

En el fondo, los modelos

Paul Gerrard, gurú de los sistemas de pruebas y consultor, analiza en una serie de artículos una gran variedad de cuestiones relacionadas con las comprobaciones. Los modelos de prueba son fundamentales para los ensayos y, en este artículo, Paul habla sobre el arte de crear y utilizar modelos. Si los probadores tienden a “resolver en origen” de la mano de los desarrolladores o, al menos, colaborando más estrechamente con ellos, ambos (probadores y desarrolladores) tendrán que crear modelos, aprender a articularlos y compartirlos, y fomentar una mayor colaboración.

Paul Gerrard
Gerrard Consulting

Con el patrocinio de



El diseño de pruebas se basa en modelos

En este artículo, quiero hablar sobre el concepto más importante en las pruebas: la técnica (¿o ciencia?) de la creación y el uso de modelos. En 1990, Boris Beizer dijo¹:

“Comprobar es un proceso en el cual creamos modelos mentales del entorno, el programa, la naturaleza humana y las propias pruebas. Cada modelo se utiliza hasta que aceptamos que el comportamiento es correcto o hasta que el modelo deje de ser suficiente para el propósito buscado”.

El diseño de pruebas es el proceso por el cual seleccionamos, a partir del infinito número posible de ellas, las que creemos que serán más valiosas para nosotros y para las partes interesadas. Nuestro modelo de prueba nos ayuda a seleccionar las pruebas de una manera sistemática. Los modelos de prueba son fundamentales para las pruebas y el resto de este artículo describe su importancia y su uso.

Proceso de prueba fundamental

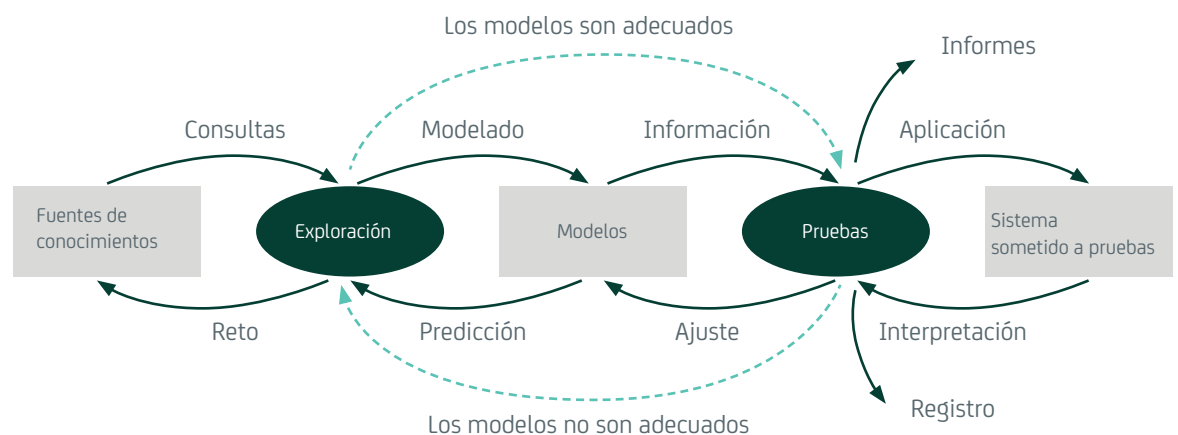
En A New Model for Testing², sugiero que, en el nivel más básico, todos los ensayos pueden describirse en los términos siguientes:

1. Identificamos y exploramos las fuentes de conocimiento para construir modelos de prueba.
2. Utilizamos estos modelos para contrastar y validar las fuentes.
3. Utilizamos estos modelos para informar sobre las pruebas (y su desarrollo).

Marco una distinción entre exploración y pruebas. La principal diferencia con respecto al punto de vista común sobre la realización de pruebas es que utilizaré el término exploración para referirme a la obtención de conocimiento sobre el sistema a comprobar a partir de fuentes de conocimiento.

