

Actualización de servicios

Unidad 3: Actividades ***Actualización de servicios replicados***

Actividad 1

Revise los objetivos que debería cumplir cualquier mecanismo de actualización dinámica (enunciados en la Unidad 2) y justifique cuáles llegan a cumplirse y cuáles no en cada uno de estos sistemas:

1. Argus (Sección 3.1 de la Unidad 3).

2. PODUS (Sección 3.2 de la Unidad 3).

Actividad 2

Justifique si las siguientes afirmaciones sobre el sistema de actualización descrito en [[ALS06](#)] (resumido en la sección 3.3 de la Unidad 3) son verdaderas o falsas.

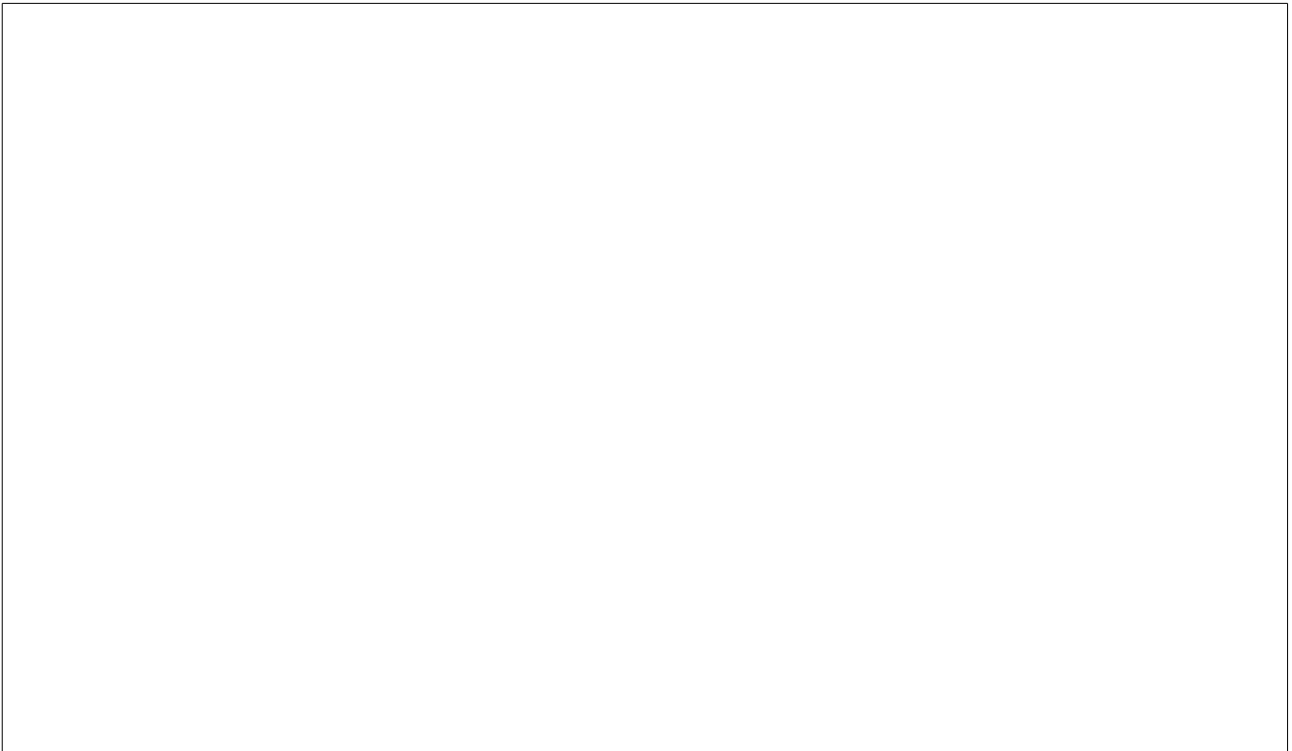
1. Internamente, el sistema no realiza una actualización inmediata multiversionada según la definición vista en la Unidad 2 ya que sólo existe una versión activa de cada componente. Tanto las versiones antiguas como las futuras, a pesar de aceptar invocaciones, son “simuladas” por diferentes proxies.

2. El usuario percibe una actualización inmediata multiversionada pues existe concurrencia entre las invocaciones ya lanzadas sobre versiones antiguas y las invocaciones iniciadas sobre versiones nuevas.

3. El sistema de actualización no es robusto. Surgirán incoherencias al permitir la ejecución concurrente de invocaciones sobre múltiples versiones de un mismo objeto.



4. El sistema no admite cambios en las interfaces de los módulos actualizados. Sólo se permiten actualizaciones que corrijan errores internos o que eliminen vulnerabilidades.



Actividad 3

Justifique si las siguientes afirmaciones sobre el sistema de actualización descrito en [[ALS06](#)] (resumido en la sección 3.3 de la Unidad 3) son verdaderas o falsas.

