ACTUALIZACIÓN INFORMÁTICA

CPAP

CURSO DE POSGRADO

Técnicas y Gestión de las pruebas de software

María Elisa Presto

DOCENTE

Tipos de pruebas

Estáticas

Dinámicas

- Funcional
- No funcional / de características
- Estructural

Vinculados a cambios

ACTUALIZACIÓN INFORMÁTICA

CPAP

TEMA

Pruebas estáticas

ACTUALIZACIÓN INFORMÁTICA
CPAP

Pruebas estáticas

No se ejecuta el código a ser probado

- Mejoran la calidad de código y documentación
- Detectar errores de forma temprana
- Pueden realizarse antes de que el sistema esté en funcionamiento

Revisión de diferentes tipos de documentos

- Técnicas de Revisión: informal, recorrido guiado, de a pares
- Inspección: roles definidos, checklists ->informes
- Auditoría: evaluación formal externa (cumplimiento con normas, estándares o reglamentaciones)
- Análisis técnico: Grupo de expertos técnicos y del negocio

ACTUALIZACIÓN INFORMÁTICA
CPAP

Pruebas estáticas

Análisis estático de código

- **Verificación** de conformidad con especificaciones y normas
- Uso manual a través de lecturas
- Uso automático utilizando herramientas de Análisis Estático
  - Una herramienta como SonarQube (SonarQube Community Edition (Gratuita))

Pruebas estáticas - Ventajas

Permiten detección muy temprana de errores

También permiten eliminación de causas de error potencial

… y la mejora global de la calidad del software

Permiten anticipar y mejorar el Plan de Pruebas

- Conocer la complejidad, los riesgos, los objetivos, la arquitectura …

Reducir el costo total y la calidad

Mejora la calidad interna del código - mantenibilidad

ACTUALIZACIÓN INFORMÁTICA

CPAP

TEMA

Pruebas dinámicas

ACTUALIZACIÓN INFORMÁTICA

CPAP

Objetivos

- Definir la noción de Pruebas Funcionales
- Presentar las diferentes técnicas
- Profundizar en algunas técnicas

ACTUALIZACIÓN INFORMÁTICA

CPAP

Índice

Pruebas funcionales

- Pruebas basadas en "**qué hace**" el sistema (SUT)
- Deduce los casos y condiciones de prueba a partir de la especificación
- Documento de especificación de requisitos del sistema
- Casos de uso
- Especificaciones funcionales
- Orientado a validar las características externas
  - Caja negra
  - Punto de vista del usuario
- Aplicables a todos los niveles de pruebas

¿Cómo probar funcionalmente?

En teoría…

- Identificar todos los valores de entrada posibles (y sus combinaciones)
- Generar un caso de prueba para cada valor o combinación
- Determinar los valores de salida esperados para cada caso
- Ejecutar cada caso y verificar que el resultado obtenido es igual al esperado

