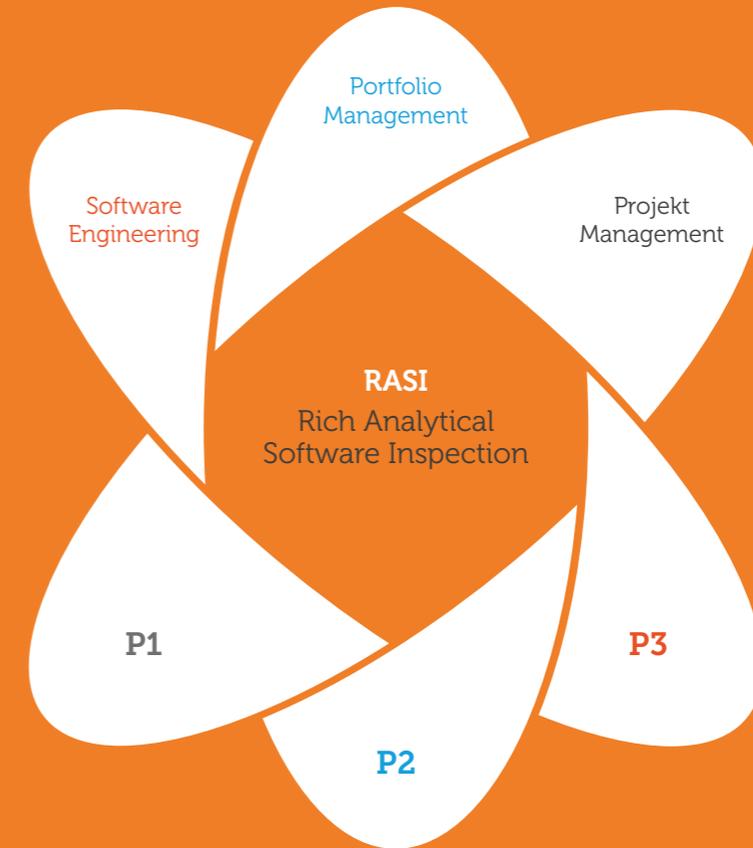
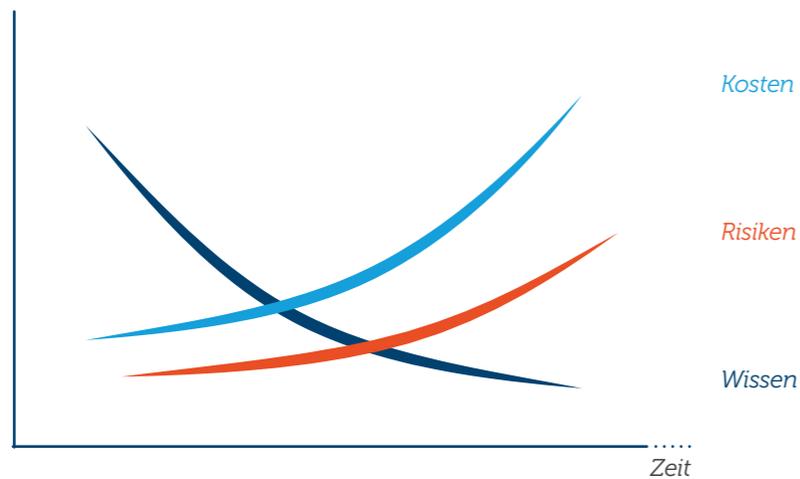


Rich
Analytical
Software
Inspection

„In God we trust.
All others bring data.“

W. Edwards Deming





Problemfeld P1

Komplexe EA-Portfolios

- Sie verantworten ein Portfolio von Eigenentwicklungen?
- Sie behüten eine undefinierte Fülle an Code-Artefakten und wollen deren Verwendung, Wert und Risiko bestimmen?
- Sie outsourcen Entwicklungen oder wollen Kandidaten identifizieren?
- Sie kämpfen gegen eskalierende Wartungs- & Betriebskosten?

Ein EA Portfolio kann so schnell wachsen, dass es zur überschnellen Alterung kommt.

Die Kosten und Risiken durch Fluktuation, Migration oder schlichtes Vergessen, können zu einem regelrechten Code-Sterben führen.

