



Estrategias de Persistencia

Introducción

Universidad Nacional de Quilmes

Contenido

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Outline

- 1 **Introducción**
 - **Aplicación y estado**
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Persistencia

Aplicación y estado

Aplicación

- Proceso ejecutado por S.O.
- Area de memoria asignada
- Paradigmas imperativos: transformación continua de ese estado (efecto de lado)

Persistencia

Aplicación y estado

Aplicación

- Proceso ejecutado por S.O.
- Area de memoria asignada
- Paradigmas imperativos: transformación continua de ese estado (efecto de lado)

Persistencia

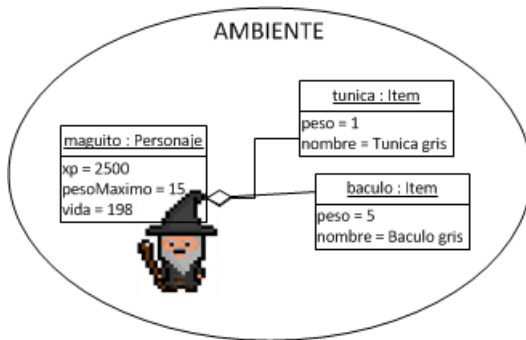
Aplicación y estado

Aplicación

- Proceso ejecutado por S.O.
- Area de memoria asignada
- Paradigmas imperativos: transformación continua de ese estado (efecto de lado)

Ambiente de objetos y estado

Ejemplo



Ambiente de objetos y estado

Ejemplo

Hacen al estado de mi sistema:

- Las referencias de personaje a números (como la cantidad de “vida”)
- Las relaciones de personaje con otros objetos que forman parte de su inventario
- Que mi personaje exista (sea un objeto que tenga una identidad)

Ambiente de objetos y estado

Ejemplo

Hacen al estado de mi sistema:

- Las referencias de personaje a números (como la cantidad de “vida”)
- Las relaciones de personaje con otros objetos que forman parte de su inventario
- Que mi personaje exista (sea un objeto que tenga una identidad)

Ambiente de objetos y estado

Ejemplo

Hacen al estado de mi sistema:

- Las referencias de personaje a números (como la cantidad de “vida”)
- Las relaciones de personaje con otros objetos que forman parte de su inventario
- Que mi personaje exista (sea un objeto que tenga una identidad)

Ambiente de objetos y estado

Requerimientos

- Almacenar el estado generado por una ejecución para ser reutilizado por ejecuciones posteriores.
- Sobrevivir a la volatilidad de la memoria.
- Poder trabajar con estados más grandes que la capacidad física de la memoria.

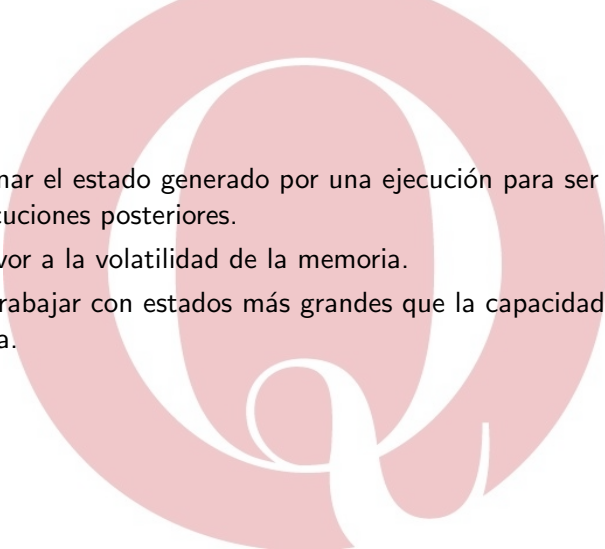
Ambiente de objetos y estado

Requerimientos

- Almacenar el estado generado por una ejecución para ser reutilizado por ejecuciones posteriores.
- Sobrevivir a la volatilidad de la memoria.
- Poder trabajar con estados más grandes que la capacidad física de la memoria.

Ambiente de objetos y estado

Requerimientos

- 
- Almacenar el estado generado por una ejecución para ser reutilizado por ejecuciones posteriores.
 - Sobrevivir a la volatilidad de la memoria.
 - Poder trabajar con estados más grandes que la capacidad física de la memoria.

Outline

1 Introducción

- Aplicación y estado

• Persistencia

- Decisiones de diseño

2 Medios persistentes

- Snapshot
- Serialización a archivos
- Servidores de Bases de Datos

3 Estrategias de Persistencia

- Estrategias de Persistencia
- Ambiente vivo + BD como backup
- Esquema prevalente
- Ambiente muerto y recreado

4 Transaccionalidad

- Transaccionalidad
- Persistencia y transaccionalidad

Persistencia

¿Qué significa persistir?

Capacidad de una aplicación de hacer frente a los requerimientos anteriores.

Conceptos:

- **Base de datos:** es el conjunto de datos persistidos.
- **Medio Persistente:** Es el soporte físico donde se almacenan los datos.
 - Disco magnético.
 - Array de discos.
 - Memoria flash.
 - Cluster de máquinas.

Persistencia

¿Qué significa persistir?

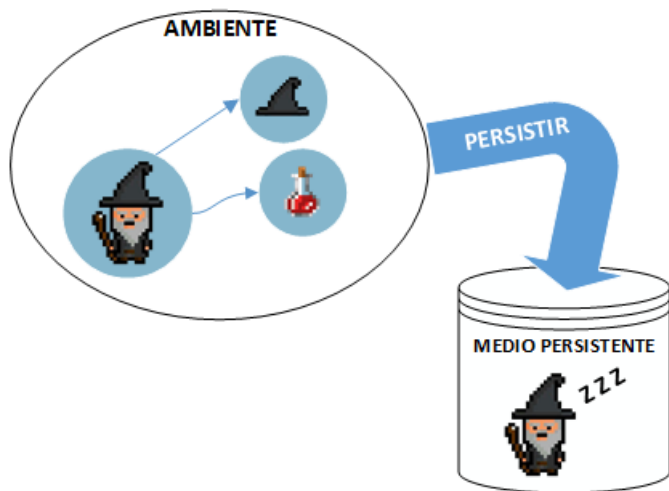
Capacidad de una aplicación de hacer frente a los requerimientos anteriores.

Conceptos:

- **Base de datos:** es el conjunto de datos persistidos.
- **Medio Persistente:** Es el soporte físico donde se almacenan los datos.
 - Disco magnético.
 - Array de discos.
 - Memoria flash.
 - Cluster de máquinas.

Persistencia

¿Qué significa persistir?