Impracticable…

- Debido a la gigantesca cantidad de casos posibles

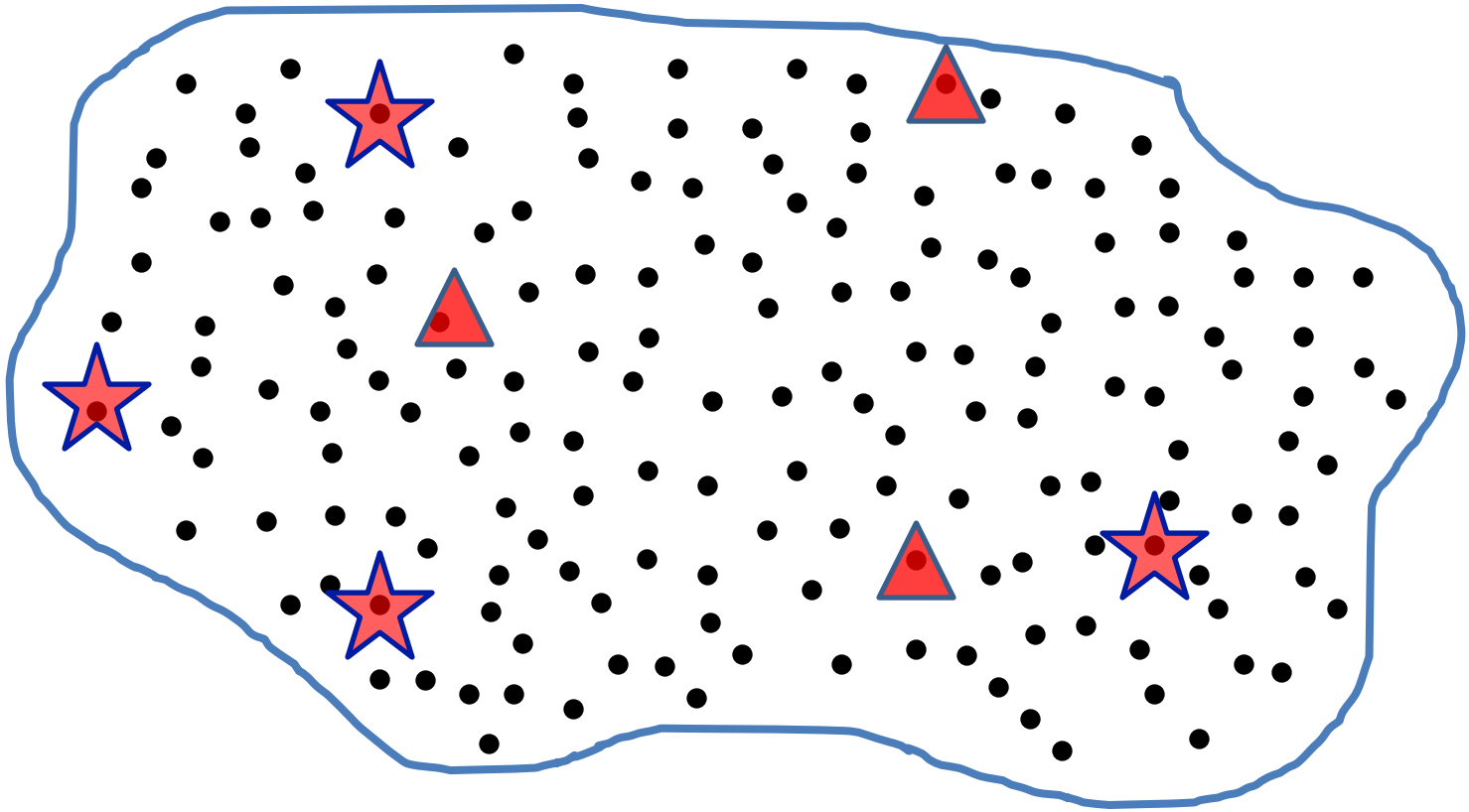
ACTUALIZACIÓN INFORMÁTICA

CPAP

Generación de casos de prueba

- Análisis de las especificaciones
- Identificación de comportamientos de la función a ser probada, considerando la relación entre entradas y salidas
- Identificar las variables de entrada pertinentes
- Generar la combinación de las variables/valores de entrada

Aleatorio



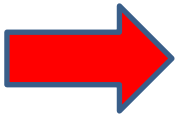
Estrategia selectiva

A partir del análisis de especificaciones

Deducir manualmente los casos más interesantes y representativos

Deducir automáticamente a partir de un modelo y herramientas de generación. Posible en forma parcial, en particular utilizando UML

Técnicas de diseño de casos de prueba



Particiones en Clases de Equivalencia

Análisis de Valores de Borde

Testing combinatorio

Tablas/Árboles de Decisión

Transición de Estados

Casos de Uso

ACTUALIZACIÓN INFORMÁTICA

CPAP

Repaso

Partición en clases de equivalencia

Valores límite

Cada valor debe utilizarse una vez en un caso válido

Partición en clases de equivalencia

Seguro de un automóvil

- No es aceptado para menores de 18 años ni mayores de 90.
- Para las personas de hasta 21 años solo se les permite asegurar autos de categoría Turista, con una tasa de 30%.
- Los mayores de 75 años serán asegurados solo para autos de categoría Turista y Sedan, con una tasa de 20%.
- Para los conductores mayores de 21 años hasta 65, la tasa será de 0% para autos de Turista, 5% para Sedan y 25% Deportivos.