Ilustración A

Nuevo modelo para las pruebas: los modelos en el corazón de los procesos de pensamiento del probador



Fuentes de conocimiento

Construimos nuestros modelos a partir de información que recopilamos en fuentes de conocimiento. Una vez recibida la misión de comprobar, nuestra primera tarea es identificar estas fuentes. Esas fuentes de conocimiento pueden ser:

- **Documentación:** especificaciones, diseños, requisitos, estándares, directrices y pautas, etc.
- **Personas:** partes interesadas, usuarios, analistas, diseñadores y desarrolladores, entre otras
- **Experiencia:** su propio conocimiento y experiencia de sistemas similares (o diferentes), sus preferencias, prejuicios, conjeturas, intuiciones, creencias y prejuicios
- **Sistema:** el sistema bajo prueba, si es que existe, está disponible y es accesible

Obtenemos información de nuestras fuentes de conocimiento para generar modelos que nos sirven para contrastar nuestras fuentes y diseñar y/o comprobar nuestros sistemas.

Todas nuestras fuentes de conocimiento son susceptibles de error e incompletas y, lo mismo que nuestros modelos. Los probadores utilizan la experiencia, la habilidad y el sentido común para tamizar esas fuentes, compararlas, contrastarlas, y confirmarlas, para llegar a un consenso. Generalmente, estas capacidades están asociadas a los sistemas o analistas de negocios, por supuesto.

¿Qué es un modelo de pruebas?

Un modelo de pruebas podría ser una lista de comprobación o un conjunto de criterios. También podría ser un diagrama derivado de un documento de diseño o un análisis de un relato. Muchos modelos de prueba no se reflejan en el papel, pueden ser modelos mentales contruidos específicamente para guiar al probador mientras explora el sistema objeto de la prueba.

Utilizamos modelos de prueba para:

- Simplificar el contexto de la prueba. Los detalles irrelevantes o insignificantes son ignorados en el modelo.
- Centrar la atención en un aspecto particular del comportamiento del sistema. Podrían ser características cruciales o de riesgo, aspectos técnicos u operaciones de interés del usuario, o elementos concretos de la construcción o la arquitectura del sistema.
- Generar un conjunto de pruebas únicas (en el contexto del modelo) que sean diversas (con respecto a ese modelo).
- Permitir el cálculo, la planificación, la monitorización y la evaluación de la integridad (cobertura) de las pruebas.

Desde el punto de vista del probador, un modelo nos ayuda a reconocer los aspectos particulares del sistema que podrían ser objeto de una prueba. El modelo centra su atención en las áreas del sistema que son de interés.

Ejemplo

Supongamos que queremos probar cómo un coche (un modelo de cambio de marchas automático) acelera desde el reposo hasta su velocidad máxima y confirmar si cumple nuestro objetivo de rendimiento (por ejemplo, desde el arranque hasta los 90 km/h en 8 segundos). El modelo de este sistema podría ser:

1. Una masa (todo el vehículo más el conductor) que actúa con un centro de gravedad definido, y que acelera de acuerdo con la segunda ley de Newton
2. Una fuente de energía (el motor) que tiene una potencia de salida que varía desde un mínimo hasta un valor máximo según la posición del acelerador
3. Un acelerador que puede tener un posición variable
4. Fórmulas que relacionan la posición del acelerador, la potencia y la aceleración

Es probable que obtengamos toda la información que necesitamos para nuestro modelo en el documento de diseño del coche.

Utilizando el modelo, se podría diseñar una prueba como esta: “A partir del reposo, presionar el pedal hasta la máxima potencia durante un período de diez segundos. Utilizar nuestras fórmulas para calcular una velocidad prevista por cada segundo de la aceleración. Comparar la velocidad real con la prevista cada segundo de la prueba.”

