

## 19.0 POHLMANN, J.: Cobol-Compiler als effiziente Programmierwerkzeuge mit der Möglichkeit des Programmtransfers für Großrechner - Personal Computer

### Zusammenfassung

PC-Cobol Compiler stellen heute leistungsstarke Programmierwerkzeuge dar, die eine schnelle Programmentwicklung auf dem PC erlauben. Ihre Hilfsmittel dazu sind die bei fast allen Compilern mitgelieferten interaktiven Debugger, die das Quellenprogramm bei der Programmausführung am Bildschirm darstellen. Da Cobol als lange gebrauchte, höhere Programmiersprache eine hohe Standardisierung und Verbreitung auf Großrechenanlagen aufweist, bietet sich mit einigen der PC-Compiler neben der Möglichkeit einer dezentralen Programmentwicklung für die Großrechner auch die Übertragung von bestehenden Großrechnerprogrammen auf PC an. Allerdings ist dazu der im Programm verwendete Sprachumfang, die Leistungsfähigkeit des PC-Compilers, evtl. verwendete Datenbanksysteme und die Art der Programme (Batch- oder Dialogprogramme) zu beachten.

### Abstract

In the last years the efficiency of personal computers has increased continuously. There is no end to this development in sight. This is the reason why software companies have been able to offer more and more powerful cobol compilers. It's true that the first of those made us smile contemptuously, but the situation has changed today. Now the cobol compilers for personal computers are powerful software tools which easily bear the comparison with their large system colleagues. They are provided not only with the compiler itself but also with a mask generator, a program translating screen layout to translate into cobol statements and a debugger to develop efficient programs quickly. Their application not only lies in the decentral development of programs for large systems, but also in the transfer of large system programs to PCs.

The four most well-known cobol-compilers in this category are the ones produced by Realia, Microfocus, mhp and RM/Cobol-8x. Their functions are described below.

## 19.1 Einleitung

In den letzten Jahren ist die Leistungsfähigkeit von Personal Computern ständig gestiegen. Auch für die Zukunft ist ein Ende nicht abzusehen. Das ermöglichte den Softwarehäusern, immer leistungsfähigere Compiler für den PC anzubieten. Während die ersten Cobol-Compiler noch belächelt wurden, sind mittlerweile gegenüber Großrechnern ernstzunehmende Cobol-Programmierzweuge für den PC auf dem Markt. Diese besitzen außer dem eigentlichen Compiler weitere Programmierhilfen. Neben dem Editor zum Erstellen des Quellenprogrammes lassen sich mit dem Maskengenerator schnell funktionsfähige Bildschirmmasken in Cobol generieren. Der Debugger erleichtert mit verschiedenen Funktionen das Testen entwickelter Programme. Die heutigen Cobol-Compiler eignen sich daher sowohl für die Programmentwicklung auf dem PC als auch im Großrechnerbereich. Die vier verbreitetsten Cobol-Compiler dieser Kategorie sind der Mikrofocus, der Realia, der mhp und der RM/Cobol-8x. Sie werden im folgenden näher beschrieben, wobei mit den ersten beiden eingehende praktische Erfahrungen in der Programmentwicklung gewonnen wurden.

## 19.2 Beschreibung der PC-Cobol-Programmierzweuge

### 19.2.1 Sprache

Cobol (Common Business Language) ist als die heute am häufigsten verwendete höhere Programmiersprache weitgehend standardisiert (ANSI-Standard 68,74,80). Den jüngsten Standard, den ANSI 80 Standard, haben noch nicht alle Compiler realisiert. Für PC's bietet ihn allein der Realia-Compiler in Teilbereichen an. Die in diesem Standard neu aufgenommenen Sprachelemente tragen wesentlich zur heute üblichen strukturierten Programmierung bei und erleichtern somit eine übersichtliche und wartbare Programmentwicklung und deren Pflege.

