda de comestibles y el de la derecha en la tienda de deportes. Por ello el guionista establece que para ir de una a otra, además de los objetivos que se hayan conseguido en el primer capítulo, el usuario se enfrenta a un juego interactivo en el que conduce una nave y esquiva los obstáculos que aparecen en la calle de la ciudad futurista.

Dependiendo del resultado de la ejecución de la rutina podemos dirigirnos a diferentes escenas, esto se indicará en el grafo con una flecha. Por ejemplo en la figura siguiente la rutina VIAJE puede llevar a destinos diferentes.

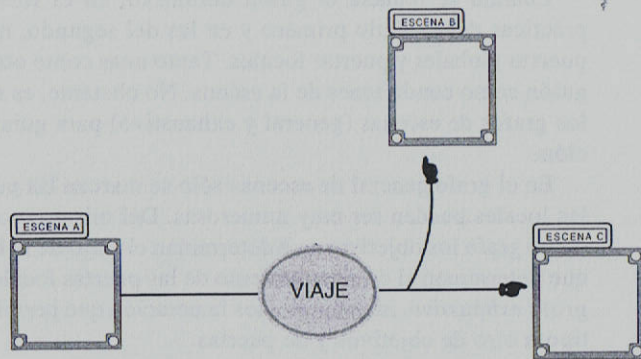


Figura P3.9. Rutina con más de un destino.

Grafo exhaustivo

Cuando al grafo de la figura P3.3 se le añaden las puertas y objetivos locales, es decir, cuando el flujo de la aplicación queda descrito por comple-

to, se dice que tenemos el *grafo exhaustivo* de la aplicación. Este grafo puede ser enorme (una aplicación puede tener cientos de escenas), por lo cual se suele organizar por capítulos. Para ver la ligazón entre éstos últimos, el equipo de producción dispone del grafo general de escenas.

Ahora bien, hemos dicho anteriormente que en el grafo exhaustivo intererará distinguir entre dos clases de objetivos: los que se necesitan para saltar entre escenas de un mismo capítulo y los necesarios para saltar entre diferentes capítulos. Se llaman, de forma análoga que en el caso de las puertas, *objetivos locales* y *objetivos globales*.

El símbolo \$, tal como apuntábamos en la práctica del capítulo segundo, se reserva en el guión para los objetivos. Dado que en el grafo exhaustivo tenemos que distinguir entre objetivos locales y globales, se señalizan con *L\$* los primeros y con *H\$* los segundos. La letra *L* indica que se trata de un objetivo local del capítulo, la letra *H* que se trata de un objetivo de la aplicación. Es decir, la *H* nos indica que el objetivo afecta a alguna puerta que permite cambiar de capítulo.