Outline

1 Introducción

- Aplicación y estado
- Persistencia
- **Decisiones de diseño**

2 Medios persistentes

- Snapshot
- Serialización a archivos
- Servidores de Bases de Datos


3 Estrategias de Persistencia

- Estrategias de Persistencia
- Ambiente vivo + BD como backup
- Esquema prevalente
- Ambiente muerto y recreado

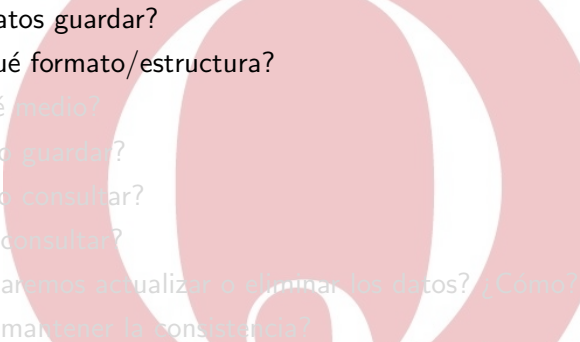
4 Transaccionalidad

- Transaccionalidad
- Persistencia y transaccionalidad

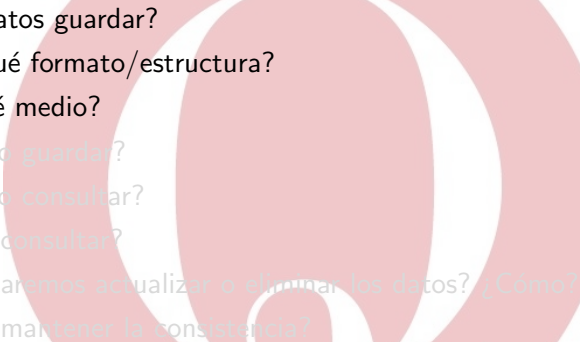
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

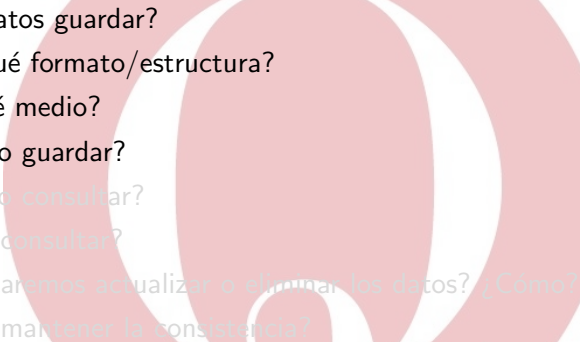
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

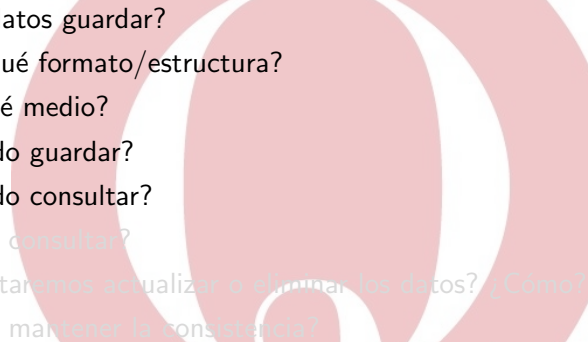
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

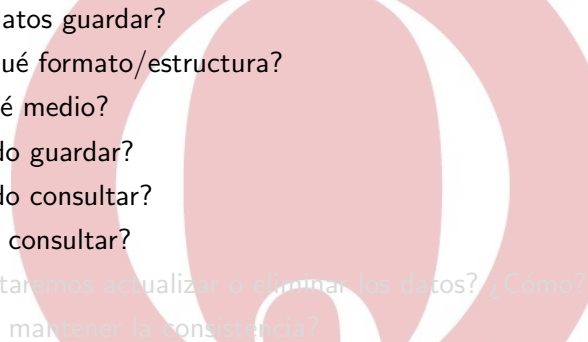
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

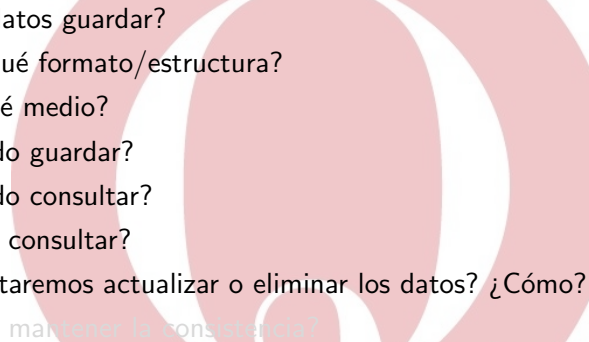
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

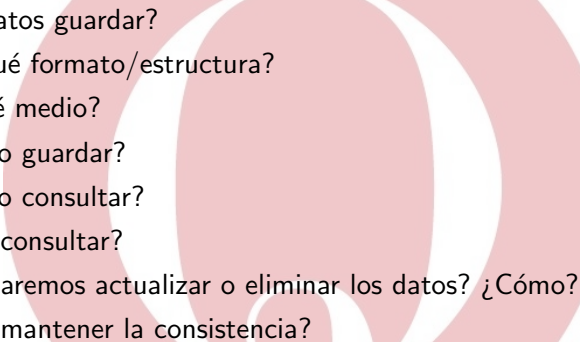
Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

Decisiones de diseño

- 
- ¿Qué datos guardar?
 - ¿Con qué formato/estructura?
 - ¿En qué medio?
 - ¿Cuándo guardar?
 - ¿Cuándo consultar?
 - ¿Cómo consultar?
 - ¿Soportaremos actualizar o eliminar los datos? ¿Cómo?
 - ¿Cómo mantener la consistencia?

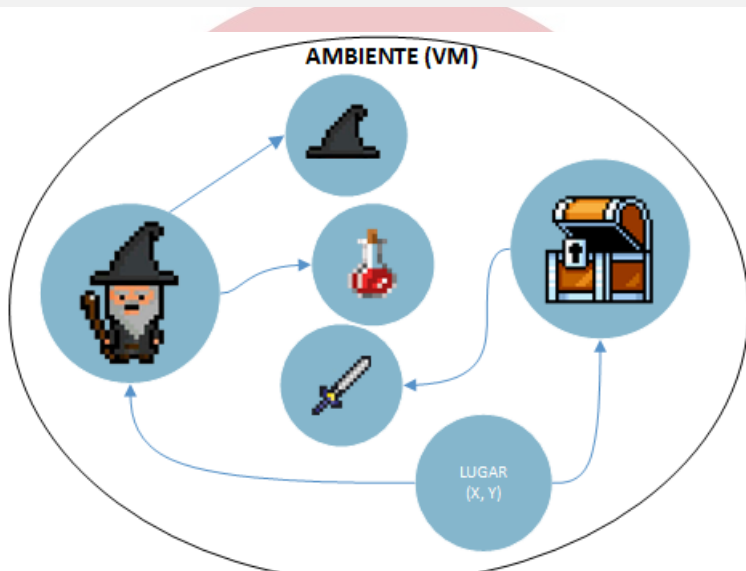
Dónde y cómo

Medio persistente:

- Cómo voy a persistir en el mismo.
- Qué estructura de datos tendrá.
- Cómo adaptar el modelo de objetos a dicha estructura.