Partición en clases de equivalencia

Edad en años:

- menores de 18
- 18 a 21
- 22 a 65
- 66 a 75
- 76 a 90
- mayores de 90

Categoría

- Turismo
- Deportivos
- Sedan

Partición en clases de equivalencia

Edad en años:

- menores de 18
- 18 a 21
- 22 a 65
- 66 a 75
- 76 a 90
- mayores de 90

Categoría

- Turismo
- Deportivos
- Sedan

Seguro es:

- Aceptado
- Rechazado

Tasa de:

- 0%
- 5%
- 20%
- 25%
- 30%

Valores límite

- Propone generar casos de prueba para los valores de borde de las particiones de equivalencia
- Es alta la probabilidad de defectos en la programación de los valores de borde
- Complementa la técnica anterior

Estrategia de selección

PARÁMETRO	CLASE DE EQUIVALENCIA	VÁLIDA/ INVÁLIDA	VALOR SELECCIONADO
Edad	• menores de 18 • entre 18 y 21 • entre 21 y 65 • entre 66 y 75 • entre 76 y 90 • mayores de 90 • Todo lo demás	V V V V V V INV	15, 0, 17, 19, 18, 21 50, 22, 65 70, 66, 75 83, 76, 90 95, 91 -1, 0,5, zz
Categoría	• Turista • Deportivos • Sedan	V V V	
SALIDAS			
Tasa %	0, 5, 20, 25, 30		
¿El seguro es?	Aceptado, Rechazado		

Generación de casos de prueba

- Análisis de las especificaciones,
- Identificación de comportamientos de la función a ser probada, considerando la relación entre entradas y salidas
- Identificar las variables de entrada pertinentes
- Para cada variable, analizar los subconjuntos de valores que deben tener el mismo comportamiento de la función : clase de equivalencia
- Generar la combinación de las clases de equivalencia de las variables pertinentes de entrada

ACTUALIZACIÓN INFORMÁTICA

CPAP

Estrategia de combinación

Each use o 1-wise

Pairwise

N-wise

Estrategia Each use o 1-wise

Casos de prueba

18, turista-> Acepta 30%

21, deportivo-> Rechaza

22, sedán-> Acepta 5%

65, turista->Acepta 0%

66, deportivo-> No definido

75, sedan-> No definido

76, turista-> Acepta 20%

90, deportivo-> Rechaza

0; 91 -> No puede sacar seguro

-1; 0,5; zz ->Edad inválida

PARÁMETRO	VALOR SELECCIONADO
Edad	15, 0, 17, 19, 18, 21 50, 22, 65 70, 66, 75 83, 76, 90 95, 91 -1, 0,5, zz
Categoría	Turista Deportivos Sedan

Generación de casos de prueba

- Análisis de las especificaciones: preguntar en caso de ambigüedades o falta de definición
- Identificación de comportamientos de la función a ser probada, considerando la relación entre entradas y salidas
- Identificar las variables de entrada pertinentes
- Para cada variable, analizar los subconjuntos de valores que deben tener el mismo comportamiento de la función : clase de equivalencia
- Generar la combinación de las clases de equivalencia de las variables pertinentes de entrada
- Agregar otros datos básicos id, autor, condiciones de ejecución, fecha

Estrategia Each use o 1-wise

Cuando usarla

- En etapas tempranas o exploratorias.
- Cuando se quiere un set pequeño de casos que cubra la diversidad de valores.
- Cuando no es crítico cubrir combinaciones

ACTUALIZACIÓN INFORMÁTICA

CPAP

Testing combinatorio

Estrategia de combinación

Each use o 1-wise

Pairwise

N-wise

Estrategia de combinación Pairwise

Algoritmos

AETG Automatic Efficient Test Generator y variaciones

ACTS (Advanced Combinatorial Testing System, de NIST)

PROW Pairwise with restrictions, order and weight

Estrategia de combinación Pairwise

Herramientas

- https://www.satisfice.com/download/allpairs (Allpais James Bach)
- https://pairwise.yuuniworks.com/ Pairwise Pict Online
- https://www.pairwise.org/tools.html Pairwise Testing
- Herramientas de IA generativa

Combinación Pairwise Pict Online

Edad: 0, 15, 17, 18, 19, 21, 50, 65, 66, 70, 75, 76, 83, 90, 91, 95
Categoria: Turismo, Deportivos, Sedan, NA

if [Edad] IN {0, 15, 17, 91, 95} then [Categoria] = "NA";
if [Edad] NOT IN {0, 15, 17, 91, 95} then [Categoria] <> "NA";
if [Edad] IN {18, 19, 21} then [Categoria] = "Turismo";
if [Edad] IN {50, 65} then [Categoria] <> "Deportivos";