Gleiches gilt für Monitoring, Diagnose, Wartung, Betrieb, Support. Alternde Applikationen zehren exponentiell das IT-Budget auf. Das lässt wenig Spielraum für neue Unterstützung des Business.

„Applications, which appeared to satisfy the basic functional requirements through user acceptance test, can proof to be very expensive to support, modify or operate over the rest of their lives [...] because of poor coding standards with too much unnecessary code complexity.“

Gartner RAS Core Research 2013

finde den Fehler





Problemfeld P2

IT-Projekte

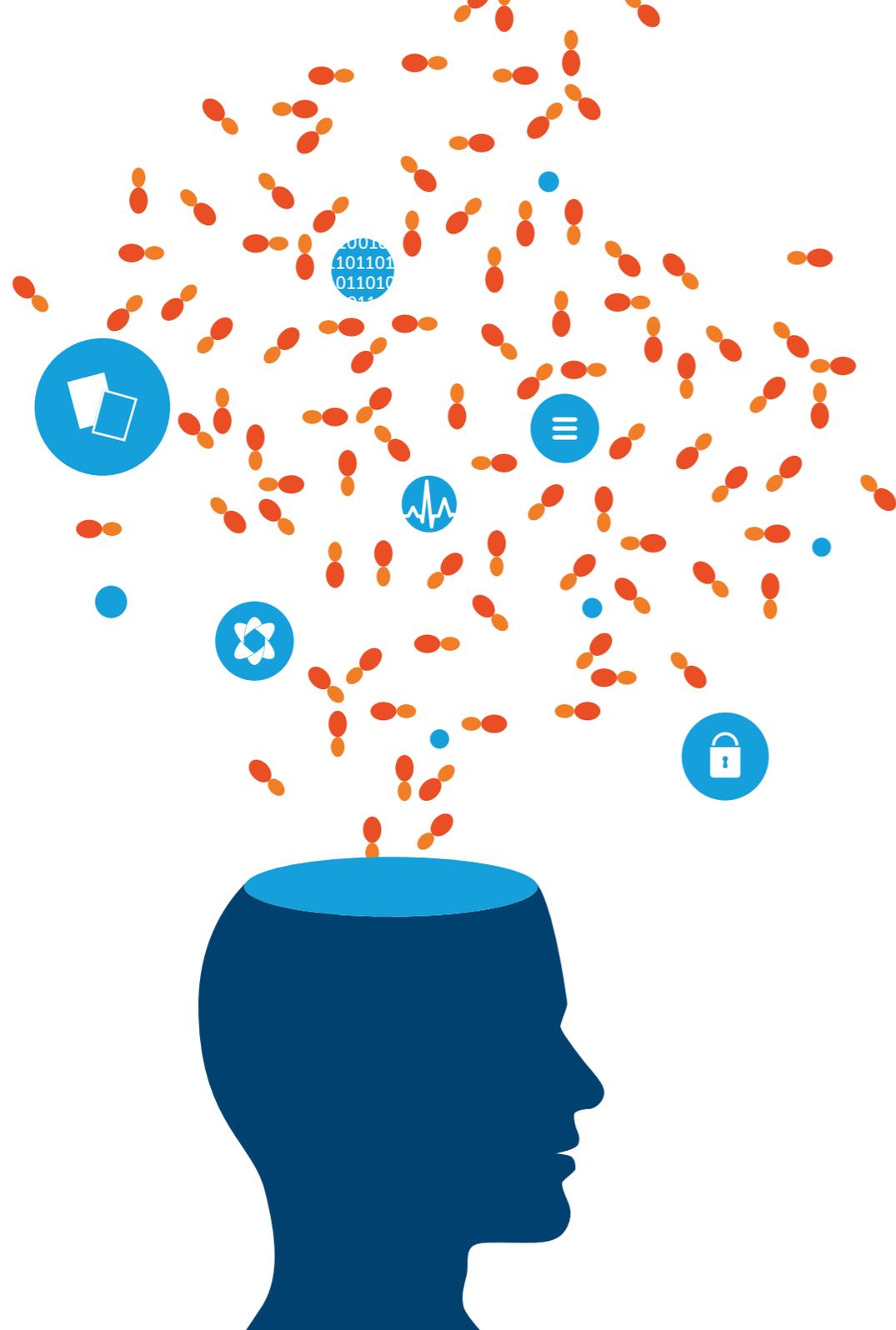
Der Grad der funktionalen Fertigstellung wird genau überwacht. In Bezug auf die qualitativen Anforderungen des Systems hingegen befinden sich die meisten Projekte über einen langen Zeitraum im fahrlässigen Blindflug. Hierbei sind seltener funktionale Defekte der Grund für spektakuläre Pleiten von Software-Engineering-Projekten. Viel öfter sind es die atemberaubend verfehlten nicht-funktionalen Anforderungen:

