Studienarbeits- und Diplomthemen: XCTL-Projekt

(K. Bothe und K. Schützler)

13. Februar 2002

Inhaltsverzeichnis

1 Aufbereitung der Komponente *Topographie* des XCTL-Systems mit Methoden des Reverse Engineering, Restructuring und Reengineering (Studien- und Diplomarbeit)

1.1 Review und Überarbeitung der *Verhaltensspezifikation (Pflichtenheft)* zur Topographie:

- neue Gliederung
- strenge Kontrolle der Informationen: alles korrekt, vollständig, eindeutig?
- Lesbarkeit dahingehendund logischen Aufbau kritisch betrachten und verbessern

1.2 Nutzerdokumentation und Online-Hilfe

1.3 Umgebungssimulation der Topographie (simulierte Abarbeitung)

1.4 Test:

- Testfälle mit dem cte
- Grobtestfälle (minimale Menge für Regressionstest)
- jeweils: erwartete Testergebnisse
- Testdurchführung (Hilfsmittel: Umgebungssimulator) (auch: Feldtest)
- Testprotokoll: Vergleich der erwarteten mit beobachteten Ergebnissen
- Regressionstest: Automatisierung der Folge aller Testläufe

1.5 Software-Architektur (Design / Entwurf)

- Logische (und physische) Trennung von Interface und Implementation des Subsystems
- Wrapper für die Topographie (Objektorientierte Schnittstelle der Topographie)
- Trennung von bisher gemeinsamen Teilen mit der Diffraktometrie (was wird tatsächlich gemeinsam genutzt - was kann abgetrennt werden)
- hierzu zur Abgrenzung: Trace-Tools (Alexander Paschold, Rene Harder, Jan Picard: nutzen und weiterentwickeln ?)
- Dokumentation der inneren Struktur mit Klassendiagrammen
- Dokumentation dynamischer Abläufe mit UML-Diagrammen (Sequenzdiagramme u. a.)
- kritische Betrachtung und Verbesserung der Struktur (Klassen und ihre Beziehungen) - Restructuring

- Implementation der Topographie: durcharbeiten, ggf. weiter kommentieren, ggf. weitere Fehler festhalten (in Quellen und Fehlersammlung)
- Fehler beheben und Test (Hilfsmittel: Umgebungssimulator, ermittelte Testfälle)

1.6 Literatur, Dokumente, Dateien:

- Marlies Gollnick: Studienarbeit (Reverse Engineering der Topographie im RTK-Steuerprogramm)
- Sebastian Freund, Derrick Hepp: Diplomarbeit zur automatischen Justage
- Kay Schützler: Diplomarbeit zur Subsystemermittlung
  (hat Kenntnisse zur allgemeinen Ablaufsteuerung und inhaltlichen Aufteilung der Dateien)
- Makrosprache (Web-Seite: Entwicklerdokumente von Kay Schützler)
- Umgebungssimulation für Motoren (Stephan Lützkendorff) und 0-dimensionale Detektoren (Kay Schützler)

2 Messung von Überdeckungsgraden für ausgewählte Komponenten ("White-Box"-Test) zur Beurteilung der Vollständigkeit von funktional gewonnenen Testdatenmengen (Studien- und Diplomarbeit)

S. Lützkendorff (E-Mail vom 11. 1. 2001):
auf der Suche nach Frage- und Aufgabenstellungen die aus der allgem. Aufgabenstellung 'Test' herauslösbar sind, habe ich im Moment nur einen Punkt finden können: und zwar die Bestimmung der WhiteBox- Testüberdeckung (Anweisungs-, Zweig-, Pfadüberdeckung, ...).

Das ist so sicher nicht direkt ein Thema zum Reverse Engineering. Wenn man aber überhaupt die Möglichkeiten Programmabläufe zu protokollieren untersucht und für solche Protokolle Auswertungsmöglichkeiten sucht, kann man dieses Thema als Fortsetzung der Arbeit, die mit dem Trace-Tool begonnen wurde, betrachten. Ich kann mir auch gut vorstellen, dass es bereits Tools zur Instrumentierung gibt, auf denen man aufsetzten kann.

Für den Test in unserer Reverse Engineering-Situation halte ich die Bestimmung dieser Testüberdeckung für wichtig, da ja die fehlende Spezifikation durch das Verhalten des Altsystems ersetzt wird, d.h. das Testprotokoll dass das unveränderte Altsystem liefert ist im ersten Schritt das Sollprotokoll und damit Teil der Verhaltensspezifikation. Zur Beurteilung, ob aber alle Teile der Software hinreichend zu diesem Sollprotokoll beitragen, ist die WhiteBox-Testüberdeckung ein sehr wichtiges Hilfmittel. Das heißt wiederum, dass ein solches Tool schon irgend wann während meiner Arbeit zum Einsatz kommen müsste.

Ich hoffe das hilft Ihnen bei der Definition einer Aufgabenstellung.

beste Grüße Stefan Lützkendorff

3 Skriptsprache: Erneuerung der Makrosprache und Erweiterung um Zyklen und Bedingungen (Diplomarbeit)

Eine Erweiterung der Skriptsprache erscheint nicht unbedingt sinnvoll, da die jetzige Einbindung der Makros viel zu festgelegt erfolgt. Es wäre sinnvoll, die Benutzung der Makros auf eine neue Grundlage zu stellen. Es müsste die Möglichkeit geschaffen werden, beliebig benannte Makros zu laden und auszuführen.

Dabei würde eine Benutzerschnittstelle zur Ablaufsteuerung (mit Zugriff auf die Detektor- und die Motor-DLL) entstehen, die eventuell ein eigenständiges Programm mit neuen Einsatzmöglichkeiten darstellen kann. Der Bedarf dafür scheint bei den Physikern vorhanden zu sein. Nachdem diese Schnittstelle existiert, kann über die Erweiterung der Skriptsprache nachgedacht werden.

4 Quellcodebereinigung: Entfernung von totem Code in den XCTL-Quelltexten (Studienarbeit)

Die Quellen des XCTL-Programms stellen eine historisch gewachsene Struktur dar. Dabei hat der ursprüngliche Entwickler nur sehr wenig Aufmerksamkeit auf die Beseitigung nicht mehr benötigter Code-Teile verwendet. Daher sind in den jetzigen Quellen zahlreiche unerreichbare Stellen vorhanden, die das Verständnis und die Übersicht unnötig erschweren.

Im Rahmen einer Studienarbeit soll untersucht werden, welchen Anteil einzelne wiederholt auftretende Strukturen - wie zum Beispiel case-Zweige mit niemals zutreffenden Bedingungen - am toten Code haben. Weiterhin sollen die erkannten Problemstellen durch Entfernen der unerreichbaren Quellcode-Teile bereinigt werden.