30 casos de prueba: inválidos + combinaciones no permitidas

https://github.com/Microsoft/pict/blob/main/doc/pict.md

ACTUALIZACIÓN INFORMÁTICA

CPAP

Combinación Pairwise Pict Online

Edad: 18y21, 22y65, 66y75, 76y90

Categoria: Turismo, Deportivos, Sedan

if [Edad] = "18y21" then [Categoria] = "Turismo";

if [Edad] = "76y90" then [Categoria] <> "Deportivos";

9 casos de prueba válidos

Combinación Pairwise Chat GPT

Haz una combinación de a pares para las variables edad y categoría y respetando las 2 restricciones: Edad: 18y21, 22y65, 66y75, 76y90 Categoria: Turismo, Deportivos, Sedan if [Edad] = "18y21" then [Categoria] = "Turismo"; if [Edad] = "76y90" then [Categoria] <> "Deportivos";

Algoritmo aplicado: Generación cartesiana con filtrado por restricciones

ACTUALIZACIÓN INFORMÁTICA

CPAP

Estrategia Pairwise

22y65	Turismo
66y75	Sedan
76y90	Turismo
66y75	Turismo
22y65	Sedan
76y90	Sedan
22y65	Deportivos
18y21	Turismo
66y75	Deportivos

CP	Entradas		Resultado esperado	
Id	Edad	Categoría	Puede sacar seguro	Con qué tasa
1	22	Turismo	Si	30
2	66	Sedan	Si	20
3	76	Turismo	Si	20
4	66	Turismo	Si	???
5	65	Sedan	Si	5
6	90	Sedan	Si	20
7	50	Deportivos	Si	25
8	21	Turismo	Si	30
9	70	Deportivos	Si	???
10	18	Deportivo	No	
11	19	Sedan	No	
12	80	Deportivo	No	
13	-1		ERROR	
14	0,5		ERROR	
15	zz		ERROR	
16	15		No	
17	0		No	
18	17		No	
19	95		No	
20	91		No	

ACTUALIZACIÓN INFORMÁTICA

CPAP

Testing combinatorio

Estrategia de combinación

N-wise

ACTUALIZACIÓN INFORMÁTICA

CPAP

Ejercicio Análisis y diseño de casos de prueba

Trabajo autónomo

3/6/2025

Ejercicio Análisis y diseño de casos de prueba

Función: Autentificación

El usuario ingresa su tarjeta bancaria y espera para ingresar su PIN.

El módulo lector de tarjetas extrae el número de tarjeta. La función sólo aceptará tarjetas que no se encuentren en la "lista negra". Si la tarjeta se encuentra en la "lista negra" será retenida por el cajero automático.

El usuario dispone de hasta una cantidad MAX de intentos de ingreso de su número de identificación. En caso de superarse esa cantidad MAX la autorización será denegada y la tarjeta será retenida.

Si la tarjeta no está en la "lista negra" y si el usuario ingresa correctamente su PIN la función autorizará al usuario a operar.

1. Identificar particiones de equivalencia
2. Combinar las particiones utilizando una herramienta de generación de a pares
3. Diseñar casos de prueba utilizando el resultado arrojado por la herramienta y
4. agregar manualmente los casos que considere necesarios.

Solo considerar el caso del funcionamiento normal, sin considerar eventuales fallas del entorno.

ACTUALIZACIÓN INFORMÁTICA

CPAP

Preguntas

Técnicas de diseño de casos de prueba

- Particiones en Clases de Equivalencia
- Análisis de Valores de Borde
- Testing combinatorio
- Tablas/Árboles de Decisión
- Transición de Estados
- Casos de Uso

Técnica Diagramas de Transición de Estado

DTE: Modelo de diseño de sistemas

- Sistemas que mantienen un estado o sistemas con memoria
- Autómatas, Diagramas de Estado de UML, Workflows, BPM

Muchos sistemas pueden ser modelados de esa forma

- Sistemas integrados, máquinas y dispositivos hardware/software
- Interfaces usuario complejas

Técnica Diagramas de Transición de Estado

Estados y Eventos

- Estados : situación estable de un sistema, hasta que un evento pueda modificarlo
- Eventos: situaciones exteriores que pueden ocurrir cuando el sistema está en cierto Estado
- DTE: presenta la síntesis de los estados, eventos y transiciones posibles entre estados.

