Engineer

Parser zur Prüfung aller FFI-Aufrufe geschrieben

Einen Parser geschrieben, der den echten C-Header mit 7.308 Zeilen liest und jede Aufrufstelle, jede Enum-Konstante sowie final für jede pointerbesitzende Klasse prüft, nachdem ein handgeschriebener Platzhalterheader Aufrufe dreier entfernter Funktionen kompilieren und linken ließ, die zur Laufzeit abstürzten.

September 2026

7,308

Zeilen C-Header gelesen statt vorausgesetzt

Situation. Früh im Projekt beschrieb ein handgeschriebener Header die erwartete C-Schnittstelle. Er kompilierte, die Anwendung linkte, und Aufrufe dreier nicht mehr vorhandener Kernfunktionen erreichten die Laufzeit und stürzten ab. Compiler und Linker waren mit einer Beschreibung der Bibliothek statt der Bibliothek selbst zufrieden gewesen. Die Lücke zeigte sich erst bei Ausführung.

Aufgabe. Der echte Header mit 7.308 Zeilen und 268 Deklarationen musste maßgeblich werden. Jede Swiftverwendung sollte automatisch dagegen geprüft werden statt allein durch Review.

Massnahmen. Ein Parser liest den tatsächlichen Upstreamheader und ermittelt deklarierte Funktionen, Enum-Konstanten und Typen. Anschließend löst er jede Aufrufstelle und Konstantenreferenz des Swiftcodes dagegen auf. Ein Aufruf einer nicht deklarierten Funktion oder ein Bezug auf eine umbenannte Enum-Konstante lässt den Build scheitern. Derzeit werden 132 der 268 Deklarationen verwendet. Die 136 ungenutzten zu kennen ist nützlich, weil es den noch unerschlossenen Kernumfang genau beschreibt. Der Parser erzwingt auch eine vom Compiler nicht prüfbare Regel: Jede Swiftklasse mit eigenem Kernpointer muss final sein. Sonst könnte eine Unterklasse Deinitialisierung oder

Lebensdauer verändern und damit den Freigabezeitpunkt verschieben. Ein Zugriff nach Freigabe wäre möglich, ohne unsafe-Schlüsselwort in der Nähe. Die Regel wird namensbezogen im ganzen Baum geprüft.

Ergebnis. Die ursprüngliche Absturzklasse kann nicht wiederkehren: Veraltete Referenzen sind nun Build- statt Laufzeitfehler. Die Grenze liegt im Parserverständnis. Er versteht Headerdeklarationen, nicht Semantik. Eine vorhandene Funktion mit passendem Namen besteht die Prüfung auch dann, wenn upstream Bedeutung oder Besitzregel bei gleichbleibender Signatur geändert wurden.

Diese Erfahrung für Ihr Projekt nutzen

Nutzen Sie „Parser zur Prüfung aller FFI-Aufrufe geschrieben“ als Ausgangspunkt für eine ähnliche Herausforderung in Ihrer Organisation. Die Leistungen hinter dieser Arbeit stehen auch für Ihr Projekt zur Verfügung. Der Teil, den Nutzer nie sehen: wo die Daten liegen, welche Regeln gelten und wie andere Systeme darauf zugreifen dürfen.

Unsere Leistungen entdecken Kontakt aufnehmen →

Hinter das Ergebnis schauen

Lesen Sie die Situation, die Entscheidungen und das Ergebnis dieses Beispiels. Wir können besprechen, welche Teile sich auf Ihre Rahmenbedingungen übertragen lassen und welche einen anderen Ansatz erfordern. Parser zur Prüfung aller FFI-Aufrufe geschrieben — 7,308 Zeilen C-Header gelesen statt vorausgesetzt

Unsere Leistungen entdecken Kontakt aufnehmen →

Würde Ihr Build den Aufruf einer entfernten Funktion erkennen?

Besprechen Sie Ihr Projekt mit der Ingenieurin, die für die Umsetzung verantwortlich ist. Wir vereinbaren Umfang, Preisgrundlage und Übergabe schriftlich, bevor die Arbeit beginnt.

Rollenkontext: macOS-Anwendungsingenieurin. Die Arbeit baut auf Backend-Entwicklung, APIs und Integration, Automatisierung und CI/CD, Sicherheit, Tests und Qualitätssicherung und Dokumentation. Und wenn Sie schon hier sind: Wir sind auch ziemlich gut in Backend- und API-Entwicklung, DevOps- und CI/CD-Automatisierung und Technische Dokumentation.

Kontakt aufnehmen Unsere Leistungen entdecken

Ein Teil einer **längeren Arbeitsbilanz** — vollständig auf dieser Website zu finden.

Jeder Eintrag folgt demselben Aufbau: die Situation, die Aufgabe, unser Vorgehen und die Veränderungen.

Alle Arbeiten ansehen

Die benötigte Verbesserung definieren

Beschreiben Sie den Arbeitsablauf, die vorhandenen Schnittstellen und die Nutzer, die die Veränderung benötigen. Wir können eine prüfbare Lieferung definieren und die Übergabe so gestalten, dass Ihr Team die gelieferten Bestandteile warten kann.

Preise ansehen Kontakt aufnehmen →

Würde Ihr Build den Aufruf einer entfernten Funktion erkennen?

Fragen Sie nach den Entscheidungen hinter „Parser zur Prüfung aller FFI-Aufrufe geschrieben“. Sie sprechen mit den Menschen, die die Arbeit erklären und Ihr eigenes Projekt einschätzen können.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

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

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/portfolio/wrote-a-parser-verifying-every-ffi-call/