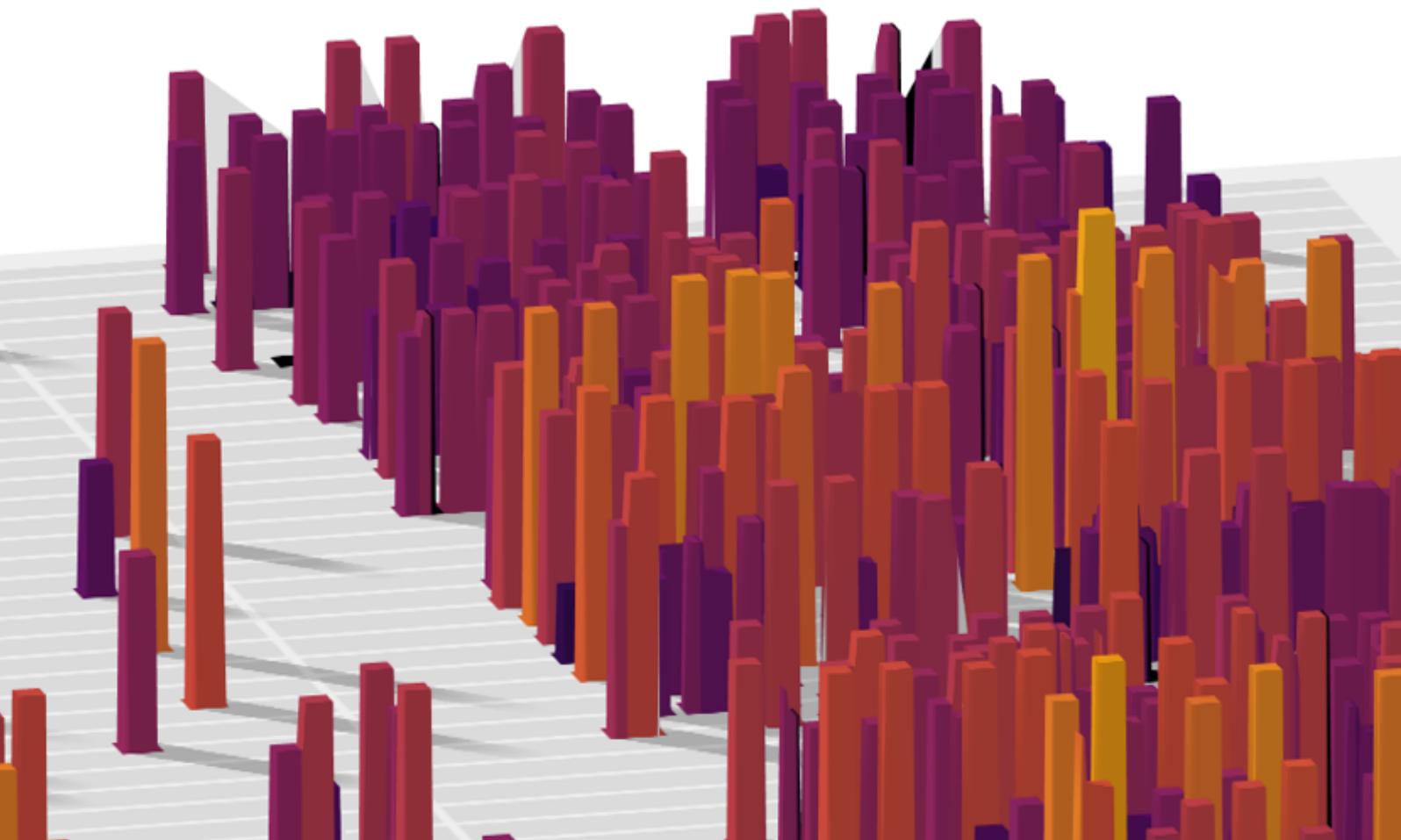




Relevante Softwaretests identifizieren und deren Ergebnis vorhersagen

scryer® ist ein KI-System, das mit einer Genauigkeit von über 70% den Ausgang von Softwaretests vorhersagt. Diese Vorhersagen erlauben signifikante Einsparungen von Rechen- & Entwickler-Ressourcen.



scryer[®] ermittelt für Softwareänderungen automatisch kritische oder wahrscheinlich fehlschlagende Tests sowie einen Konfidenzwert für die Vorhersage. Basis dafür sind historische Source-Control- und Testdaten.

Funktionen & Vorteile

So beschleunigt und vereinfacht scryer die Softwareentwicklung:

Sekundenschnelles Feedback zu potenziellen Fehlern bei einer Software-Änderung mittels IDE-Plugin

Noch vor der Ausführung langlaufender Test-Suites können Entwickler kritische Tests ausführen, ggf. den Code anpassen und einen Feedback-Zyklus einsparen.