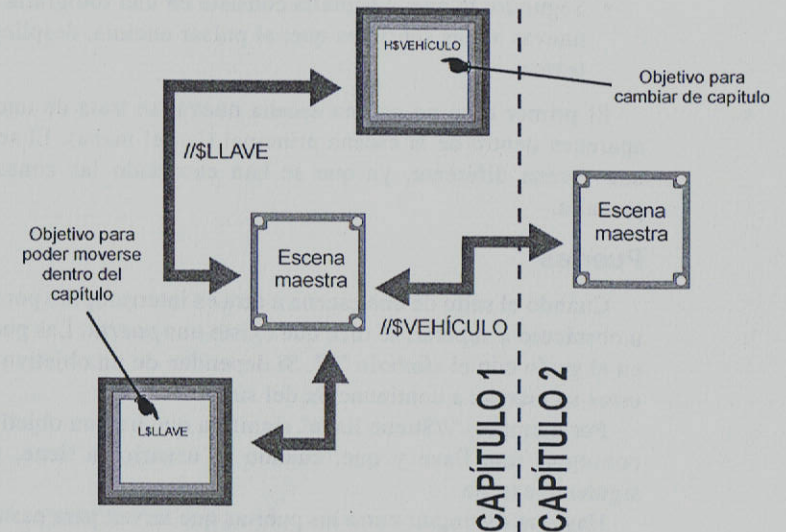
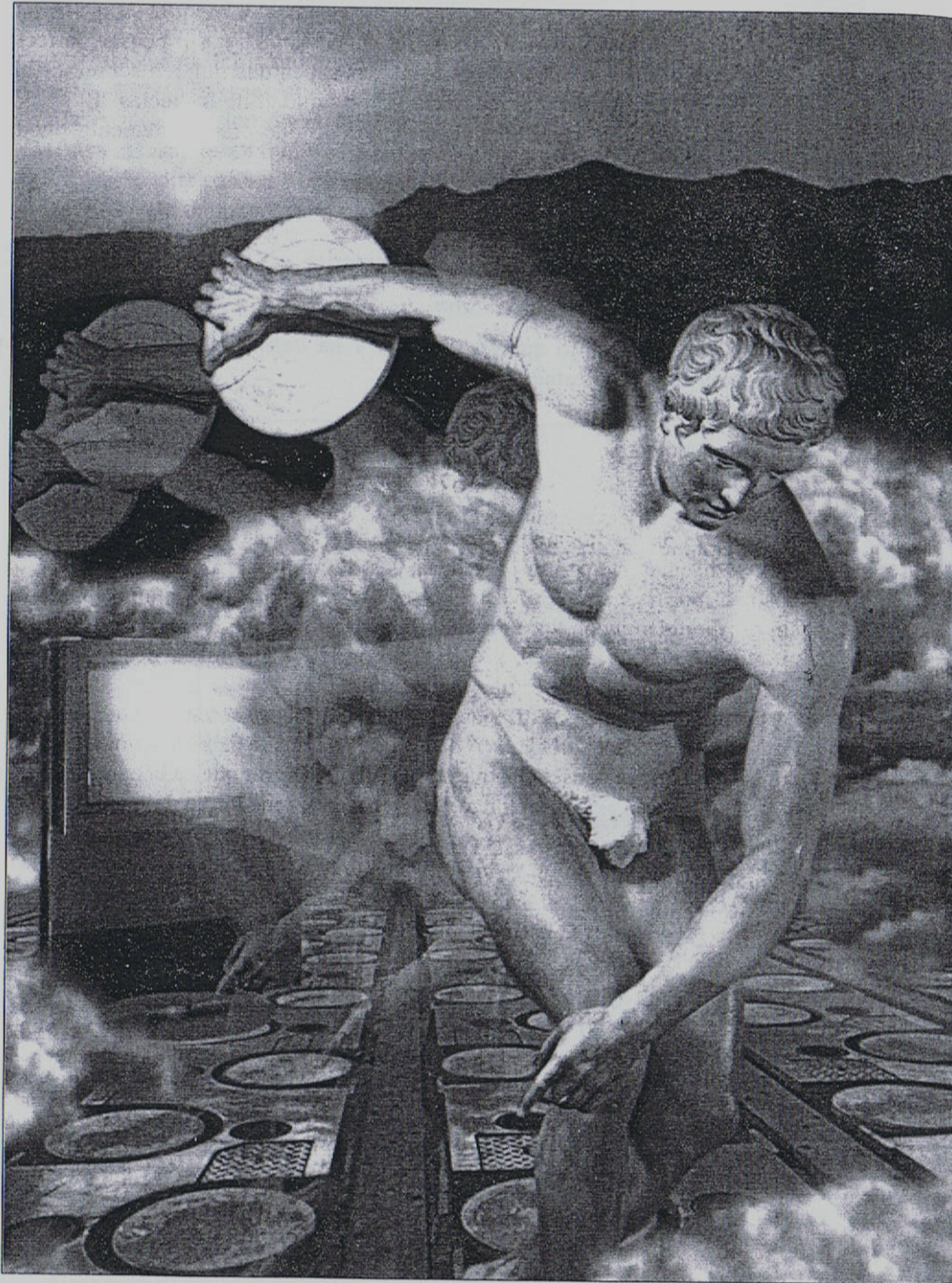


Figura P3.10. Objetivos locales y globales.