Pruebas basadas en DTE

Pruebas de Transición de Estado

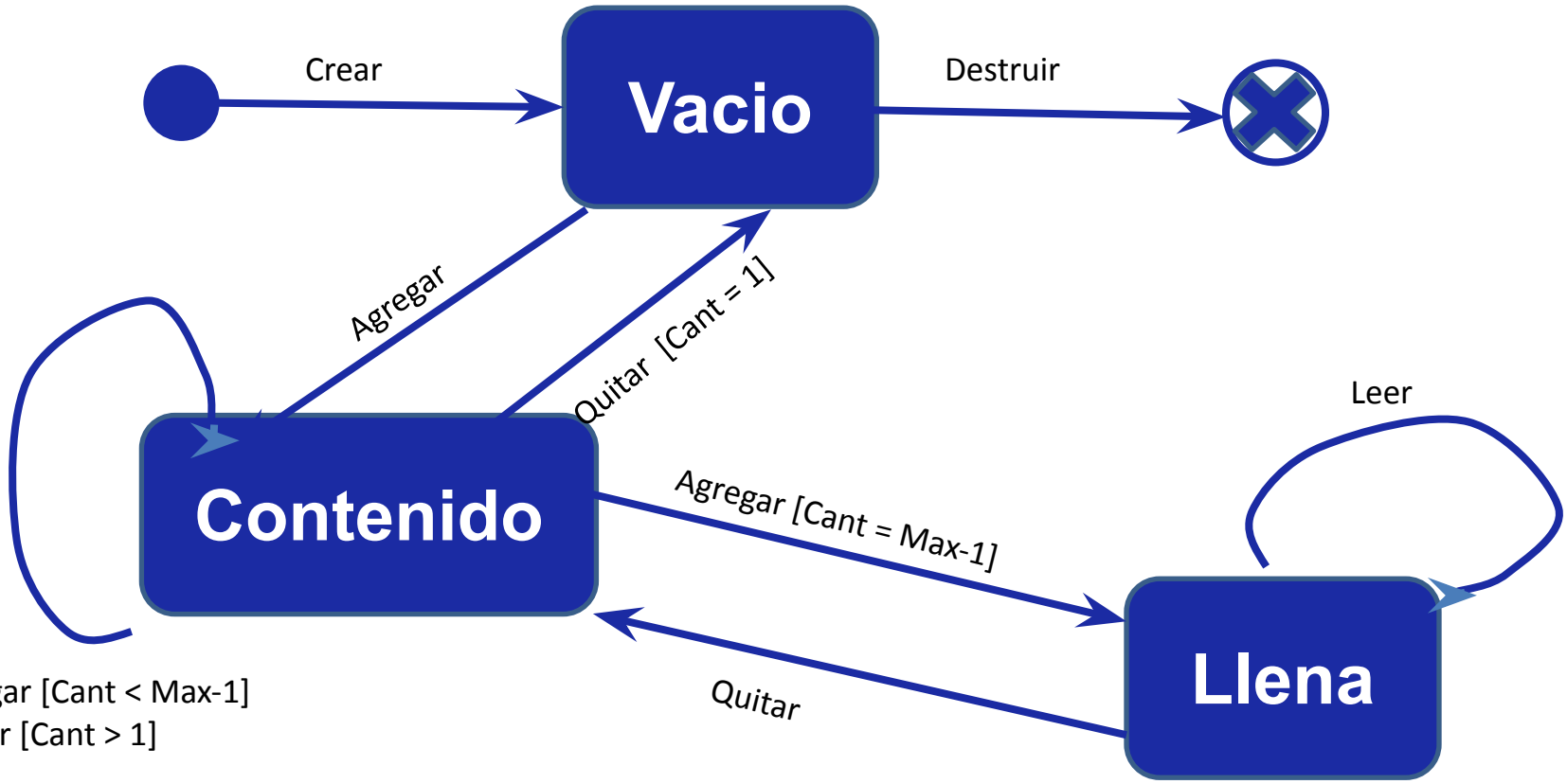
- Técnica de diseño de pruebas de caja negra en cual los casos de prueba son diseñados para ejecutar transiciones de estado válidas e inválidas.
- Orientadas a ejercercitar todas las transiciones de estado, verificando su buen funcionamiento.
- También se pueden incluir transiciones no válidas, no previstas o “imposibles”, a los efectos de verificar el comportamiento del sistema en esos casos