- Sie steuern ein Software-Entwicklungsprojekt?
- Sie entscheiden, an welcher Stelle die knappen Ressourcen am dringendsten benötigt werden?
- Sie verantworten die Abnahme von Lieferungen Dritter?
- Sie warten und betreiben Enterprise Applikationen?
- Sie balancieren das Projekt zwischen Kosten und Risiken?

- Zuverlässigkeit
- Performance
- Sicherheit
- Adaptionfähigkeit
- Wart- und Betriebbarkeit

Und oft realisieren die Verantwortlichen das erst wenn die Systeme langsam werden, Fehler produzieren, Daten korrumpieren oder knirschend zum Stillstand kommen.





Problemfeld P3

Inhouse und Outsourced Software Engineering

- Sie entwickeln eigene Software-Systeme?
- Sie verantworten die Abnahme von Lieferungen Dritter?
- Sie sind Technical Lead oder Architekt?
- Sie verantworten Repositories mit zehntausenden von Artefakten?

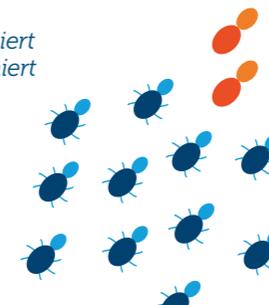
Es gibt qualifizierte Programmierer auf der ganzen Welt: in Osteuropa, Fernost, Afrika und im Büro nebenan. Architekturen und Guidelines sind die Spielregeln, die helfen sollen, alle unter einen Hut zu bekommen. Nur sehen sich Entwickler oft als Architekten in ihrer eigenen Domäne. Sie implementieren Architekturen und Guidelines solange sie ihnen passen und arbeiten um diese herum, dann wenn es ihnen notwendig erscheint.

Wissen Sie denn wirklich was Ihre Entwickler tun? Das Kernproblem ist: Jede nicht oder spät entdeckte Nachlässigkeit im Code kostet Sie viel Geld.

Wie teuer kommt z.B.:

- ein Memory- oder Security-Leak, ein Bottleneck, ein Lock, Racing Prones?
- überkomplizierte, schlecht oder schlicht fehleranfällige Schnittstellen?
- eine fehlerhafte / unerwünschte Library?
- eine mehrfach oder nachlässig umgesetzte Operation, Abwrack-Code oder ein vergessenes Juwel?
- eine über das gesamte System verteilte Protokollierung oder Fehler-, Berechtigungsprüfung?
- doppelte, inkonsistente, verlorene oder gestohlene Daten?

verprogrammiert
vorprogrammiert





RASI
Rich Analytical Software Inspection

Wir untersuchen die Struktur einzelner Systeme oder ganzer Landschaften anhand der vorliegenden Artefakte. Das gibt Ihnen „real facts“ für Ihre Entscheidungen bzgl. Risiko, Qualität und Allokation von Ressourcen.

ISO25021 spezifiziert. Das Ergebnis: ein Bericht, mit dem Sie Ihre Qualitätsziele definieren, über die Zeit Bereiche detaillieren, Fortschritt messen und schließlich Ergebnisse beurteilen können.

Dazu setzen wir einen Fundus an Werkzeugen ein, mit denen unsere Experten alle wichtigen Programmier- und Skriptsprachen, Konfigurationen, Datenbanken, Protokolle und Ablaufumgebungen analysieren können. Die > 500 von uns ermittelten QMEs sind größtenteils nach Maßgabe der

Und das objektiv, wiederholbar, konsistent, standardisiert und vergleichbar. Damit Sie faktenbasierte Entscheidungen treffen können.