Prueba de fondos

Esta prueba es una de las tareas concretas en las que el equipo de producción utiliza el grafo exhaustivo. Se trata, en definitiva, de un control de estructura que se hace antes de empezar a montar las escenas. Para realizarlo,



Práctica 4

Diseño de las hojas de configuración

Objetivo de la práctica

En el capítulo cuatro le hemos explicado las reglas para diseñar pantallas y colocar los elementos interactivos en las mismas. Algunos de estos elementos, al igual que ocurre con las zonas¹ que se hayan establecido, serán comunes a más de una escena. Es de esperar, por ejemplo, que un marcador que indique los puntos obtenidos por el usuario ocupe un lugar fijo en la pantalla, a lo largo de toda la aplicación.

Nuestras aplicaciones, por tanto, tienen toda una serie de elementos que llamamos *estables*, es decir, que se comportan de la misma manera y aparecen en los mismos lugares aunque la escena sea diferente. No tiene sentido que el guión describa estos elementos estables en cada escena, puesto que repetiríamos siempre las mismas indicaciones. Por ello, toda la descripción de la parte estable de una aplicación se redacta en un pliego de hojas que se denominan *hojas de configuración* de la aplicación.

Además de estos elementos, hay otros componentes o comportamientos de la aplicación que afectan a más de una escena. Por ejemplo, la forma en que

¹ La definición de *zona* de pantalla se da en el capítulo 4.

se felicita al usuario cuando consigue un objetivo puede ser siempre la misma a lo largo de los diferentes capítulos de la aplicación. Para recoger todas estas informaciones, en las hojas de configuración encontramos las descripciones relativas a tres tipos de datos de distinta naturaleza:

- a) elementos estables de las pantallas.
- b) patrones de comportamiento de la aplicación.
- c) tablas de control de la misma.

Los elementos estables son aquellos componentes de las pantallas que aparecen en más de una escena. Las hojas de configuración, en lo referente a ellos, suministran información sobre sus propiedades, funciones que realizan, relación con otros elementos, etc.

Los patrones de comportamiento son aquellos sucesos que se dan frecuentemente en la aplicación y que es necesario describir. Interesa especificar claramente cuándo suceden y qué serie de eventos ocurren, a fin de facilitar la tarea del equipo de producción.

Las tablas de control son cuadros orientativos que establecen el comportamiento de la aplicación. Nos informan de cuestiones como, por ejemplo, si el elemento estable X no debe aparecer en una escena concreta o si el personaje Y de la aplicación tiene o no ciertas cualidades. También, en el caso de las aplicaciones narrativas, nos recuerdan qué sucesos de un cierto capítulo tendrán repercusión más adelante en el desarrollo del argumento.

Enunciado de la práctica

La práctica que realizaremos consistirá en introducir ciertos elementos estables en una aplicación, redactar algunos patrones de comportamiento y elaborar unas tablas de control. Usaremos para ello la misma aplicación ficticia que nos ha servido de base para las prácticas de los capítulos uno y dos.

Elementos estables

Introduciremos en la pantalla de la casa de campo los siguientes elementos:

- a) Un marcador.

Informará de los puntos obtenidos por el usuario.

- b) Un marco exterior.

Tendrá la función de alojar los elementos externos a la acción que transcurre en la parte interior de la pantalla.

- c) Un punto de información de objetivos.

Indicará al usuario los objetivos que debe conseguir en cada capítulo.

- d) Un botón de vuelta atrás.

Servirá para retroceder a la pantalla anterior.

- e) Un indicador de acierto/error.

Indicará al usuario que ha acertado y ha conseguido puntos o que ha fallado y se le han restado.

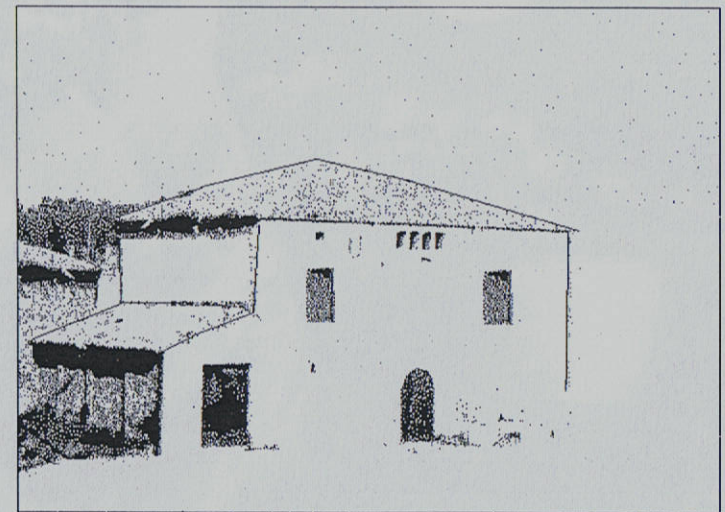


Figura P4.1. Fondo de base para las prácticas primera y segunda.