Casos de Prueba para DTE

Partir del DTE terminado y completo

1. Crear una Matriz de Transición de Estados
   Columnas y filas son los Estados
2. Para cada Estado en las filas, identificar todas las transiciones posibles y los eventos que las disparan
3. Colocar identificadores en las casillas de transiciones inválidas
4. Crear una tabla de casos de prueba
5. A partir de la Matriz, cada casilla es un caso de prueba.
6. Comenzar con las “pruebas positivas” (transiciones válidas)
7. Prever casos de transiciones inválidas, para verificar la gestión de errores de transición
8. Completar y simplificar
9. Algunas transiciones son irrealistas, otras merecen más de una prueba

Ejemplo: Operar una lista de elementos



Agregar [Cant < Max-1]
Quitar [Cant > 1]
Leer

Para: Max > 1

Matriz de combinación de Estados

	Inicio	Vacío	Contenido	Llena
Inicio		Crear		
Vacío	Destruir		Agregar	
Contenido		Quitar [cant=1]	Agregar [cant<Max-1] Quitar [cant>1] Leer	Agregar [cant= Max-1]
Llena			Quitar	Leer

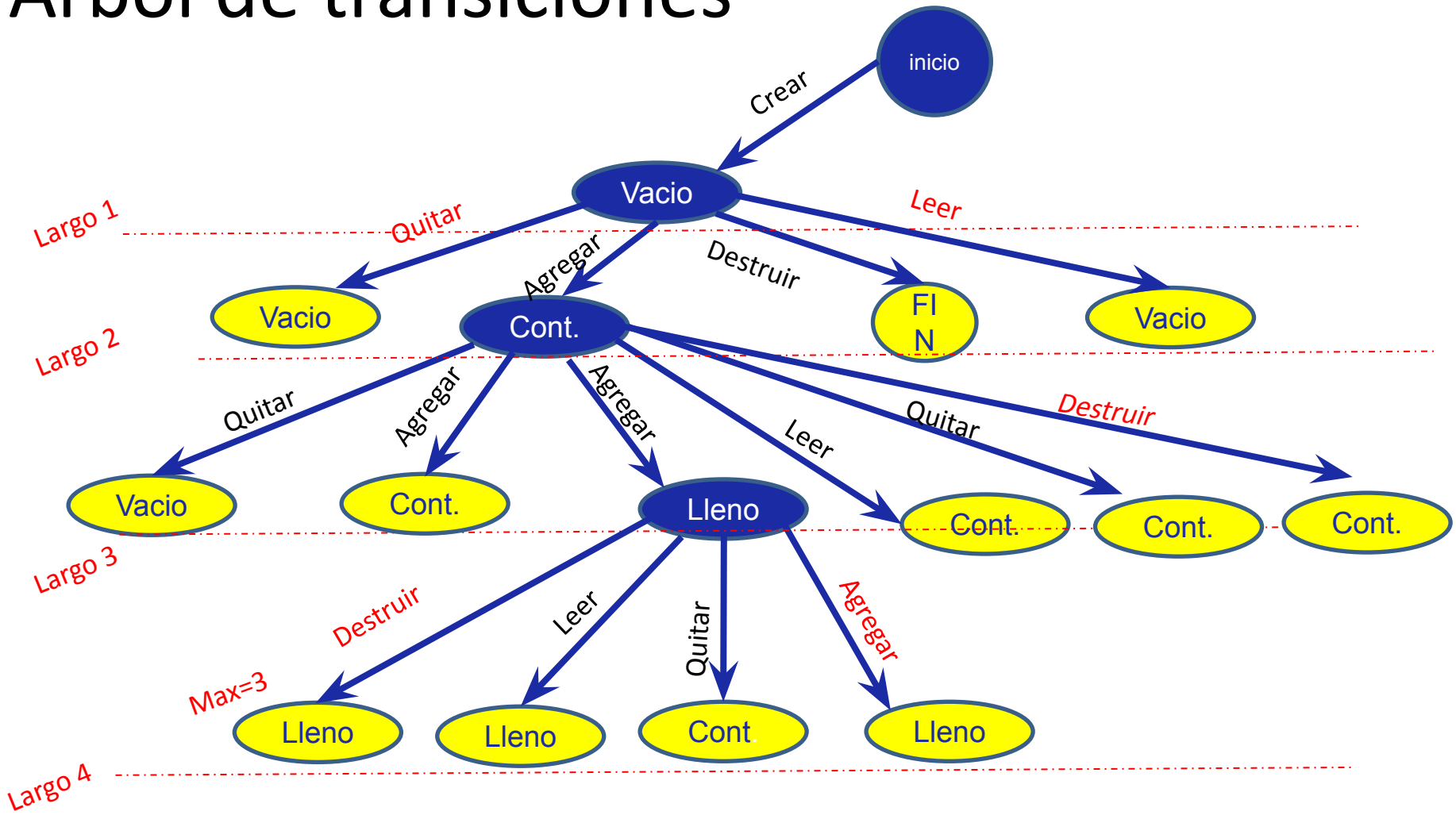
Matriz de combinación de Estados

	Inicio	Vacío	Contenido	Llena
Inicio		Crear		
Vacío	Destruir	Leer Quitar	Agregar	