Módulo de objetos

Cómo luce un ambiente de objetos



Cómo persistir objetos

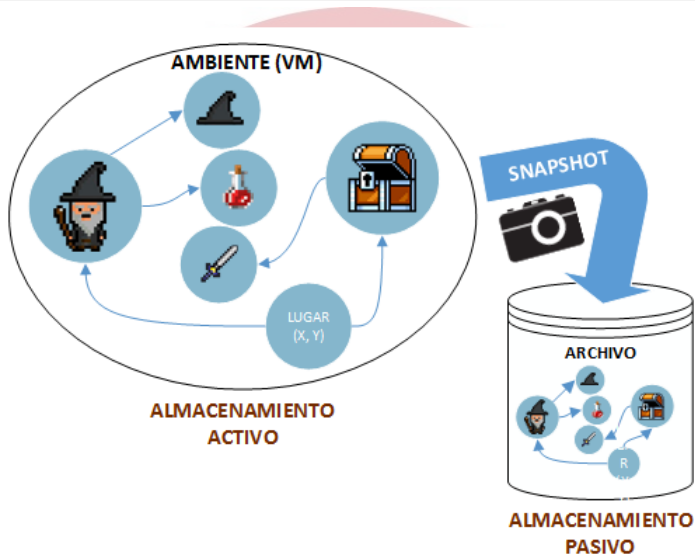
Alternativas

- Snapshot del ambiente completo
- Serializar a un formato de archivo de un subconjunto de objetos.
- Delegar un servicio de base de datos para persistir mi modelo
 - Bases de datos jerárquicas
 - Bases de datos en red
 - Bases de datos relacionales
 - Bases de datos NO relacionales (NoSQL)
 - Bases de datos de objetos

Outline

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - **Snapshot**
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Snapshot del ambiente



Snapshot del ambiente

- Foto del ambiente completo.
- Persistida a un archivo binario.
- Incluye todos los objetos (ventanas, variables globales, entorno de trabajo)
- Simple - el ambiente provee este servicio.
- No soporta cambios incrementales.

Snapshot del ambiente

- Foto del ambiente completo.
- Persistida a un archivo binario.
- Incluye todos los objetos (ventanas, variables globales, entorno de trabajo)
- Simple - el ambiente provee este servicio.
- No soporta cambios incrementales.

Snapshot del ambiente

- Foto del ambiente completo.
- Persistida a un archivo binario.
- Incluye todos los objetos (ventanas, variables globales, entorno de trabajo)
- Simple - el ambiente provee este servicio.
- No soporta cambios incrementales.

Snapshot del ambiente

- Foto del ambiente completo.
- Persistida a un archivo binario.
- Incluye todos los objetos (ventanas, variables globales, entorno de trabajo)
- Simple - el ambiente provee este servicio.
- No soporta cambios incrementales

Snapshot del ambiente

- Foto del ambiente completo.
- Persistida a un archivo binario.
- Incluye todos los objetos (ventanas, variables globales, entorno de trabajo)
- Simple - el ambiente provee este servicio.
- No soporta cambios incrementales.

Snapshot del ambiente

Viable por lo general solamente en:

- Aplicaciones monousuario
- Persistencia a intervalos discretos (grandes)

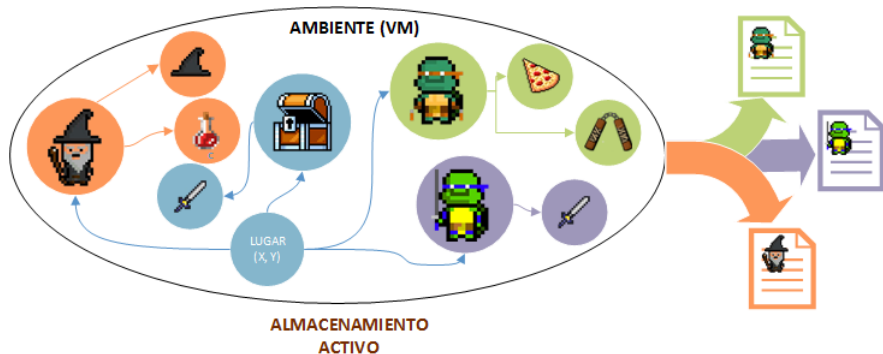
Outline

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - **Serialización a archivos**
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Serialización a archivos

Implica:

- Seleccionar un o varios **subgrafos**
- Persistirlo a uno o varios **archivos**



Serialización a archivos

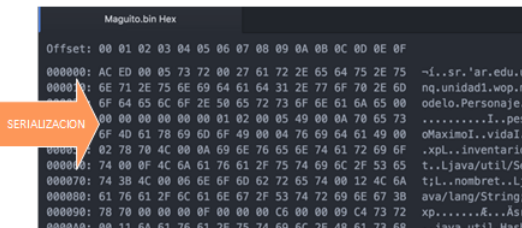
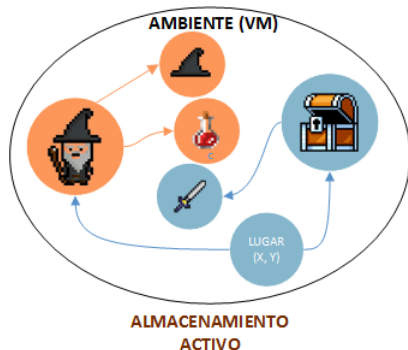
Formatos de archivo

- Archivos binarios
- Archivos de texto
 - JSON
 - XML

Serialización a archivos

Archivos binarios

- ObjectOutputStream (java)
- ReferenceStream (smalltalk)
- Apache Avro, Google's protocolbuffers, etc



**Archivo
ALMACENAMIENTO
PASIVO**

Serialización a archivos

Archivos binarios

Ventajas:

- Eficiencia (temporal y espacial)
- Mejor soporte para estructuras de tipo grafo

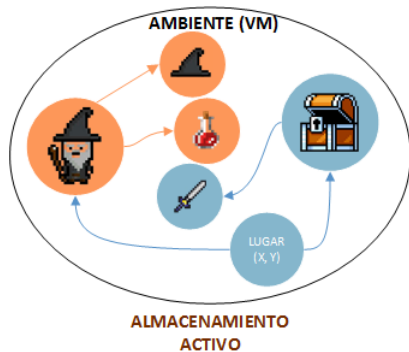
Desventajas:

- Opacos
- Estándares pobres
- Acoplados a la implementación nativa

Serialización a archivos

Archivos de texto - XML

- JAXB
- XStream



SERIALIZACION

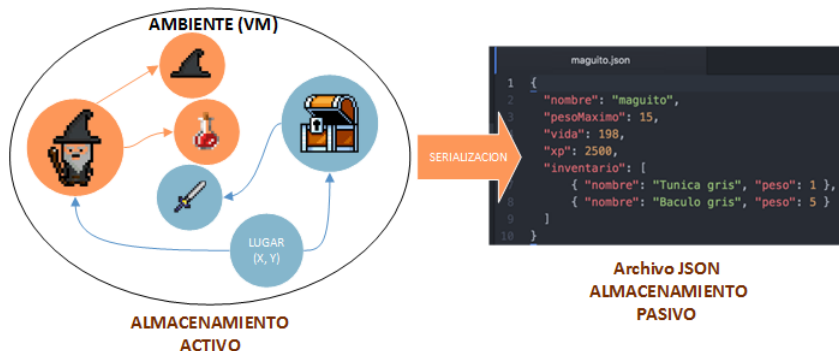
```
maguito.xml
<?xml version="1.0" encoding="UTF-8"?>
<personaje nombre="maguito" pesoMaximo="15" vida="198" xp="250">
  <inventario>
    <item nombre="Tunica Gris" peso="1" />
    <item nombre="Baculo Gris" peso="5" />
  </inventario>
</personaje>
```

