Symflower - AQA

Symflower - Automating Quality Assurance

Programm / Ausschreibung	BASIS, Basisprogramm, Budgetjahr 2019	**Status**	abgeschlossen
Projektstart	01.10.2018	**Projektende**	29.02.2020
Zeitraum	2018 - 2020	**Projektlaufzeit**	17 Monate
Keywords			

Projektbeschreibung

In der Softwareentwicklung spielt die Qualitätssicherung eine zentrale Rolle. Systemabstürze, Datenverluste und Sicherheitslücken sind beispielsweise unbedingt zu vermeiden. Die Fehlerfreiheit eines Systems lässt sich jedoch nur durch ausführliches Testen sicherstellen. Heutzutage gibt es hierfür zwei gängige Methoden: Entweder ein vorliegendes System wird manuell getestet, das bedeutet ein Mitarbeiter testet die Software durch ausprobieren, oder es werden von Experten automatisierte Tests geschrieben, welche regelmäßig ausgeführt werden können. Beide Varianten sind äußerst zeitaufwändig und betragen 40-50% der gesamt Projektressourcen.

Unser zukünftiges Produkt Symflower AQA soll vorliegende Softwaresysteme vollautomatisch analysieren, mit dem Ziel, automatisiert Tests mit hoher Abdeckung zu generieren. Unsere Kunden müssen daher nicht länger Kompromisse bezüglich der Qualität ihrer Produkte und dem investierten Testaufwand eingehen. Mit jeder Änderung im Quelltext werden Analysen gestartet und somit Tests generiert, welche das gesamte System zur Ausführung bringen. Durch die anschließende Ausführung der Tests können Abweichungen von der Spezifikation, Laufzeitfehler, Speicherzugriffsfehler sowie Performance-Engpässe aufgezeigt werden.

Ziel dieses Projektes, und somit des beantragten Forschungsjahres, ist die Entwicklung einer sprachunabhängigen Symbolic Execution Engine, nachfolgend als “Symflower SE” bezeichnet, welche die gezielte Generierung von Unit-Tests ermöglicht und die nachfolgenden Ziele verfolgt:
- Eine sprachunabhängige Implementierung, die es erlaubt eine Vielzahl an Sprachen zu unterstützen: In Entwicklungsfirmen wird heutzutage, anstatt einer einzelnen Entwicklungssprache, zumeist eine Vielzahl unterschiedlicher Sprachen eingesetzt. Dies macht es notwendig ein Testprodukt anzubieten, dass ein möglichst breites Spektrum an Sprachen unterstützt, um einen möglichst großen Kundennutzen bieten zu können.
- Generierte Eingabewerte sollen möglichst deterministisch und nachvollziehbar sein: In unseren Kundengesprächen wurde uns gegenüber vermehrt der Wunsch geäußert, dass die generierten Eingabewerte nicht zufällig wirken sollen. Es soll daher zu einem gewissen Grad nachvollziehbar sein, wie die berechneten Werte zustande gekommen sind. Zusätzlich sollten bei einer erneuten Generierung zudem immer die selben Werte berechnet werden.
- Unterstützung von Snapshots: Snapshots führen zu erhöhter Usability, da diese Zugriff auf Zwischenergebnisse alter

Analysen ermöglichen. Dies erlaubt das Wiederverwenden von erbrachten Rechenleistungen, wodurch rascher neue und gegebenenfalls umfangreichere Analyseresultate geliefert werden können.
- Parallelisierbarkeit lokal und im Clusterbetrieb: Auf eine lokale Parallelisierbarkeit sowie im Clusterbetrieb wird geachtet, um Analyseresultate in für Benutzer annehmbarer Zeit zur Verfügung stellen zu können. Dies ist insbesondere nötig, da die Algorithmen die für eine Symbolic Execution nötig sind, äußerst rechenintensiv sind. Eine lokale Parallelisierung erlaubt zudem die effiziente Nutzung von modernen Computerarchitekturen.
- Abgreifen und Anpassen von Interna: Die entwickelte Symbolic Execution Engine soll es erlauben auf interne Analyseresultate zugreifen zu können. Dazu zählen insbesondere die Wertebereiche von Variablen für einen speziellen Pfad. Dies erlaubt dem Benutzer eine bessere Nachvollziehbarkeit der berechneten Eingabewerte. Zusätzlich, stellt der Zugriff auf interne Analyseresultate eine Basis dar, um zusätzliche Metriken und Checks verfügbar machen zu können.

Projektpartner

- E&M Software Service GmbH