Patrones de comportamiento

Describiremos los siguientes comportamientos de la aplicación:

- a) Comportamiento de las zonas sensibles.
- b) Indicación al usuario de que no se puede pasar una *puerta*.
- c) Indicación de progreso al usuario.

parte de la tarea del guionista es convertir esta información descriptiva a un lenguaje no ambiguo para el equipo de producción.

En el ejemplo que realizaremos vamos a construir el guión de la siguiente escena:

Se llega a una casa de campo de dos plantas. Se ve desde el exterior. El usuario tiene que conseguir la cámara fotográfica que se halla en el interior de la casa. Dicha cámara está guardada en la habitación de Rosa, en el piso de arriba.

El usuario hará clic sobre las ventanas de la casa y se le indicará que no puede llegar a ellas. Para lograrlo, deberá descubrir una escalera que se encuentra haciendo clic sobre la puerta del almacén.

Cuando tenga la escalera, tendrá acceso a las ventanas del piso superior y, entonces, será cuestión de acertar la ventana adecuada.

La casa, además, tiene una puerta central. Cuando hagan clic sobre ella indicará que no se puede entrar e insinuará que se utilicen las ventanas.

Para entender mejor este enunciado, vea la siguiente figura:

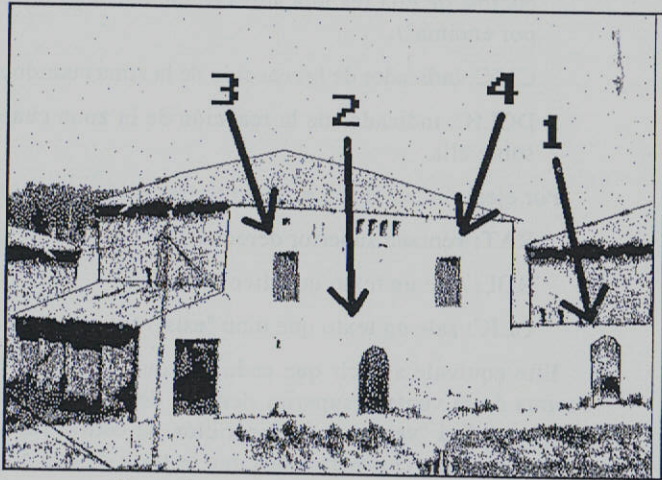


Figura P1.1. Imagen de fondo de la escena.

En ella puede observar los cuatro lugares importantes que determinan la interacción cuando el usuario entra en la escena. Vamos a comentarlos uno por uno:

1. Puerta del almacén.

Es donde se guarda la escalera. El usuario deberá descubrirla para poder acceder a las ventanas superiores.

2. Puerta central.

Es un distractor. Es decir, su presencia no es imprescindible para que el usuario resuelva el problema que le plantea la escena.

Los distractores son objetos que se añaden para que el juego sea variado. Si un usuario en esta escena no descubre el distractor no se altera el desarrollo de la historia.

3. Ventana superior izquierda.

Es otro distractor. No obstante, la diferencia con el anterior es que este tiene dos tipos de respuesta según se tenga la escalera o no.

4. Ventana superior derecha.

Es donde se guarda la cámara fotográfica. También tiene dos respuestas diferentes para el usuario dependiendo de si éste tiene o no la escalera.

El guionista tiene en mente la imagen de la figura P1.1 a la hora de escribir el guión. Muchas veces dispone de una fotografía que ya ha elegido para esta escena. En otros casos, esboza un dibujo o un esquema. Por ejemplo, puede suceder que diga "conseguirme una fotografía de una casa de dos plantas con dos puertas abajo y dos ventanas arriba".