**Archivo XML
ALMACENAMIENTO
PASIVO**

Serialización a archivos

Archivos de texto - JSON

- Apache Jackson
- FlexJSON
- Google's GSON
- json-io



Serialización a archivos

Archivos de texto


Ventajas:

- El resultado puede leerse (y editarse) manualmente
- Suelen ser formatos mas estándares
- No acomplados a una implementación, portables

Desventajas:

- Menos eficientes
- Implican muchas conversiones de tipo
- Suelen ser limitados en cuanto a estructura (persisten arboles)

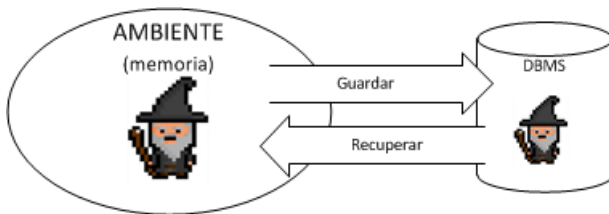
Outline

- 
- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
 - 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - **Servidores de Bases de Datos**
 - 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
 - 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Servidores de Bases de Datos

Es posible delegar en un servicio de base de datos:

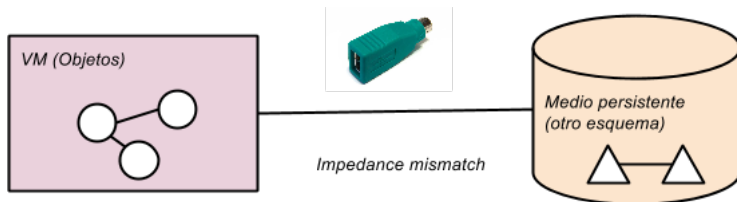
- Aplicación especializada en persistir



Servidores de Bases de Datos

Mapeo contra otros modelos

Adaptaciones que cubran esas diferencias (lo que se llama **impedance mismatch**).



Mapeo contra otros modelos

Algunas opciones posibles son:

- **Modelo relacional:** estructura la información en:
 - Campos (columnas)
 - Registros (tuplas de campos)
 - Tablas (conjuntos de registros similares)
 - Relaciones entre tablas
- **Modelo en jerárquico:** la información se persiste en una estructura de nodos padre-hijo
- **Modelo en red:** los nodos no tienen una relación padre-hijo sino de iguales (persiste grafos)
- **NoSQL:** alternativa a BD relacionales.

Mapecto contra otros modelos

Algunas opciones posibles son:

- **Modelo relacional:** estructura la información en:
 - Campos (columnas)
 - Registros (tuplas de campos)
 - Tablas (conjuntos de registros similares)
 - Relaciones entre tablas
- **Modelo en jerarquico:** la información de persiste en una estructura de nodos padre-hijo
- **Modelo en red:** los nodos no tienen una relación padre-hijo sino de iguales (persiste grafos)
- **NoSQL:** alternativa a BD relacionales.

Mapecto contra otros modelos

Algunas opciones posibles son:

- **Modelo relacional:** estructura la información en:
 - Campos (columnas)
 - Registros (tuplas de campos)
 - Tablas (conjuntos de registros similares)
 - Relaciones entre tablas
- **Modelo en jerárquico:** la información de persiste en una estructura de nodos padre-hijo
- **Modelo en red:** los nodos no tienen una relación padre-hijo sino de iguales (persiste grafos)
- **NoSQL:** alternativa a BD relacionales.

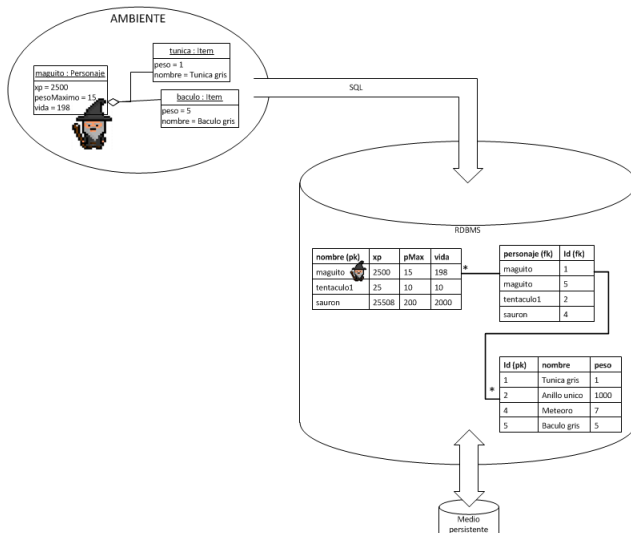
Mapecto contra otros modelos

Algunas opciones posibles son:

- **Modelo relacional:** estructura la información en:
 - Campos (columnas)
 - Registros (tuplas de campos)
 - Tablas (conjuntos de registros similares)
 - Relaciones entre tablas
- **Modelo en jerarquico:** la información de persiste en una estructura de nodos padre-hijo
- **Modelo en red:** los nodos no tienen una relación padre-hijo sino de iguales (persiste grafos)
- **NoSQL:** alternativa a BD relacionales.

Mapecto contra otros modelos

Impedance mismatch

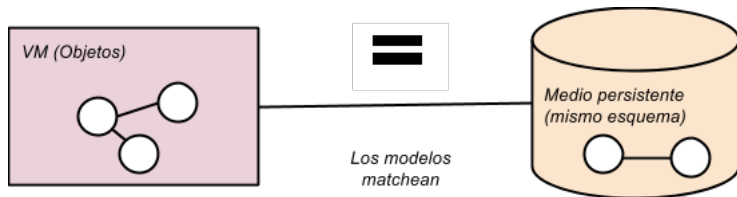


Servidores de Bases de Datos

Bases de datos de Objetos

Mismo modelo = desaparece el impedance mismatch

- Embebidas en el ambiente: Gemstone, Objectivity
- Servicios separados: DB4O, ObjectDB, Realm, Magma



Outline

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - **Estrategias de Persistencia**
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Estrategias de Persistencia

Nos interesa responder ahora las siguientes preguntas:

- Cuándo persistir
- Cómo manter la consistencia

Outline

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - **Ambiente vivo + BD como backup**
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Estrategias de Persistencia

Ambiente vivo + BD como backup

Funcionamiento:

- Mantiene todos los objetos vivos en la imagen.
- Opera sobre ellos en forma normal.
- Luego de cada cambio sincroniza con el medio persistente. (solo escribe)

Estrategias de Persistencia

Ambiente vivo + BD como backup

Funcionamiento:

- Mantiene todos los objetos vivos en la imagen.
- Opera sobre ellos en forma normal.
- Luego de cada cambio sincroniza con el medio persistente. (solo escribe)

