Engineer Software

Concurrencia estricta Swift 6 sobre C

Adopté la concurrencia estricta completa de Swift 6 sin actores, conectando un bucle de eventos C bloqueante con el actor principal mediante un productor, un consumidor y un único orden, tras constatar que una tarea por evento pierde el orden del que depende la interfaz.

septiembre 2026

concurrencia estricta con un único orden y sin actores

Situación. La comprobación estricta y completa de concurrencia de Swift 6 convierte las condiciones de carrera de datos en errores de compilación en lugar de fallos intermitentes. Adoptarla frente a una biblioteca C es donde surge la dificultad: el bucle de eventos del núcleo es una llamada bloqueante que debe ejecutarse indefinidamente fuera del hilo principal, y los valores que devuelve son punteros sin ninguna garantía de concurrencia. El compilador no puede razonar sobre nada de ello y lo rechazará todo hasta que se describa explícitamente el límite.

Tarea. Había que activar la comprobación completa sin vías de escape, lo que implicaba diseñar el paso de un bucle C bloqueante al actor principal, en lugar de rodearlo de anotaciones para eludirlo.

Acción. El enfoque obvio es un actor por subsistema y se descartó por medición, no por gusto. Crear una tarea por evento entrante permite al entorno de ejecución programarlas en cualquier orden, y el flujo de eventos del núcleo está ordenado: un evento de mensaje modificado que adelanta al de creación del mensaje al que se refiere produce una interfaz que muestra una edición de algo que aún no existe. La reentrada de los actores empeora esto en lugar de mejorarlo, porque un actor puede suspenderse a mitad de un método y procesar otra llamada. Lo que lo sustituyó es deliberadamente sencillo: un hilo productor responsable del bucle bloqueante, un consumidor, una cola entre ambos y un único salto al actor principal al final. El orden se conserva porque hay exactamente un camino y nada adelanta a nada. Los tipos inseguros que cruzan ese límite se encapsulan en tipos cuya seguridad entre hilos se afirma en la

envoltura en lugar de darse por supuesta, y la afirmación se documenta con la razón que la sostiene: un hilo es propietario del puntero y lo copia antes de entregarlo.

Resultado. La aplicación compila con la comprobación estricta y completa de concurrencia sin supresiones, y el orden de eventos es una propiedad estructural, no una esperanza. El coste es que el diseño es menos paralelo de lo que podría ser: todo pasa por un consumidor y, si alguna vez se convierte en cuello de botella, la solución exigirá volver a determinar qué eventos pueden reordenarse con seguridad, que es precisamente el análisis que esto evitó.

Aplique esta experiencia a su proyecto

Utilice «Concurrencia estricta Swift 6 sobre C» como punto de partida para un reto similar en su organización. Los servicios que respaldan este trabajo están disponibles para su proyecto. La estructura del sistema se define primero sobre el papel para que las piezas encajen y los costes se conozcan antes de escribir código.

Explore nuestros servicios Póngase en contacto →

Mire más allá del resultado

Lea la situación, las decisiones y el resultado de este ejemplo. Podemos hablar de qué partes se adaptan a sus restricciones y cuáles necesitan otro enfoque. Concurrencia estricta Swift 6 sobre C — concurrencia estricta con un único orden y sin actores

Explore nuestros servicios Póngase en contacto →

¿Quiere concurrencia cuya seguridad pueda demostrar el compilador?

Converse sobre su proyecto con la ingeniera responsable de la entrega. Acordamos por escrito el alcance, la base del precio y el traspaso antes de empezar a trabajar.

Contexto de la función: Ingeniera de aplicaciones macOS. Se apoya en Arquitectura de soluciones, Ingeniería de backend, Ingeniería de frontend, Fiabilidad y copias de seguridad, Optimización del rendimiento y Pruebas y control de calidad. Y ya que está aquí, también se nos da bien Arquitectura de plataformas y soluciones, Desarrollo de backend y API y Dirección técnica y consultoría.

Póngase en contacto Explore nuestros servicios

Una parte de **una trayectoria más amplia** — y toda ella está en este sitio.

Cada entrada sigue la misma estructura: la situación, la tarea, lo que hicimos y lo que cambió.

Vea la trayectoria completa

Defina la mejora que necesita

Describa el flujo de trabajo, las interfaces existentes y los usuarios que necesitan el cambio. Podemos definir una entrega comprobable y un traspaso que permita a su equipo mantener lo entregado.

Vea cuánto cuesta Póngase en contacto →

¿Quiere concurrencia cuya seguridad pueda demostrar el compilador?

Pregunte por las decisiones detrás de «Concurrencia estricta Swift 6 sobre C». Hablará con las personas que pueden explicar el trabajo y valorar su propio proyecto.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://software.engineer.company/es/portfolio/adopted-swift-6-strict-concurrency-over-c/