No obstante, si lo hace así, cuando quiera posteriormente añadir distractores a la escena deberá solicitar que le dejen ver la imagen y ello retrasará el trabajo. A veces, esto se resuelve en los equipos de producción dejando que un segundo equipo de guionaje se encargue de añadir distractores, mientras que un primer equipo o persona es el responsable del guión inicial.

Elementos del guión para definir la escena

La información textual es complicada aunque se acompañe del dibujo de la figura P1.1. Por ello, el guionista procede a redactarla respetando unos convenios y palabras clave que le permitirán ser más preciso y ahorrar tiempo a la hora de comunicarse con el equipo de producción.

Estas palabras clave recogen lo que se llaman "elementos" de la escena. El guión definitivo, por tanto, debe ser redactado en función de estos elementos. Veamos los que encontramos en la escena que hemos tomado como ejercicio:

Título

Las escenas normalmente se identifican por un título y un número. Ambos son importantes ya que se emplean también para referenciar otros comentarios y aclaraciones sobre cómo debe ser la escena.

En el caso que nos ocupa, la escena bien podría ser la número 12 de un capítulo de la aplicación. Podríamos ponerle el nombre "rural" o "hacienda". Supongamos que nos decidimos por el segundo. Con ello, si quisieramos poner comentarios en otros documentos anexos al guión, sabríamos que todos los que llevasen el nombre "hacienda" harían referencia a nuestra escena.

Por ejemplo, en el guión podría aparecer la nota "para una descripción textual, véase *hacienda.doc*". Con ello el guionista quiere indicar que hay un documento anexo al guión en el cual se guarda el enunciado de la escena, es decir, el texto que usted ha podido leer en el apartado anterior.

Fondos

Los *fondos* son las imágenes (normalmente fotografías o pantallas dibujadas por un grafista) que sirven de imagen de base para una escena. En este caso, el fondo sería la fotografía de una casa que correspondiera a la descripción.

Los fondos se señalan en los guiones con la palabra BMP, que es la extensión habitual de los ficheros de imágenes de windows (bitmaps). Por ello es muy habitual entre los equipos de producción oír frases como "¿Qué *bemepé* ponemos en esta escena?" o bien "¿Cambiad el *bemepé* de la casa por este otro que acaban de traer!".

Por tanto, el guionista escribiría al principio de la escena lo siguiente:

BMP: casacamp

Con ello quiere indicar que hay una imagen que servirá de fondo y que se ha guardado (o se guardará) en el disco con el nombre "casacamp".

Hay guiones que especifican incluso el formato gráfico de la imagen. Los formatos gráficos son simplemente formas diferentes de guardar en el disco las imágenes digitalizadas. Estas formas de guardar las imágenes se indican en la extensión del nombre del fichero¹, por lo que suelen aparecer tres letras tales como PCX, TIF, GIF, BMP, etc.

¹ En el sistema operativo MS-DOS los nombres de los ficheros que se graban en el disco constan de once letras como máximo. Las ocho primeras son el nombre que le da el usuario, las tres últimas se llaman "extensión" y son una especificación de lo que contiene el fichero. Las ocho primeras letras se separan de la extensión por un punto. Por ejemplo, en un fichero llamado "historia.txt" la extensión ".txt" indica normalmente que es un texto sin formato.

Por ello, si en un guión se encuentra con la anotación:

BMP: casacamp.pcx

quiere decir probablemente que se ha escaneado la fotografía de una casa de campo y se ha guardado en formato PCX.

Zonas sensibles

Las *zonas sensibles* son aquellas regiones de la pantalla en las que sucede algo al pulsar o pasar con el ratón por encima de ellas. En el ejemplo que hemos puesto, todas las ventanas serían *zonas sensibles*.

En el guión se explica siempre sobre cada fondo el inventario de zonas sensibles y una descripción de su comportamiento. Este "comportamiento" es lo que sucede cuando el usuario pasa por encima con el ratón, pulsa sobre la zona o pulsa dos veces sobre la zona (doble clic).

Para simplificar el guión, se suelen utilizar las siguientes palabras claves:

RAT: indicador de zona sensible (es una abreviatura de "ratón", queriendo decir que es una zona que reacciona a las órdenes del ratón).