Cuando llevamos a cabo la prueba en un coche real, se compara su velocidad a cada segundo con la prevista por el modelo. De esta manera, podríamos determinar si el coche cumple con su objetivo de rentabilidad. Por ejemplo, si el sistema objeto de la prueba (el coche) no llega a la velocidad requerida en cada punto de medición, entonces o cambiamos el coche o cambiamos el modelo (nuestra interpretación del diseño del coche), o el propio diseño.

Todo esto pinta bien, ¿verdad?

Los modelos sobresimplifican las cosas, utilice más de uno

En la prueba real, sin embargo, nuestro coche puede no comportarse como se espera, debido a que nuestro modelo omite varios aspectos clave del comportamiento del coche y del contexto. Podríamos preguntarnos, y con razón:

- ¿Sería un verdadero conductor tan agresivo o más suave con el acelerador?
- ¿Cuál es la velocidad y la dirección del viento?
- ¿Cuál es el estado de la carretera (húmeda, seca, asfalto, tierra, etc.)?
- ¿Qué carga lleva el vehículo, además del conductor?
- ¿Está el coche en una carretera llana, o en una pendiente cuesta arriba o cuesta abajo?
- ¿Cuál es la eficiencia energética del sistema?

Nuestro modelo está enormemente simplificado, incorpora muchas suposiciones implícitas y necesitaría un ajuste significativo para ser una representación exacta de un coche real en una prueba. Todos los modelos simplifican el contexto de las pruebas en mayor o menor grado, por lo que normalmente se utilizan varios modelos para ampliar nuestra visión y cobertura (conocidos como “medidas parciales diversas³”). El reto estriba en seleccionar los modelos que sean una representación bastante exacta del sistema objeto de nuestra prueba e interpretar cuidadosamente los resultados obtenidos.

En general, todos los modelos de prueba, incluso los propuestos por los libros de texto, son heurísticos en el sentido de que son útiles en algunas situaciones, pero siempre incompletos y falibles. Antes de adoptar uno, hemos de saber qué aspectos del comportamiento, diseño, modos de fallo o patrones de uso nos ayudan a identificar ese modelo, y qué supuestos y simplificaciones (explícita o implícitamente) incluye.

Nuestros cerebros son motores de modelado fantásticos

Nuestros cerebros son capaces de modelar y remodelar nuestro entorno. Para lograr algo tan “simple” como caminar, nuestro cerebro tiene que entender, segundo a segundo (y más rápido que eso) la configuración de los principales huesos de nuestro cuerpo. Tiene que entender las tensiones que afectan a unos cien músculos. Para dar un solo paso, nuestro cerebro necesita enviar impulsos a todos estos músculos para que trabajen según patrones de gran complejidad que nos permiten avanzar y llegar a nuestros destinos sin chocar con las cosas. Sencillamente, nuestro cerebro no es suficientemente rápido como para procesar toda esta información. Utiliza modelos para simplificar y gestionar el desafío. El modelado y la visualización son habilidades innatas, esenciales en nuestras vidas diarias. En comparación, los MEJORES robots del planeta están aún muy verdes.

Es posible que haya visto a los golfistas practicar su swing antes de un golpe. Ensayan y visualizan el golpe, la trayectoria de la pelota y el objetivo. En muchos deportes, los entrenadores graban en vídeo a los atletas y les hablan sobre sus movimientos con gran detalle para ayudarles a visualizar y controlar sus movimientos, a menudo bajo gran tensión física. Los atletas modelan conscientemente su mundo para alcanzar la perfección o el control, lo que algunos llaman “la zona”.

Pero los modelos no solo representan el movimiento físico. Por ejemplo, cuando Stephen Hawking perdió capacidades físicas, inventó una colección de potentes herramientas mentales —modelos— que le permitieron continuar trabajando con su físico, sin necesidad de utilizar pizarras ni fórmulas escritas.