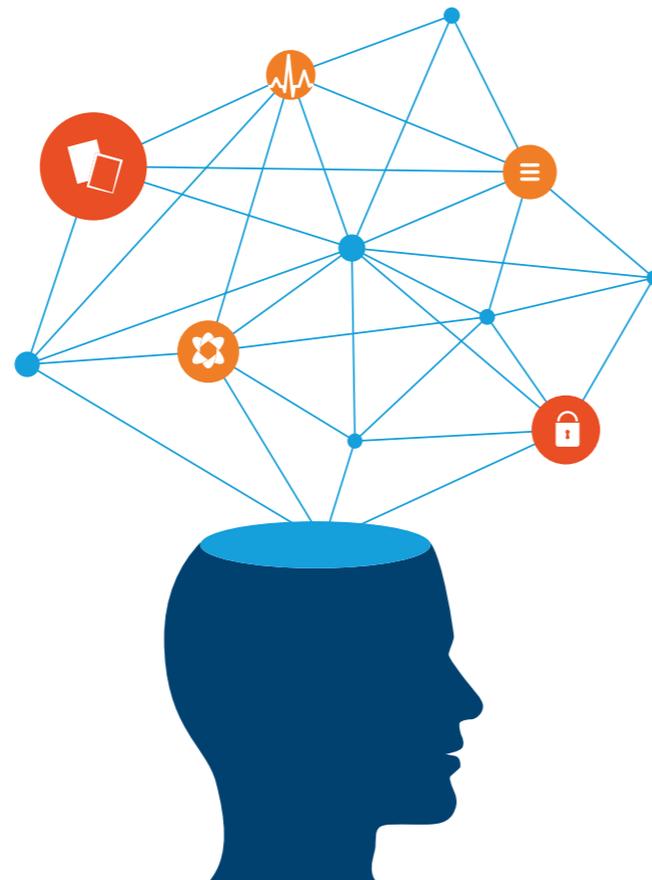


Not Just a Tool, it's a Lab!

Ihre Vorteile

Rich Analytical Software Inspection (RASI) ist:

- **Sofort umsetzbar**, da Sie keine Werkzeuge einführen oder Wissen darüber aufbauen müssen.
- **Effektiv**, da Sie aus einem Fundus an Wissen und Erfahrung unserer Experten über Methoden, Patterns und Werkzeugen schöpfen können.
- **Sicher**, da wir im Labor für die Vertraulichkeit und Sicherheit Ihres Codes bürgen.
- **Unabhängig** von Ihrem Software-Entwicklungsprozess und damit unabhängig von den eingesetzten Werkzeugen und Ihrer Technologie.
- **Umfassend**, da sie auf nahezu alle Artefakte und in verschiedenen Phasen des Lebenszyklus angewendet werden kann.
- **Wertvoll**, da sie die Steuerungsfähigkeit über Ihre IT-Delivery erhöht und die Aufmerksamkeit lenkt.



RASI verbessert und fördert die Fähigkeit zur:

- **Bewertung** der Anwendungslandschaft, da der Bestand systematisch klassifiziert, das Nutzungs- und Fehlerprofil empirisch ermittelt und das QS-Ziel für ein System oder Artefakt spezifisch festgelegt werden kann.
- **Projektsteuerung**, da eine Reihe von Entscheidungen fundiert getroffen, Ziele objektiv definiert, Maßnahmen beurteilt und Ressourcen besser allokiert werden können.
- **Kommunikation**, da die Aufmerksamkeit, Aussagefähigkeit und das Mandat gegenüber anderen Stakeholdern gestärkt wird.
- **Abnahme**, da Qualitätsanforderungen besser formuliert und bestimmt werden können; beispielsweise bei Abnahme von Leistungen Dritter, der Erstellung von Gutachten und der Vorbereitung auf Audits.
- **Abwendung** von Risiken, da Schwachstellen besser identifiziert, Wartungsaufwände prognostiziert und Fehler früher behoben werden können.
- **Investitionssicherung und Erhaltung** von geistigem Kapital, da Vorgaben besser durchgesetzt und Fluktuation im Team besser kompensiert werden können.