Estrategias de Persistencia

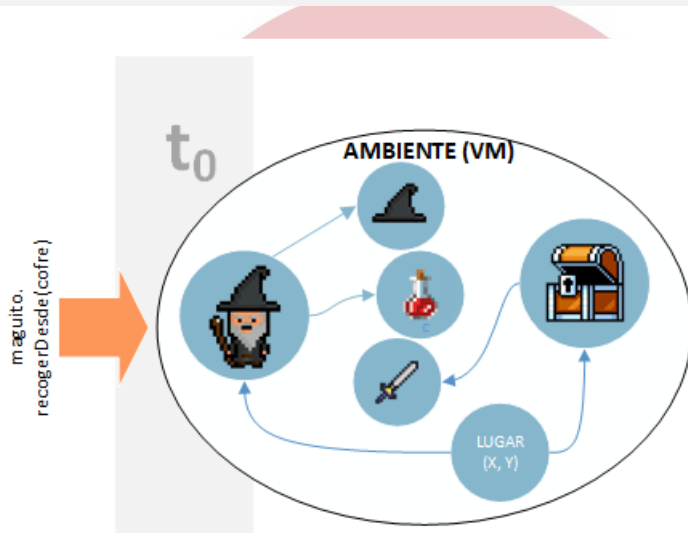
Ambiente vivo + BD como backup

Funcionamiento:

- Mantiene todos los objetos vivos en la imagen.
- Opera sobre ellos en forma normal.
- Luego de cada cambio sincroniza con el medio persistente. (solo escribe)

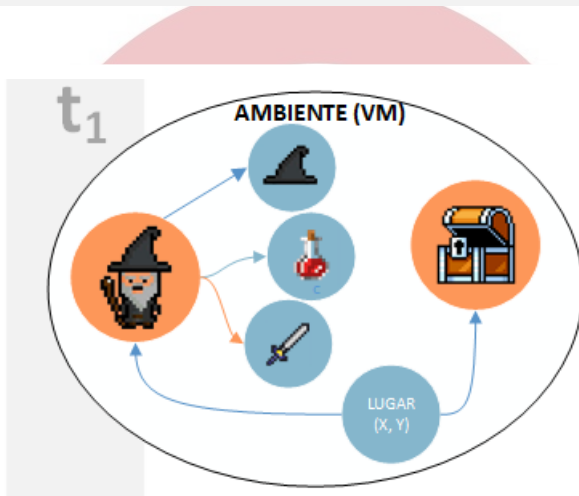
Estrategias de Persistencia

Ambiente vivo + BD como backup



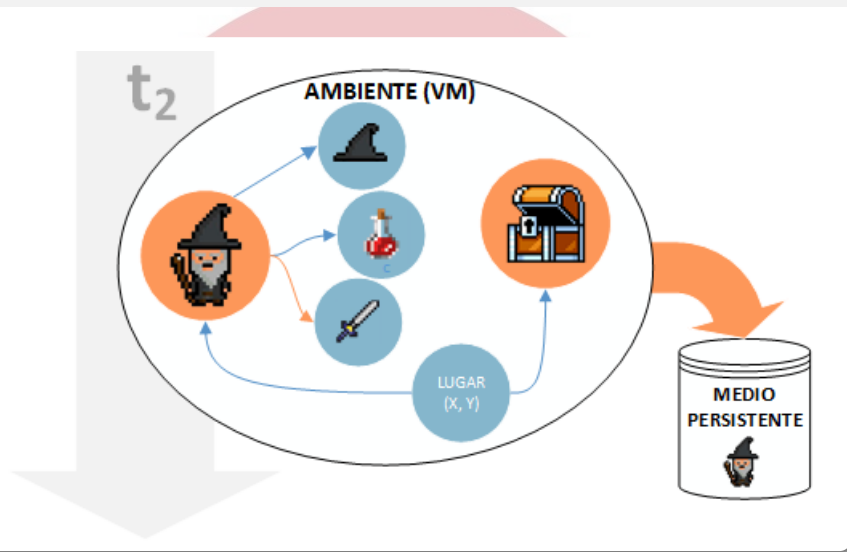
Estrategias de Persistencia

Ambiente vivo + BD como backup



Estrategias de Persistencia

Ambiente vivo + BD como backup



Estrategias de Persistencia

Ambiente vivo + BD como backup

Características:

- El ambiente es el "dueño" de los datos
- No nos ayudará a sortear la limitación de la memoria física
- Deberemos asegurar que nuestro estado en objetos es consistente y todo cambio es atómico y durable

Estrategias de Persistencia

Ambiente vivo + BD como backup

Características:

- El ambiente es el "dueño" de los datos
- No nos ayudará a sortear la limitación de la memoria física
- Deberemos asegurar que nuestro estado en objetos es consistente y todo cambio es atómico y durable

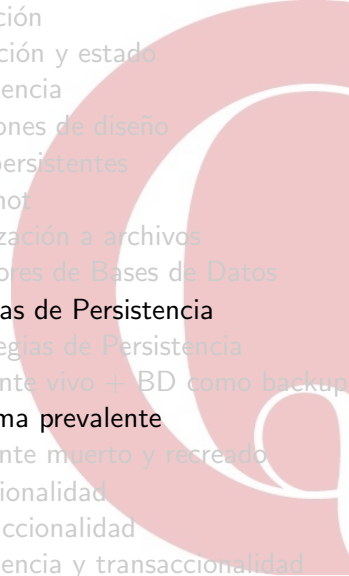
Estrategias de Persistencia

Ambiente vivo + BD como backup

Características:

- El ambiente es el "dueño" de los datos
- No nos ayudará a sortear la limitación de la memoria física
- Deberemos asegurar que nuestro estado en objetos es consistente y todo cambio es atómico y durable

Outline

- 
- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
 - 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
 - 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - **Esquema prevalente**
 - Ambiente muerto y recreado
 - 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Estrategias de Persistencia

Esquema prevalente

Funcionamiento similar a la estrategia anterior:

- **Modelo de objetos vivo**
- Se guarda un snapshot inicial de todo el ambiente a intervalos discretos
- Toda operación sobre el ambiente se modela como un comando
- Toda operación sobre el ambiente se persiste (operación, no resultado) en un log

Estrategias de Persistencia

Esquema prevalente

Funcionamiento similar a la estrategia anterior:

- Modelo de objetos vivo
- Se guarda un snapshot inicial de todo el ambiente a intervalos discretos
- Toda operación sobre el ambiente se modela como un comando
- Toda operación sobre el ambiente se persiste (operación, no resultado) en un log

Estrategias de Persistencia

Esquema prevalente

Funcionamiento similar a la estrategia anterior:

- Modelo de objetos vivo
- Se guarda un snapshot inicial de todo el ambiente a intervalos discretos
- Toda operación sobre el ambiente se modela como un comando
- Toda operación sobre el ambiente se persiste (operación, no resultado) en un log