Contenido		Quitar [cant=1]	Agregar [cant<Max-1] Quitar [cant>1] Leer Destruir	Agregar [cant= Max-1]
Llena			Quitar	Leer Destruir Agregar

Tabla de Transiciones:

Transición	Estado de partida	Evento	Estado de llegada	Condición	Acciones
1	Inicio	Crear	Vacío		Se crea la lista
2	Vacío	Quitar	Vacío		Error: La lista no tiene elementos a quitar
3	Vacío	Agregar	Contenido		Ahora hay un elemento en la lista
4	Vacío	Destruir	Fin		Destruir la lista
5	Vacío	Leer	Vacío		Error La lista no tiene elementos a leer
6	Contenido	Quitar	Vacío	La cantidad de es = 1	La lista queda vacía
7	Contenido	Agregar	Contenido		Agregar elemento, sin llegar al máximo
8	Contenido	Agregar	Lleno	La cantidad es de Max -1	Se agrega un elemento y la lista queda llena
9	Contenido	Leer	Contenido		Lectura de un elemento de la lista
10	Contenido	Quitar	Contenido	La cantidad es > 1	.Se quita un elemento
11	Contenido	Destruir	Contenido		Error: no se puede destruir la lista
12	Lleno	Destruir	Lleno		Error: no se puede destruir la lista
13	Lleno	Leer	Lleno		Lectura de un elemento de la lista
14	Lleno	Quitar	Contenido		La cantidad resultante será de Max-1
15	Lleno	Agregar	Lleno		Error: no es posible agregar elementos