Utilizamos esta misma habilidad modeladora para desarrollar y probar sistemas. Nuestros cerebros son motores de modelado increíblemente sofisticados y rápidos, y el modelado mental domina nuestro pensamiento.

Los modelos en el corazón de las pruebas

Sugiero² que la exploración y modelado del desarrollador y el probador son realmente muy similares. Si los probadores tienden a “resolver en origen” de la mano de los desarrolladores o, al menos, colaborando más estrechamente con ellos, ambos (probadores y desarrolladores) tendrán que crear modelos, aprender a articularlos y compartirlos, y fomentar una mayor colaboración.

En un blog⁴, “Courage and Ambition in Teaching and Learning” (El valor y la ambición en la enseñanza y el aprendizaje), sugiero que la forma en que enseñamos los diseños de pruebas debe replantearse. Las técnicas de diseño de pruebas se enseñan generalmente como si fueran procedimientos administrativos. Tenemos que enseñar a los probadores a crear modelos, a apreciarlos, a seleccionarlos y a desecharlos. Tenemos que enseñar los modelos en los que se basan las técnicas, no solo los procedimientos para aplicarlas.

El pensamiento y el modelado están en el fondo de lo que hacen los probadores. La escala, la complejidad y las tasas de cambio en los sistemas digitales y en Internet de las cosas están en ascenso. Si queremos hacer frente a esos desafíos, nuestras prioridades han de ser una nueva forma de pensar y adquirir y perfeccionar las habilidades de modelado.

Acerca del autor

Paul Gerrard es consultor, profesor, escritor, *webmaster*, desarrollador, probador, conferenciante, entrenador de remo y editor. Ha realizado tareas de consultoría en todos los aspectos de las pruebas del *software* y el control de calidad, y se ha especializado en el control de las pruebas. Ha presentado tutoriales y charlas destacados en congresos sobre pruebas en toda Europa, EE. UU., Australia, Sudáfrica y, en ocasiones, ha sido galardonado por ello.

Formado en la Universidad de Oxford y el Imperial College de Londres, Paul ganó en 2010 el premio Eurostar European Testing Excellence Award y, en 2013, fue galardonado con el European Software Testing Award (TESTA) en reconocimiento de su carrera profesional.

En 2002, Paul escribió el libro *Risk-Based E-Business Testing* con Neil Thompson. Posteriormente, escribió *The Tester's Pocketbook* en 2009. Paul fue coautor del libro *The Business Story Pocketbook* con Susan Windsor en 2011 y escribió *Lean Python* en 2014.

En 2014, Paul fue el Director de Programa del congreso EuroSTAR Conference que tuvo lugar en Dublín.

Es el Director de Gerrard Consulting Limited, de TestOpera Limited y anfitrión del Test Management Forum.

Correo electrónico: paul@gerrardconsulting.com

Twitter: [@paul_gerrard](https://twitter.com/paul_gerrard) Página

web: gerrardconsulting.com

Para obtener más información, visite **Desarrollo y pruebas** con CA Technologies.



Comuníquese con CA Technologies en ca.com/es



CA Technologies (NASDAQ: CA) crea software que impulsa la transformación de las empresas y les permite aprovechar las oportunidades que brinda la economía de las aplicaciones. El software se encuentra en el corazón de cada empresa, sea cual sea su sector. Desde la planificación hasta la gestión y la seguridad, pasando por el desarrollo, CA colabora con empresas de todo el mundo para cambiar la forma en que vivimos, realizamos transacciones y nos comunicamos, ya sea a través de la nube pública, la nube privada, plataformas móviles, entornos de mainframe y entornos distribuidos. Para obtener más información, visite ca.com/es.

Referencias

1. "Software Testing Techniques", Boris Beizer, 1990
2. "A New Model for Testing", Paul Gerrard, <http://dev.sp.qa/download/newModel>
3. "Lessons Learned in Software Testing", Kaner, Bach, Pettichord, 2002