Estrategias de Persistencia

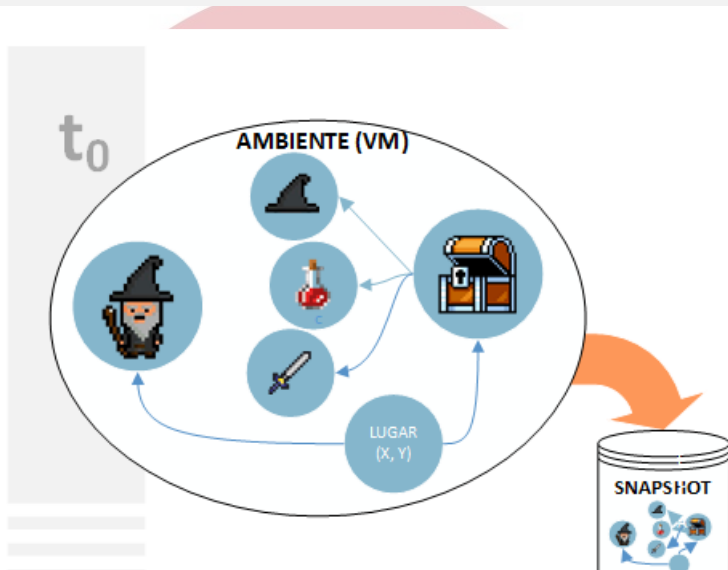
Esquema prevalente

Funcionamiento similar a la estrategia anterior:

- Modelo de objetos vivo
- Se guarda un snapshot inicial de todo el ambiente a intervalos discretos
- Toda operación sobre el ambiente se modela como un comando
- Toda operación sobre el ambiente se persiste (operación, no resultado) en un log

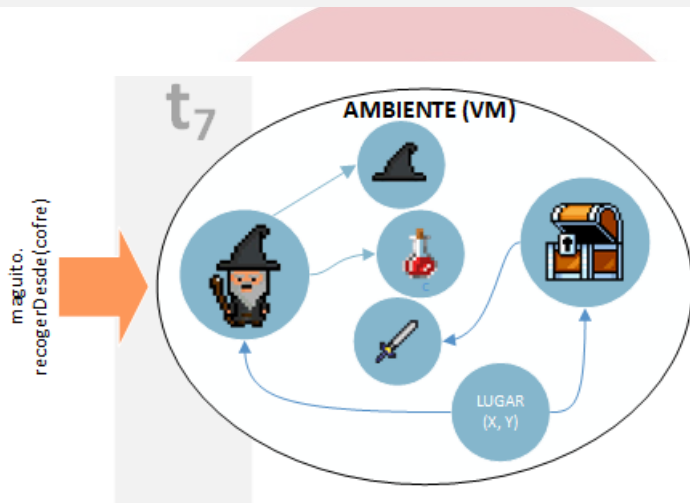
Estrategias de Persistencia

Esquema prevalente



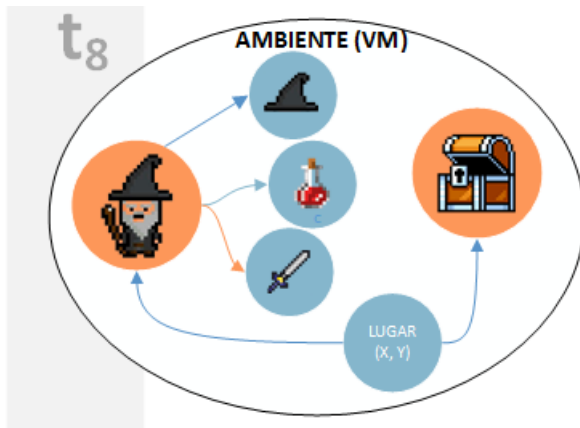
Estrategias de Persistencia

Esquema prevalente



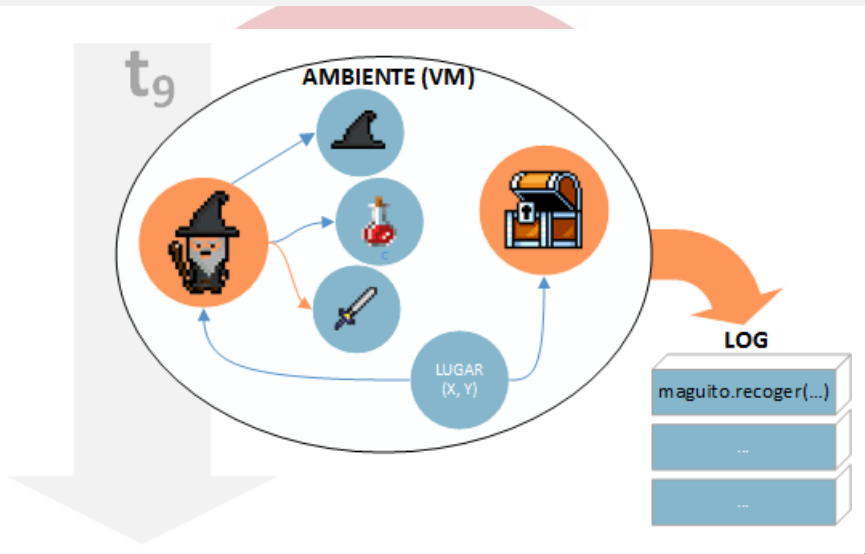
Estrategias de Persistencia

Esquema prevalente



Estrategias de Persistencia

Esquema prevalente



Estrategias de Persistencia

Esquema prevalente

Características:

- Relativamente simple de implementar
- Soporta una gran cantidad de transacciones
- Permite tener memoria de todos los estados intermedios

Estrategias de Persistencia

Esquema prevalente

Características:

- Relativamente simple de implementar
- Soporta una gran cantidad de transacciones
- Permite tener memoria de todos los estados intermedios

Estrategias de Persistencia

Esquema prevalente

Características:

- Relativamente simple de implementar
- Soporta una gran cantidad de transacciones
- Permite tener memoria de todos los estados intermedios

Outline

- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
- 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
- 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
- 4 Transaccionalidad
 - Transaccionalidad
 - Persistencia y transaccionalidad

Estrategias de Persistencia

Ambiente muerto y recreado

Funcionamiento:

- **Mantiene un ambiente vacío**
- Cada operación recupera de la BD previamente los objetos sobre los que va a operar
- Aplica los cambios
- Actualiza la BD
- Vacía el ambiente

Estrategias de Persistencia

Ambiente muerto y recreado

Funcionamiento:

- Mantiene un ambiente vacío
- Cada operación recupera de la BD previamente los objetos sobre los que va a operar
- Aplica los cambios
- Actualiza la BD
- Vacía el ambiente

Estrategias de Persistencia

Ambiente muerto y recreado

Funcionamiento:

- Mantiene un ambiente vacío
- Cada operación recupera de la BD previamente los objetos sobre los que va a operar
- Aplica los cambios
- Actualiza la BD
- Vacía el ambiente

Estrategias de Persistencia

Ambiente muerto y recreado

Funcionamiento:

- Mantiene un ambiente vacío
- Cada operación recupera de la BD previamente los objetos sobre los que va a operar
- Aplica los cambios
- Actualiza la BD
- Vacía el ambiente