Árbol de transiciones

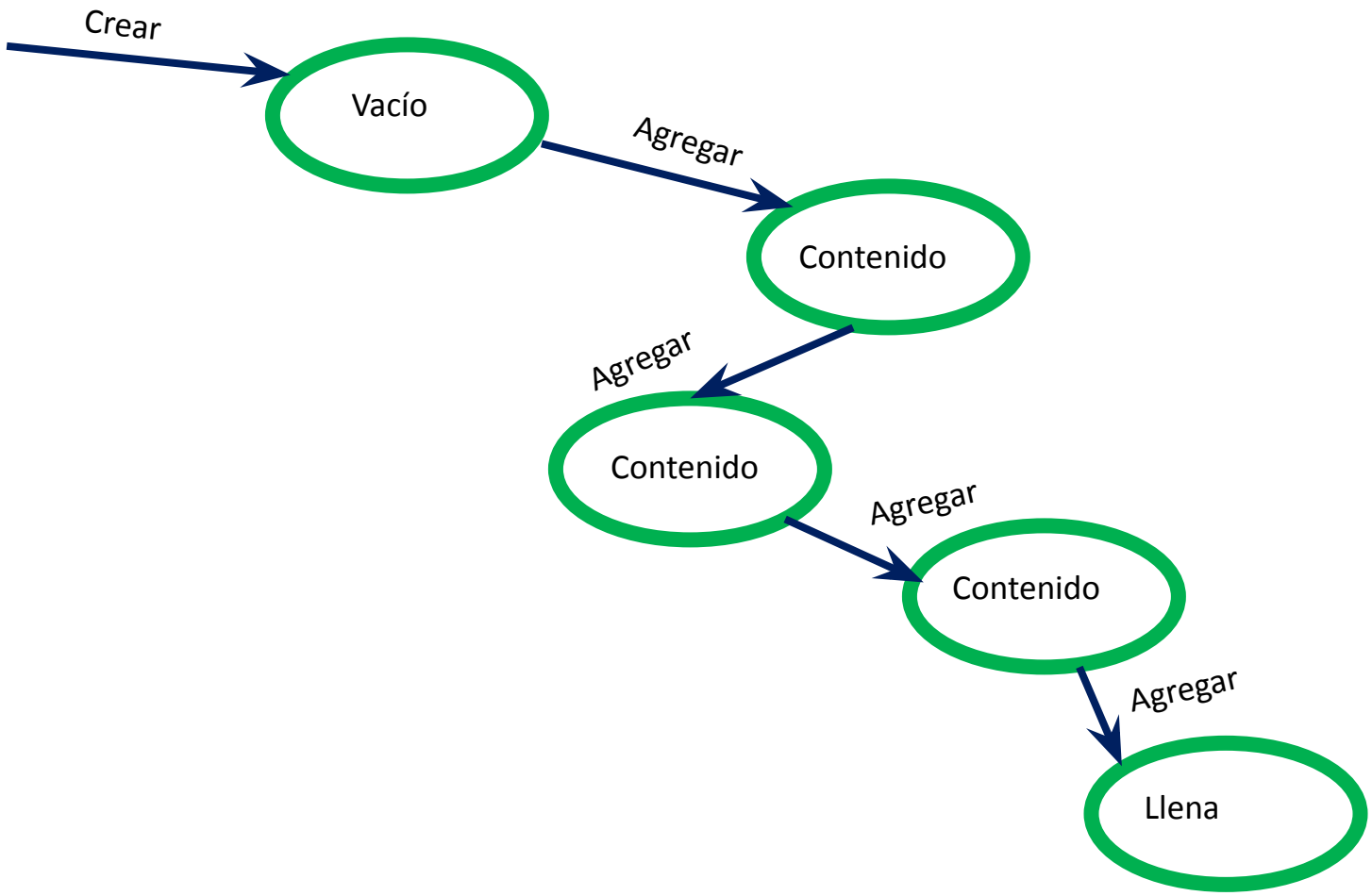


Para: Max = 2

Casos de Prueba para DTE

CP			Entradas	Resultado esperado	
Id	Estado inicial	Condición	Evento	Estado final	Acción
1	Inicio Vacío		Crear Quitar	Vacío Vacío	Se crea la lista vacía Msj Error
2	Inicio Vacío Contenido Lleno	Cant = Max-1	Crear Agregar Agregar Leer	Vacío Contenido Lleno Lleno	Se crea la lista vacía SE agrega un elemento Se agrega un elemento Se lee la lista
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					

Caso MAX=4


Crear
Vacío
Agregar
Contenido
Agregar
Contenido
Agregar
Contenido
Agregar
Llena

Criterio de cubrimiento

Todas las transiciones

- Orientadas a ejercer todas las transiciones de estado, verificando su buen funcionamiento.

Otros

- Todos los estados

Conclusiones

Las pruebas Basadas en la Especificación son indispensables

Puede detectar defectos en la especificación

- Incoherencia, ambigüedad, faltantes (no completo)

No detecta funcionalidades extra o innecesarias

Los criterios de salida son basados en las especificaciones

- Cobertura funcional

ACTUALIZACIÓN INFORMÁTICA

CPAP

Preguntas

Muchas gracias

ACTUALIZACIÓN INFORMÁTICA
CPAP

Bibliografía

Técnicas de testing combinatorio y de mutación Primera parte
Macario Polo Usaola Grupo Alarcos Departamento de Tecnologías y Sistemas de Información Universidad de Castilla-La Mancha Ciudad Real, España