Unser Angebot

Wir gewinnen für Sie umfassende und nützliche Qualitätsmetriken nach Maßgabe der ISO25021

- aus der Struktur
- dem Laufzeitverhalten
- der Historie eines Systems

Wir messen QMEs und aggregieren diese in nützliche QMs. So entstehen Metriken und Indikatoren zur Beurteilung von

- Laufzeitverhalten
- Risiken
- Struktureller Qualität

Wir suchen nach bekannten Patterns, Bibliotheken und Schwachstellen

- in allen relevanten Artefakten
- über alle Layer
- Komponenten eines Systems

Wir prüfen Ihre Codebasis auf Konformität mit

- Architekturvorgaben
- Guidelines, Best-Practices
- Reifegrad

Wir analysieren die Nutzungs- und Änderungsraten aus Historieninformation von

- Code-Repositories
- Datenbanken
- Log-Dateien

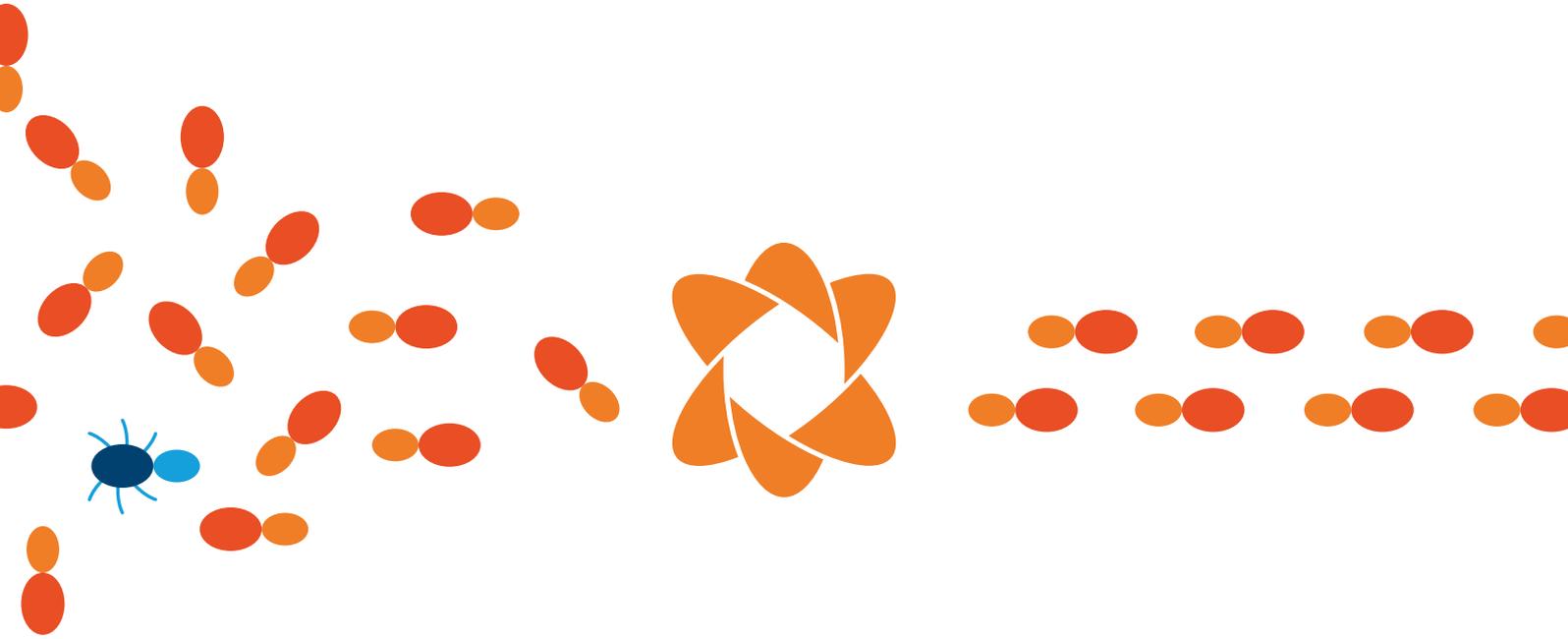
Wir überwachen Ihre Systeme in Bezug auf

- Performance
- Laufzeitfehler
- Anomalien

Wir werten Laufzeit- und Fehlerprotokolle aus nach

- statistischen Merkmalen
- Korrelationen
- Fehlermustern





COMPUTER AIDED
BUSINESS PROCESSES

www.cabp.de