Estrategias de Persistencia

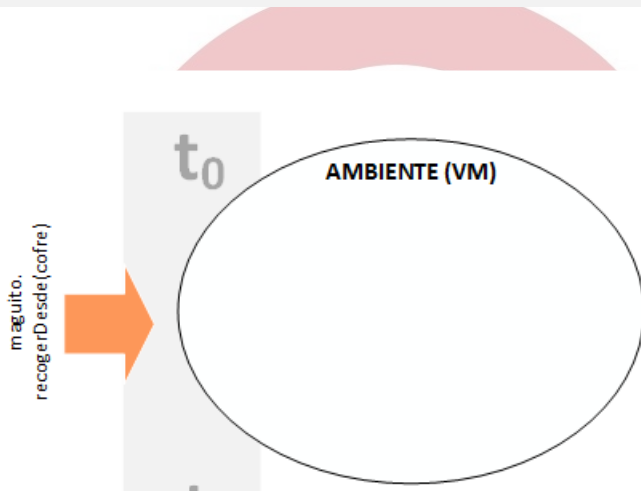
Ambiente muerto y recreado

Funcionamiento:

- Mantiene un ambiente vacío
- Cada operación recupera de la BD previamente los objetos sobre los que va a operar
- Aplica los cambios
- Actualiza la BD
- Vacía el ambiente

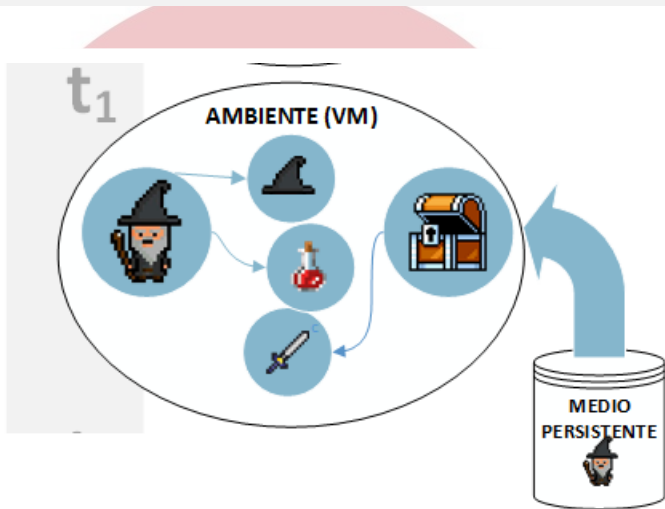
Estrategias de Persistencia

Ambiente muerto y recreado



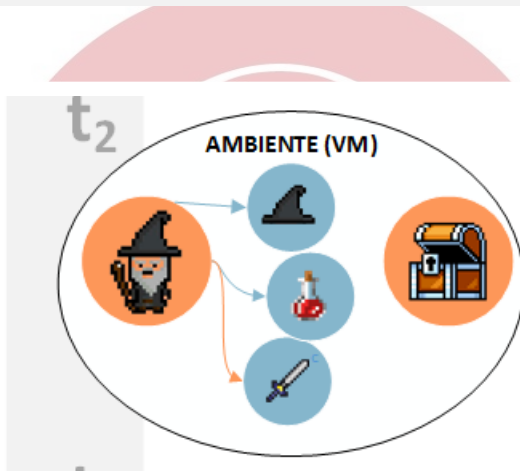
Estrategias de Persistencia

Ambiente muerto y recreado



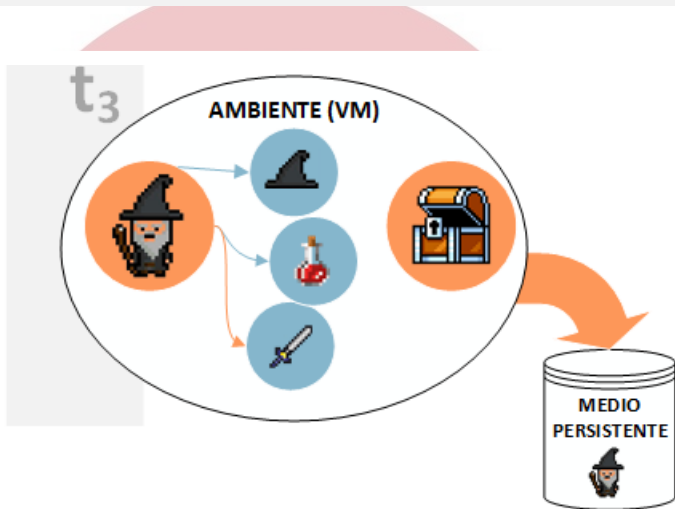
Estrategias de Persistencia

Ambiente muerto y recreado



Estrategias de Persistencia

Ambiente muerto y recreado

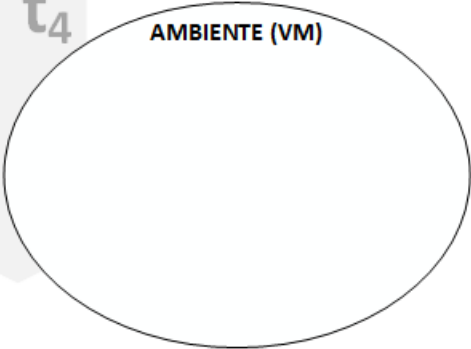


Estrategias de Persistencia

Ambiente muerto y recreado

t_4

AMBIENTE (VM)

A diagram showing a large, empty oval labeled "AMBIENTE (VM)". To the left of the oval is a grey arrow pointing downwards, with the label "t4" positioned above it. The entire diagram is set against a background with faint red and white curved shapes.

Estrategias de Persistencia

Ambiente muerto y recreado

Características:

- La BD es "dueña" de los datos
- Nos ayudará a sortear la limitación de la memoria física
- Tiene un overhead significativo de comunicación
- Pérdida de identidad de los objetos
- Permite aprovechar la transaccionalidad de la BD

Estrategias de Persistencia

Ambiente muerto y recreado

Características:

- La BD es "dueña" de los datos
- Nos ayudará a sortear la limitación de la memoria física
- Tiene un overhead significativo de comunicación
- Pérdida de identidad de los objetos
- Permite aprovechar la transaccionalidad de la BD

Estrategias de Persistencia

Ambiente muerto y recreado

Características:

- La BD es "dueña" de los datos
- Nos ayudará a sortear la limitación de la memoria física
- Tiene un overhead significativo de comunicación
- Pérdida de identidad de los objetos
- Permite aprovechar la transaccionalidad de la BD

Estrategias de Persistencia

Ambiente muerto y recreado

Características:

- La BD es "dueña" de los datos
- Nos ayudará a sortear la limitación de la memoria física
- Tiene un overhead significativo de comunicación
- Pérdida de identidad de los objetos
- Permite aprovechar la transaccionalidad de la BD

Estrategias de Persistencia

Ambiente muerto y recreado

Características:

- La BD es "dueña" de los datos
- Nos ayudará a sortear la limitación de la memoria física
- Tiene un overhead significativo de comunicación
- Pérdida de identidad de los objetos
- Permite aprovechar la transaccionalidad de la BD

Outline

- 
- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
 - 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
 - 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
 - 4 Transaccionalidad
 - **Transaccionalidad**
 - Persistencia y transaccionalidad

Transaccionalidad

Requerimiento: Confianza en que el estado es, en todo momento, correcto

Transacción Conjunto de transformaciones al estado del sistema que suponen una unidad lógica.