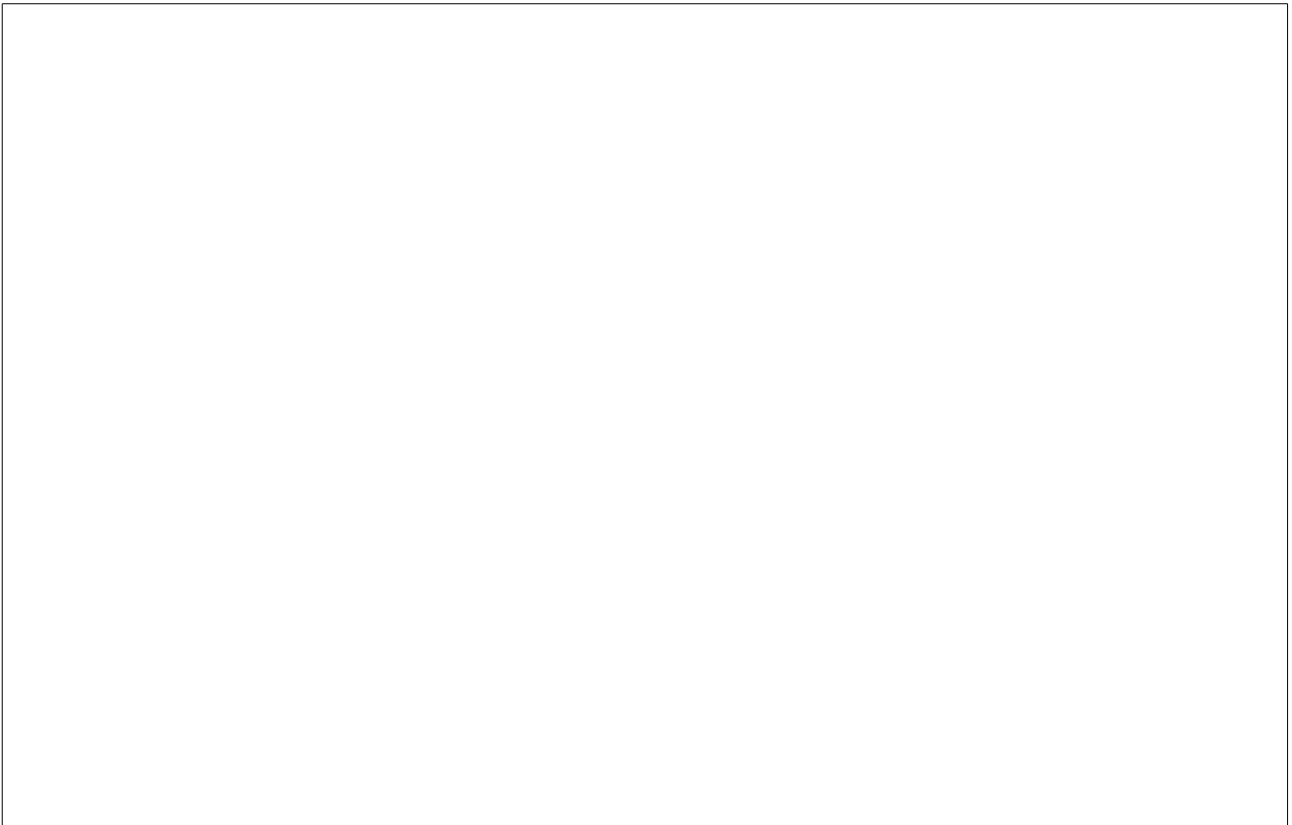
1. El sistema automatiza completamente la transformación de estado entre versiones consecutivas. El programador no debe preocuparse por esa transformación.

2. Su instante de actualización debe considerarse (al menos internamente) como diferido basado en inactividad. Para poder aplicar una función de transformación se requiere que la versión actual del objeto esté inactiva.

3. No hay que preocuparse por la “actualización de módulos dependientes” (métrica 2 de la Unidad 2) pues las versiones antiguas de un módulo actualizado pueden mantenerse perpetuamente.



4. Este sistema de actualización es óptimo pues jamás incurre en indisponibilidad y no necesita que los módulos a actualizar estén replicados.



Actividad 4

Describa qué complicaciones surgirán en el algoritmo de actualización descrito en la Sección 4.1 para el modelo de replicación activo en caso de que deba realizarse una transformación de estado entre dos versiones consecutivas.

Actividad 5

La estrategia de actualización del sistema Eternal (descrita en la Sección 4.2) proporciona un mecanismo para soportar cambios en la interfaz del módulo que se actualice y un marco adecuado para realizar transformaciones de estado.

Revise todas las métricas descritas en la Unidad 2 e indique qué alternativa utiliza Eternal en cada una de esas métricas. Revise el artículo citado en la Unidad 3 sobre Eternal en caso de duda.

--