ROL: indicador de la reacción de la zona cuando el ratón pasa por encima de ella (es una abreviación del inglés "roller", es decir, "rodar por encima").

CLIC: indicador de la reacción de la zona cuando se hace clic sobre ella.

DCLIC: indicador de la reacción de la zona cuando se hace doble clic sobre ella.

Por ejemplo, en el guión encontraríamos lo siguiente:

RAT: ventana superior derecha.

ROL: sale un texto que dice "ventana".

CLIC: sale un texto que dice "esta ventana está cerrada".

Ello equivale a decir que cada vez que el usuario pase con el ratón por encima de la ventana superior derecha de la imagen de fondo aparecerá un texto que dirá "ventana". La aparición de este texto breve normalmente se realiza para que el usuario sepa que se trata de una zona sensible y sepa que puede pulsar sobre ella.

Cuando el usuario pulse el ratón, aparecerá un texto más grande que dirá "esta ventana está cerrada".

Es práctico también numerar las zonas sensibles, ya que permite distinguir entre aquellas con una descripción similar.

Iconos

Por *iconos* entendemos figuras que se superponen sobre un fondo. Suelen ser dibujos que temporalmente aparecen y después desaparecen. En el ejemplo que estamos realizando la escalera sería un icono. Se señalan con la palabra clave ICN.

Refiriéndose a la puerta correcta, en el guión encontraríamos:

RAT: puerta del almacén.

ROL: sale un texto que dice "puerta".

CLIC: aparece el texto "En el almacén hay una escalera" y sale el dibujo correspondiente (ICN "escalera").

Con el último paréntesis, el guionista indica que no hay que sustituir el fondo (fotografía de la casa) por ninguna otra imagen, sino que aparecerá un pequeño dibujo sobre dicha imagen y luego desaparecerá. Todo ello lo indica con la palabra ICN. A continuación, escribe "escalera" porque quiere decir que esta imagen se hallará grabada en el disco duro con este nombre.

Objetivos

Los *objetivos* son elementos concretos que el usuario debe descubrir. Se señalizan con el símbolo "\$" seguido normalmente de un número o un nombre. Por ejemplo, en la escena tenemos dos objetivos: la escalera y la cámara.

Por ello, el fragmento anterior del guión debe reescribirse de la siguiente manera:

RAT: puerta del almacén.

ROL: sale un texto que dice "puerta".

CLIC: aparece el texto "En el almacén hay una escalera" y sale el dibujo correspondiente (ICN "escalera"). \$1.

Con la anotación añadida "\$1" se indica que el usuario ha conseguido el objetivo número 1. También podría escribirse "\$escalera" o "\$acceso", ya que el haber conseguido el objetivo permitirá a partir de ahora acceder a las ventanas del piso superior.

Redacción del guión: un primer borrador

Con los elementos que hemos explicado usted ya está en condiciones de entender un primer borrador del guión de esta escena. Todavía le faltan algunos conceptos para enfrentarse a un guión completo. Estos conceptos

serán introducidos en la próxima práctica, después del capítulo 2. De este modo, un primer borrador del guión que se corresponde al enunciado del apartado "Enunciado de la práctica" sería el siguiente:

Escena número: 12.

Título: hacienda.

Observaciones: el desarrollo textual se halla en el documento anexo "hacienda.doc".

BMP: casacam.pcx.

Zonas sensibles:

1) RAT: puerta del almacén.

ROL: sale un texto que dice "puerta del almacén".

CLIC: aparece el texto "En el almacén hay una escalera" y sale el dibujo correspondiente (ICN "escalera").\$1.

2) RAT: puerta central.

ROL: sale un texto que dice "puerta de entrada".

CLIC: aparece el texto "La casa está cerrada. Deberás probar a entrar por una ventana".

3) RAT: ventana superior izquierda..

ROL: sale un texto que dice "ventana".

CLIC: existirían dos posibilidades.

- Si no se tiene \$1, aparece el texto "No puedes subir a esta ventana. Está muy arriba".

- Si se tiene \$1, aparece el texto "En esta ventana no hay nada de interés".