Der PC-Cobol-Compiler von Realia besitzt beispielsweise Endbegrenzer für 16 wichtige Anweisungen. Zur Verdeutlichung ein Beispiel mit Inline-Perform Anweisungen und Endbegrenzer für IF-Bedingungen:

```
PERFORM VARYING KEY FROM 1 BY 1 UNTIL DATEI-ENDE
PERFORM READ-RECORD
IF KEIN-SATZ-GEF
    DISPLAY FEHLERMELDUNG
ELSE
    PERFORM SATZ-BEARBEITUNG
END-IF
DISPLAY 'OK'
END-PERFORM.
```

Für einen möglichen Programmtransfer zwischen IBM-Großrechner mit VS-Cobol Compiler und Personal Computer in beiden Richtungen ist er zudem sprachlich voll IBM-kompatibel. Das kommt besonders dadurch zum Ausdruck, daß er kein eigenes

Language-Reference Manual besitzt, sondern auf das entsprechende des IBM-VS-Cobol Compiler verweist.

Die meisten übrigen PC-Compiler wie der von Mikrofocus, mhp und Mikrosoft realisieren auf der sprachlichen Ebene den 74er Ansi-Standard.

### 19.2.2 Interaktive Testwerkzeuge (Debugger)

INTERAKTIVE TESTWERKZEUGE, SOGENANNTA DEBUGGER, ZÄHLEN HEUTE BEI PC-Cobol Compilern zu den festen Bestandteilen. Mit ihnen kann der Programmierer das Quellenprogramm während der Ausführung Anweisung für Anweisung am Bildschirm verfolgen. Verfügt das Programm über eine Bildschirmausgabe, wird diese automatisch eingeblendet. Über besondere Befehle läßt sich jederzeit zwischen der Programmiersicht (Quellenprogramm) und der Benutzersicht (Bildschirmmaske) hin und her schalten. Beim Realia-Compiler ist es zudem möglich, beim Vorhandensein von zwei Bildschirmen die Programmiersicht über den einen und die Benutzersicht über den anderen Schirm laufen zu lassen.

Die vorhandenen Debugger erlauben es, an beliebigen Punkten im Programm den momentanen Wert von Variablen anzuzeigen und nach evtl. Änderung wieder abzuspeichern. Dadurch lassen sich logische Fehler im Programmablauf sehr schnell lokalisieren und korrigieren. Das schrittweise Herantasten an die kritischen Stellen erfordert bei größeren Programmen allerdings sehr viel Zeit. Deshalb bieten die Debugger zusätzliche Befehle unterschiedlichster Art an, die entweder den Programmdurchlauf stufenlos beschleunigen oder einen schnellen Programmdurchlauf bis zu vom Benutzer markierten Programmstellen durchführen. Anschließend erfolgt der Programmtest der erreichten kritischen Stellen wiederum schrittweise. Neben der Änderung der Abarbeitung einzelner Cobol-Statements kann an beliebigen Punkten im Programm zu den Variablendefinitionen verzweigt bzw. einzelne Variablen direkt angewählt werden.

Debugger können die Programmentwicklungszeit, wie sich gezeigt hat, erheblich verringern, da häufig die einfachsten Fehler tagelanges Suchen erforderten. Ein weiterer Vorteil dieser Programmierwerkzeuge besteht darin, daß auch selbst größere und sehr verzweigte Programme durch die visuelle Programmablaufverfolgung viel schneller zu erfassen und zu warten sind.

### 19.2.3 Maskengeneratoren.

Besonders in der Dialogprogrammentwicklung kommt der Maskenerstellung bzw. -gestaltung erhebliche zeitliche und damit finanzielle Bedeutung zu.

Maskengeneratoren können zu einer am Bildschirm erstellten Maske automatisch die passenden Cobol-Anweisungen generieren. Mit ihrer Hilfe entwickelt der Programmierer schnell und sicher beliebige Bildschirmmasken und hat eine bequeme Möglichkeit, sie auch nachträglich zu verändern. Das fehlerträchtige, sonst übliche Auszählen einzelner Felder entfällt.