Transacción

Ejemplo

En nuestro juego **recoger un ítem desde un cofre y agregarlo a nuestro inventario** es una transacción.

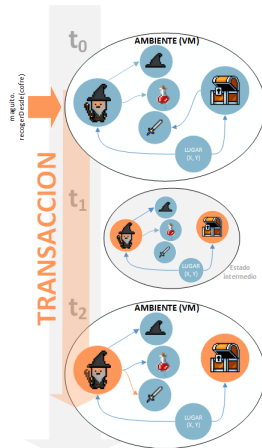
Supone varias pequeñas transformaciones:

- remover el ítem de la colección del cofre
- agregar el ítem al inventario del personaje

Pero una sola unidad de trabajo (unit-of-work)

Transacción

Ejemplo



Transacción

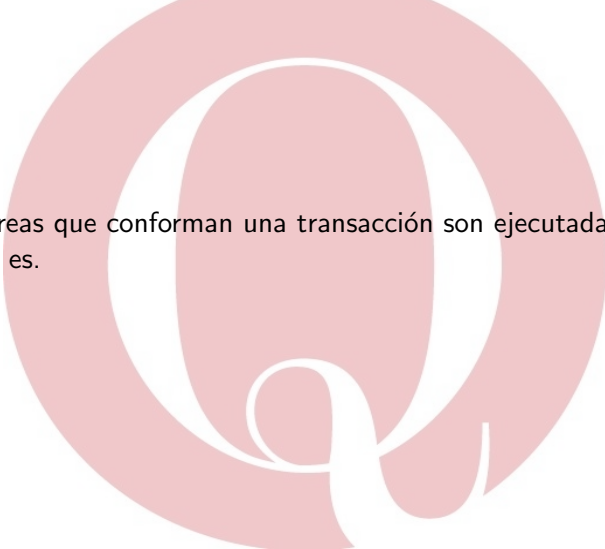
Propiedades ACID

Una transacción es **un conjunto de operaciones sobre las que se aseguran:**

- Atomicidad (**A**tomicity)
- Consistencia (**C**onsistency)
- Aislamiento (**I**solation)
- Durabilidad (**D**urability)

Atomicidad

Propiedades ACID



Todas las tareas que conforman una transacción son ejecutadas con éxito, o ninguna lo es.

Consistencia

Propiedades ACID

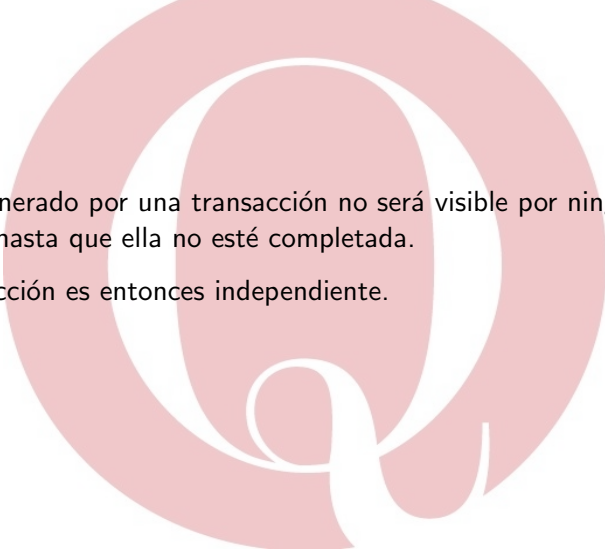
Una transacción debe cumplir con todas las reglas definidas por el sistema:

- Integridad de las relaciones
- Validaciones
- Restricciones

El estado de la aplicación cumple las reglas al principio y al final de cada transacción.

Aislamiento

Propiedades ACID

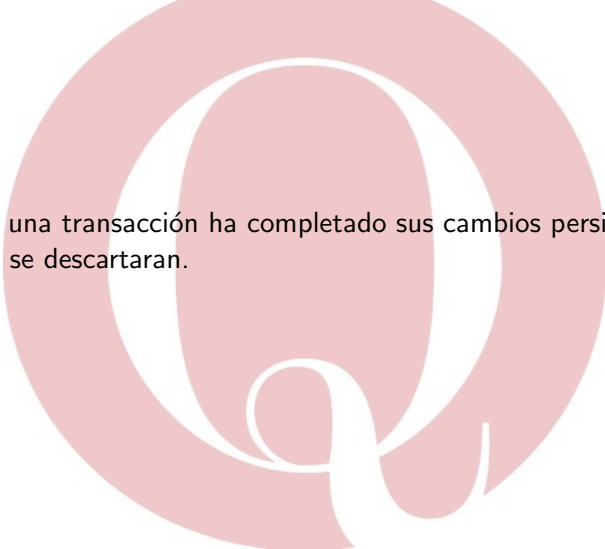


El estado generado por una transacción no será visible por ninguna otra transacción hasta que ella no esté completada.

Cada transacción es entonces independiente.

Durabilidad

Propiedades ACID

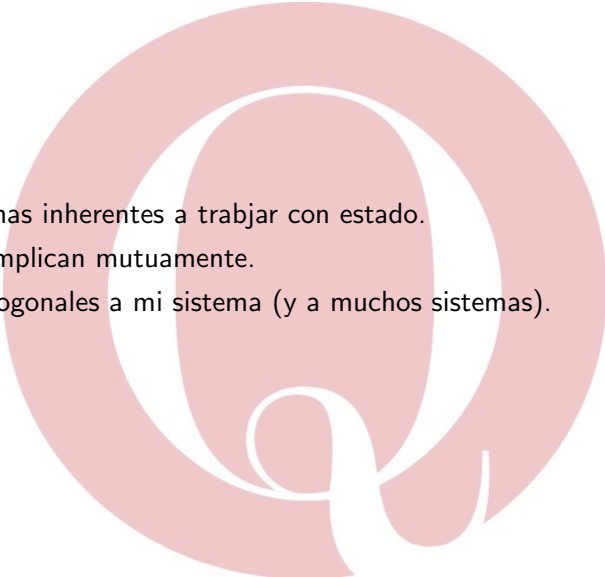


Una vez que una transacción ha completado sus cambios persistirán en el sistema y no se descartaran.

Outline

- 
- 1 Introducción
 - Aplicación y estado
 - Persistencia
 - Decisiones de diseño
 - 2 Medios persistentes
 - Snapshot
 - Serialización a archivos
 - Servidores de Bases de Datos
 - 3 Estrategias de Persistencia
 - Estrategias de Persistencia
 - Ambiente vivo + BD como backup
 - Esquema prevalente
 - Ambiente muerto y recreado
 - 4 Transaccionalidad
 - Transaccionalidad
 - **Persistencia y transaccionalidad**

Persistencia y transaccionalidad

- 
- Problemas inherentes a trabajar con estado.
 - No se implican mutuamente.
 - Son ortogonales a mi sistema (y a muchos sistemas).