Dynamische Test-Coverage

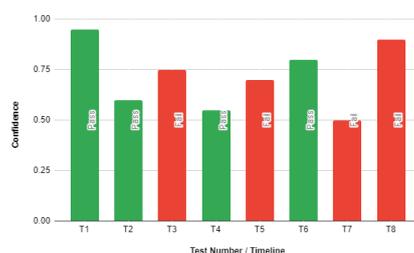
Das Mapping von „angefassten“ Dateien/Modulen zu relevanten Tests ist komplex, oftmals laufen zu wenige oder zu viele Tests. scryer ermöglicht, relevante und unwichtige Tests für jede Software-Änderung einfach und individuell zu bestimmen. So werden kritische Tests nicht übersehen und redundante Tests eingespart.

Vereinfachte Wartung von Test-Suites

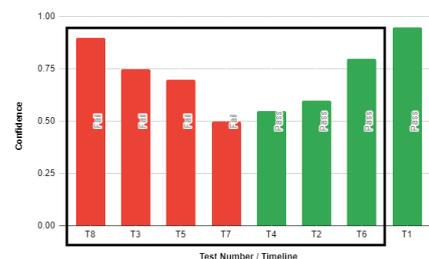
Selbst für Experten ist die Pflege der Test-Suites schwierig und zeitaufwändig. scryer unterstützt diese Arbeit zur Laufzeit durch präzise, auf die Software-Änderungen abgestimmte Vorhersagen.

Anpassung der Testreihenfolge

Test-Suites können so umsortiert werden, dass kritische Testfälle priorisiert laufen und potenzielle Fehler entsprechend früher identifiziert werden. Insbesondere bei Pre-Merge-Tests werden Wartezeiten so signifikant verkürzt.



Ursprüngliche Reihenfolge d. Testausführung



Optimierte Reihenfolge d. Testausführung

Pilotprojekt

scryer kommt bei Siemens Healthineers in der Software-Entwicklung zum Einsatz. Hier wird es für die Testvorhersage aus der IDE der Entwickler verwendet.

Training

Für das Training des Systems stehen mehr als 70.000 Testfälle, 400.000 Commits und über 700.000.000 Testausführungen zur Verfügung. Durch relevante Testevents lernt scryer deren Komplexität und Korrelationen verstehen.

Sekundenschnelles Feedback für Commit in MS Visual Studio

Ein neu entwickeltes, auf scryer basierendes Plugin für MS Visual Studio ermöglicht es den Entwicklern, direkt Feedback zu einem geplanten Commit zu erhalten. Binnen Sekunden werden kritische, wahrscheinlich fehlschlagende Tests für die Änderungen an der Software angezeigt.

Ergebnis Durch die präzisen Vorhersagen scryers werden messbar Kapazitäten eingespart und die Aufnahme neuen Codes wird beschleunigt. Zudem entlastet das System die Entwickler von monotonen Routinearbeiten.

Die Genauigkeit der Vorhersage für fehlschlagende Tests liegt bereits über 75%. scryer wird nun zusätzlich für die Testauswahl in die CI-/CD-Pipelines integriert.

- Daten** Die notwendigen Daten zum Training von scryer werden aus Source-Control- und Test-Management-Systemen exportiert; z. B. git, TFS (bzw. Azure DevOps Server), IBM Engineering Lifecycle Management (ELM) oder anderen ALM-Systemen. **scryer benötigt nur Meta-Daten**, es muss kein Source Code importiert werden.
- REST-API** Für den Daten-Import in das scryer-System steht eine REST-API zur Verfügung. Vorhersagen zu Software-Tests und deren Ergebnis können ebenfalls per REST-API abgefragt werden.
- Training** Nach dem initialen Import validiert scryer die Daten. Anschließend werden Vorhersage-Modelle trainiert und es erfolgt eine Analyse mit Blick auf die Vorhersage-Qualität basierend auf Cross-Validation.
- Updates** Das Update der Vorhersage-Modelle erfolgt im laufenden Betrieb, dafür werden lediglich neu angefallene Source-Control- und Testdaten eingespielt.
- Installation** scryer kann entweder als On-Site-Installation oder als Cloud-Lösung zur Verfügung gestellt werden.

Ihr Ansprechpartner

Haben Sie Fragen zu scryer?

Wir beraten Sie gerne.



Dr. Klaus Geyer

kge@codemanufaktur.com

Weitere Informationen:

<https://scryer-ai.com>

Youtube: [OOP 2001 Vortrag zu scryer](#)

REST-API-Beschreibung anfordern

bei kge@codemanufaktur.com