4) RAT: ventana superior derecha.

ROL: sale un texto que dice "ventana"

CLIC: existirían dos posibilidades.

- Si no se tiene \$1, aparece el texto "No puedes subir a esta ventana. Está muy arriba."

- Si se tiene \$1, aparece el texto "Esta ventana es de la habitación de Rosa. A través de ella se puede ver la cámara que estamos buscando".

Comentarios generales

Nótese la señalización de los objetivos en los RAT números 1 y 4. En el primero se indica que se ha conseguido el objetivo número 1, la escalera. En el segundo, que se ha conseguido el objetivo número 2; la cámara.

A lo largo de las prácticas del libro veremos otros elementos que son necesarios para hacer guiones. También veremos una formulación más completa de los que aquí le hemos presentado, en especial de las zonas sensibles.

Note, por ejemplo, que la formulación de éstas es todavía engorrosa. Por ejemplo, no será necesario siempre escribir "sale un texto que..." para indicar que aparece un texto en pantalla. También podrá formularse de forma más eficiente el comportamiento de la zona cuando depende de una condición (en nuestro ejemplo, haber conseguido o no la escalera).

Comentarios sobre las zonas sensibles

De este primer contacto, ya podrá adivinar que las zonas sensibles son sumamente importantes en las pantallas, ya que a través de ellas se planifica la interacción y, por tanto, se determina el discurso de la historia.

Los programas de producción multimedia permiten al usuario marcar zonas sensibles en la pantalla. Normalmente permiten encuadrar los objetos con rectángulos que se hacen transparentes.

Estas zonas pueden ser rectangulares o de forma poligonal, de forma que se ajusten mejor al dibujo que tienen debajo. Algunos lenguajes ofrecen la forma poligonal como ampliación, es decir, permiten que marquemos zonas de forma irregular y detectan cuándo el ratón entra en ellas o pulsa encima. De todas formas, esto es sólo estrictamente necesario en algunas aplicaciones (formas de mapas, diagnóstico médico, etc.).

Para una historia normal, pueden ser rectangulares y no pasa nada. El criterio es altas y delgadas: porque el usuario barre en sentido horizontal, por ello tiene que encontrársela (alta) pero no aguantar el barrido demasiado tiempo, porque despista.

Esquema gráfico de la escena

Es habitual acompañar el guión de gráficos explicativos para que la comunicación sea más eficiente y para evitar errores. En caso de que el guionista

tenga ya la imagen definitiva que sirva de fondo a la escena adjuntará el dibujo de la siguiente figura:

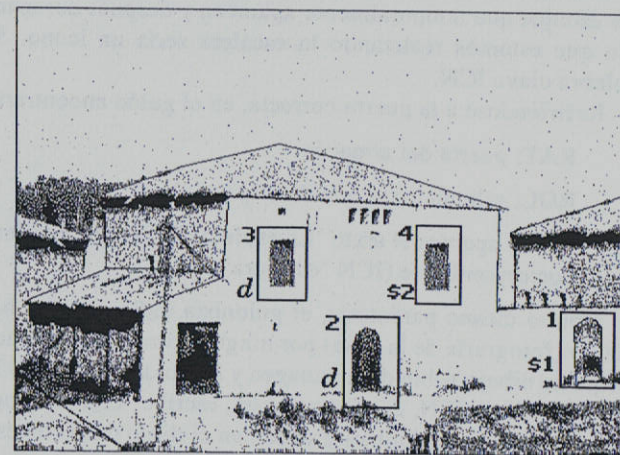


Figura P1.2. Esquema gráfico de la escena.

En él puede observarse cómo vienen indicadas las zonas sensibles. También se indican cuáles corresponden a objetivos y cuáles son distractores. Normalmente los distractores también se indican en el guión, aunque en el ejemplo no los hemos puesto todavía para no atosigar al lector. Por tanto, la última corrección al borrador del guión que haríamos sería añadir la letra "d" entre paréntesis detrás de los RAT números 2 y 3. Así que en el guión definitivo, las líneas correspondientes a estas zonas quedarían de la siguiente manera:

- 2) RAT: puerta central (d).
- 3) RAT: ventana superior izquierda (d).