Der von Mikrofocus Professional Cobol-Compiler mitgelieferte Maskengenerator FORMS bietet die Erstellung einer Bildschirmmaske im Full-Screen Modus. Hierzu stehen nicht nur die von der Tastatur möglichen Zeichen zur Verfügung, sondern auch sämtliche ASCII-Zeichen. Sie lassen sich unten am Bildschirm einblenden und durch Cursor und Funktionstaste sofort in jede gewünschte Maskenposition übertragen. Nach der Festlegung der Maskenform werden die Ein- und Ausgabefelder durch die in Cobol üblichen Picture-Attribute erstellt, die bekanntlich i.G. zu anderen Sprachen über beachtliche Druckaufbereitungsmöglichkeiten verfügen. Damit entfällt für den Cobol-Programmierer die bei anderen Maskengeneratoren

teilweise notwendige Einarbeitung in Ersatzsymbole für Ein- und Ausgabefelder. Nach der Editierung erstellt der Maskengenerator den dazu passenden vollständigen Cobol-Code und speichert ihn in einer Datei ab, die sich durch den Copy-Befehl in jedes Programm einfügen läßt.

Von den vier Cobol Compilern Microsoft, Mikrofocus, mhp und Realia verfügen nur Mikrofocus und mhp über einen derartigen Maskengenerator.

#### 19.2.4 Editoren

Zwei der angebotenen Cobol-Compiler enthalten im Lieferumfang einen Editor. Dieser bietet teilweise für Editoren normale Funktionen an wie Einfügen, Vervielfachen, Suchen nach Wörtern etc., teilweise aber auch spezielle Programmierhilfen.

Mit dem Realia-Editor lassen sich Dateien beliebiger Größe in sehr schnellem Tempo überarbeiten. Allerdings ist seine Handhabung gewöhnungsbedürftig, da fast alle Funktionen über spezielle Tastenkombinationen gesteuert werden. Für die Programmierung positiv ist es, daß der Editor automatisch ein Journal führt, in dem er alle vorgenommenen Tastaturbewegungen speichert.

Bei unterlassenem Sichern der Datei oder Stromausfällen kann mit Hilfe dieses Journals die gesamte Datei automatisch nacheditiert werden und befindet sich dann auf dem letzten Aktualitätsstand.

Über zusätzliche Optionen kann der Editor unterschiedliche Dateiformate verarbeiten, also auch solche, die mit anderen Editoren erzeugt worden sind.

Der Mikrofocus Editor ist von der Handhabung für den Anfänger einfacher. Die nötigen Befehle werden jeweils am unteren Bildrand über alternative Einblendungen angezeigt.

#### 19.2.5 Compiler

Die meisten Compiler können mitsamt der mitgelieferten Zusatzprogramme durch Menutechnik aufgerufen werden. Die Standard-Optionen sind dabei eingeblendet. Häufig ist innerhalb des Auswahlmenues noch eine Dateiverwaltung untergebracht, die über Sortier- und Auswahlfunktionen das Suchen nach bestimmten Quellen- oder Objektdateien vereinfacht. Der Realia-Compiler speichert die zuletzt bearbeiteten Dateien und blendet sie bei erneutem Aufruf des Compilers als Vorschlag ein.

Neben dem Aufruf der Compiler spielen natürlich bei größeren Programmen auch die benötigten Compile-Zeiten eine Rolle. Wie sich schon bei Tests am Institut zeigte, war der Realia Compiler bei größeren Programmen dem Mikrofocus-Compiler merklich überlegen, während bei kleineren Programmen sich die beiden zeitlich die Waage hielten. Ein Cobol-Compiler Test der Zeitschrift PC-Welt bestätigte diese Erfahrung. Hier zeigte es sich, daß bei größeren Programmen (1220 Statements) am schnellsten der Realia und der RM/Cobol-8x compiliert, während mikrofocus um 50% und der mhp-compiler sogar um 250% langsamer war.

