

EasiRun/SMART. Eine regelbasierte Plattform zur Analyse und Modifikation von Quellcode

Was bedeutet „SMART“?

Smart ist heute ein weitverbreiteter Modebegriff. Alles ist heute irgendwie „smart“. Für uns hat „SMART“ allerdings eine sehr konkrete Bedeutung:

- S** Source Code
- M** Modification
- A** Analysis
- R** Reengineering
- T** Transformation

Für wen ist EasiRun/SMART interessant?

Oft stecken in individuellen Softwarelösungen Jahre oder Jahrzehnte Entwicklung. Neuentwicklungen sind oft teuer und unkalkulierbar. Risikoärmer und kalkulierbarer ist oft eine Modernisierung bzw. eine Migration der Anwendung. Standardlösungen für die Migration von Individualsoftware sind oft nicht vorhanden oder falls angeboten, berücksichtigen diese in den wenigsten Fällen deren individuelle Besonderheiten. So liegt es nahe, eine Modernisierungslösung zu entwerfen, die genauso individuell ist wie die zu modernisierende Software. EasiRun/SMART ist die ideale Plattform zur Realisierung solch einer Lösung.

Wie funktioniert EasiRun/SMART grundsätzlich?

Der zu verarbeitende Quellcode wird zunächst geparkt, analysiert und in eine vom Verarbeitungskern lesbare Form gebracht. Anschließend durchlaufen Regeln den aufbereiteten Quellcode. Die Reihenfolge der Ausführung wird dabei durch die Regeln selbst festgelegt und kann dabei auch durch den vorgefundenen Code beeinflusst werden. Die Regeln analysieren, modifizieren oder transformieren den Quellcode. Im Anschluss an die Verarbeitung werden die modifizierten Codestrukturen wieder in Textdateien exportiert. Code, der nicht durch

Die Vorteile auf einen Blick

- ✓ Individuelle und effiziente Entwicklung von Migrations- und Modernisierungslösungen
- ✓ Komplexe Transformationen nicht nur auf syntaktischer, sondern auch auf semantischer Ebene (Coderestrukturierung)
- ✓ Sehr kurze Code-Freeze-Zeiten
- ✓ Iterative Entwicklungsprozesse möglich
- ✓ Schnelles und effektives Bugfixing
- ✓ Exakte Reproduzierbarkeit der Ergebnisse unter gleichen Ausgangsbedingungen
- ✓ Einheitliche Umsetzung analoger Codekonstrukte (bessere Les- und Wartbarkeit)
- ✓ Geringerer Testaufwand
- ✓ Integrationsmöglichkeit in automatisierte Prozesse (z. B. Continuous Integration)

Transformationsregeln tangiert wurde, erscheint dabei genau in derselben Formatierung wie im Originalcode. Neu erstellte und modifizierte Codekonstrukte können mit Hilfe eines integrierten Formatters nahtlos in den vorhandenen Code eingepasst werden, so dass – sofern nicht speziell gekennzeichnet – der neue bzw. modifizierte Code nicht vom Originalcode unterschieden werden kann.

Worin liegen die Besonderheiten von EasiRun/SMART?

Im Gegensatz zu vielen einfachen Code-Konvertern können Modifikationen nicht nur auf syntaktischer sondern auch auf semantischer Ebene stattfinden. Damit sind auch sehr komplexe Codeanpassungen, -restrukturierungen und Transformationen möglich.

Warum mit EasiRun, warum mit EasiRun/SMART?

Wir sind überzeugt von unserem Ansatz und bekommen bestätigendes Feedback von unseren Kunden. Wir bauen Ihnen keine Luftschlösser, sondern sagen Ihnen ehrlich, wo wir Grenzen sehen. Wir versuchen nicht, Ihnen eine vorgefertigte Lösung zu verkaufen, sondern interessieren uns dafür, welche Absichten und welche Anforderungen an eine Modernisierungslösung Sie haben. Wir beraten Sie gern mit unserem Erfahrungsschatz aus anderen Modernisierungsprojekten. Wir fungieren gleichzeitig als Distributor, Anbieter, Know-how-Träger sowie Schulungs- und Supportpartner für eine Reihe von Produkten, die ggf. auch einige Ihrer Anforderungen abdecken können (COBOL-Compiler, Middleware-lösungen, etc.).

Welche Sprachen unterstützt EasiRun/SMART?

EasiRun/SMART verfügt über folgende Parser:

- IBM-COBOL (Mainframe-Dialekte)
- Dialog System Screensets
- RM/COBOL
- Zeilenbasierter Text
- Micro Focus COBOL (inkl. Object COBOL)
- MVS Job Control Language
- Bull GCOS COBOL
- VSE Job Control Language

Aufgrund einer sehr flexiblen Basistechnologie ist die Erstellung eines Parsers für andere Programmier- und Deklarationsprachen bei Bedarf innerhalb kurzer Zeit möglich.

Das Export-Format ist beliebig und unabhängig vom Format des geparsten Quellcodes, d.h., es können auch Codeformate exportiert werden, für die kein Parser verfügbar ist (z. B. Fujitsu NetCOBOL for .NET, Shell-Skripte usw.).

Anwendungsbereiche¹

In folgenden Anwendungsbereichen ist und war EasiRun/SMART unter anderem im Einsatz:

- Individuelle Analyse komplexer Anwendungen
- Modifikation von Syntaxkonstrukten bei Plattform- und Compilerwechseln
- Funktionsidentische Umstrukturierung und individuelle Modifikation von Quellcode
- Transformation von Quellcode in andere Sprachen
- Erstellung von individuellen Konvertern für Datenmigrationen
- Reformatierung von Code
- Instrumentierung von Code (Logging, Tracing, Monitoring)
- Erstellung von Testmodule

¹ Weitere Anwendungsbereiche erschließen sich in der Regel aus dem Kontext unserer Modernisierungsprojekte.

Was gibt es sonst noch über EasiRun/SMART zu sagen?

Die Entwicklung von Modernisierungslösungen mit EasiRun/SMART wird durch das eclipsebasierte EasiRun/SMART Developer Studio unterstützt, welches Entwicklung, Test, Management und Deploy der Regeln wesentlich erleichtert.

EasiRun/SMART ist parallelverarbeitungsfähig und kann somit die volle Leistung von Mehrkernprozessoren und Mehrprozessorsystemen nutzen.

So werden z.B. in einem Projekt ca. 1.800 COBOL-Programme inklusive der ca. 2.800 Copystrecken in deutlich weniger als 5 Minuten umgesetzt. Der dabei verwendete Regelsatz enthält ca. 90 Transformationsregeln mit zum Teil hoher Komplexität.



© Copyright 2021 EasiRun Europa GmbH. Alle Rechte vorbehalten. Microsoft, Windows und das Windows-Logo sind Marken der Microsoft Corporation. Alle anderen Produkt- und Firmennamen sind Marken der jeweiligen Inhaber.

Alle hierin enthaltenen Materialien sind als allgemeine Information gedacht und unterliegen Änderungen. Produktbeschreibungen sind in der entsprechenden technischen Dokumentation enthalten.

Informationen zu diesem sowie weiteren Produkten erhalten Sie auch unter <http://smart.easirun.de>