Die Programmschnelligkeit wurde mit dem "Sieb des Eratosthenes", der Ermittlung der Primzahlen zwischen 1 und 8192 mit zehnmaliger Wiederholung getestet. Neben den hervorragenden Laufzeiten von Realia und Mikrofocus war der mhp-Compiler um das 20-fache langsamer. Noch einmal doppelt so langsam wie der mhp ist durch seinen Interpretercode der RM/Cobol-8x.

Neben den Compilerzeiten und dem Laufzeitverhalten bei Hauptspeicher-residenten Programmen spielt in der Praxis bei Cobol besonders der index-sequentielle Dateizugriff eine Rolle. Hier wurde im Test der PC-Welt deutlich, daß der Realia und der RM/Cobol-8x Compiler die schnellsten beim index-sequentiellem Dateizugriff waren, während Mikrofocuss doppelt und mhp fünfmal länger für diesen Test benötigten.

### 19.3 Möglichkeiten des Programmtransfers Großrechner-PC

Die Frage des Programmtransfers zwischen Großrechner und PC stellt sich von zwei Seiten. Nachdem auf Grund der Hardwareentwicklung die meisten umfangreichen Programme bisher auf Großrechnenanlagen laufen, ist es einerseits bei vorhandenen leistungsfähigen PC's effizient, sie ohne erneutes Programmieren auf den PC zu übertragen. Andererseits kann es auf Grund der heute oftmals begrenzten Großrechnerkapazität sinnvoll sein, Programme teilweise oder vollständig auf dem PC zu entwickeln und anschließend auf den Großrechner zu übertragen. Das gleiche gilt für die Wartung und Pflege bestimmter Programmteile auf Großrechnern.

Um einen erfolgreichen Programmtransfer durchführen zu können, müssen allerdings verschiedene Voraussetzungen sowohl auf dem Großrechner wie auch auf dem PC erfüllt sein.

Zunächst sollte der Cobol Compiler des Großrechners wie auch der des PC's den in den Programmen benutzten Sprachumfang verarbeiten können.

Ist dies nicht der Fall, ergeben sich daraus meistens zeitaufwendige Umprogrammierungen bestehender Programme, die unter Umständen den anschließend geplanten Programmtransfer kaum noch lohnen. Liegen die Großrechner-Programme auf IBM Anlagen mit Cobol VS Compiler, so ist das Sprachproblem durch die Verwendung des sprachlich dazu voll kompatiblen PC-Realia Compilers gelöst.

Eine weitere Einschränkung auf der PC-Seite ist die bei allen bis auf den Realia Compiler auf 64 Kbyte begrenzte DATA DIVISION und PROCEDURE DIVISION, also dem Datendefinitions- und dem Anweisungsteil eines Cobol Programmes.

Mit dem Realia sind dagegen Programmübersetzungen im Data Division Teil bis 1 MB und in der Procedure Division bis 6,4 MB möglich. Dies erlaubt auch die Übertragung großer Programme ohne Schwierigkeiten.

Für den Programmtransfer ist weiterhin wichtig, ob es sich um Batch- oder Dialogprogramme handelt und ob mit herkömmlichen - sequentiellen oder index-sequentiellem - Dateien oder mit Datenbanken gearbeitet wird.

Batchprogramme mit Verarbeitung sequentieller oder index-sequentieller Dateien können beim Vorliegen obiger Voraussetzungen ohne große Probleme schnell transferiert werden.

Arbeiten Batchprogramme jedoch unter Zuhilfenahme von Datenbanksystemen, muß neben dem eigentlichen Programm auch dieses auf beiden Rechnern verfügbar sein. Möglichkeiten gibt es dazu über die relationalen Datenbanksysteme Oracle (PC und Großrechner) und DB2 (IBM-Großrechner).

Bei der Übertragung von Dialogprogrammen kommt als Schwierigkeit die beim Großrechner und PC unterschiedliche Bildschirm- und Tastatursteuerungen hinzu. Sie stellen nach bisherigen Erfahrungen neben der Datenbankproblematik das Haupthindernis einer schnellen Programmübertragung dar.